JOURNAL OF TELECOMMUNICATIONS AND INFORMATION TECHNOLOGY

4/2011

(Contents Continued on Back Cover)

Editorial Board

Editorial Advisory Board

ISSN 1509-4553

Circulation: 300 copies

Sowa - Druk na życzenie, www.sowadruk.pl, tel. 022 431-81-40

Recently in 2009 we have celebrated 75th anniversary of our predecessor National Institute of Telecommunication (PIT) created in 1934 in Warsaw. Sixty years before, in 1951, this Institute was divided into two separate entities: Industrial Institute of Telecommunication (PIT) and our Institute of Telecommunications (IŁ). In 2005 Institute of Telecommunication has get formal status of national research centre (IŁ PIB), we use English abbreviation of NIT (National Institute of Telecommunications). Sixty years of our separate activities and development was presented in details in No. 3–4/2009 of special edition of our Polish periodic journal "Telekomunikacja i Techniki Informacyjne" (TITI).

Regarding this 60th anniversary of our Institute of Telecommunications in 2011, let me underline some important facts having place in the last two years from previous celebrate anniversary. In October 2010 our Institute has reached high category A for the research institution, certified by the Ministry of Science and High Education. I am really satisfied that this score we have got in 6th place in the group of 47 research institutions evaluated for the period of previous 5 years. As our new achievements we can also count gold medals and honorable mentions for our equipment awarded in international innovation fairs in Warsaw, Bruxelles, Seoul and Singapour.

In recent two years we significantly have developed our co-operation with outstanding research centres in Poland and abroad. We have sign up agreements with Faculty of Electronics and Information Technology (WEiTI) of Warsaw University of Technology and with Electronics Faculty of Military University of Technology (WAT) in Warsaw. Co-operation with WEiTI is finalized with common application for 7th FP project, and in the framework of the agreement with WAT we have admitted for training the group of 26 students and graduates. In 2011 we have also sign up the agreement with Korean Institute of Information Technology KETI. In 2010 we have send to the ITU in Geneva two reports from our research works conducted in NIT (pilot tests of DAB+ system and ground conductivity map of Poland). By the occasion of Plenipotentiary Conference of ITU in Guadalajara in Mexico we have issued special edition of our "Journal of Telecommunications and Information Technology" (JTIT), you can find there detail information on our co-operation with ITU and other organizations. In collaboration with partners we have organized important international conferences ICTON in Munich (2010) and Stockholm (2011), and also DSTIS conference in our headquarter in Warsaw (2011). In connection with our 60th anniversary we are going to organize important national 22nd Symposium on Telecommunication and Information Technology KSTiT'2012, which will take place in NIT in Warsaw (12–14 of September, 2012), and from many years is important conference and place of meeting of Polish researchers.

Our important activities in the last two years were devoted to the new research projects. From 2008 till 2011 we have almost duplicated the number of realized projects (from 11 to 20). In this period we have finalized an important project Polish National Project PBZ devoted to the new generation services and networks, conducted together with almost all main technical universities in Poland, and we have started new projects on Internet of the Future IIP, Information System on BroadBand Infrastructure and BB Poland Portal SIPS, Information System on the Protection Against the Threats ISOK, Mobile Testing Laboratory MLB, Information System for Radio Networks Planning PIAST and others.

We have significantly increased the value of realized research projects, however still our finance results are not satisfying us. This was the main reason of our restructuring activities started in order of better use of our human and material resources: we have created a new Department of Internet Architecture and Applications, we have canceled and re-organized some of supporting divisions, we have negotiated with our social partners and implemented new Collective Work Agreement and new work regulations. We also implement new legal framework for science sector in Poland: we have modified our Statute, in June 2011 we have elected our new Scientific Council and first time in our history important personalities from telecommunication companies in Poland have been included in this Council, we have also elected a disciplinary spokesman and Disciplinary Commission.

On the beginning of 2012 and the next years we are before next challenges of further restructuring, taking up new projects and creation of new research teams. Let me congratulate all our workers, partners, Authors and Readers of JTIT, on previous successful years of common work and achieved results. I wish you, in next decades of scientific and research operations in our National Institute of Telecommunications, a great personal and outstanding professional satisfaction from your work.

Wojciech Hałka
Director of NIT

JOURNAL OF TELECOMMUNICATIONS AND INFORMATION TECHNOLOGY

Preface

This issue of the *Journal of Telecommunications and Information Technology* contains two groups of articles. The first ten papers, **edited by Jarosław Arabas as Guest Editor**, are devoted to various aspects of applying Evolutionary Computation (EC) to global optimization, while the next seven cover various issues related to communication networks and their security, plus somewhat more remote subjects of intelligent agents and semiconductor memory devices.

Evolutionary Computation is a family of methods which are inspired by findings in fields of genetics and evolution. EC methods are quite popular in solving optimal design tasks of various kind, they can be also met in business intelligence applications, in the decision support systems and in many other fields where the task is to find an approximation to the optimal solution. A natural field of EC applications are optimization tasks where the solution cannot be represented in a standard way, e.g., as a vector of real or integer parameters, but a nonstandard or structured representation is needed, e.g., a tree of a variable size.

The first paper *Evolutionary Computation for Global Optimization – Current Trends* by Jarosław Arabas briefly introduces the idea of the EC. The paper also overviews development lines that can be observed in the literature on the EC, with a particular stress on their applications in global optimization. Then, three articles follow which report on progress on development of various EC methods.

Andrzej Obuchowicz and Przemysław Prętki in the paper *Self-Adaptive Stable Mutation Based on Discrete Spectral Measure for Evolutionary Algorithms* consider the mutation operator that uses α-stable mutations. They introduce an approach which allows to adaptively tune the mutation parameters to improve the efficiency of the resulting global optimization method based on an algorithm from the EC family.

Krzysztof Trojanowski, Mikołaj Raciborski and Piotr Kaczyński present a *Self-Adaptive Differential Evolution with Hybrid Rules of Perturbation for Dynamic Optimization*. They consider optimization tasks when the objective function undergoes changes and focus on the ability of the differential evolution algorithm to follow the changing location of the optimum. They mix the mutation mechanism specific for the differential evolution, which is based on using a difference between two points from the current population, with the "traditional" mutation that uses a normal variate. They also consider using an α-stable distribution instead of the normal, and they report on the efficiency increase thanks to the introduced mechanisms.

The article *Improving Population-Based Algorithms with Fitness Deterioration* by Adrian Wolny and Robert Schaefer is devoted to the issue of the multimodal optimization when the task is to locate as many local optima as possible. To obtain this goal they apply a technique that gradually discourages an EC based optimization method from revisiting local optima that have been discovered already. They perform deformations of the original objective function by adding penalty terms which are defined in a way to approximate shapes of attraction basins of local optima.

Next articles present applications of EC methods. Przemysław Miazga considers *Topological Synthesis of Tree Shaped Structures Based on a Building Blocks Hypothesis*. He summarizes result of applying an EC method to the design of microwave circuits fabricated as a certain shape of metallization on a laminate plate. The design consists in planning the metallization pattern which can be a quite complicated tree-form shape. Thus it is advisable to perform the optimization process directly on the tree representation of the metallization pattern. Przemysław Miazga presents results of experiments that verify effectiveness of the presented approach.

Maciej Michalak and Robert Nowak report on an *Evolutionary Algorithm that Designs the DNA Synthesis Procedure*. Synthesis of long DNA strands uses a natural tendency of DNA to form particles made of two strands which are kept together by the complementarity bonds. To facilitate the synthesis one has to split the desired long DNA strand and its complementary strand into pieces such that pieces of the desired and complementary strand mutually overlap. Then a stepwise procedure to mix these strands in one solution is applied. Maciej Michalak and Robert Nowak use an EC method to suggest a proper division of a desired long DNA strand and to plan the procedure of mixing the resulting fragments.

Ewa Niewiadomska-Szynkiewicz, Michał Marks and Mariusz Kamola discuss the results of the *Localization in Wireless Sensor Networks Using Heuristic Optimization Techniques*. They experimentally compare the results obtained by two different EC methods, by the simulated annealing and by a problem-specific heuristic method.

Paweł Zawistowski and Jarosław Arabas consider the black-box approximation problem where the goal is to create a regression model using only empirical data without incorporating knowledge about the character of nonlinearity of the approximated function. Their paper *Incrementally Solving Nonlinear Regression Tasks Using IBHM Algorithm* presents a method which is used to build a model being a weighted sum of components being monotonic and bounded functions, each characterized by a set of parameters. They use an EC method to tune the parameters of each component function to maximize correlation between them and the approximation error function.

The paper *Benchmarking Procedures for Continuous Optimization Algorithms* by Karol Opara and Jarosław Arabas is devoted to experimental procedures that accompany the development of global optimization methods, and the EC in particular. It has become a common practice to use a set of widely recognized test functions to demonstrate the quality of optimization methods. The paper overviews two popular benchmark sets and the standardized test methodology that accompanies them. The authors comment on problems with aggregating the results obtained for test functions contained in the benchmark set to judge on the relative quality of optimization methods.

The first part of this volume is concluded with the paper *The "Second Derivative" of a Non-Differentiable Function and its Use in Interval Optimization Methods* by Bartłomiej J. Kubica who contributes to optimization methods which use interval calculus. The approach has no direct relation with the EC so far but it seems to be a promising optimization tool. The author introduces a concept of a second derivative definition in the interval calculus and provides examples showing that this significantly fastens the optimization process.

Then follows the network-and technologies-related part. Pankaj Kumar Sharma, R. K. Nagaria, and T. N. Sharma in the paper *Enhancement of Power Efficiency in OFDM System by SLM with Predistortion Technique* report how a selected mapping (SLM) and proper predistortion can mitigate the problem of high peak to average power ratio characteristic of OFDM radio networks, resulting in more efficient utilization of power amplifier, lower error ratio and reduced amplifier cost.

The next article, *Providing QoS Guarantees in Broadband Ad Hoc Networks* by Marek Natkaniec *et al*. is also about radio networks, presenting a novel QoS architecture for IEEE 802.11 multihop broadband ad hoc networks integrated with infrastructure. The authors are convinced that the proposed solution will provide QoS support for a variety of services in

future mobile ad hoc networks, providing scalability and traffic differentiation absent in other solutions previously reported in literature.

The last work related to radio networks include here is *SVD Audio Watermarking: A Tool to Enhance the Security of Image Transmission over ZigBee Networks* by Mohsen A. M. El-Bendary *et al*. This paper discusses audio watermarking as a tool to improve the security of image communication over the IEEE 802.15.4 ZigBee network, with the aim being improved resistance of encrypted watermark to attacks and degradation (fading, noise etc.) experienced in the ZigBee networks.

Baibaswata Mohapatra, Rajendra K. Nagaria and Sudarshan Tiwari in their paper *Guaranteed Protection in Survivable WDM Mesh Networks – New ILP Formulations for Link Protection and Path Protection* deal with optimization of survivable optical fiber WDM networks, presenting new, single integer linear programs (ILPs) for finding routing and wavelength assignment with best utilization of network resources (wavelength channels). The solution proposed is superior to previously reported schemes in both network wavelength utilization and computation time required.

Among several biometric access control techniques, iris recognition is highly rated. However, once the reference data are stolen, the user cannot change his biometric features in contrast to a compromised password. Therefore, properly designed security system must perform iris features matching without revealing these features and the reference template. A suitable authentication protocol is described by Przemysław Strzelczyk in his paper *Privacy Preserving and Secure Iris-Based Biometric Authentication for Computer Networks*.

Ahmed M. Elmahalawy, Moukhtar A. Ali and Hany M. Harb in their contribution *Trends and Differences of Applying Intelligence to an Agent* analyze development trends of intelligent agent solutions, with the goal of mimicking human intelligence, and the differences between two main types of intelligence that can be applied to agent: accumulative (the agent accumulates information about environment before decision is made) and dynamic. In addition, detailed terminology of accumulative intelligence is introduced.

The last paper *A Comparative Study of Single- and Dual-Threshold Voltage SRAM Cells* is very different, being devoted to hardware design of semiconductor memory devices. Pragya Kushwaha and Amit Chaudhry analyze performance trade-offs of 5-, 6- and 7-transistor SRAM cells made in 65 nm and 180 nm CMOS technologies, taking into account write time, read time and device power consumption. While the shift to 65 nm technology cuts power consumption by 80–90%, device speed is reduced to variable extent.

Paweł Szczepański
Editor-in Chief

Paper

Evolutionary Computation for Global Optimization – Current Trends

Jarosław Arabas

Institute of Electronic Systems, Warsaw University of Technology, Warsaw, Poland

Abstract—This article comments on the development of Evolutionary Computation (EC) in the field of global optimization. A brief overview of EC fundamentals is provided together with the discussion of issues of parameter settings and adaptation, advances in the development of theory, new ideas emerging in the EC field and growing availability of massively parallel machines.

Keywords—evolutionary computation, global optimization.

1. Evolution of Evolutionary Computation

Since 1980'ies we have been witnessing a growing popularity of a family of algorithmic techniques which originated in 1960'ies and have been inspired by findings in fields of genetics and natural evolution. Although many pioneering approaches had been introduced (see [1] for an overview), only several survived and are nowadays called the Evolutionary Computation (EC) [2]. Until mid 1990'ies, the mainstream EC work was presented at three major conferences: International Conference on Genetic Algorithms (ICGA), Parallel Problem Solving from Nature (PPSN) and Workshop on Foundations of Genetic Algorithms (FOGA). In 1994 the IEEE Congress on Evolutionary Computation (CEC) was started, in 1995 the Genetic Programming (GP) conference was launched, and the year 1999 witnessed the birth of the annual Genetic and Evolutionary Computation Conference (GECCO) which combines the ICGA and the GP in one event. Two recognized international journals publish works on the EC: since 1993, the MIT Press has been releasing the *Evolutionary Computation* journal and the IEEE has been publishing the *Transactions on Evolutionary Computation* since 1997. The Polish accent is the annual National Conference on Evolutionary Computation and Global Optimization which started in 1996[1].

The idea behind the EC is quite straightforward – take a population of points from some search space, assign them numbers that reflect their probability to survive the selection process, and perform a randomized selection. The selected points undergo a randomized variation, yielding a new population of points, and the process is iterated many times. Despite of the simplicity of the idea, several named approaches have been defined that usually differ only in small details. A newcomer to the EC field may be greatly surprised by recognizing that it includes:

- Differential Evolution,
- Evolution Strategies,
- Evolutionary Programming
- Genetic Algorithms,
- Genetic Programming,
- Memetic Algorithms,

to mention only few important branches in an alphabetical order.

When studying EC methods one has to get used to the metaphor of genetics and evolution, which strongly influenced the vocabulary. Instead of points from the search space we speak of *chromosomes* (or *individuals*), instead of the objective function to be optimized we speak of the *fitness function* which defines the selection probability, variation of points is performed by the *genetic operations* which are called *crossover* and *mutation*, etc.

A popular opinion about EC is that an important factor that attracted researches to take a closer look at the EC is the appealing metaphor. Perhaps this observation motivated researchers to look for other metaphors from the nature, and since late 1990'ies we have been observing a tendency to introduce various population based techniques which share the idea of selection and variation, but they are named (in an alphabetical order):

- Artificial Immune Systems [3], [4],
- Estimation of Distribution Algorithms [5],
- Particle Swarm Optimization [6],

to mention a few representatives. A common name of *metaheuristics* has been suggested for the EC and the aforementioned techniques to avoid a naming burden [7], [8].

In this article it is attempted to comment on the current state of the EC, which is a very hard task and will be always more or less subjective. Therefore the bibliography will be presented that provides more detailed descriptions of ideas that have been roughly sketched in the text. Much more detailed general presentation of EC methods can be found *inter alia* in [2], [9], [10], [11].

The paper is composed of four sections. The first section briefly comments on the history of EC development. Section 2 overviews the taxonomy of optimization tasks that

[1] In year 2011 the conference was organized in Warsaw by the Warsaw University of Technology and Cardinal Stefan Wyszynski University.

are considered along with the EC and outlines the algorithm of a typical EC method for global optimization. In Section 3 it is attempted to define main lines of the EC development for global optimization that can be found in the literature. Section 4 presents concluding remarks and outlooks possible future development trends.

2. Evolutionary Computation and Optimization

2.1. Taxonomy of Optimization Tasks

Current applications of the EC are usually related to optimization of various kind. With D let us denote the domain of a problem to be solved (i.e., the set of all possible representations of the solution). The domain contains points that are feasible, i.e., any infeasible point is definitely not a solution. They can be evaluated using the objective function $q : F \to R$, where $F \subseteq D$ is the set of feasible solutions. Then the optimization task is defined as the task to find either the global minimum or any local minimum of the objective function. Depending on the domain type we can distinguish:

- combinatorial optimization when the domain is countable or finite, e.g., D is the set of binary vectors, the set of permutations or the set of graphs,
- continuous optimization when the domain is R^n.

If $F \neq D$ then the optimization task is a constrained one. Some researchers consider the multimodal optimization where in addition to the global optimum, the solver is expected to find as many local optima as possible. For an overview of optimization problems and methods with a particular stress on the global optimization the reader is referred to [12]–[14].

Another generalization of the optimization task which has gained a growing attention for past 10 years in the EC community is the multiobjective optimization where, instead the objective function, a mapping is considered $\mathbf{q} : F \to R^m$. Then the task is to find nondominated points, where a point $\mathbf{x} \in F$ is called nondominated when there exists no other point $\mathbf{y} \in F$ such that for all $i = 1, ..., m \quad q_i(\mathbf{y}) \leq q_i(\mathbf{x})$ and for some j it holds $q_j(\mathbf{y}) < q_j(\mathbf{x})$.

From the historical perspective, early EC methods have not been designed as optimization tools (cf. a famous statement "Genetic algorithms are not function optimizers" by Kenneth de Jong [15]). They have been viewed in terms of adaptation which is less strict and formalized than optimization. Moreover it is even not necessary to explicitly define any objective function, e.g., in the tournament selection, it is only needed to compare two points to choose the better one. Still it has become a common practice to use EC methods as optimization tools. This practice has been criticized by Wolpert and Macready in their famous No Free Lunch Theorem (NFL) for optimization [16]. They claimed that no search method would perform consistently better than any other if one considers all objective functions which can be defined in the search space. Publication of NFL was initially somewhat shocking for researchers working on the EC development since its naive interpretation led to the conclusion that no real development is possible. This interpretation of NFL was then criticized by authors who showed that although NFL is correct, it does not necessary mean that all algorithms are equally good for subsets of problems defined in the search space [17].

EC has a strong relationship to artificial intelligence (AI). Maybe the most popular application of EC methods in AI is the Neural Network (NN) training process. The link between the EC and NNs is so strong that it gave an inspiration to establish in 1993 the International Conference on Genetic Algorithms and Neural Networks (ICANNGA). If we take a perspective that the optimization task consists in learning the global optimum using the experience gathered from previously generated points, then we can agree that EC can be classified as an AI method [18].

Stress on optimization properties of the EC has been increasing during their development. It seems that this is a natural consequence of the tendency to apply EC methods in practice. It is also possible to theoretically analyze behavior of an EC method using the same terms as introduced for well-established optimization tools. Thus the theoretical analysis of EC methods often concentrates on the convergence in a weak sense [19], [20].

Development of EC methods as optimization tools has been facilitated by benchmark functions; some of them have been introduced even in mid-1970'ies [21]. Until very recently there was however no clear agreement about the testing procedure, which made the published results hardly comparable. Few years ago there have been two benchmark sets introduced which define a standardized testing procedure and criteria for evaluating optimization methods based on statistics of results from multiple independent runs of each method [22], [23]. It should be however noted that, although the benchmarking process is usually limited to search spaces containing either binary or real vectors, there are many EC applications where specific nonstandard representations are processed [11]. It seems that the ability to process such nonstandard representations is one of major advantages of EC.

2.2. Evolutionary Algorithm for Global Optimization

A typical Evolutionary Algorithm (EA) is depicted in Fig. 1. State of the algorithm in the iteration number t is defined by the population P^t which contains μ points; the i-th point is labeled with P_i^t. In each iteration a population O^t is created by mutation and crossover of points selected from P^t. Population P^{t+1} is defined by a replacement procedure that either accepts points from O^t only or allows to pass some points from P^t. Selection of points is a random process which favors better ones, i.e., points with a lower objective function value (in the minimization case) will be selected with a higher probability than others. In the replacement phase the new population is defined by

selecting points from O^t and P^t. In generational EAs, the replacement is defined simply as $P^{t+1} \leftarrow O^t$ and in elitist EAs a number of best points from P^t can be preserved.
When the EA is to solve the global optimization problem, typical mutation is performed as $O_i^t \leftarrow P_k^t + \mathbf{z}$ where $\mathbf{z}$ is a random variate which is normally distributed with zero mean vector and a covariance matrix $\mathbf{C}$. Typical crossover consists in averaging points which results in the following formula for the "mutation and crossover" operation

$$O_j^t \leftarrow \left(\mathbf{w} \cdot P_k^t + (\mathbf{1} - \mathbf{w}) \cdot P_l^t\right) + \mathbf{z},$$

where $\mathbf{w}$ is a vector of weights such that $0 \leq w_j \leq 1$ for all j, and the symbol '$\cdot$' stands for the component-wise product such that $\mathbf{a} \cdot \mathbf{b}$ yields a vector $\mathbf{c}$ and $c_j = a_j b_j$. When $w_j = 1/2$ we speak of the arithmetic crossover. When w_j is a random variate with the Bernoulli distribution we speak of the binomial crossover, and when the probabilities of getting 0 and 1 are equal we speak of the uniform crossover. There are no good indications about the stop criterion since there are no theoretical findings about the EA convergence speed that can be applied for sufficiently general classes of problems (but there exists the convergence speed analysis for the parabola function provided e.g. in [24], [25]). For this reason the stop criterion is usually based on the time budget.

```
t ← 0
P^0 ← initialization()
repeat
  for all i = 1,...,μ do
    if U(0,1) < p_c then
      k,l ← select from (1,...,μ)
      O_i^t ← mutation and crossover(P_k^t, P_l^t)
    else
      k ← select from (1,...,μ)
      O_i^t ← mutation(P_k^t)
    end if
  end for
  P^{t+1} ← replacement(P^t, O^t)
  t ← t + 1
until stop condition satisfied
```

Fig. 1. Outline of an example evolutionary algorithm for global optimization.

The EA is a random procedure, therefore its action can be described in terms of the probability theory and statistics. For binary encoded EAs there is a well developed and widely accepted theory which analyzes populations P^t as a Markov chain [26]. It is possible to analyze statistics of populations, e.g., to check if the probability of hitting the global optimum at least once increases with the generation index or to test if the most frequent population contents will contain the global optimum. Another approach to analyze binary encoded EAs is presented by the schema analysis which analyzes schemata – sets of solutions defined by similarity patterns. It is argued that schemata with over-average fitness are expected to increase their number of representatives in subsequent populations.
In real coded EAs which are used to perform global optimization it has been proved that weak convergence will be observed if it is possible with a nonzero probability that the EA will generate a finite series of points starting from any feasible point and terminating in a neighborhood of any other feasible point, provided that the neighborhood is a nonzero measure set [19]. This result, although important, does not give any information about the dynamics of the population contents. This can be achieved by applying an analysis of the population distribution dynamics that assumes an infinite population model introduced by Qi and Palmieri [27]. This model has been revisited by Karcz-Duleba [28], [29] and Arabas [30] who derived formulas for the limiting values of the population variance in the search space for typical selection methods, Gaussian mutation, arithmetic crossover and elite factor.

3. Selected Topics in Development of EC Methods for Global Optimization

Various mutation types. Gaussian mutation is a very popular choice when using the EA for global optimization. The normal distribution is however "thin tailed" and computer realizations of the normal random variable usually cannot generate points located further than, say, 5 times the standard deviation from its mean. This fact motivated researchers to look for other mutation schemes and in the results several alternative definitions have been introduced, including α-stable mutations and Differential Evolution.

The α-stable distribution has a property that its probability density function in the tail can be approximated by a function proportional to $\exp(-x^\alpha)$. It is stable in a sense that the sum of α-stable distributed variates is also α-stable distributed. Value of $\alpha = 2$ corresponds to the normal distribution. When $\alpha < 2$ the distribution becomes "fat tailed" which means that when α decreases it will be more and more probable to generate a variate from the tail, which may significantly increase the population diversity. A detailed discussion of α-stable mutation is provided in [31].

Differential Evolution (DE) is sometimes classified as a separate metaheuristic type but the type of variation allows to consider it as an approach to the mutation adaptation. The most innovative idea introduced in DE is to perform mutation according to the formula

$$O_i^t = P_j^t + F\left(P_k^t - P_l^t\right),$$

where P_j^t is the mutated point, F is a parameter called the scaling factor and k and l are indices of points from P^t that have been selected with the uniform distribution in $\{1,...,\mu\}$. Thus the distribution of mutants depends on the distribution of population P^t, which allows for its adaptation to the fitness function shape. DE differs also in the way of organizing relations between crossover and mutation. More details on DE can be found in [32].

Compact EAs and Estimation of Distribution Algorithms. In a classical EA the algorithm state is defined by the population contents, and the EA action can be described by the sampling distribution – the probability distribution that describes the process of generating one point from the population O^t. In compact EAs, as well as in Estimation of Distribution Algorithms (EDAs), the sampling distribution takes the role of the state definition and defines in the same time the process of generating points. In the compact EA in each iteration a small number of points (usually one or two) are generated and the sampling distribution is updated. In the EDA a number of points is generated and the new sampling distribution is estimated from them.

The most successful representative of the EDA line of algorithms is the Covariance Matrix Adaptation Evolution Strategy (CMAES) [33]. According to results of benchmarking on BBOB'09 and CEC'05 [22], [34], CMAES seems to be one of the most successful methods in EC. In CMAES the sampling distribution is normal and it is represented by the mean vector $\mathbf{m}$ and the covariance matrix C. In the iteration t the sampling distribution is used to generate the population P^t. Then a population $e(P^t)$ being a fraction of best points from P^t is selected which is used to compute the mean vector $\mathbf{m}' = E(e(P^t))$ and the matrix $C' = E((P^t - \mathbf{m}')^2)$. Values of $\mathbf{m}'$ and C' are used to update the parameters $\mathbf{m}$ and C for the next iteration.

Tuning parameter values. Practitioners which are potential users of optimization methods usually dream of a single magic button to press and get the problem solved. Therefore the chance that an optimization method will become an acceptable tool for practitioners increases when the method assumes smaller number of parameters. Unfortunately, basic EAs are characterized by many parameters with no clear intuition about their relation with efficiency. For example in the standard EA presented in Fig. 1 the user has to set the values of the population size, the covariance matrix of the Gaussian mutation, the crossover probability and the elite fraction. To reduce the set of user-defined parameters, diversity of adaptation techniques have been developed – see e.g. [35], [36] for an overview. Note that the DE and CMAES fit into the line of adaptive methods to tune the mutation distribution.

Among currently popular approaches to parameter tuning one can distinguish two dominating types. The first one is self-adaptation, where each point is coupled with the algorithm's parameter values. During mutation, the parameter values are also mutated. Is is believed that appropriate parameter values will be more frequently accompanying points that are better than average and therefore they will be reinforced by the selection process. Another popular approach is based on an ensemble of a finite number of parameter settings. Every settings is evaluated by looking at the average increase of quality of points before and after applying transformations defined by the parameter settings. Choice of the parameter settings is randomized and is influenced by their cumulated evaluation – values of parameters that are on the average better than others are more likely to be chosen to perform the transformation.

Hierarchy of metaheuristics. As mentioned in the introductory section there is a variety of approaches in domains of the EC and metaheuristics. Therefore a tendency has appeared to consider hybrid techniques of various kind. Generally speaking, the hybridization may be either generic or hierarchical. In the first case, elements from one metaheuristic may change the algorithm of the other. A good example is to introduce to the standard DE a self-adaptation process, which was originally introduced for Evolution Strategies. In the second case, one metaheuristic method is used to control the other, e.g., by tuning its parameters (metaoptimization [37]).

Parallelization. In every iteration the EA processes many points from the population P^t to define the population P^{t+1}. Observe that the process of selection, crossover and mutation can be done separately for each point from the population O^t. Therefore there is a long tradition of EA parallelization. Until very recently the parallelization was limited by the hardware, since massively parallel computers were expensive and relatively hard to access. Now we are witnessing a big change thanks to the introduction of General Purpose Graphic Processing Units (GPUs). GPUs are multicore processors with typical number of cores of the order of hundreds. It is possible to program them under a number of higher-level programming languages, and the most popular toolkits that allow for this are CUDA and openCL (see e.g. a comparison of performance in [38]).
A serious limitation of GPUs is that the cores must be run in a Single Instruction Multiple Data regime. If the problem definition allows to work in this mode then the user can count on very interesting speedup values of several tens up to several hundreds when comparing the total execution time of a program working with and without the GPU card. A convenient software platform to start from is called EASEA and has been maintained by the team from the Strasbourg University [39]. The platform uses CUDA and it allows the user to concentrate on programming the problem definition since many standard EA version have been preprogrammed as EASEA templates. The programming language is a dialect of `C++`. An alternative attempt to the use of openCL to implement an EA on GPU can be found e.g. in [40].

4. Concluding Remarks

Analysis of the dynamics of the EA population reveals two interleaving phases: the quasi-stability of the sampling distribution, when the population oscillates in an area of the domain for many iterations, and the population drift when the population changes its location in a more systematic way, which takes only few generations. When looking at the dynamics of the EA development, a similar pattern can be recognized. For this reason it is quite easily predictable that the current trends of development will remain current

for several years, but it seems also quite possible that it will be not much to win by continuing them.

In the author's personal opinion, what really lacks in EA is a tighter link between theory and practice. It seems that the bottleneck is definitely on the side of theory. Theoreticians are used to consider optimization problems where the stress is on the convergence. This kind of analysis may imply that the EA should possess some element of a "pure chance" that will allow to reach any nonempty neighborhood of every point by transformation of any point from the domain. In the light of this criterion the EA is a method which is worse than the random walk or the uniform sampling since for these methods the probability of such an event is highest! Similarly, when looking at the rate of progress obtained for the parabola function, EAs cannot compete against pseudo-Newton methods.

It seems that the focus of the analysis should be changed. One hopeful direction seems to consider the ability of the population to escape from the neighborhood of a local optimum [41]. This ability relates to the maintenance of the diversity level. Only few works on this issue can be found in the literature (see [30] for an overview). The other possible direction is to observe that EA adapts the populations' location such that better points are more likely to appear. This does not necessary imply that the most interesting area for an EA is the neighborhood of the global optimum, since it may too "narrow" in comparison to the populations' diversity. This effect is sometimes called the "survival of the flattest".

It should be also stressed that if an EA is used as a tool to solve e.g., some engineering problem, it uses an objective function which is the problem model. For this reason, even if the global optimization has succeeded, this means that the problem model, rather the problem itself, has been really solved. There is a need for systematic treatment of such situations.

It seems that the hot topic for the next few years will become the use of massively parallel computing offered by programmable GPUs. The ability to evaluate in parallel a huge number of points may shift the interest from sophisticated methods, which use a small number of points but need a large number of iterations (a typical situation for adaptive EAs), towards simpler methods that can effectively use massive parallelism. Perhaps this will change the criteria of benchmarking various EA methods, since the number of generations will influence the execution time rather than the number of the objective function evaluations.

Acknowledgements

The author is grateful to Prof. K. Trojanowski for comments on the first version of the article.

References

[1] *Evolutionary Computation: The Fossil Record*, D. Fogel, Ed. IEEE Press, 1998.

[2] K. de Jong, *Evolutionary Computation*. MIT Press, 2002.

[3] L. N. de Castro and J. Timmis, *Artificial Immune Systems: A New Computational Intelligence Approach*. Springer, 2002.

[4] S. Wierzchoń, *Sztuczne Systemy Immunologiczne. Teoria i Zastosowania*. EXIT, Warszawa 2001 (in Polish).

[5] *Estimation of Distribution Algorithms: a New Tool for Evolutionary Computation*, P. Larranaga and J. Lozano, Eds. Kluwer, 2002.

[6] M. Clerc, *Particle Swarm Optimization*. Wiley, 2006.

[7] S. Luke, "Essentials of metaheuristics", 2009 [Online]. Available: http://cs.gmu.edu/∼sean/book/metaheuristics/

[8] K. Trojanowski, *Metaheurystyki Praktycznie*. 2nd ed. Wydawnictwo WIT, 2008 (in Polish).

[9] J. Arabas, *Wykłady z Algorytmów Ewolucyjnych*. Warszawa: WNT, 2004 (in Polish).

[10] A. E. Eiben and J. E. Smith, *Introduction to Evolutionary Computing*. 2nd ed. Springer, 2007.

[11] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*. Springer, 1998 (3rd ed.)

[12] R. Horst and P. M. Pardalos, *Handbook of Global Optimization*. Kluwer, 1995.

[13] P. M. Pardalos and H. E. Romeijn, *Handbook of Global Optimization, Volume 2*. Kluwer, 2002.

[14] A. Torn and A. Zilinskas, *Global Optimization*. Springer, 1989.

[15] K. A. de Jong, "Genetic algorithms are not function optimizers", in *Proc. Worksh. Found. Genetic Algorithms FOGA 1992*, Vail, Colorado, USA, 1992.

[16] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization", *IEEE Trans. Evol. Comput.*, vol. 1, pp. 67–82, 1997.

[17] T. M. English, "Optimization is easy and learning is hard in the typical function", *Proc. Congr. Evol. Comp.*, La Jolla, CA, USA, 200, pp. 924-931.

[18] J. Arabas and P. Cichosz, "Searching for intelligent behavior", in *Proc. 16th Int. Conf. Intel. Inform. Sys. IIS 2008*, Zakopane, Poland, pp. 3–22.

[19] G. Rudolph, "Convergence of evolutionary algorithms in general search spaces", in *Proc. Third IEEE Conf. Evol. Comp. CEC 1996*, Nagoya, Japan, 1996, pp. 50–54.

[20] R. Schaefer, *Foundations of Global Genetic Optimization*. Springer, 2007.

[21] L. Dixon and G. Szegö, *Towards Global Optimization*. North- Holland, 1975.

[22] N. Hansen, A. Auger, R. Ros, S. Finck, P. Posik, "Comparing results of 31 algorithms from the black-box optimization benchmarking BBOB-2009", in *Proc. GECCO Genetic and Evol. Comput. Conf.*, Portland, Oregon, USA, 2010.

[23] P. N. Suganthan, N. Hansen, J. J. Liang, K. Deb, Y.-P. Chen, A. Auger and S. Tiwari, "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization", Tech. Rep., Nanyang Technological University, Singapore, 2005.

[24] H.-G. Beyer, *The Theory of Evolution Strategies*. Springer, 2001.

[25] G. Rudolph, "Local convergence rates of simple evolutionary algorithms with Cauchy mutations", *IEEE Trans. Evol. Comput.*, vol. 1, no. 4, pp. 249–258, 1997.

[26] M. Vose, *The Simple Genetic Algorithm: Foundations and Theory*. MIT Press, 1999.

[27] X. Qi and F. Palmieri, "Theoretical analysis of evolutionary algorithms with an infinite population size in continuous space. Part I: Basic properties of selection and mutation", *IEEE Trans. Neural Netw.*, vol. 5, pp. 102–119, 1994.

[28] I. Karcz-Dulęba, "Dynamics of two-element populations in the space of population states", *IEEE Trans. Evol. Comput.*, vol. 10, pp. 199–209, 2006.

[29] I. Karcz-Dulęba, *Dynamika Adaptacji Ewolucyjnych Metod Fenotypowych*. Wrocław University of Technology Press, 2008 (in Polish).

[30] J. Arabas, "Approximating the genetic diversity of populations in the quasi-equilibrium state", *IEEE Trans. Evol. Comput.*, 2011 (to appear).

[31] A. Obuchowicz and P. Pretki, "Evolutionary algorithms with stable mutations based on a discrete spectral measure", in *Proc. 10th Conf. Artif. Intell. Soft Comput. LNAI*, vol. 6114, part 2, pp. 181–188, Springer, 2010.

[32] R. Storn, "Differential evolution research – trends and open questions", in *Advances in Differential Evolution*, U. K. Chakraborty, Ed. Springer, 2008, pp. 1–31.

[33] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies", *Evol. Comput.*, vol. 9, pp. 159–195, 2001.

[34] N. Hansen, "Compilation of results on the 2005 CEC benchmark function set", Tech. Rep., Institute of Computational Science ETH Zurich, 2006.

[35] A. E. Eiben, R. Hinterding, and Z. Michalewicz, "Parameter control in evolutionary algorithms", *IEEE Trans. Evol. Comput.*, vol. 3, pp. 124–141, 1999.

[36] A. E. Eiben and S. K. Smit, "Parameter tuning for configuring and analyzing evolutionary algorithms", *Swarm and Evol. Comput.*, vol. 1, 2011 (to appear).

[37] J. J. Grefenstette, "Optimization of control parameters for genetic algorithms", *IEEE Trans. Systems, Man, and Cybernetics*, vol. 16, pp. 122–128, 1986.

[38] K. Karimi, N. G. Dickson, and F. Hamze, "A Performance Comparison of CUDA and OpenCL", arXiv:1005.2581v3, 2011.

[39] O. Maitre, F. Krüger, S. Querry, N. Lachiche, and P. Collet, "EASEA: Specification and execution of evolutionary algorithms on GPGPU", *Soft Comput.*, vol. 1, pp. 1–19, 2011.

[40] T. Puzniakowski and M. Bednarczyk, "Towards an OpenCL implementation of genetic algorithms on GPUs", in *Proc. 19th Int. Conf. Intel. Inf. Systems*, Warsaw, Poland, 2011 (to appear).

[41] R. Galar, *Miękka Selekcja w Losowej Adaptacji Globalnej w R^n*. Wrocław University of Technology Press, 1990 (in Polish).

Jarosław Arabas received the M.Sc., Ph.D., and D.Sc. degrees from the Warsaw University of Technology (WUT), Warsaw, Poland, in 1993, 1996, and 2005, respectively. Since 1993, he has been with the Faculty of Electronics and Computer Engineering, WUT, where he is currently a Professor. Since 1995, he has been consulting on applications of these methods for power plant control, decision making, and risk management in electricity markets. His main areas of research interests include evolutionary computation optimization methods, and learning systems.

E-mail: jarabas@elka.pw.edu.pl
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

Paper

Self-Adaptive Stable Mutation Based on Discrete Spectral Measure for Evolutionary Algorithms

Andrzej Obuchowicz and Przemysław Prętki

Institute of Control and Computation Engineering, University of Zielona Góra, Zielona Góra, Poland

Abstract—In this paper, the concept of a multidimensional discrete spectral measure is introduced in the context of its application to the real-valued evolutionary algorithms. The notion of a discrete spectral measure makes it possible to uniquely define a class of multivariate heavy-tailed distributions, that have recently received substantial attention of the evolutionary optimization community. In particular, an adaptation procedure known from the distribution estimation algorithms (EDAs) is considered and the resulting estimated distribution is compared with the optimally selected referential distribution.

Keywords— discrete spectral measure, evolutionary algorithms, heavy-tailed distributions, mutation parameters adaptation.

1. Introduction

Evolutionary Algorithms (EAs) have been successfully applied to global optimization problems in many areas of engineering. Their advantage over many other optimization techniques consists in the fact that EAs are based only on function evaluations and comparisons [1]. Thus, EAs are able to deal successfully with problems that cannot be easily solved by standard optimization procedures. Unfortunately, the EAs also suffer from many serious drawbacks. The most severe one is related to the appropriate choice of their control parameters, which to a large extent determine their performance. Usually, control parameters such as a strength of a mutation, a population size, and a selective pressure are chosen during trial-and-error process or on the base of the expert knowledge, which, unfortunately, is usually inaccessible or the cost of its collection exceeds decidedly the computational cost of the optimization process itself. One way out of these difficulties is to apply algorithms, which make use of some heuristics and dedicated techniques that aim at adjusting some of the control parameters automatically during the optimization process.

In spite of the fact that the problem of the parameter adaptation has been attacked from various angles by many authors and a number of relevant results have already been reported in the literature, there is still a lack of an unified theory that addresses the problem being undertaken.

In the case of the EAs, most attention has been directed toward a normal distribution-based mutation. Thus, several relevant approaches to the adaptation of its parameters have been already reported in the literature [2], [3]. On the other hand, it is noticeable that a normal distribution does not guarantee the highest performance of EAs, so that other distributions have recently aroused evolutionary algorithms community interest. In particular, a lot of attention has been drawn to the heavy-tailed, α-stable distribution [4]–[10]. It turns out that evolutionary algorithms, which make use of the distribution of this class, gain abilities that allow them to find a balanced compromise between exploitation and exploration of the search space [11].

In general, the application of the multidimensional stable distributions to the global optimization algorithms has been limited to the simplest cases: the mutation of the base point obtained by adding a random vector composed of stable, independent, random variables [6], [7], [9], [10], or an isotropic random vector [8], [12]. This limitation causes that many properties of the stable distributions, which can turn out valuable in the context of the optimization processes, are not exploited. In order to obtain the possibility of modeling complicated dependencies between decision variables, the Discrete Spectral Measure (DSM) is proposed to generate a wide class of random vectors [13]. Unfortunately, the mutation distribution generation for a given parent solution is of a high time complexity. In [14], an estimation distribution algorithm (EDA) [15] dedicated to the evolutionary strategy $(1,\lambda)ES$ is proposed. The aim of this work is a comparative analysis of the distributions obtained by the EDAs with optimally selected referential distributions described in [13].

The paper is organized as follows. Multivariate α-stable distributions are defined in Section 2. Section 3 contains a definition and main properties of the stable random vectors based on the discrete spectral measure. In Section 4, an adaptive scheme that aims at adjusting discrete spectral measure is briefly introduced. A set of simulation experiments and their solutions are described and concluded in Section 5. Section 6 concludes the paper.

2. Univariate Stable Distribution

Definition 1: A random variable X is *stable* or *stable in the broad sense* if for X_1 and X_2 independent copies of X and any positive constants a and b,

$$aX_1 + bX_2 \stackrel{d}{=} cX + d \tag{1}$$

for some positive c and some $d \in \mathbb{R}$ and $\stackrel{d}{=}$ means that the left and right random vectors have the same distribution.

A random variable is strictly stable or stable in the narrow sense if Eq. (1) holds with $d = 0$ for all choices of a and b.

Due to the lack of the closed form formulas for densities, the stable distribution can be most conveniently described by its characteristic function (ch.f.) $\varphi(k)$ – the inverse Fourier transform of the probability density function (pdf). The ch.f. of the stable distribution is parameterized by a quadruple $(\alpha,\beta,\sigma,\mu)$ [16], where α $(0<\alpha\leq 2)$ is a stability index (tail index, tail exponent or characteristic exponent), β $(-1\leq\beta\leq 1)$ is a skewness parameter, σ $(\sigma>0)$ is a scale parameter and μ is a location parameter. There are a variety of formulas of the ch.f. of the stable distribution in the relevant literature. This fact is caused by a combination of the historical evolution and numerous problems that have been analyzed using specialized forms of them. The most popular formula of the ch.f. of $X\sim S_\alpha(\beta,\sigma,\mu)$, i.e., a *$\alpha$-stable random variable* (called also *Lévy-stable* or *just stable*) with parameters α, β, σ and μ, is given by [16]:

$$\varphi(k)=\exp\left(-\sigma^\alpha|k|^\alpha\left\{1-i\beta\,\mathrm{sign}(k)\tan\left(-\frac{\pi\alpha}{2}\right)\right\}+i\mu k\right), \tag{2}$$

when $\alpha\neq 1$, and

$$\varphi(k)=\exp\left(-\sigma|k|\left\{1+i\beta\,\mathrm{sign}(k)\frac{2}{\pi}\log|k|\right\}+i\mu k\right), \tag{3}$$

when $\alpha=1$.

In a general case, the complexity of the problem of simulating sequences of α-stable random variables results from the fact that there is no an analytical form for the inverse of the cumulative distribution function (cdf) away from the Gaussian distribution $S_2(0,\sigma,\mu)$, Cauchy distribution $S_1(0,\sigma,\mu)$, and Lévy distribution $S_{1/2}(1,\sigma,\mu)$. The first breakthrough was made by Kanter [17], who gave a direct method for simulating $S_\alpha(1,1,0)$ random variables, for $\alpha<1$. In general cases the following result of Chambers, Mallows and Stuck [18] gives a method for simulating any α-stable random variable [7], [19].

Theorem 1: Let V and W be independent with $V\sim U(-\frac{\pi}{2},\frac{\pi}{2})$, W exponentially distributed with mean 1, $0<\alpha\leq 2$.

1. The symmetric random variable

$$Z=\begin{cases}\frac{\sin(\alpha V)}{(\cos(V))^{1/\alpha}}\left[\frac{\cos((\alpha-1)V)}{W}\right]^{(1-\alpha)/\alpha} & \alpha\neq 1,\\ \tan(V) & \alpha=1\end{cases}$$

has an $S_\alpha(0,1,0)=S\alpha S$ distribution.

2. In the nonsymmetric case, for any $-1\leq\beta\leq 1$, define $B_{\alpha,\beta}=\arctan(\beta\tan(\pi\alpha/2))/\alpha$ when $\alpha\neq 1$. Then

$$Z=\begin{cases}\frac{\sin(\alpha(B_{\alpha,\beta}+V))}{(\cos(\alpha B_{\alpha,\beta})\cos(V))^{1/\alpha}}\left[\frac{\cos(\alpha B_{\alpha,\beta}+(\alpha-1)V)}{W}\right]^{(1-\alpha)/\alpha} & \\ & \alpha\neq 1,\\ \frac{2}{\pi}\left[\left(\frac{\pi}{2}+\beta V\right)\tan(V)-\beta\ln\left(\frac{\frac{\pi}{2}W\cos(V)}{\frac{\pi}{2}+\beta V}\right)\right] & \\ & \alpha=1\end{cases}$$

has an $S_\alpha(\beta,1,0)$ distribution.

It is easy to get V and W from independent uniform random variables $U_1,U_2\sim U(0,1)$: set $V=\pi(U_1-\frac{1}{2})$ and $W=-\ln(U_2)$. Given the formulas for the simulation of standard α-stable random variables (Theorem 1), an α-stable random variable $X\sim S_\alpha(\beta,\sigma,\mu)$ for all admissible values of the parameters α, β, σ and μ has the form

$$X=\begin{cases}\sigma Z+\mu & \alpha\neq 1,\\ \sigma Z+\frac{2}{\pi}\beta\sigma\ln(\sigma)+\mu & \alpha=1,\end{cases} \tag{4}$$

where $Z\sim S_\alpha(\beta,1,0)$.

Many interesting properties of α-stable distributions the reader can find in [7], [11], [20].

3. Multivariate Stable Distribution

There are many alternative and equivalent ways to define stable multivariate distributions [20]. One of them is based on the form of their ch.f.

Definition 2: The ch.f. $\varphi(\mathbf{k})=E[\exp(-i\mathbf{k}^T\mathbf{X})]$ of the random stable vector $\mathbf{X}$ has the following form:

$$\varphi(\mathbf{k})=\exp\left(-\int_{S^{(d)}}|\mathbf{k}^T\mathbf{s}|^\alpha\left(1-i\,\mathrm{sign}(\mathbf{k}^T\mathbf{s})\tan\left(\frac{\pi\alpha}{2}\right)\right)\Gamma(d\mathbf{s})+i\mathbf{k}^T\mu\right) \tag{5}$$

for $\alpha\neq 1$, and

$$\varphi(\mathbf{k})=\exp\left(-\int_{S^{(d)}}|\mathbf{k}^T\mathbf{s}|\left(1-i\frac{2}{\pi}\mathrm{sign}(\mathbf{k}^T\mathbf{s})\ln|\mathbf{k}^T\mathbf{s}|\right)\Gamma(d\mathbf{s})+i\mathbf{k}^T\mu\right) \tag{6}$$

for $\alpha=1$, where $\Gamma(\cdot)$ is the so-called spectral measure, $\boldsymbol{\mu}$ stands for shift vector, and

$$\mathrm{sign}(x)\begin{cases}1 & \text{if} \quad x>0,\\ 0 & \text{if} \quad x=0,\\ -1 & \text{if} \quad x<0.\end{cases} \tag{7}$$

It turns out that a pair $\{\Gamma,\boldsymbol{\mu}\}$ uniquely determine a stable distribution [20]. It is worth to notice that any linear combination of the components of the stable vector described by Definition 2 is univariate α-stable variable $S_\alpha(\beta,\sigma,\mu)$. It must to stressed that the definition of stable vectors is not straightforward and the presence of spectral measure Γ causes that the class is not an ordinary parametric family. In consequence, a direct definition in practical applications is used rather occasionally. Indeed, in the subsequent part of the paper, our attention is restricted only to the class of stable distributions with discrete spectral measure which possess decidedly simpler form.

4. Stable Distributions with Discrete Spectral Measure

A DSM Γ can be defined by means of Delta Dirac distribution in the following way:

$$\Gamma(\cdot;\boldsymbol{\xi},\boldsymbol{\gamma})=\sum_{i=1}^{n_s}\gamma_i\delta_{\mathbf{s}_i}(\cdot), \tag{8}$$

where $\boldsymbol{\xi} = \{\mathbf{s_i}\}_{i=1}^{n_s}, \mathbf{s_i} \in \partial S^{(d)}$ is a set of support points concentrated on a surface of a d-dimensional unit sphere, and $\boldsymbol{\gamma} = \{\gamma_i\}_{i=1}^{n_s}, \gamma_i \in \mathbb{R}_+$ stands for the set of their weights. In this way, for every set $A \subset \partial S^{(d)}$ its measure is given by:

$$\Gamma(A) = \sum_{i=1}^{n_s} \gamma_i I_A(\mathbf{s}_i), \tag{9}$$

where $I_A(\cdot)$ is an indictor function of the set A. Characteristic function (5) and (6) in the case of spectral measure Eq. (8) has the form [19]:

$$\varphi(\mathbf{k}) = \exp\left(-\sum_{i=1}^{n_s} \gamma_i |\mathbf{k}^T \mathbf{s_i}|^\alpha \left(1 - i\,\mathrm{sign}(\mathbf{k}^T \mathbf{s_i}) \tan\left(\frac{\pi\alpha}{2}\right)\right) + i\mathbf{k}^T \boldsymbol{\mu}\right) \tag{10}$$

for $\alpha \neq 1$, and

$$\varphi(\mathbf{k}) = \exp\left(-\sum_{i=1}^{n_s} \gamma_i |\mathbf{k}^T \mathbf{s_i}| \left(1 - i\frac{2}{\pi}\mathrm{sign}(\mathbf{k}^T \mathbf{s_i}) \ln|\mathbf{k}^T \mathbf{s_i}|\right) + i\mathbf{k}^T \mu\right) \tag{11}$$

for $\alpha = 1$.

The definition of the DSM allows to use multivariate stable distributions in the simpler way. It is worth to notice that application of the DSM does not limit any properties of multivariate stable vectors. The following theorem can be proved [21].

Theorem 2: Let $p(\mathbf{x})$ be a density function of the stable distribution described by the characteristic function (5) and (6), and $p^*(\mathbf{x})$ is a density function of the random vector described by the characteristic function (10) and (11), then

$$\begin{aligned} &\forall \varepsilon > 0 \quad \exists n_s \in \mathbb{N} \quad \exists \boldsymbol{\xi}, \gamma \quad \forall \mathbf{x} \in \mathbb{R}^d : \\ &\sup_{\mathbf{x} \in \mathbb{R}^d} |p(\mathbf{x}) - p^*(\mathbf{x})| < \varepsilon. \end{aligned} \tag{12}$$

In other words, each stable distribution can be approximated by some distribution based on the DSM with any accuracy. Especially, the existence of a procedure of pseudo-random vectors generation is very important. It turns out, that a simulation procedure of stable random vectors $\mathbf{X}$ defined by the characteristic functions (10) and (11) is straightforward, and can be implemented by the following stochastic decomposition:

$$\mathbf{X} \stackrel{d}{=} \begin{cases} \sum_{i=1}^{n_s} \gamma_i^{1/\alpha} Z_i \mathbf{s}_i & \text{for} \quad \alpha \neq 1, \\ \sum_{i=1}^{n_s} \gamma_i^{1/\alpha} \left(Z_i - \frac{2}{\pi}\ln(\gamma_i)\right) \mathbf{s}_i & \text{for} \quad \alpha = 1, \end{cases} \tag{13}$$

where Z_i are i.i.d. stable random variables $S_\alpha(1,1,0)$ for which an effective generator can be found in [20].

Random vectors ($\mathbf{X} = [X_1, X_2, \ldots, X_n], X_i \sim S_\alpha S(\sigma)$) composed of independent symmetric elements possess a special status in the application to mutation operators of EAs [5], [6], [7], [14], [9], [10]. It occurs that random vectors can be enriched by adding μ and β parameters, i.e., $\mathbf{X} = [X_1, X_2, \ldots, X_n], X_i \sim S_\alpha(\sigma, \beta, \mu)$ if the DSMs are applied. It means that each component acquires additional degrees of freedom. This fact is very important in the context of application of the above random vector to modeling complicated dependencies between decision variables. It is easy to show that an exploration of such dependencies and their inclusion to a mutation operator accelerates the optimization process. In order to illustrate the possibilities of the DSM representation of the random vectors, it can be mention that the vector with independent components $X_i \sim S_\alpha S(\sigma)$ have the DSM focused in the points of orthogonal axes and surface of the unit sphere intersection with a different weights. The versatility of the DSM representation of the distribution is included in [20].

Theorem 3: The spectral measure of the stable vector $\mathbf{X}$ is described by a finite number of the support vectors $\mathbf{s}_i$ if, and only if the vector $\mathbf{X}$ can be represented by a linear combination of the independent stable random variables, i.e.:

$$\mathbf{X} = \mathbf{A}\mathbf{Z}, \tag{14}$$

where $\mathbf{A} \in \mathbb{R}^{d \times N}$, $\mathbf{Z} = [Z_1, \ldots, Z_N]^T, Z_i \sim S_\alpha(\sigma, \beta, \mu)$.

Based on the Theorem 3, it can be shown that the DSM can be also applied to represent vectors which are described by parameters σ, β, μ and by stochastic dependencies between these vectors.

One of the important properties of the stable distribution based on the DSM and a finite set of support vectors is that almost all probability mass remote from the base point is focused around directions described by the support vectors. So, macromutations take place only in direction parallel to the DSM support vectors. This effect is illustrated in Fig. 1.

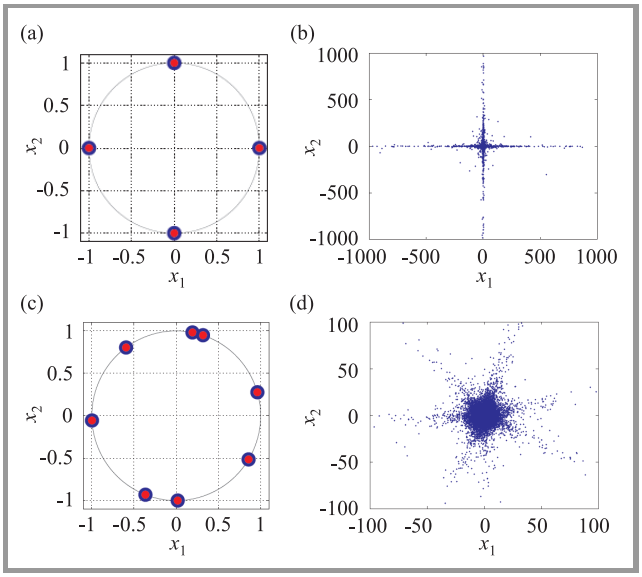


Fig. 1. Distribution of the DSM support vectors and corresponded random realizations: (a), (b) – α=0.75, (c), (d) – α=1.5.

Summarizing, two main benefits obtained by the application of the DSM representation of a multivariate stable distribution can be distinguished: macromutations which allows simple cross saddles of the searching environments, and

possibility of modeling complex stochastic dependencies. The above mentioned benefits are illustrated in [13].

5. Learning Probability Model with Discrete Spectral Measure

One of the most important factors influencing an effectiveness of a global optimization procedure is the possibility of a configuration parameters adaptation. Especially, this mechanism can be applied to mutation operators in evolutionary algorithms. There are many instances in literature, in which such procedures are proposed for mutation parameters [8], [22]–[24], however, all of them consider the Gaussian mutation, only. In the case of the whole class of multidimensional stable distributions, Rudolph shows that adaptation procedures should be different for different stable indices α [8]. In this paper an original parameter adaptation procedure of the mutations based on the multidimensional stable distributions described by the DSM is proposed.

Let us focus our attention on the evolution strategy $(1,\lambda)ES$. In this strategy, the population of descendants is generated by λ mutations of a given base point. All descendant points are evaluated, i.e., the optimized fitness function is calculated in these points. The descendant with the best fitness is chosen as a new base point. The above described operations are iteratively repeated until a given stop criterion is met.

The necessity of adjusting probability model of the mutation utilized to explore a search space is undisputed. It can be noticed that a fixed probability distribution usually causes many serious problems. On one hand, tails of the distribution might be too narrow to allow an algorithm to escape from the local solution in a reasonable number of generations. On the other hand, the probability mass focused around the center of the mutation might be insufficient to make a significant progress in improving an estimate of the global solution. An ideal adaptation procedure should be able to detect each of these situations, and adjust probabilistic model in such a way to prevent the algorithm from a stagnation.

The method of the optimal probabilistic model choice is introduced and illustrated in [13]. The goal is the correction of the current solution by a perturbation using a stable random vector $\mathbf{X}_{\boldsymbol{\xi}}^{\boldsymbol{\gamma}}$. The aim of this calculation is the selection of the optimal stable model from the class of multivariate distributions described by the DSM. If we assume the fixed set of n_s support points $\boldsymbol{\xi}$, the criterion of the best model selection is chosen in the form:

$$\boldsymbol{\gamma}^* = \arg\min_{\boldsymbol{\gamma}\in\mathbb{R}_+^{n_s}} C(\boldsymbol{\gamma}), \tag{15}$$

where

$$C(\boldsymbol{\gamma}) = E\left[\min\left\{\frac{\phi(\mathbf{x}_k + \mathbf{X}_{\boldsymbol{\xi}}^{\boldsymbol{\gamma}}(\alpha))}{\phi(\mathbf{x}_k)}, 1\right\}\right]. \tag{16}$$

It occurs that the function (16) does not possess an analytical form, thus, the problem Eq. (15) cannot be solved using standard optimization techniques. One of the possible solutions is the application of the Monte Carlo method [25]. The law of large numbers [26] allows to approximate the expectation value (16) using the following estimator:

$$\hat{C}(\boldsymbol{\gamma}) = \frac{1}{N}\sum_{i=1}^{N}\min\left\{\frac{\phi(\mathbf{x}_k + \mathbf{X}_{i,\xi}^{\boldsymbol{\gamma}}(\alpha))}{\phi(\mathbf{x}_k)}, 1\right\}, \tag{17}$$

where $\{\mathbf{X}_{i,\boldsymbol{\xi}}^{\boldsymbol{\gamma}}(\alpha))\}_{i=1}^{N}$ stands for a sequence of independent realizations of the random vector with the α-stable distribution. Using the estimator (17), the problem Eq. (15) can be rewritten into the form:

$$\boldsymbol{\gamma}^* = \arg\min_{\boldsymbol{\gamma}\in\mathbb{R}^{n_s}} \hat{C}(\boldsymbol{\gamma}). \tag{18}$$

Because the objective function possesses a stochastic properties and in order to achieve the compromise between the computation complexity and the estimator quality, the SPSA algorithm [27] can be chosen for solving Eq. (18).

The quality of the estimator (17) strongly depends on the number N of independent realizations of the random vectors $\{\mathbf{X}_{i,\boldsymbol{\xi}}^{\boldsymbol{\gamma}}(\alpha))\}_{i=1}^{N}$. Experiments shows that, in order to obtain a representative estimator $\hat{C}$ (17) in the 2D searching space, N should be up to dozens thousands. Because of such a large complexity of the optimal probabilistic model choice, an adaptive scheme known from the class of the estimation of distribution algorithms (EDAs) [15] is proposed to adjust a parametric probability model Eq. (8) [14]. The idea of the proposed algorithm is based on the assumption that the selection pressure of the evolutionary process designates the most valuable set of the independent realizations of the random vectors. This idea utilized to evolutionary strategy $(1,\lambda)$ES with the mutation operator that is based on the DSM boils down to the algorithm $(1,\lambda/\mu)ES_\alpha$ composed of the following simple steps:

Step 0: Set $k=1$ and choose an initial guess of a global solution $\mathbf{x}_0$ and initial weights vector $\boldsymbol{\gamma}_0$, i.e.:

$$\mathbf{x}_k = \mathbf{x}_0, \quad \Gamma_k = (\boldsymbol{\xi}, \boldsymbol{\gamma}_0), \tag{19}$$

where $\boldsymbol{\xi} = \{\mathbf{s}_i\}_{i=1}^{n_s}$, $\mathbf{s}_i \in S^d$ is a fixed grid of a DSM, and $\boldsymbol{\gamma}_k = \{\gamma_i\}_{i=1}^{n_s}$ stands for a vector of weights associated with support points $\mathbf{s}_i$.

Step 1: Randomly pick a set of candidate solutions using probabilistic model Γ_k.

$$\mathbf{P}_{k,\lambda} = \{\mathbf{x}_{k,1}, \mathbf{x}_{k,2}, \ldots, \mathbf{x}_{k,\lambda}\}, \quad \mathbf{x}_{k,i} = \mathbf{x}_k + \mathbf{X}_i, \tag{20}$$

where $\mathbf{X}_i \sim \Gamma_k$ are i.i.d. random vector generated according Eq. (13).

Step 2: Select μ the best solutions from the current population $\mathbf{P}_{k,\lambda}$:

$$\mathbf{P}_{k,\mu} = \{\mathbf{x}_{k,1:\lambda}, \mathbf{x}_{k,2:\lambda}, \ldots, \mathbf{x}_{k,\mu:\lambda}\} \tag{21}$$

Step 3: Build the probability model of the selected solutions $\mathbf{P}_{k,\mu}$.

$$\gamma_k = Est(\mathbf{P}_{k,\mu}) \tag{22}$$

where $Est(\cdot)$ is an estimation procedure of the weights described in the subsequent part of this section.

Step 4: Set $\mathbf{x}_k = \mathbf{x}_{k,1:\lambda}$.

Step 5: If the stopping condition are not met, go to Step 1.

The Step 3 deserves more detailed description. In the literature [19], [28]–[30] several approaches to the estimation of the DSM from a given data set can be found. The simplest and the less computational intensive one was presented in [19]. This method is based on the, so called, empirical ch.f.:

$$\hat{\phi}(\mathbf{k}) = \frac{1}{N}\sum_{i=1}^{N}\exp(j\mathbf{k}^T\mathbf{X}_i), \tag{23}$$

where $\mathbf{X}_i$ are observed random variables realizations, which are included in the data set $\{\mathbf{x}_i\}_{i=1}^{N}$. Assuming that the DSM Γ is defined by the finite set of support vectors $\boldsymbol{\xi} = \{\mathbf{s}_i\}_{i=1}^{n_s}$ and, corresponded to them, weighs $\boldsymbol{\gamma} = \{\gamma_i\}_{i=1}^{n_s}$ t.j.:

$$\Gamma = \left\{ \begin{array}{cccc} \mathbf{s}_1 & \mathbf{s}_2 & \dots & \mathbf{s}_{n_s} \\ \gamma_1 & \gamma_2 & \dots & \gamma_{n_s} \end{array} \right\}, \tag{24}$$

the estimation problem can be reduced to the optimization problem

$$\Gamma^* = \arg\min_{\boldsymbol{\xi},\boldsymbol{\gamma}} \|\hat{\phi}(\mathbf{k}) - \phi(\mathbf{k};\boldsymbol{\xi},\boldsymbol{\gamma})\|. \tag{25}$$

In order to the problem simplification, let us assume that the considered DSM is based on the fixed set of support vectors, which are uniformly distributed on the unique sphere surface. Than the model estimation problem (25) is reduced to the form:

$$\boldsymbol{\gamma}^* = \arg\min_{\boldsymbol{\gamma}} \|\hat{\phi}(\mathbf{k}) - \phi(\mathbf{k};\mathbf{S}_{n_s},\boldsymbol{\gamma})\|. \tag{26}$$

Searching for the precise solution of the problem Eq. (26) is connected with a very large computation effort and the application of this method seems to be unpractical. We should introduce another problem simplification. The expression (26) can be estimated using the set of testing points $\mathbf{K} = \{\mathbf{k}_i\}_{i=1}^{n_k}$:

$$\boldsymbol{\gamma}^* = \arg\min_{\boldsymbol{\gamma}} \sum_{i=1}^{n_k}\left(\hat{\phi}(\mathbf{k}_i) - \phi(\mathbf{k}_i;\boldsymbol{\xi},\boldsymbol{\gamma})\right)^2 \tag{27}$$

Let $\mathbf{I} = -[\ln\hat{\phi}(\mathbf{k}_1),\dots,\ln\hat{\phi}(\mathbf{k}_{n_k})]^T$ and

$$\boldsymbol{\Psi}(\mathbf{k}_1,\dots,\mathbf{k}_{n_k};\mathbf{s}_1,\dots,\mathbf{s}_{n_s}) = \begin{pmatrix} \psi_\alpha(\mathbf{k}_1^T\mathbf{s}_1) & \dots & \psi_\alpha(\mathbf{k}_1^T\mathbf{s}_{n_s}) \\ \vdots & \vdots & \vdots \\ \psi_\alpha(\mathbf{k}_{n_k}^T\mathbf{s}_1) & \dots & \psi_\alpha(\mathbf{k}_{n_k}^T\mathbf{s}_{n_s}) \end{pmatrix} \tag{28}$$

for

$$\psi_\alpha(u) = \left\{ \begin{array}{ll} |u|^\alpha(1 - i\,\mathrm{sgn}(u)\tan(\frac{\pi\alpha}{2})), & \text{for} \quad \alpha \neq 1 \\ |u|(1 - i\frac{2}{\pi}\mathrm{sgn}(u)\ln(|u|)), & \text{for} \quad \alpha = 1 \end{array} \right. \tag{29}$$

than the optimal weight set is the solution of the system of equations [19]

$$\mathbf{I} = \boldsymbol{\Psi}\boldsymbol{\gamma}^*. \tag{30}$$

In order to ensure well-conditional problem (30) let us assume $n_s = n_k$ and $\mathbf{s}_i = \mathbf{k}_i$. Finally, the problem is reduced to solving the following constrained quadratic programming problem [19]:

$$\boldsymbol{\gamma}^* = \arg\min_{\boldsymbol{\gamma}\geq\mathbf{0}} ||\mathbf{c} - \mathbf{A}\boldsymbol{\gamma}||_2, \tag{31}$$

where $\mathbf{c} = [\mathrm{Re}\{I_{1:n/2}\},\mathrm{Im}\{I_{n/2+1:n}\}]^T$ is a vector containing n real values of the vector I and n its image values, $\mathbf{A} = [\mathrm{Re}\{\boldsymbol{\psi}_1^T\},\dots,\mathrm{Im}\{\boldsymbol{\psi}_n^T\}]^T$, is similarly organized matrix (28), where $\boldsymbol{\psi}_i = [\psi(\mathbf{s}_1^T\mathbf{s}_i),\psi(\mathbf{s}_2^T\mathbf{s}_i),\dots,\psi(\mathbf{s}_{n_s}^T\mathbf{s}_i)]^T \in \mathbb{C}^{n_s}$ are its rows.
The problem Eq. (31) can be solved analytically or using one of the dedicated gradient-based optimization method.

6. Experimental Simulation

In this section we experimentally try to prove, that the self-adapted DSM can improve the optimization efficiency of the evolution strategy being considered.

6.1. Experiment 1

Three versions of the evolution strategy will be analyzed:

- **A1** – $(1,\lambda)\mathrm{ES}_\alpha$ for which mutation is based on the DSM with following support vectors:

$$\boldsymbol{\xi} = \left\{ \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} -1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \end{bmatrix} \right\}$$

  The weight vector $\boldsymbol{\gamma} = [\sigma/4,\sigma/4,\sigma/4,\sigma/4]^T$ is fixed during the evolutionary process.

- **A2** – $(1,\lambda/\nu)\mathrm{ES}_\alpha$ for which mutation is based on the DSM with following support vectors:

$$\boldsymbol{\xi} = \left\{ \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} -1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \end{bmatrix} \right\}$$

  The initial weight vector $\boldsymbol{\gamma} = [\sigma/4,\sigma/4,\sigma/4,\sigma/4]^T$ is adapted during the evolutionary process in order to the algorithm proposed in this paper.

- **A3** – $(1,\lambda/\nu)\mathrm{ES}_\alpha$ for which mutation is based on the DSM with following support vectors:

$$\boldsymbol{\xi} = \left\{ \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix}, \begin{bmatrix} -1 \\ 0 \end{bmatrix}, \begin{bmatrix} -\frac{\sqrt{2}}{2} \\ -\frac{\sqrt{2}}{2} \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \begin{bmatrix} -\frac{\sqrt{2}}{2} \\ -\frac{\sqrt{2}}{2} \end{bmatrix} \right\}$$

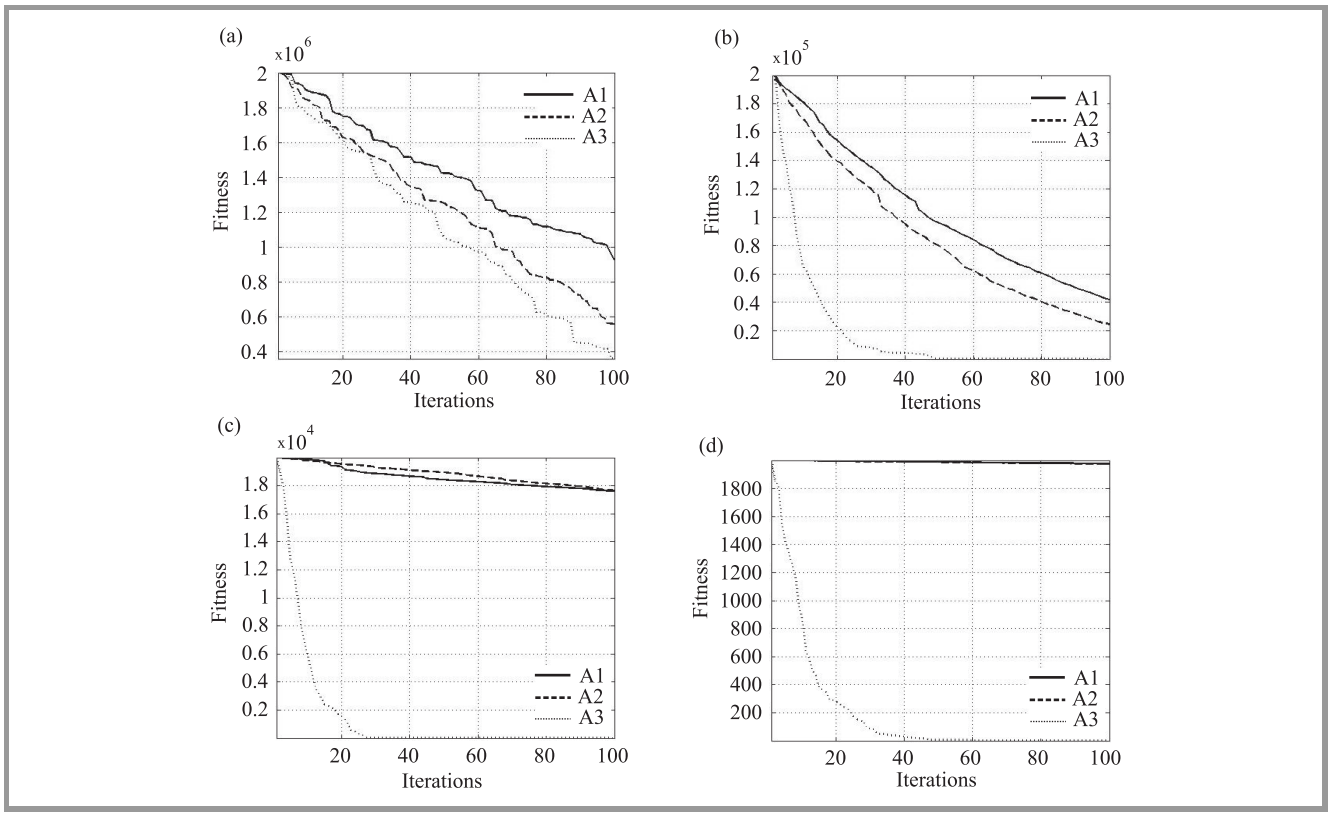


Fig. 2. Fitness of the base point versus iterations. Average results (taken over 50 algorithms independent runs) for quadratic function Eq. (32) obtained for algorithms with different DMSes: four support points without weights adaptation **A1**, four support points with weights adaptation **A2**, and eight support points with weights adaptation **A3**. Objective function parameters (e_1, e_2): (a) = $(1,1)$, (b) = $(10, 0.1)$, (c) = $(100, 0.01)$, (d) = $(1000, 0.001)$.

The initial weight vector $\boldsymbol{\gamma} = [\sigma/8, \ldots, \sigma/8]^T$ is adapted during the evolutionary process in order to the algorithm proposed in this paper.

Four two-dimensional unimodal objective functions are selected to experiments:

$$f(\mathbf{x}) = \mathbf{x}^T \mathbb{U}\mathbb{D}\mathbb{U}^T \mathbf{x}, \tag{32}$$

where

$$\mathbb{U} = \begin{pmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{pmatrix}$$

and

$$\mathbb{D} = \begin{pmatrix} e_1 & 0 \\ 0 & e_2 \end{pmatrix},$$

for different conditional factors: $(e_1, e_2) = (1,1)$, $(10, 0.1)$, $(100, 0.01)$, $(1000, 0.001)$. The initial conditions are the same for each strategy: $\lambda = 20$, $\nu = 10$, $\mathbf{x}_0 = [1000, 1000]^T$, $\sigma = 1$. Moreover, we assume that $\sum_{i=1}^{n_s} \gamma_i = \sigma$.

The results for the algorithms with the stability index $\alpha = 0.5$ are presented in Fig. 2.

By analyzing average results presented in Fig. 2 for evolutionary strategies **A1** and **A2** the advantage of the applied adaptation mechanism cannot be declared explicitly. The choice of the stable index $\alpha = 0.5$, for which macromutations in directions parallel to the axis of the reference frame take place an important role in the optimum finding processs. Only after a condensation of the support vectors strategy **A3** there is possibility to fit the exploration distribution to any distribution of the searching space. In cases presented on Fig. 2(b)–2(d) it is easy to observe that the weight $\gamma_6 \approx \sigma$ (it is connected to the vector $\mathbf{s}_6 = [-\sqrt{2}/2, \sqrt{2}/2]$), and the macromutations in this direction are preferred.

6.2. Experiment 2

The goal of the experiments is to recognize how "far-off" is the probabilistic model Eq. (31) applied in $(1, \lambda/\mu)ES_\alpha$ from the optimal one Eq. (18). As a measure of this "distance" in the parent point $\mathbf{x}_k$ obtained in the k-th iteration of evolutionary process the following expression is chosen

$$J(\mathbf{x}_k) = \frac{E_{\Gamma_k^\mu}}{E_{\Gamma_k}}, \tag{33}$$

where

$$E_{\Gamma_k^\mu} = \int_D \min\left(f(\mathbf{x}), f(\mathbf{x}_k)\right) d\Gamma_k^\mu, \tag{34}$$

$$E_{\Gamma_k} = \int_D \min\left(f(\mathbf{x}), f(\mathbf{x}_k)\right) d\Gamma_k, \tag{35}$$

and $f(\mathbf{x})$ ($\mathbf{x} \in D$) is an optimized fitness function, Γ_k^μ and Γ_k are cumulative distribution functions (cdf) of probability models Eqs. (31) and (18), respectively. The expression

$\min(f(\mathbf{x}), f(\mathbf{x}_k))$ in Eq. (34) is introduced instead of the pure fitness function $f(\mathbf{x})$ in order to avoid the possibility to obtain infinite expectation value of $f(\mathbf{x})$.

As it can be seen, the measure $J(\mathbf{x}_k)$ Eq. (33) is equal to the unity in the case of the perfect match of both distributions, and increases when the disproportion between both distributions increases. The value of $J(\mathbf{x}_k)$ also constitutes the measure of deterioration (in the probabilistic sense) of the next population quality when we use the probabilistic model Γ_k^{μ} Eq. (31) instead of the optimal one Γ_k Eq. (18).

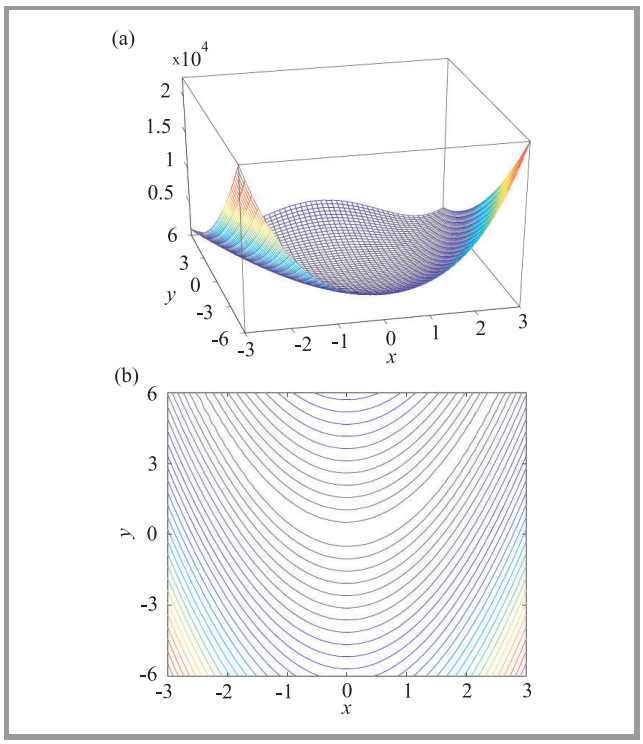


Fig. 3. Rosenbrock's function – (a) and its contour chart – (b).

Let the 2-dimensional Rosenbrock's function (Fig. 3)

$$f(\mathbf{x}) = (1-x_1)^2 + 100(x_2 - x_1^2)^2 \tag{36}$$

be chosen as a objective function for considered computation example. Moreover, we assume that the point $\mathbf{x}_0 = [10,10]^T$ $(f(\mathbf{x}_k) = 810081)$ is the initial approximation of the optimum. Let us reduce the set of rival probabilistic models to a set of four stable distributions $\Omega = \{\mathbf{X}_{\boldsymbol{\xi}}^{\boldsymbol{\gamma}}(\alpha) | \alpha = 0.5, 1.0, 1.5, 2.0\}$. Each random vector $\mathbf{X}_{\boldsymbol{\xi}}^{\boldsymbol{\gamma}}(\alpha)$ is described by the DSM spread on 5 uniformly distributed support points:

$$\boldsymbol{\xi} = \left\{ \begin{bmatrix} 1.0000 \\ 0.0000 \end{bmatrix}, \begin{bmatrix} 0.3090 \\ 0.9511 \end{bmatrix}, \begin{bmatrix} -0.8090 \\ 0.5878 \end{bmatrix}, \begin{bmatrix} -0.8090 \\ -0.5878 \end{bmatrix}, \begin{bmatrix} 0.3090 \\ -0.9511 \end{bmatrix} \right\}.$$

In the case of the $(1, \lambda/\mu)ES_{\alpha}$ algorithm the parameter $\lambda = 100$ and three values of μ are selected $(10, 40, 70)$. There are 128 algorithm runs for each pair (α, μ). In order to estimate the optimal probability model $N = 100000$ independent realizations of the random vectors $\{\mathbf{X}_{i,\boldsymbol{\xi}}^{\boldsymbol{\gamma}}(\alpha))\}_{i=1}^{N}$ is generated.

The obtained results are presented in the form of histograms of $J(\mathbf{x}_k)$ Eq. (33) for each pair (α, μ) taken over 2560 sample points (Fig. 4). It can be observed that there are distinct peaks near $J(\mathbf{x}_k) = 1$ for all cases. It suggests that proposed probability model of mutation Eq. (31) is a good estimator of the optimal probability model. The relation between the quality of this estimator and the stability index α is very interesting. The peaks of $J(\mathbf{x}_k)$ histograms are higher for extreme values of stability index $\alpha = 0.5$ and $\alpha = 2$ than for the medium values $\alpha = 1$ and $\alpha = 1.5$. The explanation of this fact is not quit clear and needs more precise research. Taking into account obtained results of our experiment, the relation between the quality of the estimator and the parameter μ is not clear. Two reasons can influence this fact. The first one, highly probable in our opinion, is that: too low number of points is considered and a statistical error is too high. The second reason can be connected with the fact that the quality of the estimator at a given point $\mathbf{x}$ depends not only on the pair (α, μ) but also on the features of the fitness landscape in the close neighborhood of $\mathbf{x}$.

7. Conclusions

The application of the discrete spectral measure to the stable mutation for evolutionary algorithms based on the real-valued representation of the individual is considered. The evolution strategy $(1+\lambda)ES_{\alpha}$ is chosen as a base evolutionary algorithm. Emphasis is focused on the self adaptation of the mutation probability model to the winner individual in each population. This self adaptation can be parted into two steps: the optimal selection of the set of support points (vectors) and the optimal selection of the weights related to this points. Both tasks are connected with the calculations of a high time and space complexity cost [13], [14]. This fact limits applicability of the proposed method to low dimensional problems. In order to avoid this cost the estimation methods for both tasks should be proposed. In this paper we focus on the second task, i.e., we assume a fixed set of support points. This work contains preliminary results of simulation experiments. Results suggest that probability model of the mutation is a good estimator of the optimal one. The analysis of the relation between the estimator quality and control parameters (α, μ) needs further thorough investigations. Our further research will also be focused on the support points selection method of the lower complexity cost.

References

[1] I. Karcz-Dulęba, "Asymptotic behaviour of a discrete dynamical system generated by a simple evolutionary process", *Int. J. Appl. Math. Comput. Sci.*, vol. 14, no. 3, pp. 79–90, 2004.

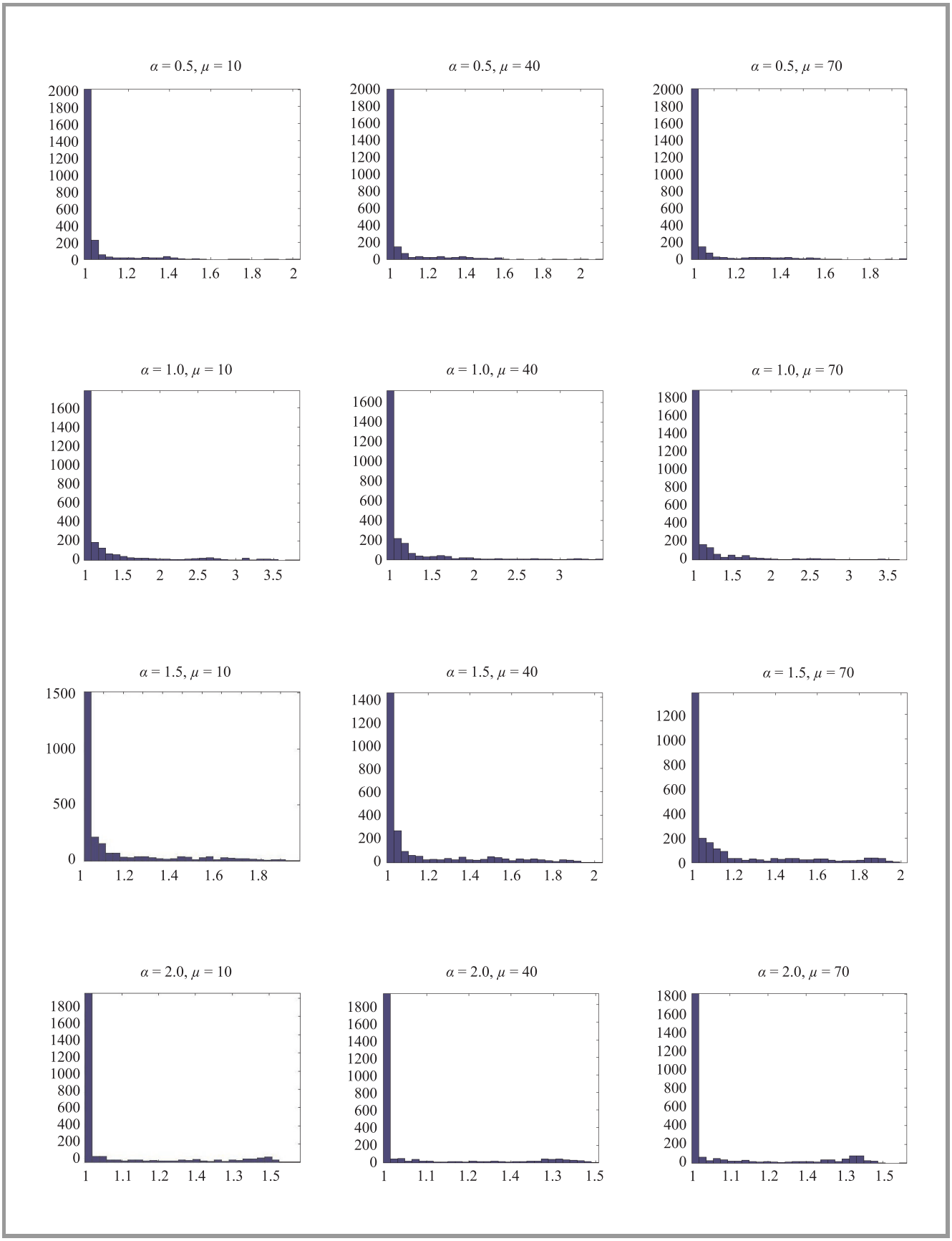


Fig. 4. Histograms of $J(\mathbf{x}_k)$ Eq. (33) for each pair (α, μ) taken over 2560 sample points.

[2] H. G. Beyer and H. P. Schwefel, "Evolutionary strategies – a comprehensive introduction", *Neural Computing*, vol. 1, no. 1, pp. 3–52, 2002.

[3] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolutionary strategies", *Evol. Comput.*, vol. 9, no. 2, pp. 159–195, 2002.

[4] N. Hansen, F. Gemperle, A. Auger, and P. Koumoutsakos, "Why do heavy-tail distributions help?" in *Parallel Problem Solving from Nature PPSN IX*, Springer, LNCS 4193, pp. 62–71, 2006.

[5] X. Liu and W. Xu, "A new filled function applied to global optimization", *Comput. Oper. Res.* vol. 31, pp. 61–80, 2004.

[6] A. Obuchowicz, *Evolutionary Algorithms in Global Optimization and Dynamic System Diagnosis*. Zielona Góra: Lubuskie Scientific Society Press, 2003.

[7] A. Obuchowicz and P. Prętki, "Phenotypic evolution with mutation based on symmetric α-stable distributions", *Int. J. Appl. Math. Comput. Sci.*, vol. 14, no. 3, pp. 289–316, 2004.

[8] G. Rudolph, "Local convergence rates of simple evolutionary algorithms wich Cauchy mutations", *IEEE Trans. Evol. Comput.* vol. 1, no. 4, pp. 249–258, 1997.

[9] X. Yao and Y. Liu, "Fast evolution strategies", in *Evolutionary Programming IV*, P. J. Angeline, R. G. Reynolds, J. R. MacDonnell, and R. Eberhart, Eds. Berlin: Springer, 1997, pp. 151–161.

[10] X. Yao, Y. Liu, and G. Liu, "Evolutionary programming made faster", *IEEE Trans. Evol. Comput.* vol. 3, no. 2, pp. 82–102, 1999.

[11] P. Prętki, "Algorytmy ewolucyjne z mutacją α-stabilną w zadaniach globalnej optymalizacji parametrycznej". Ph.D. thesis, University of Zielona Góra, Poland, 2008 (in Polish).

[12] A. Obuchowicz and P. Prętki, "Isotropic symmetric α-stable mutations for evolutionary algorithms", in *Proc. IEEE Congr. Evol. Comput. CEC'05*, Edinburgh, UK, 2005, pp. 404-410.

[13] A. Obuchowicz and P. Prętki, "Evolutionary algorithms with stable mutations based on a discrete spectral measure", in *Proc. Int. Conf. Artif. Intel. Soft Comput. ICAISC'2010*, Springer-Verlag, LNAI 6114, no. 2, pp. 181–188, 2010.

[14] P. Prętki, "Learning stable mutation in $(1,\lambda)$ES evolutionary strategy", in *Proc. 10th Conf. Evol. Algorithms Global Optimiz.*, Będlewo, Poland, 2007, pp. 233–240.

[15] P. Larranaga and J. A. Lozano, *Estimation of Distribution Algorithms: A New Tool for Evolutionary Optimization.* Boston: Kluwer Academic Publishers, 2001.

[16] A. Zolotariev,*One-Dimensional Stable Distributions*. Providence: American Mathematical Society Press, 1986.

[17] M. Kanter, "Stable densities under change of scale and total variation inequalities", *Ann. Probab.*, vol. 3, no. 4, pp. 697–707, 1975.

[18] J. M. Chambers, C. L. Mallows, and B. W. Stuck, "A method for simulating stable random variables", *J. Amer. Statist. Assoc.*, vol. 71, pp. 340–344, 1976.

[19] J. P. Nolan, A. K. Panorska, and J. H. McCulloch, "Estimation of stable spectral measures – stable non-Gaussian models in finanse and econometrics", *Math. Comput. Model.* vol. 34, no. 9, pp. 1113–1122, 2001.

[20] G. Samorodnitsky and M.S. Taqqu, *Stable Non-Gaussian Random Processes*. New York: Chapman & Hall, 1994.

[21] T. Byczkowski, J. P. Nolan, and B. Rajput, "Approximation of multidimensional stable densities", *J. Mult. Anal.*, vol. 46, pp. 13–31, 1993.

[22] T. B ack, *Evolutionary Algorithms in Theory and Practice*. Oxford University Press, 1995.

[23] I. Rechenberg, *Evolutionsstrategie: Optimierung technisher Systeme nach Prinzipien der biologishen Evolution*. Stuttgard: Frommann-Holzburg Verlag, 1973.

[24] H.-P. Schwefel, *Evolution and Optimum Seeking.* New York: Wiley, 1995.

[25] F. Kemp, "An introduction to sequential Monte Carlo methods", *J. Royal Statist. Soc.* vol. D52, pp. 694–695, 2003.

[26] R. Durrett, *Probability: Theory and Examples*. Duxbury Press, 1995.

[27] J. C. Spall, *Introduction to Stochastic Search and Optimization*. Hoboken, NJ: Wiley, 1993.

[28] L. N. Chrysostomos and M. Shao, *Signal Processing with Alpha-Stable Distribution and Applications*. Chichester, UK: Wiley, 1981.

[29] P. G. Georgiou, P. Tsakalides, and C. Kyriakakis, "Alpha-stable odeling of noise and robust time-delay in the presence of impulsive noise", *IEEE Trans. Multimedia*, vol. 1, no. 3, pp. 291–301, 1999.

[30] P. Kidmose, "Alpha-stable distributions in signal processing of audio signals", in *Proc. 41st Conf. Simulation and Modeling SIMS 2000*, Scandinavian Simulation Society Press, pp. 87–94, 2000.

Andrzej Obuchowicz received the M.Sc. and Ph.D. degrees in physics, and D.Sc. degree in automation and robotics from Technical University of Wrocław in 1987, 1992 and 2004, respectively. He is an Associate Professor in Institute of Control and Computation Engineering University of Zielona Góra, Poland. His research interests cover global optimization methods, application of pattern and image recognition techniques in medical diagnosis as well as technological system diagnosis. He is author or co-author of 2 monographs and over 100 scientific papers.
E-mail: A.Obuchowicz@issi.uz.zgora.pl
Institute of Control and Computation Engineering
University of Zielona Góra
Podgórna st 50
65-246 Zielona Góra, Poland

Przemysław Prętki received the Ph.D. degree in Computer Science from the University of Zielona Góra, Poland, in 2008. In 2003–2010 he was an Assistant Professor in the Institute of Control and Computation Engineering at the University of Zielona Góra. His research interests include methods of global optimization, machine learning, data mining, particularly mining software engineering data and automated software testing. He works now as an engineer at Advanced Digital Broadcast.
E-mail: P.Pretki@issi.uz.zgora.pl
Institute of Control and Computation Engineering
University of Zielona Góra
Podgórna st 50
65-246 Zielona Góra, Poland

Paper

Self-Adaptive Differential Evolution with Hybrid Rules of Perturbation for Dynamic Optimization

Krzysztof Trojanowski[a], Mikołaj Raciborski[b], and Piotr Kaczyński[b]

[a] *Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland*
[b] *Faculty of Mathematics and Natural Sciences, Cardinal Stefan Wyszyński University, Warsaw, Poland*

Abstract—In this paper an adaptive differential evolution approach for dynamic optimization problems is studied. A new benchmark suite `Syringa` is also presented. The suite allows to generate test-cases from a multiple number of dynamic optimization classes. Two dynamic benchmarks: Generalized Dynamic Benchmark Generator (GDBG) and Moving Peaks Benchmark (MPB) have been simulated in `Syringa` and in the presented research they were subject of the experimental research. Two versions of adaptive differential evolution approach, namely the jDE algorithm have been heavily tested: the pure version of jDE and jDE equipped with solutions mutated with a new operator. The operator uses a symmetric α-stable distribution variate for modification of the solution coordinates.

Keywords—adaptive differential evolution, dynamic optimization, symmetric α-stable distribution.

1. Introduction

Uncertainty in the optimized problem demands from the optimization algorithms specific features appropriate to the form of the uncertainty. Particularly, in the presented research our algorithm has to cope with changes in the evaluation function formula or the function parameters appearing during the process of optimum search. Problems with this type of uncertainty are called dynamic problems and searching for solutions of such problems is called dynamic optimization. It is worth noting, that this is just one of the forms of uncertainty appearing in the optimization process. The remaining three according to the classification given in [1] are represented by:

- a noise in the optimized function,
- when for some reason (for example, high computational costs), instead of using the optimized function, we use its approximation evaluation,
- when the main aim is to find a solution not only of the highest quality, but – more importantly – the one whose neighbors are also good, that is, when the most important issue is the robustness of the returned solution.

Among a number of existing metaheuristics we selected a differential evolution (DE) for the research. This approach is recently a subject of growing interests and has already been studied from many points of view (for detailed discussion see, for example, monographs [2] or [3]). Our attention has been paid to the self-adaptive version of the DE algorithm [4] which differs form the basic approach in that a self-adaptive control mechanism is used to change the control parameters F and CR during the run. Eventually, for our research we reimplemented the version of jDE presented in [5]. Our experiments were conducted with this version as well as a version extended by a new mutation operator inspired by a mechanism originating from the particle swarm optimization approach. We also modified the set of benchmark instances and extended the number of the algorithm quality measures which are evaluated during the experiments. This gave a wider view to the nature of the dynamic optimization process performed by jDE and allowed for its better understanding. The main aim of our research was a transfer of the idea of the new mutation operator to the differential evolution approach and experimental verification if such a transfer is justified, that is, allows to improve the algorithm effectiveness.

The paper is organized as follows. In Section 2 a brief description of the optimization algorithm is presented. Section 3 presents properties of new type of mutation introduced to jDE. The new benchmark suite called `Syringa` is presented in Section 4. Section 5 includes some details of the selected test-cases and their configurations as well as the applied performance measures. Section 6 shows the results of experiments. Section 7 concludes the presented research.

2. The Algorithm

The DE algorithm is a kind of evolutionary method with a very specific mutation operator controlled by the scale factor F. Three different, randomly chosen solutions are selected from the current population to mutate a target solution $\mathbf{x}^i$: a base solution $\mathbf{x}^0$ and two difference solutions $\mathbf{x}^1$ and $\mathbf{x}^2$. First, the three solutions are used to create a mutant solution. Then, the mutant undergoes discrete recombination with the target solution $\mathbf{x}^i$. The recombination is controlled by the crossover probability factor $CR \in [0,1]$. Finally, in the selection stage trial solutions compete with their target solutions for the place in the population. This

Algorithm 1 jDE algorithm

Create and initialize the reference set of $(k \cdot m)$ solutions
repeat
  for $l = 1$ to k **do** {*for each subpopulation*}
    for $i = 1$ to m **do** {*for each solution in a subpopulation*}
      Select randomly three solutions: $\mathbf{x}^{l,0}$, $\mathbf{x}^{l,1}$, and $\mathbf{x}^{l,2}$ such that: $\mathbf{x}^{l,i} \neq \mathbf{x}^{l,0}$ and $\mathbf{x}^{l,1} \neq \mathbf{x}^{l,2}$
      for $j = 1$ to n **do** {*for each dimension in a solution*}
        if $(rand(0,1) > CR^{l,i})$ **then**
          $u_j^{l,i} = x_j^{l,0} + F^{l,i} \cdot (x_j^{l,1} - x_j^{l,2})$
        else
          $u_j^{l,i} = x_j^{l,i}$
        end if
      end for
    end for
  end for
  for $i = 1$ to $(k \cdot m)$ **do** {*for each solution*}
    if $(f(\mathbf{u}^i) < f(\mathbf{x}^i)$ **then** {*Let's assume this is a minimization problem*}
      $\mathbf{x}^i = \mathbf{u}^i$
    end if
    Recalculate F^i and CR^i
    Apply aging for $\mathbf{x}^i$
  end for
  Do overlapping search
until the stop condition is satisfied

strategy of population management is called ***DE/rand/1/bin*** which means that the base solution is **rand**omly chosen, **1** difference vector is added to it and the crossover is based on a set of independent decisions for each of coordinates, that is, a number of parameters donated by the mutant closely follows a **bin**omial distribution.

Self-Adaptive Differential Evolution approach, namely the jDE algorithm extends functionality of the basic approach in many ways. First, each object representing a solution in the population is extended by a couple of its personal parameters CR and F. They are modified every generation [4]:

$$F_i(t+1) = \begin{cases} F_{\min} + rand[0,1) \cdot F_{\max} & \text{if } rand[0,1) < 0.1 \\ F_i(t) & \text{otherwise} \end{cases}$$

$$C_i(t+1) = \begin{cases} rand[0,1) & \text{if } rand[0,1) < 0.1 \\ C_i(t) & \text{otherwise} \end{cases}$$

where: $F_{\min} = 0.36$ and $F_{\max} = 0.9$ represent the lower and upper boundary for F, and $F_i(0) = 0.5$. For the CR factor: $CR_i \in [0,1)$ and $CR_i(0) = 0.9$.

The next modifications have been introduced just for better coping in the dynamic optimization environment. The population of solutions has been divided into five subpopulations of size ten (this is the configuration of the algorithm presented in [5]). Each of them has to perform its own search process, that is, no information is shared between the subpopulations. Every solution is a subject to the aging mechanism protecting against stagnation in local minima and just the global-best solution is excluded from this. To avoid overlapping between subpopulations a distance between subpopulation leaders is calculated and if too close localization is observed then one of the subpopulations is reinitialized. However, as in the previous case the subpopulation with the global-best is never the one to reinitialize. The last extension is application of a memory structure called archive. The archive is increased after every change in the fitness landscape by the current global-best solution. Recalling from the archive can be executed every reinitialization of a subpopulation, however, decision about the execution depends on a few conditions. For details of the above-mentioned extension procedures the reader is referred to [5].

3. Proposed Extension of jDE

The novelty proposed in this paper is an introduction of new type of solutions into the population of size M. The difference between the regular members of the population in the DE algorithm and the new ones lies in the way they are mutated. The regular individuals undergo classic mutation, that is, the mutation typical for DE with rules of perturbation based on the base solution and two difference ones as described above. The new solutions are modified with use of the mutation operator (similar to the one presented in [6]) which is based on the rules of movement governing quantum particles in mQSO [7].

In our approach a small number of new solutions (just one, two, or three pieces) replace the classic ones so the population size remains unchanged. In the subpopulation the new

solutions coexist with the classic ones and undergo just the same procedures of selection and suppression.
In the first phase of the new mutation, we generate a new solution uniformly distributed within a unit hypersphere surrounding the solution to be mutated. In the second phase, the generated solution is shifted along the direction determined by the hypersphere center and the solution. The distance d' from the hypersphere center to the final location of the solution is a random variable defined as:

$$d' = d\,S\alpha S(0,\sigma)\exp(-f'(\mathbf{x}_i)), \tag{1}$$

where: d is a distance from the location obtained in the first phase, $S\alpha S(\cdot,\cdot)$ denotes a symmetric α-stable distribution variate, σ and $f'(\mathbf{x}_i)$ are defined as in Eq. (2) and Eq. (3) respectively:

$$\sigma = r_{S\alpha S}\,(D_{\mathrm{w}}/2)\,, \tag{2}$$

$$f'(\mathbf{x}_i) = \frac{f(\mathbf{x}_i) - f_{\min}}{(f_{\max} - f_{\min})}\,, \tag{3}$$

where:

$$f_{\max} = \max_{j=1,\ldots,M} f(\mathbf{x}_j) \quad \text{and} \quad f_{\min} = \min_{j=1,\ldots,M} f(\mathbf{x}_j)\,,$$

where: D_{w} is the width of the feasible part of the search space, i.e., the distance between its lower and upper boundaries and $f(\mathbf{x}_j)$ is the value of the j-th solution.
The α-stable distribution (called also a Levý distribution) is controlled by four parameters: stability index α ($\alpha \in (0,2]$), skewness parameter β, scale parameter σ and location parameter μ. In our case we assume $\mu = 0$ and apply the symmetric version of this distribution (denoted by $S\alpha S$ for "symmetric α-stable distribution"), where β is set to 0.
The resulting behavior of the proposed operator is characterized by two parameters: the parameter $r_{S\alpha S}$ which controls the mutation strength, and the parameter α which determines the shape of the α-stable distribution. Solutions mutated in this way are labeled as $\mathbf{s}_{\text{Levý}}$ in the further text.

4. The Syringa Benchmark Suite

For the experimental research we developed a new testing environment Syringa which is able to simulate behavior of a number of existing benchmarks and to create completely new instances as well. The structure of the Syringa code originates from a fitness landscape model where the landscape consists of a number of simple components. A sample dynamic landscape consists of a number of components of any types and individually controlled by a number of parameters. Each of the components covers a subspace of the search space. The final landscape is the result of a union of a collection of components such that each of the solutions from the search space is covered by at least one component. In the case of a solution belonging to the intersection of a number of components the solution value equals

- the minimum (for minimization problems),
- the maximum (otherwise) value among the values obtained for the intersected components,
- the sum of the fitness vales obtained from these components.

Eventually, the Syringa structure is a logical consequence of the following assumptions:

1. The fitness landscape consists of a number of any different component landscapes.
2. The dynamics of each of the components can be different and individually controlled.
3. A component can be defined for a part or the whole of the search space, thus, in the case of a solution covered by more than one component the value of this solution can be the minimum, the maximum or the sum of values returned by the covering components.

4.1. The Components

Current version of Syringa consists of six types of component functions (Table 1) defined for the real-valued search space. All formulas include the number of the search apace dimensions n which makes them able to define search spaces of any given complexity.
There can be defined a number of parameters which individually define the component properties and allow to introduce dynamics as well. For each of the components we can define two groups of parameters which influence the formula of the component fitness function: the parameters from the former one are embedded in the component function formula whereas the parameters from the latter one control rather the output of the formula application. For example, when we want to stretch the landscape over the search space each of the solution coordinates is multiplied by a scaling factor. For a non–uniform stretching we need to use a vector of factors containing individual values for each of the coordinates. We call this type of modification a horizontal scaling and this represents the first type of component changes. The example of the second type is a vertical scaling where just the fitness value of a solution is multiplied by a scaling factor. The first group of parameters controls changes like horizontal translation, horizontal scaling, and rotation. For simplicity they are called *horizontal changes* in the further text. The second group of changes (called respectively *vertical changes*) is represented by vertical scaling and vertical translation. All of the changes can be obtained by dynamic modification of respective parameters during the process of search.

Table 1
`Syringa` components

Name	Formula	Domain
Peak (F_1)	$f(\mathbf{x}) = \frac{1}{1+\sum_{j=1}^{n} x_j^2}$	[–100,100]
Cone (F_2)	$f(\mathbf{x}) = 1 - \sqrt{\sum_{j=1}^{n} x_j^2}$	[–100,100]
Sphere (F_3)	$f(\mathbf{x}) = \sum_{i=1}^{n} x_i^2$	[–100,100]
Rastrigin (F_4)	$f(\mathbf{x}) = \sum_{i=1}^{n} \left(x_i^2 - 10\cos(2\pi x_i) + 10\right)$	[–5,5]
Griewank (F_5)	$f(\mathbf{x}) = \frac{1}{4000}\sum_{i=1}^{n}(x_i)^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	[–100,100]
Ackley (F_6)	$f(\mathbf{x}) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e$	[–32,32]

4.2. Horizontal Change Parameters

In this case the coordinates of the solution **x** (a vector, that is, a matrix of size n by 1) are modified before the component function equation is applied. The new coordinates are obtained with the following formula:

$$\mathbf{x}' = \mathbf{M}\left(\mathbf{W}\left(\mathbf{x}+\mathbf{X}\right)\right) \tag{4}$$

where: **X** is a translation vector, **W** is a diagonal matrix of scaling coefficients for the coordinates, and **M** is an orthogonal rotation matrix.

4.3. Vertical Change Parameters

Changes of the fitness function value are executed according to the following formula:

$$f'(\mathbf{x}) = f(\mathbf{x})\cdot v + h \tag{5}$$

where: v is a vertical scaling coefficient and h is a vertical translation coefficient.

4.4. Parameters control

In the case of dynamic optimization the fitness landscape components has to change the values of their parameters during the process of search. There were defined four different characteristics of variability which were applied to the component parameters: small step change ($\mathbf{T}_1$ – Eq. (7)), large step change ($\mathbf{T}_2$ – Eq. (8)), and two versions of random changes ($\mathbf{T}_3$ – Eq. (9) and $\mathbf{T}_4$ – Eq. (10)). The change Δ of a parameter value is calculated as follows:

$$\Delta = \alpha r(\max - \min), \tag{6}$$

$$\text{where } \alpha = 0.04,\ r = U(0,1), \tag{7}$$

$$\Delta = (\alpha\cdot sign(r_1) + (\alpha_{\max} - \alpha)r_2)\cdot(\max - \min),$$

$$\text{where } \alpha = 0.04,\ r_{1,2} = U(0,1),\ \alpha_{\max} = 0.1 \tag{8}$$

$$\Delta = N(0,1), \tag{9}$$

$$\Delta = U(r_{\min}, r_{\max}) \tag{10}$$

In the above-mentioned equations max and min represent upper and lower boundary of the search space, $N(0,1)$ is a random value obtained with standardized normal distribution, $U(a,b)$ is a random value obtained with uniform distribution form the range $[a,b]$, and $[r_{\min}; r_{\max}]$ define the feasible range of Δ values.

The model of `Syringa` assumes that the component parameter control is separated from the component, that is, a dynamic component has to consist of two objects: the first one represents an *evaluator* of solutions (that is, a component of any type mentioned in Table 1) and the second one is an *agent* which controls the behavior of the evaluator. The agent defines initial set of values for the evaluator parameters and during the process of search the values are updated by the agent according to the assumed characteristic of variability. Properties of all the types of components are unified so as to make possible assignment of any agent to any component. This architecture allows to create multiple classes of dynamic landscapes. In the presented research we started with simulation of two existing benchmarks: Generalized Dynamic Benchmark Generator (GDBG) [8] and the Moving Peaks Benchmark generator [9]. In both cases optimization is carried out in a real-valued multidimensional search space, and the fitness landscape is built of multiple component functions controlled individually by their parameters. For appropriate simulation of any of the two benchmarks there are just two things to do: select a set of the components and build agents which will control the components in the same manner like in the simulated benchmark.

4.5. Reimplementation of Moving Peaks Benchmark (MPB)

In the case of MPB, three scenarios of the benchmark parameters control are defined [9]. We performed experiments for two versions of just the second scenario. In this scenario the fitness landscape has been defined for the five-dimensional search space with the same boundaries for each dimension, namely $[-50; 50]$.

The selected fitness landscape consists of a set of cones (F_2) which undergo two types of horizontal changes: the trans-

lation and the scaling and just one vertical change, that is, the translation. The horizontal scaling operator has the same scale coefficient for each of the dimensions, so in this specific case this coefficient is represented as a one-dimensional variable w instead of the vector $\mathbf{W}$. Parameters $\mathbf{X}$, w and h are embedded into the cone function formula $f(\mathbf{x})$ in the following way:

$$f(\mathbf{x}) = h - w \cdot \sqrt{\sum_{j=1}^{n} (\mathbf{x}_j - \mathbf{X}_j)^2}. \tag{11}$$

All the modifications of the component parameters belong to the fourth characteristic of variability $\mathbf{T}_4$ where for every change $r_{\min}$ and $r_{\max}$ are redefined in the way to keep the value of each modified parameter in the predefined feasible interval $[f_{\min}; f_{\max}]$. Simply, for every modified parameter of translation or scaling, which can be represented as a symbol p, $r_{\min}$ equals $p_{\min} - p$ and $r_{\max}$ equals $p_{\max} - p$. For the horizontal scaling the interval is set to $[1; 12]$ and for the vertical scaling – to $[30; 70]$. For the horizontal translation there is a constraint for the Euclidean norm of the translation vector: $|\mathbf{X}| \leq 3$. In the first version of the scenario 2 there are ten moving cones whereas in the second version 50 moving cones are in use. Besides, we extended the number of the search space dimensions and performed our tests also for $n = 10, 20, 30$ for both versions of the selected benchmark instance.

4.6. Reimplementation of Generalized Dynamic Benchmark Generator (GDBG)

GDBG consists of two different benchmarks: Dynamic Rotation Peak Benchmark Generator (DRPBG) and Dynamic Composition Benchmark Generator (DCBG). There are five types of component functions: peak (F_1), sphere (F_3), Rastrigin (F_4), Griewank (F_5), and Ackley (F_6). F_1 is the base component for DRPBG whereas all the remaining types are employed in DCBG.

4.6.1. Dynamic Rotation Peak Benchmark Generator (DRPBG)

There are four types of the component parameter modification applied in DRPBG: horizontal translation, scaling and rotation, and vertical scaling. As in the case of MPB the horizontal scaling operator has the same scale coefficient for each of the dimensions, so in this specific case this coefficient is also represented as a one-dimensional variable w instead of the vector $\mathbf{W}$. The parameters $\mathbf{X}$, w and v are embedded into the peak function formula $f(\mathbf{x})$ in the following way:

$$f(\mathbf{x}) = \frac{v}{1 + w \sum_{j=1}^{n} \frac{(\mathbf{x}_j - \mathbf{X}_j)^2}{n}}. \tag{12}$$

Values of the translation vector $\mathbf{X}$ in subsequent changes are evaluated with use of the rotation matrix $\mathbf{M}$. Clearly, we apply the rotation matrix to the current coordinates of the component function optimum $\mathbf{o}$, that is: $\mathbf{o}(t+1) = \mathbf{o}(t) \cdot \mathbf{M}(t)$ (where t is the number of the current change in the component) and then the final value of $\mathbf{X}(t+1)$ is calculated: $\mathbf{X}(t+1) = \mathbf{o}(t+1) - \mathbf{o}(0)$.

Subsequent values of the horizontal scaling parameter w and the vertical scaling parameter v are evaluated according to the first, the second or the third characteristic of variability, that is, $\mathbf{T}_1$, $\mathbf{T}_2$ or $\mathbf{T}_3$.

For every change a new rotation matrix $\mathbf{M}$ is generated which is common for all the components. The rotation matrix M is obtained as a result of multiplication of a number of rotation matrices R where each of R represents rotation in just one plane of the multidimensional search space. A matrix $R_{ij}(\theta)$ represents rotation by the θ angle along the plane i–j and such a matrix can be easily generated as described by [10]. In DRPBG we start with a selection of the rotation planes, that is, we need to generate a vector $\mathbf{r}$ of size l where l is an even number and $l \leq n/2$. The vector contains search space dimension indices selected randomly without repetition. Then for every plane defined in $\mathbf{r}$ by subsequent pairs of indices: $[1,2]$, $[3,4]$, $[5,6]$, $\ldots$ $[l-1,l]$ a rotation angle is randomly generated and finally respective matrices $R_{r[1],r[2]}, \ldots R_{r[l-1],r[l]}$ are calculated. Eventually, the rotation matrix M is calculated as follows:

$$M(t) = R_{r[1],r[2]}(\theta(t)) R_{r[3],r[4]}(\theta(t)) \cdots R_{r[l-1],r[l]}(\theta(t)).$$

In `Syringa` the method of the rotation matrix generation slightly differs from the one described above. Instead of the vector $\mathbf{r}$ there is a vector Θ which represents a sequence of rotation angles for all the possible planes in the search space. The position in the vector Θ defines the rotation plane. Simply, $\Theta(1)$ represents the plane [1,2], $\Theta(2)$ represents the plane [2,3] and so on until the plane [n-1,n]. The next values in Θ represent planes created from every second dimensions, that is, [1,3], [2,4] and so on until the plane [n-2,n]. Then values in Θ represent planes created from every third dimensions, then those created from every fourth, and so on until there appears the value for the last plane created from the first and the last dimension. If $\Theta(i)$ equals zero, then there is no rotation for the i-th plane, otherwise the respective rotation matrix R is generated. The final stage of generation of the matrix M is the same as in the description above, that is, the rotation matrix M is the result of multiplication of all the matrices R generated from the vector Θ.

The matrix $\mathbf{M}$ is used twice for the evaluation of the component modification parameters: the first time when the translation vector $\mathbf{X}$ is calculated and the second time when the rotation is applied, that is, just before the application of the Eq. (12).

4.6.2. Dynamic Composition Benchmark Generator (DCBG)

DCBG performs five types of the component parameter modification: horizontal translation, scaling and rotation and vertical translation and scaling. The respective para-

meters are embedded into the function formula $f''(\mathbf{x})$ in the following way [11], [12]:

$$f''(\mathbf{x}) = (v \cdot (f'(\mathbf{M} \cdot (\mathbf{W} \cdot (\mathbf{x} + \mathbf{X}))) + h)) \tag{13}$$

where: v is the weight coefficient depending of the currently evaluated $\mathbf{x}$, $\mathbf{W}$ is called a stretch factor which equals 1 when the search range of $f(\mathbf{x})$ is the same as the entire search space and grows when the search range of $f(\mathbf{x})$ decreases, $f'(\mathbf{x})$ represent the value of $f(\mathbf{x})$ normalized in the following way: $f'(\mathbf{x}) = C \cdot f(\mathbf{x})/|f_{\max}|$ where the constant $C = 2000$ and $f_{\max}$ is the estimated maximum value of function f which is one of the four: sphere (F_3), Rastrigin (F_4), Griewank (F_5), or Ackley (F_6).

In `Syringa` the properties of some of the parameters has had to be changed. The first difference is in the evaluation of the weight coefficient v which due to the structure of the assumed model cannot depend of the currently evaluated $\mathbf{x}$. Therefore, we assumed that $v = 1$. There is also no scaling, that is, $\mathbf{W}$ is an identity matrix because we assumed that the component functions are always defined for the entire search space. The last issue is about the rotation matrix $\mathbf{M}$ which is calculated in the same way as for the `Syringa` version of DRPBG. Eventually, the `Syringa` version of $f''(\mathbf{x})$ looks as follows:

$$f''(\mathbf{x}) = ((f'(\mathbf{M} \cdot ((\mathbf{x} + \mathbf{X}))) + h)) \tag{14}$$

Thus, the `Syringa` version of DCBG differs from the original one because it does not contain the horizontal scaling, the rotation matrix $\mathbf{M}$ is evaluated in the different way and the stretch factor always equals one. However, a kind of the vertical scaling is still present and can be found in the step of the $f(\mathbf{x})$ normalization.

5. Plan of Experiments

There were performed two groups of experiments. In the first one we applied the pure version of jDE, whereas in the second one the version of jDE extended by $\mathbf{s}_{\text{Levý}}$ solutions was in use. The main aim of the first group was to study the effectiveness of the algorithm for testing environments with different complexity of the search space. The second group of experiments showed the influence of $\mathbf{s}_{\text{Levý}}$ solutions on the search process and the quality of the obtained results.

Experiments form the first group were performed with a subset of GDBG benchmark instances as well as with two versions of MPB scenario 2, for different numbers of the search space dimensions. The tests based on MPB were repeated twice. First, the tests were performed with the number of fitness function calls between subsequent changes defined as it was proposed for the CEC'09 competition, that is, for $10^4 \cdot n$ fitness function calls (n is a number of search space dimensions). Then the tests were repeated – for a fixed number of 5000 calls as it is recommended for experiments with MPB [9]. The tests based on GDBG were performed once just for $10^4 \cdot n$ fitness function calls between subsequent changes.

The second group includes experiments with a subset of GDBG benchmark functions as well as with four versions of MPB and for different numbers of $\mathbf{s}_{\text{Levý}}$ solutions present in the population. All the experiments in this group were performed for $10^4 \cdot n$ fitness function calls between subsequent changes. However, it must be stressed that in this group the number of search space dimensions was constant and equal five for every test-case.

To decrease the number of algorithm configurations which would be experimentally verified in the second group of experiments we decided to fix the value of the parameter $r_{S\alpha S}$ and the only varied parameter was α. In the preliminary phase of experimental research we tested efficiency of the algorithm for different values of $r_{S\alpha S}$ and analyzed obtained values of error. Eventually, for GDBG $r_{S\alpha S} = 0.6$ whereas for MPB it is ten times smaller, that is, $r_{S\alpha S} = 0.06$. Thus, for each of the benchmark instances there were performed just 32 experiments: for α between 0.25 and 2 varying with step 0.25 and for 0, 1, 2 and 3 solutions of new type present in the population.

For each of the parameter configurations the experiments were repeated 20 times and each of them consisted of 60 changes in the fitness landscape.

5.1. The Measures

We used measures of the obtained results proposed for both of the benchmarks by their authors. This gave opportunity for fair comparison of the algorithm efficiency. For GDBG there were defined four measures:

$$\begin{aligned}
Avg^{\text{best}} &= \sum_{i=1}^{N_{\text{exp}}} \min_{j=1,\ldots,N_{\text{ch}}} E_{\text{last}}^{i,j} / N_{\text{exp}}, \\
Avg^{\text{mean}} &= \sum_{i=1}^{N_{\text{exp}}} \sum_{j=1}^{N_{\text{ch}}} E_{\text{last}}^{i,j} / (N_{\text{exp}} \cdot N_{\text{ch}}), \\
Avg^{\text{worst}} &= \sum_{i=1}^{N_{\text{exp}}} \max_{j=1,\ldots,N_{\text{ch}}} E_{\text{last}}^{i,j} / N_{\text{exp}}, \\
STD &= \left(\frac{1}{N_{\text{exp}} \cdot N_{\text{ch}} - 1} \sum_{i=1}^{N_{\text{exp}}} \sum_{j=1}^{N_{\text{ch}}} (E_{\text{last}}^{i,j} - Avg^{\text{mean}})^2 \right)^{-1},
\end{aligned}$$

where: N_{exp} is a number of repeated experiments for the same control parameter settings of the algorithm, N_{ch} is a number of changes in the fitness landscape appearing during a single experiment run, and E_{last}^{j} is an absolute function error value:

$$E_{\text{last}}^{j} = |f(\mathbf{x}_{\text{best}}^{j}) - f(\mathbf{x}^{*j})|,$$

where: $\mathbf{x}_{\text{best}}^{j}$ – current best solution which has been found since the last j-th change in the fitness landscape, $\mathbf{x}^{*j}$ – real optimum solution for the fitness landscape after the j-th change.

In the case of MPB most of the publications contain the values of offline error measure (oe) obtained for performed

experimental research. The offline error represents the average deviation from the optimum of the best solution value since the last change in the fitness landscape. Formally:

$$oe = \frac{1}{N_{\mathrm{ch}}} \sum_{j=1}^{N_{\mathrm{ch}}} \left(\frac{1}{N_{\mathrm{e}}(j)} \sum_{i=1}^{N_{\mathrm{e}}(j)} (f(\mathbf{x}^{*j}) - f(\mathbf{x}_{\mathrm{best}}^{ji})) \right) ,$$

where: $N_{\mathrm{e}}(j)$ is a total number of solution evaluations performed for the j-th static state of the landscape. The measure *oe* should be minimized, that is, the better result the smaller the value of *oe*.

6. The Results

6.1. The First Group of Experiments – with the Pure Version of jDE

Majority of the results from the first group of experiments have already been reported in [13]. One of the more important observations was about a sensitivity of the Avg^{mean} measure to the complexity of the search space. Graphs in Fig. 1 depicts values of Avg^{mean} obtained for the tests which were performed for $10^4 \cdot n$ fitness function calls between subsequent changes, that is, for the number of fitness function calls depending on the number of the search space dimensions. Three types of Avg^{mean} error value trends can be observed in Fig. 1. They show that the proposed linear dependency of the evaluation number on n:

- does not compensate growth of the errors (e.g., DRPBG problems with F_4 components (Rastrigin functions)),
- does compensate growth of the errors for some limited range (e.g., DRPBG problems with F_6 components (Ackley functions) or with F_3 components (sphere functions)),
- does compensate growth of the errors excessively (e.g, DRPBG problems with F_5 components (Griewank functions)).

The above-mentioned difference in the sensitivity of Avg^{mean} and the offline error inspired the second group of experiments. We wanted to compare the results obtained with these four measures for the jDE version extended by $\mathbf{s}_{\mathrm{Levý}}$ solutions optimizing in five-dimensional search space.

6.2. The Second Group of Experiments – with jDE Extended by $s_{\mathrm{Levý}}$ Solutions

In this group of experiments we observed values of the above-mentioned measures for the experiments with different number of $\mathbf{s}_{\mathrm{Levý}}$ solutions present in the algorithm. We did full range of experiments for different values of α and different numbers of $\mathbf{s}_{\mathrm{Levý}}$ solutions in subpopulations. Then we selected the best results obtained with the offline error measure *oe* and with Avg^{mean}. The results are gathered in two tables: Table 2 with the results for the cases where the best values of Avg^{mean} and *oe* were obtained for the same configurations of jDE parameters and Table 3 where the best values of Avg^{mean} and *oe* were obtained for different configurations. In the latter case the best values are printed in bold characters.

Tables 2 and 3 present list of all the benchmark instances and configurations of jDE where the best values of Avg^{mean} and *oe* were obtained for the instances as well as the values of all the performance measures. In most cases the best values of Avg^{mean} were obtained for jDE without extension (except for DCBG with F_5 and with all three types of characteristics of the parameters variability and three another cases: DRPBG $10 \times F_1$, $\mathbf{T}_1$, DRPBG $50 \times F_1$, $\mathbf{T}_3$ and MPB sc.2 with 50 cones) whereas there were many test-cases where the best values of *oe* were obtained for jDE extended with $\mathbf{s}_{\mathrm{Levý}}$ solutions (DCBG with F_3 or F_6 and with all three types of characteristics of the parameters variability, majority of DRPBG configurations and again MPB sc.2 with 50 cones). In the latter group of jDE configurations the best values of *oe* can be observed for jDE with just one $\mathbf{s}_{\mathrm{Levý}}$ solution existing in every subpopulation, however, there is an exception which is DCBG with F_5 where the best values were obtained for the highest numbers of $\mathbf{s}_{\mathrm{Levý}}$ solutions.

7. Conclusions

In this paper we studied properties of jDE applied to the number of dynamic optimization tasks. A new benchmark suite called `Syringa` was implemented which is able to simulate existing benchmarks and allows to create plenty of new classes of dynamic optimization problems as well. Two existing benchmarks were selected for experiments: MPB and DCBG. Selected subsets of test-case instances originated from the two benchmarks were reimplemented within the suite. Then, a number of tests was executed for these optimization tasks. As an optimization tool the self-adaptive differential evolution approach was selected.

7.1. Conclusions from the First Group of Experiments

In the first group we used a number of test-case instances of different complexity, that is, test-cases with the number of search space dimensions growing from five to 30. To compensate growth of errors accompanying growing complexity of the optimized problems the tests were performed with the number of fitness function calls defined as it was proposed for the CEC'09 competition. Obtained different error value trends for the applied measures revealed the search process features varying for the problems of growing complexity. In spite of the fact that in every case the value of *oe* increased as the number of search space dimensions grew [13] obtained mean values of the best found solutions have different characteristics. For some test-cases the compensation formula was sufficient to keep the values of the best found solution on the same level whereas for the others

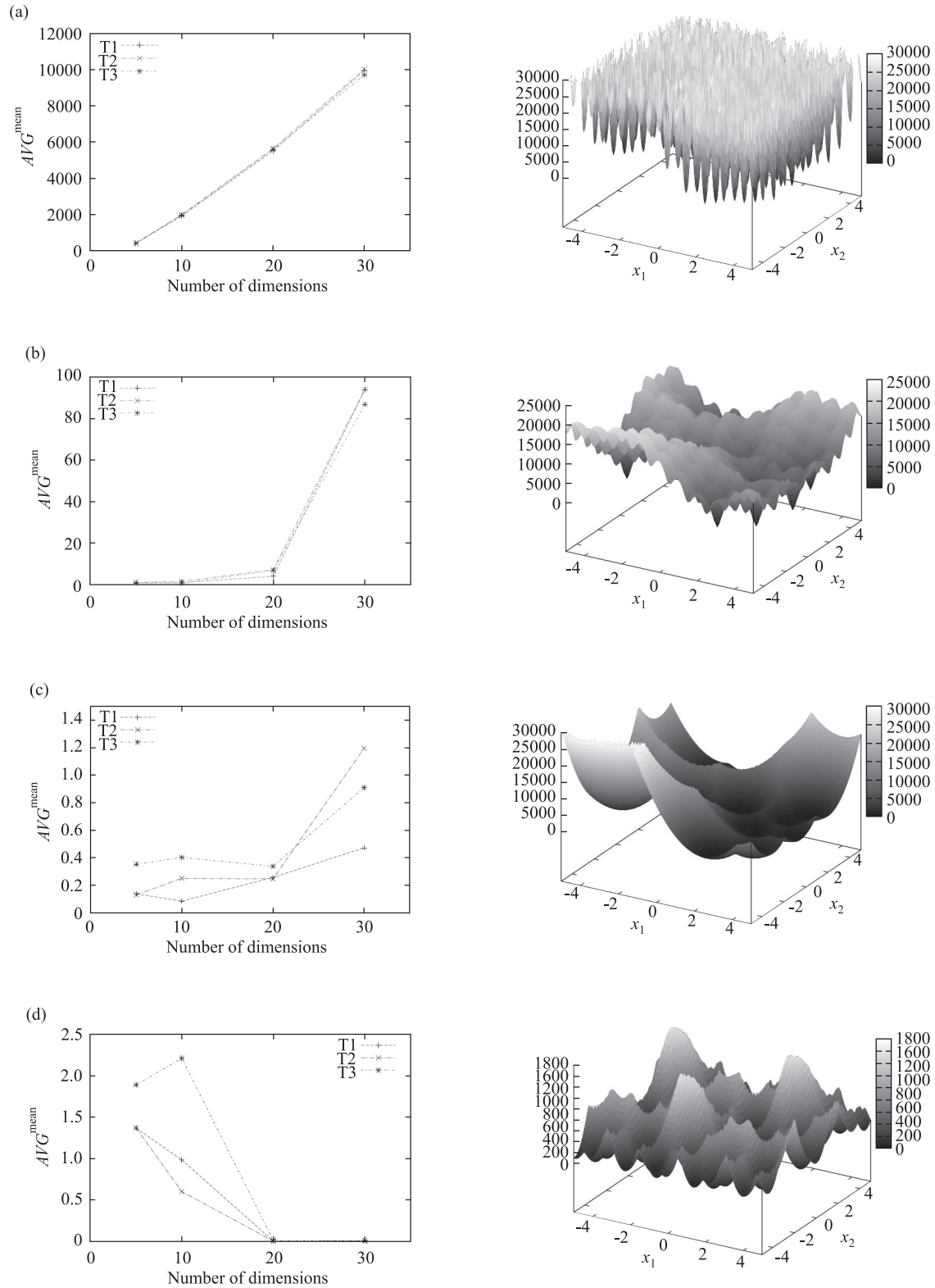


Fig. 1. *Avg*$^{\text{mean}}$ values (on the left) obtained for DCBG test-cases (respective fitness landscapes in 2D are on the right) with: (a) F_4 components (Rastrigin functions), (b) F_6 components (Ackley functions), (c) F_3 components (sphere functions), and (d) F_5 components (Griewank functions).

Table 2
The results for the cases where the best values of Avg^{mean} and *oe* were obtained for the same configurations of jDE parameters

Testing environment	α	$\mathbf{s}_{\text{Levý}}$ num.	Avg^{best}	Avg^{mean}	Avg^{worst}	*STD*	*oe*	Std. err
MPB sc.2, 10 cones	—	0	0	**2.929**	14.241	4.387	**4.111**	2.213
MPB sc.2, 50 cones	0.5	1	$8.39{\cdot}10^{-6}$	**2.730**	11.129	2.941	**3.749**	1.013
DCBG with F_4, $\mathbf{T}_1$	—	0	2.999	**368.31**	829.59	187.81	**570.72**	48.496
DCBG with F_4, $\mathbf{T}_2$	—	0	5.725	**414.39**	824.64	191.88	**610.56**	57.612
DCBG with F_4, $\mathbf{T}_3$	—	0	0.746	**402.62**	805.08	187.34	**661.39**	54.57

Table 3
The results for the cases where best values of Avg^{mean} and *oe* were obtained for different configurations of jDE parameters

Testing environment	α	$\mathbf{s}_{\text{Levý}}$ num.	Avg^{best}	Avg^{mean}	Avg^{worst}	*STD*	*oe*	Std. err
DRPBG $10 \times F_1$, $\mathbf{T}_1$	2	1	0	**0.004**	0.222	0.130	2.041	0.354
DRPBG $10 \times F_1$, $\mathbf{T}_1$	0.25	1	0	0.0208	0.878	0.312	**1.988**	0.688
DRPBG $10 \times F_1$, $\mathbf{T}_2$	—	0	0	**0.0023**	0.074	0.056	2.675	0.310
DRPBG $10 \times F_1$, $\mathbf{T}_2$	1	1	0	0.0361	1.189	0.353	**2.518**	0.689
DRPBG $10 \times F_1$, $\mathbf{T}_3$	—	0	0	**0.004**	0.264	0.141	3.985	0.678
DRPBG $10 \times F_1$, $\mathbf{T}_3$	0.5	1	0	0.058	1.644	0.728	**3.761**	1.012
DRPBG $50 \times F_1$, $\mathbf{T}_1$	—	0	0	**0.352**	4.899	1.190	3.601	0.641
DRPBG $50 \times F_1$, $\mathbf{T}_1$	1	1	0	0.653	6.976	1.471	**3.501**	0.523
DRPBG $50 \times F_1$, $\mathbf{T}_2$	—	0	0	**0.385**	5.816	1.188	4.318	0.529
DRPBG $50 \times F_1$, $\mathbf{T}_2$	1.75	1	0	0.703	6.532	1.633	**4.219**	0.659
DRPBG $50 \times F_1$, $\mathbf{T}_3$	0.75	1	0	**0.619**	7.006	2.134	5.875	1.860
DRPBG $50 \times F_1$, $\mathbf{T}_3$	1.5	1	0	0.836	6.912	1.853	**5.657**	1.214
DCBG with F_3, $\mathbf{T}_1$	—	0	0	**0.138**	3.715	0.822	2.087	0.536
DCBG with F_3, $\mathbf{T}_1$	1.5	2	$2.15{\cdot}10^{-6}$	1.249	9.288	3.032	**1.226**	0.487
DCBG with F_3, $\mathbf{T}_2$	—	0	0	**0.295**	7.664	1.673	3.290	0.860
DCBG with F_3, $\mathbf{T}_2$	0.75	1	0	0.452	7.679	1.978	**1.816**	0.506
DCBG with F_3, $\mathbf{T}_3$	—	0	0	**0.240**	4.692	0.932	8.482	1.827
DCBG with F_3, $\mathbf{T}_3$	1.75	3	0.0007	1.989	10.103	2.819	**5.325**	1.923
DCBG with F_5, $\mathbf{T}_1$	2	2	0.133	**2.113**	6.118	1.483	0.180	0.033
DCBG with F_5, $\mathbf{T}_1$	1.25	3	0.464	2.253	6.163	1.362	**0.168**	0.028
DCBG with F_5, $\mathbf{T}_2$	2	2	0.128	**1.847**	5.734	1.359	0.143	0.015
DCBG with F_5, $\mathbf{T}_2$	1.75	3	0.323	2.148	5.905	1.319	**0.133**	0.019
DCBG with F_5, $\mathbf{T}_3$	1.75	3	0.411	**2.364**	6.276	1.381	0.358	0.070
DCBG with F_5, $\mathbf{T}_3$	2	3	0.288	2.491	5.891	1.340	**0.356**	0.044
DCBG with F_6, $\mathbf{T}_1$	—	0	0	**0.545**	13.169	2.698	8.539	1.847
DCBG with F_6, $\mathbf{T}_1$	2	1	0.0004	1.031	11.291	2.790	**8.419**	2.811
DCBG with F_6, $\mathbf{T}_2$	—	0	0	**0.547**	11.992	2.238	11.381	2.077
DCBG with F_6, $\mathbf{T}_2$	1.5	1	0.0003	1.309	15.526	3.651	**9.773**	2.594
DCBG with F_6, $\mathbf{T}_3$	—	0	0	**0.590**	6.859	1.643	15.914	1.086
DCBG with F_6, $\mathbf{T}_3$	1.5	1	0.0005	1.420	8.799	2.922	**14.245**	2.167

it was not (Fig. 1). This allowed to identify some of the test cases as more difficult that the others.

The observations confirm that a new formula for the number of fitness evaluations applied to compensate growth of the errors is needed (probably different ones for different landscapes). However, different trends obtained for the two measures show also that the measures evaluate different features of the search process and thus probably it will be hard of even impossible to find the rule which would be able to compensate growth of both of the errors at the same time.

7.2. Conclusions from the Second Group of Experiments

In the second group of tests we studied influence of the new type of mutation operator on the search process effectiveness. A new type of solutions which undergo the new mutation procedure was introduced into the population. The presence of new solutions changed the population behavior in the way which allowed to explore faster new promising areas of the search space. However, from the other side the enhancement of exploring properties was the reason for less precise approaching of the population to the optimum. The effect of this can be observed in the results presented in Tables 2 and 3.

7.3. Summary

Obtained results show that application of both *oe* and Avg^{mean} in the experimental research analysis returns more information about the search process and shows more pro and cons about the tested algorithm than the application of just one of them. Our experiments and especially obtained values of *oe* showed that application of $\mathbf{s}_{\text{Levý}}$ solutions allows to explore and approach faster the promising areas of the search space in many benchmark instances. However, worse values of Avg^{mean} accompanying the best values of *oe* are the argument against this if we are interested in the highest values of the best found solutions. So, in unpredictable circumstances and especially when the available number of time, that is, the available number of fitness function evaluations is hardly predictable and it is probable that the estimated number can be unexpectedly shortened the better option is the maximization of *oe*. In the opposite case, that is, when the granted number of fitness function evaluations is sufficient and guaranteed, the strategy of Avg^{mean} maximization is much more reasonable.

Acknowledgments

This research has been partially supported by the European Regional Development Fund with the grant no. POIG.01.01.02-14-013/09: *Adaptive system supporting problem solution based on analysis of textual contents of available electronic resources*.

References

[1] Y. Jin and J. Branke, "Evolutionary algorithms in uncertain environments – a survey", *IEEE Trans. Evol. Comput.*, vol. 9, no. 3, pp. 303–317, 2005.

[2] V. Feokistov, "Differential evolution", in *Search of Solutions*, vol. 5 of *Optimization and Its Applications*. Springer, 2006.

[3] K. V. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution, A Practical Approach to Global Optimization*. Natural Computing Series. Springer, 2005.

[4] J. Brest, S. Greiner, B. Boskovic, M. Mernik, and V. Zumer, "Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems", *IEEE Trans. Evol. Comput.*, vol. 10, no. 6, pp. 646–657, 2006.

[5] J. Brest, A. Zamuda, B. Boskovic, M. S. Maucec, and V. Zumer, "Dynamic optimization using self-adaptive differential evolution", in *IEEE Congr. Evol. Comput.*, pp. 415–422. IEEE, 2009.

[6] A. Obuchowicz and P. Pretki, "Isotropic symmetric α-stable mutations for evolutionary algorithms", in *Proc. Congr. Evol. Comput.*, vol. 1, pp. 404–410. IEEE Press, 2005.

[7] K. Trojanowski, "Properties of quantum particles in multi-swarms for dynamic optimization", *Fundamenta Informaticae*, vol. 95, no. 2–3, pp. 349–380, 2009.

[8] C. Li and S. Yang, "A generalized approach to construct benchmark problems for dynamic optimization", in *Simulated Evolution and Learning, 7th Int. Conf. SEAL 2008*, Melbourne, Australia, 2008, vol. 5361 of *LNCS*, pp. 391–400. Springer, 2008.

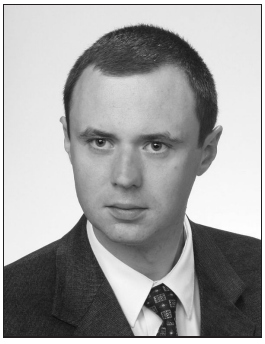
[9] J. Branke, "Memory enhanced evolutionary algorithm for changing optimization problems", in *Proc. Congr. on Evol. Comput.*, vol. 3, pages 1875–1882. IEEE Press, Piscataway, NJ, 1999.

[10] R. Salomon, "Reevaluating genetic algorithm performance under coordinate rotation of benchmark functions", *BioSystems*, vol. 39, no. 3, pp. 263–278, 1996.

[11] J. J. Liang, P. N. Suganthan, and K. Deb, "Novel composition test functions for numerical global optimization", in *Proc. IEEE Swarm Intelligence Symp.*, Pasadena, CA, USA, 2005, pp. 68–75.

[12] P. N. Suganthan *et al.*, "Problem definitions and evaluation criteria for the cec 2005 special session on real-parameter optimization", Tech. Rep., Nanyang Technological University, Singapore, 2005.

[13] M. Raciborski, K. Trojanowski, and P. Kaczyński, "Differential evolution for high scale dynamic optimization", in *Security and Intelligent Information Systems*, LNCS. Springer, 2011 (will appear in vol. 7053).

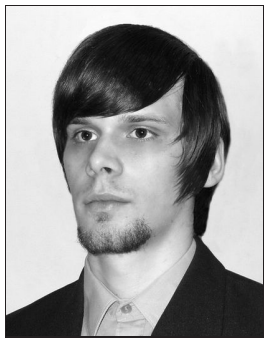


Krzysztof Trojanowski received the M.Sc. degree in Computer Science from the Warsaw University of Technology, Poland, in 1994, the Ph.D. degree from the Institute of Computer Science of the Polish Academy of Sciences (ICS PAS), Warsaw, Poland, in 2000, and the habilitation degree – also from ICS PAS in 2010. Since 1994 he has been with the Institute of Computer Science PAS. From 1995 to 2002 we was a consultant at IT companies in Poland. From 2002 to 2008 he was with University of Podlasie, Siedlce. Currently, he is with the Cardinal Stefan Wyszyński University, Warsaw, where he is an Associate Professor. From 2003 until 2008 he was a chairman of the Organizing Committee of the annual international conference Intelligent Information

Systems organized by ICS PAS. From 2006 until 2008 he was also a chairman of the International Conference on Artificial Intelligence, the oldest AI conference in Poland. His research interests include heuristic optimization techniques, such as evolutionary algorithms, immune systems, and particle swarm optimization.
E-mail: trojanow@ipipan.waw.pl
Institute of Computer Science
Polish Academy of Sciences
Ordona 21 01-237 Warsaw, Poland

Piotr Kaczyński received the M.Sc. degree in Computer Science from the Warsaw University of Technology, Poland and the M.Sc. degree in Theoretical Mathematics from Cardinal Stefan Wyszyński University in Warsaw, Poland, both in 2006. Since then he has been with the Cardinal Stefan Wyszyński University as a teaching assistant and also with various IT companies in Poland working with mathematical modeling and simulation. His research interests include both artificial intelligence applications (such as neural networks, evolutionary algorithms) and stochastic control.

E-mail: p.kaczynski@uksw.edu.pl
Faculty of Mathematics and Natural Sciences
College of Sciences
Cardinal Stefan Wyszyński University
Wóycickiego st 1/3
01-938 Warsaw, Poland



Mikołaj Raciborski received the B.Sc. degree in Natural Sciences, with a specialization of Computer Science in Applications in 2009 and the M.Sc. degree in Computer Science in 2011 both at Cardinal Stefan Wyszyński University, Faculty of Mathematics and Natural Sciences, College of Sciences. His research interests include computer science, mathematics, physics, chemistry, and electronic services.
E-mail: mikolaj.raciborski@gmail.com
Faculty of Mathematics and Natural Sciences
College of Sciences
Cardinal Stefan Wyszyński University
Wóycickiego st 1/3
01-938 Warsaw, Poland

Paper

Improving Population-Based Algorithms with Fitness Deterioration

Adrian Wolny and Robert Schaefer

Department of Computer Science, AGH University of Science and Technology, Kraków, Poland

Abstract—This work presents a new hybrid approach for supporting sequential niching strategies called *Cluster Supported Fitness Deterioration* (CSFD). Sequential niching is one of the most promising evolutionary strategies for analyzing multimodal global optimization problems in the continuous domains embedded in the vector metric spaces. In each iteration CSFD performs the clustering of the random sample by OPTICS algorithm and then deteriorates the fitness on the area occupied by clusters. The selection pressure pushes away the next-step sample (population) from the basins of attraction of minimizers already recognized, speeding up finding the new ones. The main advantages of CSFD are low memory an computational complexity even in case of large dimensional problems and high accuracy of deterioration obtained by the flexible cluster definition delivered by OPTICS. The paper contains the broad discussion of niching strategies, detailed definition of CSFD and the series of the simple comparative tests.

Keywords—basin of attraction, clustering, fitness deterioration, genetic algorithm, OPTICS, sequential niching.

1. Introduction

1.1. Global Optimization Problems in Metrizable Domains

We deal with the class of *global optimization problems* (GOP) which are leading to find all global, or even all local minimizers to the real-valued objective function defined on the set $\mathcal{D}$ (called admissible set) embedded in a finite dimensional normed vector space $V \supset \mathcal{D}$, $dim(V) = N < +\infty$, where the norm induces the complete metric (Banach space). The set $\mathcal{D}$ is assumed to be continuous with respect to the metric and regular in some way, usually having the Lipshitz boundary. Such spaces are also equipped with the Lesbegue measure based on the metric (see [1] for details).

Typical difficulties appearing by solving GOPs are the huge volume of $\mathcal{D}$, multimodality of the objective and its weak regularity (sometimes only continuity, or even discontinuity on the subset of the zero measure).

Stochastic, population-based heuristics (Monte Carlo, Evolutionary Algorithms, Simulated Annealing, Ant Colony, Particle Swarm, etc.) are best suited to solve GOPs and they frequently outperform deterministic techniques (see e.g. [2]). Most of these strategies restrict their search to a finite subset $\mathcal{D}_r \subset \mathcal{D}$ or to the set of codes U (e.g., genetic universum of codes) bijectively mapped on $\mathcal{D}_r$. It results from the inherent finite property of computer calculations as well as from algorithmic reasons as in case of genetic algorithms (see e.g. [3]). Because the remaininder of this paper utilizes mainly genetic techniques we will denote by $F : \mathcal{D}\,(\text{or}\,\mathcal{D}_r, U) \to \mathbb{R}$ the generic *fitness function* that expresses the GOP objective.

One of the well known disadvantages of some population-based, stochastic heuristics (especially Genetic Algorithms) is the *premature convergence* which consists in long-term stay of almost all population in the basin of attraction of the single local minimum. Premature convergence dramatically decreases an ability of finding many local/global minima with the acceptable computational cost assumed as the number of objective evaluations.

The basin of attraction $\mathcal{B}_{x^+} \subset \mathcal{D}$ of the local or global minimizer $x^+ \in \mathcal{D}$ may be roughly described as the connected part of maximum fitness level set that contains x^+ and does not intersect with basins of attraction of other local minimizers. The precise mathematical definition of basin of attraction was introduced by [4], [5] and may be found also in [3].

1.2. Niching

Niching techniques constitute an important class of adaptive strategies that can prevent premature convergence. One of niching techniques, called *parallel niching* works by forcing individuals belonging to the single or several populations working in parallel to search in the basins of attractions of more then one local (global) minima. Basic information about this technology may be found in [6], [7] and [8].

Another possibility is to perform niching process sequentially (*sequential niching*) which forces the single population or the group of populations to move from the basins of attraction of minima that have been recognized so far.

1.3. Fitness Deterioration Techniques

The behavior of individuals suitable for niching may be obtained by the proper fitness modification that leads to its leveling on the central part of the basin of attraction of local minima already encountered. Selection removes individuals from such areas forcing them to find regions of smaller fitness, e.g., the basins of attraction of other minima not yet found. The fitness leveling mentioned above is sometimes called *fitness deterioration* [9] or *hill crunching* [10].

Sequential niching by fitness deterioration was introduced by Beasley, Bull and Martin [11]. Exhaustive research in this direction was performed by Obuchowicz, Patan and Korbicz. They introduced the class of evolutionary algorithms called *Evolutionary Search with Soft Selection – Deterioration of the Objective Function* (ESSS-DOF). The draft of this strategy is presented in Algorithm 1. The fitness modification performed in the line 8 of Algorithm 1 is based on the formula

$$\hat{F}(x) = F_{\min} \exp\left(-\sum_{i=1}^{N} \left(\frac{x_i - \bar{y}_i}{\sigma_i} \right)^2 \right), \qquad (1)$$

where $\bar{y}_i$ are coordinates of the expected centroid of phenotypes that correspond to the population P_t and σ_i^2 stands for the variance of the individuals location in the i^{th} direction, while $F_{\min} = F(x^*)$ is the minimum individual fitness that appears in the population P_t. Finally N is the dimension of the admissible domain $\mathcal{D}$.

Algorithm 1: Draft of the ESSS-DOF strategy

1: Create initial population P_0;
2: $t \leftarrow 0$;
3: **repeat**
4: Evaluate population P_t;
5: Distinguish the best fit individual x^* from P_t;
6: **if** ($trap\_test$) **then**
7: Memorize x^*;
8: $F \leftarrow F - \hat{F}$;
9: **end if**
10: Perform selection with the fitness F;
11: Perform genetic operations;
12: $t \leftarrow t + 1$;
13: **until** ($stop\_condition$)

The logical variable $trap\_test$ is true if the mean fitness in the population increases less than $p\%$ during the last n_{trap} genetic epochs, or the standard deviation of the phenotypes displacement is less than the mutation range during the last n_{trap} genetic epochs. The logical variable $stop\_condition$ is true if the proper stopping rule for the whole strategy is satisfied. The simplest possible stopping rule may be limit the number of genetic epochs after the last fitness modification during which no trap was found.

Test results of this effective approach to the global search were presented in [12], [13], [14]. Arabas delivered another formula for fitness modification that leads to population niching [7].

1.4. Using Clustering in Fitness Deteriorartion

Telega, Schaefer and Adamska (see [10], [15]) introduced the clustering techniques applied to the random sample (population) obtained by the genetic algorithm (GA) in order to make the fitness deterioration more accurate and effective. The resulted strategy was called *Clustered Genetic Search* (CGS).

The basic idea of CGS is to recognize the clusters of individuals contained in multiset of individuals being the current population or the sum of current populations obtained from the multi-deme model or being the cumulated population (e.g., the union of all populations from the prescribed number of last genetic epochs). Clusters should be sufficiently dense and well separated from each other.

Next the *cluster extensions* being the regular subsets of $\mathcal{D}$ with the positive measure containing the cluster points are constructed. Pseudocode of the proposed strategy is depicted in the Algorithm 2.

Algorithm 2: Draft of the CGS strategy

1: $CLE \leftarrow \emptyset$;
2: Create initial population P_0;
3: **repeat**
4: Evaluate fitness F outside CLE;
5: Modify fitness according to (2);
6: Perform GA until the local stooping criterion is satisfied;
7: Recognize new clusters;
8: Construct new cluster extensions and update CLE;
9: **until** (The whole domain $\mathcal{D}$ has been processed) **or** (A satisfactory set of cluster extension has been found)

The CGS strategy utilizes the following fitness modification

$$\hat{F}(x) = \begin{cases} F(x) & \text{if} \quad x \in \mathcal{D} \setminus CLE \\ F_{\max} & \text{if} \quad x \in CLE \end{cases}, \qquad (2)$$

where $F_{\max}$ is the maximum fitness value already encountered and $CLE \subset \mathcal{D}$ stands for the union of cluster extensions already recognized.

The admissible domain $\mathcal{D}$ is divided into hypercubes that constitute a grid (raster) and the cluster extensions are unions of hypercubes. Every cluster extension can be recognized in one or many steps of the main loop. After the GA is stopped (line 6 in Algorithm 2), new parts of cluster extensions can be detected by the analysis of the density of individuals in the hypercubes. The hypercube that contains the best individual is selected as the seed. Neighboring hypercubes with the density of individuals greater than an arbitrary threshold are attached to the cluster extension. A rough local optimization method is started in each new part of the cluster extension, and the result of this optimization is retained. If this local method ends in the already recognized cluster extension then this part is attached to it.

The local stopping criterion distinguishes two kinds of behavior of the GA utilized by CGS. The first one is that it finds new parts of cluster extensions after few generations, and the second is the chaotic behavior (individuals are uniformly distributed over $\mathcal{D} \setminus CLE$). The latter corresponds to the recognition of a plateau (or areas where the fitness has small variability) outside of the already known cluster extensions. Other cases are treated as the situation when

GA does not fit to the particular problem, and a refinement of SGA parameters is suggested.
The CGS stopping strategy is to check stagnation of a sequence of some estimator of distribution of population individuals. If it does not change, then check if an arbitrary number of hypercubes has the density of individuals below the threshold. If so, then begin the clustering procedure; otherwise, check if individuals are uniformly distributed.
Computations show that CGS constitutes a "filter" that eliminates local minima with small fitness variability and narrow basins of attraction. Such property can be useful in some cases. The CGS strategy should be especially convenient for functions with large areas of small variability (areas similar to plateaus) which can be difficult for other global and local optimization methods.
The Simple Genetic Algorithm (SGA) (see e.g. [3]) was used in the CGS instance described above. Another implementation utilized the multi-deme Hierarchic Genetic Search (HGS) (see [16]) and the FMM-EMM method for finding cluster extensions (see [17]).
The CGS works well if the cluster extensions fall into the basins of attraction of local and global minimizers and fill them sufficiently. The CGS correctness and the correctness of the proposed CGS stopping strategy can be partially verified for the case of SGA sampling (see [3], [15], [18]).

Summing up, CGS may be classified as the sequential niching, even if the parallel HGS is applied as the sampling engine and if more than one cluster extension is encountered in its single step.

1.5. Critical Remarks

Deep analysis of the deterioration techniques presented in Subsections 1.3, 1.4 exhibits their several serious disadvantages:

- Huge memory complexity of memorizing the cluster extensions which appears especially in case of CGS with a raster representation of cluster extensions and a large dimension N of the admissible domain $\mathcal{D}$. In the computational practice, the GOP with N greater then 10 may bring the memory problems.
- Unsatisfactory accuracy of fitness approximation on the area of cluster extensions that leads to the incorrect deterioration and finally may lead to removing individuals from unchecked areas or the multiple check of non-promising areas already browsed.
- In cases of both strategies described in Subsections 1.3, 1.4 the error has a different origin. In the case of ESS-DOF the approximation of fitness in area surrounding local minimum (see Eq. (1)) is very rough and might be unsatisfactory in case of elongated basins of attraction. No matter how CGS finds the area of fitness leveling quite good as the cluster extensions, the fitness modification by the general constant (see Eq. (2)) may result in artifacts (artificial minima) at the borders of cluster extensions.
- The strong dependency of deterioration technique from the evolutionary technique used which can be especially observed in ESS-DOF. This feature generally prevents the use of deterioration in the transparent way in case of complex, adaptive strategies with many genetic engines. Sometimes this dependency might be helpful by profiling deterioration according to the particular GA instance.

The above discussion clearly shows the way of necessary improvements. A deterioration technique that combines low memory complexity of the exponential fitness improvement $\hat{F}$ with the accuracy of clustering will be presented in following sections.

2. New Approach for Sequential Niching with Fitness Deterioration

We suggest another approach to sequential niching which exploits some of the ideas from [9], [10], [17] and which tries to overcome problems specified in Subsection 1.5. This strategy is called *Cluster Supported Fitness Deterioration* (CSFD).

2.1. Algorithm Description

In principle the algorithm works like the CGS strategy presented in [10]. It uses GA to obtain a random sample (population of individuals) from the domain $\mathcal{D}$. If the problem space contains some robust solution located inside broad basins of attraction, it is very likely that individuals returned from the GA will gather inside one or more of such basins, so the next step of the algorithm is to apply the clustering algorithm to distinguish groups of individuals (clusters) from the population. Under the assumption that distribution of individuals inside a given cluster provides information about the topology of the basin in which the cluster resides, the algorithm approximate the basin using this information in order to deteriorate the fitness over the basins area and thus to prevent convergence to the same solutions multiple times.

2.2. Cluster versus Cluster Extension

Let us make a clear distinction between two important notions of the *cluster extension* and the *cluster* itself, which are necessary for further research.

Definition 1: We will call by *cluster* C a selected subset of the population P (the mulitset of individuals) returned by GA. We assume that each cluster will be disjoint with the other cluster from P (i.e., when C_i, C_k are clusters, then $i \neq k \Longrightarrow C_i \cap C_k = \emptyset$).

The individuals are assigned to the clusters using the method described in Section 3. Generally, individuals that belong to a particular cluster C have to be densely distributed and well separated from individuals from other clusters.

Definition 2: The *extension CE* of the cluster C is the connected and regular subset of the admissible domain $\mathcal{D}$ with the positive measure, containing the cluster points, i.e., each element from C belongs to CE. Regular set means here the set with the Lipshitz boundary (see e.g. [1]).

Cluster extensions may be obtained in many different ways. One of them is the raster procedure introduced by [15] described in Subsection 1.4.

Here we will use the cluster extensions CE being the ellypsoid whose middlepoint is located at the cluster C centroid and the measures of diversity of individuals inside C (e.g. in the form of standard deviation) in order to determine the length and the direction of its axes (see Section 4).

The CSFD strategy runs the GA and then distinguishes clusters from the resulting population and constructs their extensions in each iteration. The information contained in each cluster allows for effective finding the good approximation of the local minimizer x^+ contained in its extension (e.g., by running the accurate local method starting from the best fitted individual in the cluster). Another advantage is the ability to approximate the shape and the volume of basins of attraction $\mathcal{B}_{x^+}$ by cluster extensions. This information is utilized by the fitness deterioration technique (see Section 4). It might be also helpful for the stability analysis of x^+.

It may happen that cluster extensions obtained from two or more clusters lie inside the same basin of attraction. This may happen when in one iteration we manage to deteriorate only a part of the basin of attraction and in the next iteration we obtain the cluster inside other part of the basin. We will use the procedure suggested by [15] in order to detect such situation starting a single local, cheep method from each new cluster extension. If the method converges inside one of existing cluster extensions CE, then CE and CE' are joined (which indicates the fact that both CE' and CE lie in the same basin).

2.3. CSFD Pseudo-Code

The CSFD strategy takes a hybrid approach which uses an arbitrary GA to obtain a random sample from which it tries to extract as much information as possible in order to find approximations of basins of attraction. The only need for the GA is to be well tuned to the GOP to be solved, i.e., the population has to concentrate in a basin of attraction of at least one local robust minimizer (see [3]). Then the fitness deterioration is applied in localized areas to prevent exploration of the same basins multiple times during the course of the search. Algorithm 3 shows the general idea behind the Cluster Supported Fitness Deterioration. The algorithm components will be described in a top-down manner in next sections, here we provide the general overview only.

The condition in "while" statement (line 2, Algorithm 3) should be treated as a control statement rather than the real termination criterion. The CSFD stopping criterion is based on the conditions checked inside the main loop (line 6, Algorithm 3). If the clustering algorithm has found no group of similar individuals or the deterioration has a low quality, CSFD is stopped and returns all clusters found.

Algorithm 3: Draft of the CSFD strategy

```
1:  CL ← ∅
2:  while i < maxGenerationNumber do
3:    execute(GA)
4:    pop ← getPopulation(GA)
5:    clusterStruct ← extractClusterStruct(pop)
6:    if noClusters(clusterStruct) then
7:      return
8:    end if
9:    detFitness ←
      performFitnessDet(clusterStruct, currentFitness)
10:   if quality(detFitness, currentFitness, pop) < th
      then
11:     return
12:   end if
13:   CL ← CL ∪ clusters
14:   updateFitness(detFitness)
15:   popSize = popSize + indNum
16: end while
```

In each iteration we execute the GA (line 3, Algorithm 3) which is treated as a black-box algorithm, i.e., we do not need to modify or know the implementation of the GA used. Then we take the population returned by the GA and extract so called *clustering structure* (described with details in Subsection 3.1) which contains the information about clusters and their internal mean densities. Note that so far we have not clustered the population returned by the GA, instead we extract the information about the internal clustering structure for further processing. If the clustering structure contains promising clusters then we perform the *fitness deterioration* – the process of degradation of the fitness landscape in areas occupied by clusters of individuals which are assumed to agglomerate inside basins of attraction. In the line 9, Algorithm 3 we execute the complex procedure *performFitnessDet* (described in details in Subsection 3.1) which actually performs the *clustering* and *fitness deterioration*. Next in the line 10, Algorithm 3 we check if the new fitness (returned by the procedure *performFitnessDet*) fulfills the quality requirements described in Section 4. Finally, we save the clusters returned by the deterioration process and update the fitness function for further iterations (lines 13, 14, Algorithm 3). Exploratory capabilities can be increased during later iterations by increasing the population size of the GA (line 15, Algorithm 3). The CSFD strongly depends on the clustering algorithm used as a way to find parts of basins of attraction and as a global termination criterion for the CSFD algorithm (see Subsection 2.4).

2.4. Termination Conditions

Two types of stopping and termination criterions are used by CSFD:

- *Local termination criterion* which is used as a stopping criterion for the GA inside the main loop of the CSFD.
- *Global stopping criterion* which is based on the result of the clustering algorithm applied to the population returned by the GA and which finishes the whole CSFD strategy.

The termination criterion in classic evolutionary algorithms is hard to define and very often problem dependent, as we do not have any global information about the fitness landscape and therefore we can only compare one solution to another previously found. For a *local termination criterion* we may use some of the standard well-known stopping criteria for evolutionary algorithms like:

- *Expected first hitting time (FHT)* (see e.g. [2]) which tries to set meaningful upper bounds for the number of iterations required to reach the sufficiently small neighborhood of solution.
- *Efficiency measures* (see e.g. [6]), e.g., *Running mean*, the difference between the current best objective value found and the average of the best objective values of the last t generations is equal or less than a given threshold ε.

In terms of the *global stopping criterion* we focus on the distribution of individuals in the admissible domain $\mathcal{D}$. We follow the idea introduced by Telega [15] and proved by Schaefer and Adamska [17] to be successful for multimodal problems with robust solutions.
The global stopping criterion for CSFD algorithm is based on the clustering analysis and defined as follows:

The CSFD algorithm is being terminated if the clustering process applied to the population of individuals returned by the genetic algorithm returns no clusters or the quality measure of the fitness deterioration performed on found clusters is too low.

The clustering, which is performed in every iteration of the CSFD algorithm, gives us some clues about the global characteristics of the fitness landscape, i.e., when the clustering algorithm performed on the final population finds nothing it is very likely that in previous iterations we have deteriorated the fitness landscape in places where the most desirable solutions reside and there is no use in continuing the searching process. This is considered to be true because we are looking for robust solutions which are resistant to noise and lie in basins of attraction which are significantly wide and deep. The population of GA is likely to converge to such solutions, so having found no clusters of individuals after performing the sufficient number of GA epochs shows that the population would not converge to any robust solution.
Such kind of condition may be precisely formulated in terms of the convergence of sampling measures and partially verified in case if the genetic engine is SGA with the focusing heuristic (see [3], [15], [18]).

3. Clustering

Clustering algorithms divide a dataset into several disjoint subsets. All elements in such a subset share common features like, for example, spatial proximity. Clustering is used as a stand-alone tool to get insight into the distribution of a data set or as a preprocessing step for other algorithms operating on the detected clusters. The former is used to determine stop criteria as described in Subsection 2.4 and the latter is used in our fitness deterioration algorithm to improve its accuracy.
A cluster extension (see Definition 2) may be seen as an approximation of the basin of attraction, moreover the distribution of individuals which were flooded to the basins provides additional useful information about its shape. The clustering algorithm may be used to detect the set of individuals which belongs to the same basin of attraction. The CSFD provides the information about detected sets (basins of attraction) in the form of a proper representation of cluster extensions which are just a convenient way to describe basins of attraction (e.g, the center point, the radius of the set, covariance matrix, etc.).
Before we perform the clustering analysis of the multiset of individuals returned by the GA used in the CSFD definition, we have to explicitly map the population P being the multiset of individuals from the genetic universum U to the admissible set $\mathcal{D}$ being the subset of the vector metric space $V \supset \mathcal{D}$. We utilize the injective mapping

$$ph : U \to \mathcal{D} \tag{3}$$

being the encoding or the inverse encoding (it depends upon the convention). In case in which $U = \mathcal{D}_r$, ph becomes the identity on $\mathcal{D}_r$.
Moreover we assume that data to be clustered is the population P transformed to the multiset of elements from $\mathcal{D}_r \subset \mathcal{D}$ called also P for the sake of simplicity.

3.1. Algorithm OPTICS

We have chosen density-base clustering algorithm called *Ordering Points to Identify the Clustering Structure* (OPTICS) (see [19]). Density-base clustering generally needs the data being the subset or the multiset of objects (points) which belong to a finite dimensional vector metric space V. Clusters are regarded as subsets in which the objects are dense and which are separated by regions of low object density.
In particular each object q of the cluster C has to be surrounded by the neighborhood $N_\varepsilon(q) \subset V$ of a given radius ε that contains at least $minPts$ other objects. The formal definition for this notion is as follows:

Definition 3: An object $p \in P$ is *directly density-reachable* from an object $q \in P$ with respect to the parameters $\varepsilon \in \mathbb{R}_+$, $minPts \in \mathbb{N}$ in a set or multiset of data P if:

- $p \in N_\varepsilon(q)$,
- $Card(N_\varepsilon(q) \cap P) \geq minPts$.

The condition $Card(N_\varepsilon(q) \cap P) \geq minPts$ is called the *core object condition*. If this condition holds for an object q, then we call q a *core object*. Only from core objects, other objects can be directly density-reachable.

Definition 4: An object p is *density-reachable* from an object q ith respect to the parameters $\varepsilon \in \mathbb{R}_+$, in the set or multiset of objects P if there is a chain of objects $p_1, \ldots, p_n, p_1 = q, p_n = p$ such that $p_i \in P$ and p_{i+1} is directly density-reachable from p_i for $i = 1, \ldots, n-1$ with respect to ε and $minPts$.

The density-reachability relation is not symmetric in general in P. Only core objects can be mutually density-reachable.

Definition 5: An object p is *density-connected* to an object q with respect to ε and $minPts$ in the set or multiset of objects P if there is an object $o \in P$ such that both p and q are density-reachable from o with respect to ε and $minPts$ in P.

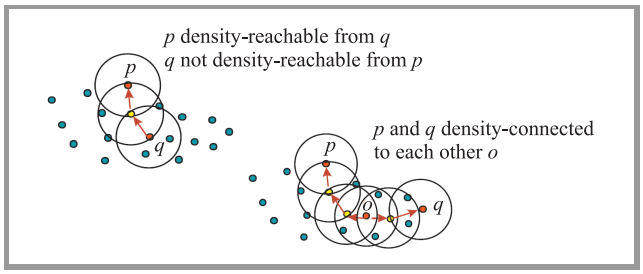


Fig. 1. Density-reachability and connectivity.

Both phenomena are illustrated by Fig. 1. A density-based *cluster* is now defined as a multiset of density-connected objects which is maximal with respect to density-reachability and the *noise* is the set of objects not contained in any cluster.

Definition 6: Let P be a set or multiset of objects. A cluster C with respect to ε and $minPts$ in P is a non-empty subset of P satisfying the following conditions:

- Maximality: $\forall p,q \in P$: if $p \in C$ and q is density-reachable from p wrt. ε and $minPts$, then also $q \in C$.
- Connectivity: $\forall p,q \in C$: p is density-connected to q wrt. ε and $minPts$ in P.

Every object not contained in any cluster is *noise*.

The algorithm DBSCAN (see [20]) discovers the clusters and the noise in a database according to the above definitions. OPTICS works in principle like an extended DBSCAN for an infinite number of distance parameters ε_i which are smaller than a *generating distance* ε. The only difference is that we do not assign cluster memberships. Instead, we store the *order* in which the objects are processed (the main principle is that we always have to select an object which is density-reachable with respect to the lowest ε value to guarantee that clusters with higher density are finished first) and the information which would be used by DBSCAN algorithm to assign cluster memberships. This information consists of only two values for each object:

Definition 7: The *core-distance* of an object p is the smallest distance ε' between p and an object in its ε-neighborhood $N_\varepsilon(p)$ such that p would be a core object with respect to ε' if this neighbor is contained in $N_\varepsilon(p)$. Otherwise, the core-distance is *UNDEFINED*.

Definition 8: The *reachability-distance* of an object p with respect to another object o is the smallest distance such that p is directly density-reachable from o if o is a core object.

The OPTICS algorithm creates the partial order in the population P, additionally storing the core-distance and a suitable reachability-distance for each object. This information is sufficient to extract all density-based clusterings with respect to any distance ε' which is smaller that the generating distance ε. The result of DBSCAN algorithm applied to sample population of 1000 2-dimensional points is shown in Fig. 2.

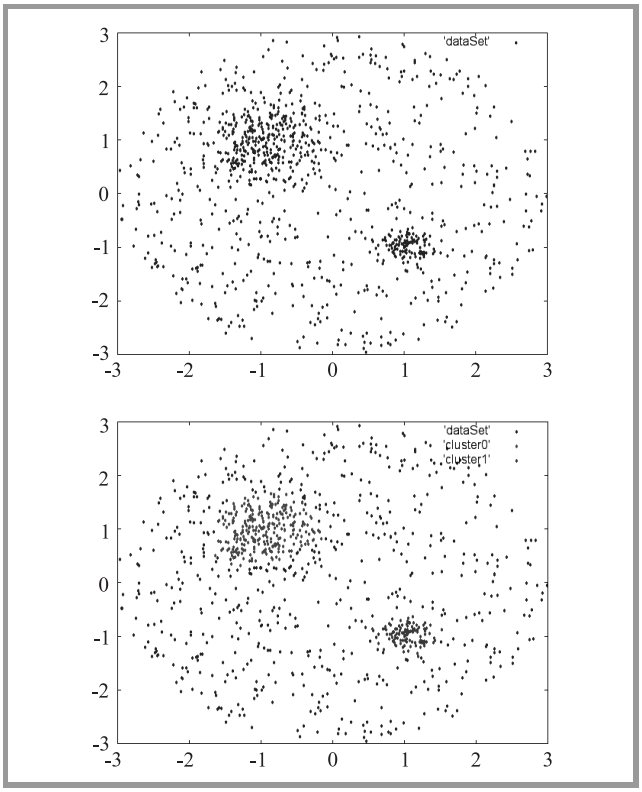


Fig. 2. Visualization of the DBSCAN algorithm applied to OPTICS ordering of simple 2-dimensional data set which consists of 1000 points. OPTICS parameters: $minPts = 20, \varepsilon = 1.2$, the two clusters was found using DBSCAN with parameters: $\varepsilon' = 0.2$

Section 4 describes how OPTICS ordering properties are used to prevent degradation of areas which have not been explored during the course of the CSFD search.

4. Fitness Deterioration

Fitness deterioration is a process of degrading the fitness function in areas occupied by groups of individuals obtained from clustering (see Subsection 1.3 and references inside). We suggest to achieve this goal in the CSFD strategy by creating a linear combination of the current fitness function and *crunching functions* which approximate the fitness in subsets of problem domain occupied by clusters. Let F_k be the fitness function in k-th iteration of CSFD (i.e., in the k-th execution of the main loop of the Algorithm 3) and $C_1,\ldots,C_{M_k}$ be M_k clusters found in k-th iteration of our sequential niching algorithm described in Section 2. For each cluster C_i we create the *crunching function* g_i and then we construct deteriorated fitness F_{k+1} (which will be used in the $(k+1)$-th iteration) as follows:

$$F_{k+1} = F_k + \sum_{i=1}^{M_k} \alpha_i g_i, \tag{4}$$

where the selection of the functional coefficients $\alpha_1,\ldots,\alpha_{M_k}$; $\alpha_i : \mathcal{D} \to \mathbb{R}_+$, $i = 1,\ldots,M_k$ depends on the type of deterioration used and will be described later in Subsections 4.2, 4.3 (see Eqs. (8), (9), (10)).

The *crunching function* $g_i : \mathcal{D} \to \mathbb{R}_+$ is constructed for each newly recognized cluster of individuals $C_i \subset P$.

Based on the assumption that clusters lie inside basins of attraction and that the distribution of individuals inside the cluster is a good approximation of the shape of the basin occupied by the cluster, the deterioration algorithm tries to exploit information provided by the clustering algorithm and based on that information it augments the fitness function in order to minimize the probability of finding already explored basins of attraction in further iterations.

We may ask ourselves why we do not prevent exploration of basins we found in previous iterations simply by remembering the regions occupied by clusters and ignoring individuals which fall in this regions. The answer is probably the most important reason why we have chosen fitness deterioration for this task. We can not prevent individuals to explore neighborhood of solutions found in previous iterations because such approach would cause our metaheuristic to loose *completeness*. If the set of search operations is not complete, there are points in the search space which cannot be reached. Then, we are probably not able to explore the problem space adequately and possibly will not find satisfyingly good solution. That is why it is better to use fitness deterioration as a way to discourage rather then prevent individuals from sinking to the same basins of attraction twice.

Here we would like to emphasize the fact that the deterioration process does not try to accurately interpolate the fitness function in the neighborhood of the solution because it would be very expensive in a high dimensional spaces. Instead it tries to find simple crunching functions which would sufficiently degrade the fitness landscape in the areas occupied by the clusters and then remove the individuals from these regions in the next CSFD steps with the sufficiently high (but even less then 1) probability.

4.1. Fitness Deterioration and Clustering

To increase the accuracy of the fitness deterioration process we want to use the maximum amount of information provided by the clustering algorithm. As we mentioned earlier (see the description of the fitness deterioration in Section 4) we create one crunching function per cluster which, depending on the shape of the cluster, may not be very accurate or may even degrade areas of the fitness landscape which have not been explored yet and potentially contain valuable solutions. Figure 3 shows cases in which crunching functions created for extracted clusters strongly affect regions outside the clusters.

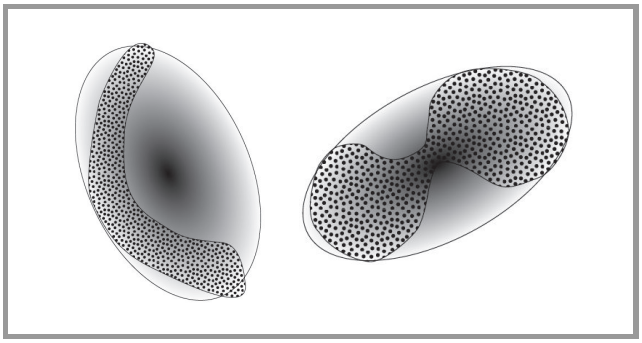


Fig. 3. Example of two clusters returned by the clustering algorithm and corresponding Gaussian crunching functions which degrade areas distant from clusters.

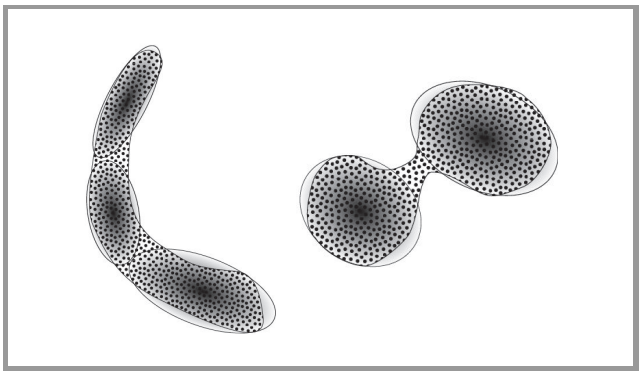


Fig. 4. With OPTICS ordering we may extract cluster of higher densities and minimize the impact of the fitness deterioration on regions outside the clusters (instead of creating one crunching function per cluster like in figure 3 we extract denser clusters from the ordering and create crunching function which degrade only the region occupied by the cluster).

To increase the accuracy of our algorithm we use the following property of the OPTICS *ordering*:

> While creating the ordering, OPTICS constructs density-based clusters with respect to different densities simultaneously. OPTICS ordering actually contains the information about the intrinsic clustering structure of the input data set (up to the generating distance ε) (see [19]).

This might be shown for sample data (see Fig. 5) by using its *reachability plot*. Once we create OPTICS ordering

we may easily extract clusters with higher densities by decreasing ε and choose clusters for which mean-square error (MSE) between the actual fitness function and the crunching function in the areas of clusters is minimal.

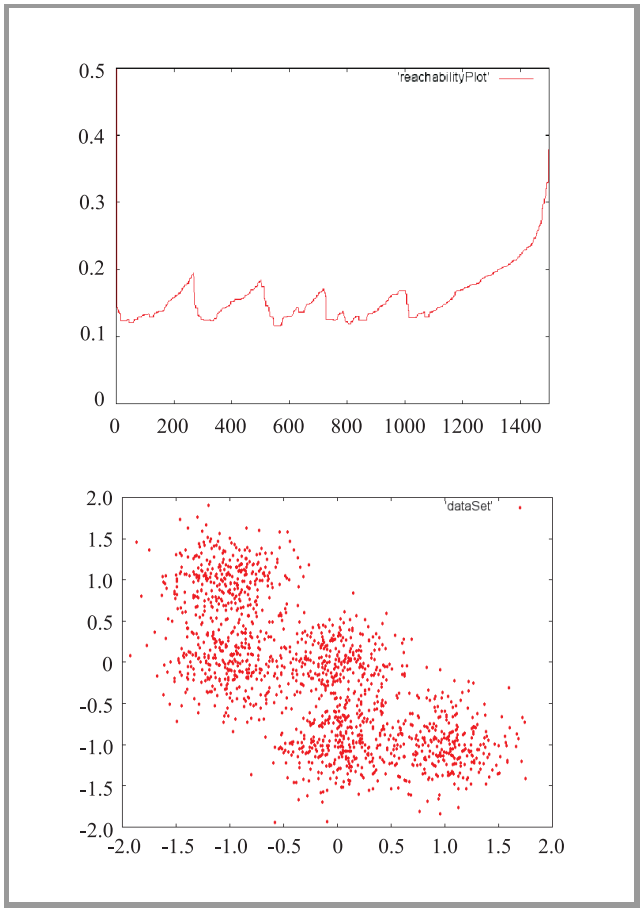


Fig. 5. Sample data set of 1500 points and the corresponding reachability plot of the ordered points (shows the reachability distance of each individual in the data set – horizontal axis corresponds to the individuals in the data set and the vertical axis shows the reachability distance of a given individual) OPITCS ordering parameters: $\varepsilon = 1.0$, $minPts = 30$. Cavities in the plot depict 5 clusters which might be extracted from the data set by the DBSCAN algorithm using proper value of ε' values

The algorithm works as follows (see Algorithm 4): having the OPTICS ordering of the population returned by the EA our method iteratively extracts clusters with higher densities by decreasing the *neighborhood radius* ε (see Fig. 6), and then by constructing crunching function for extracted clusters and checking if the resulting crunching functions is more accurate than the best found in previous iterations (MSE comparison).

Algorithm 4 effectively prevents the deterioration process from destroying the fitness landscape in regions not yet explored by the CSFD algorithm. Figure 4 shows how the ε adjustment can improve the shape of the cluster extension with respect to the initial one (see Fig. 3). To better understand how do we use OPTICS ordering to extract cluster of higher density see Figs. 5 and 6.

Algorithm 4: Improving fitness deterioration accuracy

$\varepsilon' = \varepsilon$
while $\varepsilon' > treshold$ **do**
  $cs \leftarrow extractDBSCANClustering(\varepsilon')$
  $crunchFs \leftarrow createCrunchingFunctions(cs)$
  $mse \leftarrow getMSE(crunchFs, currentFitness, cs)$
  if $mse < minMSE$ **then**
    $saveBestCrunching(mse, crunchFs)$
  end if
  $\varepsilon' \leftarrow \varepsilon' * 0.8$
end while

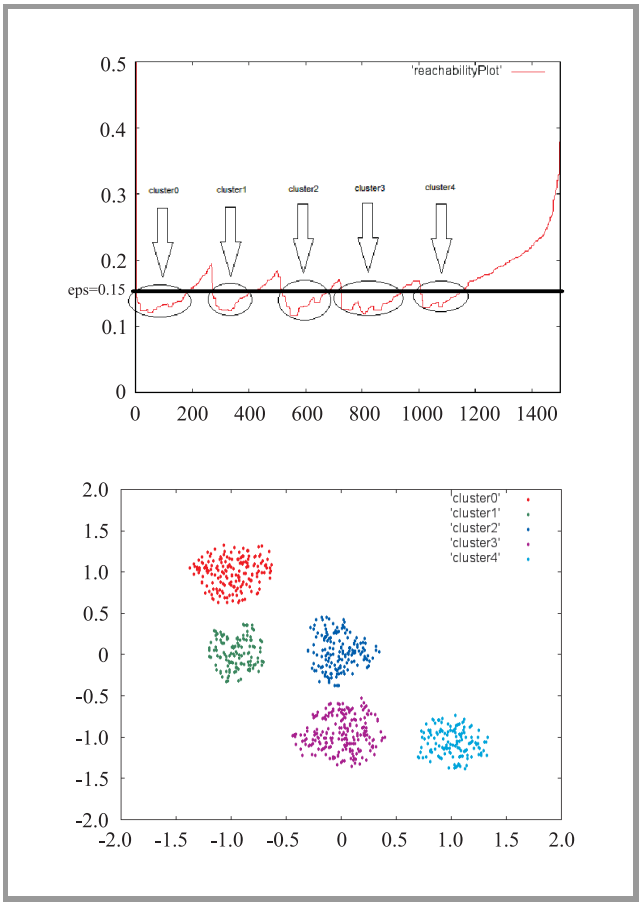


Fig. 6. Extraction of clusters from the data set presented in Figure 5 using DBSCAN algorithm with $\varepsilon' = 0.15$. The value of ε' is marked on the first plot which shows reachability distances and the "cut off" clusters

4.2. Basic Scheme

The basic version of our deterioration algorithm is as follows. For each cluster C generate a multidimensional Gaussian function $g : V \to \mathbb{R}_+$

$$g(x) = F_k(x_{max})\, exp\left(-\frac{1}{2}(x-\overline{m})^T \Sigma^{-1} (x-\overline{m})\right), \qquad (5)$$

where F_k is the fitness function in k-th iteration of the CSFD algorithm, x_{max} is the fittest individual from the cluster, the $\overline{m}$ is the cluster's centroid (mean phenotype of individuals belonging to C) and Σ is unbiased sample covariance

matrix estimated from the cluster population by the Eq. (6) (see e.g. [21])

$$\Sigma = \frac{1}{Card(C)-1}\sum_{x\in C}(x-\overline{m})\otimes(x-\overline{m}), \tag{6}$$

where $\otimes$ stands for the tensor product symbol. Please, notice, that the sum in Eq. (6) is spanned over all individuals belonging to the cluster C, i.e., the phenotype x might be counted more than once, if is repeatedly represented in C.

Fitness function in the $(k+1)$-th iteration is obtained from the Eq. (4) by seting $\alpha_i \equiv 0, i = 1,\ldots,M_k$.

Big advantage of this algorithm are simplicity and speed. However, this version may cause strong deformation of the fitness landscape in areas which are distant from the already found clusters, which is unacceptable. To overcome this issue we developed so called *weighted scheme* described in the next subsection.

4.3. Weighted Scheme

This type of fitness deterioration is more accurate and is likely to produce more stable fitness that basic scheme, cause the latter may produce sharp peaks in the fitness landscape, because of the very aggressive fitness degeneration. Weighted scheme is also more complex cause it generates more computationally intensive fitness functions.

Initial steps are the same as in basic scheme, we create multidimensional Gaussian function for each cluster. What is different is how we compute the new (deteriorated) fitness is the following way

$$F_{k+1}(x) = F_k(x) + \sum_{i=1}^{M_k}\alpha_i(x)g_i(x), \tag{7}$$

where the α-coefficients are given by the following equations

$$\alpha_1(x) + \ldots + \alpha_{M_k}(x) = 1, \tag{8}$$

$$\alpha_i(x) = \xi(x)\left(\frac{1}{r_i(x)}\right), \tag{9}$$

$$\frac{1}{\xi(x)} = \frac{1}{r_1(x)} + \ldots + \frac{1}{r_{M_k}(x)}, \tag{10}$$

where $r_i(x) = \|x-\overline{m}_i\|$ is the distance between x and the C_i centroid $\overline{m}_i$ (see Fig. 7). So in order to compute fitness for a given individual x (more correctly for $ph(x)$) we have firstly compute distances $r_i(x)$. Then we compute $\xi(x)$ from the Eq. (10), next the α-coefficients from the system Eqs. (8) and (9), and finally the fitness value from Eq. (7). α-coefficients are computed separately for each new individual and this is why this method is more costly than the previous one.

From the Eqs. (7)–(10) it is clear that regions of domain which are distant from clusters found in previous iterations of the algorithm are very little affected by the crunching functions which is a big advantage over the basic scheme. However, experiments shows that the basic scheme yields very good results and is preferable over the weighted scheme due to the lower computational cost of the former.

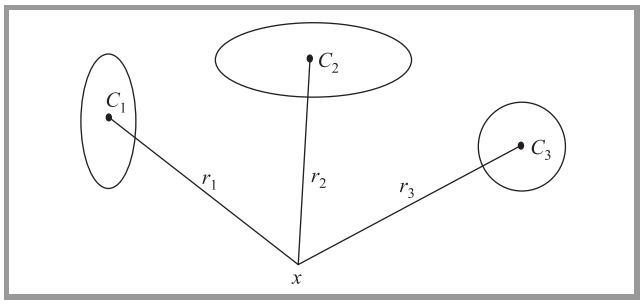


Fig. 7. The coefficient α_i is inversely proportional to the distance from the center of cluster C_i to the x. α_i may be seen as the impact C_i has on x.

4.4. Crunching Function Adjustment

Because of the fast convergence of populations generated by the GA algorithm to the local solutions and the features of the Algorithm 4 used to increase accuracy of the deterioration process, clusters sometimes become very dense in areas close to the local minimizers. Gaussians created for such clusters does not approximate a basin of attraction well, speaking informally: Gaussian functions created for such clusters consist of high and thin peaks which deteriorate only the area inside the cluster, only the narrow basin of attraction in which the cluster resides. To overcome this issue we developed so called *Crunching Function Adjustment* (CFA) algorithm described below.

We use sample covariance matrix as an estimator (see [21]), which is extremely sensitive to outliers. However we may take this property as our advantage and incorporate it CFA algorithm. Having given a cluster of points the CFA algorithms works as follows:

- We estimate the covariance matrix Σ and then compute its N eigenvectors ($N = dim(V)$ is the dimension of the problem space). They are ortogonal one to each other and define the orientation of the Gaussian "bell".
- For each eigenvector v_i we generate two points $\underline{p_i}, \overline{p_i}$ called *leading marks*

$$\overline{p_i} = \overline{m} + \sqrt{\lambda_i}\,v_i.$$
$$\underline{p_i} = \overline{m} - \sqrt{\lambda_i}\,v_i. \tag{11}$$

  where $\overline{m} \in V$ is the cluster's centroid and λ_i is an eigenvalue of the eigenvector v_i.
- Then we add these $2N$ generated leading marks to the initial multiset which constitute a cluster, and compute new covariance matrix.
- Because the sample covariance matrix is very sensitive to leading marks, the resulting covariance matrix produces a Gaussian function whose "bell hypersurface" is more stretched in directions of eigenvectors.

Please notice, that the improvement introduced above is purely heuristic, having no precise mathematical motivation. It was designed only for deterioration performed by Gauss functions and positively verified for 2D benchmarks (see Subsection 5.2) so its usefulness in other cases in unknown.

Remark 1: The whole consideration leading to define CSFD was performed for GOPs of finding local/global minimizers. It can be easily reformulated for GOPs of finding local/global maximizers for which the equivalent minimization problem can be established. In this case the landscape deterioration consists in "leveling hills" instead of "filling valleys". The particular class of such maximization GOPs is associated with continuous objectives (fitnesses) F defined on compacts in $\mathbb{R}^N$. In such cases we can set the new fitness as $-F$ plus the maximum value of F over the search domain in order to obtain the equivalent minimization problem.

5. Experiments and Comparison with other Algorithms

The aim of our experiments is to show the efficiency of the *Cluster Supported Fitness Deterioration* CSFD in finding basins of attraction of local solutions. We also want to present a simple comparison with other strategies, especially the ESSS-DOF [9] described in Subsection 1.3.

5.1. Genetic Engine

The CSFD strategy uses the *Simple Evolutionary Algorithm* (SEA). In contrast to the *Simple Genetic Algorithm* (SGA), SEA uses a real values as a parameters of the chromosome in populations without performing coding and encoding process before calculates the fitness values of individuals (see e.g. [3]). Namely, SEA is more straightforward, faster and more efficient than SGA.
We use the standard genetic operators for real-valued representation:

- Crossover: $Y = x_1 + N(mean, \sigma_c)(x_2 - x_1)$, where x_1, x_2 are individuals, $N(mean, \sigma_c)$ stands for the multivariate normal distribution, $mean = \frac{x_1+x_2}{2}$ is a $[x_1, x_2]$ centroid, σ_c is the parameter used to control exploration and exploitation of the SEA.
- Mutation: $y = x + N(0, \sigma_m)$, where x is the individual to be mutated, σ_m the configurable mutation parameter.

We want the genetic engine to be cheap and converge very quickly to local solutions, so we may efficiently deteriorate found basins of attraction in subsequent iterations. To improve the convergence of the population maintained by SEA we use proportional selection and we increase the exploitation capabilities of the SEA using proper parameters, usually: $\sigma_c \in [0.1, 0.3]$, $\sigma_m \in [0.4, 1.0]$, depending on the problem.

5.2. Test Functions

In order to visualize the deterioration process we use three 2D test functions:

- Rastrigin:

$$F(x) = 20 + \sum_{i=1}^{2} (x_i^2 - 10\cos(2\pi x_i)) \tag{12}$$

  for $x_1, x_2 \in [-4, 4]$.

- Langermann:

$$F(x) = \sum_{j=1}^{5} c_j \exp(-\tfrac{1}{\pi}((x_1 - a_j)^2 + (x_2 - b_j)^2)) \cos(\pi((x_1 - a_j)^2 + (x_2 - b_j)^2)) + 5,$$
$$a = (3,5,2,1,7), b = (5,2,1,4,9), c = (1,2,5,2,3) \tag{13}$$

  for $x_1 \in [0, 4], x_2 \in [-1, 3]$.

- Griewangk:

$$F(x) = \tfrac{1}{4000} \sum_{i=1}^{2} x_i^2 - \Pi_{i=1}^{2} \cos\left(\tfrac{x_i}{\sqrt{i}}\right) + 1 \tag{14}$$

  for $x_1, x_2 \in [-4, 4]$.

W decide to handle maximization GOPs associated with benchmarks Eqs. (12), (13), (14) because their results can be more expressively presented then the equivalent minimization ones. The CSFD instance dedicated to maximization problems were utilized (see Remark 1).

5.3. CSFD Initial Parameters

In each of the tests we use the same set of initial values for algorithm's parameters. The *Cluster Supported Fitness Deterioration* needs the following parameters to be configured:

- *popSize* – population size of the genetic engine used in CSFD; it is advisable to use small population to minimize the fitness computation costs; usually: $40 \leq popSize \leq 100$.
- ε and *minPts* – OPTICS generating distance and minimum number of points in ε-neighborhood (see Subsection 3.1); the values for the ε should be 'large enough' to allow the the creation of proper *ordering*, usually $\varepsilon \in [0.5, 1.0]$ depending on the selection pressure. *minPts* should be chosen as follows:

$$minPts = \begin{cases} \frac{popSize}{4} & \text{if} \quad 10 \leq \frac{popSize}{4} \leq 20 \\ 10 & \text{if} \quad \frac{popSize}{4} < 10 \\ 20 & \text{if} \quad 20 < \frac{popSize}{4}. \end{cases} \tag{15}$$

- σ_c and σ_m – standard deviations used for the *crossover* and *mutation* respectively (see Subsection 5.1).
- *mutationProb* – mutation probability (crossover is always performed).

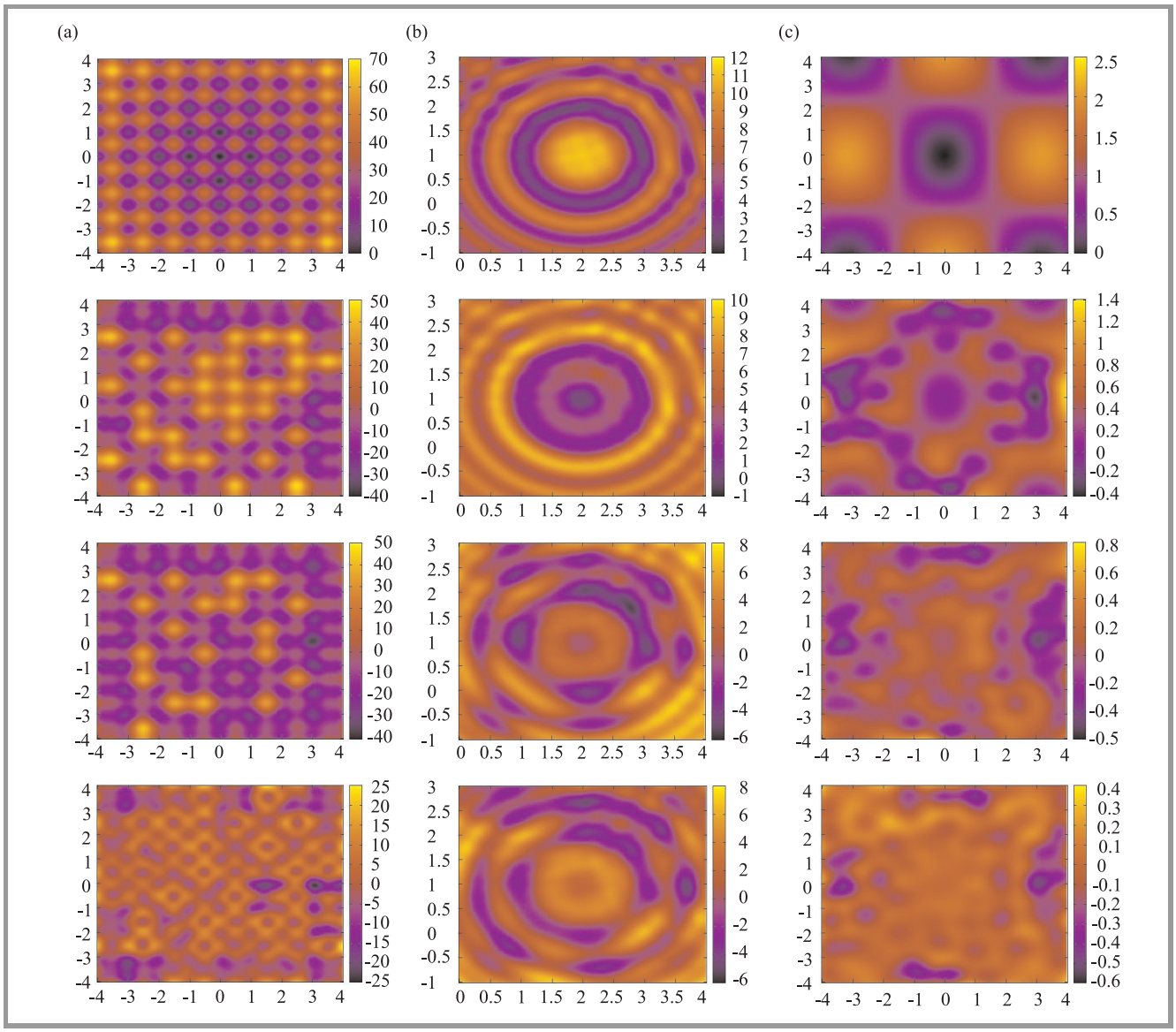


Fig. 8. Original (at the top) and detoriated fitness landscape: (a) of the Rastrigin function seen in the 33th, 45th and 64th interation of the CSFD strategy respectively. Parameters: $\varepsilon = 0.7$, $minPts = 12$, $popSize = 50$, $\sigma_c = 0.1$, $\sigma_m = 0.5$; (b) of the Langermann function seen in the 1th, 8th and 16th interation of the CSFD strategy respectively. Parameters: $\varepsilon = 0.7$, $minPts = 12$, $popSize = 80$, $\sigma_c = 0.1$, $\sigma_m = 0.5$; (c) of the Griewangk function seen in the 21th, 51th and 69th iteration of the CSFD strategy respectively. Parameters: $\varepsilon = 0.7$, $minPts = 12$, $popSize = 50$, $\sigma_c = 0.1$, $\sigma_m = 0.5$.

5.4. Efficiency Measures and CSFD Results

We will use two simple measures in order to evaluate the deterioration quality. The first one was the number of recognized basins of attraction *NBA*. The basin of attraction of the local maximizer is considered as recognized by CSFD if the cluster of individuals was established and the cluster extension is wholly included in the basin.

The second measure, called the degree of deterioration *DoD*, is defined by the following formula:

$$DoD = \frac{F_0^{\max} - F_M^{\max}}{F_0^{\max} - F_0^{\min}}, \tag{16}$$

where M is the number of iterations performed by the CSFD strategy, $F_0^{\max}$ denotes the maximum value of the fitness at the beginning of the algorithm (0-iteration), similarly $F_0^{\min}$ is the minimum value of the fitness function at the beginning of the algorithm and $F_M^{\max}$ is the maximum value of the fitness in the last iteration of the algorithm. *DoD* might be also expressed in percentage.

Table 1
Results of experiments (CSFD algorithm)

Function	*NBA*	*DoD*
Rastrigin	64/64	0.64
Langermann	16/18	0.59
Griewangk	4/4	0.84

CSFD was run several times for each of the objective Eqs. (12), (13), (14). The most typical behavior of this

strategy in case of each objective are depicted in Fig. 8. Moreover Table 1 gathers the metrics values for these computations.
The best performance was obtained for the Griewangk benchmark for which all local maxima were encountered and for which the maximum degree of fitness leveling 84% was obtained. All local maxima were also recognized in the case of the Rastrigin function, but the degree of deterioration was only 64%. The Langermann benchmark became most difficult for CSFD, since the degree of deterioration was only 59% and two basins of attraction (out of eighteen ones) remained unknown.

5.5. Comparison with other Algorithms

Experiments were also performed to investigate the performance of ESSS-DOF strategy (see Algorithm 1). This strategy was implemented according to the description contained in [12], [13]. Similarly as in the case of CSFD, ESSS-DOF was run several times for each objective Eqs. (12), (13), (14). Its most typical behavior is illustrated in Fig. 9. These snapshots show the iterations in which fitness was degenerated (i.e., when *trap_test* procedure indicated that the population converged to the local solution).
Table 2 shows the comparison of metric values obtained by both ESSS-DOF and CSFD. Here we will say that ESSS-DOF recognizes the basin of attraction if the deterioration component Eq. (1) with the center in its area was introduced.

Table 2
Comparison of ESSS-DOF with CSFD algorithms

	ESSS-DOF		CSFD	
Function	*NBA*	*DoD*	*NBA*	*DoD*
Rastrigin	8/64	0.14	64/64	0.64
Langermann	9/18	0.36	16/18	0.59
Griewangk	4/4	0.60	4/4	0.84

The current ESSS-DOF implementation never found all basins of attraction and the *DoD* measure obtained was significantly worst.
We may observe that the ESSS-DOF is efficient only for simple multi-modal functions with small amount of solutions in the problem domain (e.g., Griewangk function, see Fig. 9(c)), but it performs badly for more complex problems (e.g., Rastrigin function, see Fig. 9(a)). This might be attributed to the fact that the fitness deterioration in ESSS-DOF strategy is inaccurate due to the lack of constraints which might prevent degradation of a large area of the domain. For highly multimodal problems the sample might be spread across many neighboring basins of attraction, which is deceptive for the *trap_test* procedure and causes the Gaussian function to degrade to large area, including unexplored regions of the search space. Strong degradation of the large area of domain prevents the search process from finding new promising individuals, which follows from the stopping criterion of ESSS-DOF (see Subsection 1.3) resulting in the premature termination of the algorithm.
On the other hand the CSFD strategy which includes accurate clustering strategy may properly handle the situation in which individuals from the population reside in many basins of attraction. The algorithm is able to locate clusters separately in each basin, rejecting the "noisy" individuals and perform deterioration only in the areas of cluster extensions, which allow for successive search in unexplored regions.
A further advantage of CSFD algorithm compared to ESSS-DOF and other sequential niching methods is that the CSFD is resistant to small changes of the input parameters (specified in Subsection 5.3). Roughly speaking, there are a wide range of initial parameters for which we may expect the algorithm to be effective. Choosing reasonable values for σ_c and σ_m, and OPTICS parameters as specified in Subsection 5.3, would very likely yield a good results.

6. Conclusions

- Sequential niching with fitness deterioration is one of the most promising stochastic strategies for analyzing multi-modal global optimization problems in the continuous domains embedded in vector metric spaces.
- Existing instances of the strategy mentioned above exhibits several disadvantages: huge memory complexity of memorizing deteriorated regions (e.g., CGS with raster clustering); unsatisfactory accuracy of the fitness approximation that lead to the incorrect deterioration and finally may lead to removing individuals from unchecked areas or the multiple check of non-promising areas already browsed; dependency on the evolutionary technique used.
- The discussion presented in Subsections 1.3, 1.4, 1.5 clearly shows the way of necessary improvements. The proposed deterioration strategy CSFD combines the low memory complexity of the exponential fitness improvement with the accuracy of clustering based techniques.
- The CSFD performs very well for 2D complex multimodal functions like the ones used for testing (see Subsection 5.2) being also well suited to detect the basins of attraction of the local and global extrema as specified in Subsection 5.5. Exhaustive testing for higher dimensional problems will be the subject of future research.
- Performed experiments show that we can expect satisfactory results when the GA utilizes genetic operators based on the normal distribution. Fitness proportionate selection causes satisfactory convergence to local solutions and the normal distribution based reproduction operators tends to produce populations with useful information about fitness landscape.

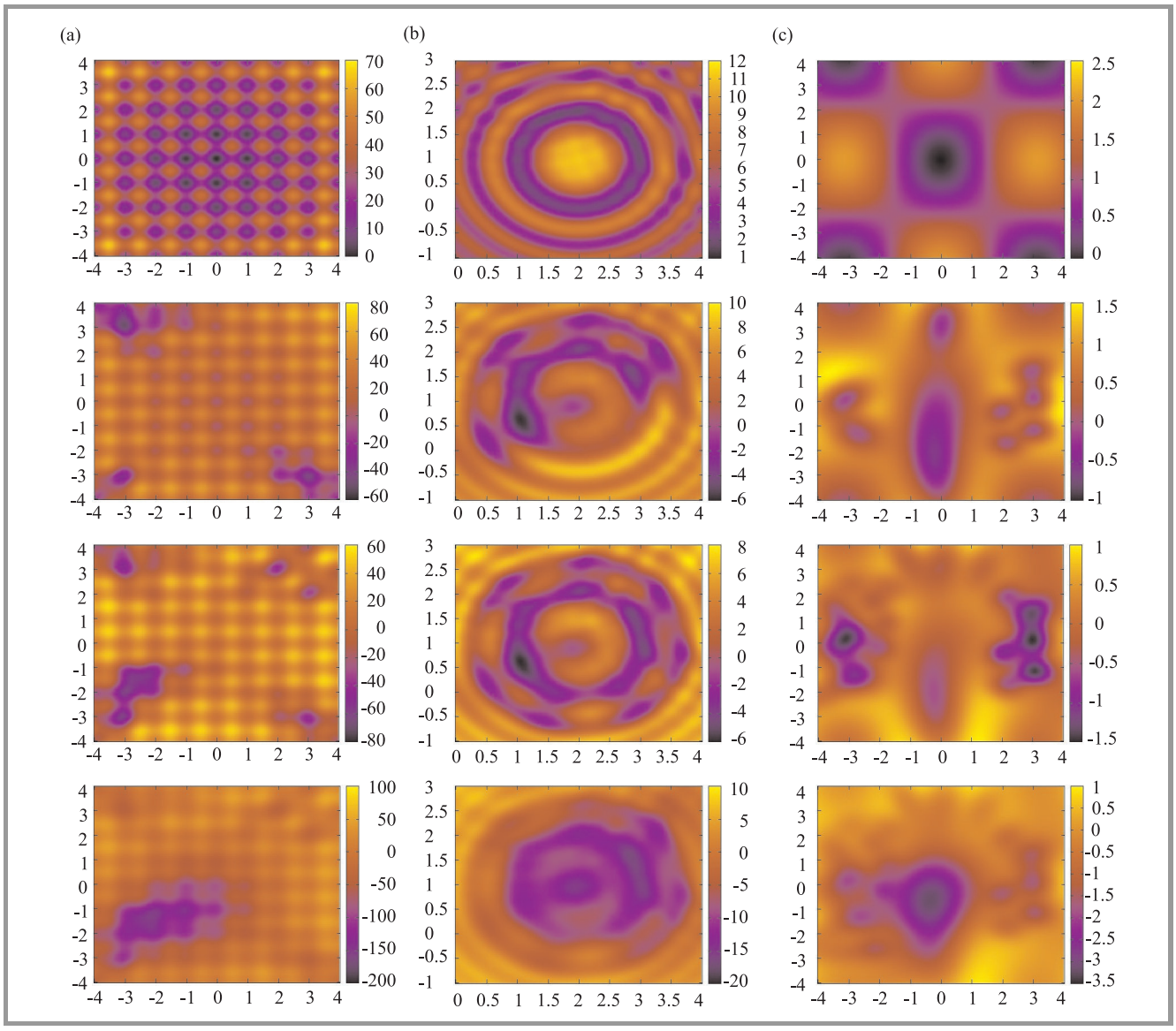


Fig. 9. ESSS-DoF original (at the top) and detoriated fitness landscape: (a) of the Rastrigin function; (b) of the Langerman function; (c) of the Griewangk function. Parameters: $p = 5\%$, $n_{trap} = 30$, $\sigma_c = 0.1$, $\sigma_m = 0.4$.

- The Hierarchical Genetic Strategy (HGS) (see [10], [16], [22]) would be very efficient from the standpoint of the deterioration process. This strategy performs an efficient concurrent search in the optimization landscape by many small populations. Creation of these populations is governed by dependent genetic processes with low complexity. Moreover, HGS is likely to find many solutions in a single run of the algorithm and that the hierarchy of populations generated by the algorithm are rapidly convergent.
- High accuracy of deterioration offered by the CSFD results from the positive synergy of two mechanisms: clusters obtained from the modified OPTICS (see Algorithm 4) and improved by leading marks approximate well basins of attraction of local/global extrema; the form of the weighted deteriorartion function (7) and form of weights (see formulas (8), (9), (10)) maximizes the effect of deterioration over the area of cluster extensions i.e. the area of basins of extrema already recognizes and prevent degradation of unexplored regions.
- Algorithm 4 which increases fitness deterioration accuracy performs well also in cases when the clusters are not convex e.g., for Langermann function (see Fig. 8(b)).
- Sequential niching obtained by CSFD preserves the asymptotic guarantee of success. The probability of sampling is significantly decreased over the cluster extensions but still greater then zero, so this regions are not excluded from future sampling, even in case of inaccurate deterioration. It seems to be the advantage over the sequential niching based on "tabu" techniques.

- A simple comparative 2D tests shows superiority of CSFD over the ESSS-DoF strategy. It is caused mainly by using the accurate clustering strategy OPTICS that allow for precise location of the central parts of basins of attraction of local solutions and then perform fitness deterioration only in these areas. This feature prevents the premature termination allowing the further search in the unexplored regions.

7. Acknowledgement

The work presented in this paper has been partially supported by Polish Ministry of Science and Higher Education grant no. NN 519 447739.

References

[1] E. Zeidler, *Nonlinear Functional Analysis and its Application*. Springer, 1985.

[2] M. P. Pardalos and H. E. Romeijn, *Handbook Global Optimization*, vol. 2, Kluwer, 2002.

[3] R. Schaefer, *Foundation of Glabal Genetic Optimization*. Springer, 2007.

[4] A. H. G. Rinnoy Kan and G. T. Timmer 1987, "Stochastic global optimization methods", *Mathematical Programming*, vol. 39, pp. 27–56.

[5] *Toward Global Optimization*, L. C. W. Dixon and G. P. Szegö, Eds. North Holland, 1975.

[6] D. Goldberg, *Genetic Algorithms and their Applications*. Addison-Wesley, 1989.

[7] J. Arabas, *Wykłady z algorytmów ewolucyjnych*. WNT, Warsaw, Poland (in Polish).

[8] S. W. Mahfoud, "Niching Methods", in *Handbook of Evolutionary Computations*, T. Bäck, D. B. Fogel, and Z. Michalewicz, Eds. Oxford University Press., 1997

[9] A. Obuchowicz, "The evolutionary search with soft selection and deterioration of the objective function", in *Proc. 6th Int. Conf. Intel. Inform. Sys. IIS'97*, Zakopane, Poland, 1997, pp. 288–295.

[10] R. Schaefer, K. Adamska, and H. Telega, "Genetic clustering in continuous landscape exploration", *Engin. Applicat. Artif. Intell. EAAI*, vol. 17, pp. 407–416, 2004.

[11] D. Beasley, D. R. Bull, and R. R. Martin, "A sequential niche for multimodal function optimization", *Evol. Comput.*, vol. 1, no. 2, pp. 101–125, 1993.

[12] A. Obuchowicz, "Adoption of the time-varying landscape using an evolutionary search with soft selection algorithm", in *Proc. 3rd Krajowa Konferencja Algorytmy Ewolucyjne i Optymalizacja Globalna KAGiOG Conf.* Potok Złoty, Poland, 1997, pp. 245–251.

[13] A. Obuchowicz and K. Patan, "About some evolutionary algorithm cases", in *Proc. 2nd Krajowa Konferencja Algorytmy Ewolucyjne i Optymalizacja Globalna KAEiOG Conf.*, Rytro, Poland, 1997, pp. 193–200.

[14] A. Obuchowicz and J. Korbicz, "Evolutionary search with soft selection algorithms in parameter optimization, in *Proc. Parallel Proces. Appl. Mathemat. Conf. PPAM'99*, Kazimierz Dolny, Poland, 1999, pp. 578–586

[15] H. Telega, "Równoległe algorytmy rozwiązywania wybranych zagadnień odwrotnych ", Ph.D. thesis, AGH University of Science and Technology, Kraków, Poland, 1999 (in Polish).

[16] R. Schaefer and J. Kołodziej, "Genetic search reinforced by the population hierarchy", in *Foundations of Genetic Algorithms 7*, K. A. De Jong, R. Poli, and J. E. Rowe, Eds. Morgan Kaufman, 2003, pp. 383–399.

[17] R. Schaefer and K. Adamska, "Well-tuned genetic algorithm and its advantage in detecting basins of attraction", in *Proc. 7th Krajowa Konferencja Algorytmy Ewolucyjne i Optymalizacja Globalna KAEiOG Conf.*, Kazimierz, Poland, 2004, pp. 149–154.

[18] R. Schaefer and Z. J. Jabłoński, "On the convergence of sampling measures in global genetic search", *LNCS*, vol. 2328, pp. 593–600, 2001.

[19] M. Ankerst, M. M. Breunig, H. P. Kriegel, and J. Sander, "OPTICS: Ordering points to identify the clustering structure", in *Proc. ACM SIGMOD Int. Conf. Manag. Data*, Philadephia, Pennsylvania, USA, 1999, vol. 28/2, pp. 49–60.

[20] M. Ester, H. P. Kriegel, J. Sander, and Xiaowei Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise", *Computer*, no. 6, pp. 226–231, 1996.

[21] J. P. Hoffbeck and D. A. Landgrebe, "Covariance matrix estimation and classification with limited training data", *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, no. 7, 1996.

[22] B. Wierzba, A. Semczuk, J. Kołodziej, and R. Schaefer, "Hierarchical genetic strategy with real number encoding", in *Proc. 6th Krajowa Konferencja Algorytmy Ewolucyjne i Optymalizacja Globalna KAEiOG Conf.*, Łagów Lubuski, Poland, 2003, pp. 231–237.

Adrian Wolny graduated in Computer Science from AGH University of Science and Technology, Kraków, Poland. His studies specialty was distributed systems and computer networks. He is a software developer with over 2 years of experience. His research interests include: distributed systems, evolutionary algorithms, clustering and sequence labeling.
E-mail: wolny101@gmail.com
Department of Computer Science
AGH University of Science and Technology
Mickiewicza Av. 30
30-059 Kraków, Poland

Robert Schaefer Prof. of techn. sciences, D.Sc., Ph.D., MEng. He is a Full Professor in the Department of Computer Science, Faculty of Electrical Engineering, Automatics, Computer Science and Electronics, AGH University of Science and Technology, Kraków, Poland. He is the author and co-author of more than 160 books, papers and conference contributions. He's research is concentrated in the following areas: genetic algorithms, multi-agent systems, solving direct and inverse problems for PDEs by the hp-adaptive Finite Element Method. Moreover he's former research activities are associated with: modeling of the blood flow in arteries and modeling of nonlinear flow in porous media.
E-mail: schaefer@agh.edu.pl
Department of Computer Science
AGH University of Science and Technology
Mickiewicza Av. 30
30-059 Kraków, Poland

Paper

Topological Synthesis of Tree Shaped Structures Based on a Building Blocks Hypothesis

Przemysław Miazga

Institute of Radio Electronics, Warsaw University of Technology, Warsaw, Poland

Abstract—In this paper a new approach to evolutionary controlled creation of electronic circuit connection topology is proposed. Microwave circuits consisting of a tree like connection of ideal transmission lines are considered. Assuming that a reasonable number of transmission lines in a tree network ranges from 10 to 100, the number of connection combinations is immense. From the engineering practice comes the hypothesis that any device can be decomposed into some functional building blocks consisting of one to dozen transmission lines. The variety of linking combinations in a tree with a limited depth is confined to hundreds or thousands of shapes. Therefore we can decrease the dimensionality of research space, applying evolution to building blocks only. Evolutionary algorithm (EA) which processes simultaneously the population of λ functional blocks and population of μ circuits is proposed. A μ, λ selection scheme with tournament together with specific encoding of solutions, and custom operators is implemented. The μ, λ, α EA was tested on an example of the design of a microwave transistor matching circuit.

Keywords—evolutionary algorithm, matching networks, minimax algorithm.

1. Impedance Matching Problem

The key issue in microwave circuit design is the matching problem. Microwave power, generated by sources (e.g., AC voltage generator) should be delivered to loads (e.g., antenna, microwave oven) with minimal losses, in the specified frequency band. Both the source and the load are characterized by their *admittances* $Y_g(f)$ and $Y_l(f)$, respectively. Admittances, in general, are complex numbers, with real and imaginary parts called *conductance* and *susceptance*, respectively. In the ideal case of perfect matching, without losses, it holds:

$$Y_l(f) = Y_g^*(f). \tag{1}$$

The matching network, shown in Fig. 1, is used to transform load admittance to fulfill condition (1).

Frequently, for passive circuits, source *susceptance* is small and can be neglected, while source conductance is constant. The procedures for matching some real to real admittance are well known [1]. The general problem of complex admittance matching is very hard to solve. Practically, only numerical optimization can be applied in that case. In general, load *susceptance* should be compensated, and load *conductance* should be transformed to source *conductance*.

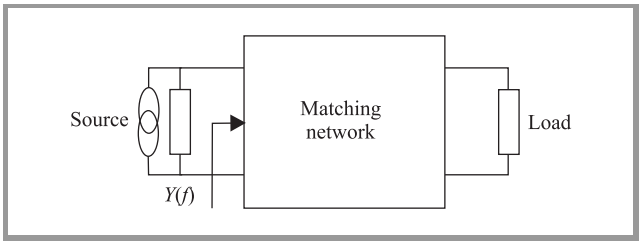


Fig. 1. General representation of a two-port matching network.

The quality of a device is usually defined by means of power loss through an *input reflection coefficient*. The *reflection coefficient* equals the square root of the ratio of power delivered to the load to power generated by the source. Assuming that transformed input admittance is equal to $Y(f)$, *reflection coefficient* $\Gamma(f)$ is given by:

$$\Gamma(f) = \frac{Y_g^*(f) - Y(f)}{Y(f) + Y_g^*(f)}. \tag{2}$$

In order to design an optimal circuit, the following minimax problem should be solved:

minimize (by changing parameters of elementary blocks)
$\{$**maximum** of $|\Gamma(f)|$ in the given frequency band$\}$. (3)

High frequency (from megahertz to terahertz) impedance matching networks (which are mostly used in amplifiers and splitters) are usually designed as a cascade connection of elementary blocks (shown in Fig. 2), mostly uniform transmission lines [1]. The number of elementary blocks of the cascade is set initially. Only transmission line parameters undergo optimization. The cascade transmission device is usually easy to design, although expensive in manufacturing, for it needs thin and long substrate and case.

More compact, lightweight packaging can be used when the circuit is synthetized as a hybrid structure, which example is shown in Fig. 3.

Intuitively thinking, a wider frequency band and compactness can be achieved with a branching-out, tree-like connection of transmission lines. An example of the tree matching network is shown in Fig. 4.

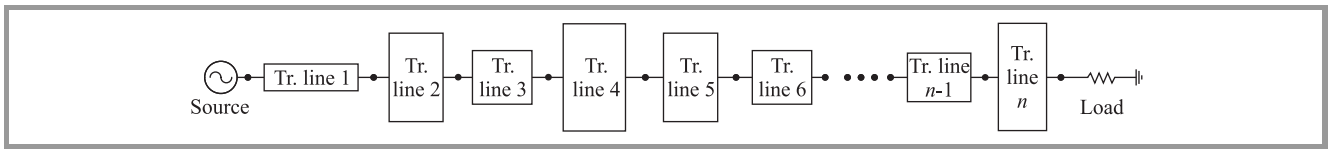


Fig. 2. Cascade two-port matching network.

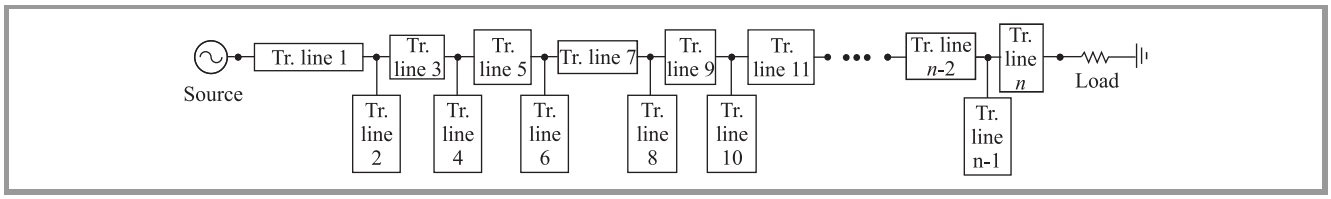


Fig. 3. Hybrid two-port matching network.

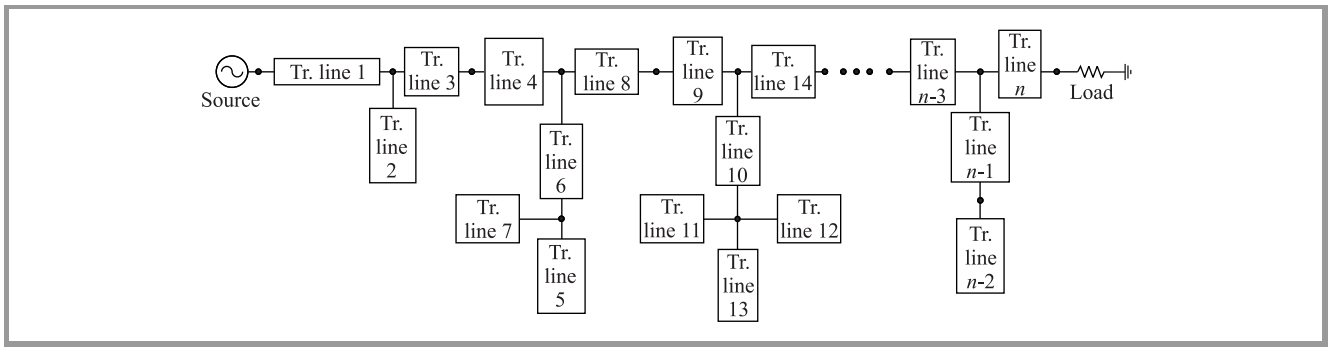


Fig. 4. Tree two-port matching network.

2. Topological Synthesis of a Tree-Shaped Structure

Evolutionary algorithm (EA) can be used as a very efficient tool for topological synthesis of tree devices. It produces the shape of a tree, which parameters (impedances and lengths of a uniform transmission line) are further optimized by a mini-max algorithm. In [2], EA with problem specific evolutionary operators, operating on a tree, was presented. The algorithm was based on the $\mu + \lambda$ selection scheme together with specific encoding of solutions. Each allele was coded as a node of the tree with links to connected blocks (nodes). Evolutionary operators: perturbation, contraction, expansion were designed to operate with a single allele (transmission line or line termination) uniformly within the whole tree. The recombination was defined in a similar way to a one-point crossover. The algorithm was tested with success. Several new topologies for networks matching real source to real load were generated.

Wideband transistor matching usually imposes higher demands than in the case of real-to-real admittance matching. The real part of CHH 27060 transistor input impedance, shown in Fig. 5, ranges from 200 to 377 Ω. Assuming that the imaginary part was compensated, we must match the varying ∼300 Ω load to the 50 Ω source in a 4–6 GHz band. It's like solving a very hard NP-complete problem. Very good matching (with a reflection coefficient better than 0.1) is achieved in a very narrow frequency band like 5.45 –5.55 GHz. In practice, the reflection coefficient better than 0.31 (one tenth of power lost) is sufficient.

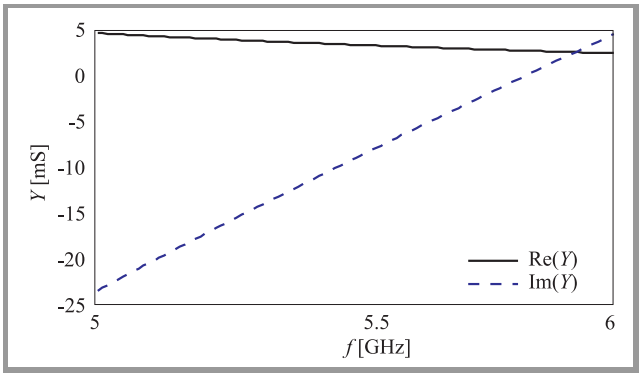


Fig. 5. CHH 27060 transistor input admittance.

Tree $\mu + \lambda$ EA [2] applied to the CHH 27060 complex load matching usually fails. This phenomenon is mostly caused by the failures of a fitness evaluation method. The presence of poles in the circuit response characteristics causes premature convergence of a mini-max solver. Therefore local optimization should be performed initially for a very narrow frequency band, in order to find a feasible starting point. Later, the frequency band can be widened, unless the quality criteria (acceptable power loss) is fulfilled.

Still, after the implementation of fitness evaluation with an expanding bandwidth, $\mu + \lambda$ EA saturates after dozen of generations and the final solution is not promising. EA very seldom generates branching out structures. More likely, EA ends with a cascade or hybrid topology. The change to μ, λ selection scheme improves convergence, although generated topologies are still unsatisfactory.

The averaged results of 12 ES simulations are shown in Figs. 6–8.The bandwidth starts with 20% of a desired frequency band (5.4 – 5.6 GHz). After the success of minimax optimization (when fitness is below acceptable power loss level) the frequency range is incremented by 10% ($\pm$0.1 GHz). The chromosome is qualified according to its bandwidth. In the case of equal bandwidths of the circuit, the value of the reflection coefficient takes the role of fitness (the lower, the better).

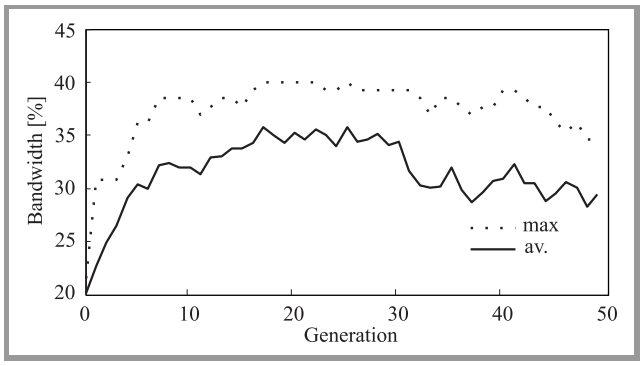


Fig. 6. Maximal and average relative bandwidth versus generation number.

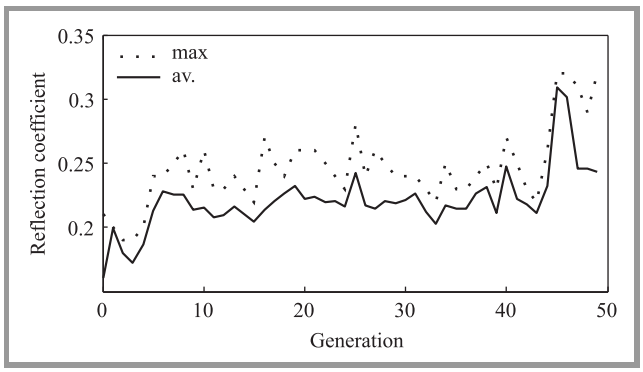


Fig. 7. Maximal and average reflection coefficient versus generation number.

Initially, the device is growing. At about 20th generation the number of transmission lines is very close to its limit (100). After that time neither bandwidth nor power loss is improving. On the contrary, we can observe circuit "collapse" and loss of performance. Analyses of topologies generated by EA show that initially various tree structures are generated. However, later cascade type topologies are growing faster than branches. The structures created by mutation and recombination operators from "branching out" trees are often unusable due to the presence of poles and therefore eliminated. After 20–30 generations only cascade

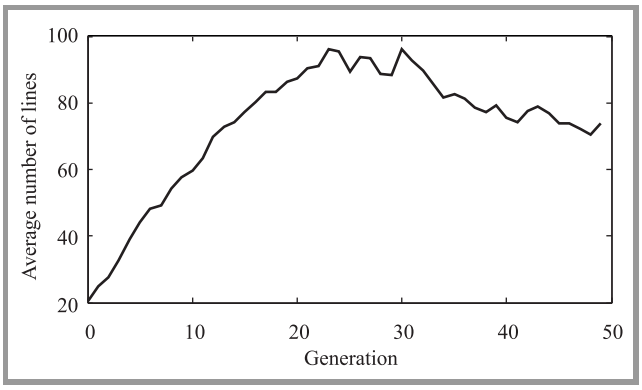


Fig. 8. Average number of transmission lines in chromosome versus generation number.

or hybrid individuals are processed. At the end of optimization, high contraction pressure (the optimal number of transmission lines is assumed to be 50) causes population degradation.

3. Building Block Hypothesis of a Tree Shaped Structure

The tree-shaped matching circuit can be decomposed to the structure shown in Fig. 9. It consists of a main path connecting input (source) with an output (load), called stem, and branches connected to the stem.

In every electronic device some parts of the circuit play different roles. The stem is usually responsible for matching the real part of load admittance, while branches add some susceptance compensating load susceptance. A subcircuit connected with a load is usually responsible for output matching. In more complicated devices, like a diplexer (a multiband filter-splitter), every branch is responsible for transmission in one frequency band. Each block plays its role. We may expect that the whole circuit can be assembled from more or less complex building blocks.

Sensitivity analysis shows also that some blocks strongly affect some aspects of overall performance, while others do not. Uniformly acting evolutionary operators are in this case strongly overloaded. Therefore building blocks should be divided into two classes: stems and branches. Evolutionary operators should be "tailored" to the role of the block.

An evolutionary algorithm, with μ, λ, selection, based on a concept of building blocks, was created. The μ, λ, α EA processes two sets of circuits, the first set consisting of stems and branches, called block library, and the second – population of chromosomes. A chromosome consists of one stem and several branches taken from the block library. The number of blocks in the library is limited to some number α. The initial population and the block library are generated randomly.

Specialized evolutionary operators are used during optimization. There are two types of mutation. The first one

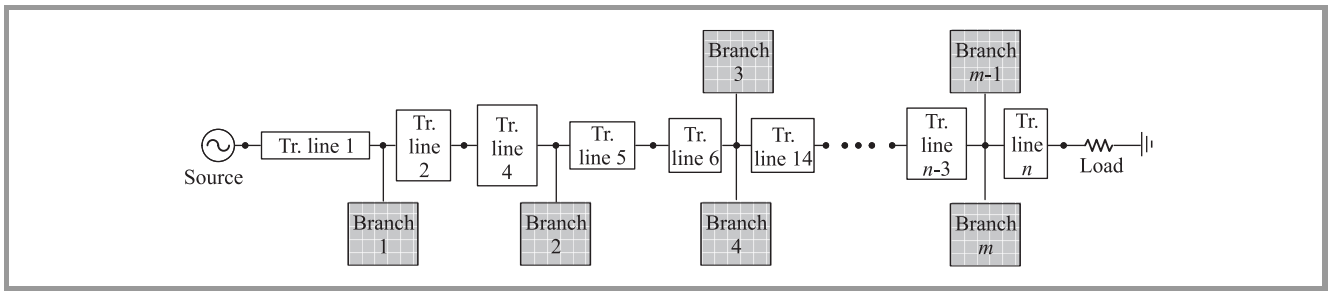


Fig. 9. Steam/branch topology.

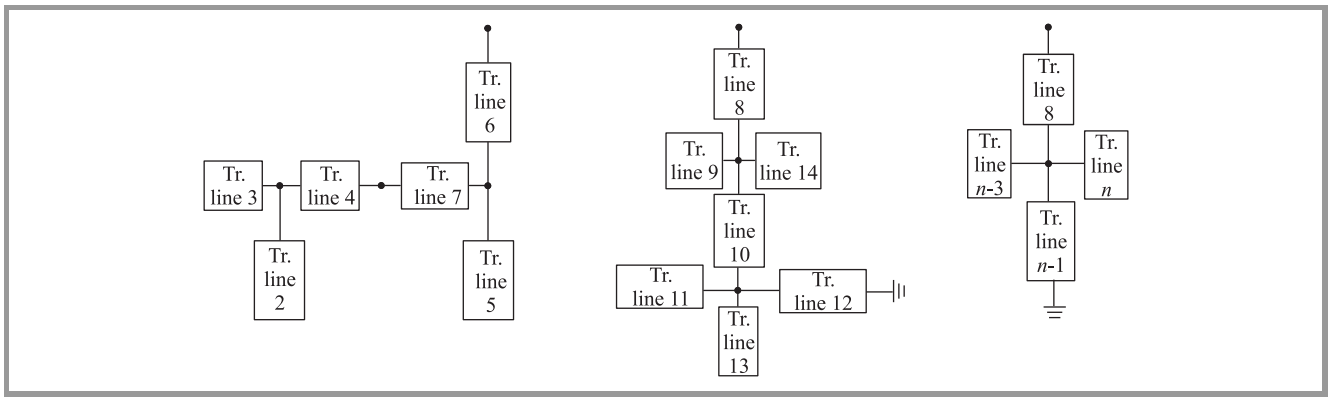


Fig. 10. Various branch topologies.

flips characteristic impedance of the line from low to high impedance. The second type, which is called meta-mutation, exchanges the whole side branch with the one randomly drawn from the library. Recombination (similar to the two point crossover) can be used only for two stems or two branches.

Contraction or expansion removes or adds a single transmission line, respectively. Contraction and expansion probabilities depend on the number of transmission lines in the circuit in order to keep the circuit size within a 10 to 100 elements range. Meta contraction or expansion removes or adds the whole branch (which is drawn from the library), respectively.

Newly created blocks are copied to the offspring library. Parental chromosomes and library blocks are either erased (μ, λ strategy) or copied ($\mu + \lambda$ strategy) to the offspring population and the offspring library, respectively.

Population is evaluated with local optimization [2], using a mini-max solver. Each chromosome is assigned a reached bandwidth and a final reflection coefficient. Library blocks are also qualified. Each one is assigned fitness equal to the fitness of the best chromosome, where the block is used. Two or three member tournament is used for selection, separately within the population and the library. The blocks, used in population after selection, are automatically preserved.

Practical realization of the algorithm ($\mu = 16$, $\lambda = 64$, $\alpha = 64$) applied to the CHH 27060 transistor matching shows overwhelming superiority of the functional blocks approach. A relative bandwidth of 50% (5.25 – 5.75 GHz) is frequently achieved after 15–16 generations for acceptable power loss. Usually, after 4–5 generations the population consists mostly of very good individuals (with 40% or more relative bandwidth).

An example of one of the generated topologies and its frequency response are shown in Figs. 11 and 12, respectively.

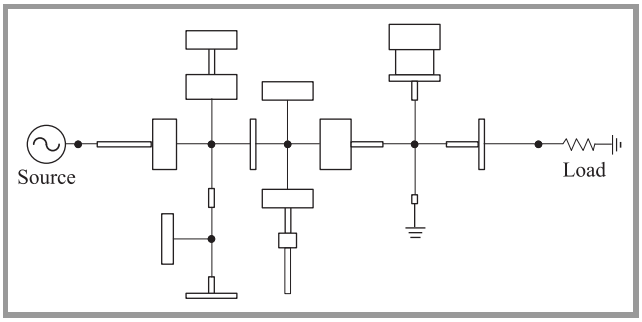


Fig. 11. Example of 5.25 – 5.75 GHz matching circuit.

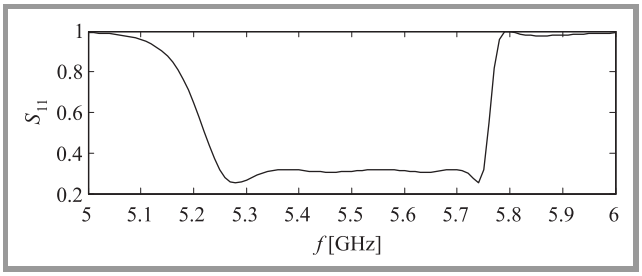


Fig. 12. 5.25 – 5.75 GHz matching circuit frequency characteristic.

It was observed that branches stored within the library are, after the initial period, only slightly modified. The tree topology usually remains untouched, only the number of cascaded sections in arcs is changing. Most likely, all improvement is achieved thanks to stem expansion and metamutation. It seems also that some blocks behave better at a given location inside the network than at another. Therefore we may suspect that EA can be used for the extraction of functional blocks responsible for a certain aspect of circuit behavior. Topologies from classes of functional blocks extracted with EA can be then used in initializing the block library.

References

[1] G. Matthaei, L. Young, andE. M. T. Jones, *Microwave Filters, Impedance Matching Networks and Coupling Structures*, Artech House 1980, pp. 365–372.

[2] J. Arabas and P. Miazga, "Computer aided design of a layout of planar circuits by means of evolutionary algorithms", *J. Comput. Inform. Technol.*, CIT 7, pp. 61–78, 1999.

Przemysław Miazga received the M.Sc. and Ph.D. degrees in Electronics at Warsaw University of Technology in 1980 and 1989, respectively. Since 1989 he has been Assistant Professor at the Institute of Radio Electronics, Warsaw University of Technology. His research interests comprise methods of computer aided design, including modeling and analysis, of microwave and digital circuits and networks, and optimization methods. He is an author of over 40 publications. As an academic lecturer he conducts classes and laboratories on microwave techniques, electromagnetic field theory, antennas, optimization techniques and digital circuits. He is the main author of QW-Optimizer (www.qwed.eu).
E-mail: p.miazga@ire.pw.edu.pl
Institute of Radio Electronics
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

Paper

Evolutionary Algorithm that Designs the DNA Synthesis Procedure

Maciej Michalak and Robert Nowak[a]

[a] *Faculty of Electronics and Information Technology, Warsaw University of Technology, Warsaw, Poland*

Abstract—Chemical synthesis of nucleotide chains is very erroneous for long sequences. Often a gene is constructed from short fragments joined with the use of complementary helper chains. The number of possible potential solutions for a long gene synthesis is very large, therefore a fast automated search is required. In the presented approach a modified method of long DNA construction is proposed. A computer program that searches for an optimal solution in the space of potential synthesis methods has been developed. This software uses an evolutionary algorithm for global optimization and a hill-climbing algorithm for local optimization. The long DNA construction method was tested on random sequences. The results are very promising. The next step is to perform experiments in a biotechnological wet laboratory involving DNA strand synthesis using the method designed by the presented software.

Keywords—bioinformatics, gene synthesis, optimum searching.

1. Introduction

The deoxiribonucleic acid (DNA) strand contains the genetic instructions of majority of living organisms. A need for artificial synthesis of these strands is important in biology and medicine.

A DNA molecule is formed from two separate DNA strands. Each strand may be viewed as a sequence of nucleotides or bases, in which nucleotides are as follows: adenine, cytosine, guanine, and thymine. Two strands are connected by hydrogen bond and this connection is selective – adenine bonds only with thymine and guanine bonds with cytosine. This specific interaction between base pairs is called *complementarity*, it is critical for all the functions of DNA and makes the information in the double-stranded DNA molecule duplicated on each strand. The nucleotide has a natural orientation denoted (according to chemical convention) as 5' and 3' end.

The DNA strand contains genetic information which encodes an amino acid sequence used for protein synthesis. There are 20 amino acids encoded by the sequence of three nucleotides ($4^3 = 64$ possible triplets), called codons. The genetic code is redundant because most amino acids can be encoded in several ways. This redundancy is essential in gene (protein-encoding molecule) synthesis since the DNA sequence can be modified for easier assembly without changing the genetic information. However it must be remembered that the frequencies of codons should satisfy particular preferences of the host (organism used for biosynthesis).

A DNA molecule of a given sequence could be chemically synthesized by repeatedly adding nucleotides [1]. To obtain a desired molecule, nucleotides that have protection groups are sequentially coupled to the growing chain in the order required by the sequence of the product. Despite the yield of this step is about 98%, the length of created strand is limited to 70 *base pairs* (bp) [2]. This restriction is significant for many biological and medical processes, e.g., for peptide biosynthesis because genes length is usually greater than 300 bp, typically 1000 bp.

Long DNA molecules synthesis techniques have been developed to obtain such molecules from smaller parts. A method of long DNA molecule construction from smaller DNA molecules is called *protocol*. Polymerase cycling assembly (PCA) is the most widely used technique [3], [4], [5] to produce long (e.g. 1000 bp) DNA chains. This technique uses the same reactions and reagents as polymerase chain reaction (PCR), but additionally the DNA polymerase [2] amplifies a complete sequence of DNA, as depicted in Fig. 1. The typical length of a fragment is 50 bp and the overlapping area is 20 bp. The correct syntheses of 800 bp strands have been reported [6].

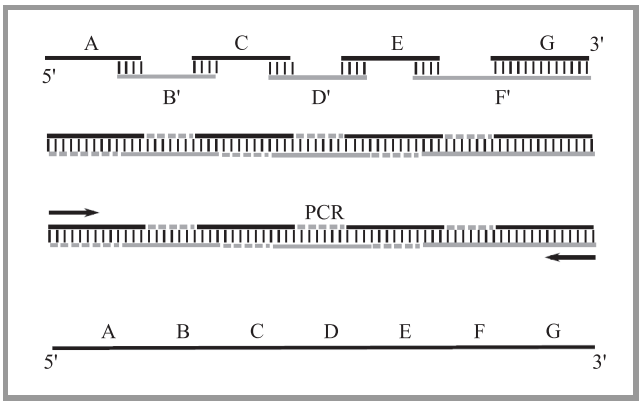


Fig. 1. The polymerase cycling assembly (PCA) used to create long DNA molecules from smaller parts. The shorter fragments are chemically synthesized, then create a longer molecule (hybridization), next DNA polymerase produces a complete sequence of DNA, then PCR with specific starters (primers) amplifies the long DNA strands. Finally, the isolation based on molecule length is performed. The fragments of complementary strand are denoted using a prime symbol.

In the presented solution we consider protocols different from PCA. Shorter fragments are separately chemically synthesized, as in the PCA, but the possibility of synthe-

sizing the parts of different length as well as carrying out the reaction at a decreasing temperature [7] is considered. Next, we accept changes in the resulting sequence if these changes do not affect the protein sequence, i.e., we include the possibility of codon substitution. Finally, the possibility of performing the reaction in separate tubes and then joining the results is considered. Our application generates a much larger number of protocols to synthesize a given strand than a typical PCA oriented algorithm. Each protocol is assigned a quality measure, which is optimized in the space of protocols using a combination of evolutionary algorithm for global optimization and a hill-climbing algorithm to perform fine tuning.

2. Synthesis Protocols

The base synthesis protocol consists of fragments of the desired DNA strand (molecules labeled A, C, E, G in Fig. 2) and the *helper chains* (molecules from complementary strands: B', D' and F' in Fig. 2). The helper chains are complementary to the two adjacent fragments and are used for joining.

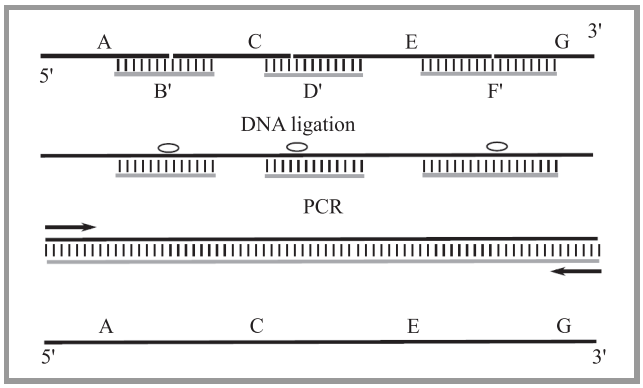


Fig. 2. Long DNA base synthesis protocol. Shorter fragments are chemically synthesized, then during hybridization and ligation the longer molecule is formed, finally PCR with specific starters (primers) amplifies the correct DNA strands. If a correct molecule cannot be created because the fragments fold in an incorrect way, the reaction is performed in separate tubes (complex protocol).

Every pair of DNA strands (also of a single strand) folds in the temperature dependent on its nucleotide sequences, therefore the idea is to start from the temperature high enough to unfold all the fragments and then gradually decrease it so that strands can join, as shown in Fig. 3.

Of course, there is no guarantee for every set of fragments that they will join in a desired order. We examine every possible pair of strand from the solution – both fragments and helper chains – including pairs consisting of two identical molecules (fragment with itself). These examinations use DNA secondary structure prediction algorithms, which tells us how the strands can fold and what temperature is needed to unfold them. The results ordered by

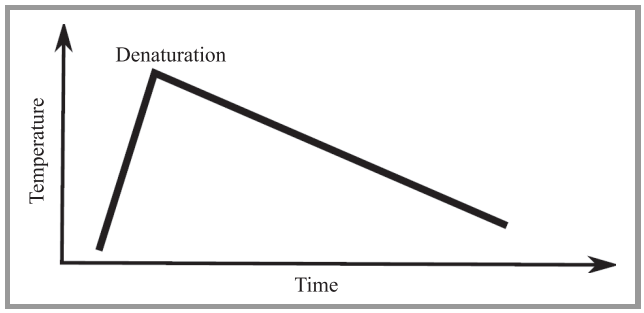


Fig. 3. Temperature in probe during synthesize reaction. Firstly, dilution is heated to denaturate, then cooled slowly.

the descending temperatures define the sequence of strands' foldings.

It is desirable that each time the fragments in solution are either correctly folded or unfolded. The correctly folded fragment is created by a pair that folds into a coherent double-stranded structure with dangling single-stranded ends. If the pair contains a strand that is already folded, then it all combines into one structure, i.e., into a single fragment with two adjacent helper chains or two fragments connected by the helper chain. Ultimately, the whole long DNA strand is composed and can be subjected to further treatment i.e., to build bounds between the chains by DNA ligase, as shown in Fig. 2.

Simulations for the base protocol for random sequences of length 60 – 80 bp have shown that about 1% of possible synthesis protocols are proper. In these simulations, we assume that the random sequence is created from 3 strands, two fragments and one helper chain. The fragments were calculated by drawing, with uniform distribution, the position on the input sequence that divides it into two subsequences of length 20 – 40 bp. The helping chain was complementary to a region near this position. Most of base protocols are unsuitable for the synthesis due to incorrect foldings. Fortunately the number of good solutions can be increased by extending the base protocol to a complex one. Long strand assembling can be done in steps, as shown in Fig. 4. In each step, conflicting strands (i.e., strands forming improper pair) are separated into different probes

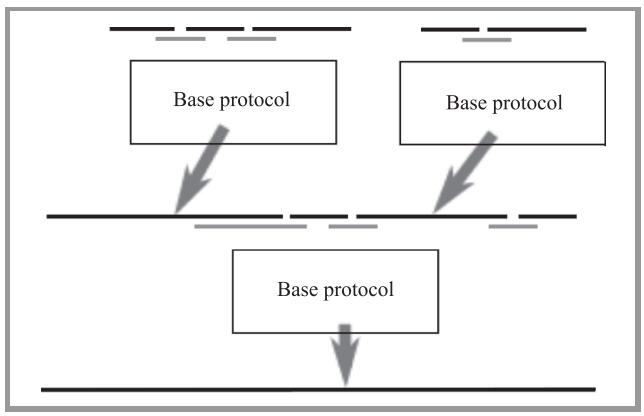


Fig. 4. Long DNA complex protocol, the multiple tubes are used.

so the base protocol can be applied for each probe. Chains formed by the base protocol become the input strands for the next and the procedure is repeated until a full long strand is obtained.

In order to determine the percentage of proper solutions five random sequences were generated. For each sequence four ranges of fragment length were considered: 6 – 12, 10 – 20, 14 – 28, 20 – 40. For every variant sequences were truncated to the length allowing them to be constructed from a given number of fragments. For each case (i.e., fragment length: 10 – 20, three fragments, first sequence truncated to 45 nucleotides) 10,000 solution candidates were randomized, simulated and searched for proper base protocols. The considered sample was relatively small, therefore the results for different variants of simulations were strongly dispersal, but some trends were visible. As it can be seen in Table 1, with the increase of the fragmentation size, the percentage of proper solutions strongly decreases. However, the contribution of complex protocols to all proper protocols grows.

Table 1
Percentage of proper solutions

Number of fragments	Base protocols [%]	Complex protocols [%]
2	5 – 20	0
3	0.2 – 5	0.02 – 0.5
4	0.05 – 0.3	0.02 – 0.25
5	0 – 0.1	0.01 – 0.13

The simulations revealed that the number of proper protocols grew insignificantly. The reason for this poor result lies in the length of helper chains in the consecutive steps. Probability of folding increases with the total length of corresponding strands, therefore two fragments are more likely to fold with each other than with the helper chain which is at least two times shorter. The solution to this problem is to replace helper chains with longer ones and, eventually, synthesize them in separate probes as well. As the next simulations indicate, this strategy induced the increase of the number of proper protocols by up to 50% for protocols with 3 fragments and more for a larger number of fragments.

Obviously, the complex synthesis protocol is more expensive than the base synthesis protocol. The execution of each step in the laboratory takes about 24 hours, multiple probes are needed and the number of required nucleotides is noticeably greater.

3. Implementation Issues

The application searches the synthesis protocol in the space of possible basic and compound protocols using a combination of evolutionary algorithm and a hill-climbing algorithm.

3.1. Evolutionary Algorithm

Intuitively, the synthesis protocol is a solution candidate (individual) for the evolutionary algorithm [8]. The individual is represented by the target DNA strand (in a form of sequence of nucleotides), the collection of positions describing the places of fragments separation and the collection of helper chains. This set of data should meet the constraints on minimum and maximum strand length. Therefore, the initial population is a set of protocols which differ in three aspects:

– codon sequence (nucleotide sequence encoding given peptide),

– chain fragmentation,

– helper chain selection.

The initiation process first creates a the target DNA strand. The codons were chosen randomly with uniform distribution. Next, places of fragments separation are chosen randomly, with uniform distribution, having regard to the minimum and maximum length of the fragment. Finally, the helper chains are calculated, their length is also random (uniform distribution). There is an option not to randomize a codon sequence, but to optimize it earlier with the use of local search – the decision is left for the user.

All individuals are chosen for reproduction. Every parent produces a single mutated child by choosing, with the equal probability, one of the following transformations:

– replacement of the random nucleotide triplet from the sequence (change a codon),

– increment or decrement of a random position describing the separation into fragments (one fragment is shortened and the other is extended),

– one helper is shortened or extended or moving left or right.

The fitness of the individual depends on:

– the root mean square of the differences between requested and obtained codon frequencies,

– the number of nucleotides needed,

– the number of protocol steps,

and can be represented by the formula:

$$F = \frac{1}{(1 + F_C + F_N + F_L)},$$

$$F_C = w_C \sum_{j=0}^{n} \frac{(C_{rj} - C_{ij})^2}{n},$$

$$F_N = w_N \frac{N - N_{\min}}{N_{\max} - N_{\min}},$$

$$F_L = w_L \frac{L-1}{2},$$

where: C_r – required codon frequency, each organism has optimal codon frequency, available at [9], C_i – individual's codon frequency, N – number of used nucleotides, L – number of protocol's steps, w – weight of the partial evaluation.
In every generation parents and their offspring are selected for succession. There are three methods of selection to choose: ranking, tournament ($k = 2$) and proportional.

3.2. Implementation

In order to compute individual's fitness, our application is able to simulate the process of synthesis, check for eventual conflicting pairs and switch from the base to the complex protocol. Due to a good trade-off between the availability of optimization techniques, portability and extensibility, C++ language for implementation was chosen. The application is set up on the Django server which provides a graphic user interface via HTTP.
The proposed algorithm for protocol calculation is quite complex so its execution is expensive. Single individual's fitness calculation time for real-life input needs seconds or even minutes, e.g., on a typical 3 GHz CPU core, it takes an average of 90 s to compute 1000 bp gene synthesis simulation. We decided to use distributed calculations (many computers) and take advantage of GPU power. Distributed calculations use CORBA (Common Object Request Broker Architecture), individuals' fitness is considered independently, so the calculations can be delegated to different cores, processors and computers.
The DNA strands' joining order is examined using the Zuker algorithm [10] for the secondary structure prediction. It was implemented in a bottom-up dynamic programming manner. The idea is to calculate – with the use of the delivered thermodynamic data – the free energy of the smallest sub-chains and iteratively find optimal solutions to increasingly longer sub-chains. An optimal solution is characterized by the minimum free energy which corresponds to the set of base pairs forming a secondary structure.
The complexity of Zuker algorithm is $\Theta(n^4)$ and with the use of heuristics can be simplified to $\Theta(n^3 + 30^2 n^2)$. This is a highly expensive operation and indeed, the profiling tool has shown that it takes over 99% of the whole application execution time. It should be noticed that the mutation can affect changes only on 1–3 strands in a probe, therefore the best effort was made in order to minimize the number of such operation calls by storing the results and making them inheritable by mutated individuals. As a result, when the 1000 bp gene synthesis protocol is once evaluated, its derivatives are calculated 10 times faster. The improvement for different gene lengths is shown in Fig. 5.
To speed up the DNA secondary structure prediction calculations, the module was implemented in CUDA (Compute Unified Device Architecture) in order to take advantage of the GPU power. As it was explained, the algorithm computes the solution for sub-chains starting from the shortest ones. For each length every sub-chain is considered independently, therefore it is possible to make these calculations

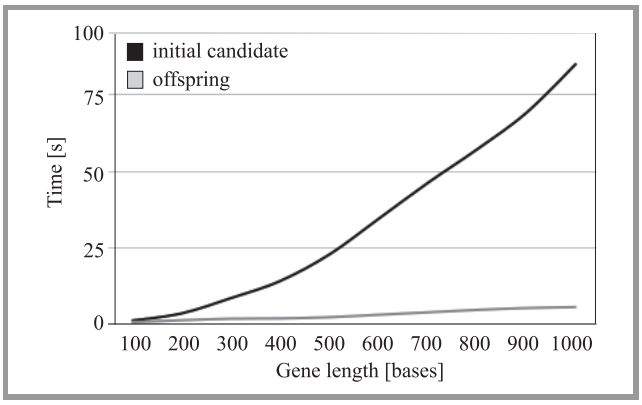


Fig. 5. Synthesis protocol evaluation time versus DNA length.

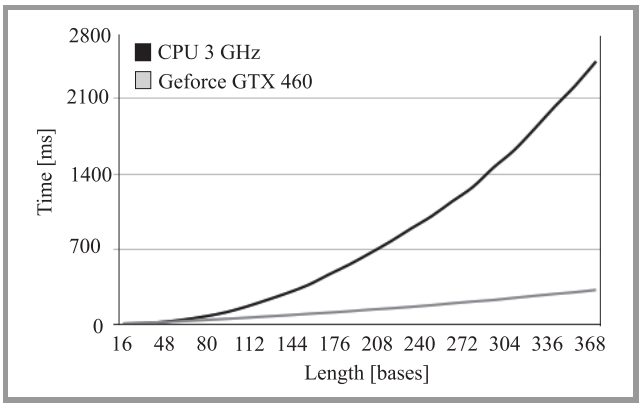


Fig. 6. Zuker algorithm's execution time for 3 GHz CPU an nVidia Geforce GTX 460 versus DNA length

parallel. As the Fig. 6. indicates, the longer strand (pair of strands) the greater speed increase – i.e., the 384 bp sequence is calculated 12 times faster on Geforce GTX 460 than on 3 GHz CPU core. It was measured that 1000 bp gene synthesis simulation with the GPU use takes 40 s on average and is 2.5 faster than CPU.

4. Results

The application was tested using a random sequence of the length 50 as an input gene, the minimum and maximum fragment length equal to 10 and 20, respectively. The search process was performed by a trial and error method and by the evolutionary algorithm in three variants: with ranking, proportional and tournament selection. For the purpose of comparison, all optimizations were set to evaluate 10,000 solution candidates, therefore, each variant was conducted for 10, 100 and 1000 generations with population size of 1000, 100 and 10 respectively. Table 2 shows the results of these experiments.
As it can be seen, when one simply draws 10,000 solution candidates (Monte Carlo method), they get only 150 correct solutions. The use of evolutionary algorithm increases this number significantly, which may indicate similarity of correct solutions.

The results of these experiments are promising, but one must remember that with the increase of the requested sequence length the time complexity grows. Due to the greater fragmentation the percentage of correct protocols also decreases. Search for the optimal synthesis protocol for real genes is greatly time consuming and that is what motivated the authors to create implementation which maximally takes advantage of all available resources.

Table 2
Properties of the solution space coverage

Method	Uniqueness [%]	Correctness [%]	Correct uniqueness [%]	Unique correct
Trial and error	100	1.5	100	150
Ranking				
1000×10	86.6	24.2	70.7	1711
100×100	51.4	55.6	45.3	2519
10×1000	20.7	33.1	12.4	410
Proportional				
1000×10	64.6	57.0	51.0	2907
100×100	67.4	80.6	64.6	5207
10×1000	68.1	69.2	65.1	4505
Tournament				
1000×10	84.3	20.4	72.5	1479
100×100	68.9	55.8	59.8	3337
10×1000	54.4	50.1	43.8	2194

5. Conclusions

The application would be confirmed by larger studies involving comparison with the existing solutions [11], and experimental work on DNA molecule synthesis. Especially, an experiment with the use of the proposed method in a genetic laboratory is required. Realization of the calculated synthesis protocol of a known gene and an examination of the product of this realization would confirm the credibility and the usefulness of the proposed method.

References

[1] S. L. Beaucage and R. P. Iyer, "Advances in the synthesis of oligonucleotides by the phosphoramidite approach, *Tetrahedron*, vol. 48, pp. 2223–2311, 1992.

[2] M. Stryer, *Biochemistry*. Freeman, 1995.

[3] W. Stemmer *et al.*, US Patent No. 6,368,861, 2002.

[4] W. P. C. Stemmer, A. Crameri, K. D. Ha, T. M. Brennan, and H. L. Heyneker, "Single-step assembly of a gene and entire plasmid from large numbers of oligodeoxyribonucleotides", *Gene*, vol. 164, no. 1, pp. 49–53, 1995.

[5] S. Huntsman, "Towards the batch synthesis of long DNA", Tech. Rep. ADA409078, Institute for Defense Analyses, Alexandria, US, 2002.

[6] S. J. Kodumal *et al.*, "Total synthesis of long DNA sequences: synthesis of a contiguous 32-kb polyketide synthase gene cluster", in *Proc. Nat. Academy Sci. United States of America*, vol. 101, no. 44, p. 15573, 2004.

[7] R. Nowak and M. Romaniuk, "Long DNA strands synthesis optimizing", *Prace Naukowe Politechnnniki Warszawskiej, z. 165, Evolutionary Computation and Global Optimization*, 2008, pp. 157–164.

[8] G. Morawski and R. Nowak, "Porównanie algorytmów optymalizujących sekwencje DNA kodujące bialka ze szczególnym uwzględnieniem algorytmu ewolucyjnego", Tech. Rep., Warsaw University of Technology, Institute of Electronic Systems, Warsaw, 2008.

[9] "Codon usage database" [Online]. Available: http://www.kazusa.or.jp/codon/

[10] M. Zuker and P. Stiegler, "Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information", *Nucleic Acids Res.*, vol. 9, no. 1, p. 133, 1981.

[11] D. Hoover and J. Lubkowski, "DNAworks: an automated method for designing oligonucleotides for PCR-based gene synthesis", *Nucleic Acids Res.*, no. 30, 2002.

Robert Nowak received the M.Sc. (1999) and Ph.D. (2004) in Computer Science from Warsaw University of Technology. He is Assistant Professor at Artificial Intelligence Division, Institute Electronic Systems, WUT. He participated and coordinated in a number of research and commercial projects in the military, biology, medicine and energy field. His interests are artificial intelligence, computational biology, data fusion, risk management and computer software development. He is the author of 40 papers and one book.
E-mail: r.m.nowak@elka.pw.edu.pl
Faculty of Electronics and Information Technology
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

Maciej Michalak was born in 1984. He studied information technology at Warsaw High School of Economy and Information Technologies and on the Faculty of Electronics and Information Technology at Warsaw University of Technology. In 2009 he wrote his B.Sc. thesis devoted to gene synthesis. He has been continuing this topic in his M.Sc. thesis. In September 2011 he took part in the XIII National Conference of Evolutionary Algorithms and Global Optimization where he introduced some results of his work on gene synthesis optimization. Simultaneously he works at Horus s.c. where he develops business management systems for telecommunication companies.
E-mail: M.Michalak@stud.elka.pw.edu.pl
Faculty of Electronics and Information Technology
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

Paper

Localization in Wireless Sensor Networks Using Heuristic Optimization Techniques

Ewa Niewiadomska-Szynkiewicz, Michał Marks, and Mariusz Kamola

Institute of Control and Computation Engineering, Warsaw University of Technology, Warsaw, Poland
Research Academic Computer Network (NASK), Warsaw, Poland

Abstract—Many applications of wireless sensor networks (WSN) require information about the geographic location of each sensor node. Devices that form WSN are expected to be remotely deployed in large numbers in a sensing field, and to self-organize to perform sensing and acting task. The goal of localization is to assign geographic coordinates to each device with unknown position in the deployment area. Recently, the popular strategy is to apply optimization algorithms to solve the localization problem. In this paper, we address issues associated with the application of heuristic techniques to accurate localization of nodes in a WSN system. We survey and discuss the location systems based on simulated annealing, genetic algorithms and evolutionary strategies. Finally, we describe and evaluate our methods that combine trilateration and heuristic optimization.

Keywords—evolutionary strategy, genetic algorithm, localization, location systems, nonconvex optimization, simulated annealing, wireless sensor network.

1. Introduction

The primary function of a location estimation method is to calculate the geographic coordinates of network nodes with unknown position in the deployment area. Most applications of wireless sensor networks (WSN) require the correlation of sensor measurements with physical locations, even if the accessible knowledge about positions of nodes is only approximate. Moreover, information about current locations are used in geographical-based routing, data aggregation and various network services. Hence, self-organization and localization capabilities are one of the most important requirements in sensor networks.

Information on the location of nodes can be obtained in two ways:

- recording data on the location of nodes during their distribution,
- fitting nodes with a GPS system.

Both methods have significant defects. Typical WSN usually consists of a large number of sensors that should be densely distributed in a sensing field. The large number of nodes usually precludes manual configuration. Moreover, manually recording and entering positions of each sensor node is impractical and impossible in many applications, in which sensors are distributed randomly in ad hoc fashion, which is cheaper, and in some cases the only possible solution. Moreover, this method cannot be used in mobile networks where nodes can travel. Another solution is to collect data on the location of sensors by means of GPS devices. This solution can be used in different types of networks, including mobile ones. Unfortunately, it is very costly, both due to the price of GPS receivers, and to the increased requirements related to power consumption that may decrease the lifetime of a WSN. Moreover, adding an additional receiver increases the size and weight of the total device (network node).

Due to the drawbacks of presented solutions, many automated location systems for assigning geographic coordinates to each node have been developed. All these schemes should work with inexpensive off-the-shelf hardware, minimal energy requirements, scale to large networks, and also achieve good accuracy in the presence of irregularities and give the solution in the short time. Various localization strategies for WSNs have been developed [1]–[4]. Position calculation can be conducted using one machine collecting data from the network (base station) or calculations can be distributed. In centralized schemes, data collected in a network is transmitted to the central machine that calculates the positions of nodes with unknown location. Distributed algorithms relay only on local measurements – each non-anchor node estimates its position based on data gathered from its neighbors.

The main contribution of this paper is to provide a survey of localization strategies and systems using nonconvex, heuristic optimization techniques to solve the localization problem. We focus on centralized schemes with heuristic optimization, and present the location systems based on simulated annealing, genetic algorithms and evolutionary strategies.

The paper is organized as follows. The introduction to localization techniques is provided in Section 2. The localization problem is formulated in Section 3. Strategies and location systems based on heuristic optimization are investigated in Sections 4 and 5. In Section 6 we evaluate two our location systems TGA and TSA. The paper concludes in Section 7.

2. Localization Techniques

A number of localization methods and location systems are described in the literature. A general survey is found

in [1]–[4]. The localization techniques can be classified with respect to various criterion. They differ on network architecture and configuration, hardware components, nodes properties and deployment, measurement and calculation methods, computing organization, assumed localization precision, etc. Recently proposed localization techniques consist in identification of approximate location of nodes based on merely partial information on the location of the set of nodes in a sensor network.

Let us consider a network formed by $L = M + N$ sensors; M anchor nodes and N non-anchor nodes. The definitions of anchor and non-anchor nodes are as follows:

anchor node – a node that is aware of its own location, either through GPS or manual recording and entering position during deployment. Its position is expressed as n-dimensional coordinates $a_k \in \Re^n$, $k = 1,\ldots,M$.

non-anchor node – a node that is not aware of its own location in the deployment area. Its position is expressed as n-dimensional coordinates $x_j \in \Re^n$, $j = 1,\ldots,N$.

The goal of a location system is to estimate coordinate vectors of all N non-anchor nodes.

In general, localization schemes operate in two stages:

Stage 1: Inter-node distances estimation based on hop connection information (hop counting) or true physical distance calculation based on inter-node transmissions and measurements.

Stage 2: Transformation of calculated distances into geographic coordinates of nodes forming the network.

2.1. Stage 1: Inter-node Distance Estimation

With regard to hardware's capabilities of given nodes, and the mechanisms used for estimating inter-node distances in *Stage 1* of the localization scheme, we divide the localization algorithms into two categories: range-free (connectivity-based) methods and range-based (distance-based) methods.

The *range-free* algorithm uses only connectivity information to locate the entire sensor network. The popular solutions are hop-counting techniques. Assume that each anchor node a_k, $k = 1,\ldots,M$ exchanges messages with other nodes. Hence, the distances in hops h_{kl} between each pair (k,l) of anchors in the network are estimated. Next, each anchor computes an average size for one hop $c_k = \frac{\sum_{l \in S_k} ||a_k - a_l||}{\sum_{l \in S_k} h_{kl}}$, $k \neq l$, where S_k denotes a set of anchors located within a transmission range of r_k, $S_k = \{(k,l) : ||a_k - a_l|| \leq r_k\}$, $l = 1,\ldots,M$. The calculated values are broadcasted into the network, and the inter-node distances expressed in hops are estimated.

The *range-based* algorithm uses absolute point-to-point distance estimates (range) or angle estimates in location calculation. Hence, distance-based methods require the additional equipment but through that we can reach much better resolution than in case of range-free ones. In accordance with the available hardware they exploit Angle of Arrival (AoA), Time of Arrival (ToA), Time Difference of Arrival (TDoA) and Received Signal Strength Indicator (RSSI). The survey and discussion of the most popular measurement technologies is available in [1], [5]–[9]. The common technique based on a standard feature found in most wireless devices is RSSI. The method for distance estimation based on a signal propagation model and RSSI is described in [10]. The advantage of this method is low cost (no additional hardware), easy configuration, calibration and deployment. The disadvantage is low level of measurement accuracy because of high variability of RSSI value. In real-world channels, multipath signals and shadowing are two major sources of environment dependence in the measured RSSI.

2.2. Stage 2: Geographic Coordinates Estimation

In *Stage 2* of the localization scheme the calculated distances are converted into geographic coordinates of network nodes. Different less and more complicated techniques may be used to perform calculations. The coordinates of nodes can be calculated using: geometrical techniques, multidimensional scaling, stochastic proximity embedding, optimization algorithms (nonlinear, quadratic and linear), hybrid schemes that use two different techniques.

The *geometrical techniques* give solutions to a set of nonlinear equations. The most popular are: triangulation, trilateration and multitrilateration. The simple and popular location system implementing the trilateration method is called Ad-hoc Positioning System (APS). It is described in [11]. *Multitrilateration* methods are proposed to reduce limitations of the typical trilateration scheme. *Atomic multitrilateration* incorporates distance measurements from multiple neighbors. The idea of *iterative multitrilateration* is to repeat trilateration for increased number of anchor nodes (every iteration each non-anchor node with estimated position changes its role to anchor). The philosophy of localization techniques based on *multidimensional scaling* (MDS) and *stochastic proximity embedding* SPE is to transform a mathematical model to convert distance information into the coordinate vector. The common idea of other methods is formulating the localization problem as a nonlinear, nonconvex optimization task solved by *global optimization* (often heuristic) solvers or relaxing the resulting problem as a convex optimization problem solved by *quadratic* or *linear* solvers. Recently, a popular grup consists of *hybrid systems* that use more than one technique to estimate location, i.e., results of initial localization are refined using another localization method.

The survey, evaluation and detailed discussion of the most popular approaches to geographical coordinates estimation and location systems are found in [1], [3], [7], [11]–[14]. In this paper we present a short overview of location systems using heuristic techniques. We start our presentation from the formulation of the mathematical model of the WSN localization problem in Section 3.

2.3. Flip Ambiguity Phenomenon

In many WSN applications it can be observed that some nodes can not be uniquely localizable. These location errors are often driven by so-called flip ambiguity phenomenon, demonstrated in Fig. 1. As the neighbors of node D are almost collinear, and the inter-node distances are estimated with measurement errors the localization algorithm usually calculates the incorrect location, i.e., D' instead of D in Fig. 1. It is obvious, the position of this node can be reflected with no significant change in the performance function in Eq. (1). This observation is discussed by many researchers, and different methods to solve this problem are proposed.

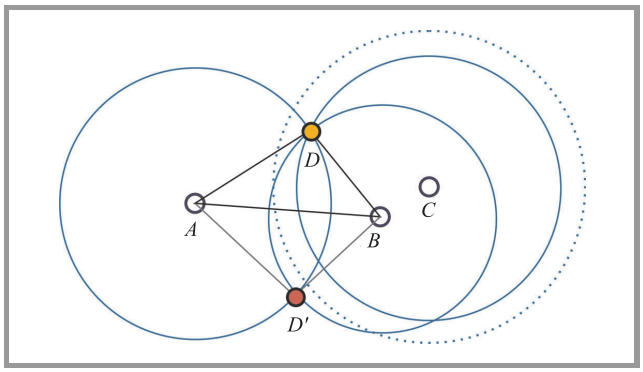


Fig. 1. Flip ambiguity phenomenon in WSN localization.

The popular approach to compensate location errors driven by a flip ambiguity phenomenon is to modify the basic localization algorithm or extend the localization process in a correction phase.

3. Mathematical Model of WSN Localization Problem

The standard approach is to formulate a localization problem as the optimization task with the nonlinear performance function J_N:

$$\min_{\hat{x}}\{J_N = \sum_{k=1}^{M}\sum_{j\in S_k}(\hat{d}_{kj}-\tilde{d}_{kj})^2 + \sum_{i=1}^{N}\sum_{j\in S_i}(\hat{d}_{ij}-\tilde{d}_{ij})^2\}, \tag{1}$$

where $\hat{d}_{kj} = ||a_k - \hat{x}_j||$, $\hat{d}_{ij} = ||\hat{x}_i - \hat{x}_j||$, a_k denotes the real position of the anchor-node k, $\hat{x}_i$ and $\hat{x}_j$ denote, respectively, the estimated positions of nodes i and j, $\tilde{d}_{kj}$ and $\tilde{d}_{ij}$ the estimated distances between pairs of nodes calculated based on measurements, and S_i, S_k sets of neighboring nodes defined as follows:

$$\begin{aligned} S_k &= \{(k,j) : ||a_k - x_j|| \le r_k\}, \quad j = 1,\ldots,N \\ S_i &= \{(i,j) : ||x_i - x_j|| \le r_i\}, \quad j = 1,\ldots,N, \end{aligned} \tag{2}$$

where x_i and x_j denote real positions of nodes with unknown locations and r_i and r_k their transmission ranges.

Various optimization techniques are used to solve the optimization problem Eq. (1). As it was mentioned the most popular approaches are: quadratic programming, linear programming, nonlinear and nonconvex optimization techniques. The first class of methods transforms the original nonconvex formulation Eq. (1) into quadratic problem and apply quadratic programming to solve the reformulated problem. A localization system OPDMQP using quadratic programming is described in [15]. The second class of methods relaxes the problem (1) in order to obtain a semidefinite programming SDP [12] or a second-order cone programming SOCP [16]. The existing linear solvers (usually interior point methods) are used to solve the transformed problem. In case of both mentioned approaches the problem with local minima can influence the solution. The third, commonly used strategy is to apply global optimization algorithms to avoid local minima of the performance function J_N in Eq. (1). Numerous approaches are proposed and described in the literature. Many researchers suggest to use popular heuristic methods, i.e., deterministic, such as tabu search TS and stochastic, such as simulated annealing SA, genetic algorithm GA, evolutionary algorithm EA or particle swarm optimization PSO to calculate the location estimates.

4. Location Systems with Heuristic Optimization – A Survey

In this section we discuss selected location systems using heuristic optimization.

4.1. Simulated Annealing based Systems

Results of simulated annealing to location estimation are provided in several papers. The simulated annealing based localization system (SAL) developed by Kannan *et al.* is described in [13]. It is the range-based system. The authors propose different modifications of basic SA to improve the results and speed up calculations. They show that for ideal measurements without any noise introduced to the system when inter-node distances are calculated with 100% accuracy the location estimates are computed with 100% accuracy, too. The measurement noise influences the results but they are quite accurate. The serious deterioration of results is observed in case when flip ambiguity situation occurs. To compensate location errors driven by the flip ambiguity phenomenon Kannan *et al.* propose a method [14], in which the localization is done through two phases, i.e., in the first phase the coordinate vectors are calculated (*localization phase*), in the second phase the errors caused by the flip ambiguity are compensated (*refinement phase*). Hence, two executions of the simulated annealing method are performed. The goal of the first execution is to solve the optimization problem Eq. (1), and calculate the coordinates of the target nodes. The second phase is performed only on non-uniquely localizable nodes. The goal of this phase is to identify these nodes, and refine their location

estimates calculated in the first phase. The SA algorithm is used again to solve the optimization problem with modified objective function defined in Eq. (3). The function value is increased when a node is placed in a wrong neighborhood.

$$J_{F_K} = \sum_{k=1}^{M} \Big(\sum_{j \in S_k} (\hat{d}_{kj} - \tilde{d}_{kj})^2 + \sum_{\substack{j \in S_k \\ \tilde{d}_{kj} < r_k}} (\hat{d}_{kj} - r_k)^2 \Big) + \sum_{i=1}^{N} \Big(\sum_{j \in S_i} (\hat{d}_{ij} - \tilde{d}_{ij})^2 + \sum_{\substack{j \in S_i \\ \tilde{d}_{ij} < r_i}} (\hat{d}_{ij} - r_i)^2 \Big), \tag{3}$$

where r_k and r_i denote the transmission ranges of the nodes k and i.

In summary, the goal of the localization phase is to calculate the accurate coordinate vectors of uniquely localizable nodes and initial coordinate estimates of non-uniquely localizable nodes. The goal of the refinement phase is to increase the accuracy of the location estimation of all non-uniquely localizable nodes.

4.2. Genetic Algorithm and Evolutionary Strategy based Systems

Another approach is to use various versions of genetic algorithm or evolutionary algorithm. The genetic algorithm based localization system (GAL) developed by Zhang *et al.* is described in [17]. The authors propose different modifications of basic GA to improve the results and speed up calculations. Two new genetic operators are adopted: a single-vertex-neighborhood mutation and a descend-based arithmetic crossover. The method was evaluated on several example problems. The authors claim that it outperforms the SDPL method (a semi-definite programming with gradient search localization) and simulated annealing based localization SAL [13].

The range-based localization system with the distances estimation based on RSSI and Imperialist Competitive Algorithm (ICA) used to calculate the coordinate vectors is presented by Sayadnavard *et al.* ICA is a new evolutionary algorithm that is based on the simulation of a human's socio-political evolution. The simulation results presented in [18] highlight that ICA-based approach considerably outperforms the APS system. Moreover, it calculates estimates characterized by higher accuracy than the ones obtained by the PSO-based localization scheme using RSSI ranging technique [19] but with more computational time.

The application of evolutionary computation to estimate locations of nodes is described in [20]. The estimates of inter-node distances calculated due to RSSI measurement are transmitted to the central unit. The central unit employs evolutionary algorithm to estimate the locations of nodes based on gathered information about inter-node distances. The authors claim that their approach gives a reasonable solution even for poor RSSI measurement but they do not provide any comparison to solutions developed by other researchers. Moreover, they do not consider the flip ambiguity situation.

Vecchio *et al.* developed a location system, in which two-objective localization problem is formulated and evolutionary algorithm is used to solve it [21]. Due to the fact that the connectivity in WSN is not sufficiently high the authors propose some modifications to a basic EA. The algorithm takes into account both the localization accuracy and certain topological constraints induced by connectivity considerations during a location estimation. In proposed scheme two performance functions are concurrently minimized. The first one is defined in Eq. (1). The second cost function is defined as follows:

$$J_{F_V} = \sum_{k=1}^{M} \Big(\sum_{j \in S_k} \delta_{kj} + \sum_{j \in \tilde{S}_k} (1 - \delta_{kj}) \Big) + \sum_{i=1}^{N} \Big(\sum_{j \in S_i} \delta_{ij} + \sum_{j \in \tilde{S}_i} (1 - \delta_{ij}) \Big), \tag{4}$$

where $\delta_{ij} = 1$ if $\tilde{d}_{ij} > r_i$ and 0 otherwise, and $\tilde{S}_i = \{(i, j) : ||x_i - x_j|| > r_i\}$. Hence, the goal of the J_{F_V} function is to count the number of connectivity constraints that are not satisfied by the current estimated locations of target nodes. The authors claim that their approach outperforms the SA-based localization algorithm proposed in [14]. The simulation results presented in [21] confirm the good performance of the algorithm.

5. Hybrid Methods

In this section we investigate selected systems using at least two different methods to calculate estimates of nodes location in a network. Moreover we describe our methods that combine geometrical techniques along with a heuristic optimization.

5.1. Hybrid Methods – A Survey

The last presented strategies are hybrid schemes that combine commonly used methods for computing geographic coordinates of nodes. In most approaches trilateration or multitrilateration is used to calculate an initial solution, which is improved in the next step. Tam *et al.* developed a two-phase method that is described in [22]. The APS system based on the basic trilateration is used to calculate the initial localization. The micro-genetic algorithm (MGA) is adopted to improve the accuracy of calculated estimates. The application of APS and MDS-based algorithm is proposed in [23].

Shekofteh *et al.* propose the localization scheme TS&SA, in which two different optimization methods executed in cascade are used to estimate locations of network nodes. The scheme is described and evaluated in [24]. It operates in two phases. Tabu search (TS) is executed in the first phase to solve the optimization problem Eq. (1) and estimate initial locations of node. In the second phase the simulated annealing (SA) method is used to refine the location estimates of all non-uniquely localized nodes. Similarly to Kannan *et al.* method described in [14] the optimization problem with the cost function J_{F_K} Eq. (3)

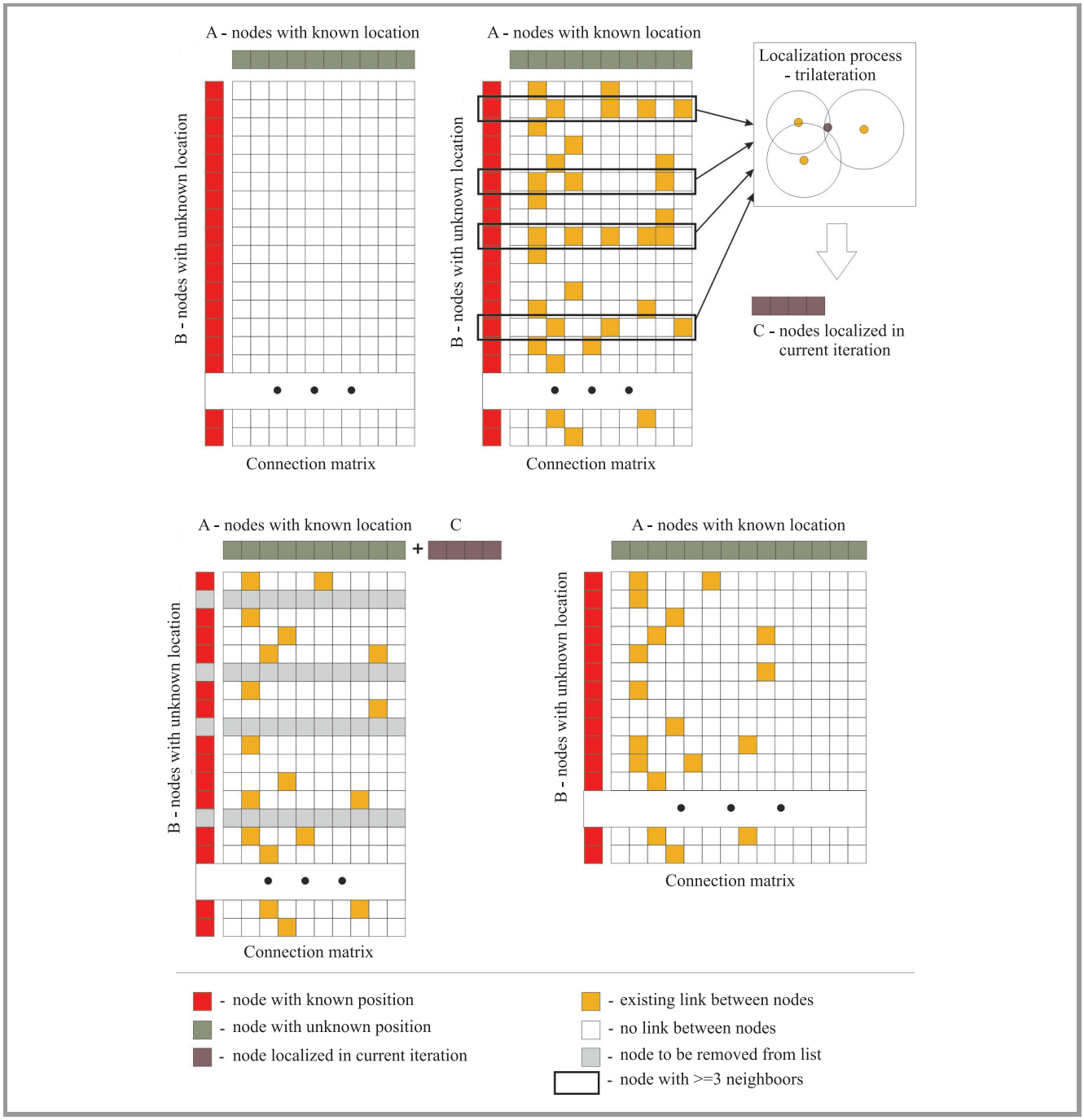


Fig. 2. Phase 1: calculating the initial solution using multitrilateration.

is solved to compensate localization errors. The method was evaluated through simulations. The authors claim that the TS&SA-based location system has better convergence characteristics compared to the SA-based system described in [13], but in the cited paper only the results of the TS&SA system simulation are demonstrated and discussed without comparison to other solutions, especially systems in which location errors driven by the flip ambiguity phenomenon are compensated.

We developed a hybrid scheme to location calculation that combines iterative multitrilateration along with nonconvex optimization and final correction. Two versions of this scheme are available: TSA: Trilateration & Simulated Annealing and TGA: Trilateration & Genetic Algorithm, and described in [25]. Both algorithms are range-based with RSSI technique used to distances estimation.

5.2. TSA and TGA Methods

TSA and TGA methods operate in two phases. In the beginning of the *first phase* all nodes in the network are divided into two sets: $A = \{a_1, \ldots, a_M\}$ containing anchor

nodes, and $B = \{x_1, \ldots, x_N\}$ of nodes with unknown location. Next, iterative multitrilateration is used to determine the relative positions of nodes from the set B based on the known locations of nodes from A, and the estimated distances between pairs of nodes. To determine the relative positions of each non-anchor on a 2D plane using trilateration at least three neighbors with known locations are needed. In every iteration each node from B with estimated position is moved to the auxiliary set C and finally, in next iteration of the algorithm changes its role to anchor and move to A. This phase stops when there are no more nodes from B that can be localized based on the available information about their neighbors. Figure 2 shows the performance of the phase 1.

In the *second phase* the optimization problem Eq. (1) is formulated, and the SA or GA algorithm is used to solve it. The goal of this phase is to increase the accuracy of the location estimation calculated in the first phase, and estimate the position of nodes that can not be calculated using iterative multitrilateration. The implementations of simulated annealing SA and genetic algorithm GA applied to TSA and TGA schemes are described below.

Simulated annealing method was implemented in TSA according to the algorithm described in [13]. It is a classical version of SA with one modification – the cooling process is slowed down. At each value of the coordinating parameter T (temperature), not one but $q \cdot N$ non-anchor nodes are randomly selected for modification (where N denotes the number of non-anchors in the network and q is a reasonably large number to make the system into thermal equilibrium). Coordinate estimations of chosen nodes are perturbed with a small displacement of the distance Δd in a random direction. The structure of the SA algorithm is presented in Fig. 3.

The algorithm consists of the following elements and operations:

Task configuration. The goal of this task is to localize N non-anchor nodes in a network. The initial location of all nodes is determined in phase 1 of the algorithm.

Moving operation. In each iteration of the algorithm a new solution is calculated. The node is randomly selected and is moved in random direction at distance Δd. The value of Δd depends on the control parameter T – the distance Δd is restricted by shrinking factor $\beta < 1$, $(\Delta d)_{new} = \beta \cdot (\Delta d)_{old}$.

Performance measure. The performance measure is defined in Eq. (1).

Cooling scheme. The simple cooling scheme is proposed: $T_{new} = \alpha \cdot T_{old}$.

A classical version of a **genetic algorithm** GA was applied to TGA. It applies the following operators:

Task configuration. The goal of this task is to localize N non-anchor nodes in a network. The abstract representa-

T = initial temperature
(Δd) = initial move distance
WHILE (final temperature not met)
{
FOR $i = 1$ to $(q \cdot N)$
{
pick a node to perturb
DO p times
{
generate a random perturbation to a node's estimated location
evaluate the change in cost function, $\Delta(CF)$
if $(\Delta(CF) \leq 0)$
//downhill move $\Rightarrow$ accept it
accept this perturbation and update the configuration system
else
//uphill move $\Rightarrow$ accept with probability
pick a random probability rp = uniform(0,1)
if $(rp \leq exp(-\Delta(CF)/T))$
accept this perturbation and update the configuration system
else
reject this perturbation and keep the old configuration system
}
}
$T_{new} = \alpha \cdot T_{old}$
$(\Delta d)_{new} = \beta \cdot (\Delta d)_{old}$

Fig. 3. Simulated annealing algorithm ([13]).

tions of candidate solutions called chromosomes are vectors of random variable – coordinates of all non-anchor nodes: $[x_1, x_2, \ldots, x_N]$, $x_i \in \Re^n$.

Initial population. The initial population consists of N chromosomes, the genes of which (initial coordinates of all nodes) were determined in the first phase of the algorithm.

Performance measure. Similarly to SA algorithm the performance measure (fitness function) is defined in (1).

Selection. The tournament selection of size $q = 2$ is used.

Crossover. Discrete recombination similar to elements exchanging applied to binary vectors is used with one modification – all coordinates of a given node are recombined simultaneously.

Mutation. The simple mutation operator is used. The components of chromosome are modified by adding a vector of generated $2N$ Gaussian random variables.

The method for correction of incorrect location estimates driven by the flip ambiguity phenomenon is provided in the second phase of TSA and TGA. Our submission is to use nested optimization to solve a problem with non-uniquely localizable nodes. The idea is to introduce the additional functionality – the correction operation to the optimization solver. The correction is triggered every iteration in the optimization process whenever the value of the performance function J_N defined in Eq. (1) is lower than a threshold θ. Trilateration is executed to relocate all nodes placed

in wrong neighborhoods by exploiting the nodes violating a smaller number of neighborhood constraints than the other randomly selected nodes. The threshold θ depends on the number of anchor nodes, network density and deployment, power of radio devices and expected noise measurement factor n_f. It is tuned according to the following formula:

$$\theta = \begin{cases} \mu \cdot n_f \cdot s^2, & \frac{N+M}{M} < \gamma \\ \lambda \cdot n_f \cdot s^2, & \frac{N+M}{M} \geq \gamma \end{cases} \tag{5}$$

where n_f is the noise measurement factor, μ, λ and γ experimentally tuned parameters. The variable s denotes an average number of neighbors of all nodes forming a network:

$$s = \frac{1}{N+M} \sum_{i=1}^{N+M} \sum_{j \in S_i} c_{ij}, \tag{6}$$

where

$$c_{ij} = \begin{cases} 1, & j \in S_i \\ 0, & j \notin S_i \end{cases}$$

where c_{ij} denotes the connectivity between i and j nodes, and S_i a set of neighbors of the node i. The correction algorithm is described in details in [25].

6. Tests and Evaluation

We validated selected location systems through simulation. All tests were performed on Intel Core2 Duo E6600 – 2.4 GHz, 2 GB RAM using our simulator, which employs Link Layer Model for MATLAB described in [26] for network model generation. The goal of all tests was to compare the accuracy and robustness of various approaches to the coordinate vector calculation. To evaluate the accuracy of tested location systems we used the mean error between the estimated and the true physical location of the non-anchor nodes in the network defined as follows:

$$LE = \frac{1}{N} \cdot \sum_{i=1}^{N} \frac{(||\hat{x}_i - x_i||)^2}{r_i^2} \cdot 100\%, \tag{7}$$

where N denotes the number of nodes in a network, which location is estimated, LE denotes a localization error, x_i the true position of the node i in the network, $\hat{x}_i$ estimated location of the node i (solution of the location system) and r_i the radio transmission range of the node i. The localization error LE is expressed as a percentage error. It is normalized with respect to the radio range to allow comparison of results obtained for different size and range networks.

All evaluated methods were range-based with inter-node distances calculated due to RSSI. We performed simulations for a network formed by 200 nodes (20 anchors and 180 non-anchors). Nodes distribution in a deployment area is presented in Fig. 4. Different network topologies were considered in our experiments. We analyzed the impact of network density and RSSI measurement errors on the accuracy of the location estimation. The density was expressed by a connectivity measure that was defined as an average number of neighbors of all nodes in a network. Two levels of inter-node distance estimation error involved by

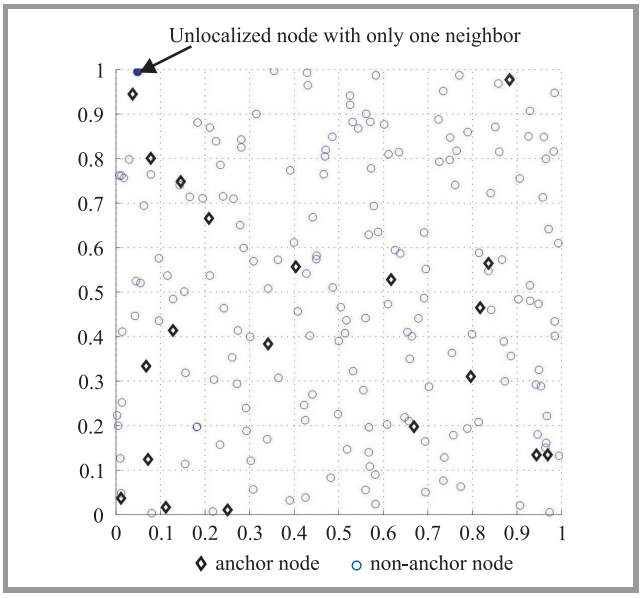


Fig. 4. Nodes deployment.

RSSI measurement errors were considered. To convert the measurements into the inter-node distances we applied the radio channel model. The algorithm is described in [10]. The detailed information about radio channel modeling can be found in [26]. Finally, we performed six series of experiments for various network density and measurement errors:

- connectivity measures: low (7–8 neighbors), medium (13–14 neighbors), high (20–21 neighbors),
- levels of distance estimation error: low (LDEE) with 0% – 0.2% error, high (HDEE) with 15% – 20% error.

The assumed distance estimation errors (in percent) for low, medium and high density networks are collected in Table 1.

Table 1
Distance estimation errors

Distance estimation error	Connectivity [%]		
	low	medium	high
LDEE	0.07	0.08	0.14
HDEE	15.83	17.67	18.24

It is obvious that lower network density usually involves increasing number of weekly connected nodes. The presence of unconnected or weekly connected nodes in a network has significant impact on localization error Eq. (7). We tested the influence of the weekly connected node on

the location estimation (a node marked with filled circle in Fig. 4; tests for low density network).
The localization problems formulated for networks defined in Table 1 were solved using three location systems: SAL, TSA and TGA. Tables 2 and 3 present the localization errors obtained respectively, for low and high measurement errors, and low, medium and high connectivity. From the experimental results we can observe that the best localization accuracy was obtained using the TSA algorithm both for low and high measurement errors. The difference in

Table 2
Localization errors for LDEE

Method	Connectivity		
	low	medium	high
SAL	6.98 (3.12*)	6.94 (5.91*)	4.09 (4.95*)
TSA	0.61 (0.60*)	0.11 (0.09*)	0.00 (0.00*)
TGA	18.85 (1.12*)	0.32 (0.33*)	0.03 (0.02*)
* The standard deviation of results obtained from five runs of each task.			

Table 3
Localization errors for HDEE

Method	Connectivity		
	low	medium	high
SAL	17.01 (3.09*)	9.60 (5.60*)	3.85 (3.85*)
TSA	5.46 (1.45*)	2.72 (0.51*)	2.82 (0.75*)
TGA	40.64 (7.70*)	23.80 (7.01*)	15.34 (5.01*)
* The standard deviation of results obtained from five runs of each task.			

localization quality is especially visible for low density networks. For higher density network the solutions calculated using the SAL system are satisfactory. In case of TGA the best results can be obtained for high density networks with low measurement error, and these results are almost as good as for TSA. It should be noted here that for low and medium connectivity it was impossible to calculate the accurate location (with 0% error) because of presence of the weekly connected node (see Fig. 4).
Simulation results confirm that TSA is an efficient and robust localization method. Using the TSA method we calculated the most accurate location estimates for all tested networks with the smallest standard deviation of the solutions. However, it should be underlined that efficiency and robustness of localization methods using heuristic techniques strongly depend on different control parameters of the algorithm. To design the general purpose algorithm to solve the localization problem the parameters should be tuned for various network size and topology. The TSA method was exhaustively tested on different networks in order to tune control parameters. It is very probable that both SAL and TGA methods can be tuned up to guarantee better accuracy, however this process is time consuming with no guarantee of success.

7. Summary and Conclusions

Sensor network localization continues to be an important research challenge. In this paper a short survey of the localization strategies and systems using global optimization methods is presented. We focus on application of heuristic techniques, such as simulated annealing, genetic and evolutionary computation. Referring to the literature and considering results of our research it seems that location systems using optimization methods, such as SA, GA, EA considerably outperforms systems based on linear or quadratic programming (SDP, SOCP, QP). In most tests described in literature heuristic algorithms gave an acceptable location accuracy in a acceptable computation time. Our experimental results presented in this paper demonstrate that the hybrid techniques are competing to the other solutions. Systems that combine geometrical and nonconvex optimization techniques extended with correction of temporary solutions provide significant robustness and improve an accuracy compared to a simple trilateration, convex and simple nonconvex optimization. Hence, from the perspective of location estimation accuracy the suggestion is to use centralized range-based hybrid location systems with measurement techniques according to the available hardware and additional correction of localization errors. In summary, we can say that sensor network localization continues to be an important research challenge. Despite, many methods and systems to estimate the location of nodes in WSN are proposed and described in literature, development of robust, accurate and scalable location system is still a challenging task.

Acknowledgment

This work was partially supported by National Science Centre grant NN514 672940.

References

[1] I. F. Akyildiz and M. C. Vuran, *Wireless Sensor Networks*. West Sussex, UK: Wiley, 2010.

[2] B. D. O. Anderson, G. Mao, and B. Fida, "Wireless sensor network localization techniques", *Comput. Netw.*, vol. 51, no. 10, pp. 2529–2553, 2007.

[3] G. Mao and B. Fidan, *Localization Algorithms and Strategies for Wireless Sensor Networks*. Information Science Reference, Hershey, USA, 2009.

[4] G. Sarigiannidis, *Localization For Ad Hoc Wireless Sensor Networks*. Paperback, LL, 2007.

[5] P. Barsocchi, S. Lenzi, S. Chessa, and G. Giuntaa, "Virtual calibration for rssi-based indoor localization with IEEE 802.15.4", in *Proc. IEEE Int. Conf. Commun. ICC*, Dresden, Germany, 2009, pp. 1–5.

[6] K. Benkic, M. Malajner, P. Planinsic, and Z. Cucej, "Using rssi value for distance estimation in wireless sensor networks based on zigbee", in *Proc. 15th In. Conf. Sys. Sig. Image Proces. IWSSIP 2008*, Bratislava, Slovakia, 2008, pp. 303–306.

[7] J. Beutel, *Handbook of Sensor Networks Compact Wireless and Wired Sensing Systems*. Boca Raton: CRC Press, 2005.

[8] H. Karl and A. Willig, *Protocols and Architectures for Wireless Sensor Networks*. West Sussex, UK: Wiley, 2005.

[9] P. Motter, R. S. Allgayer, I. Müller, and E. P. de Freitas, "Practical issues in wireless sensor network localization systems using received signal strength indication", in *Proc. Sensors Appl. Symp. SAS 2011*, San Antonio, USA, 2011, pp. 227–232.

[10] M. Marks and E. Niewiadomska-Szynkiewicz, "Self-adaptive localization using signal strength measurements", in *Proc. Fifth Int. Conf. Sensor Technol. Appl. SENSORCOMM 2011*, Nicea, Italy, 2011, pp. 73–78.

[11] D. Niculescu and B. Nath, A"d hoc positioning system (aps)", in *Global Telecommun. Conf. GLOBECOM 2001*, San Antonio, USA, 2001.

[12] P. Biswas and Y. Ye, "Semidefinite programming for ad hoc wireless sensor network localization", in *Proc. 3rd Int. Symp. Informa. Proces. Sensor Netw. IPSN'04*, New York, NY, USA, 2004, pp. 46–54.

[13] A. A. Kannan, G. Mao, and B. Vucetic, "Simulated annealing based localization in wireless sensor network", in *Proc. IEEE Conf Local Comput. Netw. 30th Anniversary LCN'05*, Washington, DC, USA, 2005, pp. 513–514.

[14] A. A. Kannan, G. Mao, and B. Vucetic, "Simulated annealing based wireless sensor network localization with flip ambiguity mitigation", in *63rd IEEE Veh. Technol. Conf.*, Montreal, Canada, 2006, pp. 1022–1026.

[15] J. Lee, W. Chung, and E. Kim, "A new range-free localization method using quadratic programming", *Comput. Commun.*, vol. 34, pp. 998–1010, 2010.

[16] P. Tseng, "Second-order cone programming relaxation of sensor network localization", *SIAM J. Optimiz.*, no. 18, pp. 156–185, 2007.

[17] Q. Zhang, J. Wang, C. Jin, J. Ye, Ch. Ma, and W. Zhang, "Genetic algorithm based wireless sensor network localization", in *Proc. 4th Int. Conf. Nat. Comput. ICNC 2008*, Jinan, China, 2008, pp. 608–613.

[18] M. H. Sayadnavard, A. T. Haghighat, and M. Abdechiri, "Wireless sensor network localization using imperialist competitive algorithm" in *Proc. 3rd IEEE Int. Conf. Comput. Sci. Inform. Technol.*, Chengdu, China, 2010.

[19] P. Chuang and C. Wu, "An effective pso-based node localization scheme for wireless sensor networks", in *Proc. 9th Int. Conf. Parallel Distribu. Comput. Appl. Technol.)*, Dunedin, Otago, New Zealand, 2008, pp. 187–194.

[20] J. Flathagen and R. Korsnes, "Localization in wireless sensor networks based on ad hoc routing and evolutionary computation", in *Proc. Military Commun. Conf. MILCOM 2010*, San Jose, CA, USA, 2010, pp. 1062–1067.

[21] M. Vecchio, R. Lopez-Valcarce, and F. Marcelloni, "A two-objective evolutionary approach based on topological constraints for node localization in wireless sensor networks (accepted)", *Applied Soft Comput.*, 2011.

[22] V. Tam, K. Y. Cheng, and K. S. Lui, "A descent-based evolutionar approach to enhance position in wireless sensor networks", in *Proc. 18th IEEE Int. Conf. Tools Artif. Intell. ICTAI'06*, Washington, DC, USA, 2006, pp. 568–574.

[23] A. A. Ahmed, H. Shi, and Y. Shang, "Sharp: a new approach to relative localization in wireless sensor networks", in *Proc. IEEE Int. Conf. Distrib. Comput. Sys. Worksh. ICDCSW'05*, Washington, DC, USA, 2005, vol. 9, pp. 892–898.

[24] S. K. Shekofteh, M. H. Yaghmaee, M. B. Khalkhali, and H. Deldari, "Localization in wireless sensor networks using tabu search and simulated annealing", in *Proc. o 2nd Inter. Conf. Comput. Automa. Engin. ICCAE 2010*, Singapore, 2010, pp. 752–757,

[25] E. Niewiadomska-Szynkiewicz and M. Marks, "Optimization schemes for wireless sensor network localization" *J. Appl. Mathem. Comput. Sci.*, vol. 19, no. 2, pp. 291–302, 2009.

[26] M. Zuniga and B. Krishnamachari, "Analyzing the transitional region in low power wireless links", in *Proc. 1st Int. Conf. Sensor Ad Hoc Commun. Netw. SECON 2004*, Santa Clara, CA, USA, 2004, pp. 517–526.

Ewa Niewiadomska-Szynkiewicz D.Sc. (2005), Ph.D. (1995), professor of control and information engineering at the Warsaw University of Technology, head of the Complex Systems Group. She is also the Director for Research of Research and Academic Computer Network (NASK). The author and co-author of three books and over 120 journal and conference papers. Her research interests focus on complex systems modeling and control, computer simulation, global optimization, parallel computation, computer networks and ad hoc networks. She was involved in a number of research projects including EU projects, coordinated the Groups activities, managed organization of a number of national-level and international conferences.
E-mail: ens@ia.pw.edu.pl
Institute of Control and Computation Engineering
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

E-mail: ewan@nask.pl
Research Academic Computer Network (NASK)
Wąwozowa st 18
02-796 Warsaw, Poland

Michał Marks received his M.Sc. in Computer Science from the Warsaw University of Technology, Poland, in 2007. Currently he is a Ph.D. student in the Institute of Control and Computation Engineering at the Warsaw University of Technology. Since 2007 with Research and Academic Computer Network (NASK). His research area focuses on wireless sensor networks, global optimization, distributed computation in CPU and GPU clusters, decision support and machine learning.
E-mail: M.Marks@ia.pw.edu.pl
Institute of Control and Computation Engineering
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

E-mail: M.Marks@nask.pl
Research Academic Computer Network (NASK)
Wąwozowa st 18
02-796 Warsaw, Poland

Mariusz Kamola received his Ph.D. in Computer Science from the Warsaw University of Technology in 2004. Currently he is associate professor at Institute of Control and Computation Engineering at the Warsaw University of Technology. Since 2002 with Research and Academic Computer Network (NASK). His research area focuses on economics of computer networks and large scale systems.

E-mail: mkamola@ia.pw.edu.pl
Institute of Control and Computation Engineering
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

E-mail: Mariusz.Kamola@nask.pl
Research Academic Computer Network (NASK)
Wąwozowa st 18
02-796 Warsaw, Poland

Paper

Incrementally Solving Nonlinear Regression Tasks Using IBHM Algorithm

Paweł Zawistowski and Jarosław Arabas

Institute of Electronic Systems, Warsaw University of Technology, Warsaw, Poland

Abstract—This paper considers the black-box approximation problem where the goal is to create a regression model using only empirical data without incorporating knowledge about the character of nonlinearity of the approximated function. This paper reports on ongoing work on a nonlinear regression methodology called IBHM which builds a model being a combination of weighted nonlinear components. The construction process is iterative and is based on correlation analysis. Due to its iterative nature, the methodology does not require a priori assumptions about the final model structure which greatly simplifies its usage. Correlation based learning becomes ineffective when the dynamics of the approximated function is too high. In this paper we introduce weighted correlation coefficients into the learning process. These coefficients work as a kind of a local filter and help overcome the problem. Proof of concept experiments are discussed to show how the method solves approximation tasks. A brief discussion about complexity is also conducted.

Keywords—black-box modeling, neural networks, nonlinear approximation, nonlinear regression, support vector regression, weighted correlation.

1. Introduction

In this paper the problem of solving nonlinear regression tasks is considered. The task consists in finding a function $\hat{f} : \mathbb{R}^n \to \mathbb{R}$ such that for the approximated function $f : \mathbb{R}^n \to \mathbb{R}$ the error between f and $\hat{f}$ function values is minimal. The f function is unknown, however sample input data $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_t\}$ and function values $Y = \{y_1, \ldots, y_t : y_i = f(\mathbf{x}_i)\}$ are given.

As regression tasks occur in many areas of research and industry, various methods of solving such tasks have been developed. These range from simple linear regression, through generalized regression to black-box modeling algorithms such as neural networks.

This paper reports on development of a black-box approximation method called IBHM, which is a shorthand for Incrementally Built Heterogenous Model, introduced in [1] and [2]. The method iteratively creates models similar in structure to MLP or RBF neural networks [3]. During this process, IBHM determines both parameters and the model structure, so no a priori assumptions are required, which makes the method very convenient to use. Although this is a relatively new approach, it has already achieved very good results in comparison to other methods [4]. This paper focuses on presenting the proper background and ideas connected with IBHM and also formulates the latest version of the algorithm.

There are various other black-box approximation methods, such as e.g., the already mentioned neural networks, that process models similar to IBHM. Ideas similar to the concept of iterative correlation based learning can also be found in other areas of research. In this paper we briefly refer to these approaches in the context of the presented method.

The paper is organized as follows. Section 2 introduces the correlation based learning used by IBHM and describes the ideas behind it. A detailed algorithm formulation along with computational complexity discussion and comparison to other methods is given in Section 3. Section 4 reports obtained experimental results. Finally a summary and discussion of future work is given in Section 5.

2. Correlation Based Learning

2.1. Genesis

The basic concepts behind IBHM originate from the well known method of linear regression. This technique creates models $\hat{f} : \mathbb{R}^n \to \mathbb{R}$ having the form

$$\hat{f}(\mathbf{x}) = \mathbf{w}^T \mathbf{x} \; + \; w_0, \tag{1}$$

where $\mathbf{w}, \mathbf{x} \in \mathbb{R}^n$ and $w_0 \in \mathbb{R}$. Linear regression fails to lead to good results if $f(\mathbf{x})$ is nonlinear.

A possible way to overcome such problems is to use a mapping function $\Phi : \mathbb{R}^n \to \mathbb{R}^m$ to transform the original variable $\mathbf{x}$ before applying linear regression. If the dependency between $\Phi(\mathbf{x})$ and $f(\mathbf{x})$ is linear then the linear regression can be applied to construct the model which effectively is nonlinear

$$\hat{f}(\mathbf{x}) = \mathbf{w}^T \Phi(\mathbf{x}) \; + \; w_0. \tag{2}$$

This approach would be a perfect solution to complicated modeling problems, but finding a proper Φ transformation is virtually impossible without detailed knowledge about the approximated function.

2.2. Single Nonlinear Component

Here we attempt to formulate a methodology that tries to guess the most appropriate form of the Φ mapping by selecting one of many possible candidates.
Let $h : \mathbb{R}^n \to \mathbb{R}$ be a scalarization function and $g : \mathbb{R} \to \mathbb{R}$ be a monotonous activation function. Now assume that $m = 1$, that is, a mapping $\Phi : \mathbb{R}^n \to \mathbb{R}$ is sufficient to build a linear model and has the form

$$\Phi(\mathbf{x}) = g\left(a \cdot h(\mathbf{x}, \mathbf{d}) + b\right) , \tag{3}$$

where $a, b \in \mathbb{R}$ are scalar parameters and $\mathbf{d} \in \mathbb{R}^n$ is a parameter vector.
The regression model has the form

$$\hat{f}(\mathbf{x}) = w_1 \cdot \Phi(\mathbf{x}) + w_0 = w_1 \cdot g\left(a \cdot h(\mathbf{x}, \mathbf{d}) + b\right) + w_0 . \tag{4}$$

Observe that the regression model parameters can be estimated by a two-step procedure:

- estimation of parameters a, b and $\mathbf{d}$,
- computing of weights w_0, w_1 via linear regression.

Prior to defining a method to estimate values of a, b and $\mathbf{d}$ a couple of observations has to be made. First notice that, as $\hat{f}(\mathbf{x})$ is to approximate $f(\mathbf{x})$, they must be linearly correlated. Inspection of Eq. (4) reveals that a high level of linear correlation is expected between $f(\mathbf{x})$ and $g(a \cdot h(\mathbf{x}, \mathbf{d}) + b)$. As the output values of the g function depend on parameters a and b, their proper values can be found by maximizing the correlation between f and g with respect to a and b.
A similar approach, which utilizes a rank correlation, can be used to find the $\mathbf{d}$ vector value. Observe that the output of the scalarization function h also has to be correlated with the output of f. In this case, however, linear correlation cannot be used, because the output values of h are transformed by the activation function which is nonlinear. In consequence, linear correlation may be improperly indicating dependencies between h and f, however, as the g function is monotonous, rank correlation will perform this task instead.Value of $\mathbf{d}$ can therefore be found by maximizing the rank correlation between f and $h(\mathbf{x}, \mathbf{d})$. The remaining weights w_0, w_1 can finally be estimated using linear regression.

2.3. Multiple Nonlinear Components

Up to this point the case of $m = 1$ has been considered. Now we consider a general case where $m > 1$ and assume that the Φ mapping has the following form

$$\Phi(\mathbf{x}) = \begin{bmatrix} g_1\left(a_1 \cdot h_1(\mathbf{x}, \mathbf{d}_1) + b_1\right) \\ \dots \\ g_m\left(a_m \cdot h_m(\mathbf{x}, \mathbf{d}_m) + b_m\right) \end{bmatrix} , \tag{5}$$

where $h_i : \mathbb{R}^n \to \mathbb{R}$ are scalarization functions, $g_i : \mathbb{R} \to \mathbb{R}$ are monotonous activation functions and $a_i, b_i, \mathbf{d}_i$ are parameters, $i = 1, \dots, m$.

Thus we consider the following regression model

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{m} w_i \cdot g_i(a_i \cdot h_i(\mathbf{x}, \mathbf{d}_i) + b_i) + w_0 . \tag{6}$$

All parameters in Eq. (6) can be estimated using the previously described procedure based on the correlation analysis iteratively, as presented in Algorithm 1. There the iteration loop starts in line 2. In iteration k first rank correlation between the current approximation residual ε_{k-1} and h output values is maximized to estimate $\mathbf{d}_k$ – line 4. Then a_k and b_k are similarly found via linear correlation maximization – line 5. Finally the residual ε_k is calculated – line 7 and linear regression is performed – line 8.

Algorithm 1:

Input: $X = \{\mathbf{x}_1, \dots, \mathbf{x}_m : \mathbf{x}_i \in R^n\}$ - training sample set
Result: $\hat{f}$ - approximation function

1 $\varepsilon_0(\mathbf{x}) \leftarrow f(\mathbf{x})$, $k \leftarrow 0$
2 **while** *the stop criterion is not satisfied* **do**
3 $\quad k \leftarrow k + 1$
4 $\quad \mathbf{d}_k \leftarrow \arg\max_j |r(h_k(X, \mathbf{d}), \varepsilon_{k-1}(X))|$
5 $\quad (a_k, b_k) \leftarrow \arg\max_{(a,b)} |r(g_k(a \cdot h_k(X, \mathbf{d}_k) + b), \varepsilon_{k-1}(X))|$
6 $\quad$ assume $\hat{f}_k(\mathbf{x}) = \sum_{i=1,\dots,k} w_i \cdot g_i(a_i \cdot h_i(\mathbf{x}, \mathbf{d}_i) + b_i)$
7 $\quad$ assume $\varepsilon_k(\mathbf{x}) = \hat{f}_k(\mathbf{x}) - f(\mathbf{x})$
8 $\quad [w_0, \dots, w_k] \leftarrow \arg\min_{[w_0,\dots,w_k]} \sum_{\mathbf{x} \in X} \left(\varepsilon_k(\mathbf{x})\right)^2$
9 **end**
10 $\hat{f}(\mathbf{x}) = \hat{f}_k(\mathbf{x})$

Finding parameters in the described fashion is in some sense a greedy approach. This is because during each iteration the method tries to fit the current scalarization and activation functions to the entire approximation residual even if this leads to a more complicated situation in the future. To get a better insight into this problem, consider

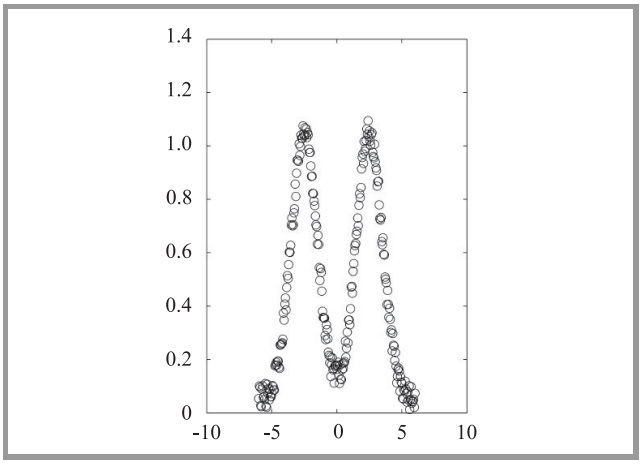


Fig. 1. An example approximated function.

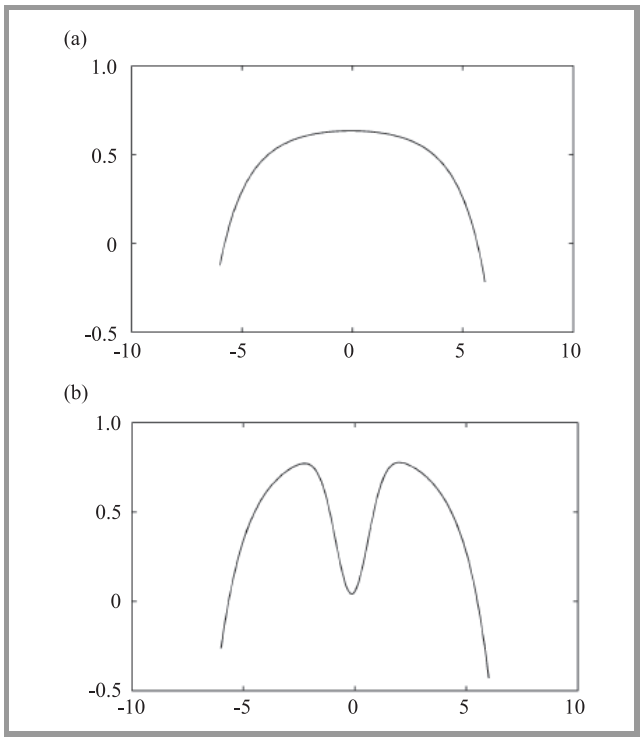


Fig. 2. Approximating function from Fig. 1 using Algorithm 1: (a) after the first iteration, (b) after the second iteration.

approximating the function given in Fig. 1. Application of Algorithm 1 would lead in this case to results similar to Fig. 2 which do not fully reflect the approximated function. This is because in the first iteration this approach tries to approximate the whole function instead on focusing on one of the two clearly distinct components. As a result it is unable to build a perfect model in two iterations.

2.4. Weighted Correlation

An improvement to the described situation can be made by using weighted correlation coefficients. Let us define the weighted linear correlation as

$$r_\omega(X,Y) = \frac{E_\omega(XY) - E_\omega(X)E_\omega(Y)}{\sqrt{(E_\omega(X^2) - E_\omega^2(X))\,(E_\omega(Y^2) - E_\omega^2(Y))}}\ , \quad (7)$$

and the weighted rank correlation as

$$\rho_\omega(X,Y) = r_\omega(rank(X), rank(Y))\ , \quad (8)$$

where

$$E_\omega(X) = \frac{\sum_{x\in X}\omega(x)x}{\sum_{x\in X}\omega(x)}\ . \quad (9)$$

Furthermore we consider a Gaussian weighting function of the form

$$\omega(x) = \frac{1}{\sqrt{2\pi}v}\, e^{\frac{-x^2}{2v^2}}\ . \quad (10)$$

When the algorithm uses the defined weighted coefficients instead of trying to decrease the approximation residual as much as possible in each iteration, the method focuses on identifying and approximating specific components of the approximated function. This means that IBHM tries to decompose the approximated function.

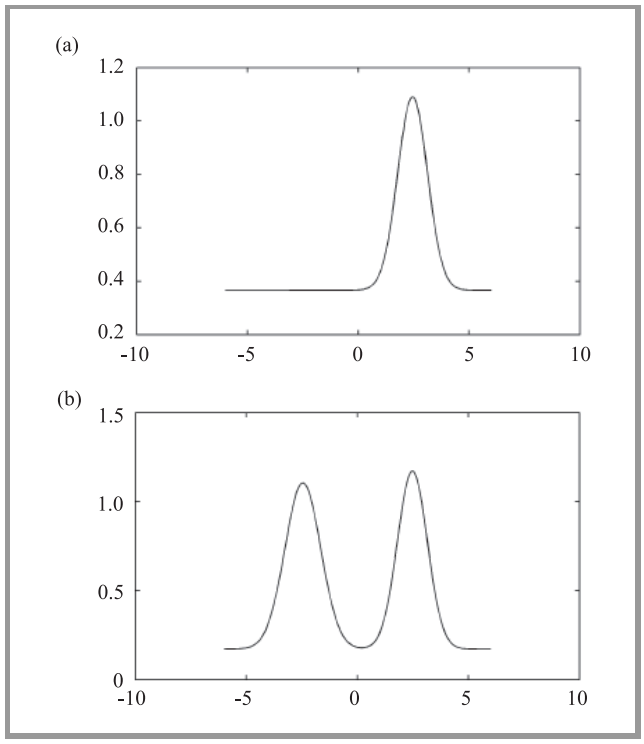


Fig. 3. Approximating function from Fig. 1 using Algorithm 1 with weighted correlation coefficients: (a) after the first iteration, (b) after the second iteration.

Coming back to the example, if we change lines 4 and 5 in Algorithm 1 to use the weighted correlation coefficients, the obtained approximation results are different as presented in Fig. 3. A close inspection of these results reveals that the weights work as a kind of a filter which puts focus on local features of the approximated function. Therefore it is possible to approximate a specific component of the approximated function, different in each iteration and get a more accurate model in the end.

3. IBHM Algorithm

3.1. Method Definition

Extending the conventions used in Subsection 2.1, let $H = \{\bar{h}_1, \ldots, \bar{h}_v : \bar{h}_i : \mathbb{R}^n \to \mathbb{R}\}$ denote the set of candidate scalarization functions and let $G = \{\bar{g}_1, \ldots, \bar{g}_u : \bar{g}_i : \mathbb{R} \to \mathbb{R}\}$ denote the set of candidate monotonous activation functions. The functions present in the final model are to be chosen from these two sets.

Algorithm 2 presents the proposed method, which builds the model given by Eq. (6), where $\forall_i\ g_i \in G \wedge h_i \in H$. The algorithm has three distinct parts which follow the procedure already described in Subsection 2.1. In the first part, for each candidate scalarization function, a proper parameter vector is found via the weighted rank correlation maxi-

mization – line 5. Then the best candidate is selected and set as the k-th scalarization function – lines 7–9. In the second part of the iteration, the parameters for candidate activation functions are found via the weighted linear correlation maximization – line 11. Then the best candidate is chosen as the k-th activation function – lines 13–15. In the third part of the iteration linear regression is used to estimate the model's weights – line 18.

Algorithm 2: IBHM

Input: $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m : \mathbf{x}_i \in R^n\}$ - training sample set
Result: $\hat{f}$ - approximation function

1 $\varepsilon_0(\mathbf{x}) \leftarrow f(\mathbf{x})$, $k \leftarrow 0$
2 **while** *the stop criterion is not satisfied* **do**
3 $\quad k \leftarrow k+1$
$\quad$ `/* Part 1 - finding scalarization function and parameter */`
4 $\quad$ **for** $i = 1, \ldots, |H|$ **do**
5 $\quad\quad \hat{\mathbf{d}}_i \leftarrow \arg\max_{\mathbf{d}} \left|\rho_\omega(\bar{h}_j(X, \mathbf{d}), \varepsilon_{k-1}(X))\right|$
6 $\quad$ **end**
7 $\quad i_k \leftarrow \arg\max_i \left|\rho_\omega(\bar{h}_i(X, \hat{\mathbf{d}}_i), \varepsilon_{k-1}(X))\right|$
8 $\quad \mathbf{d}_k \leftarrow \hat{\mathbf{d}}_{i_k}$
9 $\quad h_k \leftarrow \bar{h}_{i_k}$
$\quad$ `/* Part 2 - finding activation function and parameters */`
10 $\quad$ **for** $j = 1, \ldots, |G|$ **do**
11 $\quad\quad (\hat{a}_j, \hat{b}_j) \leftarrow \arg\max_{(a,b)} \left|r_\omega(\bar{g}_j(a \cdot h_k(X, \mathbf{d}_k) + b), \varepsilon_{k-1}(X))\right|$
12 $\quad$ **end**
13 $\quad j_k \leftarrow \arg\max_j \left|r_\omega(\bar{g}_j(\hat{a}_j \cdot h_k(X, \mathbf{d}_k) + \hat{b}_j), \varepsilon_{k-1}(X))\right|$
14 $\quad (a_k, b_k) \leftarrow (\hat{a}_{j_k}, \hat{b}_{j_k})$
15 $\quad g_k \leftarrow \bar{g}_{j_k}$
$\quad$ `/* Part 3 - extending the model */`
16 $\quad$ assume $\hat{f}_k(\mathbf{x}) = \sum_{i=1,\ldots,k} w_i \cdot g_i(a_i \cdot h_i(\mathbf{x}, \mathbf{d}_i) + b_i)$
17 $\quad$ assume $\varepsilon_k(\mathbf{x}) = \hat{f}_k(\mathbf{x}) - f(\mathbf{x})$
18 $\quad [w_0, \ldots, w_k] \leftarrow \arg\min_{[w_0,\ldots,w_k]} \sum_{\mathbf{x} \in X} (\varepsilon_k(\mathbf{x}))^2$
19 **end**
20 $\hat{f}(\mathbf{x}) = \hat{f}_k(\mathbf{x})$

Main loop of the algorithm is controlled by a stop criterion. This criterion should indicate if increase of the models complexity improves the overall results. A possible candidate method for that criterion is to stop when an increase in the Akaike Information Criterion (AIC) [5] is observed, where AIC is defined as

$$\mathrm{AIC}(X) = 2 \cdot p + |X| \cdot \ln\Big(\sum_{\mathbf{x} \in X} (\varepsilon_k(\mathbf{x}))^2\Big) \ , \qquad (11)$$

and p is the number of parameters estimated for the model[1].

[1] In case of m iterations of IBHM $p = m \cdot (n+3) + 1$.

Another possibility is to use a separate validation set to estimate the current model error and to stop when it increases.

3.2. Computational Complexity

Dominant operations which influence IBHM's complexity are connected with optimization tasks performed in each iteration. For that reason, a thorough complexity analysis of IBHM in the general case is impossible, as it would require precisely stating the complexities of solving unknown global optimization tasks. What follows however, is a rough discussion giving some insight into how costly is the algorithm in comparison with other methods.

In each IBHM iteration, a number of optimization tasks are solved. For each scalarization function from the H set, a global optimization of the $\mathbf{d} \in \mathbb{R}^n$ vector is solved. Estimation of parameters a, b is performed for each activation function from the set G. Each iteration is concluded with linear regression which is a quadratic problem of a size dependent on the iteration number.

When compared to methods which assume a fixed model structure, e.g., MLP neural networks, where only a single optimization task is solved, IBHM may seem to be overwhelmingly expensive. This is until dimensionality, a key factor connected with optimization tasks, is considered. Because of its iterative nature, the increase in the number of nonlinear components does not influence the number of parameters that undergo global optimization in each IBHM iteration. This means that in the optimization tasks solved by IBHM are simpler than in case of MLP.

Consider an example in which an approximated function $f : \mathbb{R}^n \to \mathbb{R}$ is is given. The approximation function is assumed to have the following structure

$$\hat{f}(\mathbf{x}) \ = \ \sum_{i=1}^{m} w_i \cdot tanh(a_i \cdot \mathbf{d}^T \mathbf{x} + b_i) \ + \ w_0 \ . \qquad (12)$$

This model corresponds to a MLP neural network with a single hidden layer of m-neurons and a hyperbolic tangent activation function. For MLP it is sufficient to assume that $a_i = 1$, therefore estimation of the parameters requires solving a single $m \cdot (n+2) + 1$ dimensional global optimization task.

The same model can also be constructed within m iterations of IBHM with $H = \{\mathbf{d}^t \mathbf{x}\}$ and $G = \{tanh(x)\}$. In this case we have to solve m global optimization tasks in $\mathbb{R}^n$ and $\mathbb{R}^2$ and m quadratic optimization tasks.

Table 1
Time units required to prepare the model from Eq. (12) (for $m = 10,\ n = 5$) in case of IBHM and MLP using optimization methods of various expected costs.

Alg. exp. cost	IBHM	MLP
$\Theta(N)$	70	71
$\Theta(N^2)$	2 900	5 041
$\Theta(N^3)$	133 000	357 911

If we have a global optimization method with the expected cost $\Theta(N^\alpha)$ where N is the optimization task dimensionality, we may estimate the computational effort required to prepare an MLP network as requiring $\Theta(mn+2m+1)^\alpha$ units. In the IBHM case, the cost is $\Theta(mn^\alpha+3m^\alpha)$ units plus the negligible cost of solving m quadratic optimization tasks. For various values of α the computational effort required by IBHM turns out to be far smaller then in case of MLP, as shown in Table 1.

3.3. Similar Methods

Neural networks. Multiple Layer Perceptron (MLP) or Radial Basis Function (RBF) type neural networks [3] can be used to create models very similar to those created by IBHM. The main difference is that IBHM does not require a priori assumptions about the final model structure, while constructing MLP or RBF networks requires setting the number of neurons to be used. Also the correlation based learning used by IBHM is a completely different from the techniques utilized by neural networks.
One important fact to notice is that IBHM may be utilized alongside neural networks as a preprocessing step estimating the required number of networks. This has been shown [4] to lead to good results in case of MLP models.

SVR. Support Vector machines for Regression (SVR, [6]) construct models with a similar structure to IBHM models. Another similarity is that, this method determines the model structure using the training data. The learning algorithm utilized determines most of the parameters directly using training data points, which are called support vectors. This is a different approach from the one IBHM uses.
The main drawback of SVR, not shared by IBHM, is that it tends to create very large models, which may lead to generalization problems.

GMDH. The Group Method of Data Handling [7] is a heuristic method which creates approximators using high order polynomials. The method works iteratively and puts focus on pairs of input variables in each iteration. Combination of these pairs form polynomials of higher orders which are added as new variables. The best from the new variables is treated as the current model. If the stop criterion is satisfied the algorithm terminates, otherwise the next iteration works on pairs of the variables introduced in the previous iteration and so on.
The iterative process uses a validation stop criterion, so the algorithm terminates when an error increase on the validation set is observer.
The GMDH learning process shares some similarities with the approach utilized by IBHM, however this method does not use correlation analysis during parameter estimation. Furthermore the estimation is done using a one step process, while IBHM splits this into three steps. Also GMDH works only on polynomials, while IBHM works on many different scalarization and activation functions.

CLEAN algorithms for signal processing. The incremental model building realized by IBHM can be viewed from a different perspective. In each iteration the algorithm tries to identify parts of the approximated function – its components. When such a component is identified, it is subtracted from the remaining residual, so that further components can be identified in future iterations. The idea to try and identify certain components present in a complex pattern, remove them and find other, previously not visible, components can be found in other areas of research. CLEAN radar algorithms ([8], [9]) are one such area. These algorithms address the problem of identifying multiple targets by iteratively filtering them out. Although the problem domain and methods are different from those utilized by IBHM, the core idea is quite similar.

4. Experiments

The focus of this paper is on reporting progress made on IBHM development therefore the goal of the presented experimental results is to illustrate the way the method works. For this purpose some toy problems were prepared and solved using IBHM.

4.1. Approximated Functions

The experiments were conducted using the following three functions:

$$f_1(x) = e^{-\frac{(x-8)^2}{2}} + e^{-\frac{x^2}{2}} + e^{-\frac{(x+8)^2}{2}} + \varepsilon_{N(0,0.01)} \ , \qquad (13)$$

$$f_2(x) = \frac{1}{5}\sin(x) + \frac{1}{10}x + \varepsilon_{N(0,0.01)} \ , \qquad (14)$$

$$f_3(x) = 1 - \tanh(x+6) + \tanh(x-6) + 2\cdot e^{-\frac{x^2}{2}} + \varepsilon_{N(0,0.01)} \ , \qquad (15)$$

where $\varepsilon_{N(0,0.01)}$ is a normally distributed random variable with mean 0 and variance 0.01. These functions were chosen as they represent various types of nonlinearities occurring in approximation tasks. The random component reflects the unknown factors influencing the approximated functions, therefore it compensates for the lack of knowledge.
For each of these functions two data sets were prepared: a training set of 160 examples and a test set of 80 examples. Test sets were used only to calculate the results reported in Subsection 4.3.

4.2. Algorithm Setup

IBHM was set up with $G = \{\tanh(x), x, \log\mathrm{sig}(x)\}$ and $H = \{d\cdot x, (x-d)^2\}$, where

$$\log\mathrm{sig}(x) = \frac{1}{1+\exp(-x)} \ . \qquad (16)$$

Both global optimization tasks were solved using the CMA-ES optimizer [10], while the Nelder-Mead Simplex [11] was used to perform the linear regression task. A validation criterion was used to determine when to stop the algorithm.

This criterion divided the training set into two parts: $\frac{2}{3}$ of data used to estimate the parameters, while the remaining part was used to estimate the error. The algorithm was stopped when an error increase was observed.

4.3. Results

For each approximated function, the algorithm was ran 20 times using the training data sets. Then using the test sets for each created model the mean squared error defined as

$$MSE = \frac{1}{n} \sum_{i=1,\ldots,n} \left(f(\mathbf{x}_i) - \hat{f}(\mathbf{x}_i) \right)^2 \qquad (17)$$

was calculated. Table 2 contains the aggregated results of the experiments. In this table the second column contains the standard deviations σ of the random components $\varepsilon_{N(0,\sigma)}$ present in the training data. For each problem the mean values and standard deviations of MSE values and model sizes are reported. The model size is the number of nonlinear components – activation functions used.

Table 2
The aggregated results of the experiments

Problem	Mean test MSE	Mean model size
f_1	0.0317 ± 0.0021	3.1 ± 0.3
f_2	0.0135 ± 0.0045	4.0 ± 0.3
f_3	0.0265 ± 0.0121	4.7 ± 0.8

When compared with the variance of the random components present in the training data the obtained error levels suggest that IBHM was able to capture the general approximated function structure without overfitting to the noise. This can be also noticed by inspection of the best models for each approximated function which are shown in Figs. 4, 5 and 6. The structures of these models are given in Appendix A.

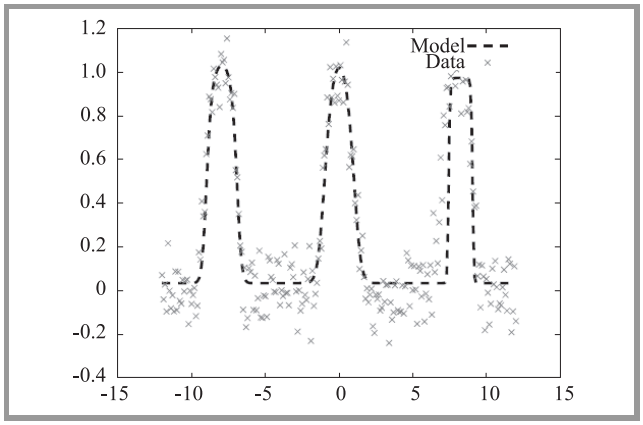


Fig. 4. The best ($MSE = 0.0320$) model created for $f_1(x)$ given in Eq. (19).

These results suggest that the noise present in the data does not degrade the correlation based learning method utilized by IBHM. Furthermore, it may be even argued that in some situations the presence of random components may be beneficial. This is because such components push the learning method away from focusing on small, insignificant parts of the approximated function and thus from overparametrization.

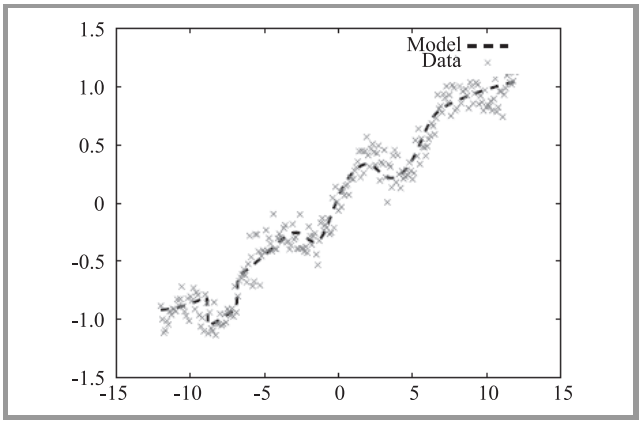


Fig. 5. The best ($MSE = 0.0109$) model created for $f_2(x)$ given in Eq. (20).

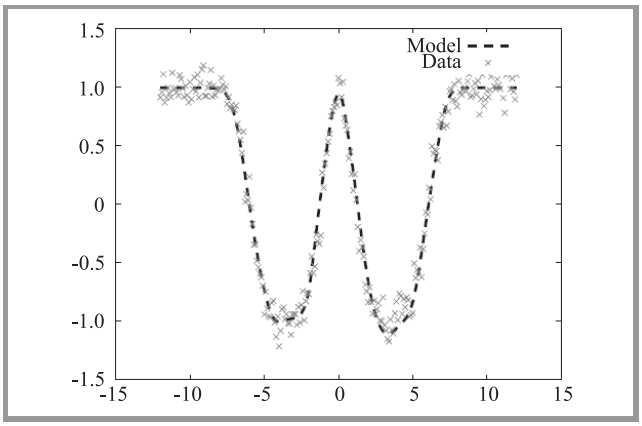


Fig. 6. The best ($MSE = 0.0165$) model created for $f_3(x)$ given in Eq. (21).

Another important aspect of the models built using IBHM was shown in [4]. This recent paper compared IBHM with MLP and SVR methods in the time series forecasting domain using 111 benchmark data sets from NN3 forecasting competition [12]. The comparison was based on results from experiments which put focus on evaluating the three different learning algorithms utilized by IBHM, MLP and SVR. The experimental procedure consisted in creating nonlinear autocorrelation models which defined the forecasts as

$$\hat{x}(t) = x(t-\tau_1) + \hat{f}\left(x(t-\tau_1), \ldots, x(t-\tau_{18})\right), \qquad (18)$$

where $\tau_1, \ldots, \tau_n$ were lags and $\hat{f}$ were nonlinear functions approximated using the three compared methods. The structure of the models where estimated using Akaike's Information Criterion. The built models' errors were estimated on test data sets and the whole training and

testing cycle was performed 25 times for each time series. The gathered data was used to estimate average MSE values and to perform pairwise comparison of models. In the comparison it was assumed that model A is better than B only when the median MSE for A was smaller than for B and the difference was statistically significant. Rank of a model for a time series was equal to one plus the number of models whose results were better for that time series. The results obtained are presented in Table 3.

Table 3
Summary of results aggregated over 111 time series from NN3 benchmark

Method	MSE mean	Model size*	Rank
IBHM	0.091 ± 0.212	1.09	1.76
MLP	0.093 ± 0.219	1.11	2.07
SVR	0.093 ± 0.209	4.80	2.46

* Number of nonlinear components averaged over all the models built by the given method.

These results show that IBHM performs well in comparison to MLP and SVR methods and that the models it creates tend to be rather small. This is an important virtue, as constructing large models may easily lead to overparameterization and generalization problems.

5. Summary and Future Work

IBHM is a promising approximation algorithm and the experimental results prove that the concepts behind it work. The method can be used in various fields ranging from time series forecasting to modeling complex physical processes. The models it constructs are similar in structure to those created using other well known and respected methods as MLP or SVR. Due to its iterative nature and decompositional properties, it does not require a priori assumptions about the final model structure, which makes it a convenient tool.

The future work on IBHM will focus on a couple of aspects. Theoretical analysis of the correlation based learning is to be conducted to provide a strong background for the algorithm. Furthermore, questions regarding optimal stop criteria and weighting functions need to be answered. There are also possibilities to extend the algorithm and to enable it to build more complex models with a structure similar to cascade correlation neural networks [13].

Another important aspect of further development is preparing an efficient implementation of the algorithm. The major amount of computation in IBHM is devoted to solving multiple optimization tasks. Fortunately these can be parallelized which opens up the possibility of utilizing General Purpose Graphical Processing Units (GPGPUs) to create a highly efficient implementation.

Appendix A

Sample Models

$$\begin{aligned}\hat{f}_1(x) \;=\; & 0.47\tanh\left(-6.98\cdot(x-8.25)^2+4.57\right)\\ & +1.06\,\mathrm{logsig}\left(-2.70\cdot(x+7.95)^2+2.58\right)\\ & -1.33\,\mathrm{logsig}\left(1.62\cdot(x-0.02)^2-1.01\right)+1.85\end{aligned} \tag{19}$$

$$\begin{aligned}\hat{f}_2(x) \;=\; & 43.06\,\mathrm{logsig}\left(0.01\cdot(x+12.42)^2+2.98\right)\\ & +0.54\,\mathrm{logsig}\left(0.64\cdot(x-4.15)^2-0.68\right)\\ & -0.27\,\mathrm{logsig}\left(-389.18\cdot(x+7.85)^2+390.57\right)\\ & -51.14\tanh\left(-0.34\cdot(x+1.30)^2-2.88\right)-93.57\end{aligned} \tag{20}$$

$$\begin{aligned}\hat{f}_3(x) \;=\; & 2.93\,\mathrm{logsig}\left(0.40\cdot(x-3.30)^2-0.93\right)\\ & -3.03\,\mathrm{logsig}\left(-0.38\cdot(x+2.70)^2+0.53\right)\\ & +0.46\,\mathrm{logsig}\left(-2.25\cdot(x+0.67)^2+2.11\right)\\ & -1.35\,\mathrm{logsig}\left(-1.02\cdot(x-5.74)^2+0.12\right)\\ & +1.90\tanh\left(0.31\cdot(x+5.25)^2+0.40\right)-3.84\end{aligned} \tag{21}$$

References

[1] J. Arabas and A. Dydyński, "An algorithm of incremental construction of nonlinear parametric approximators", in *Evolutionary Computation and Global Optimization 2006*, J. Arabas, Ed. Warsaw: WUT Press, Poland 2006, pp. 31–38.

[2] J. Arabas and A. Dydyński, "Nonlinear time-series modeling and prediction using correlation analysis", in Proc. Appl. Math. Mech. PAMM 2007, vol. 7, pp. 2030013–2030014, 2007.

[3] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Prentice Hall, 1999.

[4] P. Zawistowski and J. Arabas, "Benchmarking IBHM method using NN3 competition dataset", in *Proc. Hybrid Artif. Intel. Syst. Conf. HAIS 2011*, Wrocław, Poland, 2011. LNCS, Springer, vol. 6678, pp. 263–270, 2011.

[5] H. Akaike, "A new look at the statistical model identification", *IEEE Trans. Autom. Contr.*, vol. 19, no.6, pp 716–723, 1974.

[6] B. Schölkopf and A. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2002.

[7] J. Farlow Stanley, "The GMDH algorithm of Ivakhnenko", *The American Statistician*, vol. 35, no. 4, pp. 210–215, 1981.

[8] D. Hai, "Effective CLEAN algorithms for performance-enhanced detection of binary coding radar signals", *IEEE Trans. Signal Process.*, vol. 50, no. 1, pp. 72–78, 2004.

[9] T. L. Foreman, "Reinterpreting the CLEAN algorithm as an optimum detector", in Proc. IEEE Radar Conf., Verona, NY, USA, 2006, pp. 24–27.

[10] A. Auger and N. Hansen, "A restart CMA evolution strategy with increasing population size", *IEEE Congr. Evol. Comput.*, pp. 1769–1776. 2005.

[11] J. A. Nelder and R. Mead, "A simplex method for function minimization", *Comput. J.*, vol. 7, pp. 308–313, 1965.

[12] "Artificial Neural Network and Computational Intelligence Forecasting Competition" [Online]. Available: http://www.neural-forecasting-competition.com/NN3/index.htm

[13] S. Fahlman and C. Lebiere, "The cascade-correlation learning architecture", *Adv. Neural Inform. Process. Sys.*, no. 2, pp. 524–532, 1990.

Paweł Zawistowski was born in Poland in 1984. He received the M.Sc. degree from Warsaw University of Technology (WUT), Poland, in 2008. Since 2008 he has been a Ph.D. student at the Faculty of Electronics and Computer Engineering, WUT. His scientific interests include neural modeling, black-box learning methods and meta-heuristics. He has taken part in various projects in which he constructed black-box models of industrial systems and natural phenomena.

E-mail: p.zawistowski.2@elka.pw.edu.pl
Institute of Electronic Systems
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

Jarosław Arabas – for biography, see this issue, p. 10.

Paper

Benchmarking Procedures for Continuous Optimization Algorithms

Karol Opara[a] and Jarosław Arabas[b]

[a] Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland
[b] Institute of Electronic Systems, Warsaw University of Technology, Warsaw, Poland

Abstract—Reliable comparison of optimization algorithms requires the use of specialized benchmarking procedures. This paper highlights motivations which influence their structure, discusses evaluation criteria of algorithms, typical ways of presenting and interpreting results as well as related statistical procedures. Discussions are based on examples from CEC and BBOB benchmarks. Moreover, attention is drawn to these features of comparison procedures, which make them susceptible to manipulation. In particular, novel application of the weak axiom of revealed preferences to the field of benchmarking shows why it may be misleading to assess algorithms on basis of their ranks for each of test problems. Additionally, an idea is presented of developing massively parallel implementation of benchmarks. Not only would this provide faster computation but also open the door to improving reliability of benchmarking procedures and promoting research into parallel implementations of optimization algorithms.

Keywords—black-box optimization, comparing optimization algorithms, evaluation criteria, parallel computing.

1. Introduction

The question, which optimization algorithm performs "best", seems to be of both practical and scientific importance. However, the notion of the "best" algorithm is not well defined and, more importantly, there are little theoretical clues for its choice. For these reasons, in practice optimization algorithms are compared using specialized sets of test problems under appropriate evaluation criteria. Before going into details it may beneficial to briefly remind some of the crucial notions in the field of benchmarking.

Continuous optimization problem involve finding an argument x_{opt} minimizing a certain objective function $f: D \to \mathbb{R}$, where $D \in \mathbb{R}^n$.

$$x_{opt} = \arg\min_{x \in D} f(x). \tag{1}$$

Domain of the objective function D is called feasible set and it is often a hypercube $D = [l_1, u_1] \times [l_2, u_2] \times ... \times [l_n, u_n]$. The minimal function value is denoted by $f_{opt} = f(x_{opt})$. Optimization algorithm is a method of producing a series of points $x_1, x_2, ..., x_m \in D$. The best function value reached by an optimizer is denoted by $f_{best} = \min_{i \in \{1,2,...,m\}} f(x_i)$, while difference $f_{best} - f_{opt}$ is called optimization error.

The "best" optimization algorithm solves a problem accurately (effectively) and fast (efficiently). A combination of these two characteristics is referred to as performance, however formal definition of this notion is not clear. Moreover, all optimization methods yield nondeterministic results. This is due to two factors: random initialization of an algorithm, e.g., choice of start point (or start population) and stochastic nature of many of the state of the art optimization methods, most notably evolutionary and similar nature-inspired algorithms. Therefore, empirical estimation of any measure of performance requires many independent restarts of an algorithm.

Comparison between optimization methods is usually performed by means of running simulations for specially designed sets of optimization problems. Such process, called benchmarking, is not a trivial task and requires both specialized skills and knowledge. This paper provides an overview of benchmarking procedures discussing the major issues in this field: theoretical grounds of algorithm comparison, available benchmarks, evaluation criteria, interpretation of results and testing their statistical significance. The aim of this contribution is to promote the use of systematic procedures for comparison of algorithms and to highlight some of the most important aspects of benchmarking. Furthermore, two novel concepts are presented: criticism of rank-based comparison methods and an idea of parallel implementation of test problems, which may become a qualitative improvement to the benchmarking procedures.

1.1. Fair Comparison of Algorithms

Benchmarking emerges from the need of fair comparison of optimization algorithms. Choosing an algorithm which would perform "best" on a new function whose characteristic is unknown can be done through measuring performance over a wide range of test problems and aggregating the obtained results. A test problem consists of a test function accompanied by some additional criteria such as the feasible set, initialization area, stopping conditions etc. Benchmarking yields meaningful results only when competing algorithms are compared on the same test problems with the same performance criteria.

Comparing algorithms can be performed quite easily with the use of some readily available benchmarks. This approach has some major advantages:

- There is no need to develop testbeds and performance criteria, which saves work and protects from possible methodological mistakes.

- Using benchmarks does not require special effort, since their implementations are usually available in a few programming languages. It is enough to link one of them to an optimizer and assign processor time.
- Comparison with other algorithms, which were previously tested on a benchmark, is possible without repeating those experiments.
- Results may be postprocessed and their presentation can be standardized.

1.2. Benchmarking and Free Lunches

Although benchmarking is generally approved among scientists and practitioners, its sense is sometimes criticized on the basis of the *no free lunch theorem* introduced and formalized in papers [1], [2]. The theorem states that for any algorithm, any performance gain over one class of problems is offset by performance loss over another class. Therefore, no algorithm can be be considered consistently "best" for all possible problems. This statement highlights limitations to benchmarking, as there is no "best" general purpose optimizer. However, for a fixed class of problems, there are methods which yield better results than others.
Performance of optimization algorithms depends on exploiting problem regularities [3] such as symmetry or convexity of objective function. There is a large variety of "typical" optimization problems, which have random character with no structure [4]. On the other hand, typically solved problems usually reveal some regularities and therefore constitute only a narrow subset of all problems. For instance, the power of the set of all functions $f : \mathbb{R} \to \mathbb{R}$ equals $2^{\mathfrak{c}}$, while the power of continuous functions $f : \mathbb{R} \to \mathbb{R}$ is strictly lower and equals continuum $\mathfrak{c}$. The latter follows from the fact that a continuous function is uniquely defined by its restriction to rational numbers $f : \mathbb{Q} \to \mathbb{R}$ [5]. Thus, benchmarking can be used to find algorithms outperforming others on a narrow but quite important class of continuous functions. The resulting low performance for a wide class of discontinuous functions often bears little practical significance.

1.3. What do Benchmarks Measure?

Benchmarking is a way of evaluating an algorithm's ability to exploit regularities of certain class of problems. It is therefore important to remember that test problems and evaluation criteria were developed in order to address issues such as [6]:

- high cost of single function evaluation,
- high dimensionality of search space,
- linear and nonlinear constraints,
- high conditioning of a function,
- noisiness of a function,
- large number of local optima (thousands),
- linearly non-separable functions,
- global optimum located on a boundary of feasible set.

Some of these issues can be observed when looking at plots of two-dimensional variants of test problems from CEC'05 benchmark [7], Fig. 1. Most of the test functions in CEC and BBOB families of benchmarks are typical optimization problems known in the literature. They become, however,

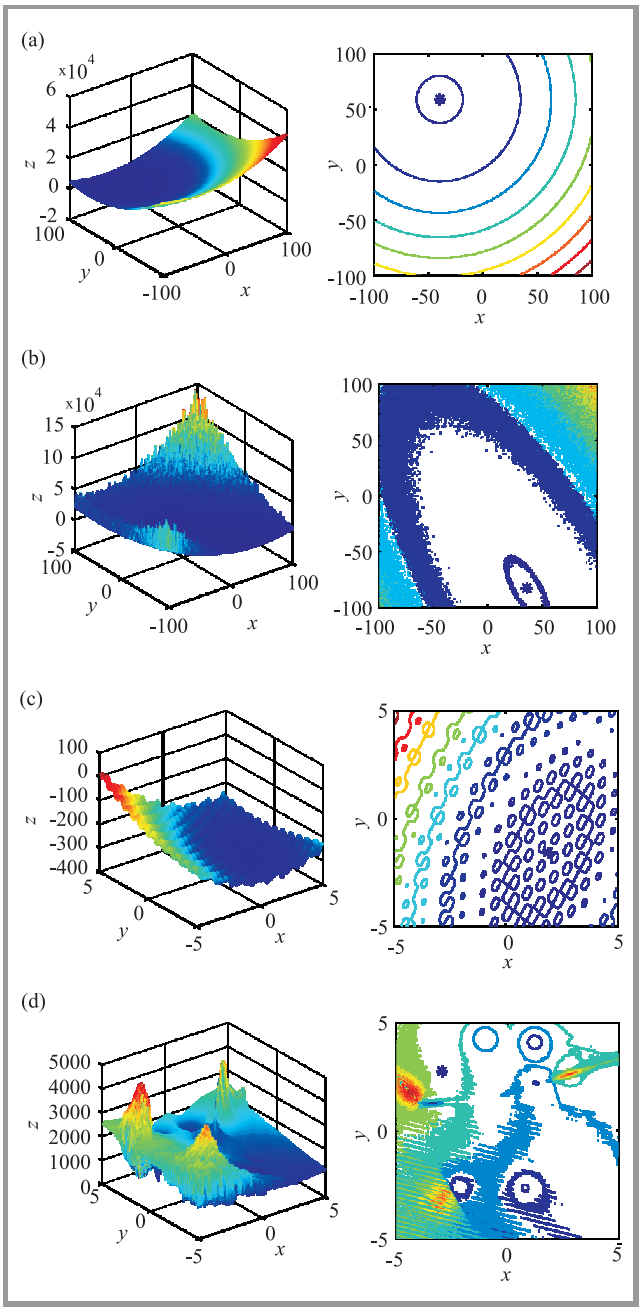


Fig. 1. Two-dimensional variants of CEC'05 benchmark problems: (a) sphere, (b) noisy ellipsoid, (c) Rastrigin, (d) hybrid. Each function is plotted within its feasible set, asterisks denote the location of global minimum.

subject to transformations such as shifting $f'(x) = f(x+c)$, adding a constant $f'(x) = f(x) + c$ and rotating. This is done in order to promote algorithms which are invariant to such changes of a coordinate system of the objective function. By this means, results obtained for a single test problem can be extended to whole class of problems.
Evaluation of algorithms is a two-step procedure. First, the algorithm's performance is measured for each test problem independently. Then, results obtained for each problem are aggregated to form a more general picture. Discussion of either of these steps is provided in Sections 2 and 3.
Benchmarking procedures for continuous domain optimizers are developed along CEC and GECCO conferences for special sessions devoted to comparing performance of optimization algorithms. Availability of these results facilitates comparing new optimization methods to the state of the art ones. CEC benchmarks cover a wide range of specialized optimization problems, Table 1. GECCO benchmarks are called BBOB (Benchmarking Black-Box Optimizers) and specialize in single-objective and noisy problems. BBOB is a carefully-developed platform providing motivated test problems, experimental setup and postprocessing of results [8].

Table 1
Scope of real-parameter benchmarks

Problem kind	Benchmarks
Single-objective	CEC'05, BBOB'09, BBOB'10
Constrained	CEC'06, CEC'10
Multi-objective	CEC'07, CEC'09
Large scale	CEC'08, CEC'10
Dynamic	CEC'09
Noisy	BBOB'09, BBOB'10
Real world	CEC'11

2. Measuring Performance for a Single Problem

There are two main ways for measuring the algorithm's performance for a single problem. The fixed cost approach consists of checking the final optimization error $f_{best} - f_{opt}$ after running the algorithm for a certain period of time. The fixed target approach consists in measuring time necessary to find a solution at an accuracy target Δf_t. In order to compare algorithms rather than their implementations and hardware used to run benchmark, computing time is expressed as the number of objective function evaluations (FEs), which is a standard approach in the literature on optimization.

2.1. Fixed Cost

Figure 2 shows convergence curves, i.e., the optimization error as a function of computing time, for four independent runs of an algorithm. Fixed cost approach can be illustrated with a vertical cut [8]. A set of error values

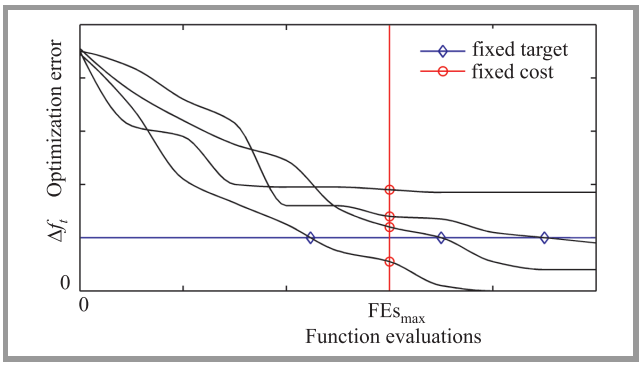


Fig. 2. Fixed cost and fixed target stopping criteria [8].

at a fixed cost can thus serve for comparisons. An algorithm A is better than algorithm B if it yields lower error values. Comparison is done on ordinal scale, which gives qualitative information (which algorithm is better?). However, no quantitative information (how much better it is?) is provided, as it is not clear how much more difficult is to reach a smaller error value [8].

2.2. Fixed target

Instead of fixing computing cost and comparing final optimization errors one can fix the desired error value Δf_t and compare the average runtime of algorithms required to reach it. Due to nondeterministic performance of algorithms, comparisons are reliable after aggregating the runtime values from multiple runs, for instance, by estimating their expected value. Again, computing time is measured as the number of function evaluations. The fixed target approach can be more precisely stated as comparing the estimates of the expected runtime needed for each algorithm to reach optimization error not greater than required, i.e., to satisfy condition $f_{best} - f_{opt} \leq \Delta f_t$. This criterion is illustrated in Fig. 2 with a horizontal line. Expected runtime values for different algorithms can be compared on the interval scale: it is possible to quantitatively state how much faster is one algorithm than another. This facilitates interpretation of results and it is the reason for choosing the fixed target approach for BBOB benchmarks [8].
There are, however, two problems with calculating the expected runtime. First, the required accuracy Δf_t must be somewhat arbitrarily chosen. This can be partially overcome by setting several accuracy targets. Second, expected runtime estimation is somewhat problematic. Some algorithm runs may fail to solve optimization problem at all. This may happen, for instance, due to premature convergence to a local optimum [9]. Consequently, the stopping criterion can not be solely dependent on the accuracy but it must also involve some "safety breaks", e.g., the maximum cost of simulation. Then, a single run of an algorithm is called successful if it reaches accuracy target within a given time limit. Without such limit, the number of successful runs in Fig. 2 would equal three. The presence of additional stopping criterion fixing maximal cost

reduces this number to one, as three runs stop due to exceeding time limit rather than reaching solution accurate enough.

Expected runtime $\hat{E}(RT(\Delta f_t))$ for a given accuracy target Δf_t is estimated as the sum of the expected number of function evaluations for one successful run $\hat{E}(N^s_{eval})$ and the maximum cost $\hat{E}(N^u_{eval})$ multiplied by odds against a successful run $(1-\hat{p})/\hat{p}$. Value $\hat{p}$ is an estimate for probability of solving a problem within a single run, in other words the fraction of successful algorithm runs [8].

$$\hat{E}\left(RT(\Delta f_t)\right) = \hat{E}(N^s_{eval}) + \frac{1-\hat{p}_s}{\hat{p}_s}\hat{E}(N^u_{eval}) \qquad (2)$$

Denoting the number of successful runs by $\#s$ and unsuccessful by $\#u$, one can note that $\hat{p} = \#s/(\#s+\#u)$. If a successful run is terminated right after meeting accuracy target then the above formula can be transformed to:

$$\hat{E}\left(RT(\Delta f_t)\right) = \frac{\#s \cdot \hat{E}(N^s_{eval}) + \#u \cdot \hat{E}(N^u_{eval})}{\#s} = \frac{N^t}{\#s}, \qquad (3)$$

where N^t denotes the total number of FEs within all algorithm runs. This estimator is however strongly dependent on the choice of maximal advisable cost in case of not meeting the target accuracy [8]. If the vertical line if Fig. 2 was shifted to the right, the number of successful runs $\#s$ would increase to two and then to three changing the value of the expected runtime estimate Eq. (3).

2.3. Statistical Analyses of Results

To check whether differences between algorithm performance are significant rather than observed purely by coincidence results should be subject to statistical testing. In papers comparing continuous optimization algorithms Student's t-test seems to be a common choice. This is a parametric test, as it relies on an assumption that sample (here final error values from several runs) is normally distributed. Alternatively, one can use non-parametric tests, which do not assume any distribution of a sample. An extensive study [10] conducted on CEC'05 results showed that conditions for safe use of parametric tests are usually not fulfilled, nevertheless results of parametric and non-parametric analysis are quite similar.

The use of non-parametric tests is encouraged in a multiple-problem analysis. In paper [10] the following non-parametric tests are suggested for analysis of optimization algorithms: Friedman, Iman-Davenport, Bonferroni-Dunn, Holm, Hochberg, and Wilcoxon. In case of performing pairwise comparisons of many algorithms the power of statistical tests decreases, as control of Family Wise Error Rate (FWER) is lost [10]. To compensate for that one should decrease the statistical significance α below typical value of 0.05 by using test variants designed for multiple comparisons.

Finally, it is worthwhile to examine results not only with statistics but also in a wider context. Superiority of final error rate of $10^{-40} \pm 10^{-41}$ over $10^{-15} \pm 10^{-16}$ has neither theoretical nor practical value in case of spherical function and double precision numbers. An old proverb says that difference is a difference only when it makes a difference.

3. Aggregation of Performance Measures

3.1. Ranking of Algorithms

Comparison of optimization algorithms A and B is a multiobjective task, as it is based on certain performance measure P_{F_i} for each of several test problems $F_1, F_2, \ldots, F_k$. The algorithms can be naturally partially ordered with Pareto improvement relation

$$A \preceq B \equiv (\forall i \in \{1,2,...,k\})\ P^A_{F_i} \geq P^B_{F_i}. \qquad (4)$$

Algorithm A is here considered better than algorithm B providing it yields better results for each test problem. This is, however, only partial ordering, as some pairs of algorithms are not comparable, e.g., when one of them performs better for some problems but worse for others. Hence, the question "which algorithm performs best?" is not always answered. For this reason, performance over the whole test set is often aggregated into a single number, which allows creating a linear ordering (ranking) among competing algorithms. Such aggregation method must be carefully chosen to ensure fairness of the comparison.

Issues such as designing decision rules to choose the best alternative out of a certain set have been widely investigated by economists and mathematicians in such fields as theory of the public sector. Some of these results can be applied to analyze ranking methods of optimization algorithms. The Weak Axiom of Revealed Preferences (WARP) [11], plays an important role among them. Its definition is equivalent to simultaneous satisfaction of the following conditions known as α rule and β rule:

- α if an algorithm performs best in a certain set of algorithms then it is also the best in its subset (of course if it is available in the subset),
- β if two algorithms are equally good, best algorithms, then in every superset either both are the best or none.

If WARP is not met, the results of comparison can be influenced by such means as shortlisting, pairwise comparisons or introducing other algorithms to increase relative advantages.

Multi-problem aggregation methods are based on results achieved by algorithms for sets of single problems. Some of them do not satisfy the β rule, which makes them susceptible to manipulation. This can be illustrated with the following example. Suppose that two algorithms, A and B, were tested on a set of four problems F_i, $i \in \{1,2,3,4\}$ and aggregated by comparing the mean rank achieved for all test problems. The ranks of final optimization errors

are shown in the upper part of Table 2. The mean rank for both algorithms equals 1.5, therefore one may conclude that A and B are equally good. Suppose now that another algorithm C had been added to this comparison. The performance of C is slightly worse than of A, therefore for each test problem C is ranked one notch lower than A, Table 2. The mean rank of algorithm A remains 1.5 while

Table 2
If mean rank is used as performance criterion, introduction of irrelevant alternative C changes preference between A and B

$\{A, B\}$		Test problem				Mean rank
		F_1	F_2	F_3	F_4	
Rank	1	A	A	B	B	$A = 1.5$
	2	B	B	A	A	$B = 1.5$

$\{A, B, C\}$		Test problem				Mean rank
		F_1	F_2	F_3	F_4	
Rank	1	A	A	B	B	$A = 1.5$
	2	C	C	A	A	$B = 2.0$
	3	B	B	C	C	$C = 2.5$

algorithm B now performs poorer, since its mean rank equals 2. This shows that introduction of an irrelevant alternative C changed the preference between algorithms A and B in favor of A. Therefore, competitions, for which mean (median, etc.) rank is used as multi-problem aggregation criterion are susceptible to manipulations. On the other hand, WARP is satisfied whenever the performance measure of an algorithm is independent from other, competing algorithms.

3.2. Empirical Runtime Distribution

Empirical runtime distribution is a way of aggregating and comparing performance of different algorithms measured for many runs over a set of test problems. It can be visualized by plotting the proportion of solved problems against the expected runtime of an algorithm. The general idea behind this plot can be illustrated with a following example. Consider benchmarking an algorithm on three test functions. Expected runtime values, over many independent runs, are given in Table 3 for fixed targets Δf_1 and Δf_2. Dotted graph in Fig. 3 presents runtime distribution for required accuracy Δf_1. Its shape represents an

Table 3
Estimated runtime values for three test problems and two accuracy targets

		Test function		
		F_1	F_2	F_3
Target	Δf_1	$3 \cdot 10^2$	$1 \cdot 10^3$	$4 \cdot 10^4$
	Δf_2	$2 \cdot 10^3$	$5 \cdot 10^3$	$7 \cdot 10^4$

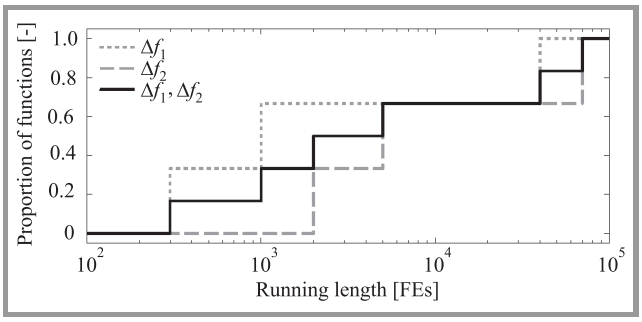


Fig. 3. Construction of empirical runtime distribution plots.

answer to question how many problems can one expect to solve for a given FEs budget? Below 300 FEs no problems can be solved, between 300 and 1000 only one (which is 33% of all problems), from 1000 to 40000 two (66%) and over 40000 three (100%). Dashed graph represents analogous data for more strict accuracy criterion Δf_2. Both lines can be aggregated by treating each of the six pairs $(F, \Delta f) \in \{F_1, F_2, F_3\} \times \{\Delta f_1, \Delta f_2\}$ as independent problems and using analogous manner to draw a new graph plotted in Fig. 3 with solid black line. Such aggregation among required accuracy targets is a way to overcome arbitrarily in choosing their levels Δf.

4. CEC and BBOB Benchmarks

Various versions of rank-based aggregation are used in practice. During CEC'06 competition ranking was based on three statistics of performance on test problems: feasible rate, success rate and expected runtime (also known as success performance) [12]. During CEC'07, ranks were assigned using two different multiobjective performance measures: R indicator and hypervolume difference to a reference set which resulted in two, quite similar rankings [13]. During CEC'08 and '09 each team was ranking others and the positions were averaged [14], [15]. In CEC'10 the Formula 1 point system was applied to 300 optimization problems and the algorithm with the highest score sum won [16].

In BBOB benchmarks [17] no ranking is created, as aggregation of results is based on empirical runtime distribution. Figure 4 presents the results of the BBOB'09 competition [17]. The horizontal axis shows runtime measured in function evaluations divided by dimension of search space n, which in this case equals 10. The vertical axis depicts the proportion of problems solved on all functions with accuracy targets $\Delta f_t \in \{10^{1.8}, 10^{1.6}, 10^{1.4}, \ldots, 10^{-8}\}$. Figure 4 was created in a slightly more complicated fashion than Fig. 3. Instead of estimating expected runtime values, for each pair $(F, \Delta f)$, 100 instances of simulated runtime were created by drawing algorithm runs at random with replacement until a successful run is found. Runtime instance is then computed as a sum of function evaluations from all runs drawn [17].

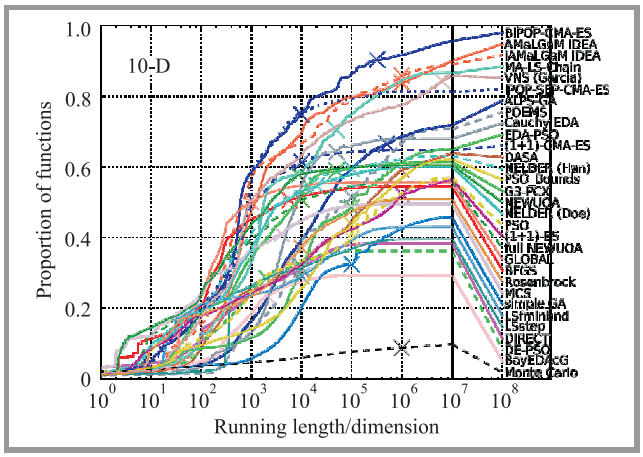


Fig. 4. Empirical runtime plots for ten-dimensional BBOB'09 benchmark for 31 algorithms [17].

Each algorithm was restarted many times and the cross indicates the maximum number of function evaluations. It is suggested [17] that a decline in steepness right after the cross (e.g., for IPOP-SEP-CMA-ES) may indicate that the maximum number of function evaluations should have been chosen larger, while a steep increase right after the cross (e.g., for simple GA) could be a sign that a restart should have been invoked earlier.

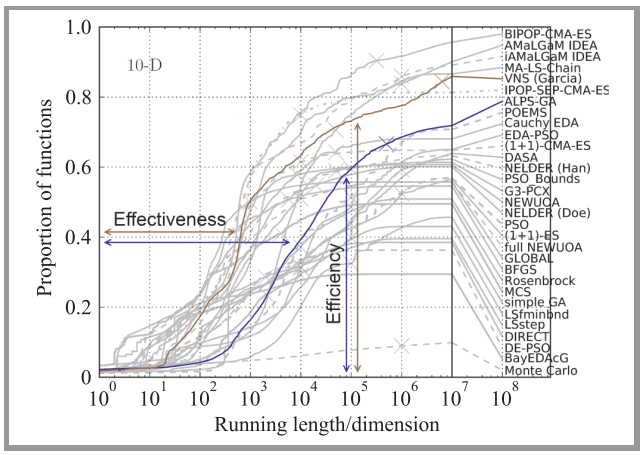


Fig. 5. Comparison of two algorithms VNS and ALPS-GA in terms of effectiveness (vertical distance between curves) and efficiency (horizontal distance).

Expected runtime plots provide significant interpretation possibilities, which are illustrated in Fig. 5 with an example of comparison of two algorithms, VNS and ALPS-GA. Within the budget of $10^5 \cdot n$ FEs, where n denotes search space dimension, ALPS-GA solves 60% of problems, while VNS over 70%, which is indicated by vertical arrows. Difference and ratio between these values show how much more efficient is one algorithm than another. On the other hand, horizontal arrows indicate how much computing time is required by each algorithm to solve a given ratio of test problems (here 40%). For VNS it is approximately $400 \cdot n$, while for ALPS-GA this time equals $10000 \cdot n$. One can therefore conclude that VNS is 25 times faster than ALPS-GA in solving 40% of problems. The area under a graph of an algorithm is, according to authors of the benchmark [17], arguably the most useful aggregated performance measure (averaged on a log scale). It might be therefore a good criterion for parameter tuning, for example, by means of metaoptimization.

5. Benchmarking and Parallel Computing

5.1. Parallel Implementation of Benchmarks

Both CEC and BBOB benchmarks are recognized and mature tools for measuring performance of optimization algorithms. Moreover, they can be easily parallelized. We believe that it would be beneficial to develop an implementation of these benchmarks exploiting massively parallel general purpose graphical processing units (GPUs). Availability of fast and scalable benchmark implementation would have significant consequences:

- GPU-based metaoptimizer could be developed for each benchmark. Fair comparison of algorithms requires choice of an "optimal" parameter set for each of them according to evaluation criteria. Currently, some optimization methods may not be tuned adequately, which decreases their performance. This might prove unjustified conclusions about superiority of some algorithms over others. To solve this problem, one can encourage all participating teams to perform metaoptimization, whose huge computing cost could be balanced by massive parallelism of GPUs.
- Additional evaluation criteria promoting research into the choice of the best parallelization methods of optimization algorithms could be introduced.
- The number of test functions could be increased. This would decrease statistical uncertainty, which is especially important in case of multiple comparison analysis, where significance α must be decreased to control FWER.
- Algorithms could be tested for larger FEs budgets.

5.2. Evaluation criteria for parallel algorithms

Evaluation criteria of optimization algorithms are based on the number of the objective function evaluations. This facilitates implementation independent comparisons and is practical, since evaluating the objective function is often the most time-consuming. On the other hand, many optimization algorithms can be easily parallelized by such means as simultaneous evaluation of the objective function in case of population-based methods. Therefore, the real execution

time can arguably be better described by the number of iterations rather than function evaluations required to solve a given problem.

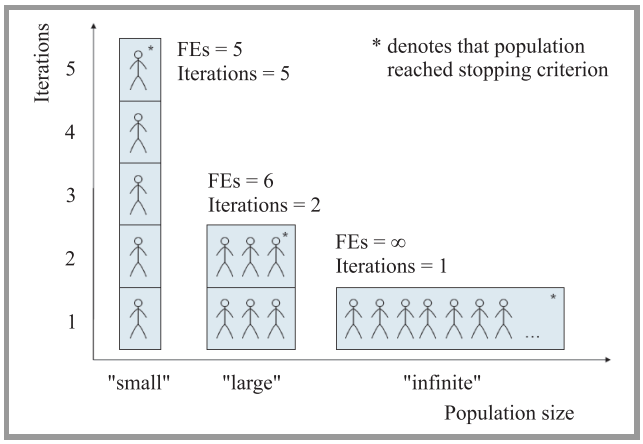


Fig. 6. Illustration of tradeoff between number of iterations and number of function evaluations in parallel computing.

Figure 6 illustrates a possible tradeoff between the number of function evaluations and the number of iterations. In this example, a population-based algorithm running with single individual requires 5 FEs and 5 iterations to solve a problem. A three-individual variant requires 6 FEs but only 2 iterations, hence it is slower for the sequential computation but faster for the parallel one. This feature would not be noticed with a traditional approach based on function evaluations. Direct adaptation of iteration counting might be, however, confusing, since for an infinite population any problem can be solved in the first iteration. For this reason, a number of iterations for some (arbitrarily chosen) maximal number of parallel processing units $\#proc$ seems to be a better criterion. It is worth noting that in case of full use of all processing units the number of iterations $\#iter$ is a product of number of function evaluations $\#FEs$ and processing units

$$\#iter = \#FEs \cdot \#proc\,. \tag{5}$$

Such criterion could additionally encourage researchers to look for the most appropriate parallelization models for their algorithms. This issue is illustrated in Fig. 7. In this simple example a test problem was solved using four processing units. For a four-individual population it took 4 iterations, for two two-individual populations it took 3 and 4 iterations and for four single-individual populations it took from 4 up to over 7 iterations. In case of many parallel instances one can stop computation after the problem is solved by any of them. Consequently, only these iterations (and function evaluations) should be taken for comparison, which are shaded in Fig. 7. In this example, problem was solved fastest when there were two populations each using two processing units. Many other parallelism models are possible and their choice is an interesting and algorithm-specific question.

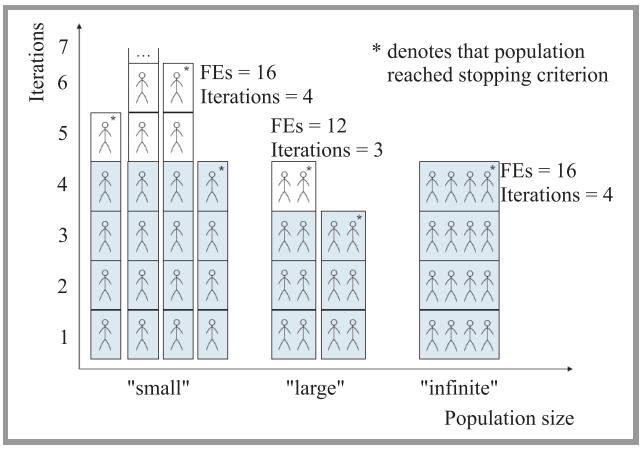


Fig. 7. Performance of three parallel optimization models for four processing units.

The proposed evaluation criteria are simplified, as they disregard some limitations of parallelism such as the Amdahl's law or memory transfer bottlenecks. For instance, selection operators in evolutionary algorithms require synchronization within population (or subpopulations in case of tournament selection). Nevertheless, popularization and further development of appropriate benchmarks seem to be an important issue for the whole community developing and using optimization algorithms.

6. Summary

This paper provides an outlook on procedures of measuring performance of optimization algorithms, emphasizing a need for standard and systematic approach in this field. Attention is paid to motivations and intuitive premises behind benchmarking as well as evaluation and comparison criteria for both single- and multiple-problem analyses. A brief summary of state of the art benchmarks and interpretation of their results is provided. Fairness of comparisons is discussed with respect to rank-based aggregation in multiple-problem analysis and testing of statistical significance of benchmarking results. Finally, a discussion of parallel implementation of test sets is provided. Not only can this make benchmarking more effective but also promote further research into parallelization schemes. Moreover, reliability of comparisons of algorithms can be improved by implementing a parallel and thus fast metaoptimizer.

References

[1] M. Köppen, D. H. Wolpert, and W. G. Macready, "Remarks on a recent paper on the "no free lunch" theorems", *IEEE Trans. Evol. Comput.*, vol. 5, no. 3, pp. 295–296, 2001.

[2] D. Wolpert and W. Macready, "No free lunch theorems for optimization", *IEEE Trans. Evolution. Comput.*, vol. 1, no. 1, pp. 67–82, 1997.

[3] T. Weise, M. Zapf, R. Chiongand, and A. Nebro, "Why is optimization difficult?", in *Nature-Inspired Algorithms for Optimisation*, R. Chiong, Ed., vol. 193 of *Studies in Computational Intelligence*. Springer, 2009, pp. 1–50.

[4] T. English, "Optimization is easy and learning is hard in the typical function", in *Proc. 2000 Congr. Evol. Comput. CEC 2000*, San Diego, CA, USA 2000, pp. 924–931.

[5] J. Kraszewski, *Wstęp do matematyki*. Warszawa: WNT, 2007 (in Polish).

[6] P. Suganthan, "Special session on real-parameter optimization at CEC 2005. Summary of feedbacks received from potential participants", 2005, access: May 2010 [Online]. Available: http://www.ntu.edu.sg/home/epnsugan/

[7] P. Suganthan *et al.*, "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization", Tech. Rep., Nanyang Technological University, Singapore and KanGAL Report Number 2005005, 2005, access June 2010 [Online]. Available: http://www.ntu.edu.sg/home/epnsugan/

[8] N. Hansen, A. Auger, S. Finck, and R. Ros, "Real-parameter black-box optimization benchmarking 2009: Experimental setup", Tech. Rep., INRIA, 2009, access: July 2010 [Online]. Available: http://coco.gforge.inria.fr/

[9] J. Arabas, *Wykłady z Algorytmów Ewolucyjnych*. Warszawa: WNT, 2004 (in Polish).

[10] S. García, D. Molina, M. Lozano, and F. Herrera, "A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the CEC 2005 special session on real parameter optimization", *J. Heuristics*, vol. 15, no. 6, pp. 617–644, 2009.

[11] A. K. Sen, "Choice functions and revealed preference", *Rev. Economic Studies*, vol. 38, no. 3, pp. 307–317, 1971.

[12] J. J. Liang and P. N. Suganthan, "Comparison of results on the 2006 CEC benchmark function set", 2006, access: Sept. 2011 [Online]. Available: http://www.ntu.edu.sg/home/epnsugan/

[13] P. N. Suganthan, "Performance assessment on multi-objective optimization algorithms", 2007, access: Oct. 2011 [Online]. Available: http://www.ntu.edu.sg/home/epnsugan/

[14] K. Tang, "Summary of results on CEC'08 competition on large scale global optimization", 2008, access: Sept. 2011 [Online]. Available: http://www.ntu.edu.sg/home/epnsugan/

[15] Q. Zhang and P. N. Suganthan. "Final report on CEC'09 MOEA Competition", 2009, access: Sept. 2011 [Online]. Available: http://www.ntu.edu.sg/home/epnsugan/

[16] K. Tang, T. Weise, Z. Yang, X. Li, and P. N. Suganthan, "Results of the competition on high-dimensional global optimization at WCCI 2010", 2010, access: Sept. 2011 [Online]. Available: http://www.ntu.edu.sg/home/epnsugan/

[17] N. Hansen, A. Auger, R. Ros, S. Finck, and P. Posik, "Comparing results of 31 algorithms from the black-box optimization benchmarking bbob-2009", in *Proc. Genetic Evol. Computat. Conf. GECCO 2010*, Portland, Oregon, USA, 2010. New York: ACM, 2010, pp. 1689–1696.

Karol Opara received M.Sc. degree in Computer Science from Warsaw University of Technology in 2010 and M.Sc. degree in Quantitative Methods in Economics and Information Systems from Warsaw School of Economics in 2011. Currently, he works as an assistant in the Systems Research Institute, Polish Academy of Sciences. His research interests focus on stochastic optimization algorithms, applied statistics and economic aspects of road management.
E-mail: karol.opara@ibspan.waw.pl
Systems Research Institute
Polish Academy of Sciences
Newelska st 6
01-447 Warszawa, Poland

Jarosław Arabas – for biography, see this issue, p. 10.

Paper

The "Second Derivative" of a Non-Differentiable Function and its Use in Interval Optimization Methods

Bartłomiej Jacek Kubica

Institute of Control and Computation Engineering, Warsaw University of Technology, Warsaw, Poland

Abstract—The paper presents an idea to use weak derivatives in interval global optimization. It allows using the Newton operator to narrow domains of non-differentiable functions. Preliminary computational experiments are also presented.

Keywords—Dirac delta, distributions, generalized derivative, interval computations, interval Newton method, non-differentiable optimization.

1. Introduction

Optimization algorithms for differentiable problems are well established and sophisticated. Also for non-smooth, but Lipschitz-continuous objective functions there are well-known methods. Replacing gradients with so-called subgradients allows to create analogs of several gradient-based methods for non-differentiable problems. For interval algorithms virtually no changes are needed [1].
In this paper it is proposed to extend this approach to using an analog of the second derivative.

2. Interval methods

Interval methods are a robust approach to global optimization.
Here, we shall recall some basic notions of intervals and their arithmetic. We follow a widely acknowledged standards (cf., e.g., [2], [3], [1]).
We define the (closed) interval $[\underline{x},\overline{x}]$ as a set $\{x \in \mathbb{R} \mid \underline{x} \leq x \leq \overline{x}\}$.
Following [4], we use boldface lowercase letters to denote interval variables, e.g., $\mathbf{x}$, $\mathbf{y}$, $\mathbf{z}$, and $\mathbb{IR}$ denotes the set of all real intervals.
We design arithmetic operations on intervals so that the following condition is fulfilled: if we have $\odot \in \{+,-,\cdot,/\}$, $a \in \mathbf{a}$, $b \in \mathbf{b}$, then $a \odot b \in \mathbf{a} \odot \mathbf{b}$. The actual formulae for arithmetic operations (see, e.g., [1], [2]) are as follows:

$$
\begin{aligned}
[\underline{a},\overline{a}] + [\underline{b},\overline{b}] &= [\underline{a}+\underline{b},\overline{a}+\overline{b}] \ , \\
[\underline{a},\overline{a}] - [\underline{b},\overline{b}] &= [\underline{a}-\overline{b},\overline{a}-\underline{b}] \ , \\
[\underline{a},\overline{a}] \cdot [\underline{b},\overline{b}] &= [\min(\underline{ab},\underline{a}\overline{b},\overline{a}\underline{b},\overline{ab}),\max(\underline{ab},\underline{a}\overline{b},\overline{a}\underline{b},\overline{ab})] \ , \\
[\underline{a},\overline{a}] \ / \ [\underline{b},\overline{b}] &= [\underline{a},\overline{a}] \cdot \left[1/\overline{b},1/\underline{b}\right] \ , \qquad 0 \notin [\underline{b},\overline{b}] \ .
\end{aligned}
$$

The so-called *extended interval arithmetic* allows division by an interval containing zero, not covered by the above formulae. The basic idea is that the result of such a division should be the set of all possible results of the division operation, executed on numbers from the argument intervals. We give here the formulae of Kahan–Novoa–Ratz arithmetic, following [1]:

$$
\mathbf{a}/\mathbf{b} = \begin{cases}
\mathbf{a} \cdot [1/\overline{b},1/\underline{b}] & \text{for } 0 \notin \mathbf{b} \\
[-\infty,+\infty] & \text{for } 0 \in \mathbf{a} \text{ and } 0 \in \mathbf{b} \\
[\overline{a}/\underline{b},+\infty] & \text{for } \overline{a} < 0 \text{ and } \underline{b} < \overline{b} = 0 \\
[-\infty,\overline{a}/\overline{b}] \cup [\overline{a}/\underline{b},+\infty] & \text{for } \overline{a} < 0 \text{ and } \underline{b} < 0 < \overline{b} \\
[-\infty,\overline{a}/\overline{b}] & \text{for } \overline{a} < 0 \text{ and } 0 = \underline{b} < \overline{b} \\
[-\infty,\underline{a}/\underline{b}] & \text{for } 0 < \underline{a} \text{ and } \underline{b} < \overline{b} = 0 \\
[-\infty,\underline{a}/\underline{b}] \cup [\underline{a}/\overline{b},+\infty] & \text{for } 0 < \underline{a} \text{ and } \underline{b} < 0 < \overline{b} \\
[\underline{a}/\overline{b},+\infty] & \text{for } \underline{a} < 0 \text{ and } 0 = \underline{b} < \overline{b} \\
\emptyset & \text{for } 0 \notin \mathbf{a} \text{ and } 0 = \mathbf{b}
\end{cases} .
$$

The definition of interval vector $\mathbf{x}$, a subset of $\mathbb{R}^n$ is straightforward: $\mathbb{R}^n \supset \mathbf{x} = \mathbf{x}_1 \times \cdots \times \mathbf{x}_n$. Traditionally interval vectors are called *boxes*.
Links between real and interval functions are set by the notion of an *inclusion function*, see, e.g., [3]; also called an *interval extension*, e.g., [1].

Definition 1: A function $\mathsf{f}\colon \mathbb{IR} \to \mathbb{IR}$ is an *inclusion function* of $f\colon \mathbb{R} \to \mathbb{R}$, if for every interval $\mathbf{x}$ within the domain of f the following condition is satisfied:

$$\{f(x) \mid x \in \mathbf{x}\} \subseteq \mathsf{f}(\mathbf{x}) \ .$$

The definition is analogous for functions $f\colon \mathbb{R}^n \to \mathbb{R}^m$.
When computing interval operations, we can round the lower bound downward and the upper bound upward. This will result in an interval that will be a bit overestimated, but will be *guaranteed to contain the true result of the real-number operation*.
Using these notions we can formulate the interval branch-and-bound (b&b) optimization algorithm, in the following way:

`Branch-and-bound-method` $(\mathbf{x}^{(0)}, \mathsf{f})$;
// $\mathbf{x}^{(0)}$ is the initial box
// $\mathsf{f}(\cdots)$ is the interval extension of the objective function
// L_{sol} is the list of solutions
$[\underline{y}^{(0)}, \overline{y}^{(0)}] = \mathsf{f}(\mathbf{x}^{(0)})$;
compute f_{min} = the upper bound on the global minimum
(e.g. objective value in a feasible point);
$L = \{(\mathbf{x}^{(0)}, \underline{y}^{(0)})\}$;

$L_{sol} = \emptyset$;
`while` ($L \neq \emptyset$) `do`
    x = the element of L with the lowest function value underestimation;
    compute the values of interval extensions of the constraint functions;
    `if` (**x** is infeasible) `then` discard **x**;
    update f_{min} if possible;
    perform other rejection/reduction tests on **x**;
    `if` (**x** is verified to contain a unique critical point or **x** is small and not infeasible) `then`
        add **x** to L_{sol};
    `else`
        bisect **x** to subboxes $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$;
        compute lower bounds $\underline{y}^{(1)}$ and $\underline{y}^{(2)}$ on the function value in the obtained boxes;
        delete **x** from L;
        `for` $i = 1, 2$ `do`
            put $(\mathbf{x}^{(i)}, \underline{y}^{(i)})$ on the list L preserving the increasing order of the lower bounds;
        `end for`
        delete from L boxes with $\underline{y}^{(i)} > f_{min}$;
    `end if`
`end while`
delete from L_{sol} the boxes with $\underline{y}^{(i)} > f_{min}$;
`return` L_{sol};
`end Branch-and-bound-method`

3. Using Weak Derivatives of a Function

What is less obvious is that for non-smooth problems we can also have an analog of the second derivative. All integrable functions – even non-differentiable ones – have so-called Sobolev generalized derivatives (also known as weak derivatives). These derivatives do not have to be functions, but may belong to a wider class of distributions (generalized functions) sometimes (see, e.g. [5], [6], [7]). Consider the following examples:

Example 1 – the absolute value function.

$$f(x) = |x| \ , \ x \in [-3,5] \ .$$

Its weak derivative can be expressed as the following multifunction:

$$\frac{Df}{Dx} = \begin{cases} -1 \ , & \text{for } x < 0 \ , \\ 1 \ , & \text{for } x > 0 \ , \\ [-1,1] \ , & \text{for } x = 0 \ . \end{cases}$$

More precisely, the weak derivative is a selection of the above multifunction.
And the second derivative is the following distribution:

$$\frac{D^2 f}{Dx^2} = 2\delta(x) \ .$$

We can approximate it using interval methods. The only thing we need is an interval extension of the Dirac delta. And such an extension can be developed quite simply:

$$\boldsymbol{\delta}(\mathbf{x}) = \begin{cases} [0,0], \text{ if } 0 \notin \mathbf{x} \ , \\ [0,+\infty] \text{ else.} \end{cases} \tag{1}$$

The classical Newton operator takes – in the univariate case – a well known form:

$$x^{new} = N(x) = x - \frac{f'(x)}{f''(x)} \ .$$

When the second derivative is the Dirac delta, the above operator does not seem very useful. Dirac delta returns either 0 (mostly) or infinity – making the division operation undefined. However, using formula (1) and the interval Newton operator (see, e.g., [1], [2]):

$$\begin{aligned} N(\mathbf{x}) &= \operatorname{mid}\mathbf{x} - \frac{\mathsf{f}'(\operatorname{mid}\mathbf{x})}{\mathsf{f}''(x)} \ , \\ \mathbf{x}^{new} &= N(\mathbf{x}) \cap \mathbf{x} \ , \end{aligned}$$

we can still obtain useful results. For the interval $\mathbf{x} = [-3,5]$ we get $\operatorname{mid}\mathbf{x} = 1$, $\mathsf{f}'(\operatorname{mid}\mathbf{x}) = 1$ and $\mathsf{f}''(\mathbf{x}) = [0,\infty]$, so – using the Kahan-Novoa-Ratz extended interval arithmetic [1] – we obtain:

$$N(\mathbf{x}) = 1 - \frac{1}{[0,\infty]} = 1 - [0,\infty] = [-\infty, 1] \ ,$$

which intersected with the original interval $[-3,5]$ gives $[-3,1]$; an interval reduced by half.
Please note that the function is not monotone in the considered interval, so no monotonicity test (using only subgradients) would allow us to narrow the domain without branching.
The above result has been obtained for the least promising case probably. Let us consider a "more friendly" objective.

Example 2.

$$f(x) = x^2 + |x| \ , \ x \in [-3,5] \ .$$

Its weak derivative can be expressed as:

$$\frac{Df}{Dx} = \begin{cases} 2x - 1 \ , \text{ for } x < 0 \ , \\ 2x + 1 \ , \text{ for } x > 0 \ , \\ [-1,1] \ , \text{ for } x = 0 \ . \end{cases}$$

The second derivative is:

$$\frac{D^2 f}{Dx^2} = 2 + 2\delta(x) \ .$$

Now for the interval $x = [-3,5]$, we get:

$$N(x) = 1 - \frac{3}{[2,\infty]} = 1 - [0,1.5] = [-0.5,1] \ .$$

So, both endpoints of the interval have been improved!
The remainder of the paper considers examples of problems that can be solved using the proposed methodology.

Apparently, use of the second weak derivative is superior to using generalized gradients only – at least for some problems.

4. Computational Experiments

Numerical experiments were performed on a computer with 16 cores, i.e., 8 Dual-Core AMD Opterons 8218 with 2.6 GHz clock. The machine ran under control of a Fedora 15 Linux operating system.The solver was implemented in C++, using C-XSC 2.5.1 library [8] for interval computations and automatic differentiation. To deal with non-smooth functions, e.g., min() and max(), the automatic differentiation code had to be modified (files `hess_ari.hpp` and `hess_ari.cpp`). The interval global optimization algorithm, on the other hand, did not have to be modified significantly (see e.g. [1]); only some maintenance changes were done.
The following optimization problems were considered.

$$\min_x f_1(x) = \sum_{i=1}^{n} |x_i| \ , \tag{2}$$
s.t.
$$x_i \in [-3, 5.5], \ i = 1, \ldots, n \ .$$

$$\min_x f_2(x) = \sum_{i=1}^{n} \left(x_i^2 + |x_i|\right) \ , \tag{3}$$
s.t.
$$x_i \in [-3, 5.5], \ i = 1, \ldots, n \ .$$

$$\min_x f_3(x) = \sum_{i=1}^{n} (-1)^{i+1} \cdot |x_i| \ , \tag{4}$$
s.t.
$$x_i \in [-3, 5.5], \ i = 1, \ldots, n \ .$$

$$\min_{x_1, x_2} f_4(x_1, x_2, x_3) = |x_1^4 - 1| + |x_2| - |x_3| \ , \tag{5}$$
s.t.
$$x_1, x_2, x_3 \in [-3, 5.5] \ .$$

$$\min_{x_1, x_2} f_5(x) = \max\{-(x-1)^2, -(x+1)^2\} \ , \tag{6}$$
s.t.
$$x \in [-0.5, 1.0] \ .$$

Tables 1–5 contain the results for these problems. Following fields are:

- var – the number of decision variables of the problem (for functions (2)–(6), where this number may be arbitrary),
- fun – the number of objectives evaluations required,
- grad – the number of objective' gradients evaluations required,
- Hesse – the number of objective' Hesse matrices evaluations required,
- bis – the number of boxes' bisections required,
- box – the number of resulting boxes, approximating the set of solutions.

Fields "fun", "grad", "Hesse" and "bis" are all some kind of measure of the algorithm performance. It seems the number of bisections (roughly corresponding to the number of iterations) is the best measure, but all of them should be taken into account.

Table 1
Results for problem Eq. (2)

	1st order				
var	fun	grad	Hesse	bis	box
1	51	27	25	24	1
2	99	53	49	48	1
4	195	113	97	96	1
8	387	225	193	192	1
16	771	481	385	384	1
32	1539	961	769	768	1
64	3075	2049	1537	1536	1
	2nd order				
var	fun	grad	Hesse	bis	box
1	29	26	14	13	1
2	39	35	19	18	1
4	53	46	26	25	1
8	73	59	36	35	1
16	107	79	53	52	1
32	173	114	86	85	1
64	301	179	150	149	1

Table 2
Results for problem Eq. (3)

	1st order				
var	fun	grad	Hesse	bis	box
1	51	27	25	24	1
2	99	53	49	48	1
4	195	113	97	96	1
8	387	225	193	192	1
16	771	481	385	384	1
32	1539	961	769	768	1
64	3075	2049	1537	1536	1
	2nd order				
var	fun	grad	Hesse	bis	box
1	7	24	23	2	1
2	11	26	25	4	1
4	19	30	29	8	1
8	35	38	37	16	1
16	67	54	53	32	1
32	131	86	85	64	1
64	259	150	149	128	1

Table 3
Results for problem Eq. (4)

	1st order				
var	fun	grad	Hesse	bis	box
1	51	27	25	24	1
2	53	28	26	25	1
4	103	55	51	50	1
8	203	117	101	100	1
16	403	233	201	200	1
32	803	497	401	400	1
64	1603	993	801	800	1
	2nd order				
var	fun	grad	Hesse	bis	box
1	29	26	14	13	1
2	37	21	15	14	1
4	43	37	21	20	1
8	61	50	30	29	1
16	89	67	44	43	1
32	139	95	69	68	1
64	237	146	118	117	1

Table 4
Results for problem Eq. (5)

	fun	grad	Hesse	bis	box
1st order	192	107	96	94	2
2nd order	88	83	44	42	2

Table 5
Results for problem Eq. (6)

	fun	grad	Hesse	bis	box
1st order	45	24	22	21	1
2nd order	25	23	12	11	1

5. Results

In all cases the proposed method performed much better than the one not using the second-order information (in terms of number of bisections, Hesse matrices evaluation, etc.). For two variables, using the 2nd weak derivatives seems to reduce the number of iterations by the factor of 2. For higher dimensions, it becomes much larger – for 64 variables it is even 10 times faster.

An interesting result was obtained for problem Eq. (3), in Table 2. The function is convex and the proposed version of the Newton operator allows – as shown in Example 2 – narrowing the interval on both sides. Thus, the number of bisections required is equal to the problem dimension multiplied by 2; it is sufficient to cut off the bounds and the interior can be narrowed instantly.

Yet more interesting results can be found in Table 4. Function is nonconvex, yet the performance of the b&b algorithm seems even better than for convex functions. Also, speedups gained by the use of 2nd weak derivatives seem as good as for convex problems. This – very desired – phenomenon should be investigated in subsequent papers, deeper.

6. Conclusions and future work

Results presented in this paper are only preliminary, yet very promising. For several simple functions the algorithm using generalized second order derivatives required far fewer iterations than the one using 1st order subderivatives only.

Some theoretical investigation of the convergence is required. It does not seem that using the Newton operator used with weak derivatives for non-differentiable functions results in quadratic order of convergence; yet it is more efficient than not using this tool, the monotonicity test only. The specific order of convergence is to be determined, yet.

Finally, the presented approach is an interesting example of the difference of features of interval and non-interval algorithms – the non-interval version of the Newton operator based on pseudo-functions seems completely inapplicable, as discussed in Section 3.

Acknowledgments

The paper has been done as a part of realization of the grant for statutory activity, financed by the Dean of Faculty of Electronics and Information Technology (WUT), titled “Interval methods for solving nonlinear problems”.

References

[1] R. B. Kearfott, *Rigorous Global Search: Continuous Problems*. Dordrecht: Kluwer, 1996.

[2] E. Hansen and W. Walster, *Global Optimization Using Interval Analysis*. New York: Marcel Dekker, 2004.

[3] L. Jaulin, M. Kieffer, O. Didrit and E. Walter, *Applied Interval Analysis*. London: Springer, 2001.

[4] R. B. Kearfott, M. T. Nakao, A. Neumaier, S. M. Rump, S. P. Shary and P. van Hentenryck, “Standardized notation in interval analysis” [Online]. Available: http://www.mat.univie.ac.at/∼neum/software/int/notation.ps.gz

[5] M. J. Lighthill, *Introduction to Fourier Analysis and Generalised Functions*. Cambridge University Press, 1959.

[6] H. Marcinkowska, *Dystrybucje i Przestrzenie Sobolewa*. Wrocław: Wydawnictwo Uniwersytetu Wrocławskiego, 1990 (in Polish).

[7] V. S. Vladimirov, *Methods of the Theory of Generalized Functions*. Taylor & Francis, 2002.

[8] C-XSC library [Online]. Available: http://www.xsc.de

Bartłomiej J. Kubica received his Ph.D. in Computer Science in 2006 from the Warsaw University of Technology. Since 2005 with WUT. Currently an assistant professor in Complex Systems Group. He co-organizes interval sessions at PPAM conferences and organizes at PARA. He co-authored a book on parallel programming and wrote several papers and presentations. His research interests focus on interval methods, optimization algorithms, multicriteria decision making, game theory and – on the other hand – multithreaded programming and parallel computations.
E-mail: bkubica@elka.pw.edu.pl
Institute of Control and Computation Engineering
Warsaw University of Technology
Nowowiejska 15/19
00-665 Warsaw, Poland

Paper

Enhancement of Power Efficiency in OFDM System by SLM with Predistortion Technique

Pankaj Kumar Sharma, R. K. Nagaria, and T. N. Sharma

Department of Electronics and Communication Engineering, Motilal Nehru National Institute of Technology, Allahabad, India

Abstract—Orthogonal Frequency Division Multiplexing (OFDM) is considered as a strong candidate for future wireless communication because it is marked by its higher frequency multiplicity and greater immunity to multipath fading. However, the main drawback of OFDM is its high amplitude fluctuations measured by peak-to-average power ratio (PAPR), which leads to power inefficiency and requires expensive high power amplifier (HPA) with very good linearity. In this paper, we propose selected mapping (SLM) with predistortion technique to decrease the nonlinear distortion and to improve the power efficiency of the nonlinear HPA. In the proposed method SLM reduces the PAPR and improves the power efficiency, the predistorter improves the bit error rate (BER) performance of the system. The PAPR reduction is possible with SLM when compared with original OFDM. After reducing the PAPR with SLM the data goes into the HPA with and without predistorter. The BER performance curves of SLM method with or without predistorter shows that, predistorter operates more effectively in SLM method than original OFDM system. At 4 dB IBO (input backoff) the conventional method with predistorter achieves 1.8 dB SNR gain than conventional method without a predistorter and at 6 dB IBO the BER performance is towards the ideal linear amplifier. The proposed system will be evaluated for OFDM system in the presence of a nonlinear power amplifier.

Keywords—nonlinear HPA, OFDM, PAPR, SLM.

1. Introduction

Orthogonal Frequency Division Multiplexing (OFDM) has been adopted as a modulation technique for high-speed wireless communications with many advantages, such as high efficiency of bandwidth and robustness against multipath fading. The OFDM based physical layer has been chosen for several wireless standards such as Digital Audio Broadcasting (DAB), Digital Video Broadcasting (DVB-T), the IEEE 802.11a [1] Local Area Network (LAN) standard and the IEEE 802.16a [2] Metropolitan Area Network (MAN) standard. In OFDM, high rate incoming signal is serial to parallel converted to N low rate data streams and each is sent over one of the N orthogonal subcarriers, because each symbol stream is transmitted over one narrowband subcarrier, each symbol stream experiences a flat fade. One of major problems in OFDM systems is large amplitude fluctuations. Due to the loss of orthogonality between subcarriers, resulting in cross talk and exhibit very large peak to average ratio (PAPR) [3] .

Here, we study the performance of OFDM in the presence of a high power amplifier (HPA). Several researchers have been proposed schemes for reducing PAPR, such as clipping method, coding method, selected mapping (SLM) and partial transmit sequence (PTS), etc. Clipping method is to clip the peak above a certain prescribed level. The merit of this clipping method is that PAPR can be easily reduced. But the BER performance becomes very worse due to many defected signals [4], [5]. Block coding is another important method for PAPR reduction. This method can reduce the PAPR without any signal distortion. However, the code rate becomes smaller than one, so that bandwidth efficiency is very poor [6]. The SLM and PTS may be classified in to the phase control scheme to escape the high peak. In SLM, one signal of the lowest PAPR is selected a set of several signals containing the same information data. In PTS, the lowest PAPR signal is made by optimally phase combining the signal sub-blocks. Both techniques are very flexible scheme and have an effective performance of the PAPR reduction without any signal distortion. However, they require much system complexity and computational burden by using of many IFFT blocks [7].

In this paper, we have used SLM technique with predistortion. Predistortion technique is applied at the transmitter side [8]–[12]. The main idea of predistortion is to shape the transmitted data symbols (data predistortion) or the input signal of the HPA amplifier (signal predistortion) so that the output signal of the HPA is less distorted. Predistortion technique improves the power density spectrum of the transmitted signal and bit error performance.

This paper is organized as follows. In Section 2, we investigate the distribution of PAPR based on the characteristics of the OFDM signals and nonlinear transmitter characteristics. Section 3, explains SLM and predistortion technique. Simulation results are shown in Section 4, and Section 5 contains the conclusion.

2. Preliminaries

2.1. The OFDM System

As shown in Fig. 1, the input data symbols are first passed through serial to parallel converter, forming a complex vector of size N. We call the vector as $X = [X_0, X_1, \ldots, X_{N-1}]^T$.

After IFFT transform the signal can be written as Eq. (1).

$$x(t) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X_n e^{j2\pi n \Delta f t}, \; 0 \le t \le NT, \qquad (1)$$

where $j = \sqrt{-1}$, $\Delta f = 1/NT$ is subcarrier spacing, and NT is OFDM symbol period.

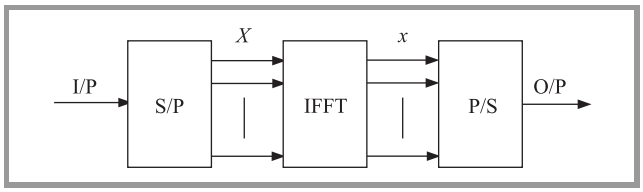


Fig. 1. Block diagram of basic OFDM system.

The PAPR of an OFDM signal, is to be

$$PAPR = \frac{\max |x(t)|^2}{E\left[|x(t)|^2\right]}, \qquad (2)$$

where $\max |x(t)|^2$ is the peak signal power and $E\left[|x(t)|^2\right]$ is the average signal power.
According to central Limit Theorem, $x(t)$ is approximately independently and identically distributed (i.i.d) the complex Gaussian random variables with zero mean and variance [8]

$$\sigma^2 = E\left[|X_n|^2\right]/2.$$

When number of subcarrier (N) is large, the complementary cumulative distributed function (CCDF) of PAPR denotes the probability of a data block exceeds a given threshold $PAPR_0$, it can be calculated as [4]

$$CCDF = 1 - (1 - e^{-PAPR_0})^N.$$

2.2. The HPA Model

2.2.1. Solid State Power Amplifier Model

In this section, we described the memory less model for the nonlinear HPA. The AM/AM and AM/PM conversion of a solid state power amplifier (SSPA) can be approximated as [2]

$$f\left[A(t)\right] = \frac{vA(t)}{\left(1 + \left[\frac{vA(t)}{A_0}\right]^{2r}\right)^{\frac{1}{2r}}}, \quad \Phi\left[A(t)\right] \approx 0, \qquad (3)$$

where $v \ge 0$ is the small signal gain, $A_0 \ge 0$ is the output saturating amplitude and $r \ge 0$ is a parameter to control the smoothness of the transition from the linear region to the saturation level.
Figure 2 shows the AM/AM and AM/PM curves for the SSPA model with $r = [2, 8, 12, 100]$ and the Saleh model with a_0, a_1, b_0, $b_1 = [2.3121,\ 1.0922,\ 5.1682,\ 10.0024]$. From the figure we see that the SSPA model with $r > 10$ resembles the limiter model which is useful for nonlinear distortion analysis [9].

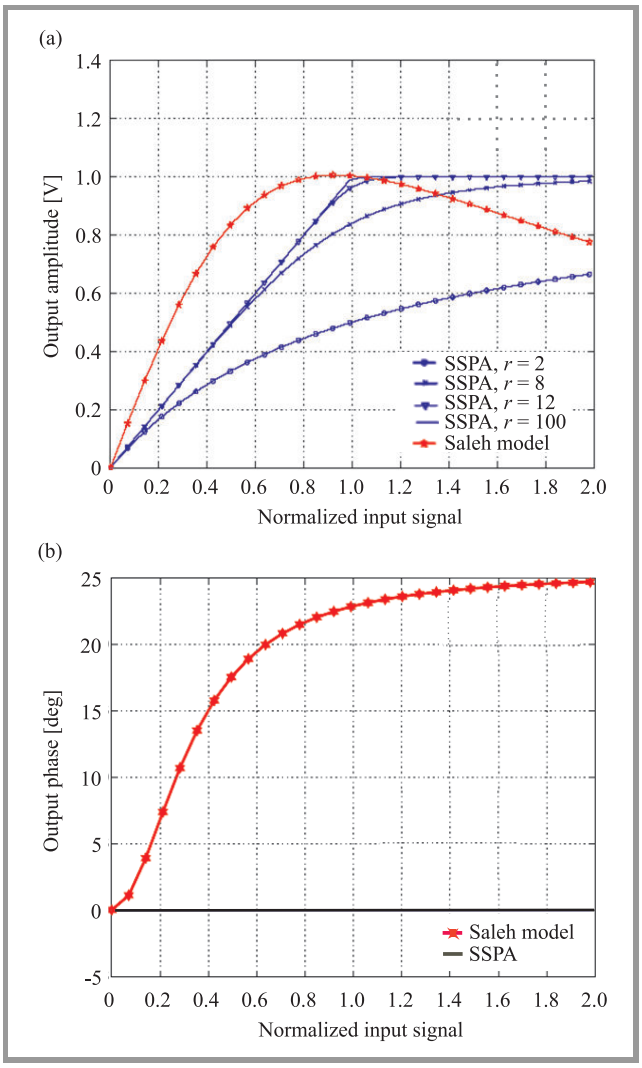


Fig. 2. (a) AM/AM and (b) AM/PM curves for SSPA model [9].

2.2.2. Effect of OFDM Signal with Nonlinearities

The nonlinear distortion at the transmitter causes interferences both inside and outside the signal bandwidth. The inside component determines the amount of bit error rate degradation of the system, whereas the outside component

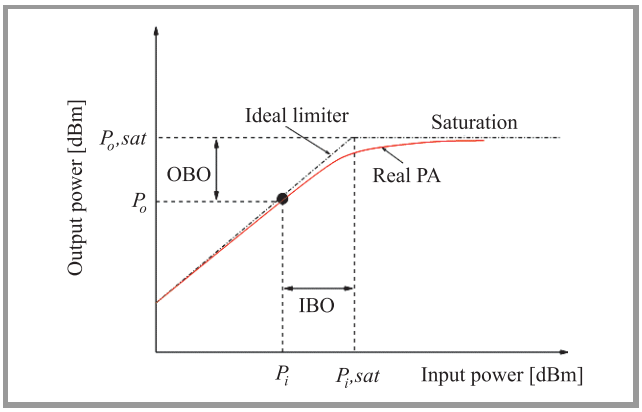


Fig. 3. A typical power amplifier response for IBO and OBO.

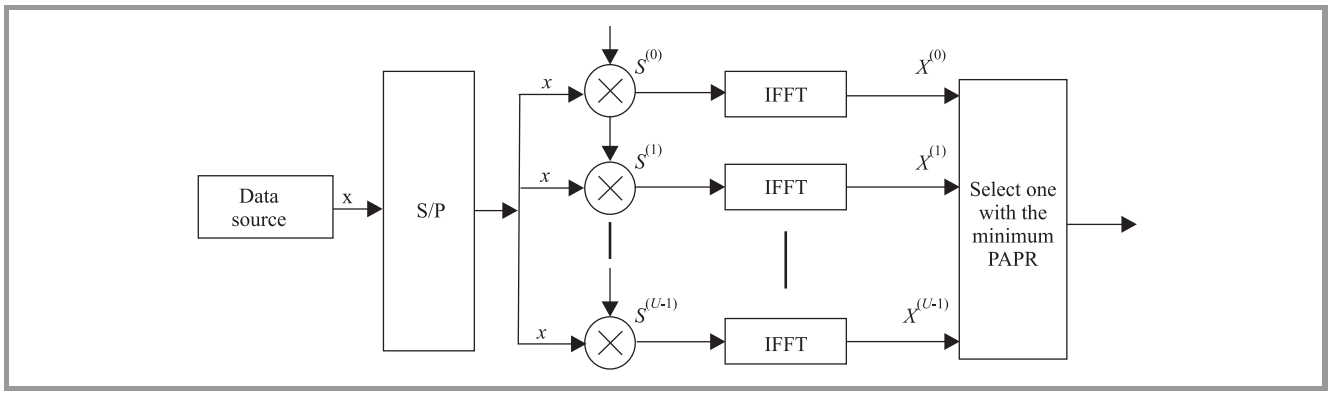


Fig. 4. OFDM system using selective mapping method.

effects the adjacent frequency bands. In other words outside component increases the out of band radiation of the signal. The transmitter nonlinear distortion include signal clipping in the analog to digital (A/D) converter, signal clipping in the IFFT and FFT processors with a limited word length, amplitude modulation (AM)/AM and AM/phase modulation (PM) distortion in the radio-frequency(RF) amplifiers. The out-of-band noise (OBN) of OFDM signals increases due to nonlinear power amplifiers operating at lower back-offs. The high PAPR of OFDM requires high backoffs at the amplifiers [7], [11].

Figure 3 shows a typical AM/AM response for an HPA, with the associated input and output back-off regions (IBO and OBO, respectively).

To avoid such undesirable nonlinear effects, a waveform with high peak power must be transmitted in the linear region of the HPA by decreasing the average power of the input signal. This is called input back-off (IBO) which results in a proportional output back-off (OBO). High back-off reduces the power efficiency of the HPA and may limit the battery life for mobile applications. In addition to inefficiency in terms of power, the coverage range is reduced, and the cost of the HPA is higher than would be mandated by the average power requirements. The input back-off and output back-off are defined as [8]:

$$IBO = 10\,\log_{10}\frac{P_{i,sat}}{P_i}\,, \tag{4}$$

$$OBO = 10\,\log_{10}\frac{P_{o,sat}}{P_o}\,, \tag{5}$$

where $P_{i,sat}$ and $P_{o,sat}$ are the input and output saturation powers, $\overline{P_i}$ and $\overline{P_o}$ are the average power of the input and output signals.

3. Proposed Technique

3.1. Selective Mapping Method

Selective mapping method (SLM) is a kind of phase rotation methods. Phase-rotated data of the lowest PAPR will be selected to transmit.

Figure 4 shows the block diagram of SLM method. Let's define data stream after S/P conversion as $X = [X_0, X_1, \ldots, X_{N-1}]^T$. Then phase-rotated data due to the phase rotation factor $B^{(u)}$ can be written as $\hat{X}^{(U)} = [\hat{X}_0^{(U)}, \hat{X}_1^{(U)}, \ldots, \hat{X}_N^{(U)}]^T$. Each $\hat{X}_N^{(U)}$ can be defined as Eq. (6).

$$\hat{X}_n^{(U)} = X_n\, b_n^{(U)}\,. \tag{6}$$

After passing through IFFT,

$$S^{(U)}(t) = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}\hat{X}_k^{(U)}\, e^{j2\pi k\Delta f t},\ 0 \le t \le NT. \tag{7}$$

Output data of the lowest PAPR is selected for transmission. PAPR reduction effect will be better as U is increased. SLM method effectively reduces PAPR without any signal distortion, but it has higher system complexity and computational burden [10].

3.2. Predistortion Technique

Although PAPR reduction method can reduce the peak power, it is not enough to suppress the out of band emission. Predistorter should be used to limit the spectral regrowth. Figure 5 shows the block diagram of the proposed scheme. In which the input signals are conversely predistorted before the HPA. Through the predistortion and nonlinear HPA, the overall characteristic can be linearized. If the modulated OFDM signal is again denoted as $s(t) = A(t)e^{j\theta(t)}$, the output samples of the predistorter can be written as

$$s_p(t) = f[A(t)]e^{j\{\theta(t)+\Psi[A(t)]\}}\,, \tag{8}$$

where $f[A(t)]$ and $\Psi[A(t)]$ are the AM/AM and AM/PM conversion of solid state power amplifier (SSPA), respectively.

The combination of a given memory less HPA and the corresponding predistorter will result in

$$s_{HPA}(t) = g\big(f[A(t)]\big)e^{j\{\theta(t)+\Psi[A(t)]+\Phi[f(A(t))]\}}. \tag{9}$$

Ideal predistortion is characterized as

$$g\big(f[A(t)]\big) = \begin{cases} \alpha A(t), & \text{if } \alpha A(t) \le A_0, \\ A_0, & \text{otherwise}, \end{cases} \tag{10}$$

where $g(.)$ is the predistorter AM/AM distortion, α is a real-valued constant ($\alpha > 0$).

In this case, the combination of the HPA and the corresponding predistorter (i.e., the total transmitter-side nonlinearity) is equivalent with the hard limiter defined in Eq. (5).

Throughout this paper we assume that the AM/PM conversion of the HPA is negligibly small and does not have to be compensated, i.e., $\Psi[A(t)] = 0$. The AM/AM conversion of the predistorter is modeled by a polynomial as [9]

$$\begin{aligned} f[A(t)] &= f_1[A(t)] + f_2[A^2(t) + \ldots + f_L[A^L(t)] \\ &= fA^L(t), \end{aligned} \tag{11}$$

where L is the order of the polynomial, $f = [f_1, f_2, \ldots, f_L]$ and $A(t) = [A^1(t), A^2(t), \ldots, A^L(t)]$.

To find the coefficient set, f, we apply the least mean square algorithm proposed in [10], which minimizes the mean squared error between real (modeled) and ideal predistorsion of the combined predistorter and HPA:

$$J(f) = E\left\{(g[fA^T(t)] - \alpha A(t))^2\right\}\sqrt{2}. \tag{12}$$

In Eq. (9), averaging is done over time. The coefficient set can be calculated recursively according to

$$\begin{aligned} f[k+1] &= f[k] - \mu \nabla f J(f[k]) \\ &= f[k] + \mu A[k] g'(f[k]A^T[k])(S_{HPA}[k] - \alpha A[k]), \end{aligned} \tag{13}$$

where ∇f denotes the gradient, $g'(.)$ is the derivative of $g(.)$ and μ a (small) positive step size. A suitable choice for the initial coefficient set is $f[0] = [1, 0, \ldots, 0]$. The steady state coefficient set is denoted as $f_\infty = \lim_{k-\infty} f(t)$. Convergence is obtained after a few thousand iterations. A drawback of this particular adaptation algorithm is the fact that $g'(.)$ and hence $g(.)$ has to be known a priori.

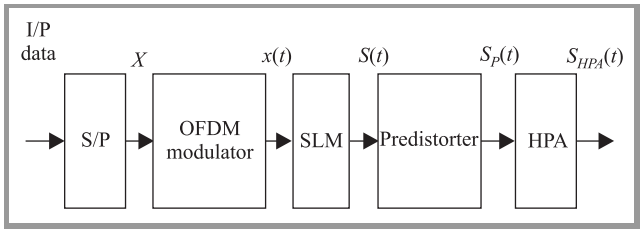


Fig. 5. Block diagram of proposed scheme.

4. Simulation Results

Figure 6 shows the CCDF performance curves with and without SLM for different values of $U = 4$, 8 and 16. From the simulation results it is clear that, for CCDF = 10^{-4}, 2 dB, 3 dB and 4 dB PAPR reduction is achieved for $U = 4$, 8 and 16 respectively. Figure 7 illustrates the BER performance of a normal OFDM with and without predistorter. From the curve, it can be observed that the normal OFDM with predistorter results 1.8 dB SNR (signal-to-noise ratio) gain than normal OFDM without predistorter in order to meet $BER = 10^{-3}$ at $IBO = 4$ dB. At 6 dB IBO, the difference between them is about 5 dB SNR. The BER performance of SLM with/without pre-

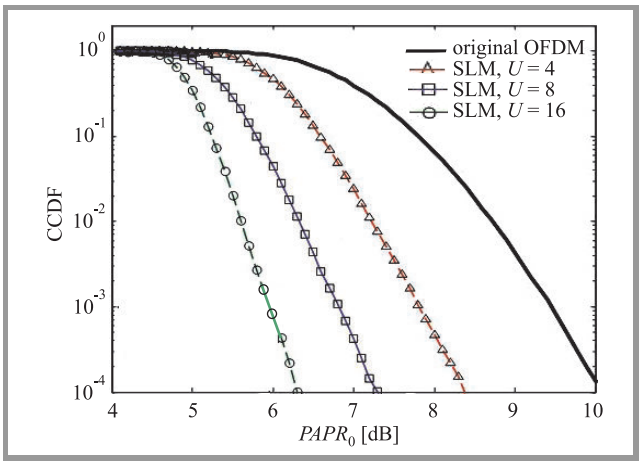


Fig. 6. CCDF of OFDM using SLM, $U = 4$, 8, 16.

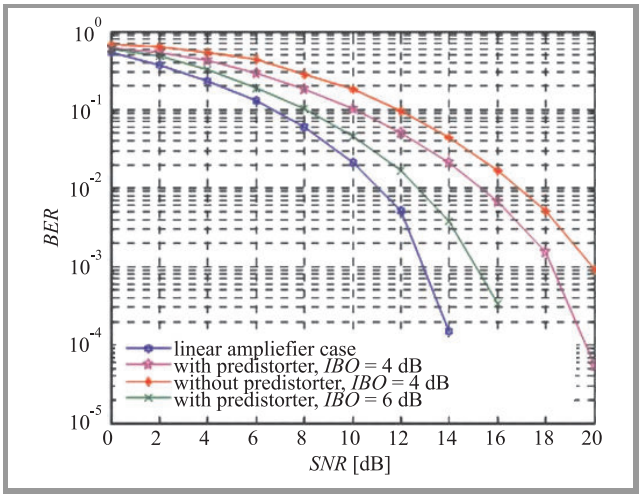


Fig. 7. BER performance of a conventional OFDM system with and without predistorter.

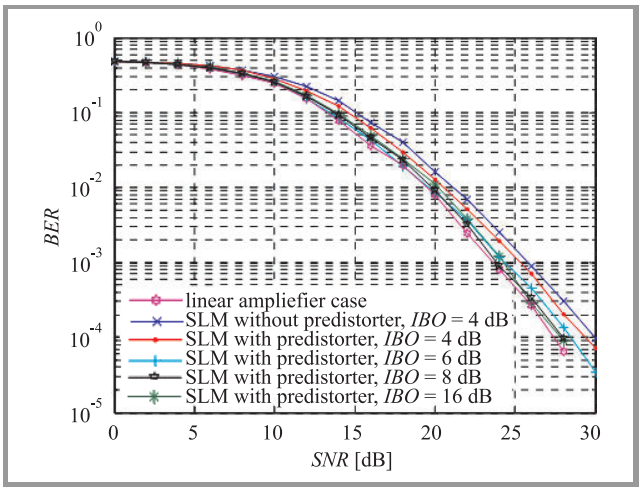


Fig. 8. BER performance of a conventional OFDM system using SLM with and without predistorter.

distorter is shown in Fig. 8. Here it is demonstrated that SLM with/without predistorter results 1 dB SNR gain improvement in order to meet $BER = 10^{-3}$ at $IBO = 4$ dB. At 6 dB IBO, the difference between them is about 2.5 dB SNR.

5. Conclusion

In this paper, we have proposed SLM method with and without predistorter to improve the power efficiency by reducing PAPR of OFDM signal. When HPA is combined with predistorter, it improves the BER performance, especially with small input back-off values. From the simulation results, it can be observed that with predistorter at 6 dB IBO, the BER curve approaches to the ideal linear amplifier as shown in Figs. 7 and 8.

References

[1] P. Banelli and S. Cacopardi, "Theoretical analysis and performance of OFDM signals in nonlinear AWGN channels", *IEEE Trans. Commun.*, vol. 48, no. 3, pp. 430–441, 2000.

[2] C. van den Bos, M. H. L. Kouwenhoven, and W. A. Serdijin, "Effect of smooth nonlinear distortion on OFDM symbol error rate", *IEEE Trans. Commun.*, vol. 49, no. 9, pp. 1510–1514, 2001.

[3] H. Ochiai and H. Imai, "Performance analysis of deliberately clipped OFDM signals", *IEEE Trans. Commun.*, vol. 50, no. 1, pp. 89–101, 2002.

[4] E. Costa and S. Pupolin, "M-QAM-OFDM system performance in the presence of a nonlinear amplifier and phase noise", *IEEE Trans. Commun.*, vol. 50, no. 3, pp. 462–472, 2002.

[5] K. R. Panta and J. Armstrong, " Effects of clipping on the error performance of OFDM in frequency selective fading channels", *IEEE Trans. Wirel. Commun.*, vol. 3, no. 2, pp. 668–671, 2004.

[6] H. Besbes and T. Le-Ngoc, "A fast adaptive predistorter for nonlinearly amplified M-QAM signals", in *Proc. IEEE Global Commun. Conf. GLOBECOM*, San Francisco, CA, USA, Nov.-Dec. 2000, pp. 108–112.

[7] P. K. Sharma, R. K. Nagaria, and T. N. Sharma, "A novel approach for power saving in OFDM system using SLM PAPR reduction technique", *IJCITAE*, vol. 3, no. 1, pp. 23–26, 2009.

[8] Y. Guo and J. R. Cavallaro, "A novel adaptive pre-distorter using LS estimation of SSPA non-linearity in mobile OFDM systems", in *Proc. IEEE Int. Symp. Circ. Sys. ISCAS*, Phoenix, Arizona, USA, May 2002, pp. 453–456.

[9] F. H. Gregorio, "Analysis & compensation of nonlinear power amplifier effects in multi antenna OFDM systems", Ph.D. thesis, Helsinki University of Technology, Nov. 2007.

[10] K. Wesołowski and J. Pochmara, " Efficient algorithm for adjustment of adaptive predistorter in OFDM transmitter", in *Proc. IEEE Vehicular Technol. Conf. VTC*, Boston, MA, USA, Sept. 2000, pp. 2491–2496.

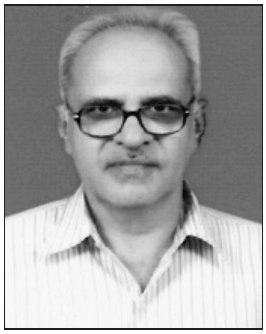
[11] R. W. Bauml, R. F. H. Fischer, and J. B. Huber, "Reducing the peak-to-average power ratio of multicarrier modulation by selected mapping", *Electron. Lett.*, vol. 32, pp. 2056–2057, Oct. 1996.

[12] J. Minkoff, "The role of AM-to-PM conversion in memory less nonlinear systems", *IEEE Trans. Commun.*, vol. COM-33, pp. 139–144, July. 1985.

Pankaj Kumar Sharma received the B.E. degree in E & T.C. Engineering from North Maharashtra University, Jalgaon (M.S), India, M. Tech. degree in Digital System, Department of Electronics and Communication Engineering from Motilal Nehru National Institute of Technology (MNNIT), Allahabad, India, and pursuing his Ph.D. degree from MNNIT, Allahabad, India. He is currently with M.I.T, Moradabad (U.P.), India, where he is working as an Associate Professor in the Department of Electronics and Communication Engineering. He is a member of Institution of Engineers (India), Institution of Electronics and Telecommunication Engineers (IETE), Computer Society of India (CSI), Indian Society for Technical Education (ISTE) and International Association of Engineers (IAEN), Hongkong. He has authored or co-authored over 21 technical papers in major national and international conferences and journals. His current research interests are in the area of wireless communication systems, especially OFDM systems with emphasis on PAPR problem.
E-mail: pankaj354518@yahoo.com
Department of Electronics
and Communication Engineering
Motilal Nehru National Institute of Technology (MNNIT)
Teliergan
Allahabad-211004, India

Rajendra Kumar Nagaria is an Associate Professor of Electronics and Communication Engineering at Motilal Nehru National Institute of Technology (MNNIT), Allahabad, India. He received a B.Tech. and M.Tech. in Electronics Engineering from Kamla Nehru Institute of Technology (KNIT) Sultanpur, India and Ph.D.(Engg.) from Jadavpur University, Kolkata, India. He has been over 22 years of teaching and research experience. He has published more than forty research papers of national and international reputes. His name is enlisted in an exclusive directory Marquis "Who's Who in the world". He is also nominated for the award as International Educator of the year 2005, by International Biographical Centre, Cambridge England. He is fellow of professional bodies like Institute of Engineers (India) and Indian Society for Technical Education. He has guided the thesis of many PG students and presently four research scholars are working under his supervision. His area of interest is mixed-mode signal processing, high speed networks/VLSI design.

E-mail: rkn@mnnit.ac.in
Department of Electronics
and Communication Engineering
Motilal Nehru National Institute of Technology (MNNIT)
Teliergan j
Allahabad-211004, India



T. N. Sharma received M.Sc.(Tech) degree in Electronics from Allahabad University in 1969 and M.E. in Electrical Engineering in 1978 and Ph.D. in 1982 from the same university. Presently, he is working as a Professor in the Department of Electronics and Communication Engineering, Motilal Nehru National Institute of Technology (MNNIT), Allahabad, India, and he has also been Head of the Department in two spells. He has more than 40 years of teaching and research experience in the area of communication engineering. He has supervised a number of M.Tech. and Ph.D. thesis. He has published several research papers in national and international journals. He is a life member of Institute of Engineers (India). His current research interests are in the area of digital communication, nonlinear control and E.M. wave propagation.

E-mail: tnsharma11@yahoo.co.in
Department of Electronics
and Communication Engineering
Motilal Nehru National Institute of Technology (MNNIT)
Teliergan j
Allahabad-211004, India

Paper

Providing QoS Guarantees in Broadband Ad Hoc Networks

Marek Natkaniec, Katarzyna Kosek-Szott, Szymon Szott, Andrzej Głowacz, and Andrzej R. Pach

Department of Telecommunications, AGH University of Science and Technology, Kraków, Poland

Abstract—This paper presents a novel QoS architecture for IEEE 802.11 multihop broadband ad hoc networks integrated with infrastructure. The authors describe its features, including MAC layer measurements, traffic differentiation, and admission control. The modules required by the network elements as well as their integration are also presented. Additionally, the paper presents results which validate its correct operation and prove its superiority over plain IEEE 802.11. The authors are convinced that the proposed solution will provide QoS support for a variety of services in future mobile ad hoc networks.

Keywords—ad-hoc networks, QoS, IEEE 802.11 EDCA.

1. Introduction

Mobile ad hoc networks (MANETs) are distributed wireless networks in which all nodes act as both terminals and routers. The deployment of MANETs is fast and effortless thanks to their autoconfiguration and lack of fixed connections. They already have multiple applications, especially in places where access to infrastructure is limited or unavailable. However, in the near future, MANETs coupled with infrastructure will allow pervasive, broadband Internet access. Special gateway routers will act as bridges connecting the ad hoc and infrastructure parts. This seems a promising solution for rural and developing areas where broadband infrastructure is unavailable. Therefore, MANETs are the focus of this paper.

When considering future networks it is important to take into account both broadband application requirements and user expectations. As a consequence, there is a need for networks to provide quality of service (QoS) support for such applications as video on demand, voice over IP, interactive entertainment, multimedia streaming, or peer-to-peer data exchange. There are several approaches which try to face this problem in MANETs. However, they lack appropriate traffic differentiation, scalability or a cross-layer approach. For these reasons they are not suitable to provide appropriate QoS.

In this paper the authors propose a service-oriented architecture for MANETs. This architecture provides a complex, cross-layer solution to the problem of QoS provisioning. Its goal is to provide end-to-end QoS in intra-MANET communication. This is accomplished by precise MAC layer measurements, traffic differentiation, traffic shaping, flexible signaling, admission control, bidirectional reservation, and resource management. Furthermore, the architecture prospectively supports IPv6, making it suitable for deployment in the near future. The authors are confident that the proposed architecture will fill the gap in providing QoS in future MANETs.

The remainder of this paper is organized as follows. Section 2 presents related work and shows the limitations of current solutions. Section 3 describes the proposed QoS architecture. The details of QoS provisioning including traffic differentiation and measurements are explained in Section 4. The measurement results presented in Section 5 validate the most important part of the proposed architecture, QoS provisioning at the MAC layer. Finally, Section 6 concludes the paper and describes future work.

2. Related Work

In order to confront the proposed architecture with the state of the art in the field, the authors briefly present research related to QoS provisioning in MANETs.

QoS protocols for MANETs need to operate in a distributed manner, provide resource reservation, admission control, traffic differentiation, and dynamic regulation. There are several solutions for QoS support in ad hoc networks which aim to meet these requirements. The most known are SWAN (Stateless Wireless Ad Hoc Networks) [2] and INSIGNIA [3]. SWAN presents a stateless approach to QoS support. This model provides resource reservation, admission control and dynamic regulation in case of congestion. It considers two traffic categories: real-time and best-effort. SWAN's negotiation capabilities are very limited and optimized to operate in a pure ad hoc scenario. INSIGNIA is a reservation-oriented QoS framework which allows resource allocation, dynamic regulation, and per-flow management. Each node maintains soft-state reservations based on available resources in order to support adaptive services. To distribute QoS information, in-band signaling is used. The main drawback of INSIGNIA is its complex signaling and lack of scalability. Other QoS mechanisms for MANETs include CEDAR [9], FQMM [8], AQOR [10], courtesy piggybacking [7], and AAC [6]. An overview of selected protocols can be found in [4].

3. QoS Network Architecture

The proposed QoS architecture consists of two main logical units, the mobile node (MN) and the gateway (GW). The MN has a double role acting both as a user terminal

which provides QoS support to the end user, and as a router which forwards the traffic of neighboring nodes. The GW can provide connectivity with infrastructure, participate in admission control and dynamic regulation.

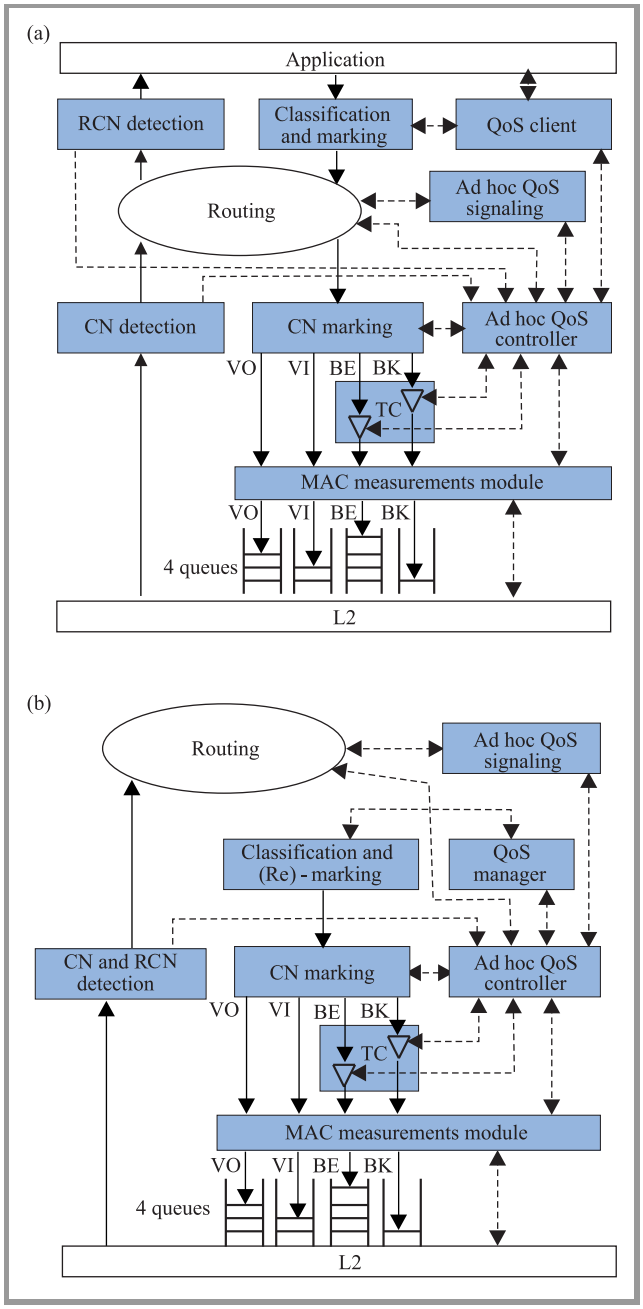


Fig. 1. QoS modules in mobile node (a) and gateway (b).

Figure 1 presents the proposed QoS modules (grey colour) in MN and GW. In this figure, the solid arrows correspond to data packet processing; the dashed arrows correspond to control information flow. End-to-end QoS resource management is supported through the interoperation of MN and GW. More specifically, this occurs through the interaction between the QoS client (QoSC) in the MN and QoS manager (QoSM) in the GW. QoSC retrieves the necessary QoS parameters from applications (through an API) and maps them to network QoS parameters. QoSC also performs per-flow end-to-end QoS signaling, controls the classification and marking module (which marks the traffic class and flow label fields in IPv6 headers), notifies the applications of the state of network interfaces, and performs the synchronization of network resource reservation. QoSM provides the same functionalities as QoSC but without the interface to the application layer.

The ad hoc QoS controller (AHQoSC) is the central module in both MN and GW, which coordinates the work of the other modules. Through interaction with the results of the MAC measurements module (MMM) it determines the available resources in the wireless medium. The main responsibilities of AHQoSC are: admission control in the ad hoc path, traffic control, reaction to congestion, and participating in resource management. If resources from higher priority access categories are available, AHQoSC reallocates them to lower priority access categories in order to maximize network utilization.

Measurements of incoming and outgoing traffic are performed by the MMM. It provides AHQoSC with information regarding bandwidth utilization, transmission delay, current transmission rate, frame statistics and idle intervals.

The ad hoc QoS signaling (AHQoSSig) module is responsible for session establishment. This includes QoS negotiations as well as probing for available bandwidth and session setup between infrastructure and ad hoc or vice versa.

The IEEE 802.11 EDCA [1] function is used to provide traffic differentiation at the MAC layer. Additionally, to enhance traffic differentiation, the traffic controller (TC) module is used to shape lower-priority flows.

To make the architecture aware of overload situations, the congestion notification (CN) signaling mechanism is proposed. It is implemented with the use of three modules: CN marking (CNM), CN detection (CND), and receiver CN detection (RCND). CNM is responsible for setting certain bits (defined as CN bits) located in the Traffic Class field of the IP header of outgoing data packets. This happens when the target bandwidth of a given access category is exceeded. CND monitors incoming data packets. If it finds that the CN bit in a packet is set, the source of the traffic is notified. In the case of real-time traffic, the source should try to re-establish the session by sending a new probing request to the destination. As a result, the application will either adapt its rate or drop the session. In the case of non real-time traffic, the source should reconfigure its TC to shape this traffic. CNM is located in the data flow after CND. Therefore, if the CN bit is set, the congestion will not be discovered at the next node after the source node. This is a very serious problem if the next node is also the destination node. To assure the functionality of discovering violations in such a situation, RCND was introduced. It operates similarly to CND.

The GW is able to support the same functionalities as the MN, but does not interact with applications (since it works only at the IP layer and below) (Fig. 1b).

The routing module implements an ad-hoc routing protocol, AODV [11].

4. QoS Provisioning

QoS provisioning is achieved by traffic differentiation at the MAC layer. The proposed architecture utilizes the four traffic access categories of the IEEE 802.11 EDCA function. Thus, a finer service granularity and better cross-layer integration are achieved in comparison to related work. MAC layer measurements are fast and efficient thanks to driver implementation. It is assumed that all wireless devices work in promiscuous mode. This allows measurements with the use of a single wireless adapter, which is an additional advantage over literature, where usually two adapters are required.

4.1. Traffic Differentiation

The proposed traffic differentiation model extends SWAN to provide a finer service granularity. SWAN considered only two traffic categories, whereas in the proposed architecture four access categories are utilized. These classes are voice (VO), video (VI), best effort (BE), and background (BK). They are ordered by priority, based on their QoS requirements. They also correspond to the access categories of EDCA, which is responsible for traffic differentiation in channel access. Each access category has separate access parameters and a separate hardware priority queue (Fig. 1). Thanks to MAC layer traffic differentiation, the differentiation at the IP layer is simplified, compared to the literature.

To provide further separation of traffic and enable better resource control, each access category has an impassable bandwidth limit. Additionally, bandwidth can be reallocated on-demand to ensure maximum network utilization. Unoccupied bandwidth of the high priority categories can be re-assigned to low priority ones. Classless traffic (e.g., traffic from stations not using the proposed architecture) is put into the BK category. This ensures interoperability with non-QoS stations.

Traffic shaping occurs in TC for BK and BE traffic. Shaping is not used for VO and VI categories, since it is unacceptable for sensitive multimedia traffic. In case of congestion, instead of shaping, a lower transmission rate with a different voice or video codec is negotiated. Traffic differentiation for these categories is further complemented by admission control performed by AHQoSC.

4.2. MAC Layer Measurements

MAC layer measurements are crucial for QoS support in MANETs. They are performed by MMM, which is located in the driver between the IP layer and the wireless card firmware. This approach allows for fast and efficient operation on transmitted and received packets.

The authors propose a universal approach to implementing MAC layer measurements in mobile devices, which usually have only one wireless adapter. The SWAN approach is capable of capturing all IEEE 802.11 frames; however, it requires two wireless adapters: one for measuring and another for normal Tx/Rx operation. In the proposed architecture, the wireless adapter is set to promiscuous mode, which allows for concurrent transmissions and measurements using only a single wireless adapter. This is a significant advantage over SWAN. However, this approach does not allow the capturing of certain control and management frames. Therefore, a new PHY layer bandwidth estimation method was implemented.

The following parameters are measured:

- Per-category and overall average packet delay – the time between receiving a packet from the IP layer and the completion of the RTS-CTS-DATA-ACK exchange in EDCA.
- Per-category and overall bandwidth utilization – achieved by sensing the medium and constructing periodic statistics about bandwidth occupancy.
- Transmission rate – the currently chosen rate (wireless cards support multi-rate modes), required to evaluate available bandwidth.
- Number of stations – the estimated number of active stations in the neighbourhood, required to determine the contention between MNs in the wireless channel and to evaluate the available bandwidth.

Each access category is distinguished by its Traffic Class field in the IPv6 header, which allows per-category measurements. The measurements are periodically analyzed by AHQoSC to adjust traffic shaping and to perform admission control decisions.

5. Measurement Results

The objective of the experiments presented in this section was to validate the most important part of the proposal (QoS provisioning at the MAC layer) and compare traffic differentiation in three different network configurations: IEEE 802.11 DCF, IEEE 802.11 EDCA, and the proposed architecture.

The measurement scenario is presented in Fig. 2. The wireless network consisted of four MNs and one GW equipped with WLAN cards set to ad hoc mode. These cards were based on the Atheros chipset and the madwifi driver [5]. The madwifi driver was modified to obtain correct EDCA operation in ad hoc mode. Missing QoS fields in certain control frames were added to allow the correct recognition of QoS capable stations. The wireless cards used HR/DSSS PHY with a constant rate of 11 Mbit/s. All possible interferences were avoided and there were no hidden stations. The testbed was implemented in an IPv6 environment.

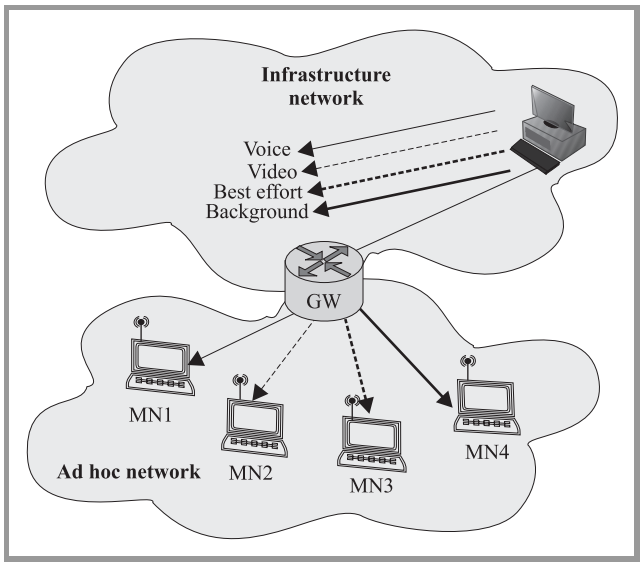


Fig. 2. Measurement scenario.

To provide the same conditions for each access category, the mgen tool was used to generate traffic. The traffic was generated in such a way so as to put the network in saturation. Three different configurations were considered and compared for their QoS capabilities: DCF, EDCA, and the solution proposed in this paper. The throughput values achieved when the DCF function was used are presented in Fig. 3. The throughput for each of the four ACs was the same, therefore only the average throughput values are illustrated in the figure.

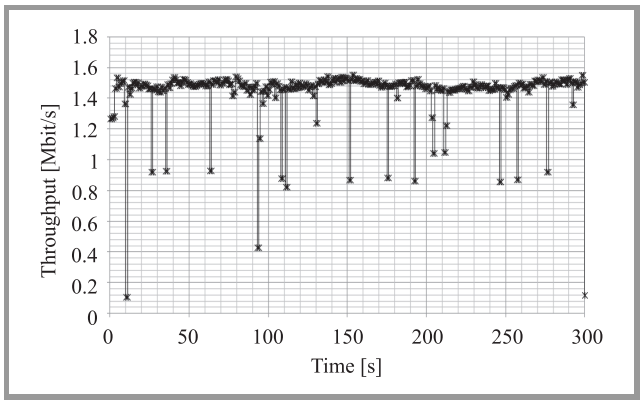


Fig. 3. Throughput for DCF.

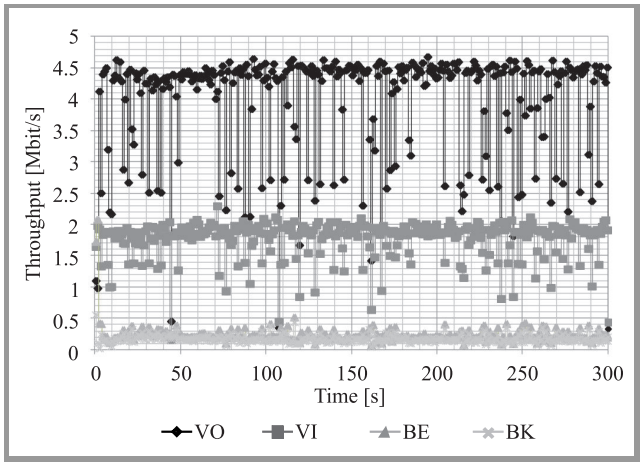


Fig. 4. Throughput for EDCA.

The throughput values achieved when the EDCA function was used are presented in Fig. 4. The differentiation between traffic classes can be observed, however, higher priority traffic highly influences lower priority traffic. This is because we are not able to control the throughput level within each access category using only EDCA differentiation. VO and VI traffic have considerably degraded BE and BK traffic. Furthermore, the throughput of the VO and VI categories has high variation. This is because the network is saturated.

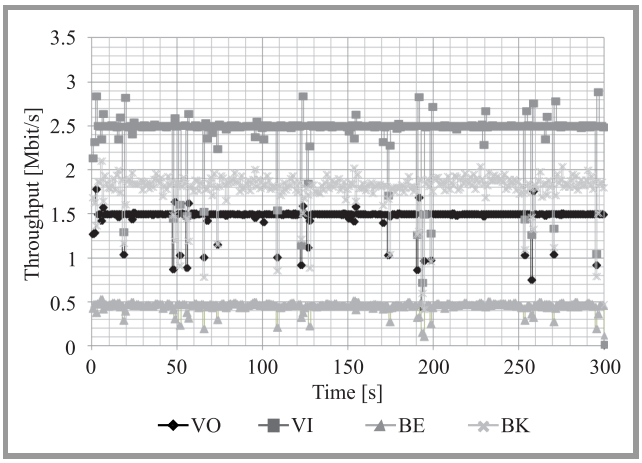


Fig. 5. Throughput for the proposed solution.

The throughput values achieved when the proposed solution was used are presented in Fig. 5. The following bandwidth utilization limits within each access category were assumed in our testbed: 1.5 Mbit/s for VO, 2.5 Mbit/s for VI, 0.5 Mbit/s for BE, and 1.8 Mbit/s for BK. The total sum of these limits did not exceed 95% of the maximum available network bandwidth, because additional control traffic was generated between MNs (e.g., beacons, probe request and response frames). The bandwidth limits for VO and VI were imposed through the use of the admission control mechanism implemented in the AHQoSC. The limits for BE and BK were imposed through the use of traffic shaping mechanisms implemented in the TC module. Using the proposed approach, excellent traffic differentiation for all categories can be achieved. Additionally, the throughput level for each access category can be easily controlled and managed. Furthermore, thanks to the proposed solution the variation of throughput has decreased considerably. This also influences the achieved delay values (Fig. 8).

The delay values achieved using the three tested methods are presented in Figs. 6, 7, and 8. In the case of DCF the delay was at least 100 ms which can be considered acceptable for real-time traffic. However, as can be seen from Fig. 6, DCF does not allow traffic differentiation. In the case of EDCA the average delay for VI, BE and BK increased.

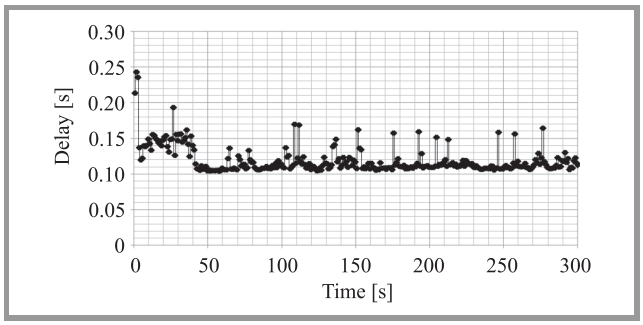


Fig. 6. Delay for DCF.

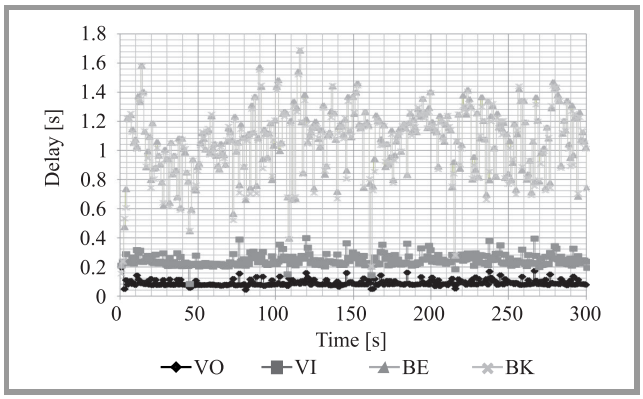


Fig. 7. Delay for EDCA.

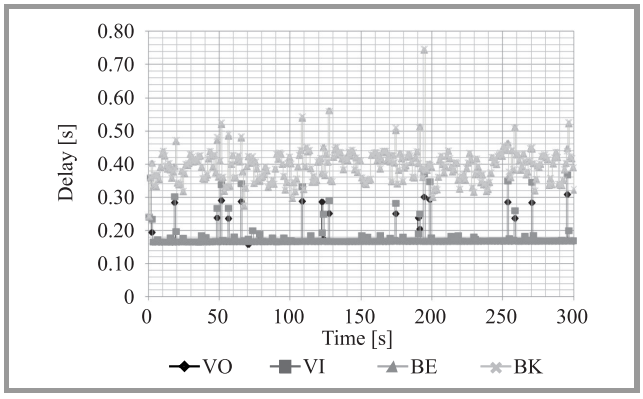


Fig. 8. Delay for the proposed solution.

Finally, with the use of the proposed solution the delay of these three ACs was improved.

6. Concluding Remarks

This paper describes an innovative approach to the problem of QoS support in mobile ad hoc networks. The proposed architecture has the following innovative features:

- QoS negotiations and traffic differentiation for the four IEEE 802.11 EDCA access categories with additional lower-priority traffic shaping.
- No requirements for per-flow or aggregate state information in the intermediate nodes (traffic classes are controlled locally using MAC layer measurements).
- Admission control performed only at the source node with the use of request/response probes which check the available per-class bandwidth on the path from the source to the destination.
- Reacting to overload situations with the use of efficient signaling techniques.
- Implementation of MAC layer measurements in WLAN cards.
- A modified driver which provides QoS capabilities in ad hoc mode.
- IPv6 support.

The initial validation of the proposed approach was presented in the form of measurements. These experiments showed that neither IEEE 802.11 without QoS support nor IEEE 802.11 with EDCA can provide adequate QoS provisioning. In the first case there was no traffic differentiation at all. In the second case, despite introducing traffic differentiation, there was no bandwidth separation between access categories (i.e., traffic of one category influenced traffic of other classes). In most cases this is insufficient to effectively provide QoS in multihop ad hoc networks. The proposed solution turned out to be the best approach with each access category achieving as much bandwidth as requested – there was no inter-category interference. Therefore, our unique solution makes it possible to achieve end-to-end QoS in an ad hoc network.

The presented architecture will hopefully meet user expectations regarding QoS guarantees in mobile ad hoc networks and provide them with ubiquitous and satisfactory, broadband Internet access. Future work will be focused on continuing the validation process in multihop environments (with different metrics) and enhancing the architecture to increase its performance.

Acknowledgements

This work has been carried out under the Polish Ministry of Science and Higher Education grant no. PBZ/G018/T02/2007.

References

[1] "IEEE 802.11 Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications". IEEE Inc., New York, June 2007.

[2] G.-S. Ahn *et al.*, "Supporting service differentiation for real-time and best-effort traffic in stateless wireless ad-hoc networks (SWAN)", *IEEE Trans. Mobile Comp.*, vol. 1, no. 3, pp. 192–207, 2002.

[3] S. B. Lee *et al.*, "INSIGNIA: An IP-based quality of service framework for mobile ad hoc networks", *J. Parallel and Distrib. Comp.*, *Wirel. Mobile Comput. Commun.*, special iss, vol. 60, no. 4, pp. 374–406, 2000.

[4] M. Inayatullah, S. Ahmad, and A. Salam, "Optimized QOS protocols for small-sized manets", in *Proc. Int. Conf. Emerging Technol. IEEE ICET 2006*, Peshawar, Pakistan, 2006.

[5] "MADWiFi – Multiband Atheros Driver for WiFi" [Online]. Available: http://madwifi.org

[6] R. de Renesse, V. Friderikos, and H. Aghvami, "Cross-layer cooperation for accurate admission control decisions in mobile ad hoc networks", *IET Commun.*, vol.1, no. 4, pp. 577–586, 2007.

[7] W. Liu and Y. Fang, "Courtesy piggybacking: supporting differentiated services in multihop mobile ad hoc networks", in *Proc. 23rd Ann. Joint Conf. Comp. Commun. Soc. INFOCOM 2004*, Hong Kong, China, 2004, vol. 2, pp. 1273–1283.

[8] R. Braden, D. Clark, and S. Shenker, "Integrated Services in the Internet Architecture-an Overview", IETF RFC 1663, June 1994.

[9] P. Sinha, R. Sivakumar, and V. Bharghavan, "CEDAR: a core-extraction distributed ad hoc routing algorithm", in *Proc. Conf. Comp. Commun. INFOCOM'99*, New York, NY, USA, 1999, pp. 202–209.

[10] Q. Xue and A. Ganz, "Ad hoc QoS on-demand routing in mobile ad hoc networks", *J. Parallel Distrib. Comput.*, vol. 63, iss. 2, pp. 154–165, 2003.

[11] C. Perkins *et al*., "Ad hoc On-Demand Distance Vector Routing", IETF RFC 3561, July 2003.

Marek Natkaniec received the M.Sc. and Ph.D. degrees in Telecommunications from the Faculty of Electrical Engineering, AGH University of Science and Technology, Kraków, Poland in 1997 and 2002, respectively. In 1997 he joined AGH University of Science and Technology. Currently he is working as an assistant professor at the Department of Telecommunications, AGH University. His general research interests are in wireless networks. Particular topics include wireless LANs, designing of protocols, formal specification of communication protocols, modeling and performance evaluation of communication networks, quality of service, multimedia services provisioning and cooperation of networks. He is a reviewer to international journals and conferences. He has/had actively participated in European projects: MOCOMTEL, PRO-ACCESS, DAIDALOS I, DAIDALOS II, CONTENT, CARMEN, MEDUSA, HECTOR, FLAVIA as well as grants supported by the Polish Ministry of Science and Higher Education. Marek Natkaniec co-authored five books and over 100 research papers.
E-mail: natkanie@kt.agh.edu.pl
Department of Telecommunication
AGH University of Science and Technology
Mickiewicza Av. 30
30-059 Kraków, Poland

Katarzyna Kosek-Szott received her M.Sc. and Ph.D. degrees in Telecommunications (both with honours) from the AGH University of Science and Technology, Kraków, Poland in 2006 and 2011, respectively. Currently she is working as an Assistant Professor at the Department of Telecommunications, AGH University of Science and Technology. Her general research interests are focused on wireless networking. The major topics include wireless LANs (especially ad-hoc networks) and quality of service provisioning within these networks. She is a reviewer for international journals and conferences. She has been involved in several European projects: DAIDALOS II, CONTENT, CARMEN, FLAVIA as well as grants supported by the Polish Ministry of Science and Higher Education. She has also co-authored a number of research papers.
E-mail: kosek@kt.agh.edu.pl
Department of Telecommunication
AGH University of Science and Technology
Mickiewicza Av. 30
30-059 Kraków, Poland

Szymon Szott received his M.Sc. and Ph.D. degrees in Telecommunications (both with honours) from the AGH University of Science and Technology, Kraków, Poland in 2006 and 2011, respectively. Currently he is working as an assistant professor at the Department of Telecommunications, AGH University. His professional interests are related to wireless networks (in particular: QoS provisioning and security of ad-hoc networks). He is a reviewer for international journals and conferences. He has been involved in several European projects (DAIDALOS II, CONTENT, CARMEN, MEDUSA, FLAVIA) as well as grants supported by the Ministry of Science and Higher Education. He is the co-author of several research papers and one book chapter.
E-mail: szott@kt.agh.edu.pl
Department of Telecommunication
AGH University of Science and Technology
Mickiewicza Av. 30
30-059 Kraków, Poland

Andrzej Głowacz finished Ph.D. studies in Computer Science in 2006 and obtained Ph.D. in Telecommunications in 2007. Currently he is Assistant Professor in AGH University of Science and Technology. He has been working as an IT Expert in numerous commercial projects, grants of the Polish Ministry of Science and Education and the European research projects: DAIDALOS, DAIDALOS 2, EuroNGI, EuroFGI, OASIS Archive, and GAMA. His main professional areas are: Wireless QoS, modern transport protocols, advanced Linux programming, network simulations, and multimedia

indexing systems. He is author of over forty scientific papers and technical reports, he also serves as a reviewer of several international journals and conferences.
E-mail: glowacz@kt.agh.edu.pl
Department of Telecommunication
AGH University of Science and Technology
Mickiewicza Av. 30
30-059 Kraków, Poland

Andrzej R. Pach received the M.Sc. degree in Electrical Engineering and the Ph.D. degree in Telecommunications from the University of Science and Technology, Krakow, Poland, in 1976 and 1979, respectively, and the Ph.D.Hab. in Telecommunications and Computer Networks from the Warsaw University of Technology in 1989. In 1979, he joined the Department of Telecommunications at the University of Science and Technology, where he is currently a Professor and Chair. He spent his sabbatical leaves at CNET, France and University of Catania, Italy. He is the Vice-President of the Foundation for Progress in Telecommunications and serves as a chairman of the IEEE Communications Society Chapter. He has been a consultant to governmental institutions and telecom operators in modern telecommunication networks. His research interests include design and performance evaluation of broadband networks, especially quality of service and network performance of access networks and wireless LANs. He has/had actively participated in COST, Eureka Celtic, and FP European programs (ACTS, ESPRIT, IST, ICT, and Security). He co-authored more than 200 publications including 6 books. He served as a technical editor to IEEE Communications Magazine and is an editor-in-chief to Digital Communications – Technologies and Services. He has also been appointed as an expert in Information and Communications Technologies by the European Commission.
E-mail: pach@kt.agh.edu.pl
Department of Telecommunication
AGH University of Science and Technology
Mickiewicza Av. 30
30-059 Kraków, Poland

Paper

SVD Audio Watermarking: A Tool to Enhance the Security of Image Transmission over ZigBee Networks

Mohsen A. M. El-Bendary[a], Atef Abou El-Azm[b], Nawal El-Fishawy[b], Farid Shawki[b], Mostafa A. R. El-Tokhy[a], Fathi E. Abd El-Samie[b], and H. B. Kazemian[c]

[a] *Department of Communication Technology, Faculty of Industrial Education, Helwan University, Egypt*
[b] *Department of Electronics and Electrical Communications, Faculty of Electronic Engineering, Menoufia University, Menouf, Egypt*
[c] *Intelligent Systems Research Centre, Faculty of Computing, London Metropolitan University, UK*

Abstract—The security is important issue in wireless networks. This paper discusses audio watermarking as a tool to improve the security of image communication over the IEEE 802.15.4 ZigBee network. The adopted watermarking method implements the Singular-Value Decomposition (SVD) mathematical technique. This method is based on embedding a chaotic encrypted image in the Singular Values (SVs) of the audio signal after transforming it into a 2-D format. The objective of chaotic encryption is to enhance the level of security and resist different attacks. Experimental results show that the SVD audio watermarking method maintains the high quality of the audio signals and that the watermark extraction and decryption are possible even in the presence of attacks over the ZigBee network.

Keywords—audio watermarking, copyright protection, IEEE 802.15.4, SVD.

1. Introduction

With the increase in utilization of wireless devices, especially Bluetooth and ZigBee devices, the need for data security has evolved. Generally, wireless network security is a problem, because the transmitted data can be easily overheard by eavesdropping devices if no security strategies have been adopted. The choice of security levels is based on the application [1]. ZigBee has a set of security services implementing the Advanced Encryption Standard (AES). In this paper, we use digital audio watermarking to enhance the security of image communication over ZigBee networks.

Digital watermarking has found several applications in image, video, and audio communication. Watermarking is the art of embedding a piece of information in a cover signal. It can achieve several objectives such as information hiding, copyright protection, fingerprinting, and authentication [2]. Several algorithms have been proposed for watermarking, especially for image and video watermarking [3]–[5]. Some of these algorithms are designed for the efficient embedding and detection of the watermark, but most of them aim at the successful extraction of the embedded watermark. On the other hand, most of the audio watermarking algorithms are designed to achieve an efficient detection of the watermark without extracting meaningful information from the watermarked audio signal [6]–[7].

There is a need for a robust audio watermarking method with a higher degree of security, which can be achieved by embedding encrypted images in audio signals. In this paper, the chaotic Baker map is used for the encryption of the watermark image [8]. Then, the watermark is embedded in the audio signal using the SVD mathematical technique. The audio signal is first transformed into a 2-D format and the SVs of the resulting matrix are used for watermark embedding. The watermarked audio signal is then transmitted over the ZigBee network [9].

Embedding encrypted images in audio signals achieves two levels of security; the level of encryption and the level of watermarking. Encryption can be performed with either diffusion or permutation-based algorithms. Diffusion-based algorithms are very sensitive to noise, while permutation based algorithms are more immune to noise. That is why the adopted encryption scheme in this paper is based on the chaotic Baker map, which is permutation-based.

The paper is organized as follows. In Section 2, the IEEE 802.15.4 ZigBee standard is discussed. Section 3 explains the SVD audio watermarking method. In Section 4, chaotic encryption is briefly discussed. The simulation results are introduced in Section 5. Finally, the concluding remarks are given in Section 6.

2. ZigBee Standard

IEEE 802.15.4 is a Low-Rate Wireless Personal Area Network (LR-WPAN) standard used for providing simple and low-cost communication networks. LR-WPANs are intended for short-range operation, and use little or no infrastructure. This standard focuses on applications with limited power and relaxed throughput requirements, with the main objectives being ease of installation and reliable data transfer. This allows small, power-efficient, inexpensive solutions to be implemented for a wide range of devices. Low power consumption can be achieved by allowing a device

to sleep, only waking into active mode for brief periods. Enabling such low-duty-cycle operation is at the heart of the IEEE 802.15.4 standard [10]. The ZigBee specification document is short, allowing a small and simple stack, in contrast to other wireless standards such as Bluetooth [11].

The IEEE 802.15.4 standard conforms to the established regulations in Europe, Japan, Canada and the United States, and defines two physical (PHY) layers; the 2.4-GHz and the 868/915-MHz band PHY layers. Although the PHY layer chosen depends on local regulations and user preference, for the purposes of this document only the higher data-rate, worldwide, unlicensed 2.4-GHz band will be considered. Sixteen channels are available in the 2.4-GHz band, numbered 11 to 26, each with a bandwidth of 2 MHz and a channel separation of 5 MHz. LR-WPAN output powers are around 0 dBm and typically operate within a 50-m range [12].

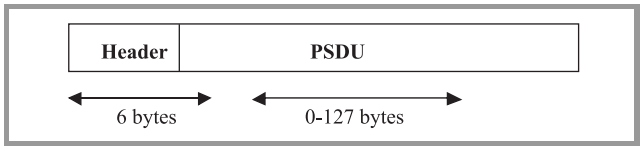


Fig. 1. ZigBee packet format.

The structure of the ZigBee packet is shown in Fig. 1. The header contains three fields; a preamble of 32 bits for synchronization, a packet delimiter of 8 bits, and a physical header of 8 bits. The Physical Service Data Unit (PSDU) field is the data field with 0 to 1016 bits. ZigBee uses an error detection/retransmission technique through a Cyclic-Redundancy Check (CRC) scheme. For image communication over the ZigBee network, data fragmentation into packets is implemented.

3. SVD Audio Watermarking

The SVD mathematical technique provides an elegant way for extracting algebraic features from a 2-D matrix. The main properties of the matrix of SVs can be exploited in audio watermarking. When a small perturbation happens to the original data matrix, no large variations occur in the matrix of SVs, which makes this technique robust to attacks [13]-[15].

The steps of the SVD audio watermark embedding algorithm are summarized as follows:

1. The 1-D audio signal is transformed into a 2-D matrix (**A** matrix).
2. The SVD is performed on the **A** matrix.

$$\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathrm{T}}. \tag{1}$$

   where: **U** and **V** are orthogonal matrices such that $\mathbf{U}^{\mathrm{T}}\mathbf{U} = \mathbf{I}$, and $\mathbf{V}^{\mathrm{T}}\mathbf{V} = \mathbf{I}$, $\mathbf{S} = diag(\sigma_1, \ldots, \sigma_P)$, where $\sigma \geq \sigma_2 \geq \ldots \geq \sigma_P \geq 0$ are the SVs of **A**, the columns of **U** are called the left singular vectors of **A**, and the columns of **V** are called the right singular vectors of **A**.

3. The chaotic encrypted watermark (**W** matrix) is added to the SVs of the original matrix.

$$\mathbf{D} = \mathbf{S} + k\mathbf{W}. \tag{2}$$

   A small value of k of about 0.01 is required to keep the audio signal undistorted.

4. The SVD is performed on the new modified matrix (**D** matrix).

$$\mathbf{D} = \mathbf{U}_{\mathrm{w}}\mathbf{S}_{\mathrm{w}}\mathbf{V}_{\mathrm{w}}^{\mathrm{T}}. \tag{3}$$

5. The watermarked signal in 2-D format ($\mathbf{A}_w$ matrix) is obtained using the modified matrix of SVs ($\mathbf{S}_w$ matrix).

$$\mathbf{A}_{\mathrm{w}} = \mathbf{U}\mathbf{S}_{\mathrm{w}}\mathbf{V}^{\mathrm{T}}. \tag{4}$$

6. The 2-D $\mathbf{A}_{\mathrm{w}}$ matrix is transformed again into a 1-D audio signal.

To extract the possibly corrupted watermark from the possibly distorted watermarked audio signal, given $\mathbf{U}_{\mathrm{w}}$, $\mathbf{S}$, $\mathbf{V}_{\mathrm{w}}$ matrices, and the possibly distorted audio signal, the above steps are reversed as follows:

1. The 1-D audio signal is transformed into a 2-D matrix $\mathbf{A}_{\mathrm{w}}^*$. The $*$ refers to the corruption due to attacks.
2. The SVD is performed on the possibly distorted watermarked image ($\mathbf{A}_{\mathrm{w}}^*$ matrix).

$$\mathbf{A}_{\mathrm{w}}^* = \mathbf{U}^*\mathbf{S}_{\mathrm{w}}^*\mathbf{V}^{*\mathrm{T}}. \tag{5}$$

3. The matrix that includes the watermark is computed.

$$\mathbf{D}^* = \mathbf{U}_{\mathrm{w}}\mathbf{S}_{\mathrm{w}}^*\mathbf{V}_{\mathrm{w}}^{\mathrm{T}}. \tag{6}$$

4. The possibly corrupted encrypted watermark is obtained.

$$\mathbf{W}^* = (\mathbf{D}^* - \mathbf{S})/k. \tag{7}$$

5. The obtained matrix $\mathbf{W}^*$ is decrypted.
6. The correlation coefficient between the decrypted matrix and the original watermark is estimated. If this coefficient is higher than a certain threshold, the watermark is present.

4. Chaotic Encryption

Chaotic encryption of the watermark image is performed using the chaotic Baker map. The Baker map is a chaotic map that generates a permuted version of a square matrix [16]. In its discretized form, the Baker map is an efficient tool to randomize a square matrix of data. The dis-

cretized map can be represented for an $R \times R$ matrix as follows:

$$B(r_1, r_2) = \left\lceil \frac{R}{n_i}(r_1 - R_i) + r_2 \bmod \left(\frac{R}{n_i}\right), \frac{n_i}{R}\left(r_2 - r_2 \bmod \left(\frac{R}{n_i}\right)\right) + R_i \right\rceil, \tag{8}$$

where $B(r_1, r_2)$ are the new indices of the data item at (r_1, r_2), $R_i \leq r_1 \leq R_i + n_i$, $0 < r_2 < R$, and $R_i = n_1 + n_2 + \ldots + n_i$.

In steps, the chaotic encryption is performed as follows:

1. An $R \times R$ square matrix is divided into R rectangles of width n_i and number of elements R.
2. The elements in each rectangle are rearranged to a row in the permuted rectangle. Rectangles are taken from left to right beginning with upper rectangles then lower ones.
3. Inside each rectangle, the scan begins from the bottom left corner towards upper elements.

Figure 2 shows an example of the chaotic encryption of an 8×8 square matrix (i.e. $R = 8$). The secret key is $S_{key} = [n_1, n_2, n_3] = [2, 4, 2]$.

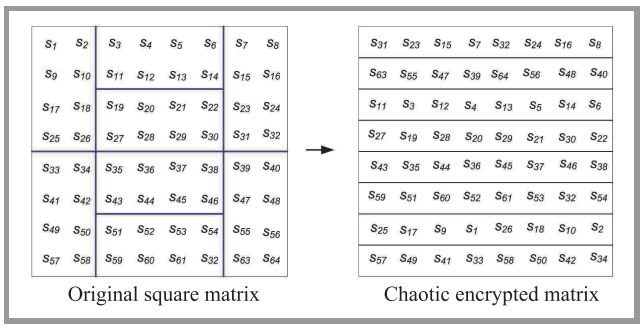


Fig. 2. Chaotic encryption of an 8×8 matrix.

5. Simulation Results

In this section, the computer simulation results are presented. The effectiveness of the SVD audio watermarking method is studied for the transmission of watermarked audio signals over fading channels. Firstly, an uncorrelated block-fading channel is considered. It is a slow and frequency non-selective channel, where symbols in a block undergo a constant fading effect. This means that the Doppler spread is equal to zero ($f_d = 0$) [17]. Also, the correlated Rayleigh fading channel is considered. The channel model utilized is the Jakes' model [18]. The assumed mobile ZigBee device velocity (v) is 10 miles/hour, and the carrier frequency is 2.46 GHz. The Doppler spread is $f_d = 366$ Hz. Both the logo and the cameraman images are used in the simulation experiments.

The chaotic Baker map is used to encrypt the watermark image. The encrypted image is then used as a watermark to be embedded in the Handel signal available in Matlab and shown in Fig. 3. This signal is transmitted over the ZigBee network at different signal-to-noise ratios (SNRs). In all our experiments, the correlation coefficient between the original and decrypted images (c_r) is used to measure the closeness of the decrypted watermark to the original one. Figures 4 and 5 show the original logo and cameraman images, respectively, with their encrypted versions.

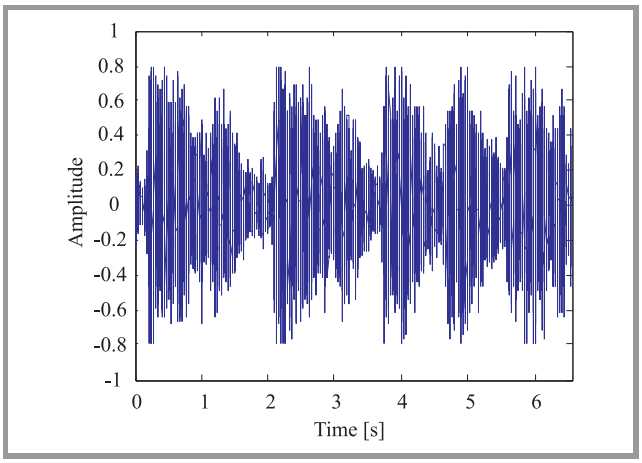


Fig. 3. Waveform of the Handel audio signal.

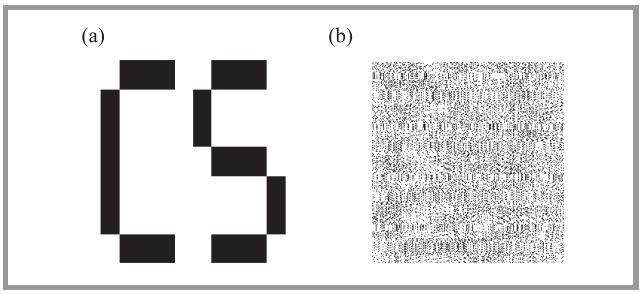


Fig. 4. Logo image: (a) original image; (b) encrypted version.

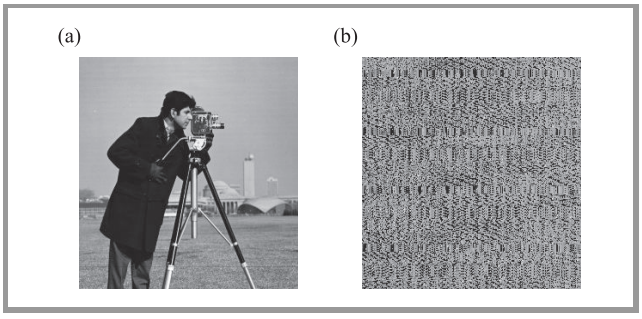


Fig. 5. Cameraman image: (a) original image; (b) encrypted version.

In the first simulation experiment, the logo image is used as a watermark. The SVD audio watermarking method has been used for watermark embedding without encryption. The watermarked audio signal has been transmitted over an uncorrelated fading channel and the results are shown in Fig. 6 at $SNR = 20$ dB. It is clear from these results that the SVD audio watermarking does not degrade the quality of the watermarked audio signal. It is also clear that

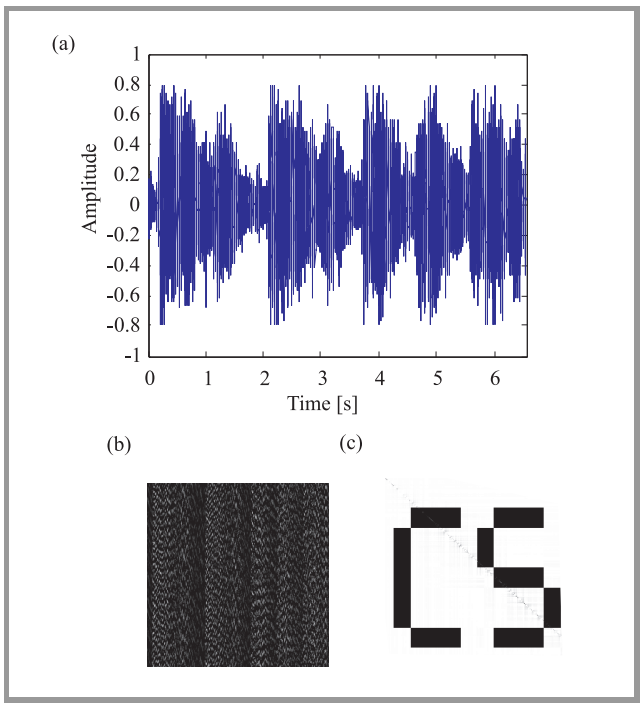


Fig. 6. Transmission of the watermarked audio signal over an uncorrelated fading channel at $SNR = 20$ dB: (a) received audio signal; (b) 2-D watermarked matrix; (c) the extracted logo image, $c_r = 0.41$.

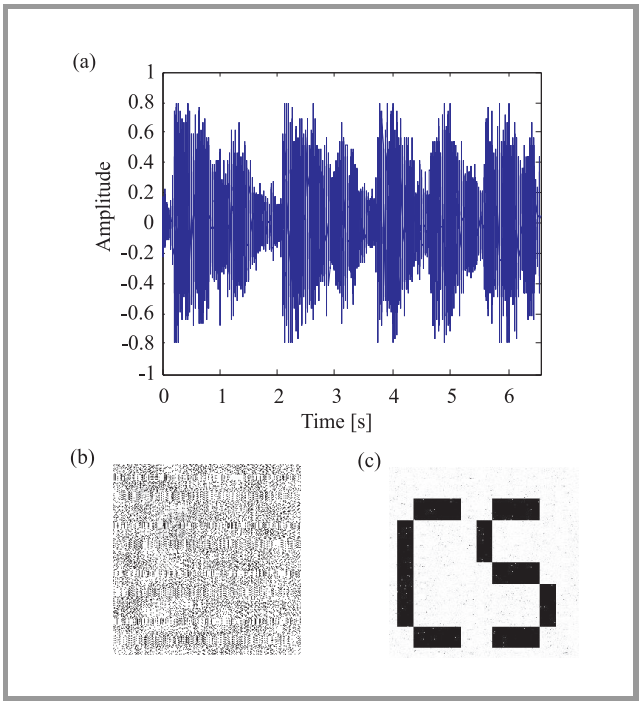


Fig. 8. Transmission of the watermarked audio signal with an encrypted watermark over an uncorrelated fading channel at $SNR = 20$ dB: (a) received audio signal; (b) 2-D watermarked matrix; (c) the extracted logo image, $c_r = 0.7$.

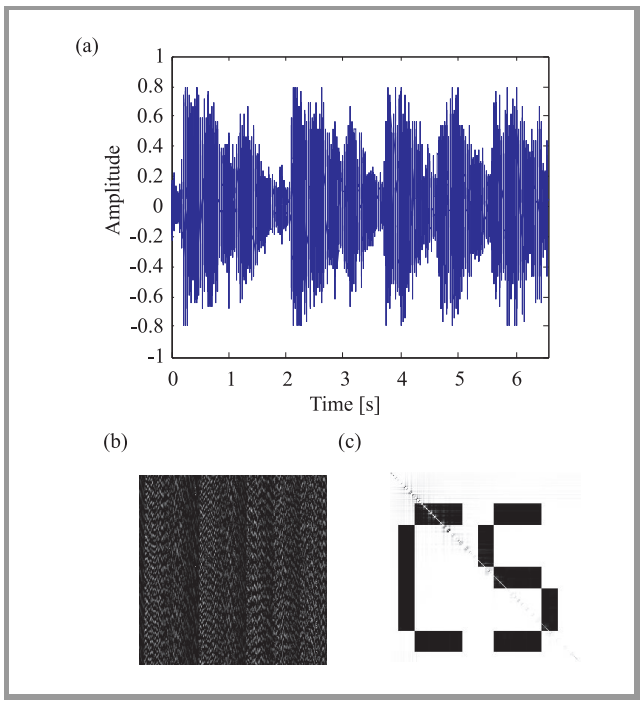


Fig. 7. Transmission of the watermarked audio signal over an uncorrelated fading channel at $SNR = 30$ dB: (a) received audio signal; (b) 2-D watermarked matrix; (c) the extracted logo image, $c_r = 0.86$.

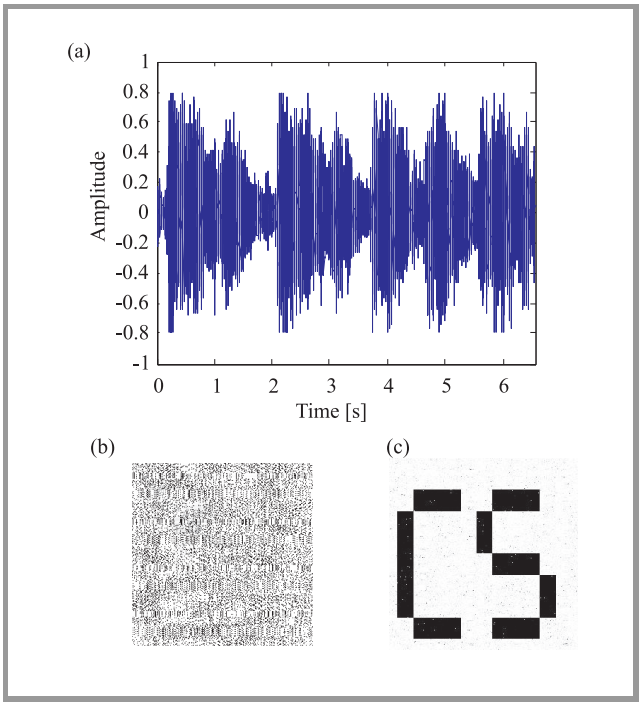


Fig. 9. Transmission of the watermarked audio signal with an encrypted watermark over an uncorrelated fading channel at $SNR = 30$ dB: (a) received audio signal; (b) the watermarked signal; (c) the extracted logo image, $c_r = 0.95$.

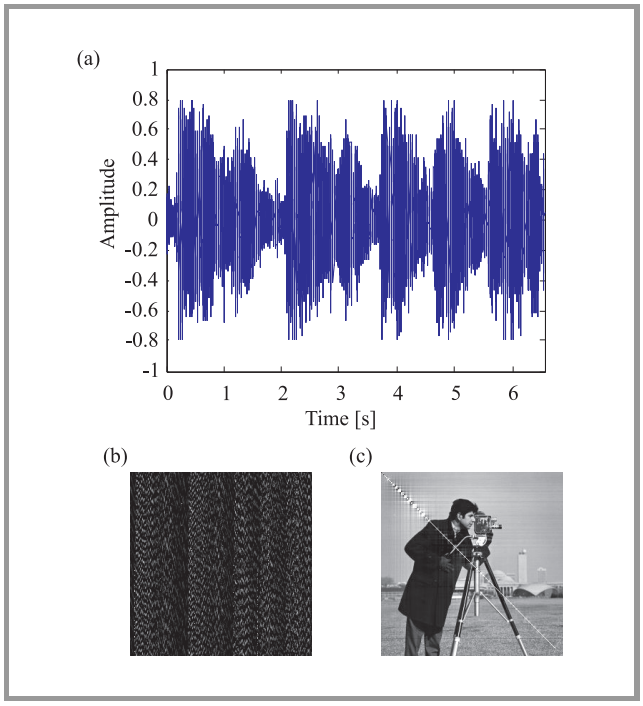


Fig. 10. Transmission of the watermarked audio signal over an uncorrelated fading channel at $SNR = 20$ dB: (a) received audio signal; (b) 2-D watermarked matrix; (c) the extracted cameraman image, $c_r = 0.31$.

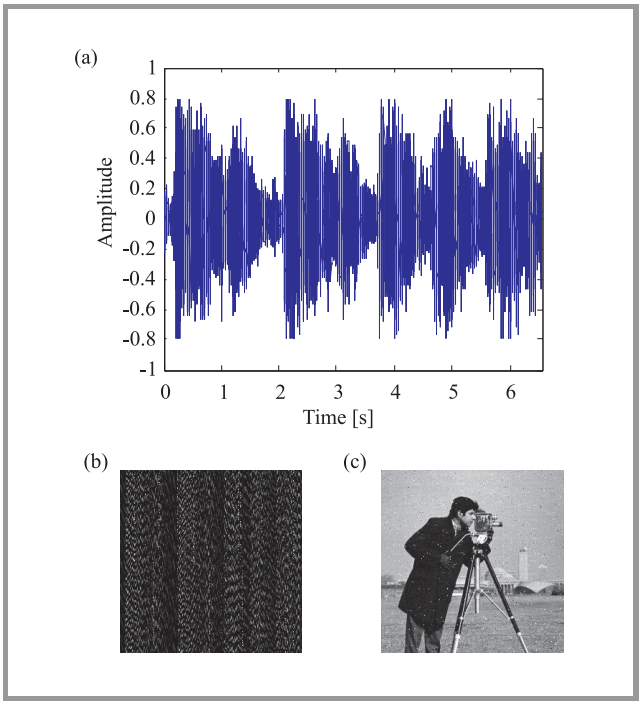


Fig. 12. Transmission of the watermarked audio signal with an encrypted watermark over an uncorrelated fading channel at $SNR = 20$ dB: (a) received audio signal; (b) 2-D watermarked matrix; (c) the extracted cameraman image, $c_r = 0.33$.

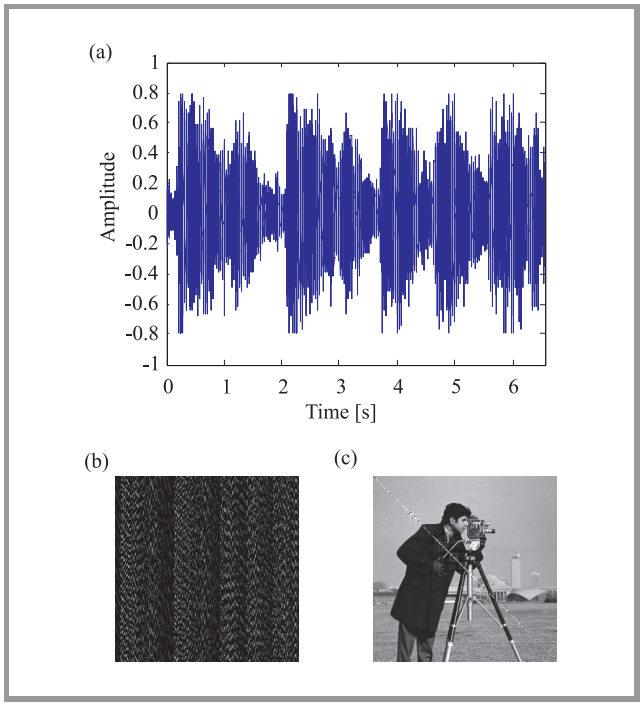


Fig. 11. Transmission of the watermarked audio signal over an uncorrelated fading channel at $SNR = 30$ dB: (a) received audio signal; (b) 2-D watermarked matrix; (c) the extracted cameraman image, $c_r = 0.92$.

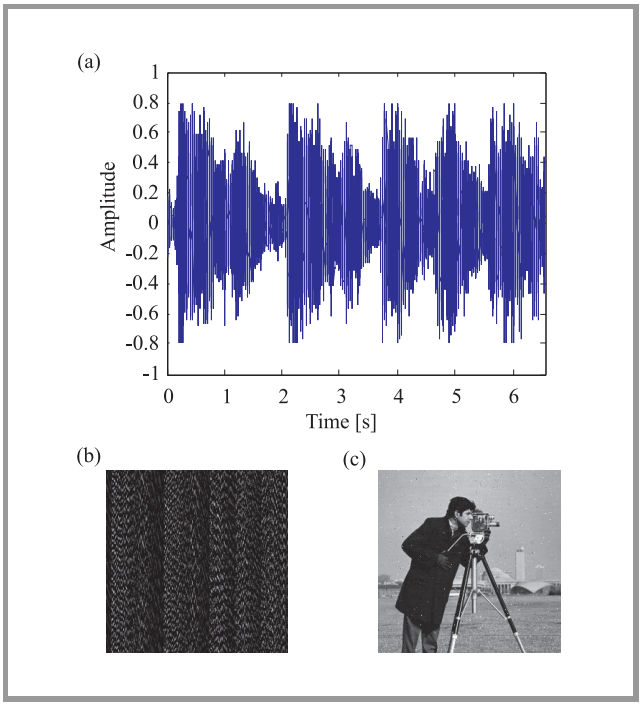


Fig. 13. Transmission of the watermarked audio signal with an encrypted watermark over an uncorrelated fading channel at $SNR = 30$ dB: (a) received audio signal; (b) the watermarked matrix; (c) the extracted cameraman image, $c_r = 0.96$.

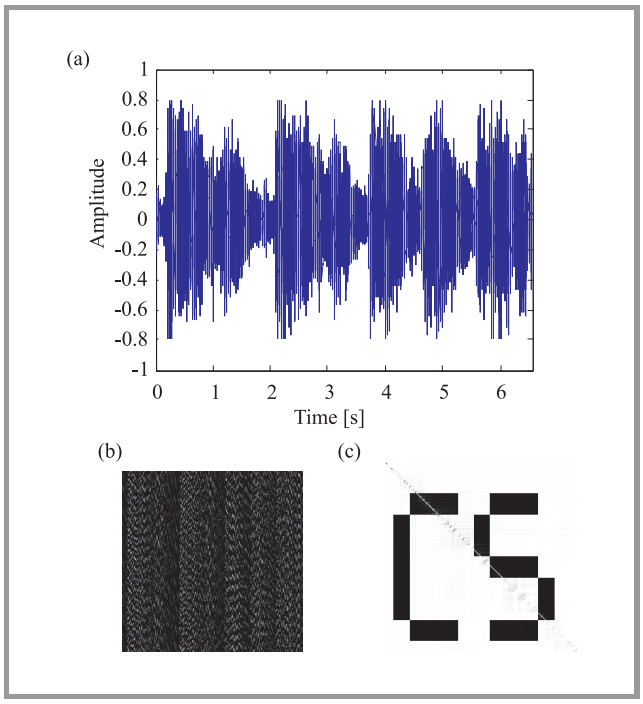


Fig. 14. Transmission of the watermarked audio signal over a correlated fading channel at $SNR = 25$ dB: (a) received audio signal; (b) 2-D watermarked matrix; (c) the extracted logo image, $c_r = 0.72$.

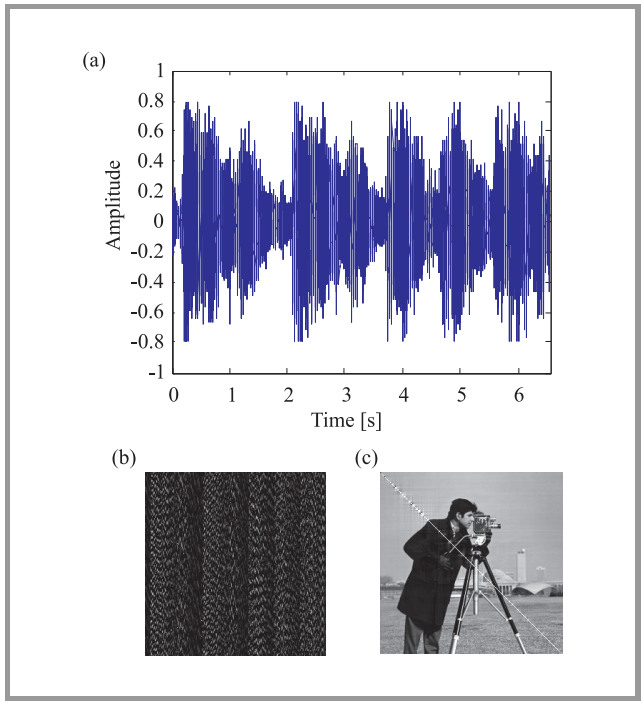


Fig. 16. Transmission of the watermarked audio signal over a correlated fading channel at $SNR = 25$ dB: (a) received audio signal; (b) 2-D watermarked matrix; (c) the extracted cameraman image, $c_r = 0.5$.

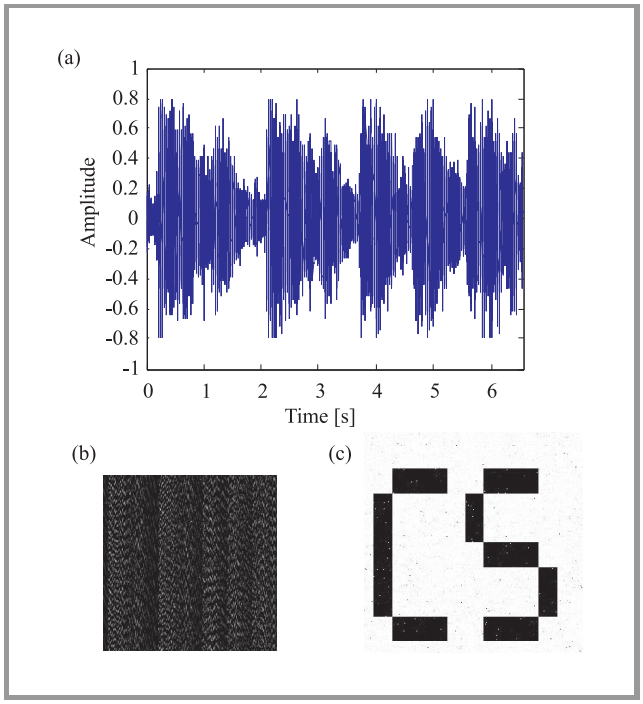


Fig. 15. Transmission of the watermarked audio signal with an encrypted watermark over a correlated fading channel at $SNR = 25$ dB: (a) received audio signal; (b) 2-D watermarked matrix; (c) the extracted logo image, $c_r = 0.75$.

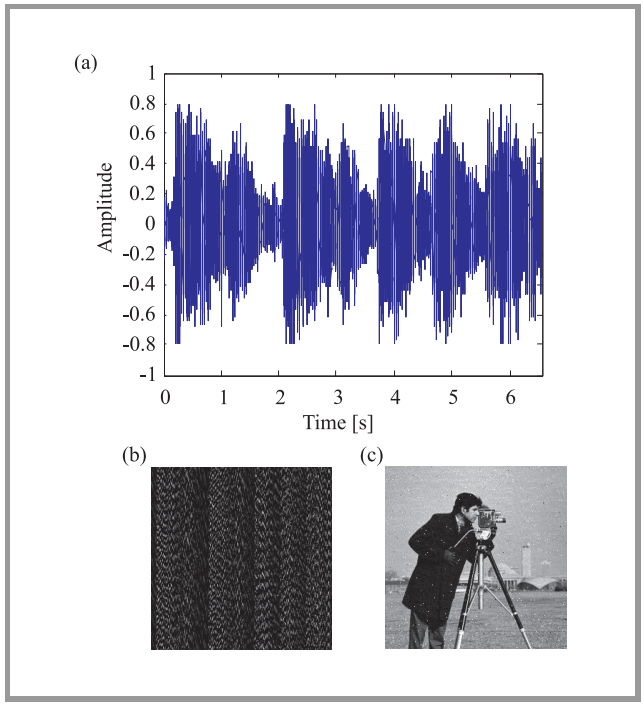


Fig. 17. Transmission of the watermarked audio signal with an encrypted watermark over an correlated fading channel at $SNR = 25$ dB: (a) received audio signal; (b) 2-D watermarked matrix; (c) the extracted cameraman image, $c_r = 0.56$.

the watermark has been reconstructed with an acceptable correlation coefficient. A similar experiment has been carried out at $SNR = 30$ dB. The results of this experiment are given in Fig. 7. As shown in these results the received audio signal and the extracted image are enhanced with the increase of the channel SNR.

Similar experiments have been carried out with encrypted watermarks. Figures 8 and 9 show the results of these experiments at $SNR = 20$ dB, and 30 dB, respectively. These results reveal that encryption enhances the quality of the extracted watermark.

In the following experiments, the cameraman image is used as the watermark. The cameraman watermark has been transmitted in the audio signal over an uncorrelated fading channel without encryption at $SNR = 20$ dB and 30 dB. The results of these experiments are given in Figs. 10 and 11, respectively.

The encrypted cameraman image has also been used in another experiment as a watermark. The results of this experiment over an uncorrelated fading channel are given in Figs. 12 and 13 at $SNR = 20$ dB and 30 dB, respectively.

After studying the performance of the SVD audio watermarking technique with the ZigBee network over an uncorrelated fading channel, the following experiments will study the performance of this method over a correlated fading channel with the Jakes' model. The results of these experiments at $SNR = 25$ dB are shown in Figs. 14 to 17.

All previous results reveal the robustness of the SVD audio watermarking method in the transmission of images over the ZigBee network. Also, the results reveal the effectiveness of chaotic encryption to increase the security and to improve the performance of ZigBee networks in image communication.

6. Conclusions

This paper presented an efficient method for image communication with ZigBee networks. This method is based on data hiding with SVD audio watermarking. Experimental results have proved that watermark embedding with the SVD audio watermarking method does not deteriorate the audio signals. It is clear through experiments that the chaotic encryption enhances the performance of the ZigBee network and increases the level of security.

References

[1] G. Pekhteryev, Z. Sahinoglu, P. Orlik, and G. Bhatti, "Image transmission over IEEE 802.15.4 and ZigBee Networks", in *Proc. IEEE ISCAS*, Kobe, Japan, May 2005.

[2] B. Macq, J. Dittmann, and E. J. Delp, "Benchmarking of image watermarking algorithms for digital rights management", *Proc. of the IEEE*, vol. 92, no. 6, pp. 971–984, 2004.

[3] Z. M. Lu, D. G. Xu, and S. H. Sun, "Multipurpose image watermarking algorithm based on multistage vector quantization", *IEEE Trans. Image Process.*, vol. 14, no. 6, pp. 822–831, 2005.

[4] H. S. Kim and H. K. Lee, "Invariant image watermark using zernike moments", *IEEE Trans. Circ. Syst. Video Technol.*, vol. 13, no. 8, pp. 766–775, 2003.

[5] W. C. Chu, " DCT-based image watermarking using subsampling", *IEEE Trans. Multimedia*, vol. 5, no. 1, pp. 34–38, 2003.

[6] L. Ghouti, A. Bouridane, M. K. Ibrahim, and Said Boussakta, "Digital image watermarking using balanced multiwavelets", *IEEE Trans. Signal Process.*, vol. 54, no. 4, pp. 1519–1536, 2006.

[7] S. Xiang and J. Huang, "Histogram-based audio watermarking against time-scale modification and cropping attacks", *IEEE Trans. Multimedia*, vol. 9, no. 7, pp. 1357–1372, 2007.

[8] Z. Liu and A. Inoue, "Audio watermarking techniques using sinusoidal patterns based on pseudorandom sequences", *IEEE Trans. Circ. Systems Video Technol.*, vol. 13, no. 8, pp. 801–812, 2003.

[9] A. N. Lemma, J. Aprea, W. Oomen, and L. V. de Kerkhof, "A temporal domain audio watermarking technique", *IEEE Trans. Signal Process.*, vol. 51, no. 4, pp. 1088–1097, 2003.

[10] M. A. M. El-Bendary, A. E. Abou-El-Azzm, N. A. El-Fishawy, M. El-Tokhy, F. Shawki, F. E. Abd-El-Samie, and H. B. Kazemian, "An efficient chaotic interleaver for image transmission over IEEE 802.15.4 Zigbee network", *J. Telelecom. Inform. Technol. JTIT*, vol. 2, no. 1, pp. 67–73, 2011.

[11] Q. He, Q. Qi, Y. Zhao, W. Huang, anf Q. Huang, "The application of chaotic encryption in industrial control based on ZigBee wireless network", in *Proc. Int. Symp. Systems Control Aerosp. Astronaut. ISSCAA 2008*, Shenzhen, China, 2008.

[12] M. Ise, Y. Ogasahara, K. Watanabe, M. Hatanaka, T. Onoye, H. Niwamoto, I. Keshi, and I. Shirakawa, "Design and implementation of home network protocol for appliance control based on IEEE 802.15.4", *International J. Comput. Sci. Netw. Secur.*, vol. 7 no. 7, July 2007.

[13] B. P. Lathi, *Modern Digital and Analog Communication Systems*. 2nd edition. Philadelphia: Holt, Rinehart and Winston, 1989.

[14] R. Liu and T. Tan, "An SVD-based watermarking scheme for protecting rightful ownership", *IEEE Trans. Multimedia*, vol. 4, no. 1, pp. 121–128, March 2002.

[15] X. Sun, J. Liu, J. Sun, Q. Zhang and W. Ji, " A robust image watermarking scheme based-on the relationship of SVD", in *Proc. Int. Conf. Intelligent Inf. Hiding and Multimedia Sig. Process.*, Harbin, China, 2008.

[16] X. Zhu, J. Zhao and H. Xu, "A digital watermarking algorithm and implementation based on improved SVD", in *Proc. 18th Int. Conf. Pattern Recognition ICPR'06*, Hong Kong, China, 2006.

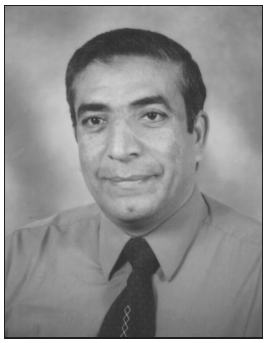
[17] W. C. Jakes, *Microwave Mobile Communications*. New York: Wiley, 1975.

[18] J. Aldrich, "Correlations genuine and spurious in Pearson and Yule", *Statist. Sci.*, vol. 10, no. 4, pp. 364–376, 1995.

Mohsen A. M. Mohamed El-Bendary received his B.Sc. in 1998, M.Sc. in 2008, all in Communication Engineering, from Menoufia University, Faculty of Electronic Engineering. He is now a lecturer assistant and Ph.D. student. His research interests cover wireless networks, wireless technology, channel coding, QoS over

Bluetooth system and Wireless Sensor Network (WSN), and security systems which use wireless technology, such as fire alarm and access control systems.

E-mail: mohsenbendary@yahoo.com
Faculty of Industrial Education
Department of Communication Technology
Helwan University, Egypt



Atef Abou El-Azm was born in 1954, Egypt. He has the B.Sc. in Electronic Engineering and M.Sc. in Antennas from Faculty of Electronic Engineering, Menofia University in 1977 and 1984, respectively. He has the Ph.D. in Communications from Warsaw University of Technology, Poland, in 1990. His research is in the area of digital communications, with special emphasis on coding theory, information theory, error control coding, and coded modulation. Topics of current interest include the use of convolutional, trellis codes and turbo codes in the development of coding standards for high speed data modems, digital satellite communication, and deep space channels. He is the author of many papers in the field of line codes, channel codes and signal processing.

E-mail: abouelazm_atef@yahoo.com
Faculty of Electronic Engineering
Department of Electronics and Electrical Communications
Menoufia University
Menouf, 32952, Egypt

Nawal El-Fishawy received the Ph.D. degree in Mobile Communications from the Faculty of Electronic Engineering, Menoufia University, Menouf, Egypt, in collaboration with Southampton University in 1991. Now she is the head of Computer Science and Engineering Department, Faculty of Electronic Eng. Her research interest includes computer communication networks with emphasis on protocol design, traffic modeling and performance evaluation of broadband networks and multiple access control protocols for wireless communications systems and networks. Now she directed her research interests to the developments of security over wireless communications networks (mobile communications, WLAN, Bluetooth), VOIP, and encryption algorithms.

E-mail: nelfishawy@hotmail.com
Faculty of Electronic Engineering
Department of Electronics and Electrical Communications
Menoufia University
Menouf, 32952, Egypt

Farid Shawki M. Al-Hosarey received the B.Sc. (Hons), M.Sc., and Ph.D. degrees from the Faculty of Electronic Engineering, Menoufia University, Menouf, Egypt, in 1995, 2000 and 2007, respectively. In 1995 and 2000, he worked as a demonstrator and assistant lecturer in the Department of Electronics and Electrical Communications, Faculty of Electronic Engineering respectively. He joined the teaching staff of the Department of Electronics and Electrical Communications since 2008. His current research areas of interest include channel coding, mobile communication systems, MIMO systems, and implementation of digital communications systems using FPGA.

E-mail: farid_shawki@yahoo.com
Faculty of Electronic Engineering
Department of Electronics and Electrical Communications
Menoufia University
Menouf, 32952, Egypt

Mostafa A. R. El-Tokhy was born in Kaluobia, Egypt, in 1970. He received his B.Sc. degree from Zagazig University, Banha branch, Egypt, and M.Sc. degree from Technical University, Eindhoven, The Netherlands in 1993 and 1998, respectively. He received his Ph.D. degree from Osaka University, Osaka, Japan in 2003. Presently, he is an Assistant Professor of Electronics Engineering at Industrial Education College, Higher Ministry of Education, Cairo Egypt. His current research interests are high performance digital circuits and analog circuits. He is a member of the IEEE.

E-mail: mostafaeltokhy@hotmail.com
Faculty of Industrial Education
Department of Communication Technology
Helwan University, Egypt

Fathi E. Abd El-Samie received the B.Sc. (Honors), M.Sc., and Ph.D. from the Faculty of Electronic Engineering, Menoufia University, Menouf, Egypt, in 1998, 2001, and 2005, respectively. He joined the teaching staff of the Department of Electronics and Electrical Communications, Faculty of Electronic Engineering,

Menoufia University, Menouf, Egypt, in 2005. He is a co-author of about 130 papers in national and international conference proceedings and journals. He has received the most cited paper award from digital signal processing journal for 2008. His current research areas of interest include image enhancement, image restoration, image interpolation, superresolution reconstruction of images, data hiding, multimedia communications, medical image processing, optical signal processing, and digital communications.
E-mail: fathi_sayed@yahoo.com
Faculty of Electronic Engineering
Department of Electronics
and Electrical Communications
Menoufia University
Menouf, 32952, Egypt

H. B. Kazemian received the B.Sc. in Engineering from Oxford Brookes University, UK, the M.Sc. in control systems engineering from the University of East London, UK, and the Ph.D. in learning fuzzy controllers from Queen Mary University of London, UK, in 1985, 1987 and 1998, respectively. He is currently a Full Professor at London Metropolitan University, UK. His research interests include fuzzy and neuro-fuzzy control of networks, 2.4 GHz frequency bands, ATM, video streaming and rate control. He is a senior member of the IEEE, IET, and Chartered Engineer, UK.
E-mail: h.kazemian@londonmet.ac.uk
Intelligent Systems Research Centre
Faculty of Computing, London Metropolitan University
London, UK

Paper

Guaranteed Protection in Survivable WDM Mesh Networks – New ILP Formulations for Link Protection and Path Protection

Baibaswata Mohapatra, Rajendra K. Nagaria, and Sudarshan Tiwari

Motilal Nehru National Institute of Technology, Allahabad, India

Abstract—In this paper we propose new simple integer linear programs (ILPs) formulations for minimizing capacity (in wavelength link) utilization in survivable WDM network. The study examines the performance of shared based protection schemes, such as path protection scheme and link protection scheme under single fiber failure. The numerical results obtained show a reduction in capacity utilization using random traffic compared to the reported ILP formulation. We also present the results using Poisson's traffic to identify the frequently used links for the widely used NSF network. The proposed work not only reduces the wavelength consumption in different traffic scenarios but also efficient in terms of simulation time.

Keywords—Lightpath, Protection, RWA, Survivability, WDM network.

1. Introduction

Wavelength-division-multiplexing (WDM) is a well established technology used in optical network that allows to support high bandwidth application such as HDTV, video conferencing etc. A connection is realized by a lightpath in WDM networks. Efficient routing and wavelength assignment (RWA) algorithms are needed in such networks to enhance the network throughput [1]–[3]. Integer programming solves the above sub-problems (routing and wavelength assignment) simultaneously irrespective of the traffic patterns, such as static or dynamic. In case of static traffic, connections are known in advance and the objective is to minimize the usage of network resources such as, wavelength utilization for a certain number of connections. Whereas in the case of dynamic traffic, connection request comes in random fashion one after another. Since enormous amount of data flows through the optical networks, it is desirable that the network has to provide service in the course of network failure, particularly fiber cut. In order to continue service in the event of failure, protection and restoration techniques are used while designing survivable WDM networks. To ensure network survivability, several techniques are realized by providing some additional capacity within the network. In protection scheme, backup resources (route and wavelength) are reserved during connection setup. On the other hand, in restoration scheme backup resources are computed dynamically for each interrupted connection. Restoration schemes are more efficient in terms of wavelength consumption as they do not reserve wavelengths in advance. Network resources can be dedicated (1+1) protection or shared based (1:N) protection. Shared based schemes offer better capacity utilization at the same time, dedicated based schemes offered better switching time [4], [5].

Survivable WDM optical networks have been exploring new design and optimization techniques. A variety of solutions have been proposed by researchers and the methods are broadly classified into two groups: heuristic method and exact methods (such as ILPs). Heuristic methods provide sub-optimal solutions and require low computational effort. On the other hand exact methods are much more computationally intensive and do not scale well with the network size. However, since the exact methods are able to provide absolute optimal solution it can be used as a direct planning tool or as a standard to validate and test the heuristic methods. The proposed work concerns exact methods to plan and optimize wavelength utilization in survivable WDM networks. In particular we focus on ILP, a widespread technique to solve exact optimization problem.

Previous work. In WDM networks, flow of different commodities corresponds to different lightpaths to be established between the nodes in the network. ILP formulations are presented in [2] for maximizing network throughput hence minimizing blocking probability. The author proposed RWA algorithms and evaluate the performance of the algorithms using uniform and non-uniform traffic patterns. A review on fault management in WDM mesh networks is presented in [6]. The author reviewed various fault management techniques involved in deploying a survivable optical mesh network. Integer linear program formulations are presented in [7] to allocate working and spare capacity in a WDM survivable network. The author considered symmetrical arc-capacity and presented ILP formulations for capacity allocation. Both node-arc and arc-path version of capacity allocation models are presented. In [8] ILP based survivable algorithms are investigated for WDM mesh networks. The author focuses on protection

and restoration schemes to calculate capacity utilization and switching time for survivable schemes such as, dedicated-path, shared-path, and shared-link protection. Restorable survivable network design with static traffic demands is also reported in [4], [9], [10]. Survivable network design is reported in [11], the author propose new routing strategies under single fiber cut scenario so that the virtual topology remains connected. Link restoration in distributed real-time environment, is reported in [12]. The objective is to find spare-capacity consumption, with restricted hop-limits on restoration routes, which are still efficient in terms of utilization of spare resources. The ILP formulation is solved both for mesh and ring topologies. Interesting routing algorithms, network flow problems with optimization techniques are available in [13], [14]. A different version of this work appeared in [15]. Earlier we have reported protection switching time for shared path and shared link protection schemes [16].

Traditional WDM transmission system consists of two fibers and two transceivers, one for forward traffic and other for backward traffic. This is called simplex transmission system. Significant cost savings is possible with so called bi-directional transmission using WDM technology where individual wavelengths used for each direction. In such case linking two nodes will involve only one fiber and this is called full-duplex transmission system. Keeping this thing in mind we have formulated ILPs to realize full-duplex survivable WDM transmission system.

In this study we examine different approaches to protect a mesh-based WDM optical network from single fiber failures. These approaches are based on two basic survivability paradigms: 1) path protection and 2) link protection. The summary of the work is as follows:

- New ILP formulations are developed which can provide optimal solution with less simulation time.
- The ILP formulations are developed for realizing shared path protection (SPP) and shared link protection (SLP) schemes against single fiber failure.
- Joint optimization technique is also developed (minimizing primary and backup capacity altogether) to minimize wavelength utilization for SPP scheme.
- To identify the links often used (frequently used/critical links) in the network by applying multiple traffic matrices as data inputs.

The proposed formulations for minimizing the wavelength utilization for a given number of connections request similar to [7], in addition to the wavelength continuity constraint. We have calculated the number of wavelength links utilized assuming random traffic demand. The results obtained are compared with the results obtained using the reported ILP formulations in [8]. Simulations are also carried out by using several traffic patterns to identify the frequently used links in NSF network. The reported ILPs and the proposed ILPs have been simulated in same environment. The rest of the paper is organized as follows. Section 2 presents mathematical models for the working capacity and the spare capacity utilization models for SLP and SPP schemes. In Section 3 we have discussed the simulation setup and the numerical results obtained and the conclusion of the work is given in Section 4.

2. Mathematical Model for Capacity Allocation

In this section, we present the ILP formulation of shared-path protection and shared-link protection scheme to protect WDM network against single fiber failure. In particular we need to calculate number of wavelength links utilized in the above two schemes in survivable WDM networks. Simulations are carried out by using two different kinds of traffic scenario, i.e., random traffic and Poison's distributed multiple traffic matrices. Multiple traffic matrices are used to find the average link utilization. The results obtained using the above schemes are compared to evaluate the performance of the ILPs. Wavelength utilization for a given connection is the sum of the wavelength channels required to establish the primary lightpath and the backup lightpath. The proposed ILP models are solved using CPLEX.

Sets and parameters:

N – set of nodes,
L – set of links,
WP – number of wavelengths used for primary paths,
WB – number of wavelengths used for backup paths,
W – number of wavelengths on each fiber link, $W = (WP + WB)$,
d_{ij} – number of requested connections between node pair i and j, where i and $j \in N$,
D – set of demand pairs $(i, j) \in D$ implies that $d_{ij} > 0$,
P_{ij} – set of paths between node pair i and j, $\forall (i, j) \in D$,
R – set of all paths,
w_p – index of wavelengths used for primary paths,
w_b – index of wavelengths used for backup paths,
A_{ij} – set of paths used link (i, j) directed from i to j,
Z_{st} – set of all paths from 's' to 't' except the direct link (s,t),
V_{ij}^{st} – set of directed paths accessible for restoration from i to j when (s,t) fail,
$\widetilde{V}_{ij}^{st}$ – set of directed paths not accessible for restoration from i to j when (s,t) fail.

Variables:

w_{ij} – working capacity on link (i, j),
s_{ij} – spare capacity on link (i, j),
x_{pw_p} – 1 if lightpath p exists with wavelength w_p , 0 otherwise,
$y_{pw_p}^{st}$ – 1 if lightpath exists on path p with wavelength w_b, when link (s,t) fail, 0 otherwise.

2.1. Working Capacity Allocation Model

The ILP model for allocating working capacity or establishing primary path and assigning a wavelength without survivable provision is presented below. In our model, we consider communication can be made in both directions. Link (i,j) in our model means a fiber link between node i and node j. If a lightpath use link (i,j) and another lightpath use link (j,i) we consider, capacity consumption on link (i,j) is 2, of course with different wavelengths. Instead of considering all possible paths between a pair of nodes, we have restricted the number of hops so as to provide three to four paths between every pair of nodes.

Minimize $\sum_{(i,j)\in L} w_{ij}$.

Wavelength continuity: $\forall(i,j)\in L,\ 1\le w_p\le WP$

$$\sum_{p\in A_{ij}} x_{pw_p}+\sum_{p\in A_{ji}} x_{pw_p}\le 1\,. \tag{1}$$

Demand between node pair: $\forall(i,j)\in D$

$$\sum_{p\in P_{ij}}\sum_{w_p=1}^{WP} x_{pw_p}=d_{ij}\,. \tag{2}$$

Working link capacity: $\forall(i,j)\in L$

$$\sum_{p\in A_{ij}}\sum_{w_p=1}^{WP} x_{pw_p}+\sum_{p\in A_{ji}}\sum_{w_p=1}^{WP} x_{pw_p}\le w_{ij}\,, \tag{3}$$

$$w_{ij}\ge 0(i,j)\in L, \tag{4}$$

$$x_{pw_p}=\{1,0\}\quad p\in R,\ \ 1\le w_p\le WP. \tag{5}$$

Equation (1) ensure that no two lightpaths passes through the link (i,j) has same wavelength. Equation (2) ensure that the number of connection between node pair (i,j) must be equal to the sum of lightpaths within P_{ij} with different wavelengths. Equation (3) says about the number of lightpaths passes through link (i,j) or (j,i). Equations (4) and (5) ensures non negative and binary variables respectively.

2.2. Spare Capacity Allocation Model

WDM networks are meant for high bandwidth applications; hence, fiber cut leads to loss of huge data as well as revenue. To provide a certain quality of service (QoS) to users, we need to guarantee the service in the event of fiber failure. Here we present mathematical models to calculate spare network resources such as, spare wavelengths need to reserve in the event of fiber failure. Since probability of multiple fiber failure is rare, we assume single link failure in our models. Out of the number of protection strategies available [1], [6], we consider shared based protection schemes as a counter part to dedicated based protection schemes. Though the later strategies provide better switching time [8], [16], it consumes high network resources. Shared protection schemes can be implemented in three ways [6]: link protection, path protection and subpath protection. In link protection backup route is discovered around the failed link, whereas, in path protection backup path is discovered from source to destination for any failed link on the primary lightpath. In subpath protection, when a link failure occurred, the upstream nodes of the failed link detect the failure and discover a backup route from itself to the corresponding destination node for each disrupted connection. Link protection and path protection schemes are discussed below.

2.2.1. Shared Link Protection (SLP)

In link based protection, we assumed that each node is capable to detect a link failure and run a rerouting algorithm around the failed link. The traffic is rerouted only around the failed link. In this strategy for calculating backup capacity we should have knowledge about the working capacity or the number of primary lightpaths passes through any link. This is being calculated from the method described before from working capacity allocation model. When any link (i,j) fails, node i looks for searching restoration path for each lightpaths that passes through link (i,j) around (i,j). The ILP model can be written mathematically as follows.

Minimize $\sum_{(i,j)\in L}(w_{ij}+s_{ij})$.

Wavelength continuity for the backup lightpaths: $\forall(i,j)\in L$, $1\le w_b\le WB$

$$\sum_{p\in A_{ij}} x_{pw_b}+\sum_{p\in A_{ji}} x_{pw_b}\le 1. \tag{6}$$

Lost link capacity: $\forall(s,t)\in L$

$$\sum_{p\in Z_{st}}\sum_{w_b=1}^{WB} y^{st}_{sw_b}=w_{st}\,. \tag{7}$$

Spare link capacity: $\forall(s,t)\in L,\ (i,j)\in L\backslash(s,t)$

$$\sum_{p\in A_{ij}}\sum_{w_b=1}^{WB} y^{st}_{pw_b}+\sum_{p\in A_{ij}}\sum_{w_b=1}^{WB} y^{st}_{pw_b}=s_{ij}\,. \tag{8}$$

Fiber capacity limit: $\forall(i,j)\in L$

$$w_{ij}+s_{ij}\le W\,, \tag{9}$$

$$s_{ij}\ge 0(i,j)\in L, \tag{10}$$

$$y^{st}_{pw_b}=\{1,0\}\ \forall(s,t)\in L,\ \ \forall_p\in R,\ \ 1\le w_b\le WB. \tag{11}$$

Where w_{ij} may be calculated from the working capacity model. Equation (6) ensures that no two backup lightpaths can have same wavelength traversing in a link. Equation (7) ensure that the number of lightpaths affected due to failure of fiber link (s,t) where $\forall(s,t)\in L$. The affected lightpaths must be rerouted through all the paths from s to t except the failed link (s,t) with the available backup wavelengths. Equation (8) ensure that the sum of backup lightpaths passes through link (i,j) and (j,i) must be less than

the total number of backup lightpaths. Equation (9) ensures that the sum of the primary lightpaths and the backup lightpaths on any link is limited to the number of wavelengths available on the fiber link. In our work we have considered homogeneous network, i.e., all the fiber links have the same number of wavelengths. Equations (10) and (11) ensure non-negative and binary variables.

2.2.2. Shared Path Protection (SPP)

Failure of a link affects all the primary lightpaths that passes through it. Upon link failure the nodes adjacent to the failed link sends a link fail message to source and destination nodes of all the lightpaths that traverse through it. The wavelengths occupied by the primary lightpaths that traverse through the failed link may be freed. Therefore, path protection scheme consume less spare resources compared to link protection scheme and is verified from Fig. 1 and 2. The ILP model for path protection may be written mathematically as follows.

Minimize $\sum_{(i,j)\in L} (w_{ij} + s_{ij})$.

Wavelength continuity for the backup lightpaths: $\forall (i,j) \in L$, $1 \le w_b \le WB$

$$\sum_{p\in A_{ij}} x_{pw_b} + \sum_{p\in A_{ji}} x_{pw_b} \le 1\,. \tag{12}$$

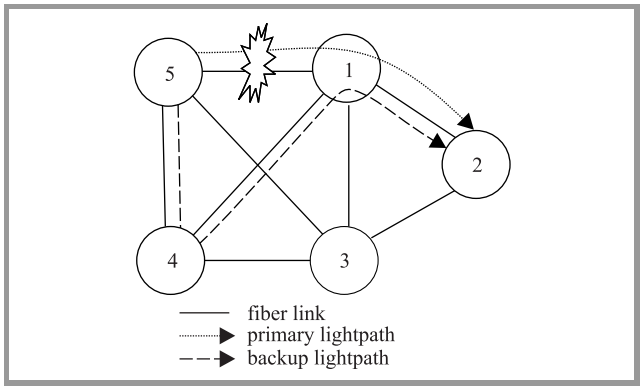


Fig. 1. Link restorration.

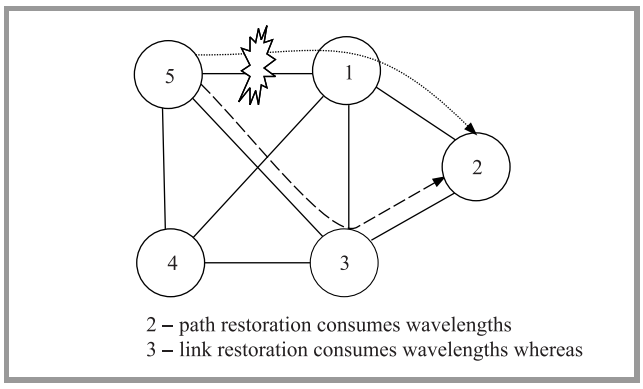


Fig. 2. Path restoration.

Spare link capacity: $\forall (s,t) \in L$, $(i,j) \in L \backslash (s,t)$

$$\sum_{p\in A_{ij}} \sum_{w_b=1}^{WB} y_{pw_b}^{st} + \sum_{p\in A_{ij}} \sum_{w_b=1}^{WB} y_{pw_b}^{st} \le s_{ij}\,. \tag{13}$$

Lost capacity in forward direction: $\forall (s,t) \in L$, $\forall (i,j) \in D$

$$\sum_{p\in v_{ij}^{st}} y_{pw_b}^{st} \sum_{p\in \widetilde{v}_{ij}^{st}} x_{pw_p}\ 1 \le w_p \le WP,\ \ 1 \le w_b \le WB\,. \tag{14}$$

Lost capacity in reverse direction: $\forall (s,t) \in L$, $\forall (i,j) \in D$

$$\sum_{p\in v_{ji}^{st}} y_{pw_b}^{st} \sum_{p\in \widetilde{v}_{ji}^{st}} x_{pw_p}\ 1 \le w_p \le WP,\ \ 1 \le w_b \le WB\,. \tag{15}$$

Fiber capacity limit: $\forall (i,j) \in L$

$$w_{ij} + s_{ij} \le W\,, \tag{16}$$

$$s_{ij} \ge 0\,(i,j) \in L\,, \tag{17}$$

$$y_{pw_b}^{st} = \{1,0\}\ \forall (s,t) \in L,\ \ \forall p \in R,\ \ 1 \le w_b \le WB\,. \tag{18}$$

Where w_{ij} may be calculated from working capacity model explained before. Equation (12) ensures that no two backup lightpaths can have same wavelength traversing in a link. Equation (13) ensure that the sum of backup lightpaths passes through link (i,j) and (j,i) must be less than the total number of backup lightpaths. Equation (14) and (15) ensures that the number of lightpaths affected from (i,j) and (j,i) respectively when link (s,t) fails. Equation (16) ensures that the sum of primary lightpaths and backup lightpaths on any link is limited to the number of wavelengths available on that fiber link. Equations (17) and (18) ensure non negative and binary variables.

2.2.3. Joint Model for Shared Path Protection

In joint mode instead of calculating working and spare capacity separately the combined working and backup capacity is minimized jointly.

Minimize $\sum_{(i,j)\in L} (w_{ij} + s_{ij})$.

Equations (1)–(5) and Eqs. (12)–(18).

3. Results and Discussion

In this section, we present the details of our proposed models implemented in AMPL language with CPLEX 10.2 to solve the ILPs for simulation. Primary and backup lightpaths are established for link protection and path protection schemes. We focus our work on single fiber cut, which is the predominant form of failures in WDM networks. Backup lightpaths are derived for the above schemes under single link failure assumption. Models presented in this paper runs for 500 different traffic matrixes, which are generated, using Poissons process, then percentage link utilization is presented for different links in the entire network

in graphical manner. We assume that, wavelength converters are not available at the network nodes; hence the connection between the source and the destination is established with a single wavelength throughout the path it traverses. *k*-shortest path algorithm is implemented using C language to generate three successive shortest paths between origin-destination pairs. Simulations are carried out assuming 32 wavelengths in each link in both directions on NSF Net [5] having 14 nodes, shown in Fig. 3. First

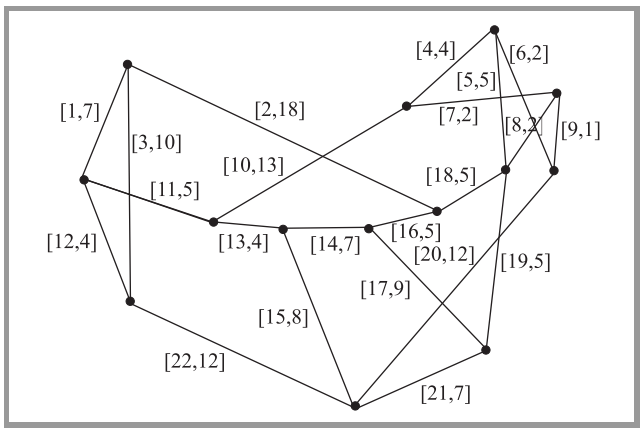


Fig. 3. NSF Net (14 Node).

number in the brackets adjacent to the links represents kth link and second number represents the corresponding link weight. The results are obtained by taking 276 paths in the network among the node pairs and the number of connections varies from 64 to 118 connections in the entire networks.

We have tabulated the results obtained for the proposed ILPs for SLP and SPP model in Table 1 and 2 respectively. The first column represents the number of connections for the random demand matrix. The second column represents the wavelength links consumption for the existing ILPs [8] and the third column represents the wavelength links consumption for the proposed ILPs. The results for the existing ILPs presented here are obtained in 12 hrs where as our method gives results less than 2 min of simulation times. The significant reduction in simulation time is achieved because in the existing ILP formulations, the wavelength of the primary lightpaths and the backup lightpaths is unchanged. On the other hand they are different in the proposed models. Though it is required to be same in before and failure, we believe that, when it is required to calculate the wavelength/capacity consumption for certain number of connections then the proposed model is much better than the available ILPs. A comparative result for wavelength links consumption is also presented in Table 3 among the proposed SLP, SPP and joint formulation for SPP schemes. It has been observed from the Table 3 that the total capacity requirement is still reduced in case of joint formulation at the cost of higher working capacity and simulation time.

Table 1
Wavelength utilization using SLP scheme for random demand/connections for NSF Net with 32 wavelengths

Demand/ connections	Wavelengths utilization reported SLP scheme	Wavelengths utilization proposed SLP scheme
40	213	149
45	243	181
50	266	193
55	309	211
60	339	230
65	353	249

Table 2
Wavelength utilization using SPP scheme for random demand/connections for NSF Net with 32 wavelengths

Demand/ connections	Wavelengths utilization reported SLP scheme	Wavelengths utilization proposed SLP scheme
40	142	130
45	162	150
50	177	159
55	218	180
60	214	196
65	229	211

Table 3
Wavelength utilization among SLP, SPP and JMSPP schemes with random demand/connections for NSF Net with 32 wavelengths

Connections	SLP scheme	SPP scheme	Joint model for SPP
40	149	130	125
45	181	150	146
50	193	159	158
55	211	180	177
60	230	196	190
65	249	211	206

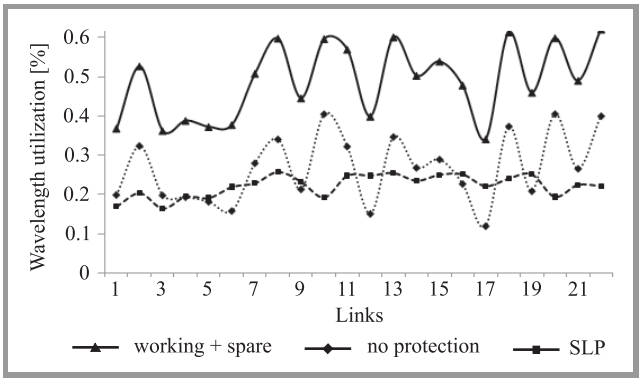


Fig. 4. Percentage of wavelength utilization in different links without and with protection (SLP).

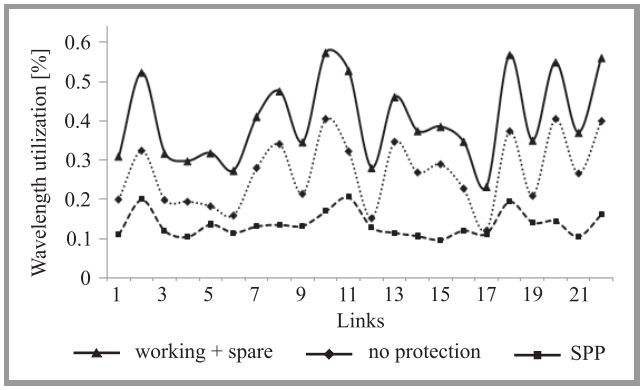


Fig. 5. Percentage of wavelength utilization in different links without and with protection (SPP) scheme.

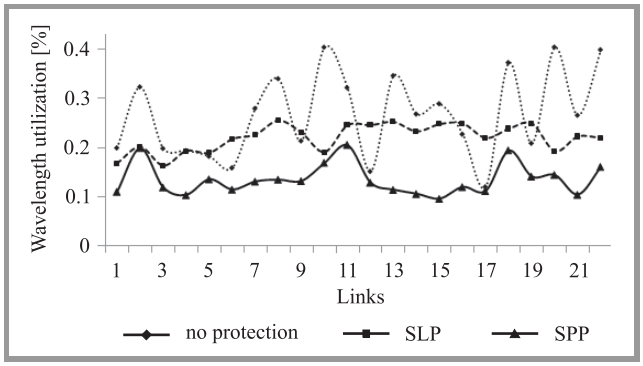


Fig. 6. Percentage of wavelength utilization in different links without and with protection (SLP and SPP) schemes.

For the second set of results, we present the percentage of wavelength link utilization in the existing infrastructure. Instead of random traffic (tabulated in Table 1, 2, and 3) we apply multiple traffic matrices (500) and then averaged to detect the fiber links which are used frequently for routing the lightpaths. SLP and SPP protection schemes comparison are presented in graphical manner in Figs. 4, 5 and 6. From the figures it is found that links numbered 10, 18, 20 and 22 are mostly used for lightpath routing. From Figs. 4 and 5 it is confirm that links numbered 8, 18, 20 and 22 are the links often used so as to provide guaranteed protection. From Fig. 6 indicates that the SPP scheme requires smaller amount of backup capacity (wavelength links) compared to SLP scheme.

4. Conclusions

Survivability is an essential and challenging issue in high speed networks. This work examine the wavelength requirement for guaranteed protection by taking path based and link based protection strategies in WDM networks. Wavelength utilization is considered in this paper which is an important network resource in such networks. The proposed models not only reduce the wavelength consumption in different traffic scenario but also efficient in terms of simulation time. The works also identify the crucial links in the network which is widely used. Further our investigations are in progress to develop efficient algorithms to reduce network capacity utilization at the same time network congestion, for survivable WDM mesh networks.

Acknowledgment

The authors are thankful to the anonymous reviewers for their precious suggestions to improve the manuscript.

References

[1] R. Ramaswami and K. N. Sivarajan, *Optical Networks: A Practical Perspective*. Morgan Kaufmann Publishers, 2002.

[2] M. Shiva Kumar and P. Sreenivasa Kumar, "Static lightpath establishment in WDM Networks – new ILP formulations and heuristic algorithms", *J. Comput. Commun.*, vol. 25, no. 1, pp. 109–114, 2002.

[3] B. Jaumard, C. Meyer, and B. Thiongane, "Comparison of ILP formulations for the RWA problem", *J. Opt. Switch. Netw.*, vol. 4, no. 3–4, pp. 157–172, 2007.

[4] J. Zhang and B. Mukherjee, "A review of fault management in WDM mesh networks: basic concepts and research challenges", *IEEE Netw.*, pp. 41–48, March/April 2004.

[5] J. L. Kennington, E. V. Olinick, and G. Spiride, "Basic mathematical programming models for capacity allocation in mesh-based survivable networks", *J. Manag. Sci.*, vol. 35, no. 6, pp. 629–644, 2007.

[6] S. Ramamurthy, L. Sahasrabuddhe, and B. Mukherjee, "Survivable WDM mesh networks", *J. Lightwave Tech.*, vol. 21, no. 4, pp. 869–883, 2003.

[7] T. Doshi, S. Dravida, P. Harshavardhana, O. Hauser, and Y. Wang, "Optical network design and restoration", *Bell Labs Techn. J.*, pp. 58–84, January-March 1999.

[8] M. Sridharan, M. V. Salapaka, and A. K. Somani, "Operating mesh survivable WDM transport networks", in *Proc. SPIE Int. Symp. Terabit Opt. Netw.: Architecture, Control, and Management Issues*, Boston, USA, 2000, pp. 113–123.

[9] M. Sridharan, A. K. Somani, and M. V. Salapaka, "Approaches for capacity and revenue optimization in survivable WDM networks", *J. High Speed Netw.*, vol. 10, no. 2, pp. 109–125, 2001.

[10] E. Modiano and A. Narula-Tam, "Survivable lightpath routing: a new approach to the design of WDM-based networks", *IEEE J. Selec. Areas in Commun.*, vol. 20, no. 4, pp. 800–809, 2002.

[11] M. Herzberg, S. Bye, and A. Utano, "The hop-limit approach for spare-capacity assignment in survivable networks", *IEEE/ACM Trans. Netw.*, vol. 3, no. 6, pp. 775–84, 1995.

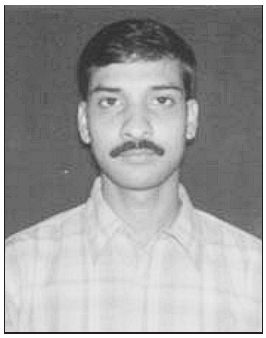
[12] W. D. Grover, *Mesh-based Survivable Networks: Options and Strategies for Optical, MPLS, SONET and ATM Networking.* Upper Saddle River: Prentice Hall, 2003.

[13] D. Medhi and K. Ramaswamy, *Network Routing: Algorithms, Protocols, and Architectures.* Morgan Kaufmann Publishers, 2007.

[14] B. Mohapatra, R. K. Nagaria and S. Tiwari, "Link utilization in survivable WDM mesh network", in *Proc. IEEE ICCCT 2010*, Allahabad, India, 2010.

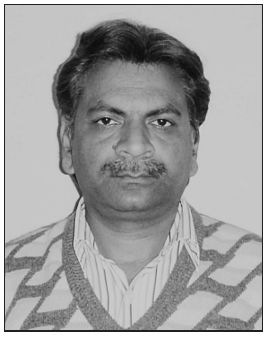
[15] H. K. Singh *et al.*, "Performance comparison of protection strategies in WDM mesh networks", *J. Telecommun. Inform. Technol.*, no. 1, pp. 62–69, 2010.

[16] G. Maier *et al.*, "Optical network survivability: protection techniques in the WDM layer", *J. Photonic Netw. Commun.*, vol. 4, no. 3–4, pp. 251–269, 2002.



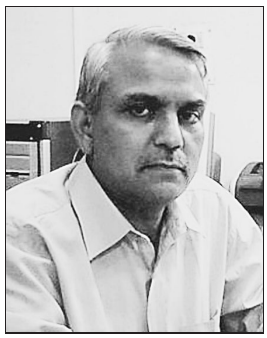
Baibaswata Mohapatra received the B.Eng. degree in Electronics and Telecommunication Engineering from Utkal University, Bhubaneswar, India in 1999, the M.Tech. degree in Electronics and Communication Engineering from U.P. Technical University, Lucknow, India, in 2006. He is currently working towards the Ph.D. degree with the M.N. National Institute of Technology, Allahabad, India, since 2007. He is a life member of Indian society of Technical Education (India) and has more than 7 years of teaching and research experience. His research interests include routing and resource optimization in WDM optical networks.

E-mail: writetobm@gmail.com
Department of Electronics
and Communication Engineering
Motilal Nehru National Institute of Technology
Telierganj, Allahabad-211004, India



Rajendra Kumar Nagaria is an Associate Professor of Electronics and Communication Engineering at Motilal Nehru National Institute of Technology (MNNIT), Allahabad, India. He received a B.Tech. and M.Tech. in Electronics Engineering from Kamla Nehru Institute of Technology (KNIT) Sultanpur, India and Ph.D.(Eng.) from Jadavpur University, Kolkata, India. He has been over 22 years of teaching and research experience. He has published more than forty research papers of national and international reputes. His name is enlisted in an exclusive directory Marquis, *Who's Who in the world*. He is also nominated for the award as International Educator of the year 2005, by International Biographical Centre, Cambridge England. He is fellow of professional bodies like Institute of Engineers (India) and Indian Society for Technical Education. He has guided the thesis of many PG students and presently four research scholars are working under his supervision. His area of interest is mixed-mode signal processing, High speed networks/VLSI design.

E-mail: rkn@mnnit.ac.in
Department of Electronics
and Communication Engineering
Motilal Nehru National Institute of Technology
Telierganj, Allahabad-211004, India



Sudarshan Tiwari received the B.Tech. degree in Electronics Engineering from I.T.BHU, Varanasi, India in 1976, the M.Tech. degree in Communication Engineering from the same institution in 1978 and Ph.D. degree in Electronics and Computer Engineering from IIT Roorkee, India in 1993. Presently, he is Professor and Head of Department of Electronics and Communication Engineering. Motilal Nehru National Institute of Technology (MNNIT), Allahabad, India. He has also worked as Dean Research and Consultancy of the institute from June 2006 till June 2008. He has more than 28 years of teaching and research experience in the area of communication engineering and networking. He has supervised a number of M.Tech. and Ph.D. thesis. He has served on the program committee of several seminars, workshops and conferences. He has worked as a reviewer for several conferences and journals both nationally and internationally. He has published over 78 research papers in different journals and conferences. He has served as a visiting professor at Liverpool John Moore's University, Liverpool, UK. He has completed several research projects sponsored by government of India. He is a life member of Institution of Engineers (India) and Indian society of Technical Education (India), he is a member of Institution of Electrical and Electronics Engineers (USA). His current research interest include, in the area of WDM optical networks, wireless ad hoc and sensor networks and next generation networks.

E-mail: stiwari@mnnit.ac.in
Department of Electronics
and Communication Engineering
Motilal Nehru National Institute of Technology
Telierganj, Allahabad-211004, India

Paper

Privacy Preserving and Secure Iris-Based Biometric Authentication for Computer Networks

Przemysław Strzelczyk

Research and Academic Computer Network (NASK), Warsaw, Poland

Abstract—The iris biometrics is considered one of the most accurate and robust methods of the identity verification. The unique iris features of an individual can be presented in a compact binary form which can be easily compared with the reference template to confirm identity. However in contrast to passwords and PINs biometric authentication factors cannot be revoked and changed as they are inherently connected to our characteristics. Once the biometric information is compromised or disclosed it became useless for the purpose of authentication. Therefore there is a need to perform iris features matching without revealing the features itself and the reference template. We propose an extension of the standard iris-based verification protocol which introduces a features and template locking mechanism, which guarantee that no sensitive information is exposed. Presented solutions can be easily integrated into authentication mechanisms used in modern computer networks.

Keywords—authentication, biometrics, iris recognition, privacy preserving.

1. Introduction

Among the biometric verification methods iris recognition is considered one of the most accurate and robust. Iris features can be easily extracted from eye images and they can be efficiently compared. However if the biometric reference template or set of biometric features are disclosed, the whole biometric system becomes useless for an individual, because the biometric information cannot be canceled or revoked as passwords. Therefore there is a need to perform iris features matching without revealing either the biometric data acquired during the verification process or the reference template from the database. In this paper we introduce a privacy preserving system for iris-based biometric identity verification. Firstly we propose a method in which iris biometric templates stored by the authenticator are locked with the keys known only to the data owner. In this scenario even when the database is compromised no private information is exposed. Secondly we introduce a privacy preserving verification protocol, which is based on the secure multi-party computation techniques.

2. Iris Biometrics

An iris is the colored ring around the pupil. Its structure is determined during the fetal development of the eye and remains unchanged. On contrary the color of the iris can change as a result of the variable pigmentation in tissues. The main role of the iris is to control the size of the pupil and adjust the amount of light which enters through the pupil into the eye interior. It is surrounded by the sclera, which is a white area of tissues and blood vessels, and it is covered by a transparent layer called cornea. The whole iris is visible only with eyes wide open, as eyelids and eyelashes usually occludes the lower and upper part of it. The possibility of using the iris to distinguish individuals is over 100 years old, but the first patent for the automated iris biometric system was obtained by Flom and Safir in 1987 [1]. However the most important work in the field of the iris recognition was due to Daugman [2], [3]. He introduced first practically verified methods for iris image segmentation, unique feature extraction and matching, which with slight modifications are used today world widely and which are the reference models for other algorithms.

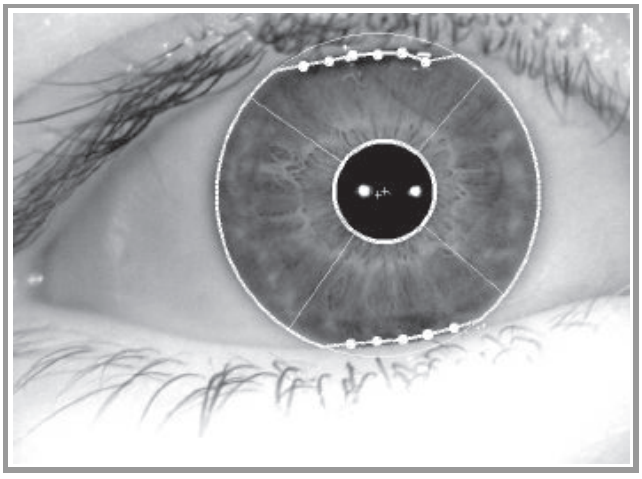


Fig. 1. An example eye image captured in infrared light with outer and inner boundaries approximsted by non-cocentric circles [www.biometriclabs.pl].

For the purpose of biometrics the eye image is captured in the near infrared light with the wavelengths between 700 – 900 nm. Usually special infrared illuminators and bandpass lens filters are used to acquire a image of good quality [4]. The infrared light reveals the detailed structure of the iris better than the visible light [5]. The iris is usually modeled as a ring with outer and inner border approximated by two circles which are nearly concentric.

Figure 1 shows an example image acquired in infrared light with the iris and pupil approximated by two circles. Because the iris area is hardly ever fully visible an additional mask is applied to the ring, which marks the areas where eyelids and eyelashes occlude the iris region. Optionally this mask also indicates image artifacts such as specular reflections from illuminators.

After the iris region is isolated, it is mapped into normalized pseudo-polar coordinate system [5]. Every point of the iris area is described by two coordinates: the angle α and the radial distance r. When the pupil and the iris centre overlap, r is the normalized distance from the iris pupil centre, and α is the angle of the line crossing the point and the pupil centre. Figure 2 shows an iris mapped into pseudo-polar coordinates. The resulting rectangular image has two important properties. Firstly the mapping models linear stretches of the iris, when the pupil contracts or dilates, which is regarded a good approximation of its nature [6]. Secondly the eye or head rotation is equivalent to the permutation of points in the α coordinate, so these effects can be easily compensated.

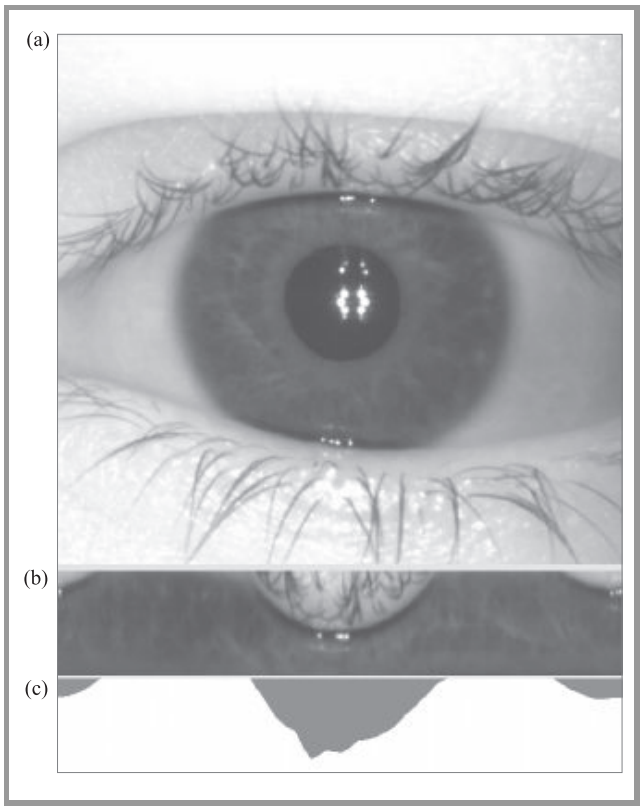


Fig. 2. An example iris image (a), iris image converted to pseudo-polar coordinates (b) and occlusions mask in pseudo-polar coordinates (c).

Most of the iris biometric methods convolve the transformed image with 2-dimensional filters designed to extract the unique iris texture patterns. Many different filters were tested but the best results were obtained for Gabor, Log-Gabor, Haar and Laplacian of Gaussian filters [3], [7]–[11]. The resulting values are re-sampled and quantized. Daugman proposed that only the sign of the filtered signal should be used as features. He introduced a binary iris representation which is summarized in 2048 bits vector, called an "iris code". Many other solutions follow this approach as it allows the iris codes to be compared efficiently using bitwise operations [2].

The iris code matching is based on the Hamming distance. The Hamming distance measures the fraction of corresponding bits of two binary vectors that disagree. Because not all of the bits of the iris code are valid due to occlusions and other disruptions, a modified Hamming distance must be introduced that take this into account. It can be implemented using three types of operations: bitwise XOR ($\oplus$), bitwise AND ($\wedge$) and bit counting as follows:

$$HD(x,w,y,m) = \frac{\sum_i (x_i \oplus y_i) \wedge w_i \wedge m_i}{\sum_i w_i \wedge m_i}, \qquad (1)$$

where x and y are two iris codes and w and m are the occlusions masks, whose bits are set if the corresponding bits of the iris codes are valid. To account for the eye rotation, the Hamming distance is computed for several different permutations of the bits corresponding to the different angles of rotations. At the end the minimal score is compared with the predefined threshold to check if the verification is successful.

3. Preserving Iris Code Privacy

Vernam's cryptographic system, known also as one-time pad is a type of encryption method, which has been proven to be impossible to crack if used properly [12]. The main idea is that the plaintext is encrypted with a substitution cipher using a secret random key. The size of the key must be equal to the size of the plaintext. As long as the key is truly random, and as large as plaintext, the resulting cipher text is also truly random. For the binary data the substitution Vernamñs system can be implemented based on cryptographically secure pseudo-random number generator and XOR operation. The pseudo-random number generator is used to prepare a key, and the XOR operation is used to randomly invert the bit values based on the key bits. One-time pad is a symmetric encryption mechanism, as the XOR operation when used twice with the same key reveals the original plaintext. The main idea behind the privacy preserving iris based verification is that iris codes encrypted with one-time pad can be matched in the same way as unencrypted ones. The Hamming distance of two iris codes before and after encryption remain the same, as long as the keys used to encrypt them are identical. It is because the XOR operation used for Hamming distance computing nullifies the encryption as follows:

$$enc_{\oplus}(x,k) \oplus enc_{\oplus}(y,k) = x \oplus k \oplus y \oplus k = x \oplus y. \qquad (2)$$

The biometric features of the same individual differ each time the biometric measurements are done. For the iris biometrics this variability is mainly due to changing capture conditions and image processing. These processes introduce a noise to the iris code. Whether the noise is a user

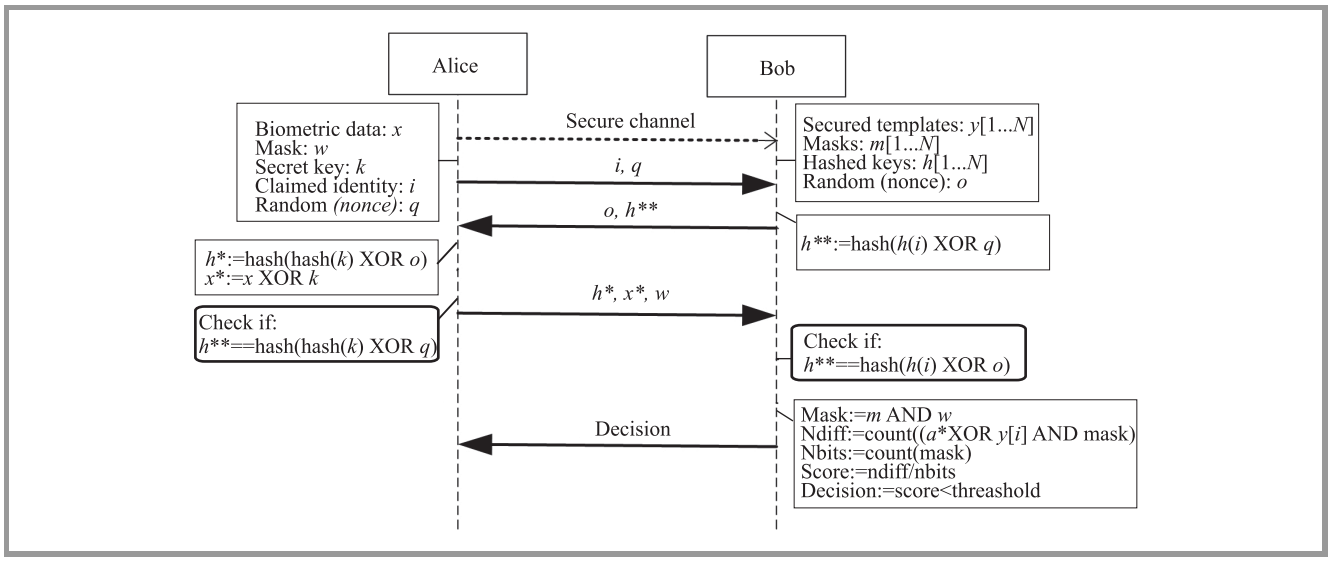


Fig. 3. A sequence diagram for the privacy preserving iris based biometric authentication protocol with two parties

characteristic or not is still an open question. Nevertheless we can assume that for each individual there exists an ideal iris code x and there is a measurement specific noise z. The real iris code can be expressed as a XOR operation between the ideal iris code x and the measurement noise z. In that situation the noise determines the Hamming distance between two iris codes of the same individual.

Now if we use the same cryptographic key in the Vernam's cipher for different real iris codes of the same individual we may reveal the information about the noise but the ideal iris code remains encrypted. The iris code matching routine will not only nullify the encryption but also the ideal iris code. The adversary will be able to determine the character of the noise but will deduce nothing about the ideal iris code.

The presented encryption of iris code has some other interesting features. The experiments indicate that there is only about 100 to 200 bits of information in the iris code in the Shannon sense [13]. This suggests that the bits of iris codes are strongly dependent. The one-time pad encryption removes the bit dependencies in the ciphertexts if the key used has its bits independent. As a result the encrypted iris code will be indistinguishable from random data for an adversary.

4. Two-Party Protocol

The privacy preserving iris based biometric authentication protocol is based on the secure multi-party computation scheme and applies the one-time pad encryption of iris code presented in the previous section.

There are two parties in this process, which we will later call: Alice and Bob. Alice represents a user who is willing to undergo biometric verification. She does not want her biometric data (namely the iris code and the binary occlusions mask) to be disclosed to any other party. Bob represents an authentication server which verifies Alice identity based on her biometric data. Bob has access to the encrypted biometric templates database, which he does not want to disclose to anyone. The authentication protocol presented in this paper employs several proven cryptographic methods. The first one is the cryptographic one-way function called later hash function [14], [15]. It is a deterministic procedure which takes an arbitrary binary vector and transforms it into fixed size bit string, called a hash value. The transformation has some fundamental properties. It is easy to compute the hash, but it is infeasible to recover the binary vector from hash value. It is infeasible to find two binary vectors with the same hash or modify the binary vector without changing its hash. a sample hash function suitable for our solution could be SHA-2. The authentication scheme uses also a secure channel establishment techniques [14], [15]. There are many algorithms which can be used for that purpose. The protocol also requires cryptographically secure pseudo-random number generator (CSPRNG). It is a pseudo-random generator, which has a very long period, and which satisfies the "next-bit test" and withstands "state compromise extensions" [14], [15].

When Alice is enrolled to the biometric system, she is assigned a unique identifier i. Next she uses a trusted biometric device, which captures her iris images and generates her reference iris code $t(i)$ with optional occlusions mask $m(i)$. Then a cryptographically secure pseudo-random number generator creates a secret encryption key k. The key is used to encrypt the reference iris code $y(i) = enc(x(i),k)$. The hash value of the key is computed $h(i) = hash(k)$ and the key is released to Alice. The hash code is needed in the verification process to prove that Bob possesses the encrypted template of Alice. Bob does not have the access to the encryption key. Instead he receives encrypted reference iris code with associated mask and the hash value of the key $\{y(i), m(i), h(i)\}$ and he inserts them into his database, which is indexed with the identification numbers i. Afterwards the system is ready for biometric verification.

The verification process is shown in Fig. 3. When Alice wants to verify herself to Bob she initializes the secure communication channel. Then she generates a random number q and send it securely to the Bob together with her claimed identity i. Bob uses the random value q and the hash value of i-th key $h(i)$ to prove Alice that her template is in the database. Additionally he generates a random number o, which will be used by Alice to prove that she owns the proper key. Bob sends Alice the number o and the proof h^{**}. After the proof is checked by Alice, the biometric system captures her iris image and extract her iris code x with the associated occlusions mask w. The iris code is encrypted with the secret key k and the proof h^{*} based on o is prepared. The encrypted iris code x^{*} with the occlusion mask w and the proof h^{*} are sent to Bob. Bob verifies the proof and calculates the Hamming distance between the encrypted iris code from Alice and the corresponding encrypted template from database (using the bitwise XOR operation, bitwise AND operation and bit counting). If the resulting score is lower than a predefined dissimilarity threshold the verification is considered successful.

5. Conclusions

The presented methods allow to authenticate an individual based on the iris biometrics without revealing the information about the biometric features. These methods use the specific proprieties of the iris code and its matching routines, and employ proven and verified cryptographic techniques. The presented protocol can be incorporated into existing authentication schemes used in the computer networks. The author of this paper is working on the specification and the reference implementation of authentication server and client, which will use the Extensible Authentication Protocol (EAP) family adapted to the presented biometric requirements. Our future work will focus on investigating the iris code noise properties. It is essential for the presented protocol to discover how much individual-specific information is still present in the noise, and what can be done to make the noise more random.

Acknowledgements

This work was co-financed by the Ministry of Science and Higher Education grant OR00 0026 07 “Platform for secure implementation of the biometric systems for verification and identification”.

References

[1] L. Flom and A. Safir, “Iris Recognition System”, February 3, 1987. US Patent 4,641,349.

[2] J. G. Daugman, “High confidence visual recognition of persons by a test of statistical independence”, *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, no. 11, pp. 1148–1161, 1993.

[3] J. G. Daugman, “Biometric Personal Identification System Based on Iris Analysis”, US Patent 5,291,560, March 1, 1994.

[4] A. Czajka and A. Pacut, “Iris recognition with adaptive coding”, *Rough Sets and Knowledge Technology*, *Lecture Notes in Artifical Intelligence*, vol. 4481, Springer, 2007, pp. 195–202.

[5] J. Daugman, “How iris recognition works”, *IEEE Trans. Circ. Sys. Video Technol.*, vol. 14, no. 1, pp. 21–30, 2004.

[6] H. J. Wyatt, “A minimum-wear-and-tear’meshwork for the iris”, *Vision Res.*, vol. 40, no. 16, pp. 2167–2176, 2000.

[7] W. W Boles and B. Boashash, “A human identification technique using images of the iris and wavelet transform”, *IEEE Trans. Signal Process*, vol. 46, no. 4, pp. 1185–1188, 1998.

[8] L. Chenhong and L. Zhaoyang, Efficient iris recognition by computing discriminable textons”, in *Proc. Int. Conf. Neural Netw. Brain ICNN&B’05*, Beijing, China, 2005, vol. 2, pp. 1164–1167

[9] C. T. Chou, S. W. Shih, W. S. Chen, and V. W. Cheng, “Iris recognition with multi-scale edge-type matching”, *Pattern Recogn.*, vol. 4, pp. 545–548, 2006.

[10] J. Thornton, M. Savvides, and B. V. K. Kumar, “An evaluation of iris pattern representations”, in *Proc. First IEEE Int. Conf. Biometrics: Theory, Appl. Systems BTAS 2007*, Washington, DC, USA, 2007, pp. 1–6.

[11] P. Yao, J. Li, X. Ye, Z. Zhuang, and B. Li, “Iris recognition algorithm using modified log-gabor filters”, *Pattern Recogn.*, vol. 4, pp. 461–464, 2006.

[12] C. E. Shannon, “Communication theory of secrecy systems”, *AT&T Bell Labs Tech. J.*, vol. 28, pp. 656–715, 1949.

[13] J. Daugman, “Results from 200 billion iris cross-comparisons”, Tech. Rep. UCAM-CL-TR-635, University of Cambridge, 2005.

[14] B. Schneier, *Applied cryptography*. Wiley, 1996.

[15] P. C. Van Oorschot, A. J. Menezes, and S. A. Vanstone, *Handbook of applied cryptography*. Boca Raton, Fl, USA: Crc Press, 1996.

Przemysław Strzelczyk received his M.Sc. degree in Information Technology in 2005 from the Warsaw University of Technology, and he is currently a Ph.D. candidate in the same field. Since 2005 he is with Reseach and Academic Computer Network (NASK) working for Biometric Laboratories, and within 2007 and 2009 he was research assistant at the Warsaw University of Technology. His main interests include: biometrics, security and methods of artificial intelligence.
e-mail: przemek.strzelczyk@nask.pl
Research and Academic Computer Network (NASK)
Wąwozowa st 18
02-796 Warsaw, Poland

Paper

Trends and Differences of Applying Intelligence to an Agent

Ahmed M. Elmahalawy[a], Moukhtar A. Ali[a], and Hany M. Harb[b]

[a] *Computer Engineering and Science Department, Faculty of Electronic Engineering, Menoufia University, Egypt*
[b] *Computer Engineering and Information System Department, Faculty of Engineering, Al Azhar University, Egypt*

Abstract—The agent technology has recently become one of the most vibrant and fastest growing areas in information technology. On of the most promising characteristics of agent is its intelligence. Intelligent agent is the agent that percepts its environment, collects all information about its environment that it needs, processes these information and then generate proper actions according to these information. This paper discusses trends and differences between two main types of intelligence that can be applied to agent: accumulative intelligence and dynamic intelligence. Accumulative intelligence is discussed with its two perspectives: moment perspective and historical perspective. Auto-vehicle driver is also discussed as an application example of accumulative intelligence. Also, MOSAIC, Mimesis, and MINDY models are reviewed as the pioneering works of dynamic intelligence.

Keywords—accumulative intelligence, dynamic intelligence, intelligent agent.

1. Introduction

Agent technology has received a great deal of attention in last few years and, as a result, wide variety of applications is beginning to get interested in using this technology to develop its own goals. Agent is a software entity that carries out some operations on behalf of a user or another program with some degree of independence and autonomy [1].
New research activities have been proposed to make agent possesses human intelligence characteristics. The concept of intelligence has been addressed by different new research approaches. These approaches have been independently developed in brain science [2], cognitive science [3], psychology [4], and conventional artificial intelligence [5]. They have pointed out that it is necessary for *intelligence* to have a body interacting with an environment which is called *embodiment* approach [6]. This approach insists that the cognitive process should be described with in interaction between both body system and the environment. In contrast, artificial intelligence usually insists that the cognitive process resides only in brain of body.
When agent (like human body) perceives its environment, collects required data knowledge through its sensor, represents this knowledge into predefined patterns (agent experience), analyzes and processes these knowledge patterns to take better decisions, and use these decisions to improve how it carries out its tasks in that environment, this agent is called *intelligent agent* (IA). As a result, applying concept of intelligence to agent [7] gives it the ability to recognize situations it has been in before and improve its performance based on priori experience (i.e., accumulative intelligence). Also, it enables agent to recognize new situations (i.e., situations with no priori experience) to decide what actions to take in this case, and evaluate these actions by predictions algorithms if they reach agent close nearly to its final target (i.e., dynamic intelligence).
The rest of paper is organized as follows. In Section 2, we introduce description of accumulative intelligence. Dynamic intelligence is described in Section 3. Finally, we conclude this paper in Section 4.

2. Accumulative Intelligence

Many research studies [7] applied the concept of accumulative intelligence on their agent technology applications. Simply, accumulative intelligence emphasizes the fact that agent (human) must collect a good deal of observed information about its environment before making decisions about the proper actions to transact with a specific situation. This paper introduces the terminology of accumulative intelligence for the first time.

2.1. Generic Model for Accumulative Intelligence

Figure 1 shows generic model and basic phases of accumulative intelligence. The model verifies that agent can be intelligent when it has the ability to interact with its environment. Interface phase makes automatic gathering of information about status of environment components and give to the agent through short-term storage phase. *Short-term storage* is a data storage technique that used to store collected information as attractors. Interface consists of a group of sensors. *Processing and decision rule* phase analyzes and processes momentarily information collected and deliverables gathered from interface phase (attractors). By using decision making rule phase, it generates output signals and transmits to effectors to do the required action into the environment.
Accumulative intelligence can be deliberated from two different perspectives: moment perspective and historical perspective. Moment perspective emphasizes that process of

information collection, processing, and generating proper actions is done momentarily from beginning of agent task to its final or sub-final target (i.e., action-sequence) without using any prediction or anticipating techniques. *Action-sequence* is reached to that target by memorized into long-term storage phase in Fig. 1. In the wake of this, *long-term storage* memorizes different action-sequences to different targets.

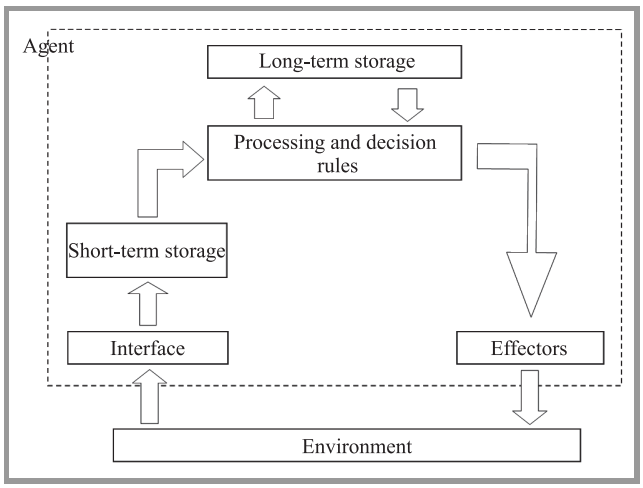


Fig. 1. Generic model of accumulative intelligence.

There are some situations in which agent target is repeated, so agent do not need to find the action-sequence of that target because it is memorized previously in long-term storage. Agent just needs to recall the action-sequence of repeated target from long-term storage. The previous description refers to historical perspective of accumulative intelligence. Some agent applications as anti-money laundering systems [8] applied another view of historical perspective. These applications perform the process of information collection over consecutive fixed time intervals and memorize them into long-term storage directly as information slices in a database. All information slices in that database is processed over random or regular time instants. Processing phase tries to find a predefined information pattern in database by a comparison function. An expert or application designers define previously this pattern. As soon as this pattern was found in database, the proper action is generated. Type and effect of action is related to its application or environment. The salient property of previous historical perspective view is that process of information collection, processing, and generating actions is not performed momentarily.

Variety of agent technology application is ranged between consummating only the concept of *moment accumulative intelligence* (MAI), only the concept of *historical accumulative intelligence* (HAI), or both two concepts which is called *full accumulative intelligence* (FAI). The following section will shed some light on an example of agent technology applications that apply the concept of accumulative intelligence.

2.2. Auto-Vehicle Driver (AVD)

Auto-vehicle driver emphasizes creation of full software and hardware system to be able of driving vehicles automatically without interference of human driver. It is similar to auto-pilot system used in aircrafts. AVD may be composed of only one intelligent agent or a group of more than one (i.e., multi-agent system (MAS)). Regardless of agent number in AVD, each one should perform its tasks of collecting its related information via its sensor. They include, but not certainly limited to the following parameter, road status (e.g., GPS information of road, road width, and etc.. . .), vehicle status (e.g., speed, fuel amount, vehicle components, position, and etc.. . .), obstacle status (e.g., is another vehicle (human, animal) in front, at backward, at right, at left of that vehicle), and climate status (e.g., is it rains, windy, sunny, and etc.. . .). According to processing quality of this information, the proper moving action-sequence will be generated. There is no full AVD system up till now due to huge information amount that must be collected and processed but there are some researches that applied agent technology with accumulative intelligence concept to assist vehicle driver by using algorithms of machine leaning.

In [9] the author addresses the problem of autonomous navigation of a car-like robot evolving in an urban environment. Such an environment exhibits a heterogeneous geometry and is cluttered with moving obstacles. The approach to the problem lies in the design of an intelligent agent able to percept and plan actions that consider explicitly the dynamic nature of the vehicle and the environment while enforcing the safety constraint.

Kolski *et al*. 2006 [10] present a hybrid navigation system that combines the benefits of existing approaches for driving in structured environments (e.g., roads) and unstructured environments (e.g., parking lots). The system uses visual lane detection and laser range data to generate a local map, which is processed by a local planner to guide the vehicle down the lane while avoiding obstacles. When driving in unstructured environments, the system employs a global map and planner to generate an efficient trajectory to a desired goal. The combined system is capable of navigating a passenger car to a given goal position without relying on road structures, yet it makes use of such structure when it is available.

Bertolazzi *et al*. 2008 [11] designed a reduced size autonomous vehicle and focuses on the control strategy which is based on a Nonlinear Receding Horizon Control (NRHC) algorithm. The NRH planner is called by a high level manager, in the outer control loop, to solve a sequence of optimal control problems. The planned motion provides a sequence of reference set points for the faster inner control loop until a new plan is available. The lateral motion is controlled through the steering command by tracking the yaw rate reference. The longitudinal motion is controlled by means of the throttle/braking coupled command by tracking the planned forward speed profiles.

Borges *et al*. 2009 [12] presented an intelligent agent that controls the vehicle speed using knowledge learned from

databases. The main effort of this paper was the induction of conduct rules from data of previous trips and collected data from sensors. Sensors capture parameters such as: current kilometer (position), speed, latitude, longitude, and adherence. Algorithms of JRip, Bagging, and Boosting have been used, employing data from past trips to extract patterns of safe and economical conduct.

Several studies have used machine learning techniques for the development of intelligent railway vehicles [13], whose goal is to improve the maintenance and operation of vehicles predicting failures in equipments.

3. Dynamic Intelligence

Dynamic intelligence emphasizes relations among observed, predicted, and recalled time sequences in sensory-motor space. Sensory-motor space applications as robots are a luminous examples of interactions between body and its environment. In the literature of dynamic intelligence references, many researchers pursue a range of different characteristics that agent should possess such as: behavior emergence, imitation learning, and spontaneous learning. Agent should be implemented to fulfill part or all of these characteristics.

Behavior emergence is defined as how agent (body) can decide what appropriate actions that can take to overcome unpredictable and complex new situations [14]. Another important factor in dynamic intelligence is imitation based on embodiment. The importance of imitation is also emphasized in brain science with *mirror neurons* [15]. The existence of mirror neurons implies that a human interprets another human's actions as actions by himself, providing an understanding of their meaning. Based on the interpretation of the mirror neurons, embodiment plays an important role in understanding the meaning behaviors. Spontaneous learning is another factor in dynamic intelligence. It can be illuminated by flow theory [16] which insists that a human is highly motivated when he/she tries to learn a task of suitable difficulty for his/her capabilities and skills. It is important to make balance between human (agent) skills and a task difficulty. This balance can be evaluated by progress of agent ability of self-spontaneous learning. For example, if the task is too easy, the agent can predict states correctly and achieve a goal easily. On the other hand, if the task is too difficult, the agent can not predict states correctly neither can it achieve the goal even it tries many times. However, if the task is a suitably difficult, at the beginning of learning, the agent can not predict states correctly and consequently goals can not achieved. But after some attempts, the agent can achieve goals eventually due to predicted states.

3.1. Generic Model for Dynamic Intelligence

Figure 2 shows the generic model of dynamic intelligence [17] which is nearly similar to general model of accumulative intelligence. Dynamic intelligence model has a new two phases. The first is *predictor/controller* phase and the second is *reduce dimension* phase.

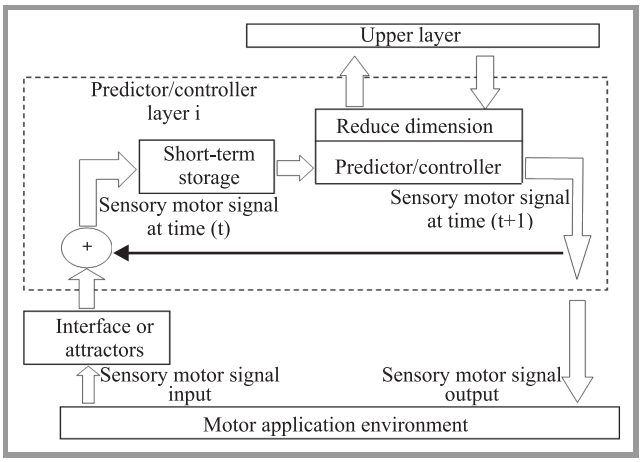


Fig. 2. Generic model of dynamic intelligence.

Predictor/controller phase is responsible of reaching sensory-motor system to its target state through exploration function. Exploration function is used to determine intermediate action-sequence to final target state and use it as training data for the predictor/controller to form the dynamic attractors. This can be done by trying some actions and deciding which action will allow it to reach to its final target.

The main key way to do the exploration is to explore in the real world which can be done using a planning and a rehearsing function. The planning function explores the proper actions to get close to final target state by using the predictor (anticipator), which can predict the state in the next time step. The rehearsing function simulates the sensory-motor signal sequence to get close to the target by using the predictor and the controller. Once the sensory-motor system found action-sequence to reach to final target state, it can be used as teaching signal pairs for the controller. Also, it is important to memorize it in long-term storage for reuse in the future if this final target is repeated.

Chess game is a good example to understand the previous description. Main target of chess player is to reach to "King Dead" state. To do that, chess player is trying to build a theoretical plan in his mind before moving any one of its chess pieces. For example, consider the following steps as a beginning for a plan: firstly, he will move the "Queen" to a specific position, and then he will attack the "Queen" of the other player by his "Bishop" to get it away from its king, and so on. This player will try to predict all possible defense movements that the other player will do against this player movement. He will use each predicted defense movement as a new start step for another subplan from his original plan. So, original plan is divided to subplans which their counts equal to all possible defense movements. Each subplan will be divided into smaller subplans and so on until one of these subplans reach closely to its final target. In the wake of this, predicted whole plan (action-

sequence) is determined by using layers of prediction steps (planner function). After that the player simulates in his mind this action-sequence (rehearsing function). Finally, he will begin to move his pieces (signals to controller).
When dealing with dynamic intelligence, it is important to use reduce dimension phase. To understand operation of it, chess example will be reused. At the beginning of chess game plan, player thinks of moving one of 16 pieces, so information about all these 16 pieces must be known (large information space). After player predict first step, number of pieces that will participate for next prediction step will be reduced causing lower information space for next prediction layer. The process of lowering information space will be continued through prediction layers until reaching to target. Reduce dimension phase is responsible of straitening dimensions of information space through predictor/controller layers.

3.2. Related Researches of Dynamic Intelligence

There are some pioneering works that deal with dynamic intelligence. Let us consider the following three works and identify their essential functions and properties: MOSAIC, Mimesis model, and MINDY model.

3.2.1. MOSAIC Model

MOSAIC [2] consists of many prediction and control pairs for sensory-motor signals such as the Locally Weighted Projection Regression (LWPR) [17] as a prediction function. Using MOSAIC the authors succeed to develop the humanoid named DB, which learns performances of juggling, devil stick handling, and so on. The important difficulty of MOSAIC is how to select a controller that properly works in each situation. It selects a controller based on the goodness of the corresponding prediction function. In a more advanced version of MOSAIC [18], the control signal is generated by adding the controllers' output weighted by the goodness of the corresponding predictors. The weighted values based on the predictors' *goodness function* are used as inputs to the next layer, which is again composed of many prediction and control pairs for the weighted values. In summary the prediction function and a layered architecture are the important points of MOSAIC.

3.2.2. Mimesis Model

The Mimesis Model consists of many Hidden Markov Models (HMMs), which are trained with motion data of human actions such as walking, kicking, jumping and so on [19]. Then, distance metrics are introduced among the HMMs. Based on the metrics; the HMMs are set in a low dimensional space named *proto-symbol* space with Multi-Dimensional Scaling (MDS) algorithm. Then, a point in proto-symbol space represents a motion sequence. When a new motion sequence is input, a Mimesis system can recognize the sequence as a point in proto-symbol space. Multiple motion sequences such as walking and then kicking can be recognized as a sequence in proto-symbol space. If we can consider that the sequence in proto-symbol space is input to the upper layer, then the Mimesis system can be considered as the layered architecture.

3.2.3. MINDY Model

MINDY model [20] employs the concept of flow theory [16]. In order to achieve various task, MINDY has a network consisted of various modules. Each of the modules, called basic module, has an identical structure and was designed to achieve a goal for a subtask. The module differs in its location in network layers, in what kind of inputs and outputs are assigned, and in what it has learned. Each module contains a pair of predictor and controller. There is a planner and rehearsing function on top of the network of basic modules. The role of predictor/controller is to observe its inputs and outputs of the module and predicts the inputs of the next time step, which can be directly learned from history of observation and output actions. Accurate Online Support Vector Regression (AOSVR) [21] is used for the predictors and controllers. The planner and rehearsing function is used to generate and search action-sequence to come close to final target value with in each predictor target variable. A planner is used with in MINDY model. MINDY evaluates the progress of spontaneous learning by appraising predictor error, controller error, and goal error.

4. Conclusion

This paper introduces and discusses the possibility of applying human intelligence to agent. Claiming, that any human or machine has intelligence, requires that it should have a body and his body should interact with its surrounded environment. Agent can be considered as human body. This paper introduces two kinds of intelligence that many agent technology applications apply. The first is accumulative intelligence which emphasizes that agent must collect momentarily a good deal of its environment information before taking actions. Accumulative intelligence can be applied from two different perspectives: moment perspective and historical perspective. The luminous application of applying accumulative intelligence is auto-vehicle driver. The paper introduces research activities in this field.
The second type of intelligence is dynamic intelligence. Dynamic intelligence emphasizes relations of observed, predicted, and recalled time sequences in sensory-motor applications. In dynamic intelligence, agent can observe initial information of its environment and determine its final target by itself or by assistance of application designers or end users. Agent can predict the action-sequence that will reach it to its final target by using initial information of its environment. Agent should possess characteristics of behavior emergence, imitation learning, and spontaneous learning to be dynamic intelligent. MOSAIC, Mimesis, and MINDY models of dynamic intelligence are also discussed.

References

[1] M. Wooldridge, *An Introduction to MultiAgent System*. Chichester, England: Wiely, 2002.

[2] M. Haruno, D. M. Wolpert, and M. Kawato, "MOSAIC model for sensorimotor learning and control", *Int. J. Neural Computation*, vol. 13, no. 10, pp. 2201–2220, 2001.

[3] R. Pefeifer and C. Scheier, *Understanding Intelligence*. Cambridge, MA: MIT Press, 1999.

[4] E. S. Reed, *From Soul to Mind: the Emergence of Psychology from Erasums Darwin to William James*. Yale University Press, 1998.

[5] S. J. Russell and P. Norvig, *Artificial Intelligence: a Modern approach*. Prentice Hall, 2003.

[6] Y. Kuniyoshi, Y. Ohmura, K. Terada, A. Nagakubo, S. Eitoku, and T. Yamamoto, "Embodied basis of invariant features in execution and perception of whole-body dynamic actions – knacks and focuses of roll-and-rise motion", *Int. J. Robotics and Autonomous Sys.*, vol. 48, no. 4, pp. 198–201, 2004.

[7] S. Weagraff, "The case for intelligent agents: preparing for the future of care", in *Proc. Int. Conf. CIMCA-IAWTIC'05*, Vienna, Austria, 2005, pp. 976–980.

[8] S. Gao and D. Xu, "Conceptual modeling and development of an intelligent agent-assisted decision support system for anti-money laundering", *Int. J. Expert Sys. with Appl.*, vol. 39, no. 3, pp. 1493–1504, 2009.

[9] R. Benenson, S. Petti, T. Fraichard, and M. Parent, "Towards urban driverless vehicles", *Int. J. Vehicle Autonomous Sys.*, vol. 6, no. 1–2, pp. 4–23, 2008.

[10] S. Kolski, D. Ferguson, M. Bellino, and R. Seigwart, "Autonomous driving in structured and unstructured environments", in *Proc. IEEE Intelligent Vehicles Symp.*, Tokyo, Japan, 2006, pp. 558–563.

[11] E. Bertolazzi, F. Biral, P. Bosetti, M. De Cecco, R. Oboe and F. Zendri. "Development of a reduced size unmanned car", in *Proc. 10th IEEE Int. Worksh. Adv. Motion Control AMC 2008*, Trento, Italy, 2008, vol. 26–28, pp. 763–770.

[12] A. P. Borges, R. Ribeiro, B. C. Avila, F. Enembreck, and E. E. Scalabrin, "A learning agent to help drive vehicles", in *Proc. 13th Int. Conf. Comput. Supported Coop. Work in Design*, Santiago, Chile, 2009, pp. 1–11.

[13] C. Yang, and S. Letourneau. "Learning to predict train wheel failures", in *Proc. 11th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Chicago, IL, USA, 2005, pp. 1–11.

[14] G. Rzevski and P. Skobelev, "Emergent intelligence in large scale multi-agent systems", *Int. J. Education and Inform. Technol.*, vol. 1, no. 2, pp. 64–71, 2007.

[15] G. Rizzolattie, L. Fadiga, V. Gallese, and L. Fogassi, "Premotor cortex and the recognition of motor actions", *Int. J. Cognitive Brain Research*, vol. 3, no. 2, pp. 131–141, 1996.

[16] M. Fujita, "Intelligence dynamics: a concept and preliminary experiments for open ended learning agents", *J. Autonomous Agents and Multi-Agent Sys.*, vol. 19, no.3, pp. 248–271, 2009.

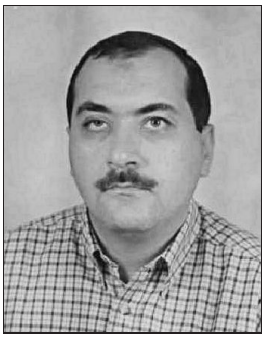
[17] S. Vijayakumar and S. Schaal, "LWPR: an O (n) algorithm for incremental real time learning in high dimensional space", in *Proc. 17th Int. Conf. Machine Learning ICML 2000*, Stanford, CA, USA, 2000, pp. 1079–1086.

[18] K. Doya, K. Samejima, K. Katagiri, and M. Kawato, "Multiple model-based reinforcement learning", *Int. J. Neural Computation,* vol. 14, no. 6, pp. 1347–1369, 2002.

[19] D. C. Bentivegna, C. G. Atkeson, and G. Cheng, "Learning tasks from observation and practice", *Int. J. Robotics and Autonomous Sys.*, vol. 47, no. 2–3, pp. 163–169, 2004.

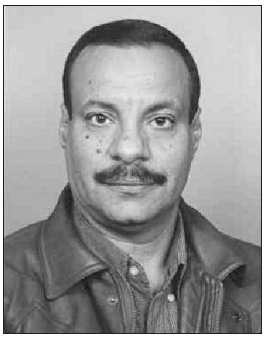
[20] K. Sabe, K. Hidai, K. Kawamoto, and H. Suzuki, "A proposal for intelligence model, MINDY for open ended learning system", in *Proc. Int. Worksh. Intelligence Dynamics at IEEE/RSJ Humanoids*, Genoa, Italy, 2006.

[21] J. Ma, J. Theiler, and S. Perkins, "Accurate on-line support vector regression", *Int. J. Neural Comput.*, vol. 15, no. 11, pp. 2683–2703, 2003.



Ahmed M. Elmahalawy receives his M.Sc. from Computer Science & Engineering Department, Faculty of Electronic Engineering, Menoufia University in 2001 and receives his Ph.D. degree from Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in 2009. Currently, he is a Lecturer in Computer Science & Engineering Department, Faculty of Electronic Engineering, Menoufia University. His research interests include artificial intelligence, agent and multi-agent technologies, and artificial life simulators.
E-mail: a_elmhalaway@Hotmail.com
Faculty of Electronic Engineering
Computer Science & Engineering Department
Menoufia University
Menouf, 32952, Egypt

Moukhtar A. Ali received his B.Sc. and M.Sc. degrees from Computer Science & Engineering Department, Faculty of Electronic Engineering, Menoufia University, Egypt, in 2002 and 2008, respectively. Currently, he is a Ph.D. candidate at the same department. His research interests include agent and multi-agent technologies, wireless ad-hoc networks, sensor networks, intelligent and expert embedded systems, and artificial intelligence.
E-mail: moukhtar_ahmed@ yahoo.com
Faculty of Electronic Engineering
Computer Science & Engineering Department
Menoufia University
Menouf, 32952, Egypt



Hany M. Harb is a Professor and Head of Computers and Systems Engineering Department, Faculty of Engineering, Al Azhar University. He receives his M.Sc. and Ph.D. from Computers and Systems Engineering Department, Faculty of Engineering, Azhar University and Computer Science, Illinois Institute of Technology (IIT) in 1981 and 1986, respectively. His research interests include artificial intelligence, expert systems, computer networks, data mining and warehousing and distributed systems.
E-mail: harbhany@yahoo.com
Faculty of Engineering
Computers and Systems Engineering Department
El Darb El Ahmer, Cairo, Egypt

Paper

A Comparative Study of Single- and Dual-Threshold Voltage SRAM Cells

Pragya Kushwaha and Amit Chaudhry[a]

[a] University Institute of Engineering and Technology, Panjab University, Chandigarh, India

Abstract—In this paper, a comparison has been drawn between 5 transistor (5T), 6T and 7T SRAM cells. All the cells have been designed using both single-threshold (conventional) and dual-threshold (dual-Vt) voltage techniques. Their respective delays and power consumption have been calculated at 180 nm and 65 nm CMOS technology. With technology scaling, power consumption decreases by 80% to 90%, with some increase in write time because of the utilization of high-Vt transistors in write critical path. The results show that the read delay of 7T SRAM cell is 9% lesser than 5T SRAM cell and 29% lesser than 6T SRAM cell due to the lower resistance of the read access delay path. While read power of 5T SRAM cell is reduced by 10% and 24% as compared to 7T SRAM, 6T SRAM cell respectively. The write speed, however, is degraded by 1% to 3% with the 7T and 5T SRAM cells as compared to the 6T SRAM cells due to the utilization of single ended architecture. While write power of 5T SRAM cell is reduced by up to 40% and 67% as compared to 7T SRAM, 6T SRAM cell respectively.

Keywords—5T SRAM, 65 nm CMOS technology, 6T SRAM, 7T SRAM, low power SRAM, power reduction technique.

1. Introduction

Static Random Access Memories (SRAM) are widely used in computer systems and many portable devices. As technology is scaling down, in result the threshold voltage is also scaling down along with the operating voltage. A huge amount of sub-threshold leakage current occurs due to low threshold voltage. SRAM based cache memories are best suited for system on chip applications due to its high speed and low power consumption. Due to device scaling there are several design challenges for micrometer SRAM design. Now we are working with very low threshold voltage and ultra-thin gate oxide due to which leakage energy consumption is getting increased. Besides this data stability during read and write operation is also getting affected in conventional 6T SRAM cell. In order to obtain higher noise margin along with better performance new SRAM cells like 5T and 7T SRAM cells have been introduced. In this paper a comparative analysis of 6T, 5T [1]–[5] and 7T [6]–[11] SRAM cell has been carried out. The major difference between the 5T and 7T SRAM cell is that in case of 5T SRAM cell a single bit line is used for read /write operation that saves area, bit line leakage and provides a read/write performance comparable to the 6T SRAM cell while in case of 7T SRAM cell, the storage nodes are isolated from the bit lines during a read operation, thereby enhancing the data stability as compared to the 6T SRAM cell. Table of the comparison of average power dissipation and delay time during write and read operation with the different combination of threshold voltage of NMOS and PMOS at 180 nm is summarized in Table 4. The paper is organized as follows. The 5T and 7T SRAM cell is presented in Section 2. Simulation results are shown in Section 3. Section 4 gives the conclusion.

2. SRAM Cells

A typical six transistor (6T) SRAM cell in a 65 nm technology is shown in Fig. 1. The robustness of an SRAM cell is characterized by the hold stability during read operation. In a 6T SRAM cell, the data storage nodes are directly accessed through the pass transistors connected to the bit lines. The storage nodes are disturbed due to the voltage division between the cross-coupled inverters and the access transistors during a read operation. The data is most vulnerable to external noise during a read operation due to this intrinsic disturbance produced by the direct data-read-access mechanism of a standard 6T SRAM circuit(destructive read) [12].

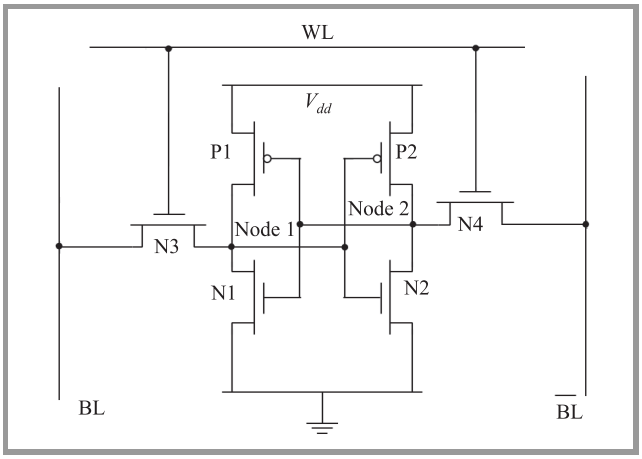


Fig. 1. Single-Vt 6T SRAM cell in a 65 nm CMOS technology: WL – word line, BL – bit line.

There are strict constraints on the sizing of transistors to be able to maintain the data stability and functionality of a standard 6T SRAM cell as shown in Table 1. The design of a 6T SRAM cell is typically characterized by the ratio (β) of the size of the pull-down transistors to the access transistors [12]–[14]. In order to maintain the read stability, N1 and N2 (Fig. 1) must be stronger as compared to

the access transistors N3 and N4. Alternatively, for write ability, N3 and N4 must be stronger as compared to P1 and P2. These requirements are satisfied with careful transistor sizing, as illustrated in Fig. 1.

Table 1
Width of transistor used for simulating 6T SRAM

Transistors	Width [nm]
N1, N2	130
N3, N4	100
P1, P2	65

In addition to the data stability issues, the increasing leakage energy consumption of the embedded memory circuits is also a growing concern. In modern high performance microprocessors, more than 40% of the total active mode energy is consumed due to leakage currents [15]–[19]. So, dual threshold voltage transistors are used for this purpose. In dual-Vt 6T SRAM cell (Fig. 2) transistors N1, N2, P1, P2 are designed as high threshold voltage (Vtn-high for NMOS, Vtp-high for PMOS) transistors because they are more appropriate to store data in memory design and access transistors N3, N4 are designed as low threshold voltage (Vtn) transistors because they can possess larger drain current. In result it reduces the access time and maintains data retention at the same time [20]–[27].

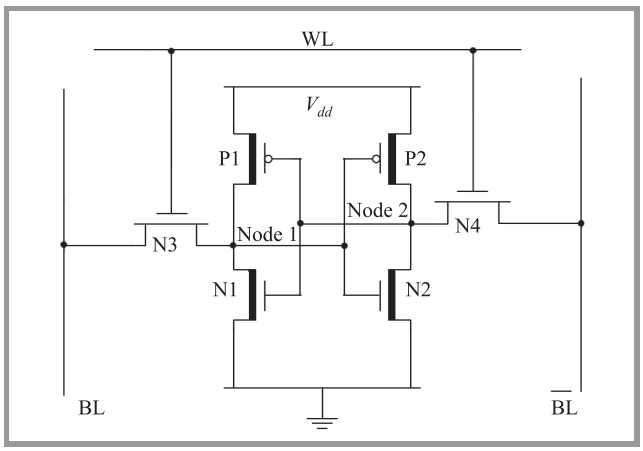


Fig. 2. Dual-Vt 6T SRAM cell in a 65 nm CMOS technology: WL – word line, BL – bit line.

2.1. The 5T SRAM Cell

The 5T cell has only one access transistor N3 and a single bit line BL. Writing of 1 or 0 into the 5T cell is performed by driving the bit line to V_{dd} (1.8 V) or V_{ss} (0 V) respectively, while the word line is asserted at V_{dd}. The write ability of the cell is ensured by a different cell sizing strategy. A sizing example in a standard 65nm CMOS is shown in Fig. 3 [1]. The trip point of the inverter N2-P2 has been decreased, while the trip-point of the inverter N1-P1 has been increased. Further, the pass-transistor N3 is sized to support both write and read operation.

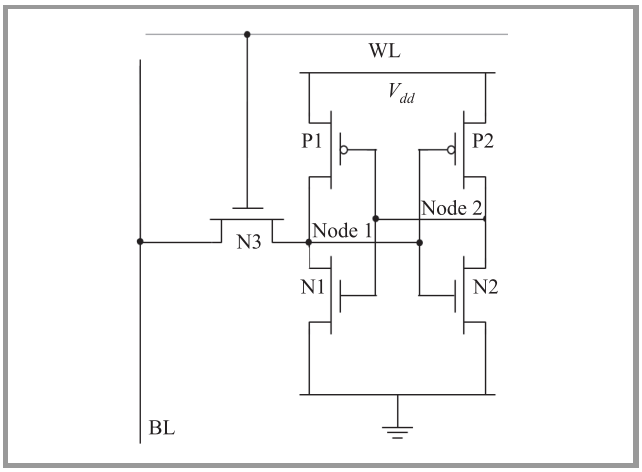


Fig. 3. Single-Vt 5T SRAM memory cell in a standard 65nm CMOS technology.

Since the 5T SRAM cell is writable at $V_{BL} = V_{WL} = V_{dd}$, a non-destructive read operation requires a bit line precharge voltage, V_{PC} ($\sim 600 - 650$ mV), where $V_{SS} < V_{PC} < V_{dd}$. This is in contrast to the conventional 6T SRAM bit lines, which are precharged at V_{dd} before a read operation.

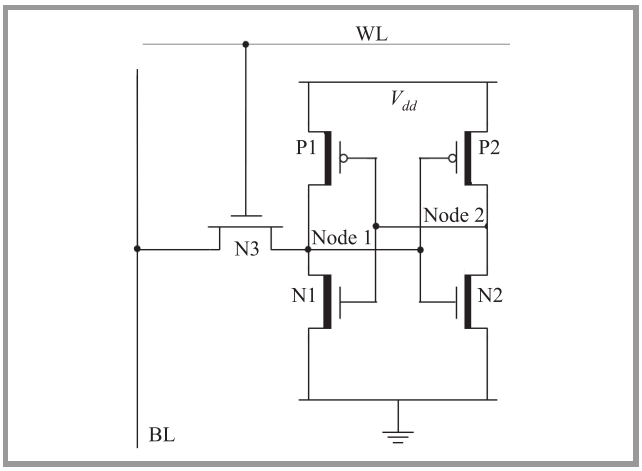


Fig. 4. Dual-Vt 5T SRAM memory cell in a standard 65 nm CMOS technology. Thick line in the channel area indicates a high-Vt transistor.

We have designed a dual-Vt 5T SRAM in this paper (Fig. 4) by high-Vt transistors P1, P2, N1, N2 and low-Vt transistor N3. And the sizes of transistors are shown in Table 2. This configuration saves leakage power dissipation in comparison to dual-Vt 6T SRAM cell.

Table 2
Width of transistor used for simulating 5T SRAM

Transistors	Width [nm]
N3	130
P1, N2	100
N1, P2	65

2.2. The 7T SRAM Cell

The schematic of the 7T dual-Vt SRAM cell [6] with transistor sized for 65 nm CMOS technology is shown in Fig. 5 and transistor's width is shown in Table 3. Prior to a read operation, the RBL is precharged to V_{dd}. To start the read operation, the read signal R transitions to V_{dd} while the write signal W is maintained at V_{ss}. If a 1 is stored at node1, RBL is discharged through the transistor stack formed by N4 and N5. Alternatively, if a 0 is stored at node1 RBL is maintained at V_{dd}. The storage nodes (node1, node2) are completely isolated from the bit lines during a read operation. The data stability is thereby significantly enhanced as compared to the conventional 6T SRAM cells. Prior to a write operation the WBL is charged (discharged) to V_{dd} (V_{ss}) to get ready to force a 1 (0) onto node1. To start the write operation, the write signal W transitions to V_{dd} while the read signal R is maintained at V_{ss}. Write 0 onto node1, the pass transistor N3 must be stronger as compared to the pull-up transistor P1.

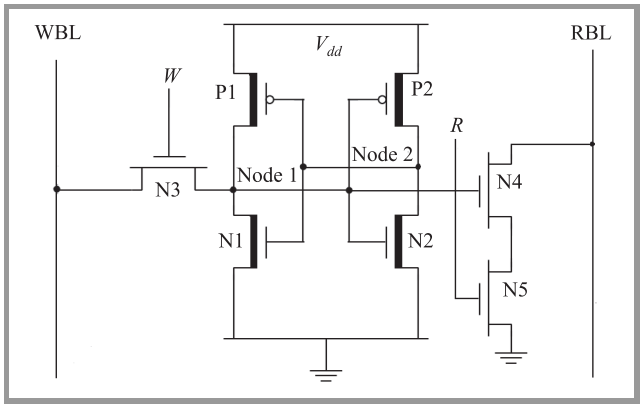


Fig. 5. The schematic of 7T dual-Vt SRAM circuit in a 65 nm CMOS technology. WBL – write bit line, RBL – read bit line, W – write control signal, R – read control signal. For data stability $\beta = 0.25$. Thick line in the channel area indicates a high-Vt transistor.

Table 3
Width of transistor used for simulating 7T SRAM

Transistors	Width [nm]
N3, N4, N5	130
P1, N2	100
N1, P2	65

Alternatively, to write 1 onto node1, the pass transistor N3 must be stronger as compared to N1.
Furthermore, since N3 transfers a degraded 1 (due to the Vt drop across the N-channel access transistor), the inverter formed by N2 and P2 is required to have a low switching threshold voltage that assists the transfer of a full 1 onto node1. Hence, these design requirements are achieved by employing dual-Vt transistors with in the cross-coupled inverters as shown in Fig. 5. As these cross-coupled inverters are not on the read-delay-path. The transistor sizing of the dual-Vt cross-coupled inverters therefore does not affect the read speed and at the same time reduces the leakage power of the cell. In this way at a time only one bit line is activated in this SRAM cell that saves bit line leakage.

3. Simulation Results

In this section comparison between conventional 5T, 7T SRAM cell has been carried out on the basis read delay, write delay, average write power and read power (Table 4).

Table 4
Comparison of average power dissipation and delay time during write and read operation with the different combination of threshold voltage of NMOS and PMOS at 180 nm

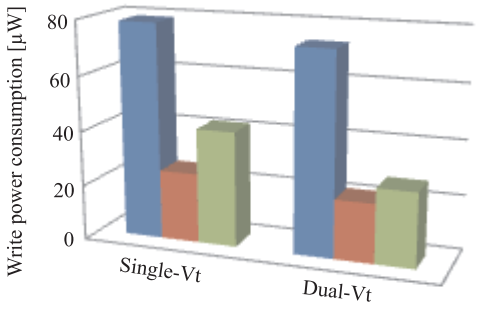
Cell	Mode	Av. write power [μW]	Write delay [ps]	Av. read power [μW]	Read delay [ps]
6T SRAM	Single-Vt	78.74	18.73	45.69	18.46
	Dual-Vt	73.4	18.68	39.42	18.48
5T SRAM	Single-Vt	25.33	18.92	34.52	14.17
	Dual-Vt	21.54	18.72	33.62	14.18
7T SRAM	Single-Vt	41.93	18.94	38.6	12.95
	Dual-Vt	27.14	18.82	35.47	12.95

3.1. Average Read/Write Power Consumption

Write power consumption of single ended structure [28]–[37] is less because the bit line capacitance is reduced as compared to 6T double bit line switching.

$$P_s = \alpha C_l V_{dd}^2 F_{cl}\,,$$

where P_s is switching power dissipation, α is activity factor, C_l is load capacitance and V_{dd} is power supply, F_{cl} is input clock frequency.
As shown in Fig. 6, the write power is significantly reduced with the proposed 5T and 7T SRAM cell as compared to the 6T SRAM cells. This reduction in the write power is due to the utilization of a single bit line for writing into the 5T and 7T SRAM cells in a memory column. For the 6T SRAM cell both bit lines in each memory column are periodically precharged to V_{dd}. After the bit line precharge is completed and once a write decision is made, one of the precharged bit lines is selectively discharged to V_{ss} (0 V) to perform a write operation. In a memory array with 6T SRAM cell, therefore, one of the bit lines needs to be fully charged and discharged during each write cycle, regardless of whether a 0 or a 1 is transferred to the cell. Alternatively, in case of writing a 1 to a memory column with the 5T SRAM, 7T SRAM cells, the write bit line (WBL) does not need to be discharged (maintained at the precharge voltage V_{dd}). The bit line dynamic switching

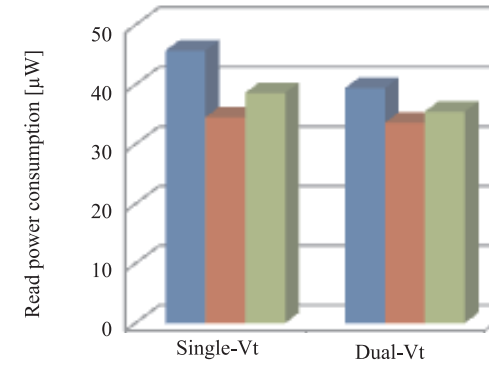


Cell	Single-Vt [µW]	Dual-Vt [µW]
6T SRAM	78.74	73.4
5T SRAM	25.33	21.54
7T SRAM	41.93	27.14

Fig. 6. Write power consumption.

power consumption is thereby significantly reduced with the 5T SRAM and 7T SRAM cells. Hence write power in 5T SRAM cell is reduced by 67% and in 7T SRAM cell it is reduced by 45% as compared to conventional 6T SRAM cell. As shown in Fig. 1, the write power of 5T SRAM cell is reduced by up to 40% as compared to 7T SRAM cell because RBL remains at V_{dd} (leakage power dissipation during write operation) in 7T SRAM cell.

During read operation, 5T SRAM precharges the single bit line to $V_{pc} = 650$ mV, which causes lower V_{ds} (drain to source voltage) over the word line pass transistor (N3 in Fig. 4). Thus read power of 5T SRAM cell is reduced by 10% and 24% as compared to 7T SRAM, 6T SRAM cell respectively (see in Fig. 7).



Cell	Single-Vt [µW]	Dual-Vt [µW]
6T SRAM	45.69	39.42
5T SRAM	34.52	33.62
7T SRAM	38.6	35.47

Fig. 7. Read power consumption.

The comparison results for power consumed during read and write operation is shown in Fig. 8. The power is higher for 7T SRAM cell due to higher parasitic capacitance.

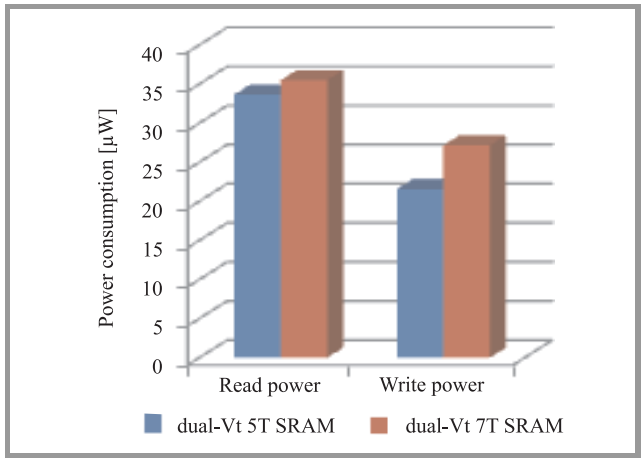


Fig. 8. Comparison between dual-Vt 5T SRAM cell and dual-Vt 7T SRAM cell on the basis of power consumption during read/write operations.

Due to the higher number of transistor used in 7T SRAM cell its leakage energy consumption increases. To reduce the leakage current in 7T SRAM dual-threshold voltage technology [20]–[27] has been used. High-threshold voltage transistors are not used for the access transistor as it increases the write delay. The threshold voltage for high-Vt and low-Vt NMOS/PMOS is shown in Table 5.

Table 5
Threshold voltage for low-Vt and high-Vt NMOS/PMOS transistors

Transistor	Low-Vt [V]	High-Vt [V]
PMOS	–0.2	–0.11
NMOS	0.47	0.76

As CMOS devices will continue to downscale into the deep nanometer range with improved device performance and lower power, it will be running into fundamental barriers of physics. Scaling below 45 nm channel length faces several fundamental limiting factors stemming from electron thermal energy and quantum mechanical tunneling. Hence in this paper we have designed SRAM at 180 nm and scaled it down to a limit of 65 nm, to study the variations occurred in power and delay factors of the memory. We can observe the effect of technology scaling on power consumption factor of SRAM memory cell in Figs. 9 and 10.

Thus technology scaling is an interesting way to lower the power consumption. Indeed, the overall parasitic capacitances (i.e., gates and interconnects) are decreased, the available active current per device is higher, and consequently, the same performance can be achieved with a lower supply voltage. It is evident from Figs. 9 and 10 that with technology scaling, power consumption decreases

by 80 to 90%, with some increase in write time (see in Fig. 11) because of the utilization of high-Vt transistors in write critical path.

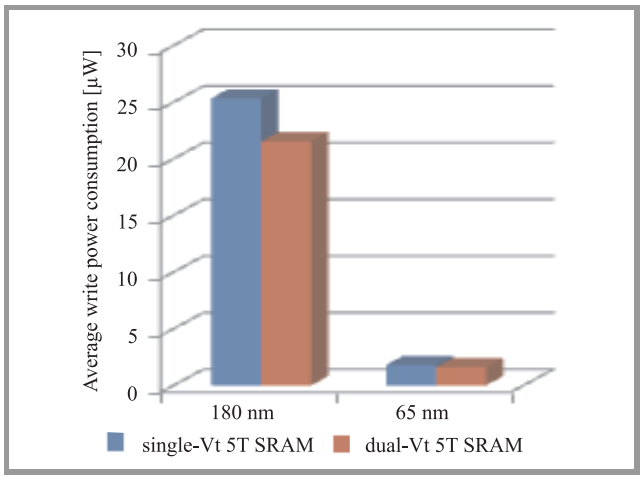


Fig. 9. Power reduction with the technology scaling in 5T SRAM cell.

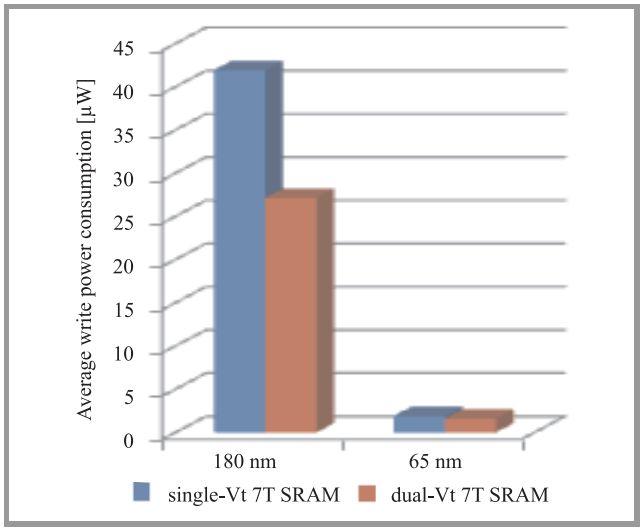


Fig. 10. Power reduction with the technology scaling in 7T SRAM cell.

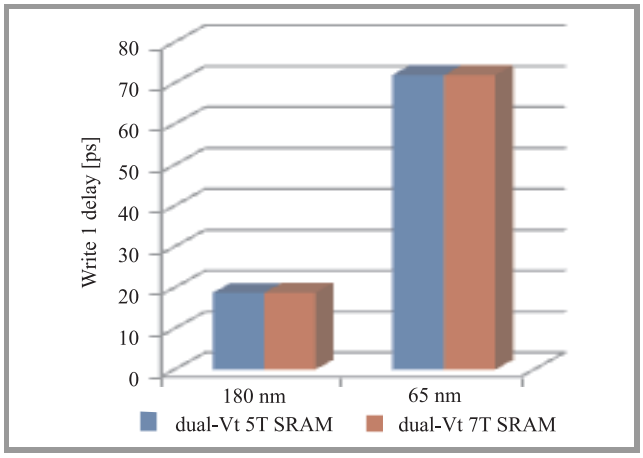


Fig. 11. Increase in write 1 delay as technology is scaled down.

3.2. Read/Write Delay Time

As shown in Fig. 12, 7T SRAM cell is not only robust but also is faster as compared to the 5T and 6T SRAM cells during a read operation. The read critical path is composed of two series transistors (N4 and N5) each sized twice a minimum sized transistor in 7T SRAM cell (see in Fig. 5).

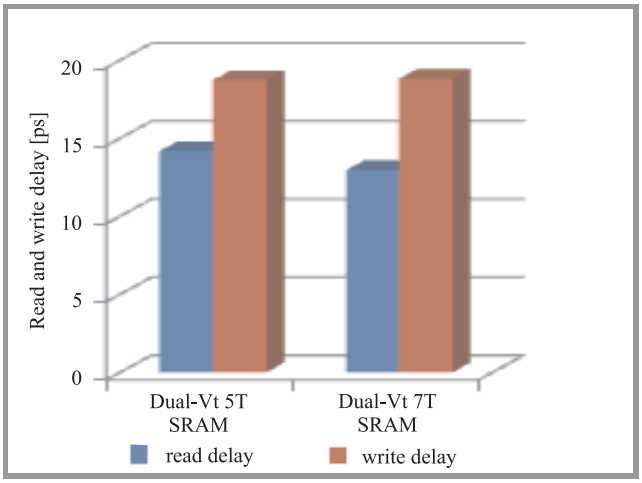


Fig. 12. Read and write delay comparison.

Alternatively, with the 5T SRAM cells, the read critical path are composed of series transistors (N3 and N1) with the access transistors (N3) sized stronger than driver transistor (N1) to maintain cell's read speed as shown in Fig. 4. The driving capability of the high-Vt transistor (N1) is lesser than any low-Vt transistor hence read delay of 5T SRAM increases. The read speed of 7T SRAM cell is 9% higher than 5T SRAM cell and 29% higher than 6T SRAM cell due to the lower resistance of the read access delay path.

The write speed, however, is degraded by 1 to 3% with the 7T and 5T SRAM cells as compared to the 6T SRAM cells due to the utilization of only a single bit line for writing into the cells with the proposed technique. This result is proving the fact that the write operation of single ended bit line SRAM cell is difficult because of strongly coupled inverters. Write 1 speed is slow because of NMOS pass transistor (N3 in Figs. 4 and 5) as it will pass weak 1 to the storage node (node1 in Figs. 4 and 5), so switching time of the memory cell get increased, as a result of this effect write 1 delay of 5T SRAM cell and 7T SRAM cell increased in comparison to 6T SRAM cell.

4. Conclusions

The 5T SRAM and 7T SRAM have been compared with respect to 6T SRAM. Read delay of 7T SRAM cell is 9% lesser than 5T SRAM cell because of the lower resistance of the read access delay path. Write delay of 5T SRAM and 7T SRAM is 1 to 2% higher than conven-

tional 6T SRAM cell due to single ended bit line architecture. Read power consumption of 5T SRAM is 10% lesser than 7T SRAM cell due to advantage of low voltage of bit line during read operation. 7T SRAM has 65% higher write power consumption as compared to 5T SRAM cell because RBL is at supply voltage during write operation which causes leakage power consumption. In this way we observe that 7T SRAM cell has high read speed with the loss of power. On the other hand 5T SRAM cell has benefit of low power consumption with the loss of read speed.

Acknowledgment

The authors thank the Director, UIET, Panjab University (Chandigarh), India for allowing to the work. The authors would like to thank to Panjab University, Chandigarh, India for providing excellent research environment to complete this work. The authors wish to thank all individuals who have contributed directly or indirectly in completing this research work.

References

[1] I. Carlson, S. Andersson, S. Natarajan, and A. Alvandpour, "A high density, low leakage, 5T SRAM for embedded caches", in *Proc. 30th Eur. Solid State Circ. Conf. ESSCIRC 2004*, Leuven, Belgium, 2004, pp. 215–218.

[2] M. Margala and M. Wieckowski, "A novel five-transistor (5T) SRAM cell for high performance cache", in *Proc. IEEE Int. SOC Conf. SOCC 2005*, Herndon, VA, USA, 2005, pp. 101–102.

[3] S. Cosemans, W. Dehaene, and F. Catthoor, "A low power embedded SRAM for wireless applications", in *Proc. IEEE 32nd Conf. Solid State Circuit*, Montreu, Switzerland, 2006.

[4] S. Cosemans, W. Dehaene, and F. Catthoor "A Low-Power Embedded SRAM for Wireless Applications", *IEEE J. Solid State Circuits*, vol. 42, no. 7, 2007.

[5] R. F. Hobson and H. Jarollahi, "Power and area efficient 5T-SRAM with improved performance for low-power SoC in 65 nm CMOS", in *IEEE 53rd Int. Midwest Symp. Circ. Sys. MWSCAS 2010*, Seatle, WA, USA, 2010, pp. 121–124.

[6] S. A. Tawfik and V. Kursun, "Low power and roubst 7T dual-Vt SRAM circuit", in *Proc. IEEE Int. Symp. Circ. Sys., ISCAS 2008*, Seatle, WA, USA, 2008, pp. 1452–1455.

[7] R. E. Aly, M. I. Faisal, and M. A. Bayoumi, "Novel 7T sram cell for low power cache design", in *Proc. IEEE Int. SOC Conf. SOCC 2005*, Herndon, VA, USA, 2005.

[8] R. E. Aly and M. A. Bayoumi, "Low-power cache design using 7T SRAM cell", *IEEE Trans. Circ. Sys.*, vol. 54, no. 4, 2007.

[9] J. Singh, J. Mathew, and K. D. Pradhan, "A subthreshold single ended I/O SRAM cell design for nanometer CMOS technologies", in *Proc. IEEE Int. SOC Conf. SOCC 2008*, Belfast, UK, 2008, pp. 243–246.

[10] M.-T. Chang, P.-T. Huang, and W. Hwang, "A robust ultra-low power asynchronous FIFO memory with self-adaptive power control", in *Proc. IEEE Int. SOC Conf. SOCC 2008*, Belfast, UK, 2008.

[11] T. Azam and B. Cheng, "Variability resilient low-power 7T-SRAM design for nano-scaled technologies", in *Proc. IEEE 11th Int. Symp. Quality Electron. Design*, San Jose, CA, USA, 2010.

[12] Z. Liu and V. Kursun, "High read stability and low leakage cache memory cell", in *Proc. IEEE Int. Symp. Circ. Sys.*, Cancun, Mexico, 2007, pp. 2774–2777.

[13] L. Chang *et al.*, "Stable SRAM cell design for the 32 nm node and beyond", in *Proc. IEEE Symp. VLSI Technol.*, Hsinchu, Taiwan, 2005, pp. 128–129.

[14] P. Athe and S. Dasgupta, "A comparative study of 6T, 8T and 9T decanano SRAM cell", in *Proc. IEEE Symp. Industrial Electron. Appl. ISIEA 2009*, Kuala Lumpur, Malaysia, 2009.

[15] G. Sery *et al.*, "Life is CMOS: Why chase life after?", in *Proc. IEEE Design Automation Conf.*, New Orleans, LA, USA, 2002, pp. 78–83.

[16] R. Keerthi and C. Chen, "Stability and static noise margin analysis of low-power SRAM", in *Proc. IEEE Int. Conf. Instrument. Measur. Technol.*, Victoria, BC, Canada, 2008.

[17] S. S. Rathod, S. Dasgupta, and A. K. Saxena, "Investigation of stack as a low power design technique for 6-T SRAM cell", in *Proc. IEEE Region 10th Conf. TENCON 2008*, Hyderabad, India, 2008.

[18] D. Hentrich, E. Oruklu, and J. Saniie, "Performance evaluation of SRAM cells in 22nm predictive CMOS technology", in *Proc. IEEE Int. Conf. Electro/Inform. Technol.*, Windsor, Ontario, Canada, 2009.

[19] S. Birla, N. Kr. Shukla, D. Mukherjee, and R. K. Singh, "Leakage Current reduction in 6T single cell SRAM at 90nm technology", in *Proc. IEEE Int. Conf. Advances Comput. Engin.*, Bangalore, Karnataka, India, 2010.

[20] C.-C. Wang, P.-M. Lee, and K.-L. Chen, "An SRAM design using dual threshold voltage transistors and low-power quenchers", *IEEE J. Solid State Circ.*, pp. 1712–1720, vol. 38, no. 10, 2003.

[21] B. Amelifard, M. Pedram, and F. Farzan, "Reducing the subthreshold and gate-tunneling leakage of SRAM cells using dual-Vt and dual-tox assignment", in *Proc. IEEE Conf. Design, Automation and Testing in Europe*, Munich, Germany, 2006, vol. 1, p. 1.

[22] F. Moradi, D. T. Wisland, H. Mahmoodi, Y. Berg, and T. V. Cao, "New SRAM design using body-bias techniques for ultra low power applications", in *Proc. IEEE 11th Int. Symp. Quality Electron. Design*, San Jose, CA, USA, 2010, p. 468.

[23] C. H. Kim and K. Roy, "Dynamic Vt SRAM: A leakage tolerant cache memory for low voltage microprocessors", in *Proc. Int. Symp. Low Power Electr. Design ISLPED'02*, Monterey, CA, USA, 2002, pp. 251–254.

[24] C. H. Kim and J.-J. Kim, "A forward body-biased low leakage SRAM cache: Device and architecture considerations", in *Proc. Int. Symp. Low Power Electr. Design ISLPED'03*, Seoul, Korea, 2003, pp. 6–9.

[25] B. Amelifard, F. Fallah, and M. Pedram, "Leakage minimization of SRAM cells in a dual-Vt and dual-tox technology", *IEEE Trans. on Very Large Scale Integra. Sys.*, vol. 16, no. 7, 2008.

[26] J. T. Kao and A. P. Chandrakasan, "Dual-threshold voltage techniques for low-power digital circuits", *IEEE J. Solid State Circ.*, vol. 35, no. 7, 2000.

[27] S. Panda, N. M. Kumar, and C. K. Sarkar, "Power, delay and noise optimization of a SRAM cell using a different threshold voltages and high performance output noise reduction circuit", in *Proc. Int. Conf. Comput. Devices for Commun. CODEC 2009*, Kolkata, India, 2009.

[28] K. W. Mai, T. Mori, and B. S. Amrutur, "Low-power SRAM design using half-swing pulse-mode techniques", *IEEE J. Solid State Circ.*, vol. 33, no. 11, 1998.

[29] E. J. Stine and H. Choday, "Single-ended half-swing low-power SRAM design", in *Proc. IEEE 42nd Conf. Signals, Sys. Comput.*, Pacific Grove, CA, USA, 2008, p. 2108.

[30] J. Singh, J. Mathew, and P. S. Mohanty, "Single ended static random access memory for low-vdd, high-speed embedded systems", in *Proc. IEEE 22nd Int. Conf. VLSI Design*, New Delhi, India, 2009.

Pragya Kushwaha is a final year student of M.Tech Microelectronics University Institute of Engineering and Technology, Panjab University, Chandigarh, India. She obtained her Bachelor Degree in Engineering (Electronics and Communication) from Raj Kumar Goel Institute of Technology, Ghaziabad, India, in 2009. Her area of research includes designing of low power memories at micrometer regime. Currently, she is working towards her thesis under the esteemed guidance of Dr. Amit Chaudhry.
E-mail: pragya189@gmail.com

Amit Chaudhry completed his Ph.D. in Microelectronics in 2010 from Panjab University, Chandigarh, India. He joined Panjab University, University Centre for Instrumentation and Microelectronics in October, 2002. He was responsible for teaching and research in VLSI and microelectronics to post graduate students. His research areas include device modeling for sub 100 nm MOSFETS. He is a life member of various societies in the area of microelectronics. Currently he is Senior Assistant Professor, University Institute of Engineering and Technology, Panjab University, Chandigarh. He has more than 35 publications in international journals/conference proceedings. He is a principle investigator in the Department of Information Technology (DIT) project on nanoscale and low temperature MOSFET modeling. He is a reviewer and member of editorial board of several peer reviewed international journals and highly reputed international conferences.
E-mail: amit_chaudhry01@yahoo.com
University Institute of Engineering and Technology
Panjab University
Chandigarh, India

Information for authors

Journal of Telecommunications and Information Technology **(JTIT)** is published quarterly. It comprises original contributions, both regular papers and letters, dealing with a wide range of topics related to telecommunications and information technology. **All regular papers and letters are subject to peer review**. Topics presented in the JTIT report primary and/or experimental research results, which advance the base of scientific and technological knowledge about telecommunications and information technology.

The JTIT is dedicated to publishing research results which advance the level of current research or add to the understanding of problems related to modulation and signal design, wireless communications, optical communications and photonic systems, voice communications devices, image and signal processing, transmission systems, network architecture, coding and communication theory, as well as information technology.

Suitable research-related papers should hold the potential to advance the technological base of telecommunications and information technology. Tutorial and review papers are published only by invitation.

Manuscript. Papers published by invitation and regular papers should contain up to 15 and 8 printed pages respectively (one printed page corresponds approximately to 3 double-spaced pages of manuscript where one page contains approximately 2000 characters). An original and one copy of the manuscript should be submitted, each completed with all illustrations, tables and figure captions attached at the end of the paper. Tables and figures should be numbered consecutively with Arabic numerals. The manuscript should include an abstract about 100 words long and the relevant keywords. The abstract should contain statement of the problem, assumptions and methodology, results and conclusion or discussion on the importance of the results.
The manuscript should be double-spaced on one side of each A4 sheet (210 × 297 mm) only. Use of computer notation such as Fortran, Matlab, Mathematica, etc., for formulae, indices and the like is not acceptable and will result in automatic rejection of the manuscript.

Illustrations. Original illustrations should be submitted. Drawings in Corel Draw and Postscript formats are preferred. Colour illustrations are accepted only under exceptional circumstances. Lettering should be large enough to be readily legible when drawing is reduced to two- or one-column width, which often means shrinking to 1/4th of original size. Photographs should be used sparingly. All photographs should be delivered in electronic formats (TIFF, JPG, PNG or BMP). Page numbers including tables and illustrations (which should be grouped at the end) should be put in a single series, with no numbers skipped.

References. All references should be marked in the text by Arabic numerals in square brackets and listed at the end of the paper in order of their appearance in the text, including exclusively publications cited inside. Samples of correct formats for various types of references are presented below:

[1] Y. Namihira, "Relationship between nonlinear effective area and mode field diameter for dispersion shifted fibres", *Electron. Lett.*, vol. 30, no. 3, pp. 262–264, 1994.

[2] C. Kittel, *Introduction to Solid State Physics*. New York: Wiley, 1986.

[3] S. Demri and E. Orłowska,"Informational representability: Abstract models versus concrete models", in *Fuzzy Sets, Logics and Knowledge-Based Reasoning*, D. Dubois and H. Prade, Eds. Dordrecht: Kluwer, 1999, pp. 301–314.

Biographies and photographs of authors. A brief professional author's biography of up to 100 words and a photo of each author should be included with the manuscript. A printed photo, minimum size 35 × 45 mm, is acceptable. The photo may be supplied as image file.

Electronic form. TEX and LATEX are preferable, standard Microsoft Word format (.doc) is acceptable. The JTIT LATEX style file is available to authors. The file(s) should be submitted by e-mail or on a floppy disk or CD together with the hard copy of the manuscript. It is important to ensure that the diskette version and the printed version are identical. The diskette should be labelled with the following information: a) the operating system and word processing software used; b) in case of UNIX media, the method of extraction (i.e., tar) applied; c) file name(s) related to manuscript. The diskette or CD should be properly packed in order to avoid possible damage in the mail.

Galley proofs. Authors should return proofs as soon as possible. In other cases, the article will be proof-read against manuscript by the editor and printed without the author's corrections. Remarks to the errata should be provided within two weeks after receiving the offprint.

Copyright. Manuscript submitted to JTIT should not be published or simultaneously submitted for publication elsewhere. By submitting a manuscript, the author(s) agree to automatically transfer the copyright for their article to the publisher, if and when the article is accepted for publication. The copyright comprises the exclusive rights to reproduce and distribute the article, including reprints and all translation rights. No part of the present JTIT should not be reproduced in any form nor transmitted or translated into a machine language without prior written consent of the publisher.

A copy of the JTIT is provided to each author of paper published.

Journal of Telecommunications and Information Technology has entered into an electronic licensing relationship with EBSCO Publishing, the world's most profilic aggregator of full text journals, magazines and other sources. The full text of *Journal of Telecommunications and Information Technology* can be found on EBSCO Publishing's databases. For more information on EBSCO Publishing, please visit www.epnet.com.

(*Contents Continued from Front Cover*)

Editorial Office

National Institute
of Telecommunications
Szachowa st 1
04-894 Warsaw, Poland

tel. +48 22 512 81 83
fax: +48 22 512 84 00
e-mail: redakcja@itl.waw.pl
http://www.nit.eu

INSTYTUT ŁĄCZNOŚCI
PAŃSTWOWY INSTYTUT BADAWCZY