JOURNAL OF TELECOMMUNICATIONS AND INFORMATION TECHNOLOGY

1/2012

(*Contents Continued on Back Cover*)

ISSN 1509-4553 on-line: ISSN 1899-8852

Warsaw 2010

Circulation: 300 copies

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

JOURNAL OF TELECOMMUNICATIONS AND INFORMATION TECHNOLOGY

Preface

Intelligent computing is usually defined as advanced computing methods and techniques based on classical computational intelligence, artificial intelligence, and intelligent agents. On the other hand, large-scale distributed systems such as grids, peer-to-peer and ad hoc networks, and clouds enable the aggregation and sharing of geographically-distributed resources from different organization with distinct owners, administrators, and policies. With the advent of large-scale distributed systems, where efficient inter-domain operation is one of the most important features, it is arguably required to investigate novel methods and techniques to enable secure access to data and resources, personalization of the distributed and gathered information, efficient scheduling, self-adaptation, decentralization, and self-organization. The concepts of computing, data and information processing and multichannel visualization in intelligent large-scale distributed systems bring together the results from various research and application areas, making a positive impact on the development of new system architectures, routing and communication protocols and system management technologies.

This issue encompass eleven research papers reporting the recent findings, applications and developments of intelligent models and management, optimization and visualization techniques in modern large-scale scalable distributed environments.

Advances in network communication technology certainly open the way to a wide range of applications in engineering, industry, business and science, such environmental monitoring, traffic control, building management, etc. The effectiveness of the system and the reliability of the network nodes depend mainly on the network bandwidth and latency, network scalability and the communication protocols. V. Stanciu *et al*. have developed a network protocol for reliable communication from one point to multiple points, on possible multiple machines connected through Internet. Their approach is based on the transmission model of UDP in order to extend the conventional TCP-like services to one-to-many communication system and overcome the limitations of the methodologies with two logical endpoints transmission.

Recently, computational cloud has been recognized as one of the most popular type of intelligent distributed system. The term “Cloud computing” is used for the modern consumption and delivery model for IT services based on the Internet. It typically involves over-the-Internet provision of dynamically scalable and virtualized resources. R. Dębski *et al*. present an interesting integration of augmented cloud environment with the multi-agent system as the new, cost effective highly scalable execution environment dedicated for the large-scale computing systems. The authors span the cloud system beyond the data center borders by utilizing web browsers for the access to the evolutionary multi-agent system (EMAS), which plays the role of a service model (Agent Platform as a Service). They constructed an effective agent-based support mechanism for solving the system load management problem in cloud systems, where non-deterministic load changes are often monitored.

The low cost data transmission is a key issue of large-scale but also small-area networks. In Wireless Body Area Networks (WBANs) technologies the communication module and types of sensors are the core technologies and protocols in the whole system. E. M. E. Hassan and K. Mohammed have developed an embedded system for cardiac monitoring in WBANs. This system is mainly dedicated to the acquisition, storage, and transfer the warning messages on detected heart arrhythmia of humans by using the GSM technologies. This system is promoted as a low-cost support tool in the medical diagnostic processes.

An effective data management remains still challenging and crucial problem in large-scale distributed systems, especially in the cases of different access policies of the users, who work in different administrative domains and different operation systems, which is typical for computational grids and wide area networks. G. Gębczyński *et al.* present a framework for simulating the security-aware scheduling the dynamic grid environment. The grid simulator is an event-based application that allows to activate and deactivate various resolution methods and scheduling criteria in a flexible way.

In the next two papers the authors present the recent technologies in the multi-stream video transmission. Y. Ebara uses the tiled display wall technology for the transmission of a high-resolution video streaming in the large-scale display environment. He constructed a remote communication environment for the display wall design and has performed several experiments for tuning and the evaluation of the system. Another methodology of improving the quality of the video streaming is presented by F. de Asís López-Fuentes and C. Cabrera-Medina. They implemented a network coding model at the intermediate nodes of the network for a low-cost processing of the video packets. The proposed methodology differs from the conventional store-and-forward techniques in the encoding protocols. The video packets are encoded by the source nodes and the intermediate nodes implement these encoding protocols before the processing the data to the end-user nodes.

An interesting practical approach of multichannel communication and visualization system is presented by A. Gentile *et al.* The authors have developed the "QRouteMe" system for supporting the management and for tracking the visitors of the museums and any kinds of exhibitions. This system allows to configure a virtual guide for each user. The user can use various kinds of devices, from touch screen to mobile- and smartphones, for the communication and synchronization of the required information and processes. One of the most important features of the system is an auto-localization module, which is personalized for each active user in the system. The current location of the visitor is approximated by using the fiduciary mark reference points distributed in the building (museum or exhibition building). A silent feature of the presented model is an excellent solution for the personalization of the information systems, which is one of the hottest research topics in the management of the sheer-size scalable systems. Another practical example of the multicomponent system, in which the personalization is one of the most important issue is a system for supporting the criminal analysis presented by J. Dajda *et al.* The authors have designed a distributed heterogeneous platform for the low-cost processing of huge amounts of data. The high accuracy of the system and privacy together with the fast decentralized management of complex criminal databases are the main attitudes of the proposed technology.

The last three papers of the issue are focused on the generic models which can be the promising and potential solutions for realistic future generation scalable systems. E. Gajda-Zagórska presents an effective clustering model for the improvement of the meta-heuristic multiobjective optimizers in the distributed environments. C. V. Suciu *et al.* propose an intelligent model for optimizing the suspensions in modern green cars. Finally, A. Plichta has implemented the neural network model for improving the management of online information and knowledge processing.

We are grateful to all the contributors of this issue. We thank the authors for their interesting papers, their time and efforts in the presentation of their recent research results. We also would like to express our sincere thanks to the reviewers, who have helped us to ensure the quality of this publication.

Joanna Kołodziej
Guest Editor

Paper

Design and Development of a UDP-Based Connection-Oriented Multi-Stream One-to-Many Communication Protocol

Valentin Stanciu, Mugurel Ionuţ Andreica, Vlad Olaru, and Nicolae Ţăpuş

Politehnica University of Bucharest, Bucharest, Romania

Abstract—A communication protocol is a set of rules defined formally that describes the format of digital messages and the rules for exchanging those messages in or between computing systems. The Internet Protocol Suite used for communications throughout the Internet uses encapsulation to provide a way of abstracting protocols and services. This abstraction is grouped into layers of general functionality. For protocols on the transmission layer, many choices exist. But while popular protocols such as TCP, UDP and SCTP do provide connection oriented communication offering reliability, ordering and data integrity, solutions that offer such connections from one point to multiple endpoints are still limited. TCP only supports point-to-point communication and SCTP offers multi-homing functionality, but the transmission is still limited to two logical endpoints. In this paper we use the simple, stateless, transmission model of UDP in order to provide TCP-like services for one-to-many communication that is not limited to just multi-homing or other particular solutions. The protocol supports reliable communication from one endpoint to multiple endpoints in different transmission modes. In order to make it easier for developers to customize the protocol to their needs and possibly extend/modify it in order to create new variants from it, the protocol is developed in user space. Because of this design restriction performance wasn't the main objective of our work, but rather the ease of customization and experimentation with new protocol variants. The protocol was implemented in the C++ programming language using classes with virtual members. New variants of components, such as packet retransmission, can easily be implemented without changing the whole code base.

Keywords—communication protocol, connection oriented, multiple streams, one-to-many.

1. Introduction

The work presented in this paper consists of the design and development of a network protocol for reliable communication from one point to multiple points, on possible multiple machines.

The main motivation behind this work is to provide enhanced communication functionality from one endpoint to multiple endpoints. It should be noted that similar transport layer protocols such as TCP and SCTP don't provide the kind of functionality that this protocol provides. SCTP's multi homing support only deals with communication between two endpoints which are assigned multiple IP addresses on possibly multiple network interfaces; it does not deal with configurations that contain multiple endpoints (for example, clustered endpoints). Our work allows an application running on a machine to connect to a collection of machines as if they were a single one. It practically virtualizes a set of machines under the same endpoint, each machine being accessible under many streams. Using this approach, one can implement features such as load balancing, which is absent in SCTP.

Similar to SCTP, our protocol supports multiple streams inside each connection. This is an improvement to TCP's single-stream connections, as using multiple streams has the advantage of better parallelization that leads to better performance in the context of today's multi-core processors.

Using the one-to-many facilities and the support for multiple streams, new programing models for network connections and new design patterns can be created. This allows for easier application implementations and shorter development times for advanced functionality.

The common way of developing a networking protocol is to implement parts of it in kernel space. This not only splits the implementation into two parts (kernel space and user space), but also makes its configuration, tweaking and portability to multiple operating systems more difficult. To alleviate such problems, we decided to implement the protocol only in user space. Even if performance may be affected because of this decision, it is beyond the scope of this paper to provide fast absolute performance. The protocol uses the UDP transport protocol, on top of which it implements the desired connection-oriented functionality.

The rest of this paper is organized as follows. In Section 2 we discuss related work. In Section 3 we present the design of our one-to-many communication protocol and in Section 4 we provide implementation details. In Section 5 we present experimental results. Finally, in Section 6 we conclude and discuss future work.

2. Related Work

2.1. Transmission Control Protocol (TCP)

One of the core protocols of the Internet Protocol Suite is the transport layer protocol name Transmission Control Protocol (TCP) [1]. Complementing the Internet Protocol (IP), it is one of the two original components of the Internet Protocol Suite, and the reason the entire suite is commonly referred to as TCP/IP. TCP's design is for use as a highly reliable host-to-host protocol between hosts in computer communication networks.

TCP provides reliable inter-process communication between pairs of processes in host computers attached to distinct but interconnected computer communication networks. Very few assumptions are made as to the reliability of the communication protocols below the TCP layer.

Because of the wide spread adoption of TCP and because the facilities offered by it are well known and have been tested thoroughly over the years, a comparison with TCP is inevitable. Our protocol implements the features offered by TCP and extends them to one-to-many connections.

2.2. User Datagram Protocol (UDP)

User Datagram Protocol (UDP) [2] is one of the most commonly used protocols of the Internet Protocol Suite. UDP's simple transmission model without implicit connectivity provides a fast way to transmit data. One of the features of UDP is multicast addressing. Multicast is the delivery of a message or information to a group of destination computers simultaneously, in a single transmission from the source. The problem is that this form of communication is not reliable and messages may be lost or delivered out of order. Moreover, the multicast facility requires support from the underlying network devices, which is rarely available.

Our protocol is built over UDP because of its simple, performance-oriented design and offers the functionality of multicast with the reliability of TCP, as well as other forms of one-to-many addressing. It is worth mentioning, though, that in UDP's multicast model, the sender does need to know all the receivers (but only a group identifier), while our multicast model requires explicit knowledge of each receiver's network address.

2.3. Stream Control Transmission Protocol (SCTP)

The Stream Control Transmission Protocol (SCTP) [3] is a new protocol existing at an equivalent level with UDP and TCP on the protocol stack and provides transport layer functions to many Internet applications. SCTP has been approved by the IETF as a proposed standard in 2000 and updated over the years. SCTP is a reliable transport protocol operating on top of a connectionless packet network such as IP. It offers the following services to its users:

– acknowledged error-free non-duplicated transfer of user data,

– data fragmentation to conform to discovered path MTU size,

– sequenced delivery of user messages within multiple streams,

– an option for order-of-arrival delivery of individual user messages,

– optional bundling of multiple user messages into a single SCTP packet,

– network-level fault tolerance through supporting of multi-homing at either or both ends of an association.

Similar to TCP, SCTP offers a reliable transport service. SCTP makes sure that data is transported across the network without errors even if packet loss is possible and that the data arrives in the correct sequence. Similar still to TCP, SCTP creates a relationship between endpoints prior to data being transferred. This relationship denotes what is called a "session-oriented" mechanism for transmitting data. In SCTP terminology, such a relationship is called an association and is maintained until all data has been successfully transmitted.

SCTP improves upon the TCP design by adding support for message-based, multi-streaming, multi-homing delivery of chunks without head-of-line blocking, path selection and monitoring, validation and acknowledgment with protection against flooding and improved error detection.

Unlike TCP which is byte-oriented, SCTP is message oriented and supports framing of individual message boundaries. Data is sent as being part of a message and sent on a stream in the form of a packet. Error detection and correction is also resolved at the message level.

2.4. Other Communication Protocols and Techniques

Some of the underlying principles used in the design of our protocol were first mentioned (as concepts) in [4]. In [5] a (multi)point-to-(multi)point communication protocol was proposed which uses delay for congestion control, rather than a congestion window (like TCP, SCTP and our protocol). In [6] a one-to-many communication method based on constructing an application-aware overlay was proposed. This differs from our approach, in the sense that an overlay needs to be constructed and maintained. In [7] the authors maintain the idea of constructing an overlay for multicast communication, but this overlay is hidden "under" an "overlay socket". In the sense of introducing new types of sockets, this approach is similar to ours.

In regard to the user space implementation of our communication protocol, many previously proposed protocols were also implemented in user space [5] [6], [7]. Recently, an Internet draft proposal [8] was published for encapsulating SCTP packets in UDP packets (i.e., implementing SCTP in user space over UDP), in order to address SCTP's lack of kernel-level implementation availability and NAT traversal issues.

3. Protocol Design

There are many design paths to take when developing a user space networking protocol. The first option considered was implementing the protocol via overloading existing functions and using callback mechanisms. However, this makes the code hard to follow and even harder to extend or tweak particular functionalities.

To make the implementation of a communication protocol easier and to have better defined functions for certain tasks, the protocol needs an entity that would always run and send/receive messages. This can be either a separate process or a separate thread. A separate process would require intensive IPC (Inter-Process Communication) and synchronization, while a separate thread would mean every application that uses the protocol will each have its own thread that performs communication.

Because of the separate logic for sending and receiving data, it was decided that there would be 2 execution units for communication. Only one unit would mean that certain bottlenecks may occur when both receiving and transmitting large chunks of data. This case also increases in frequency since for every packet sent, an ACK packet may be required to be received.

Furthermore, because of the nature of networking protocols, some packets will be lost and a good protocol needs a way to handle such cases. In order to provide this functionality, a timeout mechanism is required. The mechanism needs to be as fine grained as the operating system permits and not block the client application that uses the protocol or one of the transmission execution units. Thus, another execution unit is needed with the sole purpose of providing timeout functionality.

In Linux, a socket descriptor is just a number that has meaning only to the process who owns it. The user space side has just a number and the kernel maps the pair *(number, process)* to a particular kernel socket. If a process transfers the socket number to another process, the socket descriptor becomes invalid as the kernel doesn't know the socket descriptor got copied to the new process. To copy a socket descriptor between processes, Unix sockets have to be used and the descriptor copied through them. However, this form of socket management quickly proves to be overly complicated to synchronize. When one process modifies the socket attributes, it has to announce the other processes that have a copy of the same socket in other to maintain consistency.

Because implementing the separate execution units as processes requires more advanced and tangled synchronization and communication, the execution units were implemented as threads. The total memory usage of the applications using the protocol will increase as each one will have a copy of the 3 execution units used for transmission, but networking speeds will not necessarily be impacted. The kernel resolves the problem of multiplexing different communications and N processes transmitting data may be even faster than just one process transmitting data.

The only downside of this decision is the inability to implement QoS (Quality of Service) – like functionality in the protocol. These services require knowledge of all (or as much as possible) data transmitted from a machine so even if every application using our protocol used a common engine for transmission, applications using other protocols like TCP would interfere with QoS.

3.1. Protocol Operation

The primary purpose of the protocol is to provide a reliable logical circuit or connection service between one to many endpoints. To provide this service on top of a less reliable communication system the following facilities are required:

Basic data transfer

The protocol packs some number of octets into packets for transmission through the networking system. This way, the protocol is able to transfer a stream of octets in each direction grouped into messages.

Reliability

In case data is damaged, lost, duplicated or delivered out of order, the protocol must recover and never enter an unknown state. This is achieved by assigning a sequence number to each packet transmitted and requiring an acknowledgment (ACK) from the receiving end. If the ACK is not received within a timeout interval, the data is retransmitted. The sequence numbers are also used by the receiver to correctly order packets that may be received out of order or to eliminate duplicates. Damage is handled by having a checksum the end of the packet header, calculated by the sender and checked by the receiver, who discards damaged packets.

Multiplexing

To allow for many execution units within a single host to use the communication facilities simultaneously, the protocol provides a set of streams within each process, further developing on the address and port identification elements.

The protocol must allow for multiple distinct communications to take place on the same machine or by the same process, but each one must use a different source endpoint.

Connections

To obtain the reliability and flow control mechanisms described above, the protocol initializes and maintains certain status information for each stream. Multiple streams with the same address and port source form a connection. Each connection is uniquely specified by endpoint addresses, ports and streams.

The protocol must first establish a connection (initialize the status information on each side) before two processes can communicate and when that communication is complete, the connection is closed and the used resources freed. In order for the connections to be established over unreliable communication systems, a handshake mechanism with timeout-based sequence numbers is used to avoid erroneous initialization of connections.

Multiple endpoints

Similar protocols only implement reliable communication between two endpoints or unreliable communication from one to many.

The protocol provides a way to communicate with multiple hosts at the same time while offering the facilities of a point-to-point connection oriented communication.

In this form of communication, a retransmission model must be chosen. Either one or all of the peers must acknowledge packets so retransmission will not occur.

Congestion control

Congestion control is implemented the same way as for TCP, by using a congestion window representing the maximum number of packets (or of bytes) sent but not yet acknowledged. The difference from a standard point-to-point protocol is the definition of a packet being acknowledged (e.g., depending on the transmission mode, a packet is acknowledged if one, some or all of the destinations acknowledge the packet).

3.2. Sequence Numbers

One of the main concepts in the protocol design is that a sequence number is assigned to every packet sent over a connection. This is similar to every other protocol that provides reliable communication. Because every packet has a sequence number, each and every one of them can be acknowledged and this allows for detection of duplicates or lost packets. Every stream of the communication has a sequence number for each direction.

It is essential to remember that the actual sequence number space is finite, from 0 to $2^{32}-1$. Since the space is finite, all arithmetic dealing with sequence numbers must be performed modulo 2^{32}. If this protocol is extended, a sliding window can never be larger than 2^{31}.

Using sequence numbers, the protocol must perform the following operations:

- determine that an incoming packet contains a sequence number that is expected,
- determine that an acknowledgement number refers to a sent packet with a sequence number not yet acknowledged.

When an acknowledgement has been received for all sent sequence numbers, the protocol can conclude that all packets have been sent successfully.

3.3. Automatic Repeat Request

Automatic repeat request (ARQ), also known as *Automatic Repeat Query*, is an error-control method for data transmission that uses acknowledgements and timeouts to achieve reliable data transmission over an unreliable service. If the sender does not receive an acknowledgment before the timeout, it usually retransmits the packet until the sender receives an acknowledgment or exceeds a predefined number of retransmissions.

Our protocol uses a variant of the *Go-Back-N* ARQ for transmission. In *Go-Back-N*, the sending process continues to send a number of packets specified by a window size even without receiving an acknowledgement packet from the receiver. It can be viewed as a particular case of a sliding window protocol with the transmit window size of N and the receive window size of 1.

If the sender sends all the packets in its window and receives no ACK or only future or past ones, but not the expected one, it will go back to the sequence number of the last valid ACK it received from the receiver, fill its window starting with that frame and continue the process over again. Only the expected ACK will advance the expected sequence number. Past "duplicate" ACKs or future "out-of-order" ACKs will be ignored.

Go-Back-N ARQ is more efficient than Stop-and-wait ARQ since unlike waiting for an acknowledgement for each packet, the connection is still being utilized as packets are being sent. In other words, during the time that would otherwise be spent waiting, more packets are being sent. This method also results in one frame possibly being sent more than once and the receiver has to be aware of duplicates and discard them.

If the highest possible throughput is desired, it is important to force the transmitter not to stop sending data earlier than one round-trip delay time (RTT) because of the limits of the sliding window protocol. In order for the protocol not to limit the effective bandwidth of the link, the limit on the amount of data that it can send before stopping to wait for an acknowledgement should be larger than the bandwidth-delay product of the communication link.

The protocol can be configured in the initialization stages on how to handle retransmission in a multiple endpoint communication. In the *broadcast mode*, all the peers must acknowledge the sent packets before transmission can move forward. In the *any-cast mode*, at least one peer must acknowledge the packet. In both of these modes, each packet is actually sent to all the endpoints. In the more general case, at least P destinations must acknowledge the packet in order to consider it fully acknowldeged at the source.

We also implemented a mechanism in which a packet must be sent to at least a number K of the destinations ($K \leq$ the number of destinations). In this case, a packet may be considered acknowledged either if all, at least one or, more generally, at least $P \leq K$, of the destinations to which it was sent acknowledge it.

When we do not need to send each packet to each destination, we also implemented a load balancing mechanism, in

which the (K) destinations for the packet are selected based on a load metric computed at the source for each destination (e.g., a combination of average response time for the packets sent to it during the last few seconds/minutes and the number of lost packets sent to it in the same time interval; the response time is defined as the time difference between the moment when the packet was sent and the moment when its corresponding ACK was received).
Note that when a packet is sent to multiple destinations, each copy of the packet is handled by the sender as a separate, different packet for timeout and retransmission purposes.

3.4. Header Format

As with any protocol implemented over the network stack, each layer encapsulates the ones above. In computer networking, encapsulation is a method of designing modular communication protocols in which logically separate functions in the network are abstracted from their underlying structures by inclusion or information hiding within higher level objects. The UDP protocol encapsulates our protocol header and adds destination and source addresses and destination and source port numbers. A description of the contents of the header follows:

- Version (8 bits): The version field indicates the format of the protocol header.
- Source stream (8 bits): The source stream number.
- Destination stream (8 bits): The destination port number.
- Flags (8 bits): Flags that indicate if the packet starts or ends a message or if it is the first packet sent on this connection.
- Sequence number (32 bits): The sequence number of the first data byte in this segment. If this is the first message sent, this number is the initial sequence number and the first data byte is this number plus one.
- Acknowledgment number (32 bits): The value of the next sequence number the sender of the segment is expecting to receive.
- Data payload size (32 bits): Size of the data contained in the packet.
- Cyclic redundancy check (32 bits): The checksum field is the 32 bit one's complement of the one's complement sum of all 32 bit words in the header.
- Payload data (variable length): The actual data the application wants to transmit.

3.5. Connection Open

When opening a connection, for each stream of this connection a "three-way handshake" procedure similar to TCP's

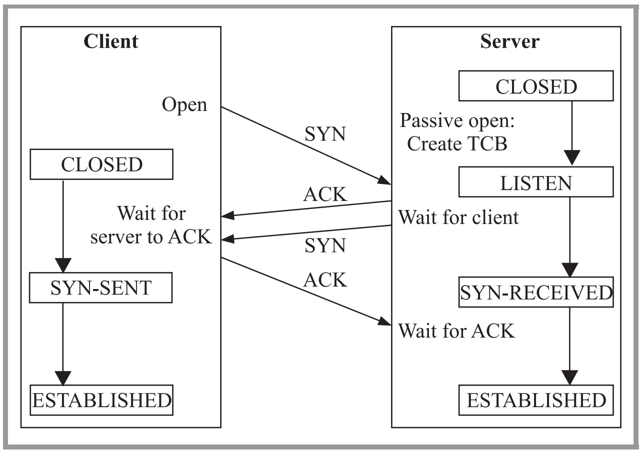


Fig. 1. Connection open.

Connection Opener is used. This procedure is normally initiated by the starting point and responded by the endpoint. In a multiple endpoints environment, each stream has its own independent opening. The three-way handshake reduces the possibility of false connections. It is a trade-off between memory and messages to provide information for this checking. The three-way handshaking works as follows:

1. A SYN (Synchronize) segment (as indicated by the bit flag) containing a 32-bit sequence number A called the Initial Send Sequence (ISS) is chosen by, and sent from, the starting point (Host 1). This 32-bit sequence number A is the starting sequence number of the data in the packet and is incremented by 1 for every byte of data sent within the segment. The SYN segment also places the value A+1 in the first byte of the data.

2. Host 2 (the destination) receives the SYN with the sequence number A and sends a SYN segment with its own totally independent ISS number B in the sequence number field. In addition, it sends an increment on the sequence number of the last received segment in its acknowledgment field. This stage is often called the SYN-ACK. It is here that the MSS is agreed.

3. Host 1 receives this SYN-ACK segment and sends an ACK segment containing the next sequence number. This is called Forward Acknowledgement and is received by Host 2. The ACK segment is identified by the fact that the ACK field is set.

Protocol peers must not only keep track of their own initiated sequence numbers but also those acknowledgement numbers of their peers. When connecting to multiple hosts, a similar procedure is followed for each endpoint and the starting host has a separate sequence number and acknowledgment number for each peer. Distinction between streams is provided by the stream destination field.

Normally, our protocol has one separate stream for each destination. However, there is no problem adding more streams for some of the destinations by using the API.
Because of the security vulnerability of this procedure, the three-way handshake is planned to be changed with a four-way cookie-based handshake similar to SCTP in a future version of the protocol.
In the rest of this paper, we will define the roles of "client" and "server" as follows. The client is the node initiating the connection and the server is the one accepting a connection. Thus, in the case of our protocol, the client is the source node and each destination node acts as a server. These concepts are also used in Fig. 1 which depicts the steps of the connection opening procedure.

3.6. Connection Close

In the case of a normal connection close, each side terminates its end of the connection by sending a special packet with the FIN (finish) flag set. The packet is called a FIN message and serves as a connection termination request to the other device. The device receiving the FIN responds with an acknowledgment (ACK) and a FIN to indicate that it was received and it is ready to close.

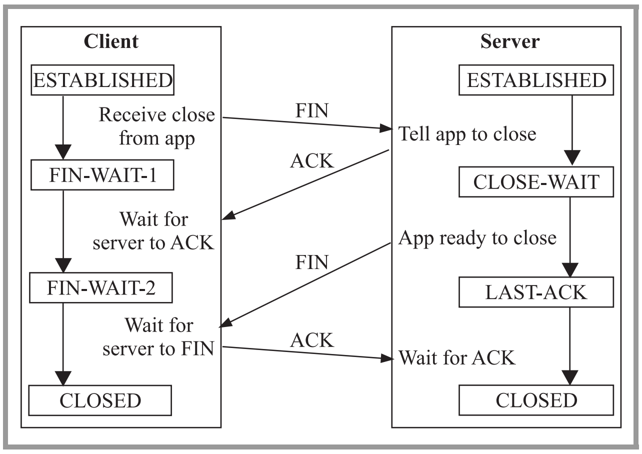


Fig. 2. Connection close.

The connection as a whole is not considered terminated until both sides have finished the shutdown procedure by sending a FIN and receiving an ACK for each stream belonging to that connection. This procedure is similar to the normal termination of a TCP connection, but repeated for every stream (see Fig. 2).

3.7. Communication System Architecture

The core of the communication system consists of three execution units: a sender, a receiver and a timer. These execution units send and receive data from the Internet and deposit them to buffers in each of the managed streams; connections can have more than one stream for transmission. Each execution unit is actually implemented as a thread pool. There are no other requirements regarding them (e.g., the thread pool may contain a fixed number of threads or it may adjust its number of threads dynamically, according to needs).
The sender unit is tasked with checking client send buffers for any data to be sent, pack it in the correct format and send it over the network. The receiver thread waits for data from the network, reads it, unpacks it and puts it into the receive buffer of the appropriate client.
When a packet is sent to the network, the sender also tells the timer to announce it after a certain period has passed. Once an ACK for the packet has been received, the timer is told to cancel that announcement. If no packet is received, the timer does notify the sending unit and that unit implements a retransmission algorithm.
From the point of view of the receiver and sender, each stream can be considered as an individual client. Each stream has its own buffers and is identified by the stream id in the header of each packet.

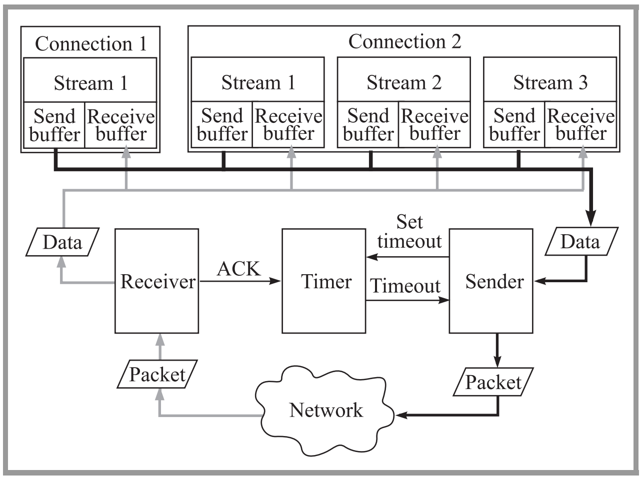


Fig. 3. Communication system architecture.

A general view of the architecture is presented in Fig. 3. Thick lines represent data being moved around. Thick grey lines show data that is received by clients while data in thick dark lines illustrate data sent by the clients. Even though in the figure lines from streams interconnect, streams have no way of seeing data from other streams.
When the application uses the API to send or receive data, it only accesses the local stream buffers. For sending, data is inserted into the send buffer (and later picked up by the sender unit). For receiving, data is fetched from the receive buffer (which was placed there by the receiver unit).
When the application tries to receive data from the network, but the receiving buffer is empty, a wait is performed on a conditional synchronized variable until the operation can be completed. The same thing happens when the application tries to send data, but the send buffer is full.
From the point of view of the receiver, if data is received from the network, but the receive buffer of the stream is full, the data is discarded and no ACK is sent. This will force the sender of the data to retransmit at a later time, when the application might have read the data and emptied the receive buffer.

4. Communication System and Protocol Implementation

The code is divided into 3 components: The API (Application Programming Interface) class, the execution units implementations and the global components. The API is implemented as a single C++ class to provide all the functionality in one place. Because the actual work of the protocol is done in an asynchronous fashion, the execution units tasked with transmission are separate from the API. These thread pools can be considered the core of the communication system and they are not part of a class or container as each thread is individual and only performs work related to itself. The only time when the core needs to interact with other entities is when exchanging data. These exchanges are governed by global synchronization mechanisms.
Besides the API and the core threads, there are also common components used by both parts. These contain both miscellaneous functionality such as CRC calculation and protocol particular structures like packet queue definition. They are called global (with respect to the protocol) because they are used by different parts of the code base and different components need access to them.
To implement this protocol extensive use of threads was required and a lot of inter-thread synchronization. To achieve this, the POSIX thread library *pthread* was used.

4.1. Linux POSIX Threads

The POSIX thread library contains a standards based thread API for C/C++. It allows one process to spawn a new concurrent execution flow. It is most effective on multiprocessor or multi-core systems where the process flow can be scheduled to run on another processor thus gaining speed through parallel or distributed processing. All threads within a process share the same address space. A thread is spawned by defining a function and its arguments which will be processed in the thread.
The threads library provides three synchronization mechanisms:

- Mutex (mutual exclusion lock) – enforces exclusive access by a thread to a variable or set of variables.
- Join – makes a thread wait until another thread is complete (terminated).
- Condition variable (data type *pthread_cond_t*) – blocks a thread's execution until a condition is met.

Mutexes are used to prevent data inconsistencies due to operations by multiple threads upon the same memory area performed at the same time or to prevent race conditions. A contention or race condition often occurs when two or more threads need to perform operations on the same memory area, but the results of computations depend on the order in which these operations are performed. Mutexes are used for serializing shared resources such as memory. Anytime a global resource is accessed by more than one thread the resource should have a Mutex associated with it. All global variables accessed by the protocol are accessed only after first obtaining a mutex that governs them.
A join is performed when one thread wants to wait for another thread to finish. The project only uses joins when the last connection of the process is closed and the protocol threads used for transmission need to be closed. A join is used to make sure the threads finish sending/receiving all the data and then exit gracefully.
A condition variable is a variable of type *pthread_cond_t* and is used with the appropriate functions for waiting and, later, continue processing. The condition variable mechanism allows threads to suspend execution and relinquish the processor until some condition is true. A condition variable must always be associated with a mutex to avoid a race condition created by one thread preparing to wait and another thread which may signal the condition before the first thread actually waits on it, thus resulting in a deadlock. The thread will be perpetually waiting for a signal that is never sent. Any mutex can be used; there is no explicit link between the mutex and the condition variable. Condition variables are used a lot throughout the code base. Waiting for a packet to be received is done via waiting for a condition to be met. Another example is the sending thread that waits for data to be sent from any of the clients.

4.2. Application Programming Interface

To make use of the protocol, an API is provided. The API is implemented in C++ using classes with virtual members. If another developer wants to extend or tweak the functionality of the protocol (s)he can do so by extending a class and changing its virtual members. It is also designed to be similar to the POSIX standard for sending and receiving packets.

4.2.1. Init

This function is called without parameters and used for initializing local communication members (i.e., the communication system). This is the first function the application must call before it can begin opening or accepting connections. Failure to initialize the communication channel will lead to inability to send or receive data. The sending thread(s), the receiving thread(s) and the timer thread(s) are created and their ids are stored globally.

4.2.2. Stop

This function is called without parameters and signals that the communication system should be stopped. The execution units are stopped.

4.2.3. Connect

This function is called with two parameters: a vector of hostname and port pairs and connection flags. If the application uses the connection as a client, it must specify to what hosts it will communicate. This function receives a vector of hostname and port pairs and connects to all of them, encompassing them in a single connection. A number of streams (by default 1) is opened for every endpoint and the Connection Open handshake is executed. Currently a TCP-like three way handshake is used. The connection flags explain the retransmission model to be used in case of packet loss for one of the connected peers. Possible values include:

- CANY to indicate that the sent message must reach at least one connected peer. In case the message is lost for some of them, but one peer still received it and sent an ACK, transmission will continue normally. In case no ACK is received, the packet is retransmitted for all peers.

- CALL to indicate that the sent message must reach all of the connected peers. All peers must receive the message and reply with an ACK. In case an ACK is not received from some peers, the message is retransmitted for those peers only.

- CBAL to indicate that the load balancing mechanism should be used.

Since all streams are independent, a common synchronization structure is created for the streams under the same connection. This will enable the retransmission behavior described above. A communication channel object is created and stored internally and a pointer to it is returned by the function.

4.2.4. Close

This function is called on a communication channel. When the application finished sending or receiving data on the communication channel, it has to call this function. This will enable all the transmission buffers to be emptied and close the communication channel. All the streams inside this channel from all associated endpoints will be closed. It will also close the operating system UDP socket and enable it for reuse. The function also removes the connection from the global client list.

4.2.5. WaitForClient

This function is called on a communication channel, with one parameter: port number. If the application uses the connection to act as a server, it must specify a port onto which clients will connect. This function listens for data on that port and then waits for a client to connect to it. The client and the server establish the connection via the connection open handshake. Currently a TCP-like three way handshake is used. When a client successfully connects to the server, a communication channel object is created and a pointer to it is returned.

4.2.6. Send

This function is called on a communication channel with the following parameters: pointer to message, message length and flags. The application instructs the protocol to send a message of a specific size to the connected peers. If the application is acting as a client, the message is sent to the ones specified at connect, following the retransmission model described by the connect flags. If the application is acting as a server, the message is sent to the client that connected to it in the WaitForClient method. If the message is sent as one big chunk and is lost, the protocol retransmits the message again. For large messages this will prove inefficient. Therefore, the message is first broken into chunks of a maximum size MaxData and packets are made with each chunk. Also, markers are placed for the start of the message and the end of the message in the corresponding packets to enable the reconstruction of the message when received. The flags parameter is used to specify different communication behaviors. The function returns only after all the data was placed into the send queue/buffer (a condition variable is used for waiting until room is available in the send queue/buffer).

4.2.7. WaitForSend

This function is called on a communication channel with one parameter: stream number. The send function only inserts data into the queues and returns. The actual sending of the data is achieved asynchronously. This function is used if the application desires to block until the data is sent. It waits on a synchronized condition until all the packets inside the sending queue associated with the stream have been sent and acknowledged.

4.2.8. Receive

This function is called on a communication channel with the following parameters: pointer to buffer to store the message, length of buffer, stream number and flags. The application instructs the protocol to retrieve a received message from a certain stream and store it in the buffer passed as a parameter. This function blocks until a new message is ready to be given to the application. This implies waiting until all the packets corresponding to that message have been received and stitching them back together using the start of message and end of message markers. The flags parameter is used to specify different communication behaviors.

4.2.9. AddDestination

This function is called on a communication channel in order to add a new destination to it. A new stream towards the destination is created and the connection open procedure is used. If the connection already contains this destination as its peer, the effect is that a new stream is created towards the destination.

4.2.10. RemoveDestination

This function is called on a communication channel in order to remove a destination from its peers. The stream towards that destination is closed. If there are multiple streams towards the destination, only one of them is closed.

4.2.11. FullyRemoveDestination

This function is similar to the previous one, except that all the streams towards the given destination are closed.

4.3. Internal Members

4.3.1. Send Queues

These queues act as a buffer space where data from the client application is stored before the sending threads pick them up. Each stream has its own queue and the API function Send inserts data here. There exists one sending queue for each communication channel. Moreover, there is one sending queue for each stream.

4.3.2. Receive Queues

These queues act as a buffer space where data for the client application is stored after the receiving threads get data from the network. Each stream has its own queue and data resides here until the client application decides to make an explicit read for the data using the Receive function. There exists one receive queue for each stream of a communication channel.

4.3.3. Socket Descriptor

The socket descriptor is an abstraction used by the operating system socket to transmit data. Each UDP "connection" has a socket descriptor and each comunication channel uses just one socket for a connection, regardless of how many peers it has connected to it. All communications send by the protocol for this connection pass through this socket. All the streams use the same socket. It is initialized when a connection is first opened, either via a Connect or via a WaitForClient function call. The socket will remain valid until the connection is closed via a Close call.

4.3.4. Destination Address Vector

A vector containing destination structures particular to the operating system. They describe the communication peers. It is initialized in the Connect or WaitForClient functions and modified in the AddDestination, RemoveDestination and FullyRemoveDestination functions. There exists one such vector for each communication channel.

4.3.5. Synchronization Primitives

These primitives are used for synchronization with the protocol threads that do the actual communication. In case of synchronized network communication, the client thread has to wait for the sending or receiving thread(s) to finish transmission. A synchronized network transmission always occurs in the handshake as no communication is possible until it has completed successfully. The synchronization is achieved through mutexes and condition variables. Whenever the application has to wait for data to be sent, it waits for the send condition to be fired. Whenever the receiving thread obtains an ACK for a packet and there are no more packets to be sent for that particular stream, a send condition is broadcasted. If the application wants to receive data, but there is no more data in the receive queue, it waits until a receive condition is fired. Whenever a receiving thread adds more data for a particular stream, it also broadcasts a receive condition. Each communication channel has its own set of synchronization primitives.

4.4. Global Elements

In order for the threads to know what clients (communication channels) are managed, a list of all the clients must be stored. Access to this list must also be synchronized and also flags must be raised when the process exists. All streams use a packet queue for receiving and another for sending. These queues are accessed by both the API and the transmission threads and access to them must be synchronized. Additional operating system elements must also be stored globally to be accessed from all components of the protocol.

4.4.1. Client List

A global client list needs to be maintained for the sending threads to know where to look for data to be sent and for receiving threads to know where to insert captured data from the network. This list is stored in a vector available to all the elements of the protocols and contains pointers to the protocol client classes. Whenever a new connection is initiated/accepted, the Connect/WaitForClient function call inserts a pointer to the created communication channel class here. Access to this list is synchronized via a mutex.

4.4.2. Thread Flag

The thread flag is a special global variable is provided globally and always accessed by the internal threads before any new round of operations is started. This flag is used to signal when the internal components of the protocol needs to stop. Other notifications can also be implemented in the future via this flag, for example temporarily stopping the protocol for a temporary duration. Access to this flag is granted only after acquiring the global resource access mutex.

4.4.3. Packet Queues

The packet queue is one of the most important data structures of the protocol. It is designed based on the producer-consumer model where one entity inserts data into the queue and another extracts data from the queue. From an implementation perspective we can compare the queue to a circular buffer where elements are always inserted in a clockwise direction and always the first element removed is the first element inserted. The queue holds packets and it is required that all the elements inserted into the queue have successive sequence numbers. Thus, the packet queue takes care of packet sequences and assigns a sequence number to every inserted element. To keep track of the elements inside the queue, 3 markers are used. One marker shows the next element to be removed, another marker shows the next position a new element will be inserted at and a third, send queue specific marker, shows the next packet to be sent.

There is a difference here on how the queue is used. If it's used for a sending buffer, the API functions insert elements into the queue where the second marker is positioned and increment it every time. When the sending thread wants to transmit a new packet, it takes the one pointed by the third marker, sends it, and increments the third marker. An element is removed from the queue only if the receiving thread obtains an ACK from the network for that particular packet and thus increments the first marker. When the queue is used as a receiving queue, only two markers are used. The receiving thread inserts into the queue a new packet and increments the second marker. The API function that receives elements removes packets from the queue and increments the first marker.

Since there is more than one execution unit accessing the data structure, all operations with the data structure is synchronized with an access mutex. Whenever an element is inserted or a marker incremented, the mutex must first be acquired and then released afterwards. Because of performance reasons, the queue must have an upper limit for how many elements it can hold. If an attempt is made to insert elements in a full queue, a return code of *QUEUEFULL* is returned. Furthermore, queries can be done to check if the queue is full or empty.

Given the circular nature of the queue, all marker operations are done modulo the size of the queue. The current size is 128 elements. This may be increased or decreased to obtain better performance according to experiment results. The size of the queue is practically the size of the sliding window in the *Go-Back-N* protocol. No more packets will be sent if there are already 128 sent that haven't received an acknowledgement. Once an ACK is received, the element is removed from the sliding window (from the packet queue).

4.5. Transmission Threads

To properly process packets and prepare them for sending over the network or receiving them from the network three types of threads are implemented. Each thread is implemented as a function that executes in a continuous while loop. In order to limit 100% processor usage and avoid a busy-waiting situation, synchronization mechanisms are implemented that wake a thread only when there is work for it to do.

4.5.1. Sender Thread

Since we may use multiple sender threads, we considered the following inmplementation. There is a global data queue, from which each sender thread extracts data in a synchronized manner. When a new packet is placed in the send queue of a stream, information regarding the communication channel and stream number are placed in the global data queue (e.g., a pointer to the stream's send queue). The global data queue's condition variable is signalled and a sender stream is woken up (if it was waiting at the global data queue's condition variable). Each sender stream continuously checks if there are any elements in the global data queue. If there are no elements, then it waits on the queue's condition variable. Otherwise, it removes the first element from the global data queue. Afterwards, using the extracted information, the sender stream sends the packet identified by the information from the global data queue to its destination.

After a packet is sent, the timer thread is announced of the need for a timeout after a certain amount of time. The current value of the timeout is 50 ms. Retransmission occurs if an acknowledgement is not received before the timeout is fired.

4.5.2. Receiver Thread

A receiving thread waits for data to be available for reading from any network socket the protocol manages. The UDP sockets are distributed among the existing receiving threads in a balanced manner. Dynamic redistribution is also possible (e.g., if a thread is very busy with receiving packets from some sockets, then some of the other sockets may be redistributed to other receiving threads, in order to avoid starvation or simply to improve performance), but was not implemented in out protocol.

Waiting is achieved using the provided operating system function called "select". If there is nothing to be read, the thread blocks and the execution scheduler gives CPU time to another thread. Because of the need for the thread to respond to events such as application shutdown, this waiting is not indefinite. Waiting is limited to 100 ms. After every waiting round, the thread checks if it needs to respond to any miscellaneous event such as shutdown.

Whenever a packet is received, the ACK message corresponding for that packet is prepared for sending. Currently the protocol sends an ACK message for each message it receives, but future iterations of the protocol can wait for multiple packets to be received and only send one ACK for the whole group.

4.5.3. Timer Thread

The timer thread acts as a ticking clock and constantly processes events every defined interval of milliseconds. This interval was chosen at 50 ms. More fine values are supported, but the protocol needs to consider other processes that use CPU time. Because a clock that ticks without anybody listening to it is not efficient, the timer is only active while there is a packet timeout that needs an alarm. The timer maintains a list of timeout events that will be fired in the future. If this list of events is empty, the timer waits on a synchronized condition. The API of any client wanting to send data also signals the timer thread and unlocks it if it was waiting for this condition.

Because of the way Linux threads are implemented, *sleep* cannot be used. Using *sleep* will cause the whole process to wait for the given time. Therefore, a way was implemented to make only a thread of a process sleep for a certain period of time. The function *pthread_cond_timedwait* waits for a condition to be signaled. If the condition is not signaled until a specified time is reached, the thread continues operation. Using this function with a mutex and a condition that never gets signaled will always make the thread wait for until the specified time is passed. To wait for a relative amount of time, one can wait until the current time (given by gettimeofday) plus the relative time desired. After the wait begins, the wait time is not affected by changes to the system clock. Although time is specified in seconds and nanoseconds, the system has approximately millisecond granularity. Due to scheduling and priorities, the amount of time it actually waits might be slightly more or less than the amount of time specified. That is why very fine wait periods (e.g., 0.1 ms) have a high margin of error.

Currently, the timer thread is designed to work with a granularity no less than 10 ms. Experiments still need to be performed to check the error margins or performance gains if this value is changed.

5. Experiments and Practical Applications

To test the protocol, several applications were implemented and some tests were performed. All tests were realized over an 802.11g wireless connection. While the signal strength was at 95% and speeds are at maximum, packet losses can still occur. This is on purpose to check that the protocol shows no issues. All computers have an Intel Core 2 Duo CPU, but at different frequencies, have at least 1 GB of free RAM, and no other programs running at the same time. The desire was to have more than one core per computer. CPU frequencies are not that important as the CPU load never increases over 1% while using the test applications.

5.1. File Download

The experiment performed was the measurement of the time it took to transfer a file from a single server. The goal was to see if the time increased in a more than linear fashion if the file size increased linearly. Normally the increment in time should be proportional with the increment in file size. If this does not happen, that means there is a problem with the protocol when large quantities of data are transferred. This can be a synchronization issue that makes the protocol block for a small period of time, a small buffer that gets filled up quickly or something else. Also, using this simple experiment, one can check for any performance benefits by tweaking different parameters of the transmission algorithm. The transfer time was measured by measuring the time since the transfer application started and until the time the application finished. Linux's time utility was used in this regard. The experiment was run with random data files of sizes 1, 2, 4, 8, 16 and 32 MB (see the results in Table 1). In each case, a client transferred the file to a server in pieces of a certain size and received the same data back with an ACK. The client checked the data to validate it and make sure it is the same one sent and proceeded to the next chunk of data.

Table 1
Transfer time for 1024 B of data payload (untweaked)

File size [MB]	Transfer time [s]
1	1.743
2	3.375
4	6.784
8	13.735
16	27.694
32	55.394

For the first run of the experiment, the sliding window had space for 32 elements, each packet carried a maximum data payload of 1024 B and the timer retransmitted a packet after 50 ms. The protocol achieved an average speed of 4.6 MB/s. The speed was consistent regardless of the size of the file with only a very small improvement as the file size went up. This indicates that the protocol runs well and the larger the file size the less noticeable the handshake or shutdown procedure gets.

Table 2
Transfer time for 8096 B of data payload (tweaked)

File size [MB]	Transfer time [s]
1	0.463
2	0.867
4	1.654
8	3.166
16	6.279
32	12.563

For the second run of the experiment different parameters have been tweaked. The sliding window now has a size of 128 elements, each packet carried a maximum data payload of 8096 B and the timer retransmitted a non-acknowledged packet after 10 ms. The results can be seen in Table 2. The protocol achieved an average speed of slightly over 19 MB/s. Again, as the file size increased, the transfer speed increased slightly. Even though the packet will be

fragmented by the IP layer into pieces of MTU size, a speed improvement is still observed. Testing these values in a less reliable medium is mandatory to check if packet fragmentation increases packet loss rate.
We can see there is a great improvement in the speed of the protocol after modifying these parameters. The 19 MB/s (for both sending and receiving) is getting closer to the advertised speed of the 802.11g wireless standard.

Table 3
Transfer time for 8096 B of data payload
and window size of 1

File size [MB]	Transfer time [s]
1	1.187
2	2.658
4	4.795
8	9.323
16	18.333
32	38.152

To verify the impact of the sliding window, a series of tests were performed where the sliding window accepted only one element. This basically means the protocol turned into a variant of the stop-and-go algorithm. All the other elements were left tweaked for improved speed. The results can be seen in Table 3. We can observe that speed has fallen back to around 6 MB/s of data transfer. The sliding window does have an impact on the performance of the protocol. This shows why, for the highest possible throughput, it is important that the transmitter is not forced to stop sending by the sliding window protocol earlier than one round-trip delay time (RTT). A different sliding window retransmission protocol might further improve performance. An example is selective repeat where the sender does not retransmit all packets starting with the lost one, but only retransmits the lost packet.

5.2. Segmented File Download

A practical implementation for the protocol is a segmented downloading program. A client connects to multiple servers at the same time and asks each of them for different segments of a file. This can also be done via existing protocols, but in these protocols a connection must be created for each server. With our protocol, a single connection can be created that touches every server and opens a stream with each of them. This makes implementation much simpler and error free. We only performed validation tests for this scenario.

6. Conclusions and Future Work

In this paper we presented a new approach for communication from one point to multiple endpoints. Our communication protocol is implemented on top of UDP and tries to provide the same facilities as TCP, but with extra functionality. The implementation of multiple endpoints is reliable, unlike multicast over UDP, and is not limited to only two endpoints that may have multiple IPs, like SCTP's implementation of multi-homing. Using the new protocol, applications can easily choose the type of connection they desire with the endpoints. Connection oriented multicast, anycast and load balancing are all fully integrated in our protocol. Experimental results showed that our protocol can provide good data transfer performance. However, as with any new protocol, further (extensive) testing is needed. We also intend to modify some of the components of the protocol, such as the connection open handshake and the limitation that a communication channel may use only a single UDP socket underneath (perhaps using multiple UDP sockets for sending and receiving data will be more efficient, due to having more buffer space allocated in the operating system kernel). Moreover, some of the parameters of the protocol (e.g., timeouts) still need to be tweaked for optimal performance.

Acknowledgements

The work presented in this paper was partially funded by the Romanian National Council for Scientific Research (CNCS)-UEFISCDI, under research grants ID_1679/2008 (contract no. 736/2009) from the PN II – IDEI program, and PD_240/2010 (AATOMMS – contract no. 33/28.07.2010), from the PN II – RU program, and by the Sectoral Operational Programme Human Resources Development 2007–2013 of the Romanian Ministry of Labour, Family and Social Protection through the financial agreement POSDRU/89/1.5/S/62557.

References

[1] W. R. Stevens, *TCP/IP Illustrated, Volume 1: The Protocols*. Addison Wesley, 1994.

[2] J. Postel, "RFC 768 – User Datagram Protocol", 1980 [Online]. Available: http://tools.ietf.org/html/rfc768

[3] R. Stewart *et al.*, "RFC 2960 – Stream Control Transmission Protocol", 2000 [Online]. Available: http://tools.ietf.org/html/rfc2960

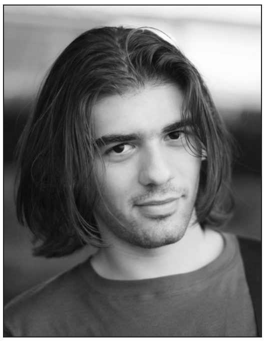
[4] M. I. Andreica, A. Drăguş, A.-D. Sâmbotin, and N. Ţăpuş, "Towards a (Multi-)User-Centric Stack of (Multi-)Point-to-(Multi-)Point Communication Services", in *Proc. 5th Worksh. Enhanced Web Serv. Technol. WEWST*, Ayia Napa, Cyprus, 2010, pp. 36–45.

[5] M. I. Andreica, A. Costan, and N. Ţăpuş, "Towards a Multi-Stream Low-Priority High Throughput (Multi)Point-to-(Multi)Point Data Transport Protocol", in *Proc. 6th IEEE Int. Conf. Intel. Comp. Commun. Proces. ICCP*, Cluj, Romania, 2010, pp. 427–434.

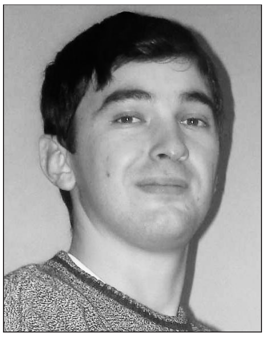
[6] T. Banka, P. Lee, A. P. Jayasumana, and V. Chandrasekar, "Application Aware Overlay One-to-Many Data Dissemination Protocol for High-Bandwidth Sensor Actuator Networks", in *Proc. 1stt Int. Conf. Commun. Sys. Software Middleware COMSWARE 2006*, New Delhi, India, 2006.

[7] J. Liebeherr, J. Wang, and G. Zhang, "Programming Overlay Networks with Overlay Sockets", *Lecture Notes in Computer Science*, vol. 2816, pp. 242–253, 2003.

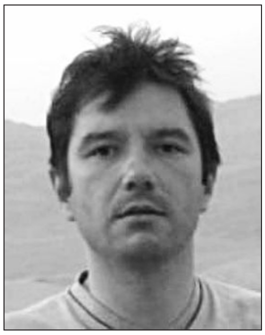
[8] M. Tuexen, and R. Stewart, "UDP Encapsulation of SCTP Packets", IETF Internet draft, 2011 [Online]. Available: http://tools.ietf.org/html/draft-ietf-tsvwg-sctp-udp-encaps-01



Valentin Stanciu is a graduate of Politehnica University of Bucharest and a teaching assistant at the same university. He obtained multiple prizes in different computer science competitions and participated at several internships with some of the biggest software companies (e.g., Adobe, Microsoft, Google). He is also an active board member of the Infoarena Association helping young students learn programming and train for competitions.
E-mail: valentin.stanciu@gmail.com
Politehnica University of Bucharest
Splaiul Independenţei 313
Bucharest, Romania



Mugurel Ionuţ Andreica, PhD, is an assistant professor in the Computer Science Department of the Politehnica University of Bucharest. His research topics include large scale distributed systems, P2P systems, distributed services, communication optimization and advanced, parallel and distributed data structures and algorithms. He was awarded the "Magna cum laude" distinction for his Ph.D. thesis and was granted an IBM Ph.D. Fellowship and an Oracle Research Scholarship for his Ph.D. research. He was the director of 2 national research projects and participated in 9 others (6 national and 3 international). He is (co-)author of 90 publications (books and papers in international journals and conferences). He obtained multiple prizes in algorithmic competitions (e.g., silver medal in the international olympiad in informatics and bronze medal in the ACM ICPC World Finals) and participated at several research internships (e.g., at Google Switzerland Gmbh and TU Delft) .
E-mail: mugurel.andreica@cs.pub.ro
Politehnica University of Bucharest
Splaiul Independenţei 313
Bucharest, Romania



Vlad Olaru holds a BS degree from the Politehnica University of Bucharest, Romania, an M.Sc. degree from Rutgers, The State University of New Jersey, USA and a PhD from the Technical University of Karlsruhe, Germany. His research interests focus on operating systems, distributed and parallel computing and real-time embedded systems. His Ph.D. work concentrated on developing OS kernel services for clusters of commodity-of-the-shelf PCs, namely cooperative caching for cluster filesystems and TCP connection endpoint migration for cluster-based servers. He was principal investigator within several EU-funded as well as national projects targeting the development of real-time OS software to control the next generation 3D intelligent sensors (with emphasis on power management and distributed control in ad-hoc wireless sensor networks), real-time Java for multi-core architectures, servers based on clusters of multi-core architectures, high-performance computing for computer vision programs.
E-mail: vlad.olaru@gmail.com
Politehnica University of Bucharest
Splaiul Independenţei 313
Bucharest, Romania

Nicolae Ţăpuş is a Professor of the Computer Science and Engineering Department, chair of this Department (1990–2008) and Vice-Rector of Politehnica University of Bucharest (UPB) since 2006. His main fields of expertise are Distributed Systems, Local Area Networks, and Computer Architecture. He is a Senior member of IEEE, ACM and Computer Society, chair of Romania Computer Society Chapter. He is director of ACM International Collegiate Programming Contest for Southeastern Europe since 1994. He received the award of the Romanian Academy, the award of Education Ministry and Innovation Award for High-Performance Applications from Corporation for Education Network Initiatives in California, CENIC. He is member of Romanian Technical Science Academy and Vice President of Information Technology and Communication Academy Section. He published more than 140 articles and papers at international conferences, 7 books and 12 university text books.
E-mail: nicolae.tapus@cs.pub.ro
Politehnica University of Bucharest
Splaiul Independenţei 313
Bucharest, Romania

Paper

Towards an Agent-Based Augmented Cloud

Roman Dębski, Aleksander Byrski, and Marek Kisiel-Dorohinicki

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

Abstract—In the paper an agent-based framework deployed in hybrid cluster and volunteer computing environment is presented. It utilizes two concepts proposed by the authors: *Augmented Cloud* and *Agent Platform as a Service (AgPaaS)*. Both concepts are discussed in the context of *Cloud Computing* as defined by NIST. The key idea of the presented solution is to span the cloud (i.e., computing infrastructure) beyond the data center borders by utilizing web browsers as computational workers. The feasibility of the approach was demonstrated by two prototypes: the first one was based on Java Applets and Adobe Flash, whereas the second one on Microsoft Silverlight. The prototypes were next used to perform simple experiments, mainly related to scalability issues. Selected results from the experiments are discussed in the final part of the paper.

Keywords—Agent Platform as a Service, Augmented Cloud, Cloud Computing, Multi-Agent Systems.

1. Introduction

Many large-scale[1] computational problems can be effectively solved in distributed environments which are based on the agent paradigm. Some examples are: marketplace simulations, modeling and control of autonomous robots and many optimization problems (complex multi-criteria, multi-modal and/or the ones in which the evaluation of a single solution is simulation based). In many of these cases *scalability* is the central issue. Both from the point of view of software (algorithms, logical architecture) and hardware (physical architecture).

Very often the hardware infrastructure scalability is the real issue. In the case of possible non-deterministic changes in the computation power demands[2], caused, e.g., by certain parameters of the computation model (cf. experiments with computational multi-agent systems presented in [1]), dynamic adaptation of the computation load is required (e.g., load balancing with diffusion-based scheduling as described in [2]). These problems are usually addressed by the choice of some cloud-oriented solutions [3]. Yet the problem of limited resources and the costs of maintaining often wasted computing power still remains open.

In the paper a new, cost effective, approach to building a highly scalable execution environment, particularly dedicated for computational systems which may be realized in agent-based paradigm, is presented. It utilizes two concepts proposed by the authors in [4]: *Augmented Cloud* and *Agent Platform as a Service (AgPaaS)*. Both concepts are discussed in the context of *Cloud Computing* as defined by NIST [5]. The key idea of the presented solution is to span the cloud (i.e., computing infrastructure) beyond the data center borders by utilizing web browsers as computational workers.

In the first section *AgE* – as an example of an agent-based computation platform (framework) is described. Next, its web browser-based implementation is presented. In the subsequent part, the concepts of *Augmented Cloud* and *Agent Platform as a Service* are defined. Since the contribution is a direct follow-up of [4], experimental results extending the ones presented there, constitute the final part of the paper.

[1]The term “large-scale problem” is used here in a broad sense, referring to all problems considered difficult for all known solution methods.

[2]When the demand for computation power is deterministic and can be reasonably estimated, a straightforward solution would be to use a computer cluster with well-defined, still limited resources.

2. AgE – Agent-Based Computation Platform

AgE platform[3] is designed to support building a wide range of agent-based optimization and simulation systems utilizing various meta-heuristics such as evolutionary algorithms. A *computation task* to be executed on the AgE platform is defined by providing a *computation description file*, which includes (among other things): the computation decomposition details (i.e., types of agents, their structure, types of operations that specify algorithms), problem-dependent parameters and the problem stop condition. Based on the description file, on the platform start-up, the computation context is built from the necessary components. Next, all the required agents and their environments are created, configured and distributed among nodes. Finally, the computation can be started and is performed until it reaches the stop condition. During the execution time some of the computational agents are attached to the platform monitors. The monitors are responsible for collecting problem-dependent data, used to visualize the current state and the final results of the computation.

The computation task is decomposed and the sub-tasks are assigned to the computational agents. The agents are structured into a tree (as shown in Fig. 1), according to the algorithm decomposition (it is worth to notice that each aggregate is also an agent). It is assumed, that all the sub-

[3]AgE is developed at AGH University of Science and Technology.

tasks assigned to the agents at the same level are executed in parallel. To increase performance, the top level agents (called *workplaces*) along with all their children can be distributed amongst many nodes.

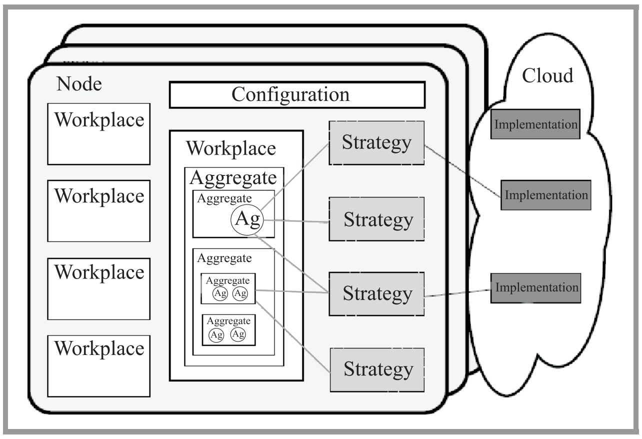


Fig. 1. Agent tree structure.

Agents, however, are not atomic assembly units, but they can be further decomposed into functional units according to *Strategy* design pattern [6]. Strategies represent problem-dependent algorithm operators and may be exchanged without intruding the agents' implementation. Their instances may be shared between agents as they provide various services to agents or others strategies.
Any part of a single agent's task can be delegated according to *Strategy* design pattern (see Fig. 1). Strategies can represent problem-dependent algorithm operators (mutation, evaluation of fitness, etc.) and may be exchanged without intruding agents' implementation. The delegated operations can be executed by external resources – this approach can be considered as a recommended way of the AgE platform expansion, especially in case of operations with a low communication-to-computation ratio. Fitness evaluation of a single solution can be given as an example of such delegated task (this approach was used in both prototypes discussed in the final section). In this case the computation can be based on the master-slave model [7], and the slaves (as fitness evaluators) can be: nodes connected in computation clusters dedicated to particular task, volunteers [8] (also web browser based [9]), or sideband computing applications [10].

3. Extending the AgE: A Web Browsers Based Approach

The main goal of the presented solution was to provide high scalability while remaining cost-effective. It has been achieved by utilizing web browsers as computational workers. The architecture of the platform is shown in Fig. 2. It is comprised of three layers:

- *The agent-based environment layer* provides a hierarchy of agents responsible for the realization of a given meta-heuristic. During their work, they generate tasks, which are delegated, using *Proxy* design pattern, to a connector that than passes it further to the application servers layer. Tasks are realized as asynchronous operations, so that agents can continue their work without waiting for results, unless they are required for further processing. Communication between the environment and the application servers layer is realized by sending requests, which identify tasks to be performed and input data.

- *The application servers layer* dispatches ordered tasks amongst available computational workers and passes back the results to the ordering agents. The number of workers depends on the number of the users whose web browsers cooperate with the platform (i.e., users that visit the services associated with the platform). It makes the workers' environment very dynamic, with the fault tolerance as the central issue to be addressed[4].

- *The web browser layer* utilizes applets[5] as computational workers which are responsible for executing the ordered tasks and returning the results to

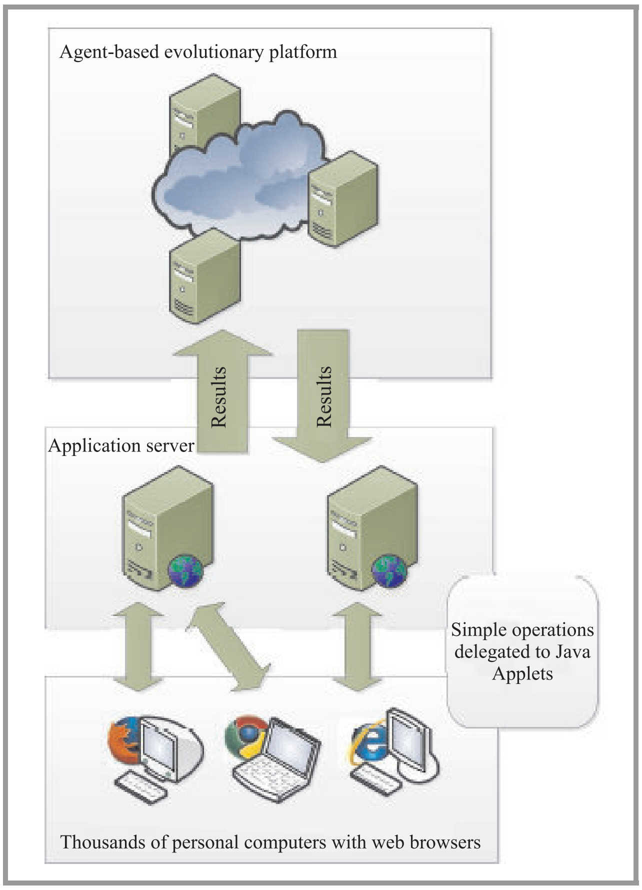


Fig. 2. Architecture overview.

[4]Consider users disconnecting the system in random moments.
[5]Programs executed in the context of a web browser, not necessary Java applets.

the application server. The applets are downloaded by browsers when a user visit any site connected to the system.

The approach can be applied both to building new computing environments and to extending the existing ones. However, one has to remember that the solution is suitable only for coarse grained problems (i.e., for which computation-to-communication ratio is high). Otherwise, the gained computing resources could be completely reduced by the communication overhead.

One can generalize the approach described here – it leads to the concept of *Augmented Cloud* [4], which is discussed in the next section.

4. Agent Platform in Augmented Cloud

An agent-based platform [11] dedicated for large-scale computations has to be designed for high scalability and deployed on a highly scalable execution environment like supercomputer, cluster, grid or cloud. Of the four, the last one is becoming more and more popular (also in High Performance Computing, e.g., [12], [13]) mainly because it is (at least can be) very cost effective [14], [15].

The approach discussed below is a hybrid of a classical cluster solution (as a backbone; can be considered as a classic cloud which spans the computers in a single data center), augmented with a web browsers based environment of volunteers, acting together as an *Augmented Cloud* [4].

4.1. Cloud Computing

A dedicated runtime environment (e.g., supercomputer, cluster) has many advantages. Yet, its scalability is restricted by the available computing resources (i.e., number of processors in a supercomputer or number of nodes in the cluster). That is why nowadays we observe a migration of services, software or even a whole computing infrastructure into “clouds” (like Windows Azure[6], Amazon EC2[7], or Google App Engine).

At this point it is worth to define *Cloud Computing* as the term has many definitions (e.g., [3] , [5], [15]) and there seems to be no consensus on what (precisely) a *Cloud* and *Cloud Computing* are [3]. According to NIST [5] *Cloud Computing* is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. The essential characteristics are: on-demand self-service, broad network access, resource pooling, rapid elasticity and measured service.

[6]http://www.microsoft.com/windowsazure/

[7]http://aws.amazon.com/ec2/

The service models defined by NIST [5] are as follows:

- Software as a Service (SaaS) – the capability provided to the consumer is to use the provider’s applications running on a cloud infrastructure. The applications are accessible from various client devices through a thin client interface such as a web browser (e.g., web-based e-mail). The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings.

- Platform as a Service (PaaS) – the capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly application hosting environment configurations.

- Infrastructure as a Service (IaaS) – the capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, deployed applications, and possibly limited control of select networking components (e.g., host firewalls).

IBM in [16] introduces also the fourth service model – BPaaS:

- Business Process as a Service (BPaaS) – any business process (horizontal or vertical) delivered through the cloud service model (Multi-tenant, self-service provisioning, elastic scaling and usage metering or pricing) via the Internet with access via web-centric interfaces and exploiting web-oriented cloud architecture. The BPaaS provider is responsible for the related business function(s).

4.2. Augmented Cloud – Agent Platform as a Service

Combining the concepts of *the cloud* (as a way of providing the illusion of infinite computing resources available on demand) and the *web browser-based volunteer computing* [9] (as a way of collecting computational resources) leads to a practically zero-cost and highly scalable solution, which may be called an *Augmented Cloud* [4]. The scalability of the classical cloud is limited by the number of nodes in the data center on which the cloud is based/deployed; the proposed solution *augments* the classical cloud by a significant increase of its scalability.

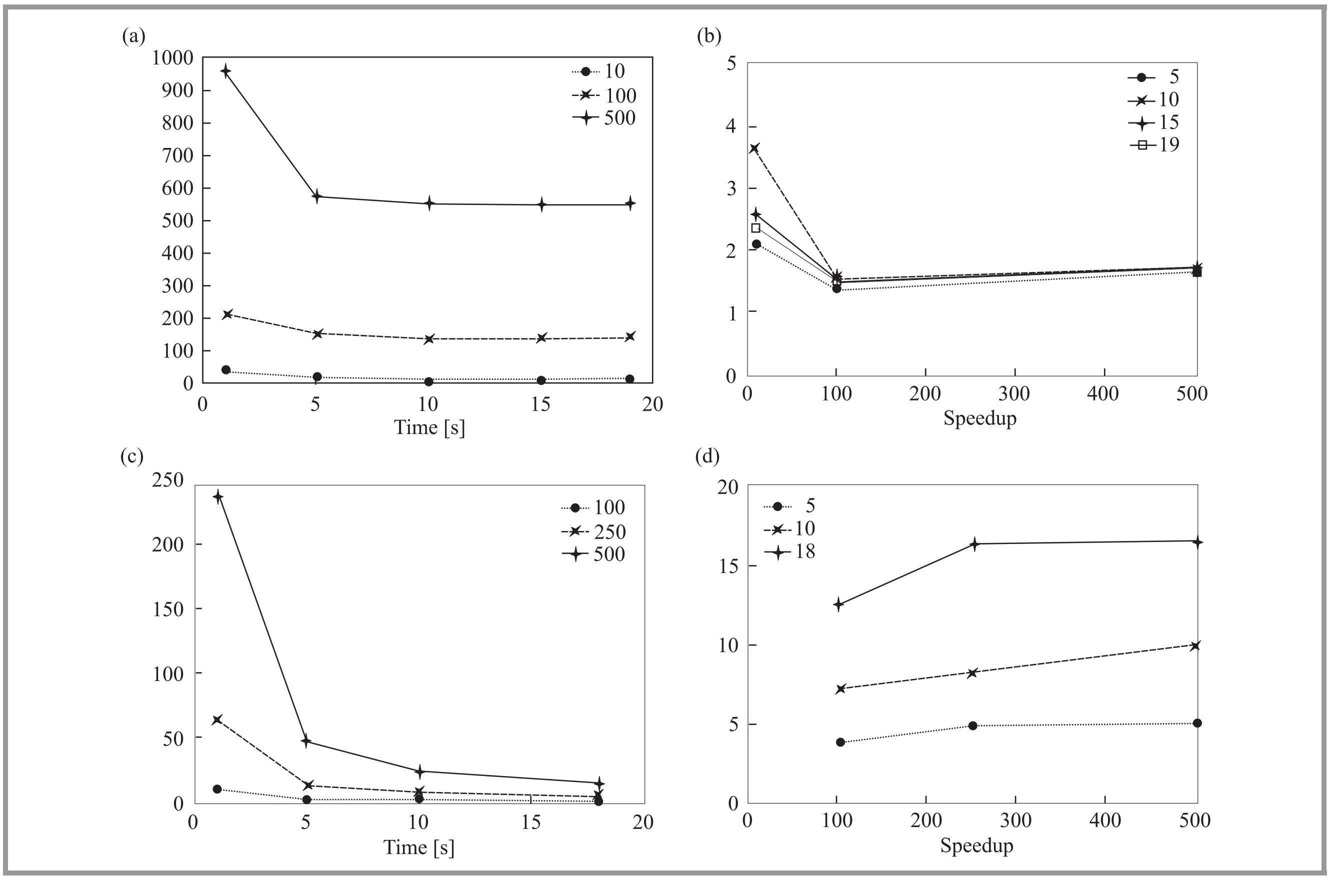


Fig. 3. Computation time [s] and speedup dependent on population sizes for different number of clients: (a), (b) – Java Applets; (c), (d) – Adobe Flash.

From the cloud computing service models [5] point of view the *Augmented Cloud* can be seen as a kind of Infrastructure as a Service (IaaS) in which (at least some of) the computational resources are web-browsers based. So it can be just a base for the target platform mostly because this level of abstraction is too low when taking into account both the development and deployment of (multi)agent-based systems.

What is needed is a set of services/libraries forming a highly scalable(software) platform dedicated for the target domain – agent-based systems. It can be considered as the second layer of the Cloud Computing Reference Architecture [16] which corresponds to the NIST Platform as a Service (PaaS) [5]. In this context it can be named *Agent Platform as a Service (AgPaaS)* to emphasize the agent-orientation of the platform [4].

5. Experimental results

In order to evaluate the concept of *Augmented Cloud*, computing time and speedup has been tested for a classical master-slave model of a parallel evolutionary algorithm, in which the computation of fitness values was delegated to browser-based slaves (as described in the previous sections). The experiments were based on two prototypes: in the first one the web browser layer was based on Java Applets and Adobe Flash, and in the second one on Microsoft Silverlight. The results are described in the next two subsections.

5.1. Java Applet and Adobe Flash

In this case the computing environment had a two-tier/layer architecture and consisted of a backbone server running a master process and two types of clients (slaves): Java Applet based (communicating with the server using Java RMI) and the Adobe Flash based. The algorithm was solving a typical benchmark optimization problem (Rastrigin function) for different population sizes and different number of clients.

According to expectations, increasing the number of slaves decreases the overall computation time (see Fig. 3). This effect is strongly dependent on the technology: the Java Applet slaves (see Fig. 3(a)) are (much) less effective than the Adobe Flash ones (see Fig. 3(c)). It is mainly because of the Java RMI communication overhead.

Figures 3(b) and 3(d) show the same data but from the computation speedup perspective. One can easily notice that in case of Java Applets there is practically no speedup. It is a good visualization of the importance of computation-to-communication ratio.

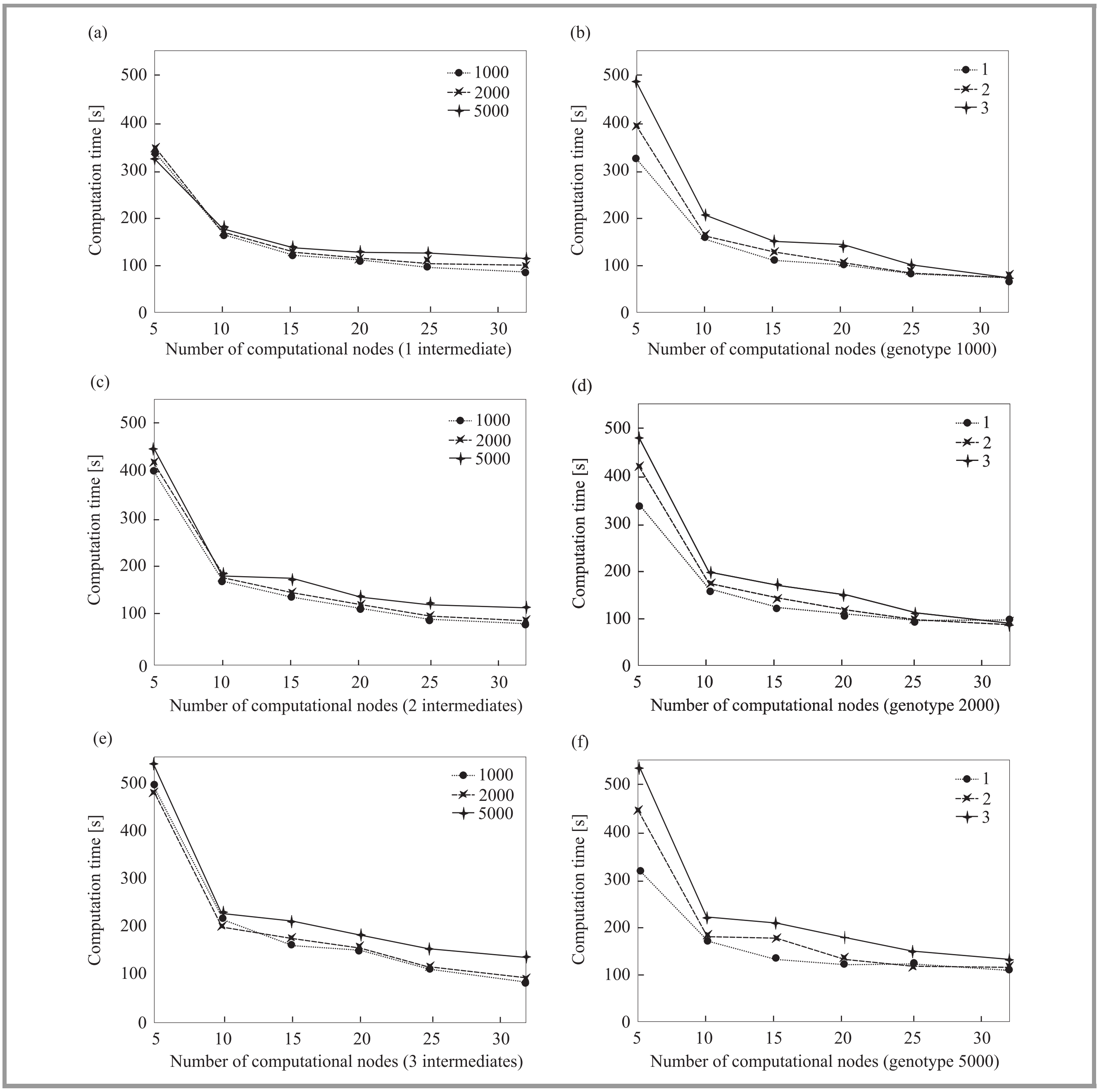


Fig. 4. In the left column: computation time as a function of the number of computational nodes, presented for different numbers of intermediate nodes and for different genotype lengths, in the right column: computation time as a function of the number of computational nodes, presented for different lengths of genotype and for different number of intermediate nodes.

5.2. Microsoft Silverlight

In this experiment the computing environment had a hierarchical[8], three-tier/layer structure.

- The backbone server (the master) – it was executing a simple genetic algorithm, delegating the calculation of each individual's fitness to web browser-based computational workers (slaves) via the application controllers layer.
- Application controllers (intermediate layer) – each controller is responsible for the pool of computational workers allocated to it; this layer can be utilized in different ways, e.g., to improve the system availability or/and reliability, to support load balancing, to control the value of computation-to-communication ratio (note: in the prototype none of the above was implemented).

[8]The environment can be visualized as a tree: level 1 – the backbone server (as the root), level 3 – the computational workers (as leaves), level 2 – controllers of sub-trees.

- Microsoft Silverlight based computational workers.

  Note: it can be noticed that the computing environment is similar to the one presented in Fig. 2, but with one exception: the AgE platform has been reduced to a single master process, which executed a simple genetic algorithm.

Figures 4(a), 4(c) and 4(e) show the computation time as a function of the number of computational nodes, for different numbers of intermediate servers (application controllers). One can notice that in the context of this computation, the introduction of intermediate nodes caused a significant overhead (in the whole range of analyzed computational node numbers), and in consequence, worsened the computation time. But as the number of the computational nodes was increasing this overhead was becoming smaller and smaller (close to zero for the last point of the curve). So as in case of most distributed systems, the performance of this one is strongly dependent on its configuration.

Figures 4(b), 4(d), and 4(f) present the same data from a different perspective.

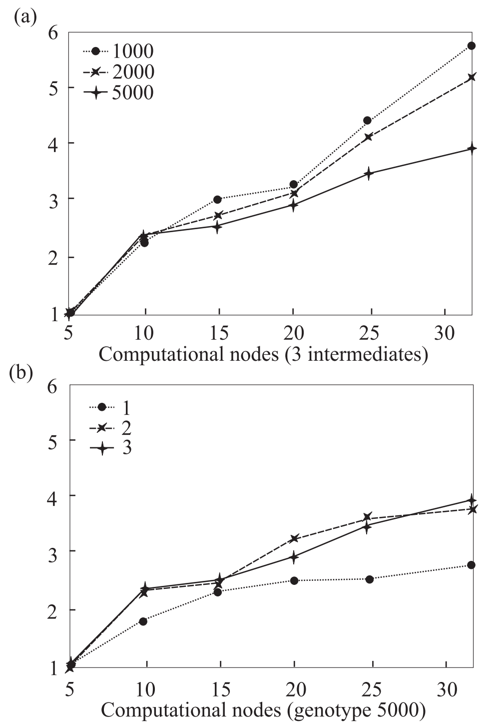


Fig. 5. Speedup as a function of the number of computational nodes: (a) in the configuration with 3 intermediate nodes, presented for different genotype lengths; (b) with the genotype length set to 5000, presented for different number of intermediate nodes.

Figures 5(a) and 5(b) present the system performance from the speedup point of view. An interesting effect is shown in Fig. 5(a): the smaller the genotype length, the bigger the speedup (and this effect is stronger as the number of computational nodes is increasing). This is again caused by the communication overhead: the bigger the genotype length, the more time is needed by the intermediate node to transfer its data. When a controller (intermediate node) handles many computational nodes its processing is I/O bound and, in consequence, it can become a bottle-neck of the whole system.

6. Conclusion

In the course of the contribution, after describing the AgE (as an example of an agent-based computation platform), and a possible way of its extension, the concepts of Augmented Cloud and the Agent Platform as a Service were introduced as way to address the scalability issues, which are present in many large-scale computations performed in the agent-based distributed environments.

In the final part of the paper, selected results obtained for a proof of concept kind of prototypes were shown. The computing environment of the first prototype consisted of a backbone server (master) and the two types of clients (slaves): Java Applet and Adobe Flash ones. The second prototype had a three-tier architecture with the computational workers (clients/slaves) based on Microsoft Silverlight.

The obtained results encourage further research and broader implementation of the Augmented Cloud concept, at the same time bringing more awareness about affecting the potential results by choosing appropriate technology for the implementation both of the server and web-browser clients, as well as for the communication.

In the near future the authors plan to conduct broader experiments with volunteer nodes based on different web technologies (e.g., JavaScript/WebWorkers). A long-term goal is to further evaluate the concept of Agent Platform as a Service (AgPaaS).

Acknowledgements

The work presented in this paper was partially supported by the Polish National Center of Research and Development grant No. 0108/R/T00/2010/11.

References

[1] A. Byrski, M. Kisiel-Dorohinicki, and M. Carvalho, “A crisis management approach to mission survivability in computational multi-agent systems”, *Comp. Science*, vol. 11, pp. 99–113, 2010.

[2] M. Grochowski, R. Schaefer, and P. Uhruski, “Diffusion based scheduling in the agent-oriented computing system”, in *Proc. Int. Conf. PPAM 2003*. Springer, 2004.

[3] S. Lu, I. Foster, and I. Raicu, “Cloud computing and grid computing 360-degree compared”, in *Grid Comput. Environments Worsh. GCE’08*, Austin, Texas, USA, 2008.

[4] A. Byrski, R. Debski, and M. Kisiel-Dorohinicki, “Agent-based computing in augmented cloud environment”, *Int. J. Comp. Sys. Sci. Eng.*, 2012 (accepted for printing).

[5] P. Mell and T. Grance, “The nist definition of cloud computing (draft)”, Tech. rep., National Institute of Standards and Technology, 2011.

[6] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison Wesley, 1995.

[7] E. Cantú-Paz, "A summary of research on parallel genetic algorithms", IlliGAL Rep. No. 95007, University of Illinois, 1995.

[8] D. P. Anderson, E. Korpela, and R. Walton, "High-performance task distribution for volunteer computing", in *Proc. 1st Int. Conf. e-Sci. Grid Comput.*, Melbourne, Australia, 2005.

[9] L. F. G. Sarmenta and S. Hirano, "Bayanihan: Building and studying web-based volunteer computing systems using Java", *Future Gener. Comp. Systems*, vol. 15, pp. 675–686, 1999.

[10] Y. Xu, "Global sideband service distributed computing method", in *Proc. Int. Conf. Commun. Netw. Distrib. Sys. Model. Simul. CNDS'98*, San Diego, CA, USA, 1998.

[11] B. Czerwinski, R. Debski, and K. Pietak, "Distributed agent-based platform for large-scale evolutionary computations", in *Proc. 5th Int. Conf. Complex Intel. Softw. Intensive Sys.*, IEEE Press, 2011.

[12] K. R. Jackson *et al.*, "Performance analysis of high performance computing applications on the amazon web services cloud", in *Proc. 2nd IEEE Int. Conf. Cloud Comput. Technol. Sci. CloudCom'10*, Indianapolis, USA, 2010.

[13] J. Napper and P. Bientinesi, "Can cloud computing reach the top500", in *Proc. UCHPC-MAW '09*, ACM, 2009.

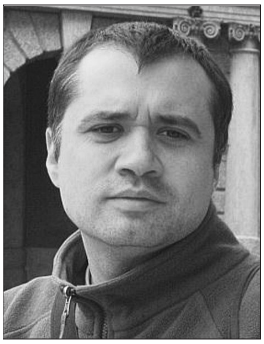
[14] J. J. Merelo *et al.*, "Browser-based distributed evolutionary computation: performance and scaling behavior", in *Proc. 9th Ann. Conf. Genetic Evol. Comput. GECCO '07*, New York, NY, USA, 2007, pp. 2851–2858.

[15] M. Armbrust *et al.*, "Above the clouds: A Berkeley view of cloud computing", Tech. rep., UC Berkeley, 2009.

[16] M. Behrendt *et al.*, "IBM cloud computing reference architecture 2.0", Tech. rep., The Open Group, 2011.

Roman Dębski holds the M.Sc. in Computer Science (AGH University of Science and Technology) and in Mechanical Engineering and the Ph.D. in Computational Mechanics (both obtained at Cracow University of Technology). He is an independent consultant with over 14 years of experience in IT. His current research focuses on the cloud computing.

E-mail: Roman.J.Debski@googlemail.com
Deptartment of Computer
AGH University of Science and Technology
Mickiewicza Av. 30
30-059 Kraków, Poland



Aleksander Byrski obtained his Ph.D. in 2007 at AGH University of Science and Technology in Kraków. He works as an assistant professor at the Department of Computer Science of AGH-UST. His research focuses on multi-agent systems, biologically-inspired computing and other soft computing methods.

E-mail: Olekb@agh.edu.pl
Deptartment of Computer Science
AGH University of Science and Technology
Mickiewicza Av. 30
30-059 Kraków, Poland

Marek Kisiel-Dorohinicki obtained his Ph.D. in 2001 at AGH University of Science and Technology in Kraków. He works as an assistant professor at the Department of Computer Science of AGH-UST. His research focuses on intelligent software systems, particularly utilizing agent technology and evolutionary algorithms, but also other soft computing techniques.

E-mail: Doroh@agh.edu.pl
Deptartment of Computer Science
AGH University of Science and Technology
Mickiewicza Av. 30
30-059 Kraków, Poland

Paper

Design and Implementation of an Embedded System for Ambulatory Cardiac Monitoring

El Mimouni El Hassan and Karim Mohammed

University Sidi Mohammed Ben Abdellah, Fès, Morocco

Abstract—Cardiac monitoring in the environment of the subject is one of the major fields of telemedicine. In this paper we present a prototype of embedded system for acquisition, storage, display on LCD or PC and transfer via GSM alarm warning in case of arrhythmias, which allows a great opportunity for rapid intervention of the physician. In terms of hardware, we have designed and implemented our system with a modular approach to facilitate development and debugging. Thus the system comprises three modules: analog module, digital module microcontroller-based for certain pretreatment, and a GSM communication module. Of course, there is appropriate software behind the material described. The system has the following features: low cost, ease to implement and versatility.

Keywords*—*arrhythmia, ECG, embedded system, GSM communication, heart rate, telemedicine.

1. Introduction

In recent years, technological development especially in microelectronics and software engineering have allowed the emergence of powerful processors (microprocessors, microcontrollers, digital signal processors, FPGA, etc.) for embedded systems, sophisticated personal computers (PC) and communications networks with varied standards (TCP/IP, GSM, Bluetooth, Wi-Fi, ZigBee, etc.). The hospital industry benefits from these assets and produces more reliable equipments, user friendly and accessible to a wide range of social strata, allowing, among other things, saving more lives and creating valuable data for research and diagnostic.

An interesting scenario is to monitor people with heart disease who may at any time be subject to a heart attack. In this context, the system we have designed and implemented is a simple prototype example of an embedded system for such scenario.

Without claiming to be exhaustive, we report that a number of studies have concerned the ECG signal (Electrocardiogram), based on standard databases [1], or by developing a standard ECG acquisition module or dedicated ASIC [2]–[4]. Other researches suggested software solutions for signal processing to reduce noise and classification of cardiovascular diseases [5], [6], remote monitoring via Bluetooth [7] or via an RF module integrated into a microcontroller [8].

This work concerns an embedded system based on standard and cheap components, built around the microcontroller PIC16F876 for the acquisition of the ECG, its storage, visualization and detection of arrhythmias with alert message transmission via GSM.

2. System Description

Figure 1 shows the block diagram of the system inspired, among others, by the application note from Analog Devices [9], common reference for many projects of this kind. The system presented is designed with a modular approach to facilitate the development and debugging. Thus, it includes three modules:

– acquisition module,

– microcontroller-based digital module,

– GSM communication module.

The ECG signal is the measure on the body surface (skin), of the electrical potential generated by the heart's electrical activity. Reading it allows an accurate evaluation of the performances of the heart.

As shown in Fig. 2, the ECG is characterized principally by 5 waves reflecting the different functions of the heart during a cardiac cycle; these waves are called by the successive letters of the alphabet P, Q, R, S and T [10].

It is then question to acquire the ECG signal typically characterized by a maximum amplitude of 1 mV and a bandwidth of 0.05 Hz to 100 Hz; also, a such signal is normally subject to several sources of noise: bad contacts and movement of the electrodes, contraction muscles, electromagnetic interference by inductive coupling, and 50 Hz or 60 Hz power line through capacitive coupling; this latter source is by far the most dominant.

Like any embedded system, our application has a hardware aspect and a software aspect. In turn the hardware consists of an analog part and a digital part.

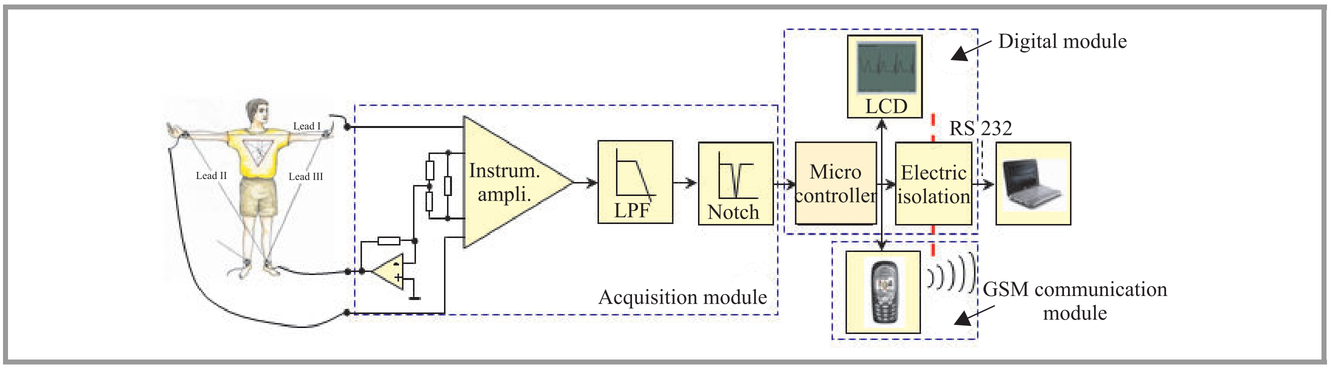


Fig. 1. Block diagram of the system.

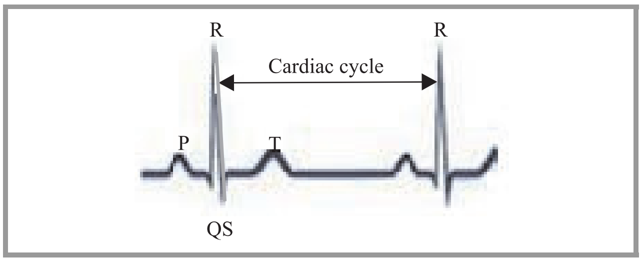


Fig. 2. The ECG waves.

2.1. Hardware Aspect

2.1.1. Analog Part

The analog module has two stages:

– an instrumentation amplifier,

– an analog filter.

As in most wearable ECG, our system uses lead II of the Einthoven's triangle, which is enough for a quick review. The two electrodes of this lead II, connected to the arms or on the chest, are applied to a differential instrumentation amplifier (IA) that we have achieved with cheap conventional operational amplifiers with acceptable performances, particularly a relative good Common Mode Rejection Ratio (CMRR), see Fig. 3.

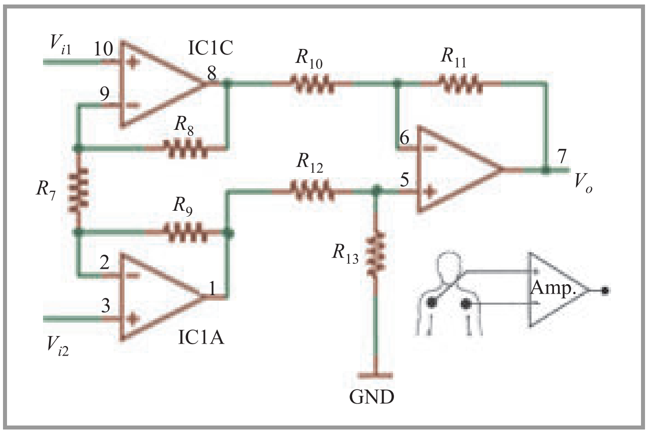


Fig. 3. Instrumentation amplifief with the LM324.

The amplification is expressed as:

$$V_0 = \left(1 + \frac{2R}{R_7}\right)(V_{i1} - V_{i2})$$

where R is the value of all the resistors except R_7.

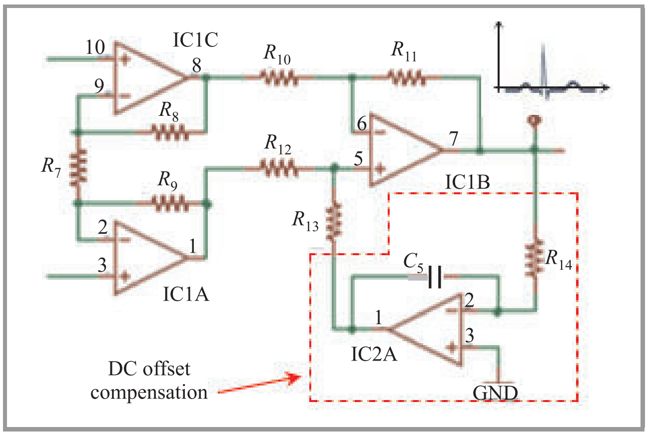


Fig. 4. DC offset compensation (baseline wander).

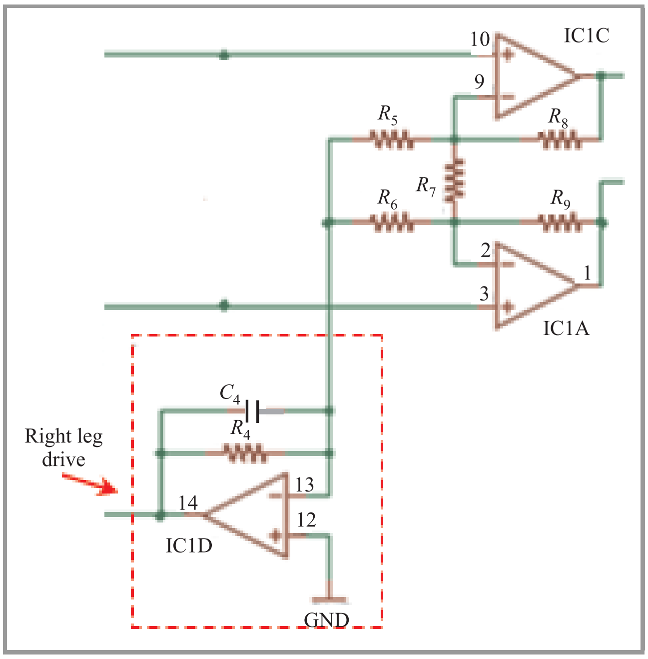


Fig. 5. Right Leg Drive.

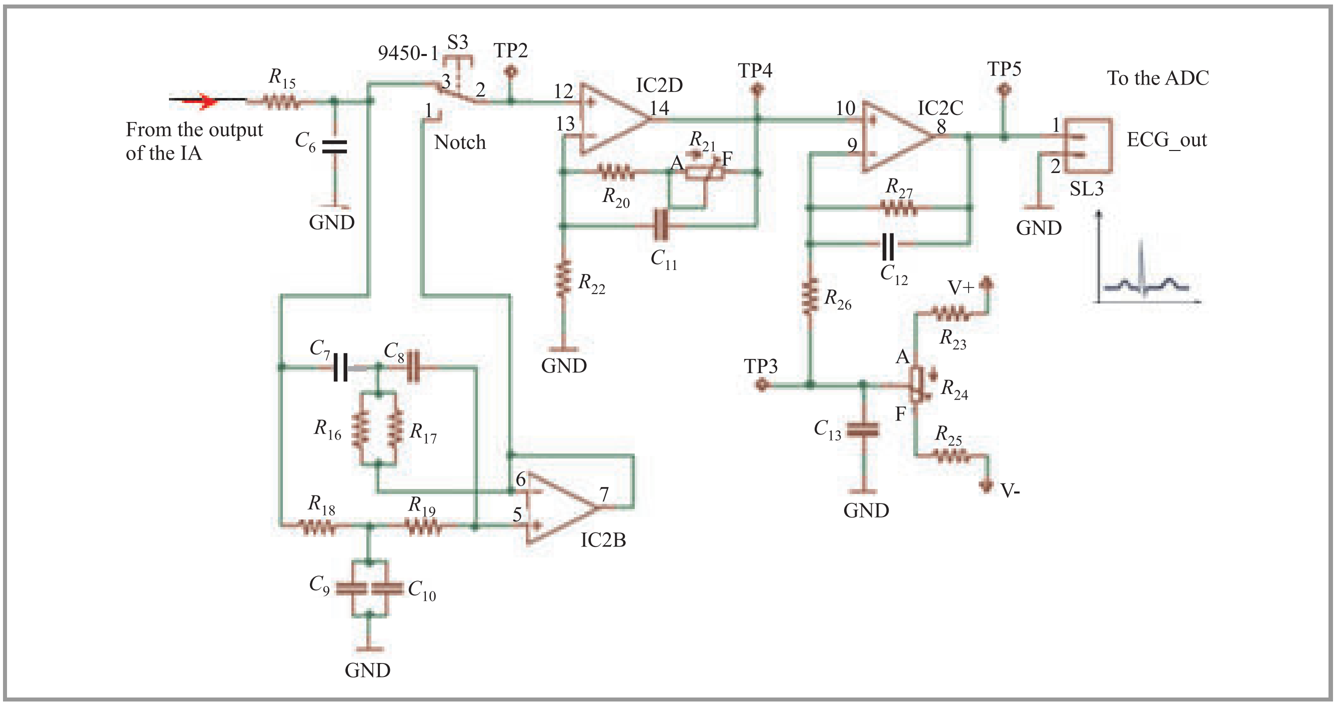


Fig. 6. Filters structure.

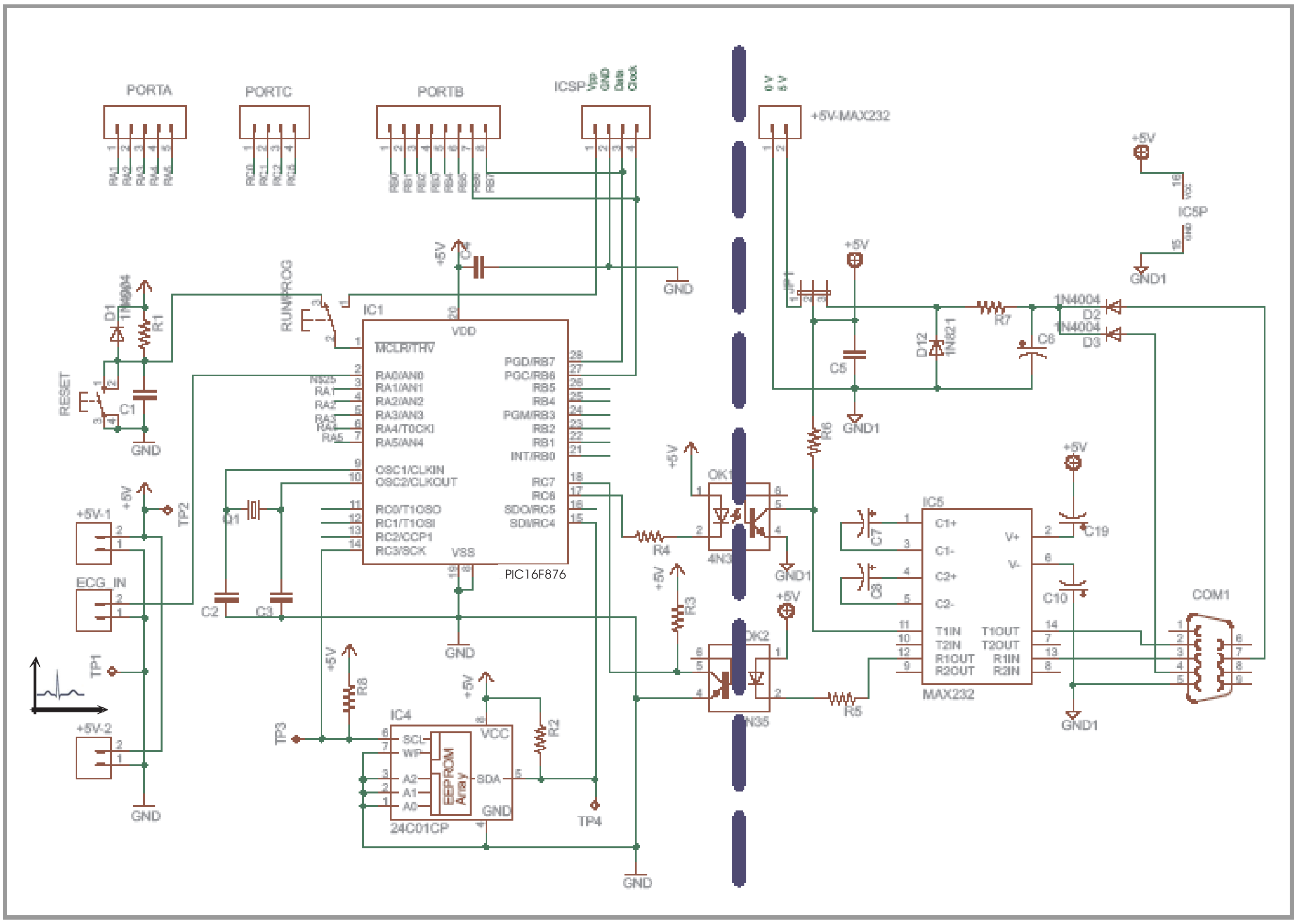


Fig. 7. Digital module.

To avoid the effect of baseline wander (isoelectric line drift), caused among others by electrodes movement, we add a control loop which detect the DC component due to this phenomenon by a low pass filter (RC circuit) and inject that component to the input, that is the reaction with IC2A in Fig. 4 [11]; thus, the final result is that the output signal of IA is adjusted automatically to a null DC component.
As in the modern electrocardiograph [12], the reference electrode is connected, with feedback, to the right leg via an amplifier to reduce common mode noise, a principle commonly known by the "Right leg drive". Figure 5 depicts the implementation of such a circuit.
The amplification stage is followed by an analog active filtering stage (Fig. 6). Thus, the different types of noise are first reduced by low pass filter (R_{15}, C_6) with band pass up 70 Hz and an optional notch filter for 50 Hz power line noise (IC2B). This latter operates only if the 50 Hz noise is at a high level; indeed, the 50 Hz frequency is natural component of the ECG's spectrum.
The system is battery powered, which contributes efficiently to improve the noise immunity, particularly against the 50 Hz power line noise. The last stage (IC2C) of this analog module is an anti-aliasing low pass active filter (Shannon sampling theorem), with adjustable DC offset, via R_{24}, allowing the output of this stage having an amplitude comprised between 0 and 5 V, which is required by the built-in analog to digital converter (ADC) of the microcontroller. The global amplification of the entire measuring chain is around 1000, as typically required.

2.1.2. Digital Part

The output signal of the analog module is then applied to the ADC of this digital module, and it is sampled at frequency of 200 Hz respecting the condition of Shannon. As shown in Fig. 7 this module includes primarily:

– a microchip PIC16F876 microcontroller, as hard core of this structure,

– an I2C EEPROM for storing samples of the digitized ECG,

– RS232 interface for communication with PC with galvanic isolation; the same interface is used for GSM communication,

– a graphic display LCD for displaying the signal and heart rate.

The choice of the PIC16F876 [13] is dictated by, among other things, its popularity, its price/quality ratio and the abundance of documentation and development tools. Thus, the analog signal ECG drives an analog input of the μC (RA0) configured in unipolar mode; the sampling frequency F_S is chosen equal to 200 Hz ($T_S = 5$ ms), principally because of the response time of EEPROM [14], which is around 3 ms. The EEPROM communicates with I2C protocol [15].

The sampled signal is saved in the I2C EEPROM, 24LC256, with a storage capacity of 256 kbits, that is 32 kbytes (IC4). Thus, with F_S of 200 Hz, we can record up to 2 mn 43 s for a further off-line treatment:

$$T_S = \frac{32 \times 1024}{200} = 163.84 \text{ s} \approx 2 \text{ min } 43 \text{ s}$$

The μC have built-in RS232 interface that permits serial communication with a PC or GSM, via his serial port. A classic MAX232 [16] achieves the RS232/ TTL conversion (physical layer protocol). With this point to point link, we transmit the sampled ECG to the PC for storage in hard disk and treatment in real-time or off-line (filtering, analysis, displaying, etc.).
The LCD displays the ECG signal and heart rate; we have used a popular and cheap graphic LCD. This is a mobile phone spare part available on the market, the Nokia 3310 which have acceptable features [17]:

– resolution of (84×48 pixels as shown in Fig. 8), it can display up 6 lines of 14 characters in the form of a matrix of (8×5) pixels with a good contrast,

– easy to drive as illustrated in Fig. 9,

– consumption of 110 μA under 3.3 V,

– controled by Philips PCD8544 [18].

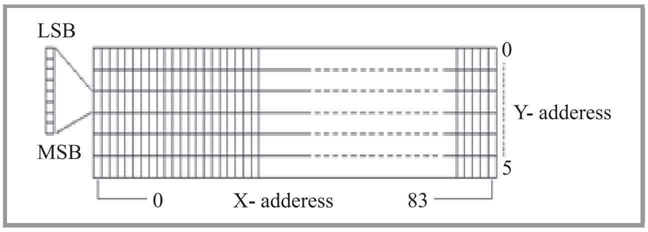


Fig. 8. LCD RAM addressing.

Personal safety is a central point for medical equipment; in fact, the subject is in direct contact with electrical voltages, and care should be taken to protect him by ensuring that the currents flowing by him respect the standards such as those of the AAMI (Association for the Advancement of Medical Instrumentation) [19]. In all cases, the currents must not exceed a value of 50 μA. In order to comply with this standard, among the techniques used for the safety of the subject, we have chosen (Fig. 10):

– using the battery power ($\pm$(3–9) V) for the entire analog and digital circuitry,

– galvanic isolation with infrared optocoupler, between the digital module and the PC, which is normally powered from the mains (220 V, 50Hz),

– clipping by diodes and current limiting in case of any electrical accident (direct contact of the line with the circuitry).

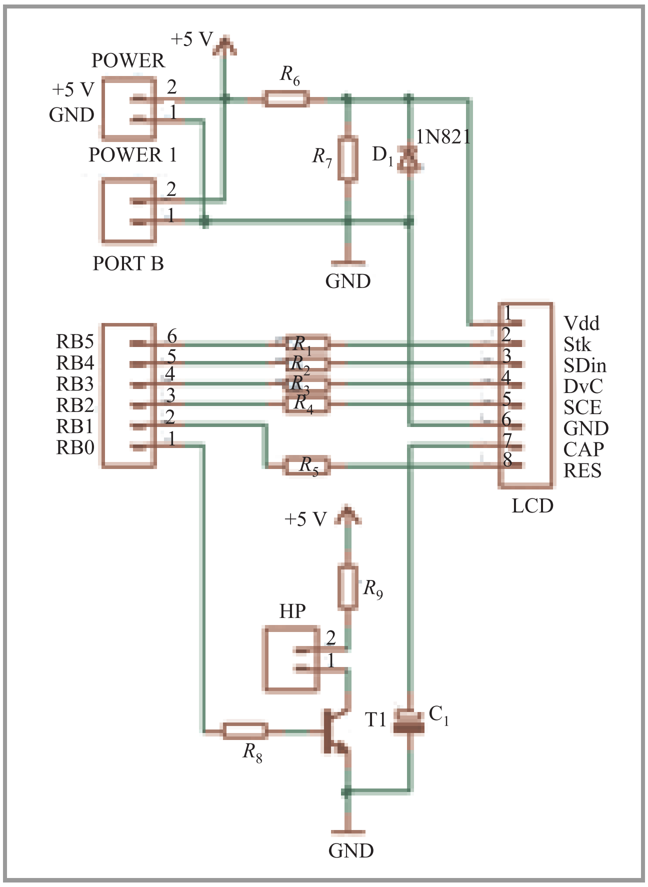


Fig. 9. ILCD control.

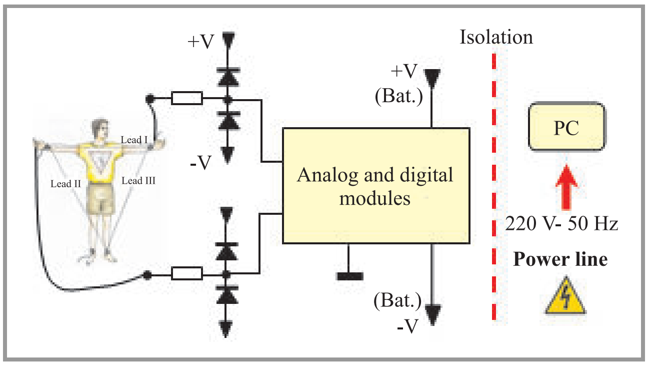


Fig. 10. Subject safety.

2.2. Software Aspect

This section treats the software aspect of our application; thus, we discuss the used software structures: the development tools, the various programs developed for microcontroller and PC.

2.2.1. μC Side

We opted for the following tools:

- The C compiler mikroC of MikroElektronika for all programs on the microcontroller, because such a compiler is rich in libraries for standard peripherals (keyboard, displays, EEPROM, etc.).
- The environment Microsoft Visual Basic (VB6) for the graphical user interface (GUI) on PC; in fact, this environment is one of the most used in the development of Windows applications.
- The MATLAB software for the calculation of digital filters.

It is then question to write a code that manages information related to all the devices that we have already briefly described in the hardware aspect of the system:

- acquisition of the ECG signal,
- storage sampled data in EEPROM,
- serial asynchronous communication,
- LCD display,
- audio signal,
- human/machine interface (buttons/LEDs).

```
const AddressBaseData=0;
unsigned int i;
char ADC_Data, FsCounter ;

//---------------Function Write_EE----------------------
void Write_EE(unsigned int Address, char DataEE)
{
 unsigned int AdressTemp;
 char AddressH, AddressL ;
 AdressTemp=Address;
 AddressL=(char)(AdressTemp & 0x00FF);
 AdressTemp=AdressTemp >> 8;
 AddressH=(char)(AdressTemp & 0x00FF);
 I2C_Start();                 // Signal I2C Start
 I2C_Wr(0xA0);
 I2C_Wr(AddressH);
 I2C_Wr(AddressL);            // Write byte in EEPROM
 I2C_Wr(DataEE);
 I2C_Stop();                  // Signal I2C Stop
 Delay_ms(3);                 // Wait writing in EEPROM
}
//---------------Function Interrupt----------------------
void interrupt()
{
 if (INTCON.T0IF==1)
 {
   INTCON.T0IF=0;
   FsCounter++;
   INTCON = 0xE0;
 }
}
//---------------Main function ---------------------
void main()
{
  i=0;
  FsCounter=0;
 USART_Init(19200);   // Initializing USART (19200 b/s, 1 bit de stop, no parity)
  I2C_Init(100000);
  ADCON1=0;           // Configuring ADC
  INTCON=0xE0;
  OPTION_REG=0x88;
  TRISA=0xFF;          // PORTA as input
  do
  {
   if (FsCounter==20)              // 20*256 µs = 5120 µs = 5.1 ms
     {
      FsCounter=0;
      ADC_Data = ADC_Read(0) >> 2;      // Reading ADC with 8 bits (RA0)
      USART_Write(ADC_Data);            // sending sampled data to au PC
      Write_EE(AddressBaseData+i, ADC_Data);// rage in EEPROM
      i++;
      if (i==1024)      // 1024 samples
        i=0;
     }
  }while(1);
}
```

Fig. 11. Acquisition, storage and sending to PC.

After many attempts to process all these tasks in real time, we finally realized that, having regard to the modest structure of the chosen microcontroller: small capacity of RAM (368 bytes), relative low speed (4 MHz), etc., the program must be implemented under a real-time operating system (RTOS).

So, in this work, we have developed all the tasks separately with success; we describe briefly the code of some of these tasks.

- **Acquisition of ECG, storage and sending to the PC**. In this program, we use the predefined basic functions of mikroC compiler:
  - *interrupt ()* to intercept the timer 0 interrupt setting the sampling period at 200 Hz,
  - *ADC_Read ()* to read the ADC,
  - *USART_Write ()* to send the sample acquired to the PC,
  - *I2C_Start (), I2C_Wr () and I2C_Stop ()* we have encapsulated in *Write_EE function ()* to write a byte to an address in the EEPROM.

  Figure 11 provides the corresponding complete C code.

- **Measurement and display of heart rate**. For this case, although the most common choice is the well known and more efficient algorithm of Pan/Tompkins [20], we opted for a simpler algorithm, an off-line simple thresholding algorithm after high pass filtering discriminating the R wave, based on [21] which is an average calculation of 3 RR cycles:
  - read the ECG signal from the EEPROM,
  - detect the maximum value of the ECG signal, that is detecting the R wave of the QRS complex (Max),
  - compare each value of the threshold ($0.7 \times$ Max) to avoid the influence of the P wave, which typically has a value ($0.4 \times$ Max) and produce a logic signal, the result of the comparison and store in EEPROM,
  - detect rising edge of the logic signal and count up 4 occurrences of this event, which corresponds to three cycles of the RR interval of the ECG,
  - identify the indices 1 (Index1) and 4 (Index4),
  - deduce the cardiac cycle, knowing that $T_S = 0.005$ s:

$$\text{heart rate} = \frac{60 \times 3}{(\text{Index4} - \text{Index1}) \times 0.005}$$

  Figure 12 provides the corresponding complete C code.

```
void main()
{
unsigned int i, Index1, Index4 ;
char ADC_Data, DataTmp, DataTmp1, DataTmp2,
HeartRate, ValMax , NbrMax, HeartRateTxt[4] ;
  I2C_Init(100000);
  TRISA=0xFF;
  TRISB=0b11000000;
  initlcd();
  clearscr ();
  gotoxy(3,0);
  putstrCte("ECG ISAI 2010");
  ValMax = 0;
  for (i=0;i<1024;i++)
   {
   DataTmp=Read_EE(AddressBaseData+i);
   if (DataTmp > ValMax)
     ValMax = DataTmp;
   }

  for (i=0;i<1024;i++)
   {
   DataTmp=Read_EE(AddressBaseData+i);
   if ( DataTmp >= 0.8 * ValMax)
     Write_EE(AddressDetectRR+i, ValMax);
   else
     Write_EE(AddressDetectRR+i, 0);
   }
  i = 0;
  NbrMax = 0;
  while(NbrMax <= 4)
  {
   DataTmp1=Read_EE(AddressDetectRR+i);
   DataTmp2=Read_EE(AddressDetectRR+i+1);
   if ((DataTmp2 - DataTmp1) > 0)
   {
     NbrMax++;
     switch (NbrMax)
     {
      case 1: Index1 = i ;break;
      case 4: Index4 = i ;break;
      default:  ;
     }
   }
   i++;
  }
  HeartRate = (char)(60*3/((Index4 - Index1) * 0.005));
  ByteToStr(HeartRate, HeartRateTxt);
  gotoxy(4,3);
  putstrCte("HR = ");
  putstr(HeartRateTxt);
  putstrCte(" bmp");
}
```

Fig. 12. Measure and display of heart rate.

2.2.2. PC Side

The program of this human/machine interface (HMI) on the PC we have developed in VB6 environment allows to:

- Communicate with the μC through the serial port.
- Display the original ECG signal (noisy signal) and filtered signal.
- Calculate and display the heart rate.
- Display signals related to the measurement of heart rate (threshold, QRS binary signal, etc.).
- Record the signal on hard drive for archiving and the possibility of further processing.

- Communicate by e-mail with the GSM protocol "AT Commands" for arrhythmia warning.

```
» [b,a]=butter(1,[0.05/100,70/100],'bandpass')
b =
    0.6620         0   -0.6620

a =
    1.0000   -0.6739   -0.3241
```

Fig. 13. IIR design with *butter*.

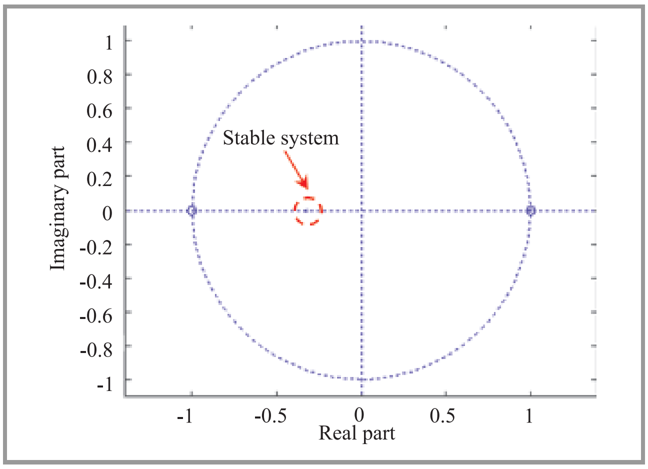


Fig. 14. IIR stability with *zplane*.

In what follows, omitting the code purely related to the HMI, we discuss only the application's core, i.e., digital filtering [22]–[24], measurement of cardiac rhythm and GSM communication.
So, we used the MATLAB functions *butter* and *zplane* (Figs. 13 and 14), respectively for coefficients determining of the transfer function and providing graphic display of poles and zeros, which allows to check the stability of the IIR filters.
Thus the transfer function and the difference equation of this IIR filter are found to be, respectively:

$$H(z) - \frac{Y(z)}{X(z)} = \frac{0.662 - 0.662z^{-2}}{1 - 0.6739z^{-1} - 0.3241z^{-2}}$$

$$y(n) = 0.662x(n) - 0.662x(n-2) + 0.6739y(n-1) \\ +0.3241y(n-2)\,.$$

```
TabDataBP_IIR_1(0) = 0.662 * TabData(0)
TabDataBP_IIR_1(1) = 0.662 * TabData(1) +
0.6739 * TabDataBP_IIR_1(0)
For i = 2 To 1023
  TabDataBP_IIR_1(i) = 0.662 * TabData(i) –
  0.662 * TabData(i - 2) + 0.6739 * TabDataBP_IIR_1(i - 1) +
  0.3241 * TabDataBP_IIR_1(i - 2)
Next i
```

Fig. 15. VB program sequence.

The sequence of the following VB program (Fig. 15) provides an extract of the complete implementation of such an IIR filter.

2.3. GSM Communication

Radio frequency (RF) communication allows removing the gene of the patient caused by cables; on other hand, it permits to transmit important data over long distances. In our application, we are interested in transmitting an alarm warning of arrhythmia. As stated in the medical documents [25], the heart normal rhythm at rest is between 50 and 100 bpm (beats per minute), otherwise there is arrhythmia:

- < 50: bradycardia,
- > 100: tachycardia.

There are several wireless communication standards (GSM, Bluetooth, Wi-Fi, ZigBee, etc.). All use the microwave band. We have chosen GSM technology for practical considerations. Indeed, the GSM (Global System for Mobile Communications) has become ubiquitous in recent years. With the wide variety of GSM devices in the world market, standardization was necessary. Thus, the protocol "AT Commands" was normalized by ETSI (European Telecommunications Standards Institute), allowing different systems to communicate via this protocol, for example, an embedded system μC-based can generate an alarm message by SMS (Short Message Service). So, such communication has 3 layers:

- physical layer: RS232,
- link layer: asynchronous serial communication protocol character-based,
- application layer: AT protocol.

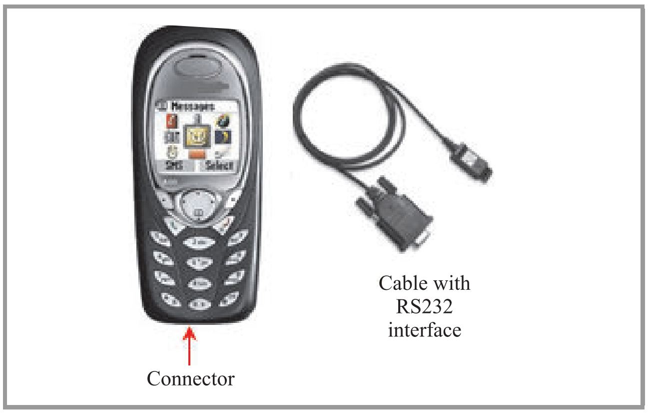


Fig. 16. Used GSM phone.

We have used the Siemens A60 GSM phone with embedded RS232 interface in cable (Fig. 16), allowing direct connection between GSM and PC or μC.

```
void main(){
unsigned char i ;
const unsigned char CmdCMGS[11]="AT+CMGS=19";
const unsigned char PayLoad[41]=
"0011000C911262450294760000A705D3303BDC06";
   Usart_Init(19200);
   TRISB=0x00;
   PORTB=0x80;
   Delay_ms(1000);
   PORTB=0x00;
   for(i=0;i<10;i++)
   {
    Usart_Write(CmdCMGS[i]);
   }
   Usart_Write(13);
   Delay_ms(3000);
   for(i=0;i<40;i++)
   {
    Usart_Write(PayLoad[i]);
   }
   Usart_Write(26);
   Delay_ms(3000);
}
```

Fig. 17. C code for sending SMS.

```
If ((HeartRate < 50) Or (HeartRate > 100)) Then
    MSComm1.Output = "AT+CMGS=57" & vbCr
    Delay 1000
    MSComm1.Output="0011000C91126245029476 _
    0000A731C2B75BFDAECB41C4F7985EAECB _
    5920EB9B2E2F83A0617ABAECA683C 2A072 _
    1D5477974161799E8E6EA7CB2E" & Chr(26)
     Delay 1000
End If
```

Fig. 18. VB code for sending SMS.

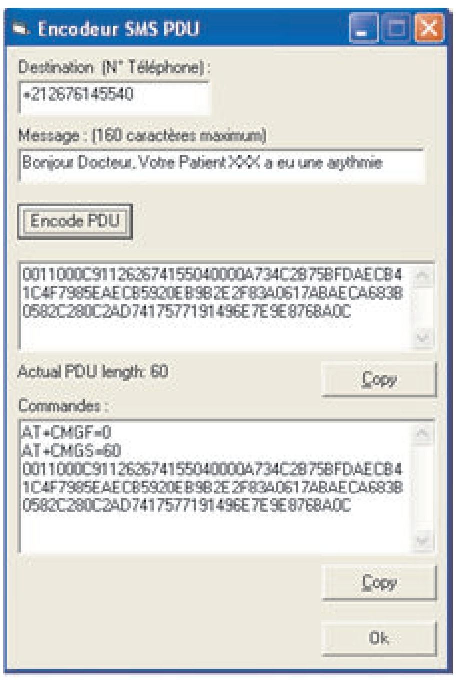


Fig. 19. PDU encoding.

AT commands form a protocol for controlling modems, mobile phone, GPRS, etc. This allows configuring, dialing numbers, sending messages, etc. [26], [27].
In our application, we need sending SMS; so, we use the commands “AT”, “AT + CMGF” and “AT + CMGS”. Any AT command starts with two characters ”AT” (ATtension), and finally the control character “Carriage Return” symbolized by <CR>. The following simple description gives the basic syntax of the three commands we have handled in our application.
The “AT” command is a status request used for testing if a compatible modem is connected and that the serial interface is working properly. This is like the well known “ping” command for communication test in the TCP/IP world.
The “AT+CMGF” command is used to set input and output format of SMS messages. Two modes are available:

- text mode: reading and sending SMS is done in plain text,
- PDU (Protocol Description Unit) mode: reading and sending SMS is done in a special encoded format. This mode compresses data to transmit allowing gain in time and space.

The “AT+CMGS” command enables the user to send SMS messages. A message can contain up to 160 7-bit characters. The transmission format is as follows:

“AT+CGMS=Message lenght”<CR> “PDU Message”<Ctrl+Z>

The PDU message is the text message to transmit after coding in PDU format (compression). We developed a program for μC that has remained at the stage of debugging (Fig. 17). In return, we transcribe the same program in VB6 (Figs. 18, 19), using with adaptation an open source implementation for PDU encoding algorithm [28]. Obviously, the GSM is connected to the PC serial port; the program gives a good result.

3. Summary

The ECG signal is captured by the analog module and sampled and digitized by the digital module, stored in EEPROM and treated (IIR and FIR filtering, measurement of heart rate by simple algorithm and communication with GSM). Figure 20 shows the results display on PC (unfiltered signal, filtered signal, heart rate). The signal can be recorded on hard disk and later reloaded for study purposes.
Compared with the works quoted in the introduction paragraph, briefly reflecting the state of the art, our project is prototype equipment having the following features: low cost, easy to implement and versatile. In fact, the hardware is made with cheap and well known components; regarding the software, it is also developed in known and widely disseminated environments. On one hand, this could contribute, on the commercial point of view, to democratize such often expensive equipment, especially in developing countries, which may help in diagnosis and improves

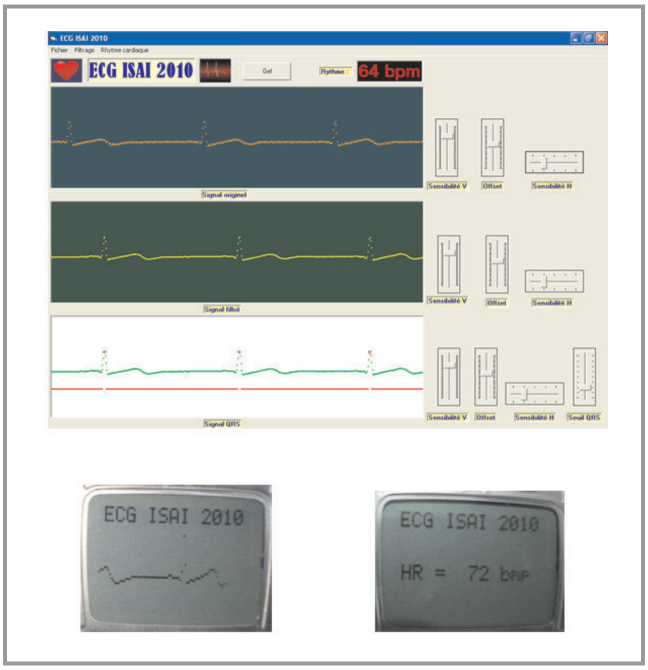


Fig. 20. Results display.

the level of health care; on the other hand, this could help novice researchers to touch closely and to be well acquainted with the different problems of biomedical engineering (hardware and software), in general and particularly for the cardiac signal (ECG). Furthermore, the system is naturally open and versatile. So, it can be certainly improved with dedicated hardware and reliable software.

4. Conclusions

We have developed an embedded system that can be used as ECG display equipment and arrhythmia monitor for patients at risk for heart attack. The system can be improved by using more specific components such as instrumentation amplifiers with better performances, dedicated microcontrollers with more speed and RAM space and RF technologies more suited to the medical context, such as the ZigBee standard.
Furthermore, the system has many tasks to be programmed, which requires a real-time operating system; this offers the opportunity of a certain parallelism in treatment. In this context of multitasking, a future study is underway to implement the system in a modern FPGA because of his reconfigurable characteristic. This approach represents the current trend thanks to the continuous development in the microelectronics science; it allows programming tasks with hardware parallelism [29]. The future work has as start point, like for any cardiac signal processing, the QRS detection with the efficient Pan/Tompkins algorithm.

References

[1] Physionet [Online]. Available: http://www.physionet.org/physiobank/database

[2] S. L. Toral, J. M. Quero, M. Elena, and L. G. Franquelo, "A microprocessor based system for ECG telemedicine and telecare", in *Proc. IEEE Int. Symp. Circuits Sys. ISCAS 2001*, Sydney, Australia, 2001, vol. 4, pp. 526–529.

[3] Welch Allyn OEM Technologies, "ECG/Respiration ASIC: ECG 3-lead, 5-lead, 12-lead and SP Signal Processing", (11/09/2003).

[4] D. Lucani, G. Cataldo, J. Cruz, G. Villegas, and S. Wong, "A portable ECG monitoring device with Bluetooth and Holter capabilities for telemedicine applications", in *Proc. 28th IEEE EMBS Ann. Int. Conf.*, New York City, USA, 2006.

[5] I. Romero Legameta, P. S. Addisson, N. Grubb, C. E. Robertson, K. A. A. Fox, and J. N. Watson, "Real-time classification of ECGs on a PDA", *IEEE Trans. Inform Technol. Biomed.*, vol. 30, pp. 565–568, 2003.

[6] R. Istepainan, R. García, A. Alesanco, and S. Olmos, "A novel real-time multilead ECG compression and de-noising method based on the wavelet transform", *IEEE Trans. Inform. Technol. Biomed.*, vol. 30, pp. 593–596, 2003.

[7] S. Borromeo, C. Rodriguez-Sanchez, F. Machado, J. A. Hernandez-Tamames, and R. de la Prieta, "A reconfigurable, wearable, wireless ECG system", in *Proc. 29th Ann. Int. Conf. IEEE EMBS*, Lyon, France, 2007.

[8] C.-N. Chien, H.-W. Hsu1, J.-K. Jang, C.-L. Rau1, and F.-S. Jaw, "Microcontroller-based wireless recorder for biomedical signals", in *Proc. 27th IEEE Ann. Conf. Engin. Medicine and Biology*, Shanghai, China, 2005.

[9] ECG Front-End Design is Simplified with MicroConverter'. Analog Devices Application Note [Online]. Available: http://www.analog.com/library/analogdialogue/archives/37-11/ecg.pdf

[10] U. Rajendra Acharya, A. E. Spaan, J. Suri, and S. M. Krishnan, *Advances in Cardiac Signal Processing*. Springer, 2007.

[11] T. Degen and H. Jäckel, "A pseudodifferential amplifier for bioelectric events with DC-offset compensation using two-wired amplifying electrodes", *IEEE Trans. Biomed. Eng.*, vol. 53, no. 2, pp. 300–310, 2006.

[12] J. Webster, *Medical Instrumentation Application and Design*. Wiley, 2010.

[13] Microchip. PIC16F876 Datasheets [Online]. Available: http://ww1.microchip.com/downloads/en/devicedoc/30292c.pdf

[14] Microchip. 24LC256 Datasheets [Online]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/21203P.pdf

[15] Philips Semiconductors. The i 2c-bus specification [Online]. Available: http://www.nxp.com/documents/other/39340011.pdf

[16] MAXIM. Max232 Datasheets [Online]. Available: http://www.datasheetcatalog.net/datasheets_pdf/M/A/X/2/MAX232.shtml

[17] Nokia 3310 LCD open source [Online]. Available: http://webspace.webring.com/people/fd/dariuskrail20/nokia3310lcd.html

[18] Philips Semiconductors. PCD8544 Datasheets [Online]. Available: http://www.nxp.com/acrobat_download2/datasheets/PCD8544_1.pdf

[19] AAMI, American National Standard, Safe Current Limits for Electromedical Apparatus (ANSI/AAMI ES1-1993). Association for the Advancement of Medical Instrumentation, 1993.

[20] J. Pan and W. J. Tompkins, "A real time QRS detection algorithm", *IEEE Trans. Biomed. Eng.*, vol. 32, pp, 230–236.

[21] Heart-Rate and EKG Monitor Using the MSP430FG439. Texas Instruments Application Report, 2008.

[22] Li Tan, *Digital Signal Processing: Fundamentals and Applications*. Elsevier, 2008.

[23] P. S. R. Diniz, E. A. B. da Silva, and S. L. Netto, *Digital Signal Processing: System Analysis and Design*, Cambridge University Press, 2002.

[24] L. Sornmo and P. Laguna, B*ioelectrical Signal Processing in Cardiac and Neurological Applications*. Elsevier, 2005.

[25] A. J. Guyton and J. E. Hall, *Textbook of Medical Physiology*. Elsevier, 2006.

[26] On the use of AT commands for controling the mobile phone with microcontroller [Online]. Available: http://www.pwt.et.put.poznan.pl/srv07/papers/PWT%202007_7835.pdf

[27] AVR323: Interfacing GSM modems. ATMEL Apllication Note [Online]. Available: http://www.atmel.com/dyn/resources/prod_documents/doc8016.pdf

[28] SMS PDU open source [Online]. Available: http://www.scampers.org/steve/sms/samples.htm

[29] B. Halak, J. Murphy, and A. Yakovlev, Electrocardiogram Monitor Design from Concept to Silicon. Technical Report Series. Newcastle University, 2011.

El Mimouni El Hassan received the B.Sc. in Electronics in Ecole Normale Supérieure d'Enseignement Technique (ENSET) in Rabat, Morocco, in 1986. He received the M.Sc. in Process Control, in University of Fès, Morocco, in 2010. Currently, he is with EVIA research group (www.fsdmfes.ac.ma) as Ph.D. student. His interests cover cardiac digital signal processing and software development.
E-mail: elmihass@gmail.com
University Sidi Mohammed Ben Abdellah
Fès, Morocco

Mohammed Karim is a professor of computer science and electronics at the Faculty of Sciences in University of Fès, Morocco. He is a head of the EVIA research group (www.fsdmfes.ac.ma), and a head of the ISAI Master. He is a chairman of the organization committee of a series of workshops and confrences. He is currently working on different projects: E-learning, Computer Vision, Biomedical engineering.
E-mail: karim_lessi@yahoo.fr
University Sidi Mohammed Ben Abdellah
Fès, Morocco

Paper

Secure-Sim-G: Security-Aware Grid Simulator – Basic Concept and Structure

Grzegorz Gębczyński[a], Joanna Kołodziej[a], and Samee Ullah Khan[b]

[a] Faculty of Mechanical Engineering and Computer Science, University of Bielsko-Biała, Bielsko-Biała, Poland
[b] Department of Electrical and Computer Engineering, North Dakota State University, USA

Abstract— Task scheduling and resource allocation are the key issues for computational grids. Distributed resources usually work at different autonomous domains with their own access and security policies that impact successful job executions across the domain boundaries. In this paper we present a security-aware grid simulator Secure-Sim-G, which facilitates the evaluation of the different scheduling heuristics under various scheduling criteria in several grid scenarios defined by the security conditions, grid size and system dynamics. The simulator allows the flexible activation or inactivation of all of the scheduling criteria and modules, which makes the application well adapted to the proper illustration of the different realistic scenarios and avoids the possible restriction to the specific scheduling resolution methods. The simulation results and traces may be graphically represented and stored at the server and can retrieved in different formats such as spreadsheets or pdf files.

Keywords—computational grid, grid simulator, scheduling, security.

1. Introduction

Grid computing has emerged as a wide area distributed platform for solving the large-scale problems in science and engineering. Computational Grid (CG) involves the combination of many computing resources into a network for the execution of computational tasks. The resources are distributed across multiple organizations, administrative domains with their own access and usage policies. The effective task scheduling and the management of the grid resources are the complex key issues for CGs. They usually demand sophisticated tools for analyzing the algorithms performances before applying them to the real systems, which is in fact necessary in the case when the main scheduling objectives, such as the maximization of the resource utilization and profits of the resource owners, may conflict with grid users' security requirements and system reliability. The grid resource may not be accessible if the grid network or grid cluster is under an external attack or the grid users' security priorities in scheduling may not cope with the offer of the resource providers. Therefore, it is desirable to have a prior knowledge about the security demands from submitted applications and the trust level assured by a resource provider at the grid cluster. An effective grid scheduler must be then security-driven and resilient in response to all scheduling and risky conditions. It means that in order to achieve the successful tasks executions according the specified users' requirements, the relations between the assurance of secure computing services by a grid site or by a cluster node (security) and the behavior of a resource node (trust) must be defined, analyzed and simulated in all possible scenarios.

Simulation seems to be the most effective solution for the comprehensive analysis of the security-aware scheduling algorithms in large-scale distributed dynamic systems such as grid or cloud environments. It simplifies the study of schedulers performances and avoids the overhead of coordination of the resources, which usually happens in the real-life grid or cloud scenarios. Simulation is also effective in working with very large problems that require the involvement of a large number of active users and resources, which is usually very hard for the management in real-life approaches. In such cases a considerable number of independent runs is needed to ensure significant statistical results, that can be easily realized with the system simulator.

In this work, we present the main concept of a security-aware grid simulator Secure-Sim-G for independent batch scheduling, which is an extension and modification of the HyperSim-G framework [1]. Secure-Sim-G is an event-based application, which facilitates the evaluation of the different scheduling heuristics under various scheduling criteria in several grid scenarios defined by the security conditions, grid size and system dynamics. The simulator allows the flexible activation or inactivation of all of the scheduling criteria and modules, which makes the application well adapted to the proper illustration of the different realistic scenarios and prevents the possible restriction to the specific scheduling resolution methods. The simulation results and traces may be graphically represented and stored at the server and can retrieved in different formats such as spreadsheets or pdf files. The simulator structure enables an easy association with the external or internal embedded database systems for storing the historical executions, which allows a comprehensive study of the use cases for different types of grid schedulers. In order to illustrate the impact of the security conditions on the scheduling results, we provided a simple evaluation analysis of the simulator by using two risk-resilient metaheuristic-based schedulers under the varying heterogeneity and large-scale system dynamics.

Table 1
Main attributes of grid scheduling

Attribute	Type	Brief description
Environment	Static	The number of resources is fixed and all of them are available
	Dynamic	The availability of the resources can dynamically change
Grid architecture	Centralized	The schedulers have a full knowledge and control over resources
	Decentralized	No central entity controlling the resources, the local schedulers are responsible for managing and maintaining the tasks
	Hierarchical	The coordination of different schedulers at certain levels, the full knowledge of resources available for the schedulers at the lowest level
Task processing policy	Immediate	Tasks are scheduled as soon as they enter the system
	Batch	Available tasks are grouped into batches and the scheduler assign the batch to the resources
Tasks interrelations	Independency	Tasks are scheduled independently of each other
	Dependency	There are precedence constraints among tasks
Security conditions	Risky mode	All risky and failing conditions are ignored
	Secure mode	All security and resource reliability conditions are verified for the possible task-machine pairs

The rest of the paper is organized as follows. Related work is discussed in Section 2. The main types of the grid scheduling problems are defined in Section 3. The concept of the Secure-Sim-G and its main modules and parameters are presented in Section 4. We report the results of simple experimental analysis in Section 5. We summarize and conclude our work in Section 6.

2. Related Work

Using the simulators for an evaluation of the grid schedulers is feasible, mainly because of high complexity of the grid environment. Many simulation packages, useful in the design and analysis of scheduling algorithms in grid systems, have been recently proposed in the literature. Among many others, MicroGrid [2], ChicSim [3] and Grid-Sim [4] seem to be the most popular in the domain. Some of them are integrated with the grid portals in order to provide the users with an easy access to the simulation packages as well as the online monitoring of the scheduling process. A web-based platform for simulating scheduling methods in grid computing with Grid-Sim package was proposed in [5].

The security aspects in grid scheduling in risky environments are explored in numerous research. Song et al. ([6] and [7]) developed a security aware model in online grid scheduling, where security demand and trust levels are expressed as scalar parameters. Humphrey and Thompson presented [8] a classification of security-aware grid models for an immediate job execution mode. They define a job control system for accessing grid information services through authentication. However, they did not elaborate on how a scheduler should be designed to address the security concerns in collaborative computing over distributed cluster environment. An extensive survey of the research endeavors in this domain is presented in [9].

In [10] the authors present an approach on fault-tolerance method in CG scheduling. They provided a failure detection service, which enables the detection of both task failures and user secure requirements, and a flexible failure handling framework as a fault-tolerant mechanism on the grid. Abawajy [11] developed a model, in which jobs are replicated at multiple grid sites to improve the probability of the satisfaction of the security requirements and successful job executions. Resource reliability and security have been defined as additional scheduling criteria in independent grid scheduling [12], [13], [14]. Matching grid users' security requirements and the "reputation" of the grid clusters impact the behavior and strategies of users, task managers, and resource brokers. A game-theoretical support to the users' decision making activities was presented in our previous work [15].

3. Scheduling problems in CGs

The main purpose of the schedulers is an efficient and optimal allocation of tasks originated by applications to a set of available resources. In dynamic large-scale heterogeneous environments both tasks and resources could be dynamically added/dropped to/from the system. Additionally, various system's entities such as the system's users, managers, and resource providers may operate in different autonomous domains with incoherent local policies. Therefore scheduling in grids is usually considered as a family of highly parametrized problems. The type of the scheduling problem in CGs is specified by setting up the main scheduling attributes presented in Table 1.

In this paper, we focus on an Independent Batch Scheduling in Hierarchical CG problem, where it is assumed that the tasks are grouped into batches and can be executed independently in a hierarchical multi-level grid system in both static and dynamic modes. We consider two possible security scenarios: risky and secure modes. Due to the massive capacity of parallel computation in CGs, this kind of scheduling is very useful in illustrating a lot of realistic scenarios, where the users, independent of each other, submit their jobs to the system, all grid-enabled applications run periodically, and large amounted of data are simultaneously transferred, replicated, and accessed by those applications.

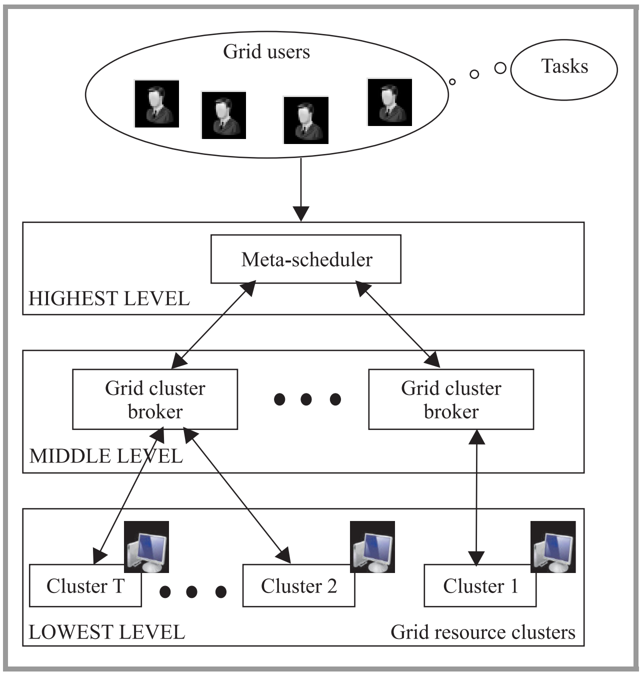


Fig. 1. The model of hierarchic grid architecture

The multi-level large-scale hierarchical structure of CG usually consists of two or three levels as it is shown in Fig. 1.

A central meta-scheduler is the main module of the system that works at the highest level. In today's grid applications a region of activity of the meta-scheduler is in fact restricted to a wide grid cluster, so all global network is managed by few fully cooperated meta-schedulers. The meta-scheduler interacts with local task dispatchers (brokers) and CG users in order to generate the optimal schedules. If "security" is considered as an additional scheduling criterion, the meta-scheduler must analyze the security requirements for the execution of tasks and requests of the CG users for trustful resources available in the system. The system brokers collect information about the "computing capacities" of the resources supplied by the resource owners within the clusters, and additionally analyze the "reputation" indexes of the machines received from the resource managers. They moderate the resources, and send all of the data to the meta-scheduler. The brokers also control the resource allocation and communication between CG users and resource owners.

4. Security Aware Grid Simulator – Basic Concept

To simulate the secure independent batch scheduling we developed a Secure-Sim-G simulator by extending the HyperSim-G framework [1]. HyperSim-G simulator is based on a discrete event model. The sequence of events and the changes in the state of the system capture the realistic grid dynamics. The simulator provides the full simulation trace by indicating a parameter for the trace generation.

The main concept of the Secure-Sim-G simulator is presented in Fig. 2.

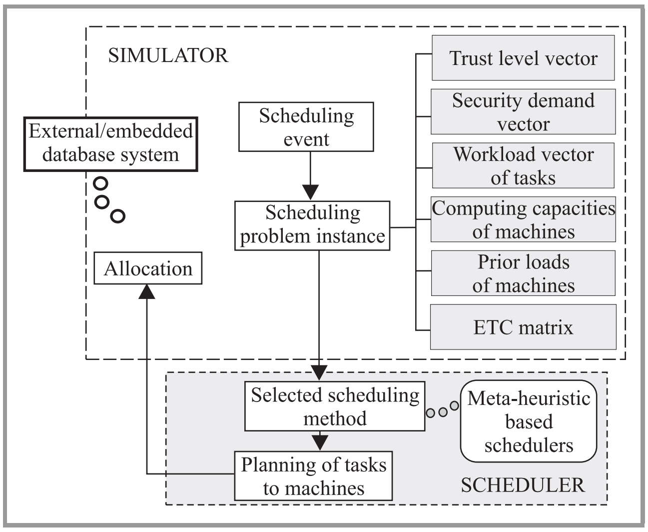


Fig. 2. General flowchart of the Secure-Sim-G simulator linked to scheduling.

There are two main modules in the system, namely Simulator module and Scheduler module. The main simulation flow can be defined as follows. When a scheduling event is triggered, the simulator creates an instance of the scheduling problem, based on the current task batch and the pool of available machines. The simulator computes an instance of the scheduling and passes it on to a given scheduler which computes the planning of tasks to machines. Finally, the scheduler sends the schedules back to the simulator, which makes the allocation and re-schedules any tasks assigned to machines not available in the system.

The main structure of the Secure-Sim-G application is based on the 2-module HyperSim-G architecture. We modified the Simulator module used in HyperSim-G by defining a security submodule, which allows to specify the security conditions and to define the settings of the risky and secure scheduling scenarios. The secure Simulation submodule is also equipped with the resource failure monitoring system, which reports the unsuccessful resource allocation results due to too strong security requirements. The scheduling methods in Scheduler module defined in HyperSim-G are extended in Secure-Sim-G by the verification of the security conditions and recalculating the values of the objective functions. The software is written in C++ for Linux

Ubuntu 10.10. In the following subsections we briefly characterize the main simulator modules and parameters.

4.1. Input Data and Scheduling Instance

The Secure-Sim-G simulator generates an instance of the scheduling problem by using the following input data:

- the trust level vector of the machines,
- the security demand vector of tasks,
- the workload vector of tasks,
- the computing capacity vector of machines,
- the vector of prior loads of machines,
- the ETC matrix of estimated execution times of tasks on machines.

The number of tasks in a given batch and the number of machines must be also specified. These parameters are constant in the static scheduling and may vary in the dynamic case. For the dynamic scheduling we defined the probability distributions in order to estimate the changes in the system states. We implemented the Constant, Triangle, Normal, Exponential, Trace, Zipf and Uniform distributions for this purpose.

The task in our system are defined as monolithic applications or metatasks with no dependencies among the components. Each task j is characterized by the following parameters:

- wl_j is a computational load of j expressed in millions of instructions per second (MIPS), we denote by $WL = [wl_1, \ldots, wl_n]$ a *workload vector* for all tasks in the batch,
- sd_j is a security demand parameter, which is a component of a *security demand vector* $SD = [sd_1, \ldots, sd_n]$.

Each machine $i, (i \in M_l)$ in the system is characterized by the following three parameters:

- cc_i – is a computing capacity of i expressed in millions of instructions per second (MIPS), we denote by $CC = [cc_1, \ldots, cc_m]$ a *computing capacity vector*,
- $ready_i$ – is a ready time of i, which expresses the time needed for the reloading of the machine i after finishing the last assigned task, a *ready times vector* for all machines is denoted by $ready\_times = [ready_1, \ldots, ready_m]$,
- tl_i – is a trust level parameter, which specifies how much a grid user can trust a given resource manager and is the component of a *trust level vector* $TL = [tl_1, \ldots, tl_m]$.

The trust level and security demand parameters are expressed as scalar quantities, which are generated by the aggregation of several scheduling and system attributes at users' and resource owners' sites. We base our approach on the fuzzy-logic trust model developed by Song *et al*. [7]. In this model the task security demand is supplied by the user programs as a single parameter. The demand may appear as request for authentication, data encryption, access control, etc.

The values of the sd_j and tl_i parameters are real fractions within the range [0,1] with 0 representing the lowest and 1 the highest security requirements for a task execution and the most risky and fully trusted machine, respectively. A task can be successfully completed at a resource when a *security assurance condition* is satisfied. That is to say that $sd_j \leq tl_i$ for a given (j,i) task-machine pair.

SD and TL vectors are used for generation of a *Machine Failure Probability* matrix P_f, the elements of which, are interpreted as the probabilities of failures of the machines during the tasks executions due the high security restrictions. These probabilities, denoted by $P_f[j][i]$, are calculated by using the negative exponential distribution function as follows:

$$P_f[j][i] = \begin{cases} 0, & sd_j \leq tl_i \\ 1 - e^{-\alpha(sd_j - tl_i)}, & sd_j > tl_i \end{cases} \tag{1}$$

where α is interpreted as a failure coefficient and is a global parameter of the model.

For estimating the execution times of tasks on machines we used the *Expected Time to Compute (ETC)* matrix model [16]. The elements of the ETC matrix, $ETC = [ETC[j][i]]_{n \times m}$ are defined as the expected (estimated) times needed for the completion of the tasks on machines.

In the simplest case, these times can be computed as the ratios of the proper coordinates of WL and CC vectors. That is to say:

$$ETC[j][i] = \frac{wl_j}{cc_i}. \tag{2}$$

All of the values of wl_j and cc_i are generated by using the Gamma or simple Gaussian probability distributions for the expression of tasks and machines heterogeneities in the grid system. In cases when: (a) meta-tasks are submitted by the users and (b) multiprocessor machines are proposed by the resource providers, the values of ETC matrix can be computed by using some special local scheduling policies and resolution methods.

4.2. Resolution Methods

The *Secure-Sim-G* simulator allows and facilitates integration of different scheduling implementations. The design of the simulator enables scheduling algorithms to be decoupled from the simulator main body. Various types of the evolutionary based grid schedulers are plugged in the simulator by using and *Adapter* pattern as it is presented in Fig. 3.

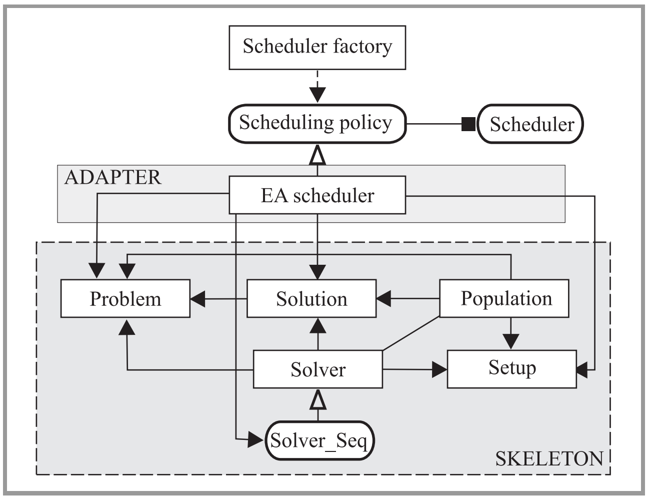


Fig. 3. The simulator adapter pattern used for different evolutionary based grid schedulers

We divided the implemented scheduling heuristic into three main classes, namely *ad hoc*, *local search-based*, *population-based* heuristics.

Ah hoc methods. These methods are usually used for single-objective optimization. They are simple and distinguished from their low computational cost, thus, they are also very useful in generating the initial solutions for population-based schedulers. The ad hoc heuristics could be grouped into an immediate mode heuristics and batch mode heuristics.
The *Immediate Mode Heuristics* group includes, among others, the following schedulers:

- *Opportunistic Load Balancing (OLB)*, where a task is assigned to the earliest idle machine without taking into account its execution time in the machine.

- *Minimum Completion Time (MCT)*, in which a task is assigned to the machine yielding the earliest completion time.

- *Minimum Execution Time (MET)*, in which a task is assigned to the machine having the smallest execution time for that task.

The *Batch Mode Heuristics* group contains, among others, the following methods:

- *Min-Min*: In this method for each task the machine yielding the earliest completion time is computed, then the task with the shortest completion time is selected and mapped to the corresponding machine.

- *Max-Min*: This method differs to the Min-Min in the final selection of the task with the latest completion time.

- *Sufferage*: The main idea of this method is to assign to a given machine a task, which would "suffer' more if it were assigned to any other machine.

- *Relative Cost*: In allocating tasks to machines, this method takes into account both the load balancing of machines and the execution times of tasks in machines.

- *Longest Job to Fastest Resource - Shortest Job to Fastest Resource (LJFR-SRFR)*: This method tries to simultaneously minimize both makespan and flowtime values: LJFR minimizes makespan and SJFR minimizes flowtime.

Local search methods. These methods explore the optimization domain by starting from an initial solution and constructing a path in solution space. The most effective local-based grid scheduler is Tabu Search (TS) due to its mechanisms of tabu lists, aspiration criteria, intensification and diversification (see [17]). TS can be easily hybridized with more sophisticated schedulers (like GAs) to improve their efficiency.

Population-based heuristics. In this methods a population of individuals, which is evaluated, crossed over and mutated, is used to explore the solution space for the problem along a number of generations. The most popular in this group are Genetic Algorithms (GA), proposed by many authors [18], [19], [20]. Recently, a multi-population hierarchical GA-based scheduler has been defined in [21], [22]. In this method a set of dependent genetic processes is executed simultaneously. Each process creates a branch in the tree structure of the whole strategy, by using the GA-based scheduler with different settings. The search accuracy in a given branch (expressed as the branch degree parameter) depends on the mutation probability set for the scheduler activated in this branch (the higher mutation prob. – the lower accuracy).

4.3. Schedule Representation and Scenarios

We use in our approach two different encoding methods of schedules, namely *direct encoding* and *permutation-based encoding.* In the direct encoding the coordinates of a schedule vector $x = [x_1, \ldots, x_{nb\_task}]^T$ are defined as the indexes of machines to which the particular tasks are assigned. In permutation-based encoding we define for each machine a sequence of tasks assigned to that machine. The tasks in the sequence are increasingly sorted with respect to their completion times. Then, all of the task sequences are concatenated into one global vector $u = [u_i, \ldots, u_{nb\_task}]^T$; $u_i \in Tasks$, which is in fact the permutation of tasks to machines. In this representation some additional information about the numbers of tasks assigned to each machine is required. We defined then the vector $v = [v_1, \ldots, v_{nb\_machines}]^T$, in which the numbers of tasks assigned to the following machines are specified as its coordinates.

4.3.1. Optimization Criteria and Objective Function

The problem of scheduling tasks in CG is multiobjective in its general setting as the quality of the solutions can be measured under several criteria. In this work, for the purpose of a simple experimental evaluation of the simulator, we consider the scheduling in CGs as a bi-objective global optimization problem with the hierarchical procedure of the minimization of makespan and flowtime objectives with makespan as a privileged criterion.
Let us denote by F_j the time of finalizing task j and let *Schedules* be a set of directly encoded schedules in a given batch.

- A *makespan* is defined as the finishing time of the latest task in the batch. That is to say:

$$makespan = \min_{s \in Schedules} \max_{j \in N_l} F_j, \tag{3}$$

- We define a *flowtime* as the sum of the finalization times of all the tasks in the batch in the following way:

$$flowtime = \min_{s \in Schedules} \sum_{j \in N_l} F_j\,. \tag{4}$$

Both makespan and flowtime are expressed in arbitrary time units. In fact, the numerical values are in incomparable ranges: flowtime has a higher magnitude order over makespan and its values increase as more jobs and machines are considered. Therefore, in this approach we use $mean\_flowtime = flowtime/m$ for the evaluation of the flowtime criterion.
Using the *ETC* matrix model we can express the makespan and flowtime in terms of the completion times of the machines. The time of finishing the last task can be interpreted as the maximal completion time of the machines. Let us denote by *completion* a vector of the size *nb_machines*, which indicates the time that machine i finalizes the processing of the previously assigned and planned tasks. That is to say:

$$completion[i] = ready_i + \sum_{\substack{j \in N_l:\\ s[j]=i}} ETC[j][i]. \tag{5}$$

The makespan can be now expressed as:

$$makespan = \max_{i \in M_l} completion[i]. \tag{6}$$

We calculate the flowtime of the sequence of tasks on a given machine i by using the following formula:

$$\begin{aligned} flowtime[i] = \quad & ready_i + \\ & + \textstyle\sum_{\substack{j \in Sort[i]:\\ s[j]=i}} ETC[j][i] \end{aligned} \tag{7}$$

where $Sort[i]$ denotes the set of tasks assigned to the machine i sorted in ascending order according to their *ETC* values.

Having makespan and flowtime as two main scheduling criteria, we define the objective of the scheduling problem as the following function:

$$obj = \lambda \cdot makespan + (1-\lambda) \cdot mean\_flowtime. \tag{8}$$

The weight coordinate λ is used in fact for the specification of the priority of the considered scheduling criteria. Following the experimental tuning results presented in [23] for a classical independent scheduling problem we set the λ value as 0.75. That is to say that in our approach the makespan is the preferred schedulers performance measure. The Secure-Sim-G simulator allows the users to define and integrate the other scheduling criteria, such as the resource utilization and matching proximity [1].

4.3.2. Scheduling Scenarios

To express the impact of the verification of security condition to the scheduling results, we consider two scheduling scenarios, namely *secure* and *risky* modes. In security mode the scheduler analyzes the *Machine Failure Probability* matrix in order to minimize the failure probabilities for task-machine pairs. We assume that additional "cost" of the verification of security assurance condition for a given task-machine pair: (a) may delay the predicted execution time of the task on the machine and (b) is proportional to the probability of failure of the machine during the task execution. We define this "cost" as a product $P_f[j][i] \cdot ETC[j][i]$ and the completion time of the machine i can be calculated as follows:

$$\begin{aligned} & completion[i] = ready\_time[i] + \\ & + \sum_{\{j \in Tasks_i\}} (1 + P_f[j][i]) ETC[j][i]\,, \end{aligned} \tag{9}$$

where $Tasks_i$ denotes a set of tasks assigned to the machine i in a given batch.
In risky mode the scheduler performs as an "ordinary" scheduler without any prior analysis of the security conditions. It aborts the task scheduling in the case of machine failure, and reschedules this task at another resource. It means that the scheduling is performed just by analyzing the *ETC* matrix. If failures are observed during some tasks executions, then the unfinished tasks are temporarily moved into the backlog set. This set is defined as a considered batch supplement and the tasks are re-scheduled as in the secure mode. The total completion time of machine i in this case can be defined as follows:

$$\begin{aligned} & completion^r[i] = \\ & completion_{(I)}[i] + completion_{(II)}[i]\,, \end{aligned} \tag{10}$$

where $completion_{(I)}$ is the completion time of machine i calculated by using the Eq. (5) for tasks primarily assigned to the machine, and $completion_{(II)}$ is the completion time of machine i calculated by using the Eq. (9) for rescheduled tasks, i.e., the tasks moved to the machine i from the other resources.

5. Experimental Analysis

In this section we present the results of a simple experimental evaluation of two variants of genetic-based grid schedulers working in risky and secure modes in static and dynamic grid environments. The setting and configuration of GA scheduler are presented in Table 2.

Table 2
GA settings for large static and dynamic benchmarks

Parameter	Value
Evolution steps	$5 \times (nb\_jobs)$
Population size (pop_size)	$\lceil (\log_2(nb\_jobs))^2 - \log_2(nb\_jobs) \rceil$
Intermediate pop.	$pop\_size - 2$
Selection method	LinearRanking
Crossover method	Cycle Crossover
Cross probab.	0.9
Mutation method	Rebalancing
Mutation probab.	0.2
$replace\_only\_if\_better$	false
$replace\_generational$	false
Initialization	LJFR-SJFR + MCT + Random
$max\_time\_to\_spend$	40 s $(static)$ / 25 s $(dynamic)$

The detailed definition of the genetic operators used in GA configuration can be found in [24].

The Secure-Sim-G simulator is highly parametrized to reflect the various realistic grid scenarios. The values of key input parameters[1] for the simulator are presented in Table 3.

We considered the following four grid size scenarios in our study: (a) small grid (32 hosts/512 tasks), (b) medium grid (64 hosts/1024 tasks), (c) large grid (128 hosts/2048 tasks), and (d) very large grid (256 hosts/4096 tasks).

The user can specify his own scenario by changing the number of tasks and machines. The capacity of the resources and the workload of tasks are randomly generated by a normal distribution. It is also assumed that all tasks submitted to the system must be scheduled and all machines in the system can be used.

The number of hosts initially activated in the grid environment is defined by the parameter *Init. number of hosts*. The parameters *Max.hosts* and *Min.hosts* specify the range of changes in the number of active hosts during the simulation process[2]. The frequency of appearing and disappearing resources is defined by *Add host* and *Delete host*, according to constant distributions for the static case, and normal distributions in dynamic case. The initial number of tasks is given by *Init. tasks*, which is kept constant in the static case. New tasks in the dynamic scheduling can arrive at the system with the frequency *Interarrival* until *Total tasks* is reached. The *Activation* parameter establishes the activation policy (it is usually modeled by an exponential distribution in the dynamic case). The assigned tasks which have not been executed yet cannot be rescheduled if the value of the boolean parameter *Reschedule* is false. The *Scheduler strategy* parameter denotes the Scheduler type. Its value $GA\_Scheduler(25, s)$ means that the simulator runs the GA-based scheduler for 25 s in simultaneous optimization mode[3].

We used the following three metrics to evaluate the scheduling performance:

- *Makespan* (see Eq. 3) for secure and risky scenarios,
- $Flowtime$ (see Eq. 4) for secure and risky scenarios,
- *FailureRate* F_r parameter defined as follows:

$$F_r = \frac{n_{failed}}{n} \cdot 100\%, \qquad (11)$$

where n_{failed} denotes the number of unfinished tasks, which must be rescheduled.

Each experiment was repeated 30 times under the same configuration of operators and parameters.

5.1. Results

The results of the experiments expressed by the averaged flowtime and makespan values achieved by two variants of security-aware GA-based schedulers in static and dynamic cases are presented in Fig. 4.

It can be observed that in both static and dynamic cases the secure version of GA-based scheduler, namely *GA-Secure* algorithm, outperforms its risky variant. The differences are significant especially for the makespan values. The results suggest that it is more resilient for the grid users to pay some additional scheduling cost due to verification of the security conditions instead of taking a risk and allocating them at untrustful resources. It can be also noted that as the instance size is doubled, the flowtime values increase considerably for all applied schedulers, while the makespan is almost at the same level. The good results in secure scenario are confirmed by the low failure rates achieved by the *GA-Secure* scheduler presented in Table 4.

In all instances the values achieved by secure scheduler are approximately two times lower than in risky scenario,

[1] We use the notation $U[x,y]$, $N(a,b)$ and $E(c,d)$ for uniform, Gaussian and exponential probability distributions respectively.

[2] In the case of dynamic scheduling, they are different from the initial number of hosts.

[3] Similarly, the parameter h can be used to indicate hierarchic mode optimization, e.g., $GA\_Scheduler(25, h)$.

Table 3
Values of key parameters of the grid simulator in static and dynamic cases

Parameter	Small	Medium	Large	Very large
Static case				
Number of hosts	32	64	128	256
Resource cap. (in MHz CPU)	$N(5000, 875)$			
Total nb. of tasks	512	1024	2048	4096
Workload of tasks	$N(250000000, 43750000)$			
Security demandss d_j	$U[0.6; 0.9]$			
Truest levels tl_i	$U[0.3; 1]$			
Failure coefficient α	3			
Dynamic case				
Init. number of hosts	32	64	128	256
Max.hosts	37	70	135	264
Min.hosts	27	58	121	248
Resource cap. (in MHz CPU)	$N(5000, 875)$			
Add host	$N(625000, 93750)$	$N(562500, 84375)$	$N(500000, 75000)$	$N(437500, 65625)$
Delete host	$N(625000, 93750)$			
Init. tasks	384	768	1536	3072
Total tasks	512	1024	2048	4096
Interarrival	$E(7812.5)$	$E(3906.25)$	$E(1953.125)$	$E(976.5625)$
Workload	$N(250000000, 43750000)$			
Security demandss d_j	$U[0.6; 0.9]$			
Trust levels tl_i	$U[0.3; 1]$			
Failure coefficient α	3			

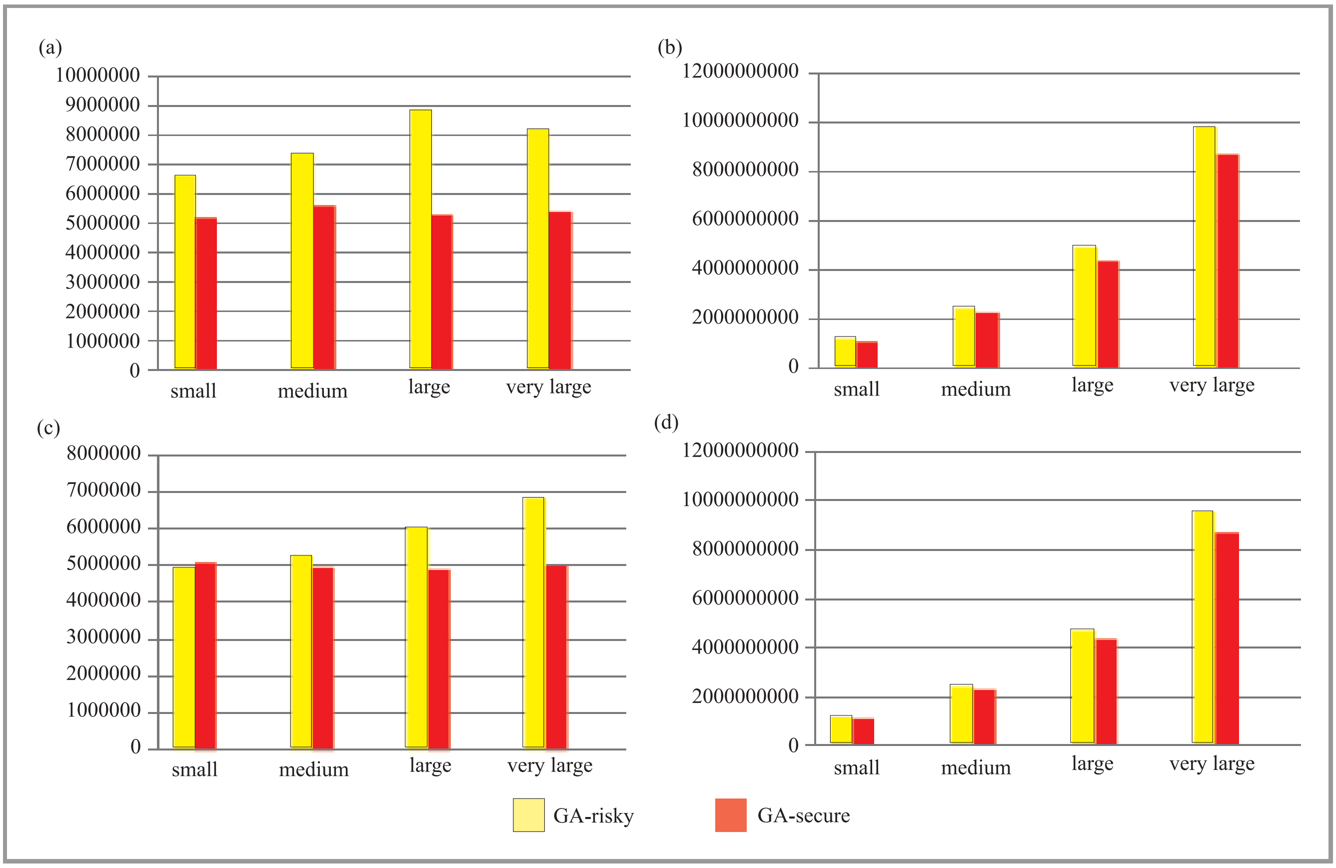


Fig. 4. Experimental results achieved by security-aware GA-schedulers: in static case – (a) average makespan, (b) average flowtime; in dynamic case – (c) average makespan, (d) average flowtime.

what may explain the additional scheduling costs in this case, i.e., many tasks have been re-scheduled.

Table 4
Average values of failure rate parameter for six GA-based schedulers

Strategy	Small	Medium	Large	Very large
Static Instances				
GA-Risky	15.632	18.405	9.351	20.345
GA-Secure	5.682	9.415	6.134	8.435
Dynamic Instances				
GA-Risky	19.522	24.265	28.563	25.455
GA-Secure	10.223	12.635	11.546	10.535

6. Conclusions

In this paper we presented the main concept, architecture and parameters of the *Secure-Sim-G* grid simulator for independent batch scheduling, which allows the evaluation of the different scheduling heuristics under various scheduling criteria in several grid scenarios defined by the security conditions, grid size and system dynamics. The simulator is an extension and modification of the event-based *HyperSim-G* framework [1] With *Secure-Sim-G* the user can flexibly activate and inactivate all the scheduling criteria and modules, which makes the application well adapted to the proper illustration of the different realistic scenarios and reduces the possible restriction to the specific scheduling resolution methods.

We provided a simple evaluation analysis of the simulator by using two risk-resilient metaheuristic-based schedulers under the varying heterogeneity and large-scale system dynamics. The results suggest that it is more resilient for the grid users to pay some additional scheduling cost due to verification of the security conditions instead of taking a risk and allocating them at untrustful resources.

The presented software package can be easily extended by plugging in additional scheduling methods and scheduling criteria, like energy consumption, which a hot research topic in intelligent green networking ([25]–[32]). It can be also adapted to the scheduling simulation in cloud systems, which will be our next research effort.

References

[1] F. Xhafa, J. Carretero, L. Barolli, and A. Durresi, "Requirements for an event-based simulation package for grid systems". *J. Interconnection Netw.*, vol. 8, no 2, pp. 163–178, 2007.

[2] H. J. Song, X. Liu, O. Jakobsen, R. Bhagwan, X. Zhang, K. Taura, and A. Chien, "The microgrid: a scientific tool for modeling computational grids", *J. Sci. Program.*, vol. 8, no. 3, pp. 127–141, 2000.

[3] K. Ranganathan and I. Foster, "Simulation studies of computation and data scheduling algorithms for data grids", *J. Grid Comput.*, vol. 1, no. 1, pp. 53–62, 2003.

[4] R. Buyya, and M. M. Murshed, "Grid-Sim: a toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing", *Concurrency and Comput.: Practice and Experience*, vol. 14, no. 13–15, pp. 1175–1220, 2002.

[5] O. Kang and S. Kang, "Web-based dynamic scheduling platform for grid computing", *Intern. J. Comp. Sci. Netwo. Secur.*, vol. 6, no. 5, pp. 67–75, 2006.

[6] S. Song, K. Hwang, and Y. K. Kwok, "Risk-resilient heuristics and genetic algorithms for security- assured grid job scheduling", *IEEE Trans. Comput.*, vol. 55, no. 6, pp. 703–719, 2006.

[7] S. Song, K. Hwang, and Y. K. Kwok, "Trusted grid computing with security binding and trust integration", *J. Grid Comput.*, vol. 3, no. 1-2, pp. 53–73, 2005.

[8] M. Humphrey and M. R. Thompson, "Security implications of typical grid computing usage scenarios", in *Proc. 10th IEEE Int. Symp. High Performa. Distrib. Comput.*, San Francisco, CA , USA, 2001.

[9] E. Cody, R. Sharman, R. H. Rao, and S. Upadhyaya, "Security in grid computing: a review and synthesis", *Decision Supp. Sys.*, vol. 44, pp. 749–764, 2008.

[10] S. Hwang and C. Kesselman, "A flexible framework for fault tolerance in the grid", *J. Grid Comput.*, vol. 1, no. 3, pp. 251–272, 2003.

[11] J. Abawajy, "An efficient adaptive scheduling policy for high performance computing", *Future Gener. Comp. Sys.*, vol. 25, no. 3, pp. 364–370, 2009.

[12] J. Kołodziej and F. Xhafa, "Meeting security and user behaviour requirements in grid scheduling", *Simul. Modell. Practice and Theory*, vol. 19, no. 1, pp. 213–226, 2011.

[13] J. Kołodziej, F. Xhafa, and M. Bogdański, "Secure and task abortion aware ga-based hybrid metaheuristics for grid scheduling", in *PPSN XI*, Schaefer *et al.*, Eds. LNCS, vol. 6238, 2010, pp. 526–535.

[14] J. Kołodziej and F. Xhafa, "Integration of Task Abortion and Security Requirements in GA-based Meta-Heuristics for Independent Batch Grid Scheduling", *Computers and Mathematics with Applications*, DOI: 10.1016/j.camwa.2011.07.038, 2011.

[15] J. Kołodziej and F. Xhafa, "A game-theoretic and hybrid genetic meta-heuristic model for security-assured scheduling of independent jobs in computational grids", in *Proc. CISIS 2010*, USA: IEEE Press, 2010, pp. 93–100.

[16] S. Ali, H. J. Siegel, M. Maheswaran, and D. Hensgen, "Task execution time modeling for heterogeneous computing systems", in *Proc. 9th Heterogen. Comput. Worksh. HCW 2000*, Cancun, Mexico, 2000, pp. 185–199.

[17] F. Xhafa, J. Carretero, E. Alba, and B. Dorronsoro, "Tabu search algorithm for scheduling independent jobs in computational grids", *Comp. Informatics J.*, special issue on *Intelligent Computational Methods*, J. Burguillo-Rial, J. Kołodziej, and L. Nolle, Eds., vol. 28, no. 2, pp 237–249, 2009.

[18] F. Xhafa and A. Abraham, "Meta-heuristics for grid scheduling problems", in *Meta-heuristics for Scheduling in Distributed Computing Environments*, Chapter 1, Series *Studies in Computational Intelligence*, Springer, 2009, pp. 1–37.

[19] F. Xhafa, J. Carretero, and A. Abraham, "Genetic algorithm based schedulers for grid computing systems", *Int. J. Innovative Comput. Informa. Control*, vol. 5, pp. 1–19, 2007.

[20] F. Pinel, J. E. Pecero, P. Bouvry, and S. U. Khan, "A two-phase heuristic for the scheduling of independent tasks on computational grids", in *Proc. ACM/IEEE/IFIP Int. Conf. High Performa. Comput. Simulation HPCS 2011*, Istanbul, Turkey, 2011.

[21] J. Kołodziej, F. Xhafa, and Ł. Kolanko, "Hierarchic genetic scheduler of independent jobs in computational grid environment", in *Proc. 23rd Euro. Conf. Modell. Simulation ECMS 2009*, Madrid, Spain, 2009, J. Otamendi, A. Bargieła, J. L. Montes and L. M. Doncel Pedrera, Eds. Dudweiler, Germany: IEEE Press, 2009, pp. 108–115.

[22] J. Kołodziej and F. Xhafa, "Enhancing the genetic-based scheduling in computational grids by a structured hierarchical population", *Future Generation Computer Systems*, vol. 27 (2011), pp. 1035–1046, DOI: 10.1016/j.future.2011.04.011, 2011.

[23] F. Xhafa, J. Carretero, and A. Abraham, "Genetic algorithm based schedulers for grid computing systems", *Int. J. Innovative Comput. Informa. Control*, vol. 3, no. 5, pp. 1053–1071, 2007.

[24] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*. Springer, 1992.

[25] J. Kołodziej, S. U. Khan, and F. Xhafa, "Genetic algorithms for energy-aware scheduling in computational grids", in *Proc. 6th IEEE Int. Conf. P2P, Parallel, Grid, Cloud, Internet Comput. 3PGCIC*, Barcelona, Spain, 2011.

[26] D. Kliazovich, P. Bouvry, Y. Audzevich, and S. U. Khan, "GreenCloud: a packet-level simulator of energy-aware cloud computing data centers", in *Proc. 53rd IEEE Global Commun. Conf. Globecom 2010*, Miami, FL, USA, 2010.

[27] S. U. Khan and I. Ahmad, "A cooperative game theoretical technique for joint optimization of energy consumption and response time in computational grids", *IEEE Trans. Parallel and Distrib. Sys.*, vol. 20, no. 3, pp. 346–360, 2009.

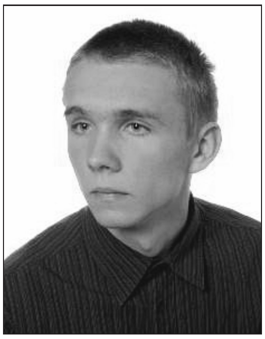
[28] S. U. Khan, "A goal programming approach for the joint optimization of energy consumption and response time in computational grids", in *Proc. 28th IEEE Int. Performa. Comput. Commun. Conf. IPCCC 2009*, Phoenix, AZ, USA, 2009, pp. 410–417.

[29] S. U. Khan and I. Ahmad, "Non-cooperative, semi-cooperative, and cooperative games-based grid resource allocation", in *Proc. 20th IEEE Int. Parallel Distrib. Process. Symp. IPDPS 2006*, Rhodes Island, Greece, 2006.

[30] G. L. Valentini *et al.*, "An overview of energy efficiency techniques in cluster computing systems", *Cluster Computing*, DOI: 10.1007/s10586-011-0171-x, 2011.

[31] L. Wang and S. U. Khan, "Review of performance metrics for green data centers: a taxonomy study", *J. Supercomput.*, pp. 1–18. DOI:10.1007/s11227-011-0704-3, 2011.

[32] S. Zeadally, S. U. Khan, and N. Chilamkurti, "Energy-efficient networking: past, present, and future", *J. Supercomput.*, pp. 1–26. DOI:10.1007/s11227-011-0632-2, 2011.



Grzegorz Gębczyński is an undergraduate student of Computer Science at the Faculty of Mechanical Engineering and Computer Science at the University of Bielsko-Biała. The topics of his engineering diploma thesis are the dynamic routing, trust and decentralized management in IPv6 networks.

E-mail: grzegorz.gebczynski@gmail.com
Faculty of Mechanical Engineering and Computer Science
University of Bielsko-Biała
Willowa st 2
43–309 Bielsk-Biała, Poland

Joanna Kołodziej graduated in Mathematics from the Jagiellonian University in Kraków in 1992, where she also obtained the Ph.D. in Computer Science in 2004. She joined the Department of Mathematics and Computer Science of the University of Bielsko-Biała as an Assistant Professor in 1997. She has served and is currently serving as PC Co-Chair, General Co-Chair and IPC member of several international conferences and workshops including PPSN 2010, ECMS 2011, CISIS 2011, 3PGCIC 2011, CISSE 2006, CEC 2008, IACS 2008-2009, ICAART 2009–2010. Dr. Kołodziej is Managing Editor of IJSSC Journal and serves as a EB member and guest editor of several peer-reviewed international journals.

E-mail: jkolodziej@ath.bielsko.pl
Department of Mathematics and Computer Science
University of Bielsko-Biała
Willowa st 2
43–309 Bielsko-Biała, Poland

Samee Ullah Khan is an Assistant Professor of Electrical and Computer Engineering at the North Dakota State University, Fargo, ND, USA. He has extensively worked on the general topic of resource allocation in autonomous heterogeneous distributed computing systems. As of recent, he has been actively conducting cuttingedge research on energy-efficient computations and communications. A total of 115 (journal: 40, conference: 54, book chapter: 12, editorial: 6, technical report: 3) publications are attributed to his name. For more information, please visit: http://sameekhan.org/.

E-mail: samee.khan@ndsu.edu
Department of Electrical and Computer Engineering
North Dakota State University
ND 58108; USA

Paper

An Experiment on Multi-Video Transmission with Multipoint Tiled Display Wall

Yasuo Ebara

Central Office for Information Infrastructure, Osaka University, Osaka, Japan

Abstract—In order to realize realistic remote communication between multipoint remote places via the Internet, displaying the appearance of remote participants by transmission of a video streaming with the large-sized display system is effective. However, the display of video streaming with sufficient quality is difficult because the specification of a commercial projector and large-sized display equipment is low-resolution. In order to these issues, we focus on the tiled display wall technology which configure effective wide-area screen system with two or more LCD panels and tried to display a high-resolution video streaming on the large-scale display environment. In this paper, we have constructed remote communication environment with tiled display wall in multipoint sites and have conducted experiment in order to study the possibility of realizing realistic remote communication with multi-video streaming. As these results, these video streaming from each site have been shown to display more high-quality than magnified view of video image by a single small camera. Moreover, we have measured the network throughput performance for each transmitted and received video streaming in this environment. From measurement results, the steady throughput performance has been gained at the case of each transmitted and received video streaming.

Keywords—multipoint remote communication, tiled display wall, tele-immersion technique, video streaming.

1. Introduction

Development of information technology systems and high-speed network technology has contributed to remote communication using the Internet. Displaying the existence of remote participants by transmission of a video streaming in real scene is effective in order to conduct smooth remote communication between multipoint sites via the Internet. Currently, various software such as video phone or chat for remote communication with PC (Personal Computer) are now provided by number of users, and the demands for these software tend to increase each year [1]. However, the window size of video image in these software applications is small and low-quality. Therefore, the video image in these is difficult to display the existence of participant with sufficient image quality in remote place.

On the other hands, many high-quality video-conference systems become popular, and these are used in wide range of research and development fields. The use of large-scale display system is effective in order to conduct high-presence remote communication with video-conference system and it is considered to use display equipment such as projectors and large-sized monitors. However, the display of video streaming with sufficient quality is difficult because the specification of a commercial projector and large-sized display equipment is low-resolution.

In order to solve these issues, we focus on the tiled display wall technology which configure effective wide-area screen system with two or more LCD panels and tried to display a high-resolution video streaming on the large-scale display environment. We have showed that an ultra high-resolution video streaming with tiled display wall is effective to display the existence of participant and ambiance in remote place as a result from current experiments [2]. However, the video image captured by a single small camera is difficult to display with realistic high-resolution on a tiled display wall, because there is a limit in the display resolution of the video image captured by a single small camera.

In this paper, we have constructed a remote communication environment between multipoint sites by using tiled display wall and have conducted experiment on multi-video transmission in order to study the possibility of realizing realistic remote communication with multi-video streaming.

2. Related Works

2.1. Tele-Immersion Technique

In recent year, there has been a growing interest in tele-immersion technique because it is expected to contribute creating super realistic communication on the Internet. Tele-immersion was originally defined as the integration of audio and video conferencing, via image based modeling, with collaborative virtual reality in the context of data mining and significant computations [3]. It enables users to share a single virtual environment from different places. In the field of practical tele-immersion, it is considered necessary that everyone who is participating in the communication is able to see any part of the image transmitted from a distant place as he/she wanted [4], [5]. In the 3-D tele-immersion system, a user wears polarized glasses and a head tracker as a view-dependent scene is rendered in real-time on a large stereoscopic display in 3-D [6], [7]. Ideally, there exists a seamless continuum between the user's experience of local and remote space within the application. The communication system using the tele-immersion environment includes Tele-Cubicles [8] and 3-D video-conference system by 3-D modeling and display techniques [9]–[11].

Recently, IPT (Immersive Projection Technology) such as the CAVE system [12] has become popular, and tele-immersive virtual environments are constructed by using IPT. In addition, when several immersive projection environments have been connected through high-speed network, the real-world oriented 3-D human image is also required as a high presence communication tool between remote places [13]. In order to realize such a demand, the video avatar technology has been studied [14]. The video avatar is a technique to represent a human image with high-presence by integrating the live video image of the human into the 3-D virtual world. However, collaborative works in IPT environment requires deflection glasses and HMD. Under such conditions, carrying out smooth remote communication between participants is usually difficult.

On the other hand, we have constructed a 3-D display environment using a merged video image obtained from the multi-viewpoint videos merging system set up in the actual space [15]. The merged video image is displayed in 3-D using the auto-stereoscopic display system which does not require tools such as deflection glasses or HMD. In addition, we have examined how well two estimators can establish eye-to-eye contact with a gazer on the auto-stereoscopic display during a face-to-face communication and proved that two estimators could realize eye-to-eye contact at each direction [16]. However, we have considered that the 3-D environment is difficult to realize high-presence remote communication by the restriction of space which can move in collaborative work since the autostereoscopic display indicates an unfocused 3-D video image by participant's positions. Therefore, we consider that it is important to develop the technique to display more realistic information with simple ways for participants in order to realize high-presence remote communication.

We use tiled display wall to display high-quality video streaming which realize remote communication with highly realistic sensation. In this research, we conduct the experiment to display high-resolution video streaming on tiled display wall in multipoint sites.

2.2. Tiled Display Wall Technology

Tiled display wall is a technology to display a high-resolution image on the large-scale display with two or more LCD panels in order to construct effective wide-area screen system [17], [18]. Much research has been developed tiled display wall and remote displays by using distribute rendering technique. For example, WireGL provides the familiar OpenGL API to each node in a cluster, virtualizing multiple graphics accelerators into a sort-first parallel renderer with a parallel interface [19]. It can drive a variety of output devices, from stand-alone displays to tiled display walls. However, it has poor data scalability due to its single source limitation.

On the other hand, Chromium have designed and built a system that provides a generic mechanism for manipulating streams of graphics API commands [20]. It can be used as the underlying mechanism for any cluster-graphics algorithm by having the algorithm use OpenGL to move geometry and imagery across a network as required. In addition, Chromium's DMX extension allows execution of multiple applications and window control. Moreover, CGLX (Cross-Platform Cluster Graphic Library) is a flexible, transparent OpenGL-based graphics framework for distributed high performance visualization systems in a master-slave [21]. The framework was developed to enable OpenGL programs to be executed on visualization clusters such as a high resolution tiled display wall and to maximize the achievable performance and resolution for OpenGL-based applications on such systems. However, we assume the application of tiled display wall in remote communication environment via WAN. Therefore, we consider the use of these middleware is unsuitableness.

2.3. SAGE

In this research, we apply SAGE (Scalable Adaptive Graphics Environment) [22], [23] developed by Electronic Visualization Laboratory at the University of Illinois to delivery streaming pixel data with virtual high-resolution frame buffer number of graphical sources for tiled display wall. SAGE is a graphics streaming architecture for supporting collaborative scientific visualization environments with potentially hundreds of megapixels of contiguous display resolution. The network-centered architecture of SAGE allows collaborators to simultaneously run various applications on local or remote clusters, and share them by streaming the pixels of each application over high-speed networks to large-scale tiled display wall.

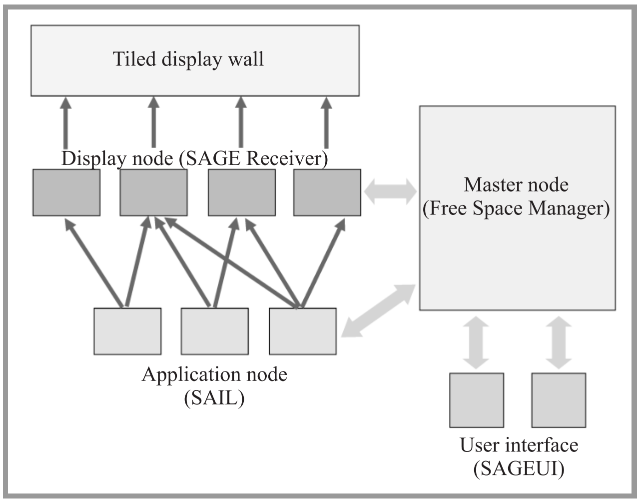


Fig. 1. SAGE components [23].

SAGE consists of the Free Space Manager, SAGE Application Interface Library (SAIL), SAGE Receiver, and User Interface (SAGEUI) as shown in Fig. 1. SAGE over WAN from SAGEUI and controls pixel streams between SAIL and the SAGE Receivers. SAIL captures application's output pixels and streams them to appropriate SAGE Receivers. A SAGE Receiver can receive multiple pixels streams from different applications and displays streamed pixels on multiple displays. A SAGE Receiver can handle multiple displays. A SAGEUI sends user commands to control the Free Space Manager and receives messages that inform users of current status of SAGE.

2.4. Multi-Video Streaming on Tiled Display Wall

We have constructed tiled display wall environment consist of 1 master node, 2 display nodes and 4 LCD panels. An example of the tiled display wall is LCDs are located at 2×2 arrays as shown in Fig. 2. Master node and all display nodes are connected by gigabit Ethernet network, and 2 LCDs are connected to 1 display node with DVI cables. In our environment, we apply SAGE as middleware of tiled display wall to delivery streaming pixel data with virtual high-resolution frame buffer number of graphical sources for tiled display wall environment.

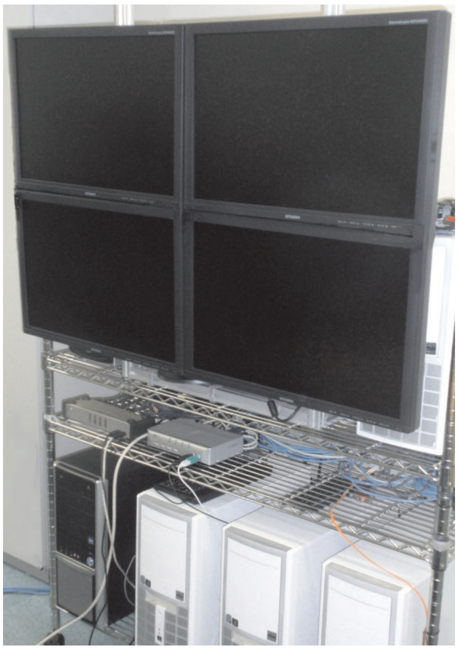


Fig. 2. Example of tiled display wall environment.

We have implemented an application of high-resolution real video streaming with a small camera by adding API code of SAIL (SAGE Application Interface Library) in source program of application. In the application, pixel information obtained from an small camera is rendered as video image by glDrawPixels on tiled display wall. The video image is captured by a small camera is transmitted from application node to master node in remote place, its video streaming is displayed on tiled display wall.

We have showed that an ultra high-resolution video streaming with a tiled display wall is effective to display the existence of participant and ambiance in remote place as a result from current experiments [2]. However, the video image captured by a single small camera is difficult to display with realistic high-resolution on a tiled display wall, because there is a limit in the display resolution of the video image captured by a single small camera. In addition, the magnified view of original video image on tiled display wall causes to degrade the quality of video image.

We have tried the construction of the environment to display realistic high-resolution video streaming on tiled display wall [24]. In this environment, each video image data which captured by multiple cameras is transmitted to display nodes of tiled display wall in remote sites from multiple application nodes via LAN. In addition, a panorama video with high-resolution which is generated by compositing these transmission video images are displayed on tiled display wall.

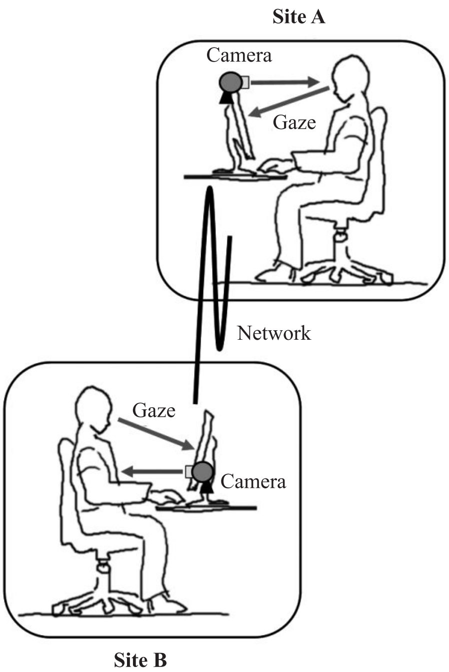


Fig. 3. Issue of eye-to-eye contact in video conference.

On the other hand, since a capturing camera which used in usual video-conference system is located at the top or both sides of display. The participant's direction of eyes often stare slides at different direction for participant of remote place. In these camera setting method, participant's direction of eyes is pointed at remote participant displayed on screen, although the setting way enable to point the camera at local participant as shown in Fig. 3. Therefore, it is difficult to have a conversation by face to face, and cause to interfere with smooth communication between participants of remote places.

In contrast, tiled display wall enables to locate freely the position of camera by using frame of LCD panel. We have conducted experiments on the possibility of eye-to-eye contact in remote communication with tiled display wall, and examined the effective location method of camera in order to solve these issues [25]. From these experiments, we have shown experimentally that each participant enables to turn direction of eyes to a camera in a natural way become important factor for realization of eye-to-eye contact in remote communication. In using these findings, we have located 2 sets of small cameras at the horizontal position on LCDs of tiled display wall to establish eye-to-eye contact naturally as shown in Fig. 4. 2 sets of Pointgray Firefly (resolution: 640×480 pixels, frame rate: 30 fps) as captured camera have been used in this environment.

Fig. 4. Cameras setting on tiled display wall.

3. Experiment on Multi-Video Transmission in Multipoint Tiled Display Wall

3.1. Experimental Environment

In this research, we conduct fundamental experiment on the display of high-resolution video streaming on tiled display wall in each site by transmitting multi-video images which captured by multiple small cameras connected to each application node in multipoint remote sites. The network environment in this experiment is shown in Fig. 5. In this environment, 3 tiled display walls have been located in 2 network subnets. 2 tiled display walls (site A and B) in subnet 1 and 1 tiled display wall (site C) in subnet 2 have been located. Subnets 1 and 2 have been connected by L3 switch (gigabit Ethernet) in campus LAN. In each site, master node, display node, and 2 application nodes are located and connected by gigabit Ethernet with HUB (L2 switch). Table 1 shows hardware configuration of tiled display wall in each site.

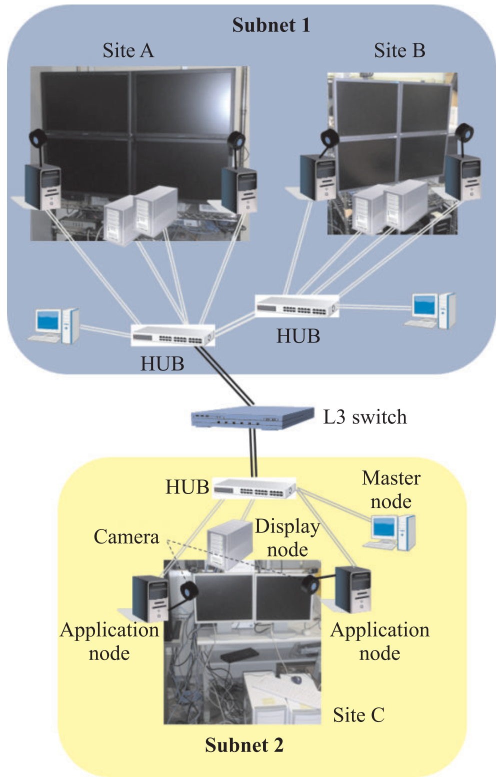


Fig. 5. Network environment between tiled display walls of 3 sites.

Figure 6 shows system configuration in this experiment. The multi-video streaming which captured by 2 cameras located at the frame of LCDs on tiled display wall in sites A, B and C are transmitted to display nodes of tiled display wall in remote sites via network from 2 application nodes of each site. In addition, a high-resolution video streaming of each site is generated by compositing these transmission video images are displayed on tiled display wall.

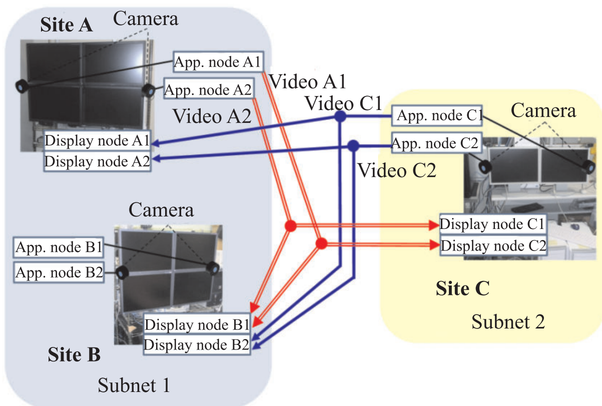


Fig. 6. System configuration in this experiment.

We have applied SAGE Bridge to realize transmission and displaying same video streaming on tiled display wall in multi-site. SAGE Bridge receives pixel streams from appli-

Table 1
Hardware configuration of each tiled display wall

	Site A	Site B	Site C
Master node			
CPU	Intel(R) Core i7 2.67 GHz	Intel(R) Core i7 2.80 GHz	Intel(R) Xeon (Dual Core) 1.6 GHz (×2)
Memory	8,192 GB	8,912 GB	8,192 GB
VGA	NVIDIA GLADIAC 786	NVIDIA GTS 250G	NVIDIA GLADIAC 786
OS	OpenSUSE 10.3	OpenSUSE 10.3	OpenSUSE 10.3
Display node			
Number	2	2	1
CPU	Intel(R) Core i7 2.93 GHz	Intel(R) Core i7 2.93 GHz	Intel(R) Core i7 2.80 GHz
Memory	8,192 GB	4,096 GB	4,096 GB
VGA	NVIDIA GTS 250G	NVIDIA GTS 250G	NVIDIA Quadro FX 570
OS	OpenSUSE 10.3	OpenSUSE 10.3	OpenSUSE 10.3
Application node			
CPU	Intel(R) Core i7 2.80 GHz	Intel(R) Core i7 2.80 GHz	Intel(R) Core i7 2.80 GHz
Memory	4,096 GB	4,096 GB	4,096 GB
VGA	NVIDIA Quadro FX 570	NVIDIA Quadro FX 570	NVIDIA Quadro FX 570
OS	OpenSUSE 10.3	OpenSUSE 10.3	OpenSUSE 10.3
LCD panel			
Number	4	4	2
Size	24.1 inch	17 inch	17 inch
Resolution	1,920×1,200	1,280×1,024	1,280×1,024

cations and distributes to multiple SAGE sessions and is supposed to be executed on high-performance PCs bridging rendering clusters and display clusters [26]. In this experiment, all application nodes are used as SAGE Bridge node, and are transmitted multi-video streaming to tiled display walls in multiple remote sites via LAN.

3.2. Experimental Results

Multi-video streaming of each site which generated by compositing transmission video captured by 2 sets of small cameras in sites A and C are displayed on tiled display wall in site B as shown in Fig. 7. From these results, we consider that these video streaming from each site have

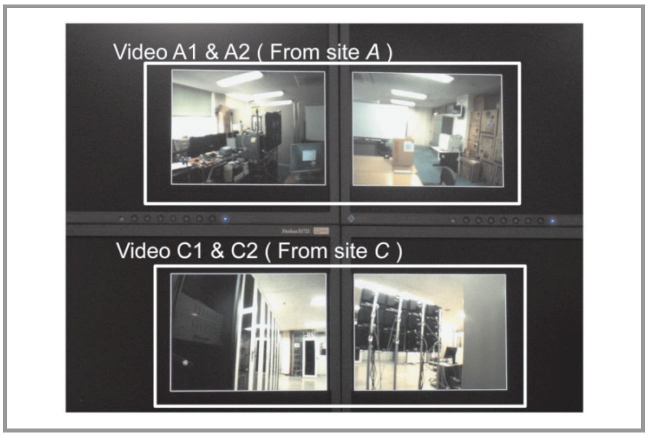


Fig. 7. Display result of multi-transmission video image on tiled display wall in site B.

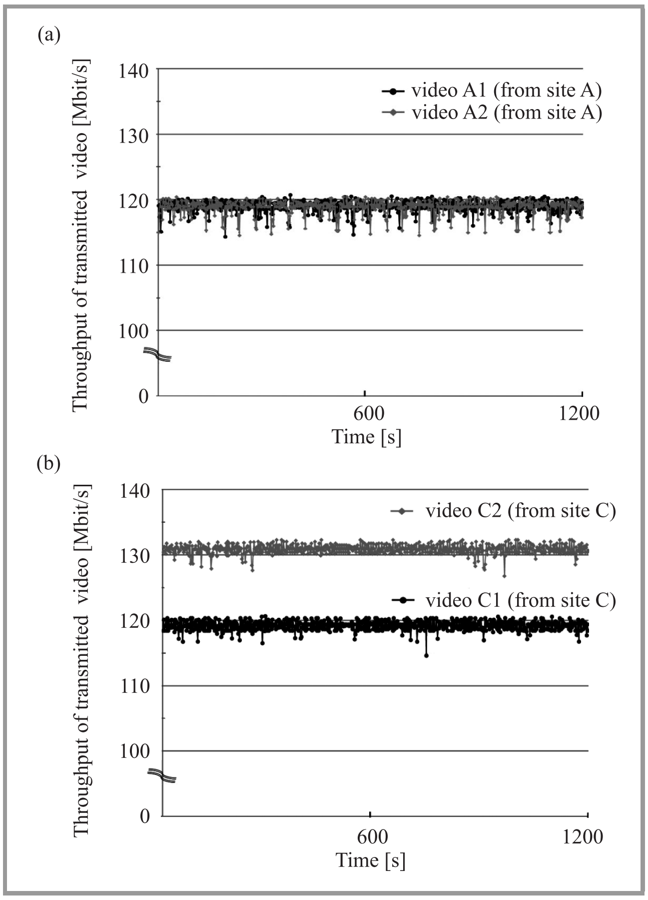


Fig. 8. Network throughput of each transmitted video: (a) A1 and A2; (b) C1 and C2.

been displayed more high-quality than magnified view of video image by a single small camera. As future works, we will try to display more high-quality video streaming of remote place over a wide range on tiled display wall in remote sites by increasing number of captured camera. In addition, we have not implemented the compositing for the boundary parts of each video by automatic processing, and have depended on the detail manual setting of each camera in this environment. We will need to study the technique which can composite the boundary parts of each image by the automated processing that doesn't depend on the position of the camera in the future works.

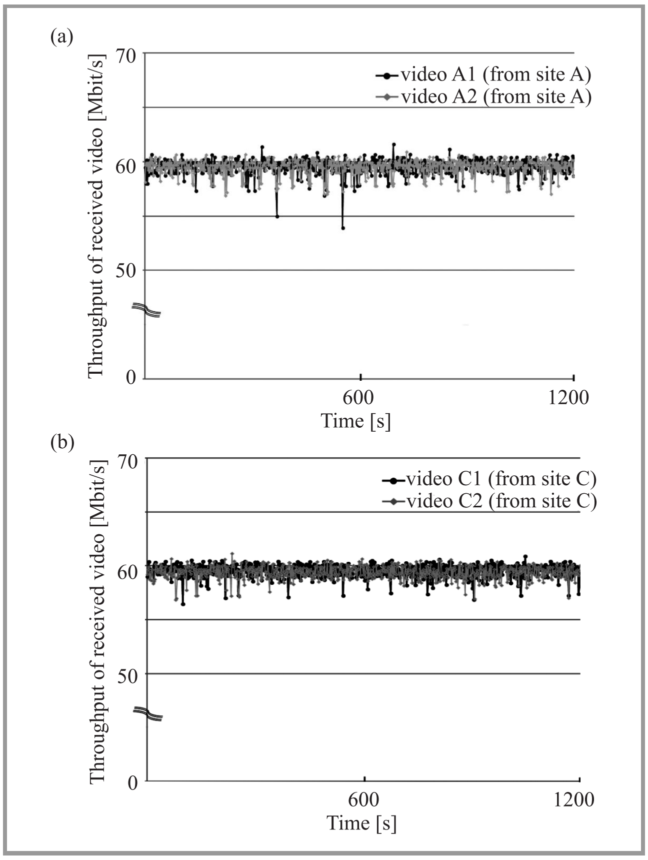


Fig. 9. Network throughput of each received video in tiled display wall *B*: (a) A1 and A2; (b) C1 and C2.

Then, we have measured the network throughput performance for each transmitted and received video streaming in this environment. The network throughput of transmitted video streaming from site A and C, and received video in site B is measured in this experiment. The measured results are shown in Figs. 8 and 9. The network bandwidth of transmitted and received video streaming about for 20 min are plotted the average value for measurement results of every 2 s. From measurement results, transmitted video streaming from each site have been observed about 120 Mbit/s. These results are considered that double network throughput has been generated to transmit same video streaming to tiled display walls in 2 sites. In addition, the steady throughput performance of 60 Mbit/s on average has been gained at the case of each received video streaming, and the values is considered a full performance of a camera.

4. Conclusion

In this paper, we have constructed remote communication environment with tiled display wall in multipoint sites and have conducted experiment in order to study the possibility of realizing realistic remote communication with multi-video streaming. From these results, these video streaming from each site have been shown to display more high-quality than expanded video image by single small camera. Moreover, we have measured the network throughput performance for each transmitted and received video streaming in this environment. From measurement results, the steady throughput performance has been gained at the case of each transmitted and received video streaming.

In remote communication between multipoint sites with video-conference system, the issue by sense of discomfort in communication to remote participants which wants to speak is caused. As a result, it becomes the factor of producing the misunderstanding is caused among participants in remote places. We will examine the possibility of high-presence remote communication to realize eye-to-eye contact with each other by using tiled display wall between multipoint sites. In addition, we will study that the possibility of new technique for automatic switching of transmission video in remote communication by the estimation of face direction from participant's video to realize effective remote communication between some participants distributed in multipoint remote sites.

References

[1] G. Kong, J. Stanton, S. Newhouse, and J. Darlington, "Collaborative visualization over the Access Grid using the ICENI Grid Middleware", in *Proc. UK e-Science All Hands Meeting*, Nottingham, UK, 2003, pp. 393–396.

[2] Y. Ebara, N. Kukimoto, J. Leigh, and K. Koyamada, "Tele-immersive collaboration using high-resolution video in tiled displays environment", in *Proc. IEEE 21st Int. Conf. Adv. Inform. Netw. Appl. AINA 2007*, Niagara Falls, Canada, 2007, vol. 2, pp. 953–958.

[3] J. Leigh, T. A. DeFanti, A. E. Johnson, M. D. Brown, and D. J. Sandin, "Global tele-immersion: better than being there", in *Proc. 7th Int. Conf. Artif. Real. Tele-Existence*, Esbjerg, Denmark, 1997, pp. 10–17.

[4] A. Sadagic, H. Towles, L. Holden, K. Daniilidis, and B. Zeleznik, "Tele-immersion Portal: towards an ultimate synthesis of computer graphics and computer vision systems", in *Proc. 4th Ann. Int. Worksh. Presence*, Philadelphia, Pensylvania, USA, 2001.

[5] K. S. Park, A. Kapoor, C. Scharver, J. Leigh, "Exploiting multiple perspectives in tele-immersion", in *Proc. Immersive Projection Technol. Worksh. IPT 2000*, Ames, Iowa, USA, 2000.

[6] N. Kelshikar, X. Zabulis, J. Mulligan, K. Daniilidis, V. Sawant, S. Sinha, T. Sparks, S. Larsen, H. Towles, K. Mayer-Patel, H. Fuchs, J. Urbanic, K. Benninger, R. Reddy, and G. Huntoon, "Real-time terascale impementation of tele-immersion", in *Proc. Int. Conf. Comput. Sci.*, Saint Petersburg, Russia, 2003.

[7] H. Towles, S. Kum, T. Sparks, S. Sinha, S. Larsen, and N. Beddes, "Transport and rendering challenges of multi-stream 3D tele-immersion data", in *Proc. NSF Lake Tahoe Worksh. Collabor. Virtual Real. Visual. CVRV 2003*, Tahoe City, CA, USA, 2003.

[8] S. J. Gibbs, C. Arapis, and C. J. Breiteneder, "TELEPORT-towards immersive copresence", *Multimedia Systems*, no. 7, pp. 214–221, Springer, 1999.

[9] W. Chen, H. Towles, L. Nyland, G. Welch, and H. Fuchs, "Toward a compelling sensation of telepresence: demonstrating a portal to a distant (static) office", in *Proc. IEEE Visualization 2000*, Salt Lake City, Utah, USA, 2000, pp. 327–333.

[10] P. Kauff and O. Schreer, "An immersive 3D video-conferencing system using shard virtual team user environments", in *Proc. 4th Int. Conf. Collab. Virtual Environ. CVE'02)*, Bonn, Germany, 2002, pp. 105–112.

[11] W. Matusik and H. Pfister, "3D TV: a scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes", *ACM Trans. Graphics*, vol. 23, no. 3, pp. 814–824, 2004.

[12] C. Cruz-Neira, D. J. Sandin, and T. A. DeFanti, "Surround-screen projection-based virtual reality: the design and implementation of the CAVE", in *Proc. 20st Ann. Conf. Comp. Graphics Interact. Techniq. SIGGRAPH 1993*, Anaheim, California, USA, 1993, pp. 135–142.

[13] J. Leigh, T. A. DeFanti, A. E. Johnson, M. D. Brown, and D. J. Sandin, "Global Tele-Immersion: Better than Being There", in *Proc. Int. Conf. Artif. Real. Tele-existence ICAT'97*, Tokyo, Japan, 1997, pp. 10–17.

[14] T. Ogi and M. Sakai, "Communication in the networked immersive environment", in *Proc. ASIAGRAPH 2006*, Yokohama, Japan, 2006, pp. 67–72.

[15] T. Ando, K. Mashitani, M. Higashino, H. Kanayama, H. Murata, Y. Funazou, N. Sakamoto, H. Hazama, Y. Ebara, and K. Koyamada, "Multiview image integration system for glassless 3D display", *SPIE*, vol. 5664, pp. 158–166, 2005.

[16] Y. Ebara, Te. Nabuchi, N. Sakamoto, and K. Koyamada, "Study on eye-to-eye contact by multi-viewpoint videos merging system for tele-immersive environment", in *Proc. IEEE Int. Worksh. Netw.-based Virt. Real. Tele-existence INVITE 2006*, 2006, pp. 647–651.

[17] C. Li, H. Lin, and J. Shi, "A survey of multi-projector tiled display wall construction", in *Proc. 3rd Int. Conf. Image Graphics ICIG'04*, Hong Kong, China, 2004, pp. 452–455.

[18] T. Ni, G. S. Schmidt, O. G. Staadt, M. A. Livingston, R. Ball, and R. May, "A survey of large high-resolution display technologies, techniques, and applications" in *Proc. IEEE Conf. Virtual Reality VR 2006*, Alexandria, Virginia, USA, 2006, pp. 223–236.

[19] G. Humphreys, I. Buck, M. Eldridge, and P. Hanrahan, "Distributed Rendering for Scalable Displays", in *Proc. IEEE Supercomputing Conf. SC 2000*, 2000, Dallas, Texas, USA.

[20] G. Humphreys, M. Houston, Y. Ng, R. Frank, S. Ahern, P. Kirchner, and J. T. Klosowski, "Chromium: a stream-processing framework for interactive rendering on clusters", in *Proc. 29th Ann. Conf. Comp. Graphics Interactive Techn. SIGGRAPH'02*, San Antonio, Texas, USA, 2002, pp. 693–702.

[21] K. Doerr and F. Kuester, "CGLX: a scalable, high-performance visualization framework for networked display environments", *IEEE Trans. Visualization and Comp. Graphics*, vol. 17, no. 3, pp. 320–332, 2011.

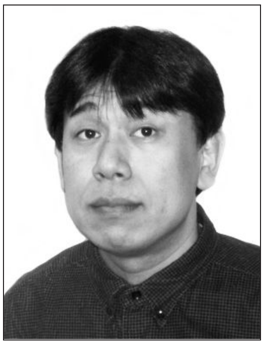
[22] L. Renambot, A. Rao, R. Singh, B. Jeong, N. Krishnaprasad, V. Vishwanath, V. Chandrasekhar, N. Schwarz, A. Spale, C. Zhang, G. Goldman, J. Leigh, and A. Johnson, "SAGE: the scalable adaptive graphics environment", in *Proc. Worksh. Adv. Collaborative Envir. WACE 2004*, Nice, France, 2004.

[23] J. Byungil, L. Renambot, R. Jagodic, R. Singh, J. Aguilera, A. Johnson, and J. Leigh, "High-performance dynamic graphics streaming for scalable adaptive graphics environment", in *Proc. Supercompuing SC 2006*, Tampa, Florida, USA, 2006, p. 24.

[24] Y. Ebara and Y. Shibata, "Study on tele-immersive communication with multi-video streaming on tiled display wall", in *Proc. IEEE 13th Int. Conf. Netw.-Based Inform. Sys. NBiS 2010*, Takayama, Gifu, Japan, 2010, pp. 439–444.

[25] Y. Ebara, N. Kukimoto, and K. Koyamada, "Evaluation experiment on eye-to-eye contact in remote communication with tiled displays environments", in *Proc. IEEE 22nd Int. Conf. Adv. Information Netwo. Appli. -Workshop- AINA 2008*, Okinawa, Japan, 2008, pp. 1017–1022.

[26] "SAGE: Scalable Adaptive Graphics Environment" [Online]. Available: http://www.sagecommons.org/



Yasuo Ebara received his M.Sc. and Ph.D. degrees in the Graduate School of Information Science, Tohoku University, Sendai, Japan, in 1997 and 2000, respectively. Since 2000 he has been with Kyoto University, Japan, where he was an Assistant Professor. Since 2008 he is Associate Professor at Osaka University, Japan. His research topic focuses on tele-immersion technique, network collaboration, and application of information system.
E-mail: eba@cmc.osaka-u.ac.jp
Central Office for Information Infrastructure
Osaka University
Ibaraki, Osaka, 567-0047, Japan

Paper

Video Transmission Using Network Coding

Francisco de Asís López-Fuentes and Cesar Cabrera-Medina

Department of Information Technology, Autonomous Metropolitan University, México

Abstract—Network coding is a new technique in the field of information theory and coding theory. This emerging technique offers great benefits in the communication networks such as increased throughput and energy savings. In this paper, we evaluate network coding for video transmission scenarios. In these scenarios, the source nodes encode the video packets, while the intermediate nodes implement network coding before forwarding the encoded packets to the end nodes. Finally, the end nodes decode the received packets in order to recovery the original video. H.264/MPEG-4 AVC is used as the video compression standard in this work. We have used the network simulator (NS-2) for our simulations. Our results show that network coding improves the system throughput, reduces the packet loss and improves the video quality in the end nodes.

Keywords—communication networks, information theory, multicast, network coding video streaming.

1. Introduction

Recently, a large amount of traffic is generated on the Internet. Mainly, this traffic is generated by multimedia applications such as video streaming, video download or videoconferencing. Additionally, many Internet users use this infrastructure to share and distribute data, music, videos and photos. Many of these applications involve systems with multiple senders. Multiple sources help to alleviate the unpredictability congestion in the Internet and it has been proposed as an alternative to edge streaming to provide smooth video delivery [1], [2].

The great traffic generated by all these applications, has created problems on the communications networks, such as delays and a bad quality in the received files. Due this situation, new techniques need to be introduced to deal with these problems. For example, video multicast is a network service through which a video is streamed to a group of interested receivers. Video streaming typically requires high data rate, low-latency, or high throughput in order to offer video quality to the viewers. However, traditional transmission techniques, based on a method known as store-and-forward, are not capable of supporting such applications. Store-and-forward is a technique in which data packets received from an input link of an intermediate node are stored and a copy is forwarded to the next node via an output link [3]. The intermediate node verifies the integrity of the message before forwarding it. This traditional transmission technique introduces a delay at the input to each link along the packet's route, which can affect the video quality to the viewers. Our proposed work aims to give a solution to some problem in this area.

Network coding was introduced by Ahlswede *et al*. [4] as a new technique for the diffusion of the information in the field of information theory. This new technique allows that the intermediates nodes encode the received packets on the intermediate nodes, for immediately forwarding the encoded packets to the end nodes. This fact has generated a considerable debate between the traditional technique of storing and forwarding against the network coding technique. Tuninety *et al*. [5] study both schemes and they found that network coding improves the throughput, scalability and robustness of a system. Network coding has impacted in the following areas: Mesh, VANETs, MANETs, ad-hoc, sensor networks, security issues and content distribution. Several benefits in the communication networks by using network coding have been reported in the literature.

This paper evaluates the transmission of video streaming for multicast schemes using network coding. Our network coding implementation realized in the intermediate nodes is inspired by the model introduced by Chou *et al*. in [6]. Our simulations use video sequences in the CIF (Common Intermediate Format) video format. CIF defines a video sequence with a resolution of 352×288. These video sequences are compressed using the standard H.264/MPEG-4AVC. The communication between all nodes is established via the User Datagram Protocol (UDP). We compare the performance of a multi-source system using network coding versus a traditional multi-source system in terms of packets loss, throughput and video quality. To this end, we evaluate video transmission sessions with network coding for two strategies. In the first strategy, video sequences with different size are transmitted by each source, while in the second strategy all video sequences transmitted by all sources have the same size. The obtained results show that network coding improves the system throughput, decreases the packet loss and improve the video quality in the end nodes.

The rest of this paper is organized as follows. Section 2 presents the main idea of network coding. Then, we give an overview about the related work in Section 3. In Section 4 we describe our model based on network coding. The performance of our proposed model is evaluated in Section 5. This paper concludes in Section 6.

In this paper we present an extended version of our other work [7]. Specifically, the following new material has been

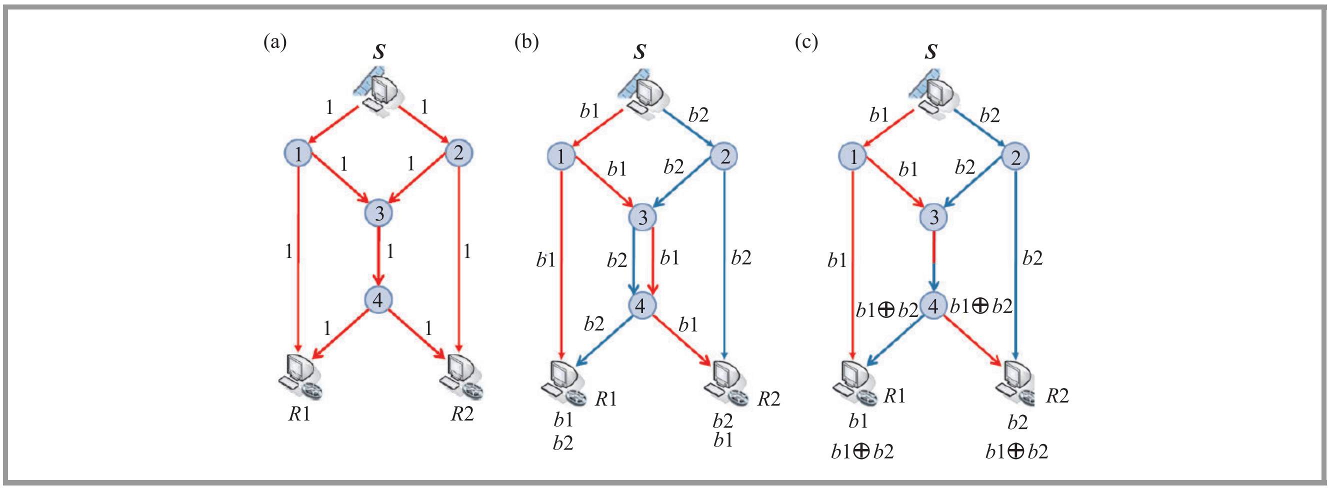


Fig. 1. Communications network: (a) capacity of the edges, (b) traditional approach and (c) approach with network coding.

added. First, an exposition about the features and benefits of network coding for video streaming are introduced in Section 3. Second, specific video frames pictures comparing the different reconstruction quality by using network coding are presented in Section 5. Percentages of packet loss for experiments 1 and 2 have been added too. Finally, a third experiment to evaluate the buffer usage during a video transmission with network coding is added in Section 6.

2. Network Coding Overview

Network coding is a new technique in the field of information theory, proposed by Ahlswede *et al.* in [4]. In this paper, a new problem on networks communications related to information flow is discussed. The authors reveal that it is in general not optimal to regard the information to be multicast as a fluid, rather, by employing coding at the intermediates nodes on the networks, to increase the flow without exceeding the channel capacity. For this study, the network is represented as a directed graph $G = (V,E)$, where the network nodes are represented by V, and the edges E represent the communication channels. The channel capacity is of one data unit per unit time. The source node is a node without any incoming edge.

Network coding is inspired by the Max-Flow Min-Cut theorem which states that [5]: “The maximum amount of flow through the source to the destination equals the minimum capacity required to remove from the network flows that cannot pass from source to destination”.

In a single-source multicast session, source node $s \in S$ transmits information at rate R to all receivers $t \in T$, and the maximum multicast information rate in this scenario can be achieved only by allowing coding at intermediate nodes [4]. This optimal multicast rate can be given by finding the capacity through the Max-Flow Min-Cut theorem above described, which relates the maximum information flow through a network to the minimum cut capacity.

To understand better network coding, Fig. 1 presents a scenario for a butterfly network with a source node and two receiver nodes. Figure 1a shows the capacity of each edge. We can observe that the values of the maximum flow of S to any receiver, either $R1$ or $R2$ are equal to two. Therefore, in Fig. 1b, the source S can send two bits $b1$ and $b2$ to $R1$ and $R2$ simultaneously. In this scheme, each intermediate node only replicates and sends out the bit(s) received from upstream. On the other hand, Fig. 1c shows the same network configuration, only that now network coding is implemented. Here, operator $\oplus$ denotes the sum modulo 2. Thus, the receiver $R1$ can recover the two bits $b1$ and $b2$.

Only that $b2$ must be retrieved from $b1 \oplus b2$. Similarly, $R2$ can recover the two bits. In this example, network coding is applied in the node 3. Another important point to note is that the rate multicast increases, because for traditional transmission is 1 bit/time unit, while applying network coding the rate increases to 2 bits/time unit.

The most common benefits by using network coding in a communication network are [4]:

- bandwidth saving,
- improved system throughput,
- reduced delays.

Video streaming systems are distinct from general data delivery systems, in which a client has to download the entire file before using it. In a video streaming session, the receiver can already consume the video while downloading. However, packets with large delay can be perceived by the video application as packet losses and the transmission of these useless packets waste network resources. Therefore, this kind of multimedia application requires high data rate, low-latency and low packet lost rate, which represent a significant challenge for the design of future network architectures. Thus, we believe that network coding can help to improve the performance of the video streaming systems.

3. Related Work

Several research works showing the benefits of network coding have been published in the literature. Mainly, network coding has impacted in the following areas: Mesh, VANET's, MANET's, ad-hoc, sensor networks, security issues, video streaming and content distribution. In the following, we briefly describe some of these applications.

Gkantsidis *et al*. [8] introduced a novel scheme for large scale content distribution using network coding. Based on these findings, Gkantsidis *et al*. report that the system increases the throughput between 20% and 30% compared with the coding applied only on the server. Also each block is unique and is more likely to be useful for other nodes, which facilitates the exchange of these. The authors argue that the distribution rate of 2–3 times improvement in comparison with systems without applying network coding. An application of network coding for wireless networks is introduced by Sundararajan *et al*. [9]. The authors argue that network coding has a positive impact on the TCP/IP (Transmission Control Protocol/Internet Protocol). To show this, a new layer is introduced between the transport layer (TCP) and network layer (IP). Here, the congestion control is not modified. The results show that the throughput has a slight increase when network coding is applied to TCP. Another important point is that traditional TCP fails when packet loss is increased, while the TCP with network coding shows robustness during packet loss situation.

SenseCode [10] is a work in sensor networks, which involves network coding techniques. In this work, the authors are interested in the following parameters: reliability and transmission energy. Reliability refers to the useful information that reaches the receiver, while the transmission energy refers to the amount of energy expended during the transmission of packets from each node. To realize the measurements of these parameters, SenseCode is evaluated and compared with the CTP protocol (Collection Tree Protocol) [11], on the platform TOSSIM [12]. The obtained results show that network coding provides a measure of balance between energy efficiency and reliability. Finally, a scheme where network coding is used to design a secure architecture for video transmission over wireless networks is proposed by Lima *et al*. [13]. The authors evaluate their proposed model using NS-2 [14], for a wireless scenario with losses. The results indicated that the encrypted packets using network coding techniques have a lower size that packets encrypted by the traditional method. Additionally, the probability of packet loss is lower, which allows a better video quality.

Recently, network coding has found a fertile application area in the video streaming systems. For example, Nguyen *et al*. [15] studied the application of network coding in wireless networks for video broadcasting. They propose an optimal scheme to generate erasure codes by using NC to retransmit lost packets. One of the most attractive features of network coding is its ability to achieve the optimal multicast rate [4]. In this paper, we extend these features to a multi-source scenario for the transmission of video.

4. Our Model

The main benefits of using network coding can be found in multicast communications where there are link capacities in network, the issue is only to compute the maximum multicast throughput possible for communication between a source node and a set of sink nodes [16]. In this section we propose a multi-source scenario based on a many-to-many distribution scheme in order to study the benefits offered by network coding during a video transmission. Our reference scheme is shown in Fig. 2. In this distribution scheme there are three source nodes, three receiver nodes and six intermediate nodes. We use this distribution scenario to deploy our multicast schemes. Bottleneck problems are most likely to be present in this scenario. Thus, we can evaluate the systems robustness by applying network coding techniques. Two multicast schemes are deployed over the many-to-many distribution scheme:

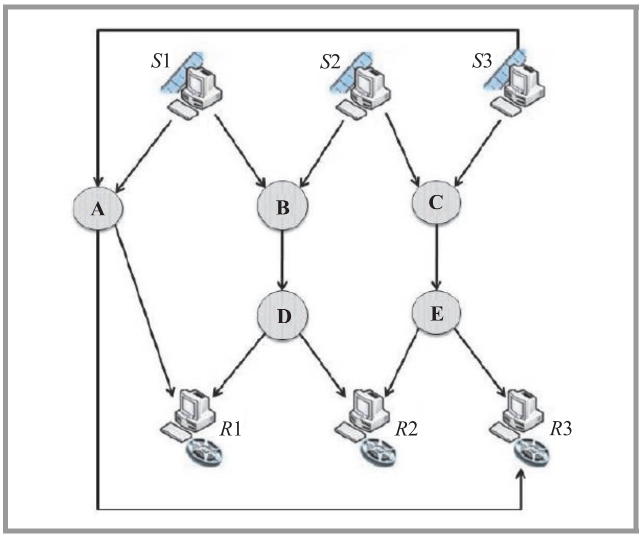


Fig. 2. Our many-to-many distribution scheme.

- Without network coding. In this first strategy, the intermediates nodes act as in the traditional networks, that is to say, they take to store in its buffer packets from entering edges, for later forwarding these packets to the out edges. This buffer management policy in each node is called FIFO policy (First-In-First-Out).
- With network coding. In this scheme, network coding techniques are applied in the intermediate nodes of the network. Our proposed scheme is inspired by a work proposed in [6]. In our scheme (see Fig. 2), the intermediate nodes A, B and C perform a linear combination of packets, which are stored in its buffers. On the other hand, nodes D, and E only are responsible for storing and forwarding the received

packets. Once the receiver nodes receive the encoded packets encoded. They are responsible for decoding and reconstructing the original video.

4.1. Network Coding Scheme

Network coding has generated several proposals of encoding schemes. In the following, we describe the network coding technique used for a many-to-many distribution scenario. Our network coding scheme follows the concept introduced in [6], where a centralized knowledge of the networks is not required. This novel approach introduces a new package format, which consists of a global coding vector and the payload. The payload is divided into fields of size 2^8 or 2^{16}, that is to say, each symbol is 8 or 16 bits. This scheme assumes an acyclic graph (V,E), where each edge is of unit capacity and also has a source $s \in V$ and set of receivers $T \subseteq E$. Each edge $e \in E$ emanating from a node v carries a symbol $y(e)$ that is a linear combination of the symbols $y(e')$ on the edges e' that are incoming edges to node v.
Thus,

$$y(e) = \sum_{out(e')=v} m_e(e')y(e') \tag{1}$$

and

$$m(e) = [m_e(e')]_{e':out(e')=v} \tag{2}$$

represents the encoding function of the node v on the edges e. This obtained vector $m(e)$ is known as vector-linear coding.
Now, if v is the source node S, then the notation is maintained and the artificial edges $e'_1,\ldots,e'_h$ are introduced. These edges are considered as the input edges to node S, which are the symbols of the source $y(e'_i) = x$, where $i = 1,\ldots,h$. Thus, $y(e)$ on any edges $e \in E$ is a linear combination defined as

$$y(e) = \sum_{i=1,h} g(e)x_i \tag{3}$$

of the symbols of the source, where the coefficients $g(e) = [g_1(e),...,gh(e)]$ can be recursively determinate by

$$g(e) = \sum_{e':out(e')=v} m_e(e')g(e')\,. \tag{4}$$

Thus, the vector $g(e)$ is called the global encoding vector along the edges e. Each node receives packets that are linear combinations of the source packages and stores them in an array.
In the scheme proposed by Chou *et al.* [6], each node sends packets obtained as a random linear combination of packets stored in its buffer. So within each package is a global coding vector $g(e)$ of dimension h. Any receiver can then recover the source vectors $x_1,\ldots,x_h$ using Gaussian elimination on the vectors in its h received packets. Packets arrive at the receiving nodes on the incoming edges, and they may find related packages, that is to say, with the same source vector $x_1,\ldots,x_h$. This type of packages is known as packets of the same generation, where h represents the size of the generation. All packages of the same generation are labeled with the same generation number. The policy used by Chou *et al.* is "empty the current generation before it reaches the next generation". This policy indicates that we must attend the packets of a same generation that existed before to attend to other packets of a different generation.

5. Evaluation

Our simulation is implemented on the network simulator NS-2, which is a discrete event simulator targeted at networking research [14]. Several simulations of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks can be supported by this simulator. NS-2 is divided into two hierarchies, the compilation written in C++ and the interpreter in OTcl. Both hierarchies are closely linked. Our simulation scenario is shown in Fig. 3. Video sequences used in our simulations are represented in each source node. All receiver nodes expect to receiver all video sequences.

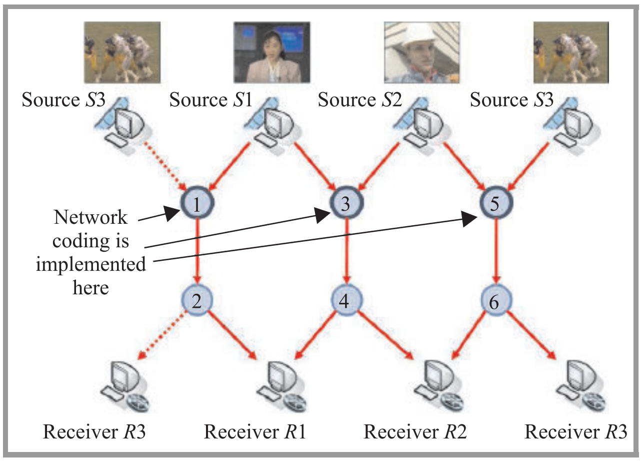


Fig. 3. Network coding simulation scenario with three sources and three receivers

We use the following simulation parameters. The edges delay is 10 ms, the buffers size in each node is 819 kbit, and UDP is the transport protocol. A dense mode version is used in the routing protocol. This protocol works as follows. Initially, the protocol assumes that all nodes in the network are receivers of multicast traffic and therefore packets are transmitted across the network. Each node that receives such traffic, but not requires it, sends a prune message via the interface where it is received. This process is known as tree pruning and runs every few minutes.
To simulate the video traffic in our experiments, we have used the video traffic generated from the EvalVid project (a video quality evaluation tool). EvalVid is a research project carried out by the TKN group at the Faculty of Electrical Engineering and Computer Science at the Technische Universität Berlin [17]. The video traffic generation in the NS-2 simulator follows the next steps:

1. We use uncompressed video sequences in CIF format (352×288), which are available from [18]. This video sequences are sampled in a videoconferencing format (format: 4:2:0).
2. After this, we compress the video files according to the H.264/MPEG-4 AVC standard. To this end, the videos sequences were encoded to 30 frames per second with a variable bit rate using the JM 17.1 software [19].
3. Then, the video frames are packaged using the Real-Time Protocol (RTP). The MP4Box software is used to carry through this task [20]. The size of each packet is 1024 B.
4. Next, a trace file is generated in order to be used in the NS-2 network simulator. This file is created using the mp4trace software, which is part of the EvalVid project [17].
5. Finally, a traffic source is generated in the NS-2 network simulator. This goal is reached by establishing an UDP shipping agent and a receiving agent in the simulator [17].

5.1. Throughput Evaluation

Our average throughput evaluation considers two different experiments related to the size of the video files. In the first experiment, we use the Foreman (300 frames), Akiyo (300 frames) and Football (90 frames) video sequences, which are allocated in the source 1, source 2 and source 3, respectively. We can see that a video sequence (Football) has a different size. In our second experiment, we use the following video sequences, Container (300 frames), Akiyo (300 frames) and Foreman (300 frames), which are allocated in the source 1, source 2 and source 3, respectively. Unlike the first experiment, now all video sequences have the same size. Figure 4 shows the system average throughput for both experiments. We can see that in experiment 1, the average throughput is higher if network coding is applied while the bandwidth is below 1 Mbit/s. After this threshold, the bandwidth is increased above 1 Mbit/s and the average throughput is stabilized. With respect to the experiment 2, we can see that the results are similar, but the average throughput is increased for both strategies, either using or not using network coding. However, the average throughput is still higher when the network coding strategy is applied.

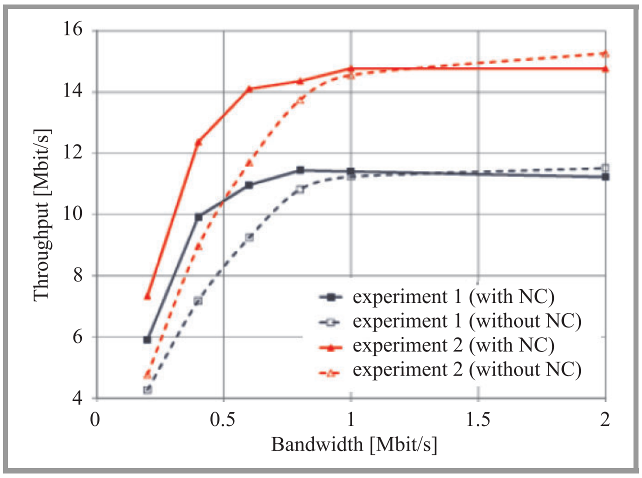


Fig. 4. Average system throughput for both experiments.

5.2. Video Quality Evaluation

Our second parameter to be evaluated is the video quality. We use the peak-signal-to-noise-ratio (PSNR) as the quality metric. For this experiment, the Foreman, Akiyo and Football video sequences are encoded at different bit rate using the JM software [19]. The bit rates used for encoding these video sequences are 16, 64, 200, 500 and 1000 kbit/s. We measure the PSNR in the three receiving nodes. However, in this paper, the PSNRs obtained in the receiving node 1 are shown only in Fig. 5. We can see how the Akiyo and Foreman video sequences reach a higher PSNR when network coding is applied than traditional video transmission without network coding.

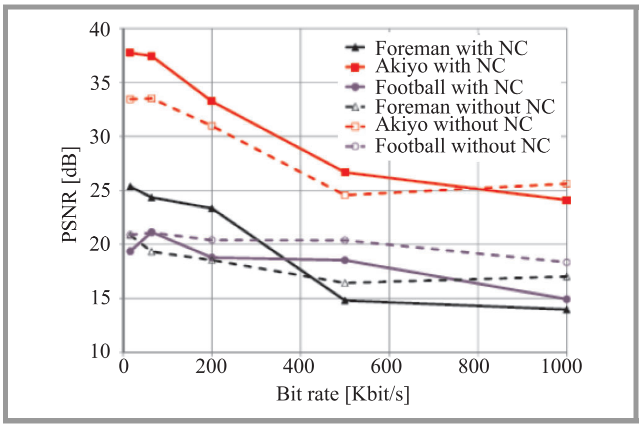


Fig. 5. Average PSNR for experiment 1.

On the contrary, the results obtained for the Football sequence are opposites. This is due the packet loss in the network. The Football video sequence is of smaller size than Akiyo and Foreman video sequences; therefore its PSNR is most affected by the packet loss. Thus, the video quality

Table 1
Percentages of packet loss for experiment 1

Sequence	Foreman	Akiyo	Football
Receiver $R1$			
With NC	0.02	0	37.4
Without NC	17.78	34.2	41.85
Receiver $R2$			
With NC	9.82	4	32.63
Without NC	17.76	34.14	51.3
Receiver $R3$			
With NC	0	0.2	46.57
Without NC	18.89	15	41.84

for the Football sequence in the receiving node could be not acceptable. From these findings, we can argue that video quality can be improved by using network coding during a packet loss event in the network.

The percentages of packet loss in each receiver node are summarized in Table 1. Receiver nodes $R1$ and $R2$ have a lower packet loss using network coding than if it is not used. However, the football video sequence has a high percentage of packet loss in all receiver nodes. This is because the football video sequence has a small size (90 frames) in comparison with the Akiyo and Foreman video sequences, which have 300 frames. Figure 6 compares a video frame

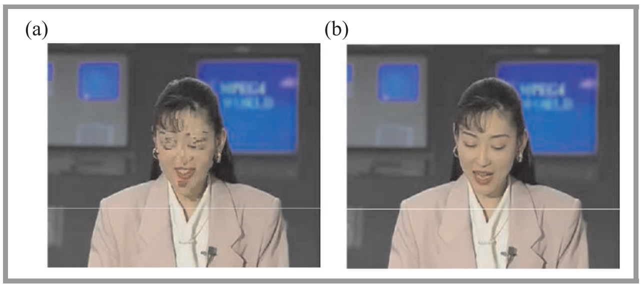


Fig. 6. A frame of the Akiyo video sequence during a transmission with packet loss: (a) without network coding; (b) using network coding.

of the Akiyo video sequence during a transmission with packet loss. Video frame in Fig. 6a is transmitted via a traditional network, while video frame in Fig. 6b is transmitted using network coding. We can observe that video quality is improved in the picture where network coding is used.

Figure 7 depicts the PSNR obtained from experiment 2. In this experiment, all video sequences have the same size. The resulting PSNRs are similar to PSNRs obtained from experiment 1. We can see how once again the PSNR is higher when network coding is applied. Contrary, if network coding is not used, the number of lost packet is increased and the video quality (PSNR) is proportionally reduced. Table 2 summarizes the percentages of packet loss in each receiver node for the experiment 2. All video sequences present a high percentage of packet loss in all

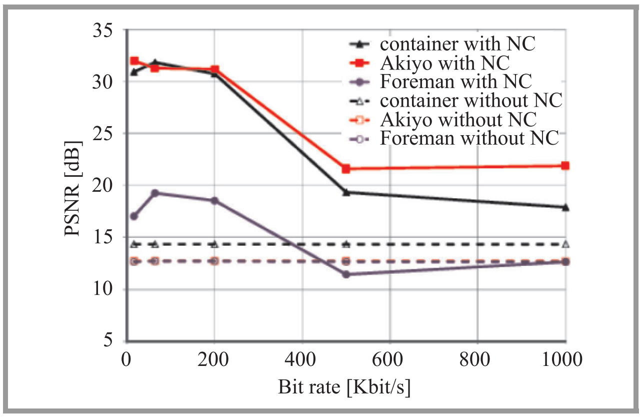


Fig. 7. Average PSNR for experiment 2.

Table 2
Percentages of packet loss for experiment 2

Sequence	Foreman	Akiyo	Football
Receiver $R1$			
With NC	0.0	0	5.10
Without NC	38.0	36.67	35.43
Receiver $R2$			
With NC	0.0	0.0	14.51
Without NC	38	36.67	46.02
Receiver $R3$			
With NC	14.27	0.0	0.0
Without NC	47.47	35.10	35.43

receiver nodes when network coding is not used. In contrast, we can observe that some video sequences are received free of packet loss by the receiver nodes when the network coding technique is used during the video transmission.

5.3. Evaluation of the Buffer Occupancy

Finally, a third experiment is implemented in order to evaluate the buffer occupancy during a video transmission with network coding. We realize this evaluation in the buffers of the intermediate nodes because network coding is realized in these nodes. Specifically, we measure the buffer occupancy at the node 3 from our simulation scenario (see Fig. 3). Figure 8 shows a comparison of buffer usage using

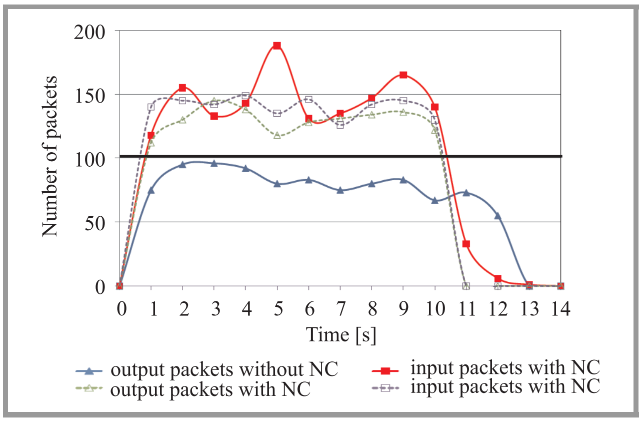


Fig. 8. Buffer occupancy during a transmission rate of 64 kbit/s.

network coding (NC) and a traditional transmission (without network coding) during a video transmission with a bit rate of 64 kbit/s. Threshold indicates the buffer size, which can store 100 packets of 1KB. We can see that during a traditional transmission, the number of input packets arriving to the buffer is higher than the number of output packet from it. This fact can produce packet loss. Contrary, in a video transmission with network coding, packets arriving to the buffer are combined and sent immediately to the next node. Thus, two or more packets can be attended at same time. Our experiments indicate that the combining time at the buffers are small and the packet loss is re-

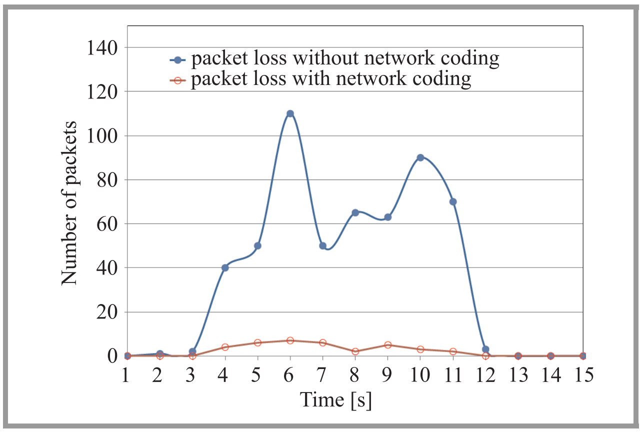


Fig. 9. Packet loss during a video transmission.

duced. Figure 9 compares the packet loss for these cases. Results shows that network coding can notably reduce the packet loss during a video transmission in communication networks with limited bandwidth. However, this comparison of results can be different when the communication networks have abundant bandwidth.

6. Conclusions

In this paper, we presented and evaluated a multi-source scheme using network coding for video delivery to a set of receiving nodes. Our network coding scheme does not require a centralized knowledge of the network and introduces a new package format, which contains a global coding vector and the payload. The video packets are streaming via UDP connections. We evaluated our multi-source scheme using the network simulator NS-2. Different video sequences were compressed under the H.264/MPEG-4 AVC standard and evaluated in this simulator. Mainly, we interest to evaluate the network performance and video quality during a video transmission using network coding. Additionally, buffer performance using network coding is evaluated. The results show that our scheme with network coding achieves a promising performance in terms of throughput, delay, packet loss, buffer usage and video quality compared with traditional networks.

We plan to extend our effort toward the peer-to-peer (P2P) networks. P2P networks are a promising infrastructure for video delivery. We believe that network coding can introduce several benefits in multi-source video multicast systems based on peer-to-peer networks. Also, scalable video techniques combined with network coding can be investigated.

References

[1] T. Nguyen and A. Zakhor, "Multiple sender distributed video streaming", *IEEE Trans. Multimedia*, vol. 6, no. 2, pp. 315–326, 2004.

[2] F. A. López-Fuentes and E. Steinbach, "Multi-source video multicast in peer-to-peer networks", in *Proc. IEEE Int. Symp. Parallel Distrib. Process. IPDPS'08*, Miami, FL, USA, 2008, pp. 1–8.

[3] R. W. Yeung, S.-Y. R. Li, N. Cai, and Z. Zhang, "Network coding theory", *Foundation Trends in Communications And Information Theory*", vol. 2, no. 4 and 5, pp. 241–381, 2005.

[4] R. Ahlswede, N. Cai, S.-Y. Li and R. W. Yeung, "Network information flow", *IEEE Trans. Inform. Theory*, vol. 46, pp. 1204–1216, 2000.

[5] B. Bollobas, *Graph Theory, and Introductory Course*. New York: Springer, 1979.

[6] P. Chou, Y. Wu, and K. Jain, "Practical network coding", in *Proc. 51st Allerton Conf. Communi., Control and Comput.*, Monticello, IL, USA, 2003.

[7] C. Cabrera-Medina, F. A. López-Fuentes, "Video transmission in multicast networks using network coding", in *Proc. Int. Conf. P2P, Parallel, Grid Cloud and Internet Comput. 3PGCIC-2011*, Barcelona, Spain, 2011, pp. 350–354.

[8] C. Gkantsidis and P. R. Rodriguez, "Network coding for large scale content distribution", in *Proc. IEEE INFOCOM 2005*, Miami, FL, USA, pp. 2235–2245, March 2005,.

[9] J. K. Sundararajan, D. Shah, M. Medard, M. Mitzenmacher and J. Barros, "Network coding meets TCP", in *Proc. IEEE INFOCOMM 2009*, Rio de Janeiro, Brazil, 2009, pp. 280–288.

[10] L. Keller, E. Atsan, K. Argyraki, and C. Fragouli, "SenseCode: Network Coding for Reliable Sensor Networks", Tech. Rep. 2009, Ecole Polytechnique Federale Lausanne (EPFL). Last updated July, 2010.

[11] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss and P. Levis, "Collection tree protocol", in *Proc. 7th ACM Conf. Embedded Netw. Sensor Sys. SenSys*, Berkeley, CA, USA, 2009.

[12] P. Levis, N. Lee, M. Welsh, and D. Culler, "TOSSIM: Accurate and scalable simulation of entire TinyOS applications", in *Proc. First ACM Conf. Embedded Netw. Sensor Sys. SenSys*, Los Angeles, CA, USA, 2003.

[13] L. Lima, S. Gheorghiu, J. Barros, M. Medard, and A. Toledo-López, "Secure network coding for multi-resolution wireless video streaming", *IEEE J. Selec. Areas Commun.*, vol. 28, no. 3, pp. 377–388, 2010.

[14] S. McCanne, S. Floyd, and K. Fall, "The network simulator, ns-2" [Online]. Available: http://www.isi.edu/nsnam/ns

[15] D. Nguyen, T. Nguyen, and X. Yang, "Multimedia wireless transmission with network coding," in *Proc. Packet Video*, Lausanne, Switzerland., 2007, pp. 326–33.

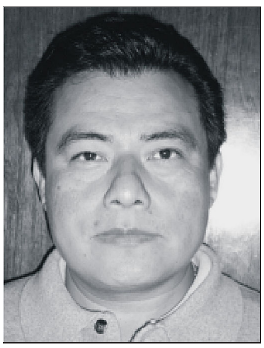
[16] T. Noguchi, T. Matsuda, and M. Yamamoto, "Performance evaluation of new multicast architecture with network coding", *IEICE Trans. Commun.*, vol. E86-B, no. 6, pp. 1788–1795, 2003.

[17] EvalVid, "A video quality evaluation tool" [Online]. Available: http://www.tkn.tuberlin.de/research/evalvid/

[18] "Video sequences" [Online]. Available: http://www.tkn.tu-berlin.de/research/evalvid/cif.html

[19] "JM v.17.1 software", Fraunhofer Heinrich-Hertz-Institut [Online]. Available: http://ip.hhi.de/index.htm

[20] "MP4Box software" [Online]. Available: http://www.tkn.tu-berlin.de/research/evalvid/



Francisco de Asís López-Fuentes received the B.Sc. degree in Electrical Engineering from the Oaxaca Institute of Technology, México. He received the M.Sc. degree in Computer Science Minoring in Networking from the Monterrey Institute of Technology (ITESM), and the doctoral degree in Electrical Engineering from Technical University of Munich, Germany. From 1988 to 1994, he was a design engineer in Siemens, Mexico. From 1996 to 2003, he was an Associate Professor at Technological University of Mixteca, México.

Dr. López-Fuentes is currently the Associate Professor of the Department of Information Technology at Autonomous Metropolitan University – Cuajimalpa Campus, México City, México. He has also served as a reviewer for several journals and conferences. His current research interests are in the area of networked multimedia systems, peer-to-peer networks, distributed systems as well as network security.
E-mail: flopez@correo.cua.uam.mx
Department of Information Technology
Autonomous Metropolitan University – Cuajimalpa
Constituyentes Av. 1047
11950 México, D. F., México

Cesar Medina-Cabrera is an information technology professional currently employed in the electricity industry. He spends most of his work time engaged in communication networks and video streaming applications. He holds the bachelor degree in Electronic Engineering from Autonomous Metropolitan University in México. At present he is finishing his master's degree in sciences and information technology at the same university. His research interests include the computer networks and communication systems.

QRouteMe: A Multichannel Information System to Ensure Rich User-Experiences in Exhibits and Museums

Antonio Gentile[a,b], Salvatore Andolina[a,b], Antonio Massara[a], Dario Pirrone[a,b], Giuseppe Russo[a], Antonella Santangelo[a], Salvatore Sorce[a,b], and Eleonora Trumello[a]

[a] *InformAmuse s.r.l., Palermo, Italy*

[b] *Department of Chemical, Management, Computer Science and Mechanical Engineering, University of Palermo, Palermo, Italy*

Abstract—In this article the QRouteMe system is presented. QRouteMe is a multichannel information system built to ensure rich user experiences in exhibits and museums. The system starts from basic information about a particular exhibit or museum while delivering a wide user experience based on different distribution channels. The organization of the systems' components allow to build different solutions that can be simultaneously delivered on different media. A wide range of media from touch-screen installations to portable devices like smartphones have been used. The used devices can communicate each others to increase the usability and the user experience for the visitors. Another important feature of the system is the definition of an inexpensive auto-localization system based on fiduciary marks distributed all around the building. In this article the system is presented from an architectural and functional point of view. A case study and analysis of experimental results are also provided in a real environment where the system was deployed.

Keywords—Exhibition, Integration, Multichannel information system, User-experience, Museums, Personalization.

1. Introduction

QRouteMe is an information system built to provide rich user-experienced for users attending museums or exhibits. The system infrastructure is organized in a modular way and the deploy is multichannel. The definition of an information system involves different research aspects starting from the methodology choices that have to be sustained. The modularization of the system is a direct consequence of the methodology chosen. At the top level is important to define the interaction process between the users and the system. The definition of a good user experience which is basically the interaction process starts from the analysis of the users' needs. Another important aspect is the goal of the system. This goal is described as a set of product objectives.

According to this objectives the contents are organized. The definition of a proper user experience starts from the definition of the typical classes of users for the information system. To classify the users a set of parameters are explicitly defined and, accordingly to a proper configuration, a set of models of typical categories. This classification process is often performed off-line as a prerequisite of the information system. Many classifications have been proposed. Falks [1] proposes five different categories of users: explorers, facilitators, experience seekers, professional/hobbyist, and rechargers. The personal context of the user includes aspects such as prior knowledge and specific interests, the physical context of the exhibit and the socio-cultural context related to inter and between group interactions. In [2] a simple taxonomy of the possible models is presented. To define users' models can be organized auto-evaluation tests or other support materials so the model can be implicit or explicit.

The definition of a complex information systems for museums and exhibits involves three main research areas. The first is related to contents organization: the proper content organization for the system is a knowledge definition problem. Different approaches can be used to perform the tasks related to content organization. The second research area is user modeling: the definition of a user model is a multidisciplinary research field and involves aspects related to the organization of relevant information to classify users. Another related aspect is the localization and tracking problem in an indoor environment. The user position and his movements inside the building are crucial information to define the class of the users. The third aspect is strictly related to contents fruition and user experience definition: the way and the media used in the fruition of the user experience are related to human computer interaction and his aspects. This paper addresses some of the presented aspects in this introduction. The architectural and functional solutions adopted led to the QRouteMe system. QRouteMe is a complex, multichannel information system to build and deliver rich user-experiences in museums and exhibits.

The system is able to personalize contents and to delivery them in different types of media. We use both stationary (totem/kiosk) media handlers based on surface computing to allow users a simple gesture-based interaction process, and mobile (smartphones, tablet) ones to allow adaptation in contents fruition. The purpose is to build a more personalized user experience. An important characteristic of the system is his adaptivity. The system is able to adjust the presentation of information and the user representation.

This is a key feature for systems that present limitations related to technical resources such as screen sizes, battery consumption, ergonomics, connectivity and limitations related to the user interaction process like haptic capabilities, working memory or even simply limited amount of time for a visit. The aspect of contents organization related to adaptation has been observed in several related studies.

A first important classification is presented in [3] where essentially three types of adaptive strategies are described. The first, defined as "adapted strategy", induces pre-optimization of contents and resources from the awareness of limitations.

The second type is an "adaptive strategy", where the system reacts to external changes in a sort of parameterized way and "adapting strategies" where is possible to handle different strategies according to environmental inputs. The adopted solutions for the QRouteMe system are essentially of the second type. We have observed of a series of environmental limitations and produced a set of strategies related to each of the initial constraints. A typical example is the visualization of information in smartphones with different screens. The produced output is able to adjust the contents organization according to screen size without any additional processing.

Another important feature in terms of adaptation is related to positioning process. In particular, the system is able to determine the user's position inside an indoor environment by means of fiduciary posts or interaction at kiosk location. Current smartphones localization system such GPS (Global Positioning System) have a resolution for indoor environments that is approximatively of 10 m. This precision is clearly inadequate in many situations, such as fairs and museum exhibits where a considerable amount of information can be located in a 10 m radius.

An important achievement for our system is related to his low-level deployment cost. The produced system doesn't need an expensive infrastructure to produce a rich user experience. The infrastructure organization is a typical client-server solution where one of more servers performs the tasks related to organization information and communication between the users and the system and a set of stationary media serving points connected to servers show information. A wireless network allow users' mobile devices to exchange information and to connect the client side to servers. To locate users no other sensor/actuator devices are needed. We are focused mostly on the interaction process. To such purpose the utilization of new technologies, such as surface computing used to develop more intuitive gesture-based interfaces have been proven to be effective.

In addition to this, the increased computing capacity of current portable devices shifts some of the computational load to the client side of the system. A significative plus aspect is the possibility to constantly interact with everyday life devices that can collect large amount of information to be used in personalized systems. Also the integration between devices with different connection capabilities is a technological key aspect able to allow different levels of fruition.

The rest of the paper is arranged as follows: next section reports related works. The third section of the paper depicts the system infrastructure. The fourth section presents the first implemented use case and the related experimental results. The paper ends with conclusions and future works.

2. Related Works

The definition of systems able to support users in indoor environments like an exhibit or a museum is an active and multidisciplinary research field with different research areas. As stated the main aspects are related to the process of contents definition and organization, to personalization for different users. In addition to this, many related problems have been investigated such as the localization of users for indoor environments.

One of the first works in this field is the Cyberguide project [4]. The main purpose was the definition of a set of different prototypes both for indoor and outdoor guides designed as a combination of four main components: a cartographer component including the map (or maps) of the physical environments, a librarian component, the information repository containing all the information to be presented, a navigator component used to keep track of the users' positions in the environment and a messenger component used to record and exchange messages to/from users and the system.

Other relevant systems are the Hippie/HIPS project [5] that is focused on development of an exhibition guide, providing guidance and information services. From the observations about the visitor's movements through the exhibition the systems creates a user profile and suggests other interesting exhibits or paths inside the current exhibits.

The TellMaris [6] system is an example of a mobile tourist guide developed combining both two and three-dimensional graphics running in a mobile phone. The possibility to automatically define related information for a guide has been exploited in many projects such as the PEACH project [7] where the generation of some position related contents and post-visit reports are automatically performed. The CHIP project [8] tries to combine different semantic web techniques to provide personalized access to digital museum collections both online and in the physical museum. Most of the works in this area are focused on an explicit definition of a knowledge base while some works tries to implicitly define a user model. The user model definition is mostly based on statistical models rather than recommendation techniques [9].

Another point of view to build a museum guide is to target not just a single user but a group visiting a museum. The Sotto-voce [10] system is designed specifically with this goal providing a communication mechanism to support interaction. From an architectural organization a complex system able to produce and adapt contents for different media has been organized mainly as a client server architec-

ture or as a multi-agent system. The drawbacks of the two approaches are well known. In the first case the server machine is obviously a point of failure and also the communication through the network can be critical while an explicit message passing mechanism has to be implemented for an agent based system together with a knowledge base used to define the communication ontology for the agents.
According to users' localization there are essentially two types of location technologies: the indoor positioning and the outdoor positioning. The second class of problems has been solved using satellite infrastructure with GPS. The reached level of accuracy is in the order of some meters and is generally a good accuracy for outdoor-based information applications. The indoor location suffers of the degrading reception of GPS-based systems. Furthermore the accuracy is not so important while in many cases the users needs a system able to recognize boundaries and able to positioning a person through a symbolic location (e.g., "in the main hall" or in "the first room"). Several methods have been proposed to solve the problem using different media like infrared beacons [11], [12] or radio signals from wireless LAN [13], [14], RFID technology [15] or cameras and microphones [16] to detect user location.
One of the main drawbacks of the proposed approaches is related to the initial cost to organize a large-scale event like an exhibit. Another way to achieve the same functionality is through the detection of the position by comparison between a set of floorplans and an image taken from the cell-phone camera [17]. This method has a major disadvantage because it requires all the floorplans for a particular building to be known in advance.
All the proposed methods require an electronic infrastructure to facilitate measurements with all the necessary sensor/actuator devices. In this case a good compromise in a costs/benefits tradeoff can be an approach able to give to users not continuous information of their position but discrete information.
We are looking to provide an inexpensive, building-wide infrastructure to be used in a large-scale type of events with a number of users that can be measured in thousands of people. So the utilization of a fiduciary marker able to be easily recognized from users is a natural solution for this type of problems. In this way the user localization shifts from a continuous to a discrete problem. Some similar approaches have been recently used to solve this problem. In [18] a system used as a location-based conference guide called Signpost is presented. The system works only with Windows mobile phones but it can be used in large-scale events with no further costs due to other equipment.

3. System Infrastructure Definition

The main goal of the QRouteMe system is to provide users with domain information, according to their actual need and to the context they are currently part of. To this end, we decided to design the system infrastructure according to the client-server paradigm. This choice allowed us to obtain an easy-to-implement, easy-to-scale and easy-to-manage framework of components, which can be suitably used to reach our intended goal. The QRouteMe infrastructure is composed of three main components:

– QRouteMe Platform,

– QRouteMe Front End,

– QRouteMe On Site.

The first one operates at the server side, whereas the second and third components operate at the client side. Figure 1 shows the three main components, their modules and the overall data flow among them. In the following sub-sections we will give an overview of each component.

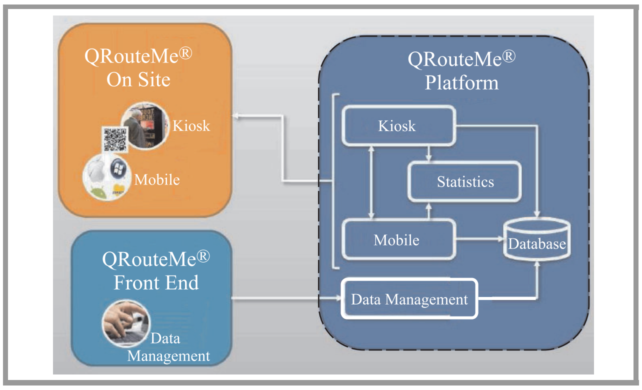


Fig. 1. QRouteMe infrastructure definition and data flow diagram.

3.1. QRouteMe Platform

On the server side, the QRouteMe Platform component main activities are data related: management, processing and storage. This component carries out most of the processing tasks of the whole system, and it is composed by four different modules (see Fig. 1):

– Kiosk,

– Mobile,

– Statistics,

– Data Management.

All of them interact, whether directly or not, with a database based on the relational model. Its schema has been defined in order to be easily adapted to the changing data domains and to the kind of service users have to be provided with. The data set views are generated using a server side scripting language, so they can be suitably and quickly adapted to the customer requirements in terms of data presentation. The database can also be populated by the client himself, using the QRouteMe Front End component. This feature gives the customer the possibility to keep the system up-to-date at any time, and to keep the full control on data integrity and correctness (these are critical properties, especially at the deployment time). The Kiosk and the mobile modules manage the users' requests according to the device they are actually using. These modules extract the useful data from the database, and compose the consequent information taking into account which channel the user is currently using to interact with the system. The Statistics module traces the users as they surf the information whilst completely preserving their anonymity and protecting their privacy. By monitoring a great number of parameters, it can provide administrators with a wide range of useful information about the user behavior. Customers may know how many users accessed the application, which and how many pages they were viewing, and so on, according to their feedback needs. This way, they could fine tune the interaction to improve the final user experience, or simply they could evaluate the system effectiveness. Of course, the information provided by the module can go through different levels of detail. It can carry on the overall system evaluation, the separate kiosk and mobile evaluation, or the evaluation of each device involved in the interaction. The Data Management module handles the interaction between the Platform component and the applications running on the Front End component.

3.2. QRouteMe Front End

The QRouteMe Front End component consists of ad-hoc tools made available to the customer and implemented according to his needs. The main goal of these tools is to allow the customer to create, read, update and delete data, either directly or by giving data owners the possibility to do it by themselves. This component interacts directly with the Data Management module within the Platform component (see Fig. 1). This ensures that all its activities can be carried out by keeping data integrity and correctness, while avoiding possible conflicts. Furthermore this component improve the system adaptability to the application domain changes, as well as the preservation of data privacy, making possible to avoid the intervention of personnel not related to the customer (such as technicians).

3.3. QRouteMe On Site

The QRouteMe On Site component provides the end users with the information suitably composed by the Platform component (actually by its Kiosk and Mobile modules, see Fig. 1). It represents the system interface with the users. This component consists of a set of applications which can be made to work together in different ways according to the customer needs. The QRouteMe On Site component allows people to access the available information by means of kiosks suitably configured, or by their personal mobile devices. Concerning the mobile devices in particular, applications can be natively designed and implemented for the most common operating systems (Android, iOS, Windows Mobile and Symbian), thus exploiting all software and hardware features. Nevertheless, if there is no need to use specific hardware features (such as accelerometers, cameras, positioning systems), a cross-platform web-based solution can be used, in order to have a better portability and a faster development. This component also links both kiosk- and mobile-based information access ways by means of QR codes (Quick Response) [19]. People can search for information on a kiosk and then transfer the desired output on their mobile, or people can directly access pieces of information by shooting at QR codes, provided that their mobile device is equipped with a QR reader.

4. Case Study

4.1. Deployment Event

QRouteMe has been implemented for the first time at the Vinitaly 2011, the largest fair dedicated to the Italian wine industry, which takes place every year in Verona (Italy) in April. It is organized by Verona Fiere that is responsible for all the logistic tasks. Our customer was the Istituto Regionale della Vite e del Vino (IRVV, Sicilian Institute of Vine and Wine), who asked us to implement an information system inside the "Sicilia" pavilion within the fair. This allows us to have a great number of information coming from a unique source. This is the most common case for exhibits and museums. The main goals of the system, as requested by IRVV, were:

- allow visitors to access information about Sicilian wine producers, even in mobility,
- help them moving around the pavilion with a localization system,
- obtain the list of services visitors usually look for (help-desk, secretary).

4.2. QRouteMe Layout at Vinitaly 2011

After the study of the location and of the booth layout, we decided that the optimal deployment of our system could be as follows:

- 14 fixed touch totems (InformaPoint),
- 2 web and database servers for redundancy,
- 1 wired LAN (designed by us and implemented by Verona Fiere),
- 1 wireless LAN with controlled access (designed by us and implemented by Verona Fiere),
- 1 database to store producers data, the booth layout, and other additional data;
- 1 app for mobile users of iOS-based devices (iPhone, iPod, iPad),
- web-based presentation of information for mobile users with other operating systems,

– 234 QR codes for the quick access to contextualized information.

Figure 2 shows the position of the fourteen touch-screen totems in the pavilion. The kiosk position was decided based on surface area and previous event knowledge about visitor entrance patterns to maximize the usage. The triangles in Fig. 2 represent the ways of access to the pavilion, whereas the placeholders "i" show the points where the fourteen touch totems InformaPoint were placed. Kiosk covered an average surface area of about 500 m^2, with higher concentration around the main entrance (at the lower left corner) and along the left side, closest to neighboring pavilions, where they could be easily accessed by visitors upon entering the pavilion. The overview of the installation setup is shown in Fig. 3.

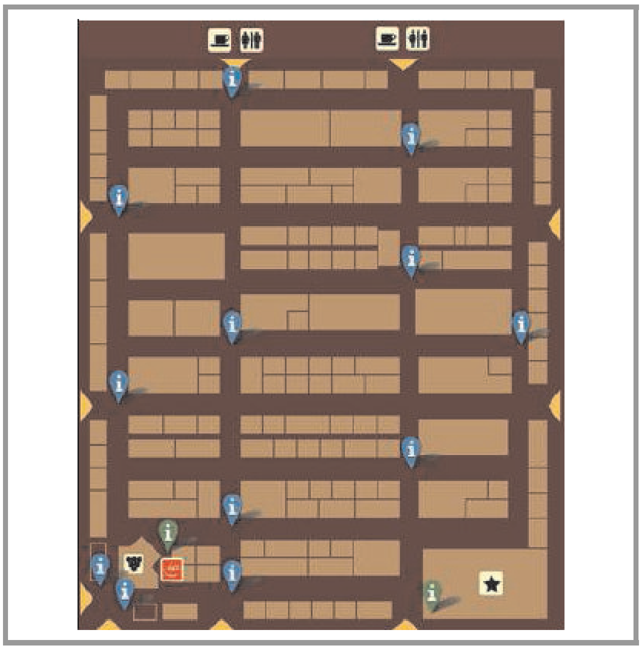


Fig. 2. Kiosks layout within the pavillion.

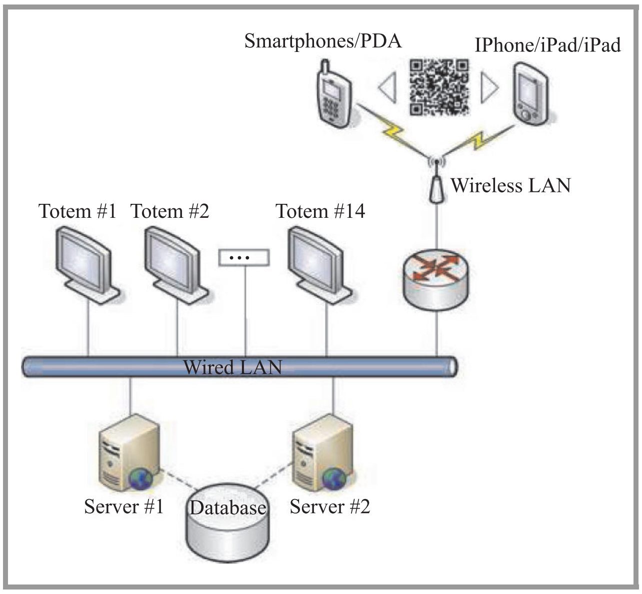


Fig. 3. QRouteMe installation setup.

4.3. Information Provided by QRouteMe

According to the IRVV needs and requirements, the QRouteMe system deployed at Vinitaly 2011 provided visitors with information about:

- Producers list, with alphabetical and direct search features, and related details:
  - producer information,
  - wine of the year for each producer,
  - list of produced wines for each producer,
  - QR code to quick access producer's information (audio, video, information forms),
  - producers booth position within the pavilion.
- Wine areas and related details:
  - area description and pictures,
  - producers belonging to a given vine area.
- Information about the IRVV.
- Information about services available within the pavilion.
- Information about the system.

The information were available both in Italian and English. Visitors equipped with a smartphone could access the system by means of their devices, exploiting the self-positioning feature based on the use of QR codes. They could see their current position within the pavilion, could access directly to information about the producer they were looking at, and could see the shortest path to reach a given producer's booth from their current position. Besides this, users of iOS-based devices could download an application (http://itunes.apple.com/it/app/sicilia-vinitaly-2011/id427870112?mt=8) that allow them to navigate a 3D model of the pavilion, to contact a producer via phone or email, and to share their own experience on the most common social networks.
This application includes a software module capable of decoding QR codes and interpret them correctly, according to the type of information that they contain. In addition to that, was developed a module capable of locating the visitor without using geo-referencing systems. In order to identify its location, the system asks the user to photograph a QR code. The decoded string indicates uniquely the position that photographed QR takes on the map (see Fig. 4), which coincides with that of the visitor who stands before it. Furthermore, this position can be used by the visitor as a starting point for generating a path to another point of interest.

Fig. 4. Path displayed in the iOS application for Vinitaly 2011.

5. System Usage Reports and Considerations

QRouteMe has been developed to allow friendly and intuitive interaction with the user. To this end the fourteen InformaPoints were equipped with a GUI designed only for touch-based interaction that proved to be very effective, and easily understood by users. The QRouteMe system was deployed at the Vinitaly 2011 international wine fair, held in Verona (Italy) from 7 to 11 April 2011. It worked for five days, during opening hours (about ten hours a day). This proved to be a good test bed for our system, both for its effectiveness and for its robustness against any kind of human or technical fault. The whole fair counted 8 big pavilions of about 8,000 m^2 each (72,000 ft^2), with a total count of more than 150.000 visitors from all over the world, either business operators or tourists. The system was actually deployed inside the "Sicilia" pavilion, one of the largest pavilions within the fair. During the working hours, we observed the behavior of people while interacting with totems, taking into account comments and observations. At the first day of the fair, we observed that the main goal of people was to single out a given wine producer from the available alphabetical list. This task was at first carried out by means of a vertical scrolling bar, with the aid of an alphabetical index allowing for the fast scrolling to the corresponding letter. Despite this possibility, we saw that that task was not so easy to accomplish, mainly due to the length of the list (more than 220 producers). We then decided to test the modularity of our system, by adding an interaction way while the system was up and running. In fact, we set up the on-screen keyboard direct search feature actually modifying the Kiosk module within the Platform component (see Fig. 1 for reference), with no need to interrupt the service operation.

Once deployed, the direct search by means of the superimposed on-screen keyboard became one of the features people found most useful during their interaction with totems This feature allowed them to easily find their preferred producer, its location within the pavilion, the path to reach it from the current position and the list of produced wines. During our survey, we observed that users experienced no other particular difficulty while interacting with the system. Once facing a totem, users started to interact with it in a few touches, independently of age and without any previous training. This is directly related to familiarity that users have with GUIs and gesture based interaction. Starting from the second fair day, we observed growing queues of people in front of totems, thus confirming the effectiveness of the interaction model and of the totem layout within the pavilion.

Fig. 5. People waiting for InformaPoint usage

Our informal observations about QRouteMe usage by InformaPoint totems were confirmed by the final reports generated by the system, with more than 40,000 accesses to the pages, and more than 13,000 accesses to the producers' pages. Table 1 shows a short resume of those reports both for the Italian and English version of pages.

Concerning the interaction from mobile devices, of course we could not observe the behavior of people while using their devices to access the system. We can only shortly discuss the reports resumed in Table 2 taken from the Statistics module of the system.

Table 1
QRouteMe usage report (totems)

Date	Single P.		Producer P.	
	Italian	English	Italian	English
07/04/11	4987	688	1480	180
08/04/11	8019	1063	2867	366
09/04/11	11469	945	4091	307
10/04/11	8379	529	2793	187
11/04/11	4146	122	1402	42
Subtotal	37000	3347	12642	1082
Total		40347		13724

Table 2
Mobile devices usage report

Data	Single page access
07/04/11	813
08/04/11	810
09/04/11	626
10/04/11	427
11/04/11	172
Total	2848

The number of total accesses to single pages is low if compared to that obtained by totem (with a ratio of about one to fifteen), due to different reasons:

– totems were well placed all around the pavilion, so that users could easily see them,

– totem places were clearly indicated with big banners hanging from the pavilion ceiling,

– there were no equivalently visible signs about the possibility to access the system by means of mobile devices,

– people who knew about the mobile access found anyway easier to use totems, instead of registering to the wireless network, download the app, install it, and finally use it.

For those users who did access the system via mobile devices (and available to fill a short satisfaction report), the level of satisfaction was rather high, as despite a small overhead, the application offered more contextualized information, practically out of your pocket, quoting one of those users.

6. Conclusions and Future Works

In this works QRouteMe has been presented. QRouteMe is a multichannel information system able to build and deliver rich user experiences in exhibits and museums. The overall system infrastructure and an instance, deployed in a large exhibit, were presented. The system presents several advantages starting from low deployment costs for very large environments. In the QRouteMe the user localization is approached as a discrete problem using a set of fiduciary markers that are able to auto-locate the users. The system architecture is built as a modular set of components that are chosen and selectively deployed according to the specific needs of the application, without any loss of effectiveness. Current works aims at implementing customizable front-end for final customers to allow for complete contents organizations and easiness of user experience and a more complete statistic component.

References

[1] J. H. Falk, *Identity and the Museum Visitor Experience*. Left Coast Press, 2009.

[2] T. Kuflik , J. Kay, and B. Kummerfeld, "Lifelong personalized museum experiences", in *Proc. User Modeling, Adaptation and Personalization UMAP*, Big Island, Havaii, 2010, pp. 9–17.

[3] R. Wasinger, A. Kruger, and O. Jacobs, "Integrating intra and extra gestures into a mobile and multimodal shopping assistant", in *Proc. 3rd Int. Conf. Pervasive Comput.*, Munich, Germany, 2005, pp. 297–314.

[4] S. Long, D. Aust, G. Abowd, and C. Atkeson, "Cyberguide: prototyping context-aware mobile applications", in *Proc. Conf. Human Factors in Comput. Sys.*, Vancouver, Canada, 1996, pp. 293–294.

[5] R. Oppermann and M. Specht, "A context-sensitive nomadic exhibition guide", in *Proc. 2nd Symp. Handheld Ubiquitous Comput. HUC2K*, Bristol, UK, 2000, pp. 127–149.

[6] C. Kray, K. Laakso, C. Elting, and V. Coors, "Presenting route instructions on mobile devices". in *Proc. Int. Conf. Intell. User Interfaces IUI'03*, Miami, USA, 2003, pp. 117–124.

[7] O. Stock, M. Zancanaro, P. Busetta, C. Callaway, A. Krüger, M. Kruppa, T. Kuflik, E. Not, and C. Rocchi, "Adaptive, intelligent presentation of information for the museum visitor in PEACH", *User Modeling and User Adapted Interaction*, vol. 17, no. 3, pp. 257–304, 2007.

[8] L. Aroyo, N. Stash, Y. Wang, P. Gorgels, and L. Rutledge, "CHIP demonstrator: semantics-driven recommendations and museum tour generation", in *Proc. Sixth Int. Semantic Web Conf. ISWC'07*, Busan, Korea, 2007, pp. 879–886.

[9] D. W. Albrecht and I. Zukerman, "Special issue on statistical and probabilistic methods for user modeling", *User Modeling and User-Adapted Interaction*, vol. 17, no. 1–2, 2007.

[10] P. M. Aoki, R. E. Grinter, A. Hurst, M. H. Szymanski, J. D. Thornton, and A. Woodruff, "Sotto voce: exploring the interplay of conversation and mobile audio spaces", in *Proc. SIGCHI Conf. Human Factors in Comput. Sys. CHI'02*, New York, NY, USA, 2002, pp. 431–438.

[11] A. Butz, J. Baus, and A. Kruger, "Augmenting buildings with infrared information", in *Proc. Int. Symp. Augmented Reality ISAR*, Munich, Germany, IEEE Computer Society Press, 2000.

[12] A. Harter and A. Hopper, "A distributed location system for the active office", *IEEE Network*, vol. 8, no. 1, pp. 62–70, 1994.

[13] P. Bahl and V. Padmanabhan, "Radar: an in-building rf-based location and tracking system", in *Proc. IEEE INFOCOM 2000*, Tel-Aviv, Israel, 2000.

[14] J. Krumm, G. Cermak, and E. Horvitz, "Rightspot: a novel sense of location for a smart personal object", in *Proc. UbiComp 2003*, Seattle, Washington, USA, 2003, pp. 36–43.

[15] B. Brandherm, T. Schwartz, "Geo referenced dynamic bayesian networks for user positioning on mobile systems", in *Proc. Int. Worksh. Location and Context-Awareness LoCA*. LNCS 3479, Springer, 2005.

[16] B. Xuehai, G. Abowd, J. Rehg, "Using sound source localization in a home environment", in *Proceedings of Pervasive Computing 2005*, LNCS 3468, Springer, 2005.

[17] H. Hile, G. Borriello, "Positioning and orientation in indoor environments using camera phones", *Comp. Graphics and Appl.*, IEEE, vol. 28, no. 4, pp. 32–39, 2008.

[18] A. Mulloni, D. Wagner, I. Barakonyi, D. Schmalstieg, "Indoor positioning and navigation with camera phones", *IEEE Pervasive Comput.*, vol. 8, no. 2, pp. 22–31, 2009.

[19] "QR Code.com", June 2011 [Online]. Available: http://www.denso-wave.com/qrcode/aboutqr-e.html

Antonio Gentile is an Associate Professor at the Department of Chemical, Management, Computer Science and Mechanical Engineering of the University of Palermo. He holds the Ph.D. in Computer Engineering from the Georgia Institute of Technology (USA) and a Ph.D. in Computer Science from the University of Palermo. His research areas are human computer interaction and computer architecture.

E-mail: gentile@informamuse.com
InformAmuse s.r.l.
Viale delle Scienze Ed 16
90128 Palermo, Italy

E-mail: antonio.gentile@unipa.it
Department of Chemical, Management, Computer Science and Mechanical Engineering
University of Palermo
Viale delle Scienze Ed 6
90128 Palermo, Italy

Salvatore Andolina holds the Laurea degree in Computer Engineering from the University of Palermo and is a Ph.D. student in Computer Engineering at the same university. His expertise is in the field of software engineering and web application design.

E-mail: andolina@informamuse.com
InformAmuse s.r.l.
Viale delle Scienze Ed 16
90128 Palermo, Italy

E-mail: salvatore.andolina@unipa.it
Department of Chemical, Management, Computer Science and Mechanical Engineering
University of Palermo
Viale delle Scienze Ed 6
90128 Palermo, Italy

Antonio Massara has an econonomical degree from the University of Palermo. Since 2010 he is a member and active contributor to InformAmuse. He has extensive experience in advertising, communications, and marketing management. He is the marketing expert of the group and has a very high experience in start-up and new companies. His expertise is in the development of new media and business models.

E-mail: massara@informamuse.com
InformAmuse s.r.l.
Viale delle Scienze Ed 16
90128 Palermo, Italy

Dario Pirrone holds the Laurea degree in Computer Engineering from the University of Palermo and is a Ph.D. student in Computer Engineering at the same university in the Department of Chemical, Management, Computer Science and Mechanical Engineering. His expertise is in the field of knowledge engineering, Human Computer Interaction and web application design.

E-mail: pirrone@informamuse.com
InformAmuse s.r.l.
Viale delle Scienze Ed 16
90128 Palermo, Italy

E-mail: sdario.pirrone@unipa.it
Department of Chemical, Management, Computer Science and Mechanical Engineering
University of Palermo
Viale delle Scienze Ed 6
90128 Palermo, Italy

Giuseppe Russo is currently a post-doc fellow at the Department of Computer Engineering of the University of Palermo. He holds a Ph.D. degree in Computer Engineering from the same University. His expertise is in the field of knowledge engineering and in the interaction design for web and mobile applications

E-mail: russo@informamuse.com
InformAmuse s.r.l.
Viale delle Scienze Ed 16
90128 Palermo, Italy

Antonella Santangelo currently works as a researcher at Informamuse s.r.l. an Academic SPIN-OFF of the University of Palermo. She holds the Ph.D. degree in the Computer Science and Artificial Intelligence from the University of Palermo. She received her "cum laude" Laurea degree in Ingegneria Informatica on July, 2005. Her research interests are in the field of natural language processing and multimodal human-computer interfaces.

E-mail: santangelo@informamuse.com
InformAmuse s.r.l.
Viale delle Scienze Ed 16
90128 Palermo, Italy

Salvatore Sorce studied Computer Science and Engineering at the University of Palermo, and obtained his M.Sc. (2001) and Ph.D. (2005) degrees. Currently he is post-doc fellow with the Department of Chemical, Management, Computer Science and Mechanical Engineering at the University of Palermo. His main research interests deal with: pervasive systems; wearable computers; positioning systems; personal mobile devices programming.

E-mail: sorce@informamuse.com
InformAmuse s.r.l.
Viale delle Scienze Ed 16
90128 Palermo, Italy

E-mail: salvatore.sorce@unipa.it
Department of Chemical, Management, Computer Science and Mechanical Engineering
University of Palermo
Viale delle Scienze Ed 6
90128 Palermo, Italy

Eleonora Trumello holds the Laurea degree in Computer Engineering from the University of Palermo. Her expertise is in the field of human computer interaction. She participated in the production of an e-learning platform for digital terrestrial. She participated in the production of speech interfaces for desktop and telephone applications. She is a member and active contributor to InformAmuse since 2010. She participated in the production of seven iPhone applications, two Android application, and an innovative system for large events and large spaces, deployed to Vinitaly 2011, London International Wine Fair 2011, Fruit Logistica 2012, ProWein 2012, and Vinitaly 2012.

E-mail: turmello@informamuse.com
InformAmuse s.r.l.
Viale delle Scienze Ed 16
90128 Palermo, Italy

Paper

Component-Based Architecture for Systems, Services and Data Integration in Support for Criminal Analysis

Jacek Dajda, Roman Dębski, Aleksander Byrski, and Marek Kisiel-Dorohinicki

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

Abstract—Criminal analysis processes is based on heterogeneous data processing. To support it, analysts utilize a large set of specialized tools, however they are usually designed to solve a particular problem are often incompatible with other existing tools and systems. Therefore, to fully leverage the existing supporting tools, their technological integration is required. In this paper we present original approach for integrating systems based on the component-driven paradigm. Firstly, a problem of supporting criminal analysis is described with a strong emphasis on the heterogeneity issues. Secondly, some theoretical information about integration is depicted followed by the details of the proposed architecture. Finally, the technological assumptions are discussed and prototype integration based on proposed concept is overviewed. om the experiments are discussed in the final part of the paper.

Keywords—criminal analysis, component-based systems, software integration.

1. Introduction

The continuous technological development affects various domains and aspects of life resulting in progressing informatization of organizations, procedures but also daily routines and habits. Interestingly, this process can be also observed in such area as police operation activities and criminal analysis. On the other hand, contemporary criminals use more and more sophisticated methods based on modern information technologies, which again, cause police analysts to develop new techniques. While these techniques often prove useful, every new information source or analyzing scheme requires additional time to be processed or executed.

Therefore, to support analyst in this process, various software tools are developed. They operate on different data and data formats as well as provide different set of features. These tools are developed by different software teams and often in different technologies and architectures. As result while specific activities are supported, there are still significant support gaps related to data handling and colligation as well as operation flows arrangement and execution. These gaps are also caused by the problem of new data sources and operations that are introduced continuously into the whole process. These sources and operations often remain unhandled by the existing tools that were designed for other purposes and are not generic enough.

Taking all of these under consideration, it must be assumed that the proper solution should be based on the concept of integration. The proper designed integration-driven architecture should be opened for adding new information sources and analytical operations provided by existing libraries and systems. On the other hand, it should remain lightweight and scalable so that can applied both to single desktop applications as well to larger systems.

The goal of this article is to present the concept of such architecture. To make the architecture flexible, it is proposed to take advantage of the component paradigm [1]. This paradigm assumes assembling applications from a set of independent components through well-defined functional contracts. By adding new components to the system, it can be extended to cover new requirements. What is more, the proposed architecture assumes the reuse of the component-based approach on all the architectonic levels from the data layer to the graphical user interface layer.

The structure of the paper is as follows. In the next section, the basics of the criminal analysis process are presented. Next section depicts current state of art. Section 4 is dedicated to the overview of the proposed architecture. In Section 5, the key mechanism behind the architecture are explained. Section 6 discusses the technological assumptions for architecture realisation. Section 7 presents an example implementation and preliminary evaluation of the proposed architecture. Finally, Section 8 contains the summary and plan for future work.

2. Supporting Criminal Analysis

The process of criminal analysis can be briefly described as a loop of the following intertwined activities: data-retrieval, processing, visualisation and interpretation. The rule of operation depends on a given scenario. The general concept is presented in Fig. 1.

It assumes that the analyst operates on various information sources such as police databases and Internet services and also on such operational data as phone billings. The data retrieval from a specific source requires a dedicated tool or parser. After the data is retrieved it can be filtered, searched, and analyzed with the use of available operations and algorithms such as pattern recognition algorithms [2] or hypothesis testing based on social networks

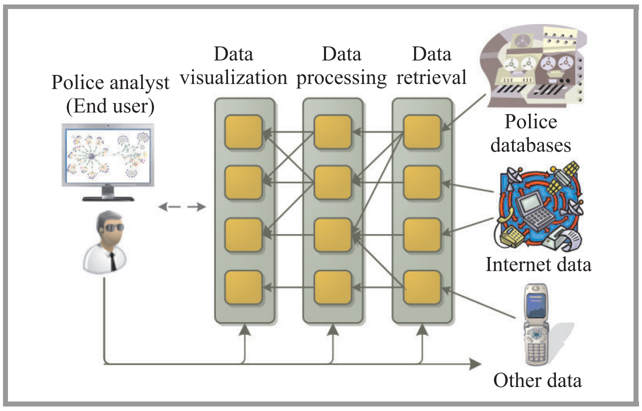


Fig. 1. Criminal analysis.

approach [3], [4]. Typically, there is a specific range of operations which can be executed on a data coming from a given source.

The result of the data processing stage is usually a subset of the initial data, which can be visualized using some predefined visualization type. The visualization stage assumes a considerable amount of human interaction in order to extract additional information based on such prerequisites as specific layout of data.

As stated in the beginning, the whole process is not a one-time sequence of operations but a continuous loop in which the analyst, based on his/her findings inspired by the specific visualization, can go back to specific stage and repeat the whole process by performing new visualizations or other operations on the actual data as well as adding new data or even new data sources to the analysis.

No doubt, the whole process can be time-consuming and error-prone. Thus, a proper software support is needed. While the results interpretation must be still carried out by the analyst (although in near future it can be also supported by artificial intelligence reasoning), the rest three activities can be greatly optimized by the proper software tool support. However, these tasks pose certain challenges.

First of all, the range of data sources is unlimited and can vary from Internet pages, through data-bases to simple text files containing data in structure or unstructured form. What is more, each data source may produce a unique set of data types which require specific processing. Once the data is obtained and the processing results are ready, they need to be visualized in order to be interpreted by the analyst. Again, each data type may require a dedicated visualization type, which, to make it more complex, may be linked with other visualization to form an efficient graphical user interface for the analyst.

What the data sources have in common is that they usually produce large quantities of information which is practically impossible to process manually. This constitute another challenge for the designers of supporting solutions.

Currently, there is a variety of tools and systems available that offer a different amount of support for certain data domains or operations. These solutions can be classified into two groups: specific and general-purpose ones. An example of the specific solution is the LINK platform [5] which allows for data analysis and visualization using object graphs. As for the general-purpose tool, Excel can be given as an example. During the process of analysis, these solutions are often utilized together. However, their integration is often limited or none. As result, while the individual solutions are of great help for the analyst, he/she spends a considerable amount of time on data transformations and moving the results from one solution to the other in order to continue the investigation plot.

Theoretically, the problem could be solved by constructing a large, scalable all-purpose analytical environment which will allow for handling all possible data sources and will provide all the necessary services and visualizations. However, from the practical and economical point of view this is impossible and cannot be accepted as a solution to the problem. Another utilized approach is the integration of the available legacy solutions and databases. In most cases, this assumes integration of specific products such as database and analytical system. However, in the discussed case, this is not a fully satisfactory solution due to the number of possible sources and utilized systems. What is more, new sources or systems can be added in future which emerges from the organizational and other aspects that are beyond the decisions of software architects.

All of this makes the problem even more challenging. It seems that the core of the problem lies in the heterogeneity. The heterogeneity can be observed onto 3 following levels:

– data sources,

– service providers (operations),

– technologies.

Data sources can vary in the data containers, formats and standards of the data. Service providers can differ in the rule of operation, semantics of the input arguments and produced results. As for heterogeneous technologies, this aspect must be considered on every stage of the analytical process which means data sources, service providers and visualization libraries.

No doubt, the heterogeneity should be approached and handled on the general architectural level. In the next section the current state of art in terms of software integration is overviewed.

3. Software Integration Styles

Application integration is nowadays a mature engineering discipline – good practices are cataloged in the form of patterns [6]. All the patterns can be generalized and divided into four main categories: *file transfer, shared database, remote procedure invocation* and *messaging*. The following sections describe briefly each of them.

3.1. File Transfer

This approach to integration utilizes files – universal storage mechanism available in all operating systems. One application (producer) creates a file that contains the information needed by the other applications. Next, the other

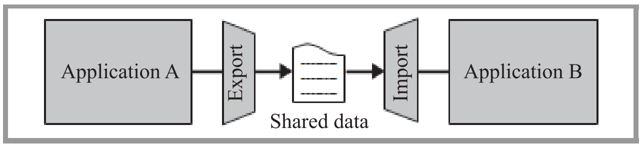


Fig. 2. File transfer.

applications (consumers) can read the content of the file (Fig. 2). Choosing this approach has the following consequences [6]:

- it is *data sharing oriented* (not *functionality sharing oriented*),
- files are (effectively) the public interface of each application,
- choosing the right file (data) format is very important (nowadays it is often XML),
- applications are decoupled from each other,
- applications are responsible for managing the files (creation, deletion, following file-naming conventions etc.),
- updates usually occur infrequently and, as a consequence, the communicating applications can get out of synchronization,
- integrators need no knowledge of the internals of applications.

3.2. Shared Database

In this pattern the integrated applications store their data in a single (shared) database. The stored data can be immediately used by the other applications (Fig. 3). Choosing this approach has the following consequences [6]:

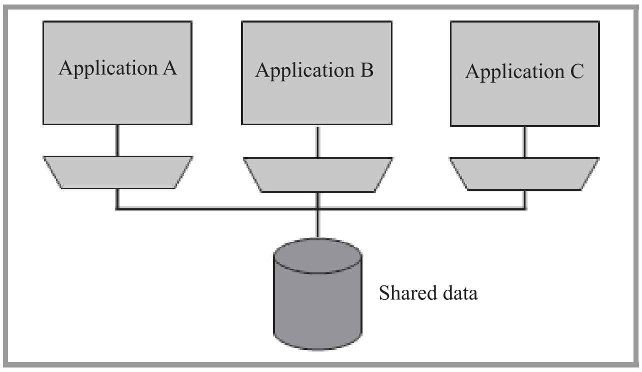


Fig. 3. Shared database.

- it is *data sharing oriented* (not *functionality sharing oriented*),
- data in the database are always consistent,
- defining a unified database schema that can meet the needs of many applications can be a really difficult task,
- any change of the shared database schema may have impact on all integrated applications (applications are strongly coupled),
- since every application uses the same database, there is no problem with *semantic dissonance* [6],
- shared database can become a performance bottleneck and can cause deadlocks.

3.3. Remote Procedure Invocation

In this approach each part of the integrated system (a set of cooperating applications) can be seen as a large-scale object (or component) with encapsulated data. Shared functionality of each application is accessible via its public interface[1] (Fig. 4). Choosing this approach has the following consequences [6]:

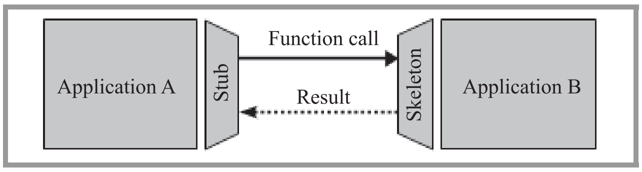


Fig. 4. Remote procedure invocation.

- it is *functionality sharing oriented* (not *data sharing oriented*),
- applications can provide multiple interfaces to the same data,
- applications are still fairly tightly coupled together (often each application of the integrated system perform a single step in many-step algorithms: in such a case one application's failure may bring down all of the other applications),
- communication is (usually) synchronous,
- developers often forget that there is a big difference in performance and reliability between remote and local procedure calls – it can lead to slow and unreliable systems.

[1]A number of technologies such as CORBA, COM, .NET Remoting, Java RMI and Web Services implement Remote Procedure Invocation (also referred to as Remote Procedure Call or RPC).

3.4. Messaging

This approach combines all the benefits of the previous three and is often considered [6] as the best one. Messages transfer packets of data frequently, immediately,

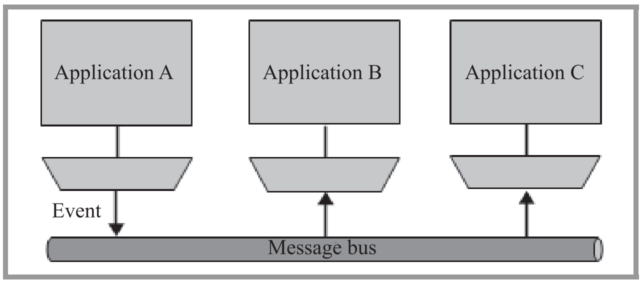


Fig. 5. Messaging.

reliably and asynchronously using customizable formats (Fig. 5). Choosing this approach has the following consequences [6]:

- sending small messages frequently allows applications to collaborate behaviorally as well as share data,
- applications are decoupled (it has many consequences e.g., integration decisions can be separated form the development of the applications),
- the integrated applications (usually) depend on a messaging middleware.

4. Component-Driven Integration Architecture

Each of the presented integration styles has its advantages and disadvantages. Also, each one of them imposes specific requirements on the integrated systems. While fulfilling these requirements is possible for developers of the integrated systems, the same cannot be assumed when dealing with legacy, or worse, external systems. In this case, the choice of the proper integration style may vary on the specifics of given systems.

It seems that in the discussed problem a greater level of flexibility is needed. One that will allow to embed all the styles and utilize the one that suits best the integrated systems. To achieve this flexibility, it is proposed here to take advantage of the component paradigm.

Using the paradigm, it is possible to build an extensible architecture in which various integration styles can be realized by adding new dedicated components. In such approach, the integration can be view from the following two perspectives: low-level and high-level.

The high-level perspective is the one already discussed in the previous section. It means integration of data sources and systems by establishing data and service links. In the low-level perspective, integration refers to the individual components and their contracts. By providing proper components in the low-level, it is possible to achieve the high-level integration of desired style. For example, the shared database approach can be achieved by providing proper components dealing with the aspects of data retrieval (e.g., communication with the database) and conversion (e.g. object-relationship mapping of specific tables to a specific object model).

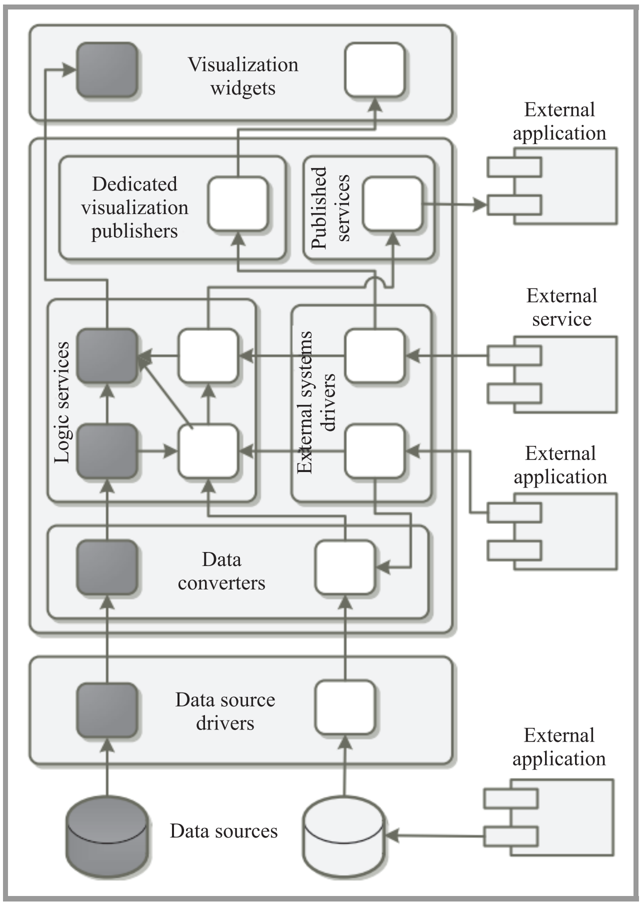


Fig. 6. Component-driven integration architecture.

In Fig. 6 the proposed architecture is presented. The architecture is driven by the component paradigm. The architecture consists of the following types of elements:

Externals. They include all the external entities that operate within the architecture such as available data sources, services and systems.

Drivers. They are specific elements of the architecture that enable the low-level integration of the externals in the whole architecture. Drivers are responsible for handling communication with the externals and providing a model (data or service based) for other architectural elements that need to communicate with the externals. A JDBC driver or specific web service proxy are the examples of drivers.

Services. These elements are responsible for data processing and publishing to other elements. Processing includes data conversion, filtering (transformers) and execution of domain specific operations and algorithms (functional services). Also, there is a group of publishing services for both external systems (such as web services) and the visualization elements which can be utilized to prepare data for visualization purposes, in this way, improving the visualization performance.

Visualization elements. They are responsible for visualization of the data and processing results. An example of a visualization element, a graph-based view of the data can be given.

The layout of the elements and scheme of their dependencies presented in Fig. 6 are illustrative only.
There are two significant aspects of the presented architecture to be emphasized. First, all of the architectural elements (except externals of course) are software components, and thus are represented by identical symbols. Being components, they can be attributed with some contracts based on which they can be assembled and communicate with each other. This is represented by arrows showing how the communication and data can flow between these components. This shows how the whole integrated system can be extended with the support of new data source or service. In that case, a specific component or components (driver at least, maybe some transformer) need to be provided and linked with other existing components to provide new features.
The second aspect is how linking the components can provide new features. Obviously, a feature (in terms of user functionality) is usually built around several cooperating components. Let's take a closer look on the selected components which are distinguished in Fig. 6 by the dark colour. There are 6 interlinked components: starting from a data source, through its driver, data transformer, functional services with a visualisation component at the top. These components clearly form a kind of processing sequence. This sequence is a flow of certain operations performed on a given data. Therefore, it can be treated as a dataflow. One can notice that the operation of the whole architecture is based on various dataflows. The concept of dataflow is the core mechanism of the architecture and reflects the analytical process presented in Fig. 1. Next section describes the concept in more details.

5. Realization Assumptions

Having overviewed the architecture concept, the initial technological assumptions can be made.
When the technological assumptions are made there are always several aspects and levels on which the choice of proper technologies must be concerned. Here, the following aspects are taken under consideration: modularity, data persistence, communication, graphical user interface, semantic integrity.

5.1. Modularity

The presented concept emphasizes the system modularity which can be realized by a component-based approach. There are several available component frameworks available from simple and efficient PicoContainer [7], Guice [8] to complex and advanced ones such as Spring framework [9] and Eclipse RCP [10], which provide also a number of other advanced features.

5.2. Data Persistence

While the proposed architecture assumes data retrieval from external data sources, the integration infrastructure also requires internal persistence mechanism, for example for caching and versioning purposes. This makes the persistence a crucial aspect in which the performance is very important for efficient processing of massive quantities of data. Also, an important aspect is the flexibility which would allow for handling completely new data models without corrupting the existing data. For this reason, it is assumed that the persistence mechanism can be realized through ORM paradigm [11].

5.3. Communication

There are several methods of communication that might be required in the discussed architecture. In the simplest case, a component-based framework can be sufficient platform for plugin-based communication. However, in more distributed approach, in particular when services integration is considered, network-oriented technologies are required. Here the Java RMI [12] or SOA [13] concept and its implementations can be utilized such as Apache Geronimo [14] or Microsoft's Windows Communication Foundation WCF [15].

5.4. Graphical User Interface

The choice of proper GUI technology is always a difficult task as it strongly depends on the user requirements and preferences, which are often very individualistic. The situation is also aggravated by the fact that currently at the market there is a number of libraries and technologies available.
Another aspect refers to the architecture type of the system: whether is it web-based application, server-client application (with heavy client) or desktop application for off-line work. Another issue is the target platform whether this is Windows, Linux or Mac.
With a high level of uncertainty concerning above aspects, it seems that the following assumptions need to be taken:

- It is preferable to use a technology which is portable. With a use of web-based architecture this requirement is of course easier to fulfill than for desktop application. As for example, solution J2EE technologies with Ajax support (such as RichFaces [16] library) might be pointed out.

- It is preferable to use a technology which is flexible with respect to the way of execution (web application or desktop). Currently, this assumption is the latest trend realized by such technologies as Windows Presentation Foundation WPF [17] and Silverlight [18], Adobe AIR [19] or JavaFx [20].

A careful look should be given to WPF library as it is aimed at clear presentation of the data transformations that may be performed in the system.

5.5. Semantic Integrity

To achieve the semantic integrity between various models emerging from different data sources the ontology paradigm might be used. As for technological choices, one of the option can be Web Ontology Language (OWL) and Resource Description Framework (RDF) graphs [21].

6. Selected Implementation Aspects

To validate the proposed architecture, a prototype implementation is realized. For experimental purposes, a simple integration scenario is proposed. The goal of the scenario is to integrate two analytical applications. The first one is LINK tool [5] which allows for importing, preprocessing and visualising data coming from file-based sources. The second application is Mammoth [22] which offers specialized pattern recognition and discovery functionalities.
Both applications have their strong an weak points, however when utilized together they can form an advanced analytical platform. The integration scenario assumes that LINK application can be used as data import, preprocessing and visualization tool, while Mammoth can provide routines for finding the patterns in the imported data, which can be visualized in a form of graph. To make the integration more difficult, the applications are realized in different technologies: LINK is realized in Java technologies while Mammoth is developed in .NET technologies.

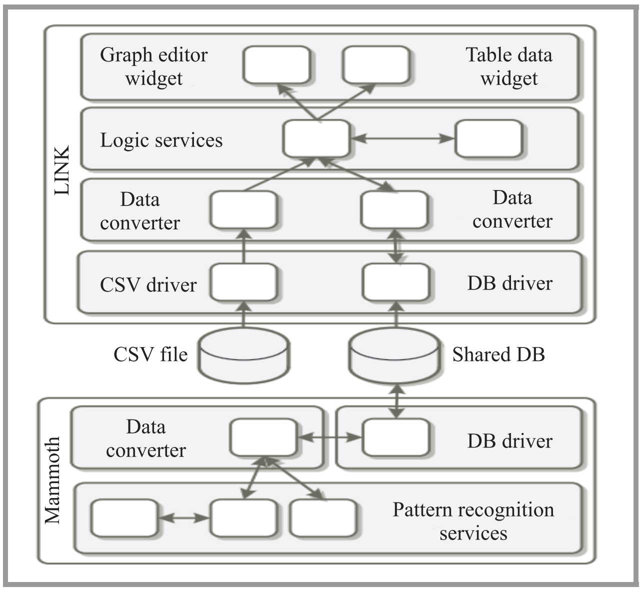


Fig. 7. Architecture overview of the prototype integration.

Figure 7 illustrates how the proposed component-drive architecture is applied to the discussed scenario. First thing to be noticed is that the chosen integration style is the shared database approach. There is also another data source which is a CSV file. The idea is that LINK provides dedicated components for reading data coming from CSV files and preprocessing the data. Once the data is ready, it is put into the shared database and can be visualized in form of tables or visual graphs.
In order to perform an advanced analysis, user can take advantage of the features of Mammoth which connects to the shared database using dedicated components. After the data is converted to a proper format for the available algorithms, it is processed and the results (found patterns) are written back to the database, from which, the found patterns can be read, converted and visualized in a form of graph.
One can notice that the rule of operation which is described in the above paragraph is a plain data-flow. This flow is illustrated in greater detailed in Fig. 8.
As for the technological choices, the shared database was realized using SQLite. As for LINK application it utilizes Eclipse RCP component platform and dedicated ORM layer based on JDBC. The visualization was realized with SWT and GEF libraries [5]. As for Mammoth application, it utilizes MEF as component platform and ADO.NET for database integration. The logic services were realized using various algorithm models for pattern recognition [22].
This prototype implementation shows that by utilization a component-driven approached, it is possible to integrate existing systems and applications and provide analyst with the more advanced environment that consists of multi-data sources and applications, as presented in the conceptual architectural scheme in Fig. 6.

7. Conclusions and Further Work

In this paper, a component-driven architecture is presented. This architecture is designed as a solution to the integration problem which occurs during development of maintainable software support for criminal analyst. To show how this architecture can be realized, the paper discusses the technological assumptions and provides selected details from the prototype implementation, which follows one of the discussed integration styles.
Further work should proceed into two following directions. The first one will be examining other integration styles. The second one is experimenting with larger integration examples that include more heterogeneous technologies and data models.

Acknowledgements

The work described in this paper was partially supported by The European Union by means of European Social Fund, PO KL Priority IV: Higher Education and Research, Activ-

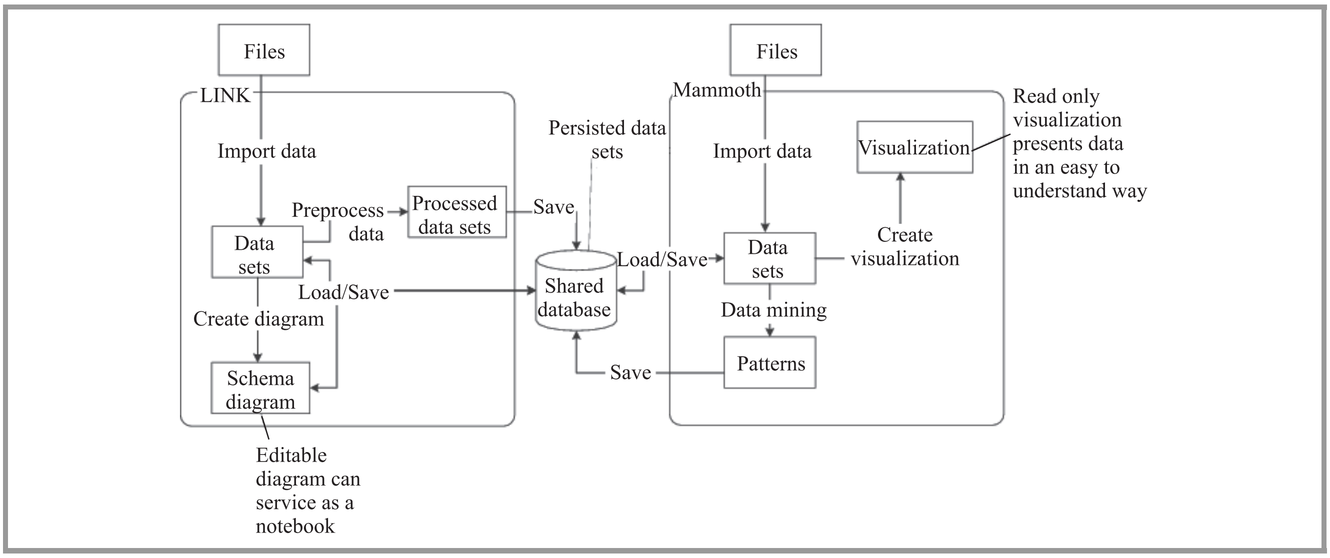


Fig. 8. Dataflow in the prototype integration.

ity 4.1: Improvement and Development of Didactic Potential of the University and Increasing Number of Students of the Faculties Crucial for the National Economy Based on Knowledge, Subactivity 4.1.1: Improvement of the Didactic Potential of the AGH University of Science and Technology "Human Assets", no. UDA-POKL.04.01.01-00-367/08-00.

The research leading to these results has received partial funding from the European Community's Seventh Framework Program – Project INDECT (FP7/2007-2013, grant agreement no. 218086).

References

[1] C. Szyperski, *Component Software: Beyond Object-Oriented Programming*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2002.

[2] Pei J. Han J. and Yin Yiwen, *Mining Frequent Patterns without Candidate Generation*. ACM, USA, 2000.

[3] R. A. Hanneman and M. Riddle, *Introduction to Social Network Methods*. University of California Press, 2005.

[4] A. Kirschner, "Overview of common social network analysis software platforms", Tech. Rep., Monitor Group, San Francisco, 2008.

[5] R. Dębski, M. Kisiel-Dorohinicki, T. Milos, and K. Pietak, "Link – a decision-support system for criminal analysis", in *Proc. IEEE Int. Conf. Multim. Commun. Serv. Secur. MCSS 2010*, J. Danda, J. Derkacz, and A. Głowacz, Eds., Kraków, Poland, 2010, pp. 110–116.

[6] G. Hohpe and B. Woolf, *Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions*. Addison-Wesley Professional, 2003.

[7] P. Hammant, K. Pribluda, and J. Schaible, "Picocontainer: a highly embeddable, full-service, inversion of control (ioc) container for components honor the dependency injection pattern" [Online]. Available: http://www.picocontainer.org/

[8] R. Vanbrabant, *Google Guice: Agile Lightweight Dependency Injection Framework*. Apress, 2008.

[9] C. Walls, *Modular Java: Creating Flexible Applications with Osgi and Spring*. Pragmatic Bookshelf, 2009.

[10] C. Aniszczyk J. McAffer, J. M. Lemieux, *Eclipse Rich Client Platform*. Addison-Wesley Professional, 2010.

[11] T. Halpin and T. Morgan, *Information Modeling and Relational Databases*. Morgan Kaufmann, 2008.

[12] W. Grosso, *Java RMI*. O'Reilly Media, 2001.

[13] T. Erl, *Service-Oriented Architecture (SOA): Concepts, Technology, and Design*. Prentice Hall, 2005.

[14] K. Kumar, *Pro Apache Geronimo*. Apress, 2006.

[15] J. Lowy, *Programming WCF Services*. O'Reilly Media, 2007.

[16] M. Katz, *Practical RichFaces*. Apress, 2008.

[17] A. Nathan, *Windows Presentation Foundation Unleashed*. Sams, 2006.

[18] M. MacDonald, *Pro Silverlight 3 in C#*. Apress, 2009.

[19] B. Gorton, R. Taylor, and J. Yamada, *Adobe AIR Bible*. Wiley, 2008.

[20] S. Morris, *JavaFX in Action*. Manning Publications, 2009.

[21] D. Allemang and J. Hendler, *Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL*. Morgan Kaufmann, 2008.

[22] P. Włodek, A. Świerczek, and B. Śnieżyński, "Pattern searching and visualization supporting criminal analysis", in *Proc. IEEE Int. Conf. Multim. Commun. Serv. Secur. MCSS 2010*, J. Danda, J. Derkacz, and A. Głowacz, Eds., Kraków, Poland, 2010, pp. 212–218.

Jacek Dajda received his Ph.D. degree in Computer Science from AGH University of Science and Technology in 2008. At present, he works as Assistant Professor at the Department of Computer Science of AGH-UST. His research interests involve software engineering with an emphasis on software architectures and frameworks.

E-mail: dajda@agh.edu.pl
Deptartment of Computer Science
AGH University of Science and Technology
Mickiewicza Av. 30
30-059 Kraków, Poland

Roman Dębski, **Aleksander Byrski**, and **Marek Kisiel-Dorohinicki** – for biographies, see this issue, p. 22.

Paper

Recognizing Sets in Evolutionary Multiobjective Optimization

Ewa Gajda-Zagórska

AGH University of Science and Technology, Faculty of Electrical Engineering, Automatics, IT and Electronics, Krakow, Poland

Abstract—Among Evolutionary Multiobjective Optimization Algorithms (EMOA) there are many which find only Pareto-optimal solutions. These may not be enough in case of multi-modal problems and non-connected Pareto fronts, where more information about the shape of the landscape is required. We propose a Multiobjective Clustered Evolutionary Strategy (MCES) which combines a hierarchic genetic algorithm consisting of multiple populations with EMOA rank selection. In the next stage, the genetic sample is clustered to recognize regions with high density of individuals. These regions are occupied by solutions from the neighborhood of the Pareto set. We discuss genetic algorithms with heuristic and the concept of well-tuning which allows for theoretical verification of the presented strategy. Numerical results begin with one example of clustering in a single-objective benchmark problem. Afterwards, we give an illustration of the EMOA rank selection in a simple two-criteria minimization problem and provide results of the simulation of MCES for multimodal, multi-connected example. The strategy copes with multimodal problems without losing local solutions and gives better insight into the shape of the evolutionary landscape. What is more, the stability of solutions in MCES may be analyzed analytically.

Keywords— basin of attraction, clustering, genetic algorithm, multiobjective optimization.

1. Introduction

The aim of this paper is to present new algorithmic methods for recognizing sets and separating neighbourhoods of the Pareto sets in multiobjective problems (Multiobjective Clustered Evolutionary Strategy, MCES). We propose theoretical and experimental verification of the presented strategy.

Presented algorithmics allows to interpret the neighbourhoods of the Pareto sets like basins of attraction of the sought solutions defined for single-objective optimization problems. It also helps to separate groups of solutions when the Pareto set is non-connected. What is more, recognizing sets in multiobjective problems provides better insight into understanding the properties of the problem and the shape of the search landscape which can be helpful when further postprocessing is required (e.g., engineering problems). Another important advantage of MCES is the possibility to reduce the number of starting points for local search methods to the number of sets found. This is crucial for many-objective functions which often have an infinite number of optimal solutions. Finally, we mention difficult multiphysics inverse problems which are extremely costly and hard to solve (see e.g. [1]).

We will focus on the idea of recognizing sets by clustering dense regions. Whereas in many papers (see i.e. [2], [3], [4]) a genetic algorithm is used as a help tool in clustering, we consider a combination of the two methods in the opposite way. Genetic algorithm here is used to provide a clustering method with the input data set. The advantages of clustering in single-objective genetic algorithms were studied by Schaefer, Adamska and Telega (CGS, see i.e. [5], [6]; well-tuning, see [7]). For other examples of two-phase global optimization strategies see [8] and [9]. Separation and estimation of the number of basins of attraction was performed by Stoean, Preuss, Stoean and Dumitrescu in [10] and in [11].

There are multiple algorithms that solve multiobjective optimization problems. The class of stochastic algorithms which approximate the Pareto set is called Evolutionary Multiobjective Optimization Algorithms (EMOA or MOEA). Usually, an EMOA aims at finding a set of Pareto-optimal solutions which may not give enough information in some cases, for example, in problems with non-connected Pareto fronts. It is difficult to extract knowledge about stability of solutions and how small perturbations affect domination among solutions from the existing algorithms. In our approach, solutions from the neighborhood of the Pareto set are detected and may be analyzed with regard to stability. For an example of analysis of stability of Pareto-optimal solutions, refer to [12]. Several examples of EMOA are presented below (for comparison see e.g. [13]).

The first method based on calculating an individual's fitness according to Pareto dominance was suggested by Goldberg in [14]. Nondominated Sorting Genetic Algorithm (NSGA) was implemented e.g. by Srinivas and Deb [15]. The selection pressure in NSGA was achieved by giving ranks determining fitness values in an iterative way: nondominated solutions are assigned rank one and temporarily removed from the population. New nondominated solutions are given rank two and so forth.

Fonseca and Fleming in [16] proposed a Pareto-based selection (*FFGA*), where an individual's rank equals the number of solutions by which it is dominated. We will refer to this type of selection later on.

In the third presented method, called Strength Pareto Evolutionary Algorithm (SPEA, see [17]), selection pressure is obtained by using an external set (archive) into which all nondominated solutions are copied in each iteration. Ranks of solutions are calculated basing on strength values of individuals stored externally. SPEA was later improved and introduced as SPEA2 in [18].

The next EMOA, by using the hypervolume measure (see e.g. [19]), maintains selection pressure as well as good distribution on the Pareto front. Hypervolume measure or S-metric corresponds to the size of dominated space [17]. Individuals are rated according to their contribution to the dominated hypervolume of the current population, therefore ranks are not based on relations between pairs of individuals but on relation between an individual and the whole population.

Pareto sets and fronts in multiobjective problems were investigated i.e. by Preuss, Naujoks and Rudolph in [20].

1.1. Preliminaries

We focus on global minimization problems with continuous objective functions of the form $\Phi : D \to \mathbb{R}$, $D \subset \mathbb{R}^n$, $0 \leq \Phi(x) \leq M < +\infty$, $\forall x \in D$, where D is the set of admissible solutions.

In the multiobjective optimization, we are given $k \geq 2$ objective functions

$$f_i : U \to [0,M] \subset \mathbb{R},\ M < +\infty,\ i \in \{1,\ldots,k\} \qquad (1)$$

defined over some search space U, which might be implicitly defined by constraints. We assume the search space U to be finite $\#U = r < +\infty$ and that all objectives shall be minimized. Therefore we are interested in solving

$$\min\left\{f(p) = (f_1(p),\ldots,f_k(p))^T \mid p \in U\right\}. \qquad (2)$$

Definition 1: (Pareto dominance) For any pair $(p,q) \in U \times U$, p is said to dominate q, denoted as $p \succ q$, if and only if

$$f(p) \leq f(q) \text{ and } \exists_{i=1,\ldots,k}\ f_i(p) \neq f_i(q). \qquad (3)$$

One of the possible ways to solve Eq. (2) is to find or approximate the *Pareto set* $\mathscr{P}$ being the set of non-dominated elements from U and its image $f(\mathscr{P}) \subset [0,M]^k$ called the *Pareto front*.

2. Strategy

The idea of the proposed strategy MCES of detecting neighborhoods of the Pareto sets consists of combining a genetic algorithm with a clustering method.

Among many GA we would like to extinguish those which may provide best samples for clustering. The most important property, which is held, e.g., by Simple Genetic Algorithm (SGA, for details refer to Subsection 3.3), is high selection pressure to obtain solutions in the neighborhoods of extrema. The second property is maintaining global search during computations. In case of single-population algorithms (like SGA), early convergence may eliminate global search. Such a behavior may result in losing information about parts of the Pareto front, as well as not recognizing local Pareto fronts. Therefore we propose to use an algorithm having both high selection pressure and globality, called Hierarchic Genetic Strategy (HGS, see [21]).

2.1. Genetic Engine

Hierarchic Genetic Strategy is an algorithm which produces a tree-structured set of concurrent evolutionary processes (see Fig. 1). The structure changes dynamically and the depth of the HGS tree is bounded by $m < +\infty$. In the simplest form of HGS, each process' evolution is governed by SGA.

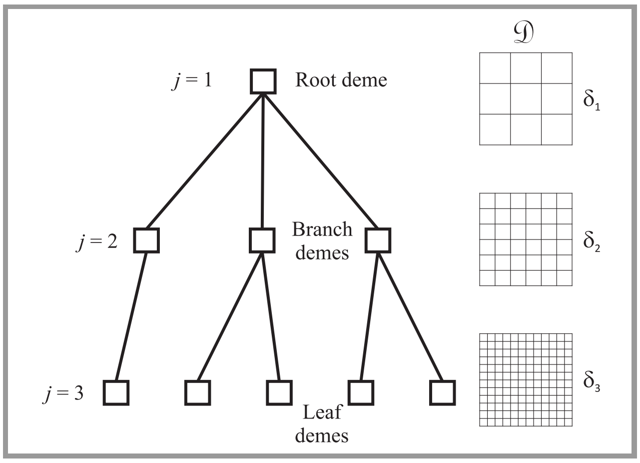


Fig. 1. HGS tree and corresponding two-dimensional meshes, $m = 3$.

HGS starts with a single root deme performing chaotic search with low accuracy. After a constant number of genetic epochs K called the *metaepoch* the root deme sprouts child-demes in the promising regions of the evolutionary landscape surrounding the best fitted individuals distinguished from the parental deme. Child-demes perform more local search with higher accuracy. The evolution in existing demes continues in the second metaepoch, after which new demes are sprouted. Demes of order m (leaves) perform local and most accurate search. The algorithm continues until the global stop condition is reached.

HGS implements two mechanisms that prevent redundancy of the search. The first one, called *conditional sprouting*, allows new demes to be sprouted only in regions which are not explored by sibling-demes (demes sprouted by the same parent). The second mechanism, called *branch reduction*, reduces demes of the same order that perform search in the common landscape region or in the regions already explored.

Different search accuracies are obtained by various encoding precisions and by manipulating the length of binary genotypes in demes at different levels. The root utilizes the shortest genotypes, while the leaves utilize the longest ones. To obtain search coherency for demes of different orders, a special kind of hierarchical nested encoding is used. Firstly, the densest mesh of phenotypes in $\mathscr{D}$ for the demes of the m-th order is defined. Afterwards, the meshes for lower order demes are recursively defined by selecting some nodes from the previous ones. The maximum diameter of the mesh δ_j associated with the demes of the order j determines the search accuracy at this level of the HGS tree (see Fig. 1). The mesh parameters satisfy $\delta_m < \ldots < \delta_1$.

Selection pressure is tightly connected with the probability of sampling measure in central parts of basins of attraction. The latter was formally proved for HGS in [21]. The theorem follows that, with certain assumptions, the sampling measures spanned by the sum of leaves in HGS are sufficiently close to the sampling measure associated with the unique fixed point of the genetic operator. Therefore, HGS is capable of detecting the same local extrema as SGA.

HGS is also more effective than SGA in finding multiple local extrema (see [22]). It consists of multiple populations which explore different areas of the search space. Even when considering only highest-order demes, the algorithm performs global search and, with a small number of individuals, can cover the whole domain.

2.2. Selection Scheme

In order to solve multiobjective optimization problems, evolution in each deme of HGS tree must be governed by an EMOA. Among EMOAs there are some selection schemes that fulfil the high selection pressure condition (several examples are described in Section 1). We will focus on selection scheme proposed by Fonseca and Fleming in [16].

Fonseca and Fleming proposed a Pareto-based selection (*FFGA*), where an individual's rank equals the number of solutions by which it is dominated. After sorting population according to the rank, fitness values are assigned to individuals by interpolating from the best (with the lowest rank) to the worst (with the highest rank) according to some function. Fitness of individuals with the same rank should be equal, so that all of them will be sampled at the same rate. We will refer to this type of selection later on, presenting a heuristic operator utilizing it.

In FFGA, selection pressure can be manipulated by using different validating functions $g \in C([0,1] \to [0,1])$ (see Subsection 3.5) which is a decreasing function transforming normalized ranks into probability distribution of the rank selection.

By applying a proper selection scheme, an EMOA converges to the Pareto front and solutions group around Pareto sets. When coupled with a multi-population strategy like HGS, an algorithm can provide a propitious sample for clustering.

2.3. Recognizing Sets

We do not restrict clustering to any particular method. Clustering here is applied to recognize regions with high density of individuals. In the presented strategy, individuals created in leaves concentrate on the neighborhoods of the Pareto-set which may be interpreted like basins of attraction of the sought solutions defined for single-objective optimization problems (see Subsection 3.1). We are aiming at finding a full-measure hull of the set of optimal solutions. Two problems appear while considering a method of recognizing sets by clustering the regions with high density of the sampling measure: What genetic algorithms should be used to provide a sample for clustering? Is it possible to verify such a strategy theoretically? The first question is already answered – we should take advantage of algorithms combining global search with high selection pressure, e.g., used in MCES. The second question will be addressed in the following chapter.

3. Theoretical Verification

To verify the strategy theoretically, we present several concepts. Firstly, we show the theorem of clustering to recognize the basins of attraction in single-objective optimization problems. Afterwards, we move on to Simple Genetic Algorithm and the definition of genetic algorithms with heuristic. Next, a heuristic for a particular class of EMOA is presented. We finish the theoretical part with algorithms preserving the property of being well-tuned to the problem.

3.1. Basins of Attraction

We begin with necessary definitions. Let $L(y) = \{x \in D : \Phi(x) \leq y\}$ and $\widehat{L}(y) = \{x \in D : \Phi(x) < y\}$ stand for two types of level sets of function Φ. $L_x(y)$ and $\widehat{L}_x(y)$ denote the connected parts of $L(y)$ and $\widehat{L}(y)$ (respectively) that contain x. For an arbitrary fixed x^* being a stationary point of function Φ let $\overline{y}(x^*) \in \mathbb{R}$ be defined as follows:

$$\overline{y}(x^*) = \begin{cases} \min \left\{ \begin{array}{l} y : \exists x^{**} \text{ isolated stationary point of } \Phi, \\ x^{**} \neq x^*, x \in L_{x^*}(y) \end{array} \right\} & \\ & \text{if } x^{**} \text{ exists} \\ \min_{x \in \partial D} \Phi(x) & \text{otherwise}, \end{cases} \tag{4}$$

where $x \in \partial D$ denotes points on the boundary of the domain.

Definition 2: [5] The basin of attraction $\mathscr{B}_{x^*}$ of a local minimizer x^* is the connected part of $\widehat{L}_{x^*}(\overline{y}(x^*))$ that contains x^*.

The process of set recognition begins with a genetic sample produced by a selected genetic algorithm. The sample is divided into clusters to discover groups in the data. Formally, *clusters* are non-empty, exclusive subsets $X_1, \ldots X_k$; $k \leq m$ which are the results of constructiong a partition of a discrete data set $X = x_1, \ldots x_m$. In the presented approach, a cluster is a discrete data set located in the basin of attraction of an isolated local minimizer x^+ of Φ.

After detecting clusters, we look for a cluster extension for each local minimizer x^+. A *cluster extension* is a closed set of positive measure which is included in $\mathscr{B}_{x^+}$ and contains x^+ in its interior. In this sense, cluster extensions approximate the basins of attraction and are located in their central parts.

Cluster extensions detection has several advantages. It allows detection and approximation of central parts of basins of attraction thus helps to determine groups of points from which local search may be started. The desired situation is to separate local extrema to reduce the number of local searches to one in each basin of attraction of a local extremum.

What is more, the combination of genetic algorithms with clustering methods provides the possibility to analyze the stability of minimizers. Cluster extensions recognition is also useful in sequential niching strategy, to deteriorate fitness. Basins of attraction can be recognized and separated, which prevents repeated search of depressed regions of the space and repeated convergence to the same solutions. Therefore, computation time can be reduced.

The importance of set detection is even more clear in case of multiobjective optimization where we seek for a full-measure hull of Pareto-optimal solutions. This set may be interpreted as basin of attraction defined for local minimizers.

3.2. Basic Theorem of Genetic Algorithms

In this paper we consider genetic algorithms, from which the simplest operate on a single population being the multiset $P=(U,\eta)$ of the search space members called *individuals*, while U is called now *genetic universum*. A genetic universum is denoted by Ω when it is composed of all binary strings of the finite, prescribed constant length $l \in \mathbb{N}$. In this case $\Omega=\{(a_0,a_1,\ldots,a_{l-1})\,;\, a_i\in\{0,1\}, i=0,1,\ldots,l-1\}$.

The occurrence function $\eta : U \rightarrow \mathbb{Z}_+ \cup \{0\}$ returns $\eta(i)$ which is the number of individuals with the genotype $i \in U$. The population cardinality is denoted by μ and $\mu = \sum_{i\in U} \eta(i) < +\infty$.

The algorithm consists in producing a sequence of populations $\{P^t\}$ in the consecutive *genetic epochs* $t = 1,2,\ldots$ starting from the population P^0 uniformly sampled from U. The mixing and selection operations depend on the algorithm. In particular, in case of MOEA the latter is often performed with respect to the Pareto–dominance relation (see e.g. [23]).

Each finite population represented as the multiset $P = (U,\eta)$ may be identified with its frequency vector $x = \{\frac{1}{\mu}\ \eta(p)\}, p \in U$ and all such vectors belong to the finite subset X_μ of the Vose simplex (see e.g. [9])

$$\Lambda^r = \Big\{x = \{x_p\}; 0 \leq x_p \leq 1, p \in U, \sum_{p\in U} x_p = 1\Big\}. \quad (5)$$

3.3. Simple Genetic Algorithm

Simple Genetic Algorithm (introduced by Vose in [24]) applies to optimization problems with one fitness function $f : \Omega \rightarrow [0,M], M < +\infty$. It is a method to transform a population P_t to the next epoch population P_{t+1}. Both populations are multisets of binary strings from the binary genetic universum Ω of the final cardinality $r < +\infty$. Selection of two individuals x,y from population P_t is performed by multiple sampling in proportional roulette selection. An individual added to the next epoch population P_{t+1} is produced from x and y with the mixing operation (see below). Creation of new individuals by selection and mixing is performed until P_{t+1} contains μ elements.

The proportional selection operator $F : \Lambda^r \rightarrow \Lambda^r$ is a mapping

$$F(x) = \frac{diag(f)x}{(f,x)}, \quad (6)$$

where the fitness function f is represented by the vector of its values $f\ (f_1, f_2,\ldots, f_r) \in \mathbb{R}^r; f_p = f(p), p \in U$ and $diag(f)$ denotes the $r \times r$ diagonal matrix with the diagonal f.

The mixing operator $M \in C^1(\Lambda^r \rightarrow \Lambda^r)$ introduced by Vose expresses the binary mutation and positional crossover

$$M(x)_p = (\sigma_p x)^T \mathbf{M} \sigma_p x,\ \ \forall\, x \in \Lambda^r,\ p \in U\,, \quad (7)$$

where σ_p stands for the $r \times r$ dimension permutation matrix with the entries $(\sigma_p)_{q,k} = [q \oplus k = p],\ p,q,k \in U$. The entries $\mathbf{M}_{p,q}$ of the symmetric $r \times r$ matrix $\mathbf{M}$ express the probability of obtaining the genotype being the string of zeros from the parents $p,q \in U$ by crossover and mutation.

3.4. Algorithms with Heuristic

An important group of algorithms which properties can be theoretically verified are genetic algorithms that admit a heuristic operator. Such algorithms will be called genetic algorithms with heuristic.

Definition 3: The mapping $\mathscr{H} \in C(\Lambda^r \rightarrow \Lambda^r)$ will be called the heuristic of the particular class of genetic algorithms if:

1. $\mathscr{H}(x)$ is the expected population in the epoch that immediately follows the epoch in which the population vector $x \in \Lambda^r$ appeared,
2. H is the evolutionary law of the abstract, deterministic, infinite population algorithm (we assume that it exists in the considered class). In other words, the infinite population algorithm is the dynamic system that starts from a particular initial population $x^0 \in \Lambda^r$ and then passes consecutively by $\mathscr{H}(x^0), \mathscr{H}^2(x^0), \mathscr{H}^3(x^0), \ldots$.
3. Each coordinate $(\mathscr{H}(x))_p$ is equal to the sampling probability of the individual with the genotype $p \in U$ in the epoch that immediately follows the epoch in which the population $x \in \Lambda^r$ appears.

The heuristic operator (also called the genetic operator) for SGA is a mapping $H : \Lambda^r \rightarrow \Lambda^r$ composed of selection and mixing

$$H = M \circ F. \quad (8)$$

SGA is one of a few instances of genetic algorithms for which the probability distribution of sampling the next epoch population can be delivered explicitly (see [24]).

3.5. EMOA Heuristic

The second example of a genetic algorithm with heuristic pertains to multiobjective optimization. It was introduced in [25].
Selection operator in the presented algorithm was inspired by the Pareto-based ranking procedure FFGA described in Subsection 2.2.
Let us start with the definition of the *binary Pareto dominance matrix*

$$\Xi \in \{0,1\}^r \times \{0,1\}^r; \;\; \Xi_{p,q} = \begin{cases} 1 & \text{if } q \succ p \\ 0 & \text{otherwise.} \end{cases}, \;\; \forall\, p,q \in U, \tag{9}$$

which characterizes the Pareto dominance relation among the genotypes from U for the particular multiobjective optimization Eq. (2). The p-th entry of the vector $(\Xi\;\eta)$ represents the number of individuals which dominate the individual with the genotype p belonging to the population $P = (U,\eta)$ (e.g. $\eta(p) > 0$).
Next, we introduce function $\xi : \Lambda^r \to [0,1]^r$ of the form

$$\xi(x) = \Xi\; x\,. \tag{10}$$

The function is well defined for both finite and infinite populations. Its value $\xi(x)_p$ gives the rank of all individuals with the genotype $p \in U$ contained in the population P represented by its frequency vector x and in case of finite population of the cardinality $\mu < +\infty$ may be interpreted as the relative number of individuals that dominate the individual with the genotype p.
It is also required to introduce two following functions. A decreasing *validating function* $g \in C([0,1] \to [0,1])$ is necessary to obtain the probability distribution of the rank selection. As a simple example of a function correlated with the rank-based fitness assignment method [16] we can take $g(\zeta) = 1 - \zeta$. The second function $G : [0,1]^r \to [0,1]^r$ such that $G(x)_p = g(x_p),\; p \in U$ is introduced for technical purposes.
The probability of selecting the individual $p \in U$ from the current EMOA population P represented by the vector $x \in \Lambda^r$ equals to

$$\Pr(p) = \frac{1}{x^T\; G(\xi(x))}\; g((\xi(x))_p)\; x_p\,. \tag{11}$$

Using previously introduced functions, we define the selection operator $F : \Lambda^r \to \Lambda^r$ for the EMOA rank selection

$$F(x) = \frac{1}{x^T\; G(\Xi\; x)}\; \mathrm{diag}(x)\; G(\Xi\; x)\,, \tag{12}$$

where $\mathrm{diag}(x)$ denotes the $r \times r$ diagonal matrix with the diagonal x.
In each EMOA epoch, selection is followed by genetic operations (e.g., mutation, crossover) which can be represented by the mixing operator $M \in C^1(\Lambda^r \to \Lambda^r)$. No specific restrictions for this mapping are imposed. For an exemplary mixing operator see Eq. (7).
Finally, similarly like in case of SGA, we compose selection and mixing to obtain a heuristic operator of the particular class of EMOA considered in this paper

$$\mathscr{H} = M \circ F. \tag{13}$$

If the mixing operator is strictly positive, e.g., $M(x)_p > 0$, $\forall x \in \Lambda^r$, $\forall p \in U$, then the algorithm possesses the asymptotic guarantee of success, e.g., it will reach the population which contain all points lying in the Pareto set after an infinite number of epochs.

Definition 4: We say that H is *focusing* if there exists a nonempty set of fixed points $\mathscr{K} \subset \Lambda^r$ of H that for all $x \in \Lambda^r$ the sequence $\{H^t(x)\}$ converges in Λ^r to $w \in \mathscr{K}$ for $t \to +\infty$.

Theorem 1: [25] Assuming that the heuristic $\mathscr{H}$ is focusing and the mixing operator is strictly positive, the sampling measure concentrates on the set of fixed points of $\mathscr{H}$ if $\mu \to +\infty$ and $t \to +\infty$.

The theorem (for details, refer to [25]) is an extension of a similar theory introduced by Vose for SGA and has great importance in verifying MCES. Applied rank selection causes the individuals to concentrate on the neighborhood of Pareto-optimal solutions and produces a sample ready to clustering.

3.6. Well-Tuning

For genetic algorithms with heuristics it is possible to introduce a condition which is connected with the property of the frequency of solutions included in some central parts of basins of attraction being significantly higher than in other parts (see e.g. [7]).

Definition 5: [7]
A particular class of SGA with heuristic H is *well-tuned* with respect to a finite set of local minimizers $\mathscr{W}$ if:

1. H is focusing and the set of its fixed points $\mathscr{K}$ is finite,
2. $\forall x^* \in \mathscr{W}\; \exists C(x^*)$ closed set in D such that $x^* \in C(x^*) \subset \mathscr{B}_{x^*}$, $meas(C(x^*)) > 0$ and

$$\rho_w(x) \geq threshold, x \in C(x^*) \tag{14}$$

$$\rho_w(x) < threshold, x \in D \setminus \bigcup_{x^* \in \mathscr{W}} C(x^*)\,, \tag{15}$$

where $w \in \mathscr{K}$ is a fixed point of H, ρ_w is a measure density over D corresponding to a population $w \in \Lambda^r$ and the positive constant threshold stands for the definition's parameter.

The parameter introduced by the above definition allows distinguishing whether the measure density induced by a limit population can be successfully used to separate local minimizers and to roughly locate them in the admissible set.
An important feature of algorithms well-tuned to the problem is that by increasing population size we get a higher chance of recognizing sets by cluster analysis methods.

Basing on the obtained results, we claim that Evolutionary Multiobjective Optimization Algorithm with heuristic is *well-tuned* if the fixed points of the heuristic correspond to densification of sampling measures in the neighborhoods of the Pareto set. Densification of sampling measure causes points to group around Pareto-optimal solutions where they can be recognized by clustering methods.
It was proved in [9] that if a heuristic H is focusing, the sampling measures of the algorithm converge to the measure given by the set of fixed points of H. Taking into consideration the presented EMOA heuristic with rank selection, we conclude that the level set of a particular density of the sampling measure for this selection will be the neighborhood of the Pareto set. Besides that, it is possible to asymptotically approximate that level set.

4. Experiments

We present three experimental examples. The first one refers to clustering genetic sample in a single-objective problem. The second shows rank selection in a simple multiobjective task. Finally, the third example is an application of the MCES to the benchmark problem.

4.1. Clustered Genetic Search in Single-Objective Problems

Clustering in single-objective problems was investigated by Schaefer, Adamska and Telega. The following example was presented in [5].
A two-dimensional test function

$$f(x,y) = sin(xy) + 1, \quad (x,y) \in [-3,3] \times [-3,3] \tag{16}$$

was selected to illustrate CGS abilities of coping with multimodal functions.
The objective is shown in Fig. 2. The multiple minima of the function constitute one-dimensional manifolds, which provide an additional difficulty. The authors used HGS as a genetic engine and compared two types of CGS (HC-CGS and DR-CGS, for details refer to [5]) which can be applied to solve the problem. In both cases, the algorithms found several cluster extensions which were recognized only by means of analysis of the density of individuals. They conclude that the recognized sets can be treated as central parts of basins of attractors; starting from each point of a cluster extension at least one point of the same manifold may be reached.

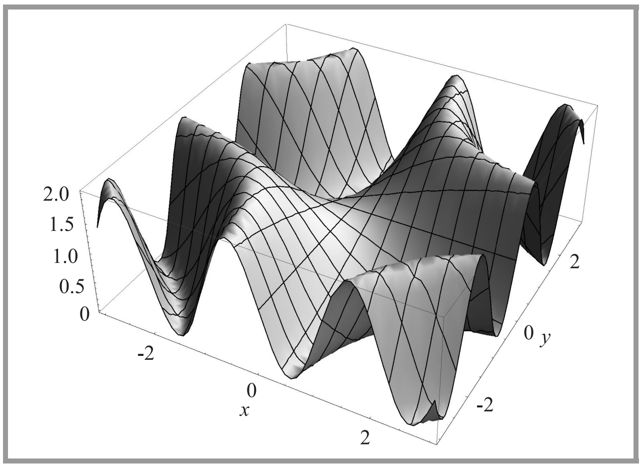


Fig. 2. The objective function f.

4.2. EMOA Rank Selection Example

The next example shows EMOA rank selection (see Eq. 12) in a two-criteria, two-dimensional minimization problem.

We take two simple objective functions with $(x,y) \in [0,4] \times [0,4]$:

$$f_1(x,y) = x + y \tag{17}$$

$$f_2(x,y) = (x-2)^2 + (y-2)^2. \tag{18}$$

We represent each individual as a binary code of length 12. The objective space is divided into a mesh of 2^{12} tiles and each tile has one representing individual corresponding to the centre of the tile. We consider a whole set of individuals and begin with computing the values of the binary Pareto dominance matrix (see Eq. (9)). Next, for each individual, we calculate rank (the number of individuals dominating it) and normalize that value. Ranks are presented as a landscape in Fig. 3. One should notice, that ranks are calculated in discrete domain but in the plot are linked for visualization purposes. For the same reasons, we focus on solutions with $rank < 0.5$.

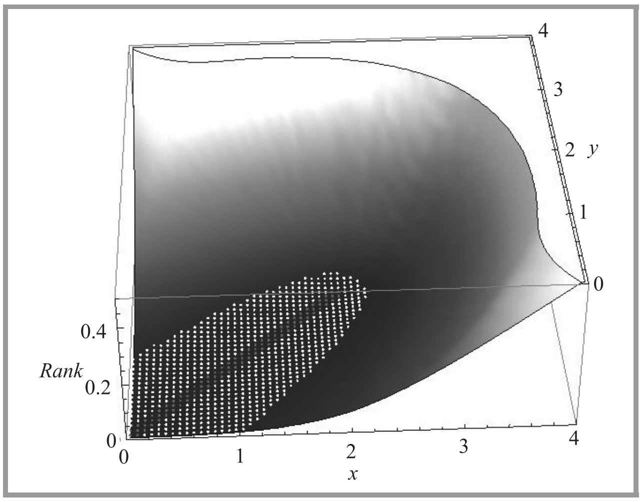


Fig. 3. Ranks of solutions in decision space. Dark grey points represent Pareto-optimal solutions, light grey points represent solutions close to optimal.

Additionally, Pareto-optimal solutions (with $rank = 0$) are marked dark grey and solutions being close to optimal ($rank < 0.01$) are marked light grey. Therefore, we may see a central part of the valley which may be interpreted analogically as the basin of attraction in single-objective problems.

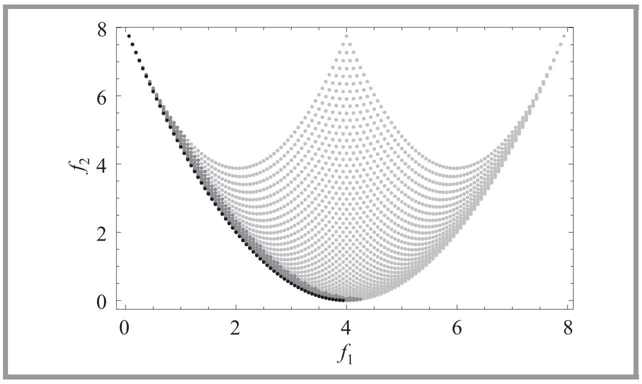


Fig. 4. Solutions in objective space. Black points represent the Pareto front, dark grey points represent solutions close to optimal.

In Fig. 4 we present solutions in the space of objectives. Pareto-optimal solutions (in this case – the Pareto front) are marked black and solutions being close to optimal ($rank < 0.01$) are marked dark grey. It is clear, that solutions from the level set of Pareto-optimal are located in the neighbourhood of the Pareto-front. What is more, concentrating of individuals on the set surrounding Pareto-optimal solutions may be used to construct a stop criterion for a particular class of EMOA.

4.3. Clustering in Multiobjective Case

As a third example we present results of a simulation of MCES combining HGS engine with EMOA rank selection and clustering.

For a case study we have chosen a two-criteria, two-dimensional minimization problem with the following objective functions:

$$f_1(x,y) = x \tag{19}$$

$$f_2(x,y) = g(y)\left(1 - \sqrt{\frac{x}{g(y)}} - \frac{x}{g(y)}\sin(10\pi x)\right), \tag{20}$$

where $g(y) = 1 + 9y$, $(x,y) \in [0,1] \times [0,1]$ (see Fig. 5).

The problem is quite difficult to solve because it is multimodal and its Pareto-optimal front consists of several non-connected parts.

As a genetic engine in the example we use a two-level HGS with rank selection presented in the paper. Root deme consists of 50 individuals and the mutation probability is 0.05. The stop condition is fulfilled when the root population finishes the 20th metaepoch. After each metaepoch leaves are sprouted in the best places found by root (around individuals with lowest ranks). Each leaf population consists of 10 individuals and the mutation probability is 0.005. We have limited leaf evolution to 5 metaepochs.

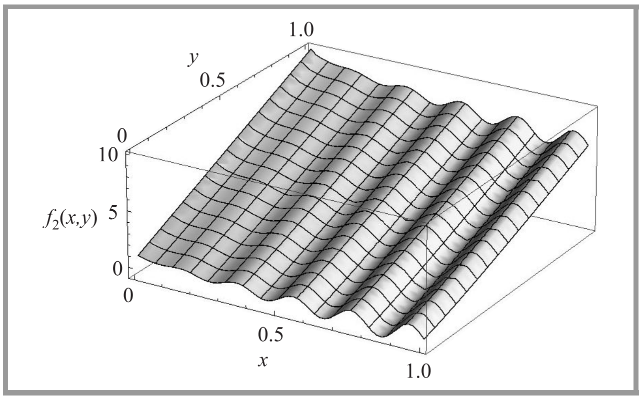


Fig. 5. Objective function f_2 (see Eq. (20)).

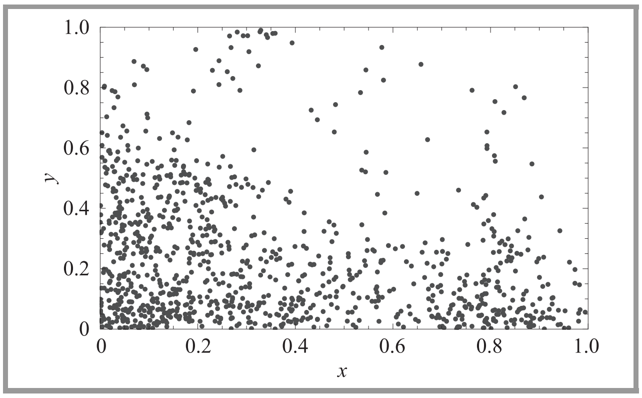


Fig. 6. Root individuals in the decision space.

In Fig. 6 we present all individuals created by root. The individuals are quite well-spread in the entire search space and group in the regions which contain solutions with low ranks. The same individuals are presented in Fig. 7 in the objective space. Recognized parts of the Pareto front are visible in the lower part of the plot.

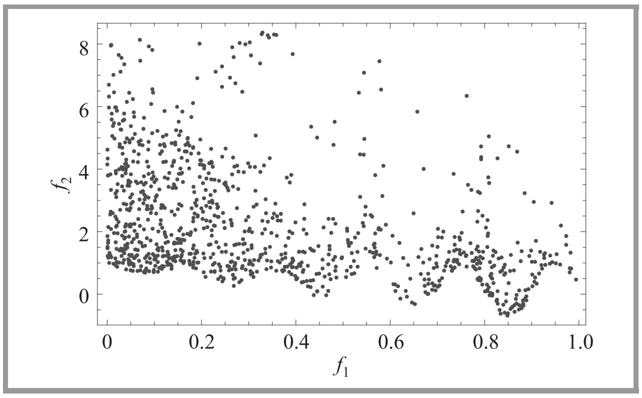


Fig. 7. Root individuals in the objective space.

Leaves continue exploration in most interesting parts of the landscape. Most of these regions are the neighborhoods of the Pareto-optimal sets (see Fig. 8).

Afterwards, the results of search in leaves are being clustered my k-medoids method (see, i.e., [26]). In the pre-

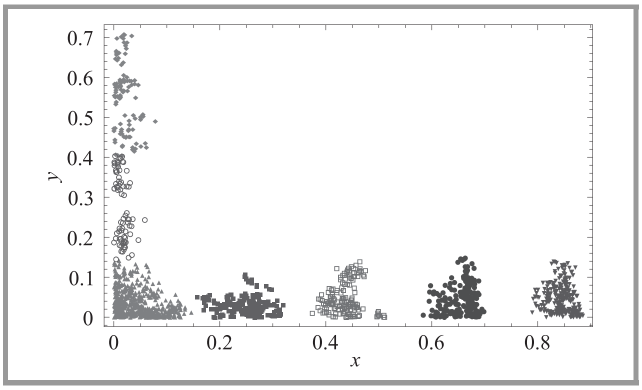


Fig. 8. Individuals created in leaves, decision space.

sented example problem, found clusters represent existing parts of Pareto front very well. Two upper clusters (Fig. 9) are the results of early sprouting in regions interesting at the beginning of computation in root whereas the remaining ones are exactly the solutions we were looking for.

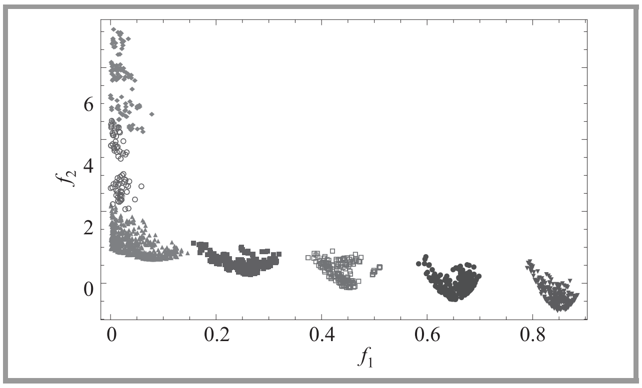


Fig. 9. Individuals created in leaves, objective space.

To conclude, in this chapter we presented examples which show the strategy in practice. It may be successfully applied to multimodal problems and gives a better insight into the shape of problem landscape. Clustering results of genetic search allows detecting basins of attractions of solutions in single-objective optimization tasks as well as analogical sets of individuals in neighborhoods of Pareto-optimal solutions in multiobjective case.

5. Conclusions and Future Research

- The presented strategy of solving a Pareto optimization problem gives additional knowledge about the shape of the evolutionary landscape. What is more, it copes with multimodal problems without losing local solutions.
- Set recognition allows for detecting central regions of the basins of attraction and, as a result, starting points for local search methods can be limited to one in each basin of attraction.
- MCES can be partially theoretically verified by using concepts of EMOA heuristic and well-tuning.
- We suppose that presented methods can be applied to solve multiobjective inverse problems in cooperation with hp-adaptive direct problem solving methods.
- In future papers, we plan to develop the theorem allowing for verification of the strategy and investigate the property of well-tuning of EMOA.

Acknowledgements

Author would like to thank EFS of POKL 4.1.1 (POKL.04.01.01-00-367/08-00) European Union programme for support.

References

[1] B. Barabasz, S. Migórski, R. Schaefer, and M. Paszyński, "Multi-deme, twin adaptive strategy hp-hgs", *Inverse Problems in Sci. Engin.*, vol. 19, no. 1, pp. 3–16, 2011.

[2] M. C. Cowgill, R. J. Harvey, and L. T. Watson, "A genetic algorithm approach to cluster analysis", *Comput. Mathem. Appl.*, vol. 37, no. 7, pp. 99–108, 1999.

[3] E. Hruschka and N. Ebecken, "A genetic algorithm for cluster analysis", *Intell. Data Analysis*, vol. 7, no. 15, pp. 15–25, 2003.

[4] U. Maulik and S. Bandyopadhyay, "Genetic algorithm-based clustering technique", *Pattern Recognition*, vol. 33, no. 9, pp. 1455–1465, 2000.

[5] R. Schaefer, K. Adamska, and H. Telega, "Clustered genetic search in continuous landscape exploration", *Engin. Appl. Artificial Intell.*, vol. 17, no. 4, pp. 407–416, 2004.

[6] K. Adamska, "Genetic clustering as a parallel algorithm for approximating basins of attraction", in *Parallel Processing and Applied Mathematics*, R. Wyrzykowski, J. Dongarra, M. Paprzycki, and J. Wasniewski, Eds., vol. 3019, LNCS. Berlin/Heidelberg: Springer, 2004, pp. 536–543.

[7] R. Schaefer and K. Adamska, "Well-tuned genetic algorithm and its advantage in detecting basins of attraction", in *Proc. 7th Conf. Evol. Algorithms Global Optimiz.*, Kazimierz Dolny, Poland, 2004, pp. 149–154.

[8] *Handbook of Global Optimization vol. 2*. P. M. Pardalos and H. E. Romeijn, Eds. Kluwer, 1995.
Kluwer, 1995.

[9] R. Schaefer and H. Telega, *Foundation of Global Genetic Optimization*. Springer, 2007.

[10] C. Stoean, M. Preuss, R. Stoean, and D. Dumitrescu, "Ea-powered basin number estimation by means of preservation and exploration", in *Parallel Problem Solving from Nature – PPSN X*, G. Rudolph, T. Jansen, S. M. Lucas, C. Poloni, and N. Beume, Eds., vol. 5199, LNCS, Berlin: Springer, 2008, pp. 569–578.

[11] C. Stoean R. Stoean, M. Preuss, "Approximating the number of attraction basins of a function by means of clustering and evolutionary algorithms", in *8th Int. Conf. Artif. Intelli. Digit. Commun. AIDC 2008*, *Research Notes in Artificial Intelligence and Digital Communications*, N. Tandareanu, Ed., Reprograph Press, 2008, pp. 171–180.

[12] V. A. Emelichev and E. E. Gurevsky, "On stability of a pareto-optimal solution under perturbations of the parameters for a multicriteria combinatorial partition problem" [Online]. Available: http://www.math.md/publications/csjm/issues/v16-n2/8827/

[13] E. Zitzler, "Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications", Ph.D. thesis, ETH Zurich, Switzerland, 1999.

[14] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learnings*. Reading, Massachusetts: Addison-Wesley, 1989.

[15] N. Srinivas and K. Deb, "Multiobjective optimization using nondominated sorting in genetic algorithms", *Evolutionary Comput.*, 1994.

[16] C. M. Fonseca and P. J. Fleming, Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. *Genetic Algorithms: Proceedings of the Fifth International Conference, pp. 416-423*, 1993.

[17] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength pareto evolutionary algorithm", *IEEE Trans. Evolutionary Comput.*, vol. 3, no. 4, pp. 257–271, 1999.

[18] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization", in *Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems (EUROGEN 2001)*, International Center for Numerical Methods in Engineering (CIMNE), 2002, pp. 95–100.

[19] M. Emmerich, N. Beume, and B. Naujoks, "An EMO algorithm using the hypervolume measure as selection criterion", in *Evolutionary Multi-Criterion Optimization: Third International Conference EMO 2005*, Springer, 2005, pp. 62–76.

[20] M. Preuss, B. Naujoks, and G. Rudolph, "Pareto set and EMOA behavior for simple multimodal multiobjective functions", in *Parallel Problem Solving from Nature – PPSN IX*, T. P. Runarsson, H.-G. Beyer, E. Burke, J. J. Merelo-Guervós, L. D. Whitley, and X. Yao, Eds., vol. 4193, LNCS, Berlin: Springer, 2006, pp. 513–522.

[21] J. Kolodziej, "Modelling hierarchical genetic strategy as a family of markov chains", in *Proc. Int. Conf. Parallel Proces. Applied Mathem.-Revised Papers*, Springer, 2002, pp. 595–598.

[22] R. Schaefer and J. Kolodziej, "Genetic search reinforced by the population hierarchy", in *Foundations of Genetic Algorithms 7*, R. Poli, K. A. De Jong, and J. E. Rowe, Eds., Morgan Kaufman, 2003, pp. 383–388.

[23] C. A. Coello Coello and G. B. Lamont, *Applications of Multi-objective Evolutionary Algorithms*. World Scientific, 2004.

[24] M. D. Vose, *The Simple Genetic Algorithm*. MIT Press, 1999.

[25] E. Gajda, R. Schaefer, and M. Smolka, "Evolutionary multiobjective optimization algorithm as a markov system", in *Parallel Problem Solving from Nature – PPSN XI*, Springer, 2010, pp. 617–626.

[26] H. Park and C. Jun, "A simple and fast algorithm for k-medoids clustering", *Expert Systems with Applications*, vol. 36, no. 2, part 2, pp. 3336–3341, 2009.

Ewa Gajda-Zagórska received the M.Sc. degree in Computer Science at Jagiellonian University, Kraków, Poland, in 2009. Since then, she has been a Ph.D. student of Computer Science at AGH University of Science and Technology, Kraków, Poland. Her research interests are evolutionary algorithms and multi-objective optimization.

E-mail: gajda@agh.edu.pl
AGH University of Science and Technology
Faculty of Electrical Engineering, Automatics,
IT and Electronics
A. Mickiewicza Av. 30
B-1 building
30-059 Krakow, Poland

Paper

Modeling and Simulation of a Vehicle Suspension with Variable Damping versus the Excitation Frequency

Claudiu Valentin Suciu[a], Tsubasa Tobiishi[b], and Ryouta Mouri[b]

[a] Department of Intelligent Mechanical Engineering, Fukuoka Institute of Technology, Fukuoka, Japan
[b] Graduate School of Engineering, Fukuoka Institute of Technology, Fukuoka, Japan

Abstract—In this work, three types of vehicle suspensions were considered and modeled as follows: oil damper mounted in parallel with a compression helical spring, for which a Kelvin-Voigt model, consisted of a dashpot and an elastic element connected in parallel is considered; colloidal damper without attached compression helical spring, for which a Maxwell model, consisted of a dashpot and an elastic element connected in series is considered; and colloidal damper mounted in parallel with a compression helical spring, for which a standard linear model, consisted of a Maxwell unit connected in parallel with an elastic element is considered. Firstly, the vibration transmissibility from the rough road to the vehicle's body for all these suspensions was determined under the constraint that damping varies versus the excitation frequency. Then, the optimal damping and stiffness ratios were decided in order to minimize the transmissibility of vibration from the rough pavement to the vehicle's body.

Keywords—Kelvin-Voigt-Maxwell models, optimal damping and stiffness, ride comfort, transfer function of the human body.

1. Introduction

Major sources of excitation in motor vehicles are the engine, transmission system, air-conditioning system, road and aerodynamic excitations. Thus, major structural resonances and their frequency ranges are [1]: rigid body vibrations of bouncing, pitching and rolling on suspension system and wheels (0.5–2 Hz), forced vibrations of the vehicle body due to the engine shake (11–17 Hz), bending and torsional vibration of the body as a whole (25–40 Hz), as well as ring mode vibration of the passenger compartment and bending vibration of the driveline (50–100 Hz). Although resonances connected to suspension systems and wheels are in the domain of 0.5–2 Hz, structural resonances of various systems in motor vehicles are going up to 100 Hz [1]–[3]. Road excitation frequency increases with the vehicle speed and decreases with the wavelength of the road roughness; excitation frequencies 0.1–0.5 Hz are important for evaluation of the motion sickness, and the domain of 0.5–100 Hz is recommended for evaluation of the ride comfort [4], [5].

In order to improve the vehicle's ride comfort, from a technical standpoint, the suspension designer has a single alternative: to minimize the transfer function of vibration from the rough road to the vehicle's body, over the entire concerned range of frequencies (0.1–100 Hz). Usual vehicle suspensions employ hydro-pneumatic absorbers (e.g., oil [2]–[7], colloidal [8]–[14] and air dampers [2]–[7]) mounted in parallel with compression helical springs. Although the damping coefficient of the vehicle suspension is changing versus the excitation frequency [10], conventional design method is based on simplified models that assume for constant damping and elastic properties [4]. For this reason, the optimal damping and stiffness ratio, to maximize the vehicle's ride comfort, cannot be accurately predicted [15], and also discrepancies between theoretical and experimental results can be observed [16].

In this work, three types of suspensions are considered and modeled as follows: oil damper mounted in parallel with a compression helical spring, for which a Kelvin-Voigt model, consisted of a dashpot and an elastic element connected in parallel is considered; colloidal damper without attached compression helical spring, for which a Maxwell model, consisted of a dashpot and an elastic element connected in series is considered; and colloidal damper mounted in parallel with a compression helical spring, for which a standard linear model, consisted of a Maxwell unit connected in parallel with an elastic element is considered. Firstly, the vibration transmissibility from the rough road to the vehicle's body for all these suspensions is determined under the constraint that damping varies versus the excitation frequency. Then, the optimal damping and stiffness ratios are decided in order to minimize the vibration transmissibility, i.e., to maximize the vehicle's ride comfort.

2. Methods to Estimate the Ride Comfort

Perception of the vehicle's ride comfort is different from one passenger to another, depending on its taste and physical constitution. However, the ride comfort of a certain vehicle can be evaluated based on the equivalent acceleration a_c which is proportionally depending on the root-mean-square of the weighted transfer function of vibration from the rough road to the vehicle's body [4], [5]:

$$a_c \propto \sqrt{\sum_i \left[|F(f_i)||H(f_i)|\right]^2}. \tag{1}$$

Discrete frequency values are taken in the range 0.1–100 Hz, as follows: $f_i = 0.1$, 0.125, 0.16, 0.2, 0.25, 0.315, 0.4, 0.5, 0.63, 0.8, 1, 1.25, 1.6, 2, 2.5, 3.15, 4, 5, 6.3, 8, 10, 12.5, 16, 20, 25, 31.5, 40, 50, 63, 80 and 100 Hz. The so-called filter or weighting function $F(f_i)$ represents the vibration transfer function of the human body. For vibration transmitted in vertical direction from seat to the vehicle's rider, according to the K-factor method, the filter can be taken as [4]:

$$F(f_i) = \begin{cases} 10^{(3f_i-15)/20}, & 0 \le f_i \le 4 \\ 10^{-3/20}, & 4 \le f_i \le 8 \\ 10^{(-0.75f_i+3)/20}, & f \ge 8 \end{cases} . \tag{2}$$

On the other hand, according to ISO 2631 method, the frequency weighting can be introduced as [5]:

$$F(f_i) = \Gamma(f_i)\Delta(f_i)\frac{7.875 f_i^2}{\sqrt{0.0256 + f_i^4}}\,\frac{10^4}{\sqrt{10^8 + f_i^4}}, \tag{3}$$

where the functions $\Gamma(f_i)$ and $\Delta(f_i)$ can be calculated as:

$$\Gamma(f_i) = \sqrt{\frac{f_i^2 + 156.25}{0.3969 f_i^4 + 32.21875 f_i^2 + 9689.94141}}, \tag{4}$$

and:

$$\Delta(f_i) = \sqrt{\frac{0.8281 f_i^4 - 3.68581 f_i^2 + 26.1262}{0.8281 f_i^4 - 7.36421 f_i^2 + 104.29465}}. \tag{5}$$

In order to estimate the transfer function of vibration from the rough road to the vehicle's body, an adequate model should be adopted. In general, a vehicle with four wheels can be modeled as a system with 6 degrees of freedom (full-vehicle model [4], [17]). However, when the frequency in vertical direction of the vehicle's body is below 2 Hz, it is possible to neglect the rolling and to assume that the left and right parts of the vehicle are identical (half-vehicle model [17], [18]). Moreover, experience has proven that even if the pitching movement is neglected (quarter-vehicle model [17]), the ride-comfort can be predicted quite accurately. Accordingly, in this theoretical work, a quarter-vehicle moving on a rough pavement is considered as a suitable model to estimate the transmissibility and comfort.

3. Models of the Considered Vehicle Suspensions

Three types of suspensions are considered and modeled as follows: oil damper mounted in parallel with compression helical spring (Fig. 1), colloidal damper without attached compression helical spring (Fig. 2), and colloidal damper mounted in parallel with compression helical spring (Fig. 3). In (a) parts of Figs. 1–3, models with two-degrees of freedom are considered, based on the following assumptions. Travel speed is constant; there is no frontal-rear and/or axial rolling of the vehicle's body; contact between tire and road is linear; finally, suspension and tires have linear characteristics.

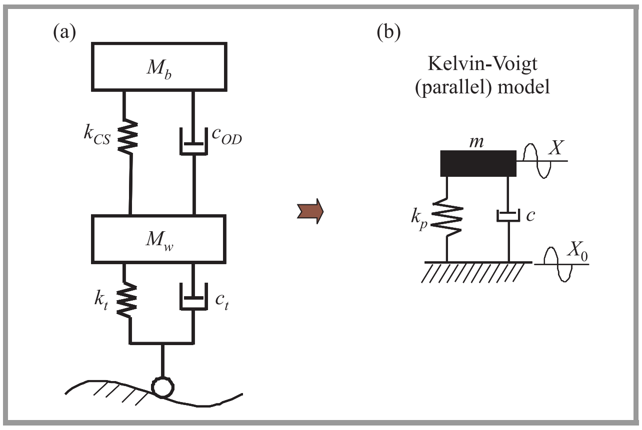


Fig. 1. Two-degrees of freedom (a) and one-degree of freedom (b) models for oil damper mounted in parallel with compression helical spring.

On the Figs. 1–3, M_b is the body (sprung) mass, M_w is the wheel (unsprung) mass, k_t is the tire spring constant, k_{CS} is the compression spring's constant, c_t is the tire damping coefficient, c_{OD} is the damping coefficient of the oil damper and c_{CD} is the damping coefficient of the colloidal damper. In the case of usual suspension (Fig. 1), compression spring provides the necessary restoring force to bring back the suspension to its initial position after a cycle of compression-extension.

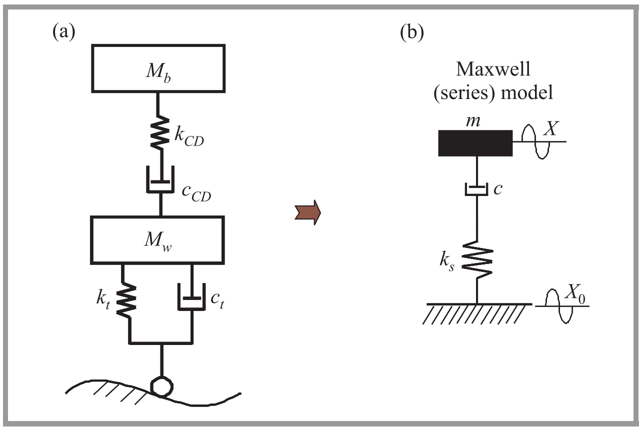


Fig. 2. Two-degrees of freedom (a) and one-degree of freedom (b) models for colloidal damper without compression helical spring mounted in parallel.

Energy of shock and vibration is mainly stored by the spring during the compression phase, and then, it is transferred to and dissipated by the oil damper during extension. On the other hand, colloidal damper is able to intrinsically provide the restoring force [15], [16] and the spring can be omitted (Fig. 2). Thus, colloidal damper is a machine element with a dual function, of absorber and spring of constant k_{CD}.

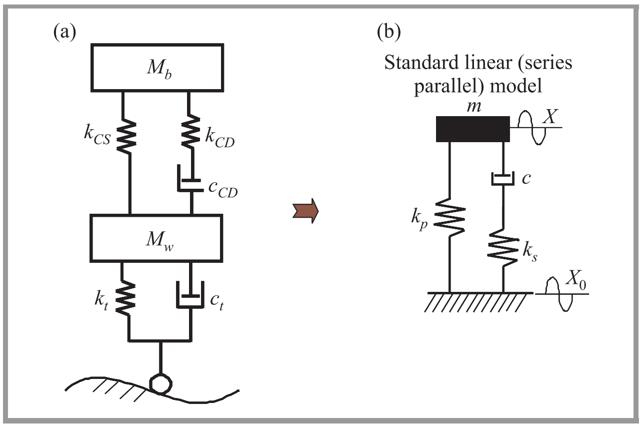


Fig. 3. Two-degrees of freedom (a) and one-degree of freedom (b) models for colloidal damper mounted in parallel with compression helical spring.

Additionally, colloidal damper is able to dissipate the energy of shock during its compression, this reducing the delay between excitation and response; since it has higher speed of reaction to excitation, one expects that the ride-comfort of the vehicle can be considerably improved.

For the model shown in Fig. 1, one observes two peaks on the graph of transmissibility of vibration from the rough road to the vehicle's body $|H|$ against the excitation frequency f, as follows: a first resonance peak at lower frequency f_n that approximately corresponds to the vehicle's body (sprung) mass, and a second resonance peak at higher frequency f_t that approximately corresponds to the wheel (unsprung) mass [19]:

$$f_n = \frac{\omega_n}{2\pi} \cong \frac{1}{2\pi}\sqrt{\frac{k_{CS}}{M_b}}, \quad f_t = \frac{\omega_t}{2\pi} \cong \frac{1}{2\pi}\sqrt{\frac{k_t}{M_w}}. \tag{6}$$

Generally, one observes two opposite requirements in the design of actual vehicle suspension (oil damper mounted in parallel with spring): large damping c_{OD} is desirable at lower frequency f_n to reduce the first resonant peak, but on the other hand, low damping c_{OD} is needed at higher frequency f_t to reduce the second resonant peak [19]. One observes from Eq. (1) that, in order to improve the vehicle's ride comfort, from a technical standpoint, the suspension designer has a single alternative: to minimize the transfer function of vibration from the rough road to the vehicle's body, over the entire concerned range of frequencies (0.1–100 Hz). One way to reduce damping in a passive manner at higher frequencies is to use a "relaxation damper", where the dashpot c_{OD} is replaced by a Maxwell unit, consisted of a dashpot, e.g., c_{CD} and a spring, e.g., k_{CD} mounted in series (Figs. 2 and 3). Since the peak at lower frequency f_n is critical, the model with two-degrees of freedom can be further simplified to a quarter-vehicle with one-degree of freedom ((b) parts of Figs. 1–3), by defining the equivalent mass m of the vehicle, the equivalent spring constant of the parallel k_p and serial k_s elastic elements, and the equivalent damping coefficient c of the dashpots as follows:

$$\begin{cases} m = M_b + M_w \\ \dfrac{1}{k_p} = \dfrac{1}{k_t} + \dfrac{1}{k_{CS}}\dfrac{M_b}{M_w} \\ k_s = k_{CD} \\ c = c_{OD} + c_t \quad \text{or} \quad c = c_{CD} + c_t \end{cases}. \tag{7}$$

Thus, the considered suspensions can be modeled as follows:

- Oil damper placed in parallel with a compression spring can be described by a Kelvin-Voigt model, consisted of a dashpot and an elastic element connected in parallel (Fig. 1).
- Colloidal damper without attached compression spring can be described by a Maxwell model, consisted of a dashpot and an elastic element connected in series (Fig. 2).
- Colloidal damper mounted in parallel with a compression helical spring, can be described by a standard linear model, consisted of a Maxwell unit connected in parallel with an elastic element (Fig. 3).

4. Modeling of the Variable Damping

Variation of the damping ratio ξ versus the excitation frequency can be taken as:

$$\xi = \xi_n \left(\frac{\omega}{\omega_n}\right)^i, \tag{8}$$

where the natural circular frequency ω_n and the damping ratio at resonance ξ_n can be calculated as:

$$\omega_n = \sqrt{\frac{k}{m}}, \quad \xi_n = \frac{c}{2\sqrt{km}}. \tag{9}$$

Subscripts p or s will be added to ω_n and ξ_n in the sections below, according to the type of spring coefficient used for their calculus: k_p (parallel) or k_s (serial), respectively. The exponent i can be taken:

- as $i = 1$ for oil dampers at higher piston speeds [4];
- as $i = 0$ for oil dampers at lower piston speeds (see the simplified model of constant damping [2]–[4]);
- as $i = -1$ for colloidal dampers [15], [16], as well as control active oil dampers at higher piston speeds [17], [18].

5. Transmissibility of Vibration from the Rough Pavement to the Vehicle Body

Transfer function of vibration $|H|$ from the rough road to the vehicle's body is defined as the ratio of the amplitude X

of the equivalent mass m to the amplitude X_0 of the displacement excitation produced by the rough pavement (see Figs. 1–3):

$$|H| = X/X_0 . \tag{10}$$

5.1. Kelvin-Voigt (parallel) Model

In the case of Kelvin-Voigt model (Fig. 1), consisted of a dashpot and an elastic element connected in parallel, the transfer function of vibration is calculated by using Eq. (11). Then, variation of vibration transmissibility versus frequency $\omega/\omega_{n,p} = f/f_{n,p}$ is shown in Fig. 4 for $i=0$ and various damping ratios, and in Fig. 5 for different exponents i.

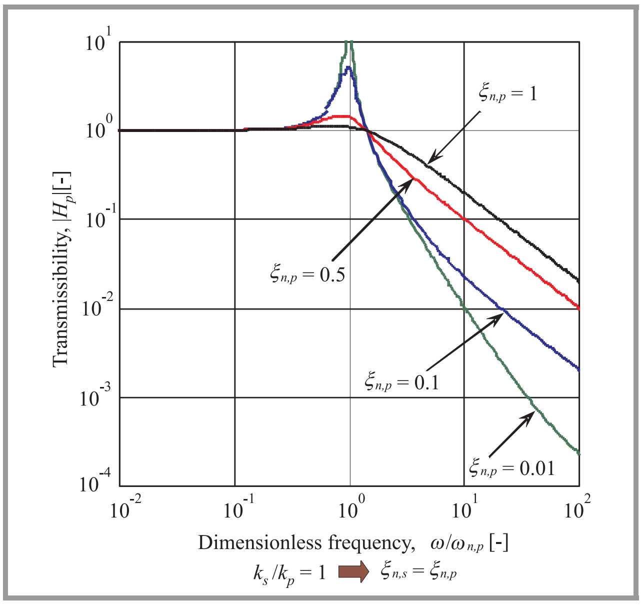


Fig. 4. Variation of transmissibility versus dimensionless frequency in the case of Kelvin-Voigt model for $i = 0$ and various damping ratios.

$$|H_p| = \sqrt{\frac{1+\left[2\xi_{n,p}\left(\frac{\omega}{\omega_{n,p}}\right)^{i+1}\right]^2}{\left[1-\left(\frac{\omega}{\omega_{n,p}}\right)^2\right]^2+\left[2\xi_{n,p}\left(\frac{\omega}{\omega_{n,p}}\right)^{i+1}\right]^2}} \tag{11}$$

From Eq. (11) and Fig. 5 one observes that for a given $\xi_{n,p}$ the resonant peak has the same height regardless the type of absorber:

$$|H_p| = \left(\frac{\omega}{\omega_{n,p}} = 1\right) = \sqrt{1+\frac{1}{4\xi_{n,p}^2}}, \quad (\forall)i . \tag{12}$$

Additionally, from Eq. (1) and Fig. 4 one observes that:

$$|H_p| = \left(\frac{\omega}{\omega_{n,p}} = \sqrt{2}\right) = 1, \quad (\forall)\xi_{n,p}, \quad (\forall)i . \tag{13}$$

Since all curves $|H_p|$ are above 1 for $\omega < \sqrt{2}\omega_{n,p}$ and below 1 for $\omega > \sqrt{2}\omega_{n,p}$one concludes that the critical frequency $\sqrt{2}\omega_{n,p}$ separates regions of amplification and attenuation, regardless the type of absorber and its damping ratio. When $\xi_{n,p} = 0$ the highest decay rate of transmissibility is obtained in the frequency domain $\omega > \sqrt{2}\omega_{n,p}$ but this is accompanied by very large amplitudes near the resonance. Consequently, one observes that there are two opposite requirements in the design of classical parallel-type suspension: large damping is desirable at lower frequencies to reduce the resonant peak, but on the other hand, low damping is needed at higher frequencies to minimize the transmissibility. Traditional way to reduce damping at higher frequencies in a passive manner [19] is to replace the dashpot by a Maxwell unit consisted of a dashpot and a spring connected in series (Fig. 3). Since the resonant peak is the same for all values of the exponent i but the decay rate in the higher frequency domain is the highest for $i = -1$ (Fig. 5), one arrives to a different way of reducing damping at higher frequencies in a passive manner. Concretely, the traditional oil damper ($i = 0$) can be replaced by a colloidal damper ($i = -1$), which has a dynamic behavior resembling the well-known case of structural damping.

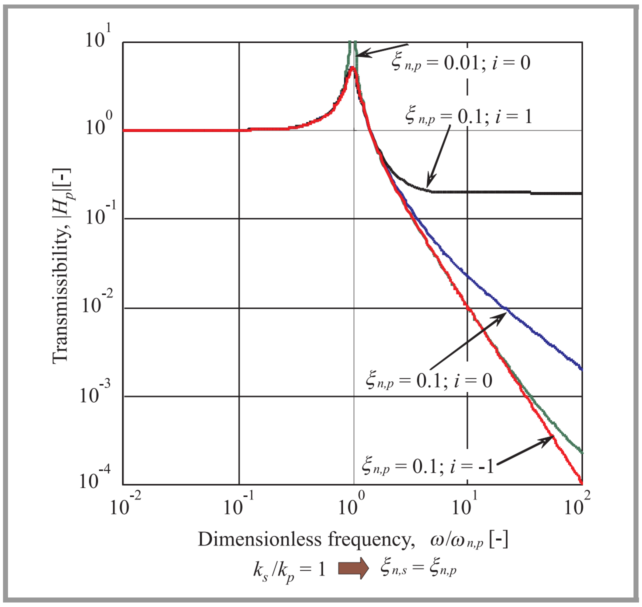


Fig. 5. Variation of transmissibility versus dimensionless frequency in the case of Kelvin-Voigt model for various exponents i.

5.2. Maxwell (series) Model

In the case of Maxwell model (Fig. 2), consisted of a dashpot and an elastic element connected in series, the transfer function of vibration can be calculated as:

$$|H_s| = \frac{2\xi_{n,s}\left(\frac{\omega}{\omega_{n,s}}\right)^i}{\sqrt{\left(\frac{\omega}{\omega_{n,s}}\right)^2+\left[2\xi_{n,s}\left(\frac{\omega}{\omega_{n,s}}\right)^i\right]^2\left[1-\left(\frac{\omega}{\omega_{n,s}}\right)^2\right]^2}} . \tag{14}$$

Then, variation of vibration transmissibility versus frequency $\omega/\omega_{n,s} = f/f_{n,s}$ is shown in Fig. 6 for $i = 0$ and various damping ratios, and in Fig. 7 for different values

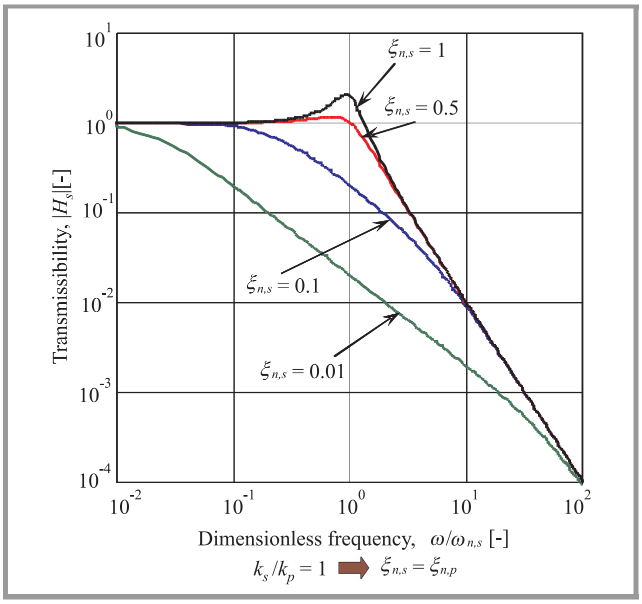


Fig. 6. Variation of transmissibility versus dimensionless frequency in the case of Maxwell model for $i = 0$ and various damping ratios.

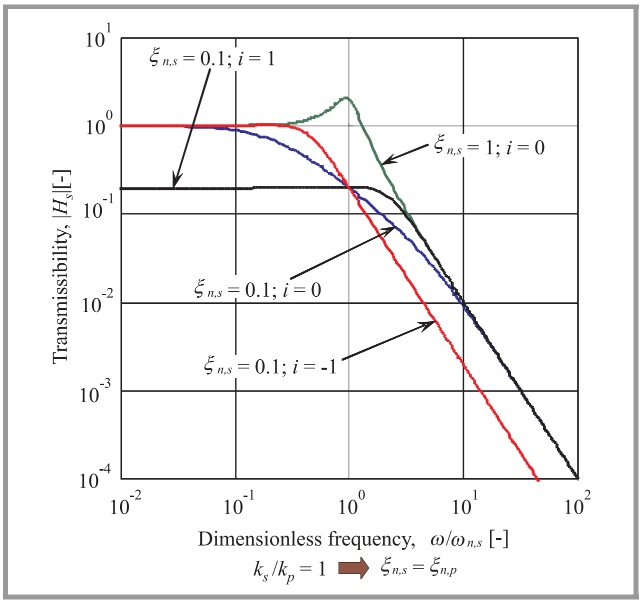


Fig. 7. Variation of transmissibility versus dimensionless frequency in the case of Maxwell model for various exponents i.

of the exponent i. Note that transmissibility reduces as the damping ratio $\xi_{n,s}$ decreases (Fig. 6); the lowest transfer of vibration is achieved in the region $\omega < \omega_{n,s}$ for the highest exponent i, but in the region $\omega > \omega_{n,s}$ for the lowest exponent i (Fig. 7).

From Eq. (14) one observes that for a given damping the "resonant peak" has the same height regardless the absorber:

$$|H_s|\left(\frac{\omega}{\omega_{n,s}} = 1\right) = 2\xi_{n,s}, \quad (\forall)i. \tag{15}$$

5.3. Standard Linear (series-parallel) Model

In the case of standard linear model (Fig. 3), consisted of a Maxwell unit connected in parallel with an elastic element, the transmissibility can be calculated as:

$$|H_{sp}| = \sqrt{\frac{\left(\frac{k_s}{k_p}\right)^2 + \left[2\xi_{n,p}\left(\frac{\omega}{\omega_{n,p}}\right)^{i+1}\left(1+\frac{k_s}{k_p}\right)\right]^2}{\left\{\frac{k_s}{k_p}\left[1-\left(\frac{\omega}{\omega_{n,p}}\right)^2\right]\right\}^2 + \left\{2\xi_{n,p}\left(\frac{\omega}{\omega_{n,p}}\right)^{i+1}\left[1+\frac{k_s}{k_p}-\left(\frac{\omega}{\omega_{n,p}}\right)^2\right]\right\}^2}}. \tag{16}$$

Then, variation of vibration transmissibility versus frequency $\omega/\omega_{n,p}$ is shown in Fig. 8 for a stiffness ratio $k_s/k_p = 1$, an exponent $i = 0$, and different damping coefficients $\xi_{n,p}$.

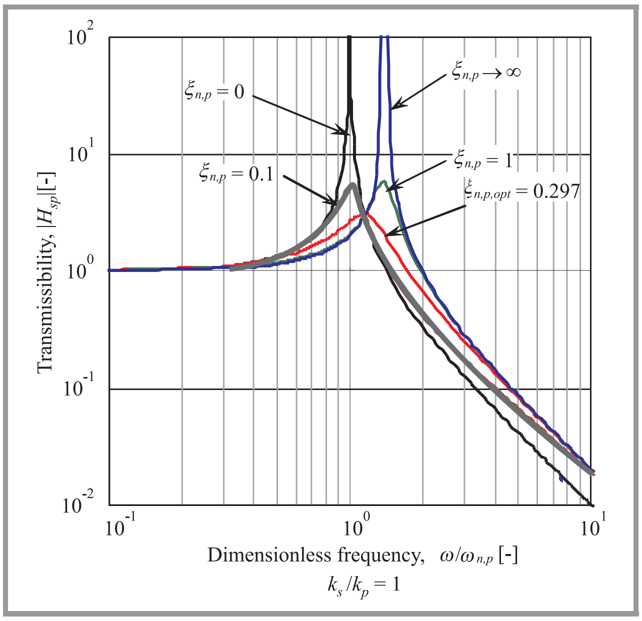


Fig. 8. Variation of transmissibility versus dimensionless frequency in the case of standard linear (serial-parallel model) for $i = 0$ and various damping ratios.

Figure 8 shows that for $\xi_{n,p} = 0$ (undamped suspension) a resonant peak occurs at $\omega = \omega_{n,p}$ and for $\xi_{n,p} \to \infty$ (over-damped suspension) a different resonant peak occurs at higher frequency ($\omega = \omega_{n,p}\sqrt{1+k_s/k_p}$). The lowest curve, that displays the lowest resonant peak at the junction point of the graphs shown for undamped and over-damped suspensions, corresponds to the optimal damping ratio $\xi_{n,p,opt} = 0.297$ which minimizes the vibration transmissibility.

6. Optimal Design of Serial-Parallel Suspension

Analyzing the structure of Eq. (16), one concludes that the resonant peaks shown by Fig. 8, i.e., the undamped peak occurring at $\omega = \omega_{n,p}$ and the over-damped peak occurring at $\omega = \omega_{n,p}\sqrt{1+k_s/k_p}$ are not depending on the type of damper (value of the exponent i). Based on the theory of the maximum achievable modal damping [19], in the same way as found in Fig. 8, for any given stiffness ratio k_s/k_p one always finds an optimal damping ratio $\xi_{n,p,opt}$ that minimizes the vibration transmissibility, as follows:

$$\xi_{n,p,opt} = \frac{1}{2}\frac{k_s}{k_p}\left(1+\frac{k_s}{k_p}\right)^{-3/4}, \quad (\forall) i. \tag{17}$$

Figure 9 illustrates a monotonous variation of the optimal damping ratio $\xi_{n,p,opt}$ versus the stiffness ratio k_s/k_p.

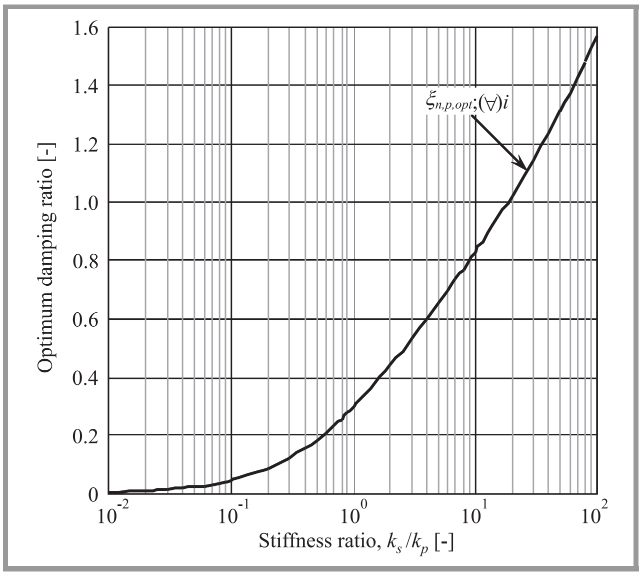


Fig. 9. Variation of the optimal damping ratio versus the stiffness ratio to minimize the transmissibility, for all types of dampers ($i = -1, 0, 1$).

Next, in order to obtain the optimal stiffness ratio, i.e., the optimal ratio of the colloidal spring's constant to the compression spring's constant, one compares the vibration transmissibility obtained with the parallel, series, and series-parallel models for values of the stiffness ratio ranging from low to high (e.g., $k_s/k_p = 0.01 - 100$). It can be shown that at augmentation of the stiffness ratio the resonant peak decreases, but the transmissibility in the higher frequency domain increases. In order to maximize the vehicle's ride-comfort in the whole frequency domain, one integrates the graphs showing the variation of vibration transmissibility versus frequency. In this way, as Fig. 10 illustrates, one obtains the variation of area below the graph of transmissibility versus the stiffness ratio, both for the case without filter and for the cases when filters (see Eq. (2) for the K-factor method, and Eq. (3) for the ISO 2631 method) are used to account for the effects of vibration on the human body. All the graphs shown in Fig. 10 are convex (valley-like) curves. Stiffness ratio corresponding to the deepest point of the valley represents the optimal stiffness ratio that minimizes the vibration transmissibility, i.e., maximizes the ride-comfort over the whole frequency range. Thus, based on the positions of minima observed in Fig. 10, the optimal stiffness ratio $(k_s/k_p)_{opt}$ is decided

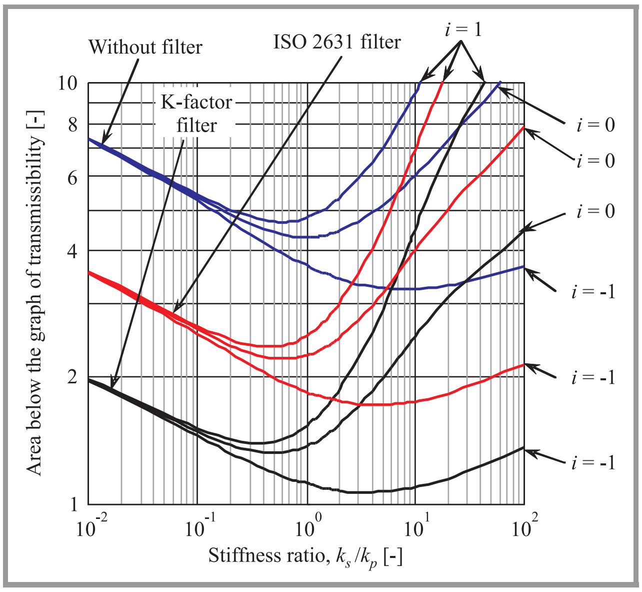


Fig. 10. Decision of the optimal stiffness ratio to achieve minimal transmissibility, i.e., maximal ride comfort for various dampers and filters.

for various types of absorbers ($i = -1, 0, 1$), both for the case without filter and for the cases when filters are used to account for the effects of vibration on the human body (see Table 1).

Table 1
Optimal stiffness ratio for various types of dampers and different methods to estimate the ride comfort

Optimal $(k_s/k_p)_{opt}$	Type of damper		
	$i = -1$	$i = 0$	$i = 1$
Without filter	8.0	1.0	0.6
ISO 2631 filter	5.0	0.6	0.5
K-factor filter	4.0	0.5	0.4

7. Conclusion

In this work, three types of suspensions were considered and modeled as follows: oil damper mounted in parallel with a compression helical spring, for which a Kelvin-Voigt model, consisted of a dashpot and an elastic element connected in parallel was considered; colloidal damper without attached compression helical spring, for which a Maxwell model, consisted of a dashpot and an elastic element connected in series was considered; and colloidal damper mounted in parallel with a compression helical spring, for which a standard linear model, consisted of a Maxwell unit connected in parallel with an elastic element was considered. Firstly, the vibration transmissibility from the rough road to the vehicle's body for all these suspensions was determined under the constraint that damping varies versus the excitation frequency. Then, the optimal damping and stiffness ratios were decided in order to minimize the transmissibility of vibration from the rough pavement to the vehicle's body, i.e., to maximize the vehicle's ride comfort.

References

[1] S. K. Jha, "Characteristics and sources of noise and vibration and their control in motor cars", *J. Sound and Vibration*, vol. 47, pp. 543–558, 1976.

[2] N. P. Chironis, *Spring Design and Application*. New York: McGraw-Hill, 1961.

[3] C. M. Harris and C. E. Crede, *Shock and Vibration Handbook*. New York: McGraw-Hill, 1968.

[4] S. Komamura, *Automotive Suspension*. Tokyo: Kayaba Technical, 2005 (in Japanese).

[5] "Mechanical Vibration and Shock – Evaluation of Human Exposure to Whole-Body Vibration". ISO 2631-1, 1997.

[6] D. Bastow, G. Howard and J. P. Whitehead, *Car Suspension and Handling*. New York: SAE International, 2004.

[7] R. F. Kuns, *Automotive Essentials*. New York: Bruce Publisher, 1973.

[8] C. V. Suciu, T. Iwatsubo and S. Deki, "Investigation of a colloidal damper", *J. Colloid and Interface Science*, vol. 259, pp. 62–80, 2003.

[9] C. V. Suciu, T. Iwatsubo and S. Deki, "Novel principle of mechanical energy dissipation; Part 1: Static performances of a colloidal damper", *JSME Int. J.*, vol. C47, pp. 180–188, 2004.

[10] C. V. Suciu and T. Iwatsubo, "Novel principle of mechanical energy dissipation; Part 2: Dynamic performances of a colloidal damper", *JSME Int. J.*, vol. C47, pp. 189–198, 2004.

[11] T. Iwatsubo, C. V. Suciu, M. Ikenaga, and K. Yaguchi, "Dynamic characteristics of a new damping element based on the surface extension principle in nanopores", *J. Sound and Vibration*, vol. 308, pp. 579–590, 2007.

[12] V. A. Eroshenko, "A new paradigm of mechanical energy dissipation; Part 1: Theoretical aspects and practical solutions", Proc. Inst. Mech. Eng., Part D, *J. Automobile Eng.*, vol. 221, pp. 285–300, 2007.

[13] V. A. Eroshenko, I. Piatiletov, L. Coiffard, and V. Stoudenets, "A new paradigm of mechanical energy dissipation; Part 2: Experimental investigation and effectiveness of a novel car damper", Proc. Inst. Mech. Eng., Part D, *J. Automobile Eng.*, vol. 221, pp. 301–312, 2007.

[14] C. V. Suciu and K. Yaguchi, "Endurance tests on a colloidal damper destined to vehicle suspension", *Experimental Mechanics Int. J.*, vol. 49, pp. 383–393, 2009.

[15] C. V. Suciu and T. Tobiishi, "Investigations on the optimum design of a colloidal damper for autovehicle suspension", in *Proc. 9th World Congr. Struct. Multidiscipl. Optim.*, Shizuoka, Japan, CD-ROM, pp. 1–10, 2011.

[16] C. V. Suciu, "Ride-comfort of an automobile equipped with colloidal dampers at its frontal suspension", in *Proc. Int. Conf. on Noise and Vibration Eng.*, Leuven, Belgium, CD-ROM, pp. 4233–4245, 2010.

[17] A. Hac and I. Youn, "Optimal semi-active suspension with preview based on a quarter car model", *Trans ASME, J. Vibration Acoustics*, vol. 114, pp. 84–92, 1992.

[18] A. Hac and I. Youn, "Optimal design of active and semi-active suspensions including time delays and preview", *Trans ASME, J. Vibration Acoustics*, vol. 115, pp. 498–508, 1993.

[19] A. Preumont and K. Seto, *Active Control of Structures*. New York: Wiley, 2008.

Claudiu Valentin Suciu received the Dr.Eng. from Polytechnic University of Bucharest (Romania) in 1997, and Dr.Eng. from Kobe University (Japan) in 2003. Currently, Professor C. V. Suciu works at the Department of of Intelligent Mechanical Engineering. His research interests are in tribological design of various machine elements.

E-mail: suciu@fit.ac.jp
Faculty of Engineering
Department of Intelligent Mechanical Engineering
Fukuoka Institute of Technology
3-30-1 Wajiro-Higashi, Higashi-ku, Fukuoka
811-0295, Japan

Tsubasa Tobiishi is master-course student in Graduate School of Engineering, Fukuoka Institute of Technology.

E-mail: mcm10007@bene.fit.ac.jp
Faculty of Engineering
Department of Intelligent Mechanical Engineering
Fukuoka Institute of Technology
3-30-1 Wajiro-Higashi, Higashi-ku, Fukuoka
811-0295, Japan

Ryouta Morii is master-course student in Graduate School of Engineering, Fukuoka Institute of Technology.

E-mail: mcm11010@bene.fit.ac.jp
Faculty of Engineering
Department of Intelligent Mechanical Engineering
Fukuoka Institute of Technology
3-30-1 Wajiro-Higashi, Higashi-ku, Fukuoka
811-0295, Japan

Paper

The Application of Neural Networks to the Process of Gaining and Consolidating the Knowledge

Anna Plichta

Cracow University of Technology, Kraków, Poland

Abstract—The e-learning course is one of the most efficient and promising didactic policies. It must be grounded on the revision because it was proved that it enhances the long-term memory. However, human mind is not a uniform phenomenon. Each man memorizes and learns in a different manner. The purpose of the intelligent e-learning system presented in this paper is to teach orthography and this system is based on the multilayer neural network. Such structure enables a learner to adjust the crucial period between revisions to personal learning habits and policy.

***Keywords—** e-learning, neural network, orthography.*

1. Introduction

Owing to the modern information and communication technologies in education (ICT) a teacher is given a great opportunity to prepare his/her students to deliberate and controlled dealing with the available information.

The functioning educational policy, grounded on so called 'just in case learning', is not efficient enough [1]. It helps to gain knowledge, but the extent of the practical application of that knowledge is limited. Learners are aware of this fact and lack a commitment to learning, what worsens the entire didactic process. Hence, according to e-learning the 'just in case learning' should be replaced with methods and policies which:

– are adjusted to suit the learner's needs ('just for me' policy),

– are provided exactly when they are necessary ('just in time' policy),

– enable a learner to acquire enough knowledge ('just enough' policy).

The e-learning course fulfils such requirements. It is one of the most effective teaching strategies, because it uses only those teaching methods which are suited to the needs of the learner. As a result, they have positive effect on the learner's motivation.

E-learning is an innovative method of knowledge because the methods of solidifying the learning material are based on the profound research on the human brain, conducted in order to examine how it is constructed and how it functions. Numerous experiments clearly prove the existence of potential, though still not discovered, possibilities of memory processes. Nowadays, these possibilities can be enhanced via various mnemonic techniques [2]. Among them, regular repetitions seem to be the most crucial, because they facilitate effective knowledge acquirement, selection of information and its further use. Model of the functioning of the e-learning system is presented in Fig. 1.

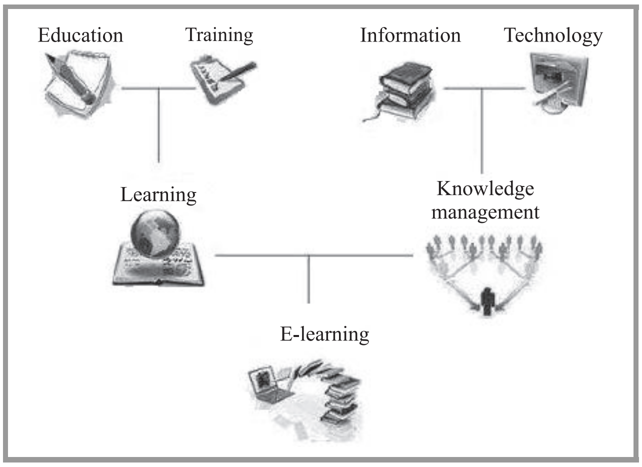


Fig. 1. Model of the functioning of the e-learning.

The suggested application of learning orthography of the Polish language refers to the above-mentioned recommendations, emphasizing above all the interactivity and virtuality of the learning reality.

2. Methods of Memorizing the Learning Material through E-learning Courses

The standards of an e-learning course (regarding the design of the learning material) concern establishing and describing training aims, defining the conditions of the course distribution, and using the prototypes of the projected system. First of all, the learning content should serve principally the educational needs. Gaining new competencies and skills is much easier for a user if the content underwent hierarchization and the legibility and cohesion of the transfer was strengthen. Thus, we should take into consideration the possible equipment limitations of the users and thought out the layout of the course, providing cohesive navigation. Portioning the material is also a significant element, because we better learn smaller fragments than big chunks. For a learning process to be effective, the material which is supposed to be memorized should be divided into smaller fragments, i.e., lesson units. A kind and method of division also influences the quality of the learning process [1].

It is proved, that breaks have a beneficial effect on the process of learning. We remember better the things we read at the beginning and at the end, worse the middle part of the material. So, thanks to the breaks we have more beginnings and endings.

Last but not least, to learn better one should not forget about the crucial role of the memory, which is dependant on the metabolic processes the brain is involved in. For instance, eating a heavy meal causes temporary sluggishness and low responsiveness. Avoiding heavy meals and sugars before a bigger mental effort is also advisable, because it increases the level of insulin in blood and thus affects the functioning of the brain. However, it is important not to starve either, because the brain needs fuel to operate.

3. Methods of Planning Repetitions

An important element of the learning process is repetition. Repetitions are very important because learning means creating new tracks in one's mind. The more often they are used, the better are shaped connections between neurons. Revising the material is a necessity so it needs to be carefully planned. Thanks to repetitions,knowledge is systematically solidified and it stays longer in memory. The optimum time after which a material should be revised varies for different people, it depends on individual predispositions of a learner. To be able to remember the memorized material for a longer time, repetitions should be organized not accidentally but with suitable breaks [2]. Consequent revisions can be planned in the following way:

– after about ten minutes from studying,
– the same day before going to sleep,
– in the morning of the next day,
– after a week,
– after a month,
– right before using the gained knowledge, e.g., before the exam.

3.1. Determining the Best Possible Time of Revision

The optimal time for revision is different for various learners. I will concern the e-learning system intended to teach the orthography. I used the neural network and forgetting curve to set the optimal time for the revision.

German psychologist Hermann von Ebbinghaus drew the forgetting curve (also known as the Ebbinghaus curve) which stems from his research on human memory. It reflects the observed regularities in the process of learning and forgetting, namely, it illustrates the relationship between the amount of acquired information and the time which passed since the moment of learning. At the very beginning the curve is falling rapidly but then it turn almost flat. The forgetting curve is presented in Fig. 2.

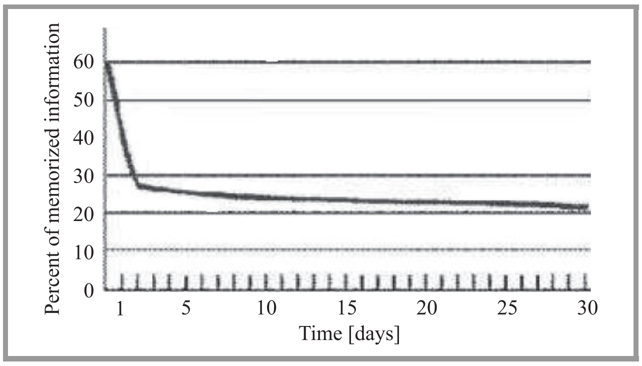


Fig. 2. The forgetting curve.

Once the learning is finished the amount of memorized information decreases rapidly. Half of the material is forgotten in an hour. Two days later the forgetting process is remarkably less rapid. Thanks to the revision of the acquired material the pace of forgetting is slower and slower. The Ebbinghaus curve can be approximately projected by means of the following functions:

- Exponential function:

$$m = (a-c)e^{-bt} + c, \tag{1}$$

where: m stands for the amount of memorized information, b for the coefficient of forgetting, a for the coefficient of memorizing, c for the asymptote, t for the time.

- Power function:

$$m = g(1+bt)^{-i}, \tag{2}$$

where: m stands for the amount of memorized information, g for the level of the long-term memory, t for the time, i for the coefficient of forgetting.

The forgetting curves projected when the learning material is revised is presented in Fig. 3

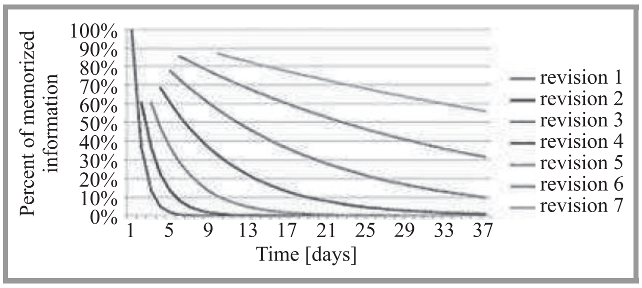


Fig. 3. Forgetting curves projected when the learning material is revised.

According to the forgetting curve, ten minutes after the termination of the learning process a man holds on 90% of the information he has gained. If we repeat that information we will enhance the relative power of memorizing and thus lengthen the period when it can't be forgotten.
The system uses the projection of the forgetting curve in the power form, namely in the form of the following equation:

$$R = e^{-\frac{t}{s}}, \tag{3}$$

where: R stands for the amount of easily-recalled information, t for the number of days measured from the revision, s for the relative power of the memory.
Now, what we seek is the proper value of the time between two revisions. Hence, we transform the Eq. (3) to the form:

$$t = -s \log R. \tag{4}$$

For the sake of convenience we assumed that $R = 70\%$. It is the threshold value for the system, which means that the revision takes place whenever the level the memorized information reaches 70%. The chart in Fig. 2 illustrates how the revisions affect the shape of the curve. We should notice that the time of the revision is calculated by means of the universal methods. Hence, the system of revisions should be deprived of any personalization.

3.2. Determining the Following Revision by Means of the Neural Network

The concept of the artificial neural network is derived from the research on human mind and the interrelationship between the artificial neurons. Nowadays, artificial neurons are interconnected variously, either in the software or within the integrated circuits. The e-learning system is grounded on the multilayer perceptron neural network [3], [4].
The provided network is taught by means of the method of the backward propagation of the errors (abbr. to back propagation), which belongs to supervised methods. It rests on the providing the network being learnt with some raw data and the expected output data. The network learns correcting the neural weights: the final error committed by the network should be less severe than the error set at the beginning. The name "backward propagation" is derived from the way the errors are computed in each particular layer of the network. It begins with the computation in the output layer by comparing the received and expected data. Then the errors for the preceding layer are computed (as the functions of errors of the preceding layer) and so on, up to the input layer. For better effect, the number of learning cycles is fixed. It enables the value of the final error to be decreased [5]–[7].
The artificial neural network was taught by means of the experimental set including the values of data for each of the 4 inputs and expected output values. The student is provided with the pre-taught network which enables him to use the system. Then the network is adjusted to the way the student's memory functions. The adjusting process has global character because it appertains to the entire set of questions related to the particular orthographic rule (and not to a single question). The network is taught again whenever:

- the student was graded very well; it could be up to 5th revision and there was more than 10 days between revisions,
- the student was graded very well; it was more than 10th revision and more than 200 days between revisions.

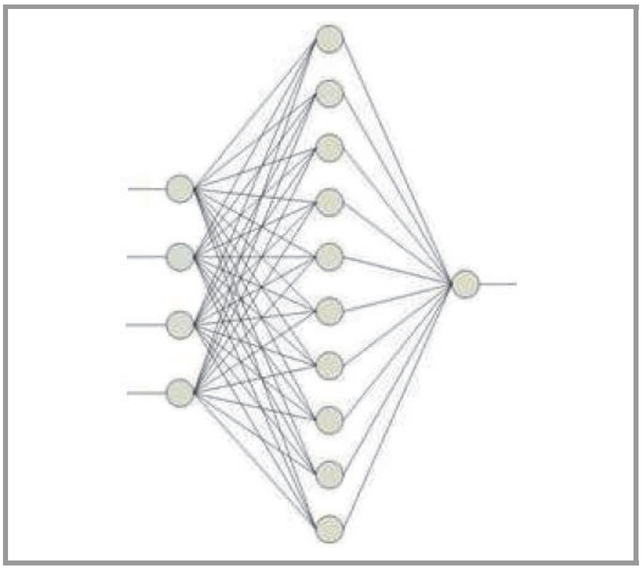


Fig. 4. Neural network provided for the purpose of the e-learning system.

The network is protected from over-learning by the threshold value which sets the maximal number of teaching cycles, e.g., 10000. Neural network provided for the purpose of the e-learning system is presented in Fig. 4.

4. E-learning Course to Learn Polish Spelling

The survey concerning the use of IT in teaching the humanities in primary schools conducted in 2005 shows that nowadays e-learning is a very important aspect of teaching. The results indicate that multimedia programs supporting traditional teaching are becoming more and more popular – they facilitate searching out information and broadening knowledge. They influence the effectiveness of teaching, because pupils absorb information faster, remember more, develop new skills – self-reliance, ability to draw conclusions, logical thinking, cross-domain knowledge and expertise, and perceptiveness [8].
The results show positive personality traits observed among students who use multimedia programs. Perceptiveness, ability to search out the most important information and drawing logical conclusions are most often observed among people trained by e-learning methods. The results are op-

timistic and encourage using e-learning courses. But only conscious and thought-out methods, adequate to student's needs, age and capabilities, are genuinely effective.

The studies also point to the computerization of the humanities curriculum. Teaching grammar and spelling of Polish is difficult and time consuming, and therefore the most popular programs are those which support teaching these branches.

Among many offers available, most are programs for students who already have some knowledge and are willing to organize and systematize it. The most frequent are collections of dictations read by native speakers. For each text a detailed description and proposed exercises are available in order to verify the knowledge and improve the skills. Theoretical issues are grouped into thematic blocks concerning both punctuation and spelling. One can also create their own exercises or add new dictations, as well as use the built-in spelling dictionary.

Another advantage is the possibility to customize the interface (there are several options to choose from) to user preferences and graphic statistics editor which shows educational progress of individual users in the form of graphs and fractions. It is an interesting option for students who already have established knowledge of orthography. The program also allows monitoring the work progress. Half of the surveyed teachers use this type of support in teaching spelling of the Polish language in primary schools.

An equally popular form of learning spelling are adventure games designed for younger students who, along with virtual characters, solve spelling puzzles and reach subsequent levels. As they are attractive, they facilitate learning the principles and rules of spelling and they support reading comprehension. A Polish language teacher may therefore use a wide range of multimedia programs to support conventional teaching. However, the offers available are mainly educational games and interactive tests. A student takes on a role, e.g., of a traveler or a detective, and, on the basis of the existing knowledge, they overcome obstacles, gaining further degrees of initiation until they solve the puzzle. To take full advantage of this training material, the gained and established knowledge is required. Moreover, the correctness of the tasks given to students is verified, but unexplained. There is no feedback with the extensive commentary, as well as no virtual teacher/mentor to support the trainee's activities.

A proper selection of an educational program stimulates students, even the weaker ones. Familiar topics, attractive graphics and sounds, interesting and diverse in terms of difficulty tasks with elements of fun, different forms of rewards for good work are all able to attract students' attention for much longer than other teaching aids.

The solution which offers both an attractive layout and an abundance of learning content is the e-learning course. Despite the growing interest and demand for this form of training, e-learning courses for teaching spelling to students in grades IV–VI are currently not included in the primary schools' offer.

The proposed e-learning system is based on the knowledge of Polish spelling. Exercises, lessons and tests – all training material included in the course is developed in accordance with the principles and rules of spelling. It offers substantial support to students – a set of principles and rules of spelling tailored to their intellectual and cognitive capabilities and a wide base of exercises and tests that can be continuously expanded and modified. The teacher can also define lesson units by selecting predefined exercises that students from given classes will implement in a given class. The proposed exercises and lessons take into account the individual pace of students' work and allow the teacher to monitor the results.

The students have an opportunity to get acquainted with theoretical knowledge concerning the principles and spelling rules of the Polish language and to perform a set of exercises whenever and wherever they want. They have the possibility to verify their answers, check the mistakes and re-do the exercise.

The e-learning support of teaching spelling may be used in primary schools which use electronic forms of learning in classes. It is an easily adaptable application – it can be extended by additional modules compliant with the requirements of the educational system.

It suits both motivational expectations of students, who nowadays prefer remote teaching, and contemporary tendencies in the psychology of learning, which emphasizes active forms of teaching through playing. It also allows the learner to extract the feedback and correct their errors, which increases the motivation to learn. One should note the possibility of integrating teaching of spelling with teaching Polish grammar, phraseology and punctuation. The designed e-learning system is a methodological proposal for ill students or those from integrated classes.

The system is divided into two modules: teacher and student panel – depending on the user's authorization.

4.1. Teacher Panel

After logging in, the teacher is able to use the panel, which is divided into four parts:

- Exercises module lets the teacher define, view and edit tasks that students should do on their own.
- Students module allows users to define the roles of users allowed to use the system.
- Lesson module allows the teacher to suggest any set of exercises on spelling rules that the student should do within the teaching unit. Exercises constituting a lesson unit can be freely selected from a pool of available exercises. If the teacher wants to use the new set of exercises, they must enter them into the database earlier using the tab Exercises – Add new.
- Test module allows defining, viewing and editing multiple choice tests which check the student's acquired knowledge of the spelling of the Polish lan-

guage. The tests allow the teacher to assess the effectiveness of the course and the personal attributes of each student. The applied data validation prevents them from choosing more than one answer. Scoring is distributed evenly – each question scores one point. The maximum number of points one can get for all the correct answers is 24. The system informs the student about the number of points obtained and the mark given for the test (numerical and descriptive suggesting a direction for further work in the event of a negative evaluation, or encouragement to achieve even better results).

The teacher has the possibility to view individual students' test results, they can also print them in the form of reports. Test results apply to an individual student attending given class. All tasks solved in a given test are displayed, as well as the student's answers and the correct answers. A personal report gives the number of correct answers and the mark suggested. The obtained results allow for a credible assessment of the student's skills and competencies.

4.2. Student Panel

After logging in the student is able to use the student's control panel, which is divided into three parts:

- Exercises module allows getting acquainted with theoretical knowledge on spelling principles and rules of the Polish language and their practical application – the implementation of a set of exercises illustrating the correct spelling within a particular rule. The student checks the answers and verifies the mistakes, e.g., using a spelling dictionary.
- Lessons module allows the student to choose a set of exercises, done within the lesson unit. Exercises, suggested by the teacher, match the spelling rule/rules.
- Test module includes multiple choice tests with assessment that the student does independently. After selecting the type of the test the first question is displayed. After choosing the answer, the next question is shown.

The student has an opportunity to check the correct solution of the test, but only after its completion. On the monitor there is a report displayed, informing the student, which answer was chosen and giving the correct answer. At the end of the report there is a summary – the number of correct answers and the proposed mark.
Thanks to the data base of the questions, the system is able to organize repetitions. Repetitions involve showing consequent questions by the system on the main panel of the application, to which the learner answers. The answer is then verified by comparing it with the correct one. The system enables also course and user's accounts administration, their rights to get to specific functions and platform's resources. It allows adding a new course or adding new information to the existing ones. The important element is communication between the individual members of the course and the teacher in a synchronous way (chat) and asynchronous one (forum, email) and creating virtual groups of members and teachers. The valuable source of information about the effectiveness of the proposed course is the ability to examine the activity of the learners, surveying the members and teachers and checking the learning progress.

5. The Results of the Research

The e-learning system was tested on two groups of 15 students. Each of them had to perform the same task: get acquainted with the theoretical material concerning Polish spelling and do a set of several exercises testing in practice the acquired knowledge. At the end the students had to do a final test summarizing their skills in the proper use of the spelling of the Polish language.
The first group implemented the program of the experiment based on traditional methods of teaching, i.e., learning the rules and on their basis doing exercises given by the teacher, dictations from the whole of the material, writing down difficult words. The students were passive participants of the course; they received information and executed commands.
The second group used the proposed e-learning system supporting learning spelling. The students learned new rules and did the exercises on the spot. They immediately got feedback on their mistakes and successes. Error correction resulted in faster and more efficient mastery of the theoretical material, and thus better results in the performed exercises and tests.
The results clearly show that active learning, engaging students in the process of learning new information, brings much better results than traditional methods. The students who use e-learning system were more likely to learn, showed greater initiative and self-discipline throughout the entire educational process. The effectiveness of assimilating knowledge of the Polish spelling in the second group was nearly 15% higher than in the first group.
Multimedia education affects students' senses to a greater extent than conventional teaching. Thanks to that, the effectiveness of learning has increased by as much as 15%, understanding of the subject by 46% and the range of absorbed knowledge was higher by 35%. In addition, e-learning makes it possible to reduce the difficulties and ambiguities in the implementation of learning – in this case by 28% and time saving of 40% while increasing the pace of work by 25%. Using these programs, however, requires a teacher care. Student's independence should be supported by the teacher's experience and knowledge.
Tests also were conducted in two 15-person groups. Students from the first group were provided with e-learning course without the support of scheduling the term of revision, whereas the second group of students were provided with the system where the optimal time of revision was scheduled by the neural network built-in the system.

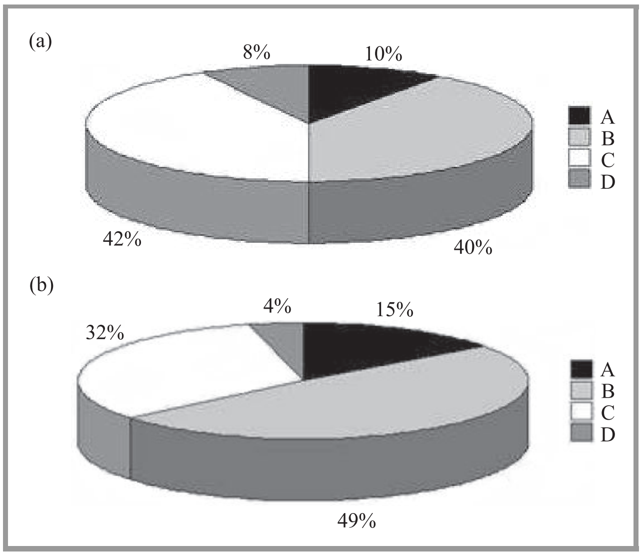


Fig. 5. Results of the research for (a) – first group and (b) second group. Evaluation used in the experiment: A – very good, B – good, C – fair, D – failed.

Among students using the system with neural network the efficiency of memorizing the training material has increased by 15%. Experiment results have been shown in the diagrams in Fig. 5.

6. Summary

The e-learning system enhancing the process of learning the orthography in school has also the additional module. It manages the gained knowledge, enabling the revisions to be planned. Similar systems lead the education to its future.

So far, one could set the conditions under which the time of revision may come, but they apply to every orthographic rule. Perhaps we should enable the users to set a distinct conditions for each orthographic issue. Transferring the coefficients of the artificial neural network, so that they accommodate to the conditions of the particular rule, would make the following revisions for the particular sets of questions more correct. The questions concerning the particular orthographic rules could be more or less difficult, what would affect the process of memorizing the information. Apart from that, the system could be expanded with the statistics providing the fixed dates and results of the revisions. Owing to them the student would be aware of his current state of knowledge and of particular difficulties. He would know if his learning policy is efficient or needs to be changed.

References

[1] M. Hyla, *Przewodnik po E-learningu*. Kraków: Oficyna Ekonomiczna, 2005 (in Polish).

[2] G. Dryden, J. Vos, *Rewolucja w Uczeniu*. Poznań: Moderski i S-ka, 2000 (in Polish).

[3] *Sieci Neuronowe*, W. Duch, J. Korbicz, L. Rutkowski, and R. Tadeusiewicz (Red.), Tom 6, Seria: *Biocybernetyka i Inżyniera Biomedyczna*. Warszawa: Wydawnictwo EXIT, 2000 (in Polish).

[4] R. Tadeusiewicz, B. Borowik, T. Gąciarz, B. Lepper, *Odkrywanie Właściwości Sieci Neuronowych przy Użyciu Programów w Języku C#*. Kraków: Wydawnictwo PAU, 2007 (in Polish).

[5] A. Konar, *Artificial Intelligence and Soft Computing: Behavioral and Cognitive Modeling of the Human Brain*. CRC Press, 2000.

[6] R. Tadeusiewicz, *Sieci Neuronowe*. Warszawa: Akademicka Oficyna Wydawnicza, 1993 (in Polish).

[7] A. White, A. R. Gallant, K. Hornik, M. Stinchcombe, and J. Wooldridge, *Artificial Neural Networks: Approximation and Learning Theory*. Cambridge, MA: Blackwell Publishers, 1992.

[8] *Dydaktyka Informatyki i Informatyka w Dydaktyce*, A. Szewczyk (Red.). Szczecin: Printshop, 2006 (in Polish).

Anna Plichta was born in Cracow. She studied comparative literature at the Jagiellonian University and obtained her degree in 2007. She also studied computer science at Cracow University of Technology and obtained her degree in 2010. Currently, she works as a teaching fellow at Cracow University of Technology.

E-mail: aplichta@pk.edu.pl
Cracow University of Technology
Powstańców st 30a/80
31-422 Kraków, Poland

Information for Authors

Journal of Telecommunications and Information Technology **(JTIT)** is published quarterly. It comprises original contributions, dealing with a wide range of topics related to telecommunications and information technology. **All papers 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.

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. TEX and LATEX are preferable, standard Microsoft Word format (.doc) is acceptable. The author's JTIT LATEX style file is available:
http://www.itl.waw.pl/publ/jtit/doc/jtitaut.zip

Papers published should contain up to 10 printed pages in LATEX author's style (Word processor one printed page corresponds approximately to 6000 characters).

The manuscript should include an abstract about 150–200 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. Abstracts must not include mathematical expressions or bibliographic references.

Keywords should not repeat the title of the manuscript. About four keywords or phrases in alphabetical order should be used, separated by commas.

The original files accompanied with pdf file should be submitted by e-mail: redakcja@itl.waw.pl

Figures, tables and photographs. Original figures should be submitted. Drawings in Corel Draw and PostScript formats are preferred. Figure captions should be placed below the figures and can not be included as a part of the figure. Each figure should be submitted as a separated graphic file, in .cdr, .eps, .ps, .png or .tif format. Tables and figures should be numbered consecutively with Arabic numerals.

Each photograph with minimum 300 dpi resolution should be delivered in electronic formats (TIFF, JPG or PNG) as a separated file.

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 200 words and a photo of each author should be included with the manuscript.

Galley proofs. Authors should return proofs as a list of corrections 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 one week 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.
For copyright form see:
http://www.itl.waw.pl/publ/jtit/wskazowki.html

A copy of the JTIT is provided to each author of paper published.

(Contents Continued from Front Cover)

INSTYTUT ŁĄCZNOŚCI
PAŃSTWOWY INSTYTUT BADAWCZY

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