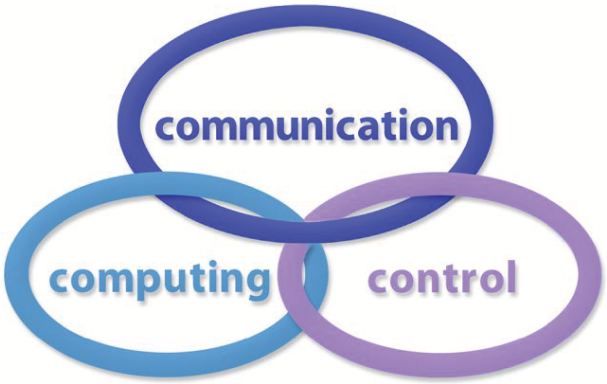
INTERNATIONAL JOURNAL

of

COMPUTERS COMMUNICATIONS & CONTROL

ISSN 1841-9836



A Bimonthly Journal
With Emphasis on the Integration of Three Technologies

Year: 2014 Volume: 9 Issue: 6 (December)

This journal is a member of, and subscribes to the principles of, the Committee on Publication Ethics (COPE).

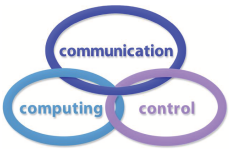
CCC Publications - Agora University Editing House

CCC Publications

http://univagora.ro/jour/index.php/ijccc/

BRIEF DESCRIPTION OF JOURNAL

Publication Name: International Journal of Computers Communications & Control.
Acronym: IJCCC; **Starting year of IJCCC:** 2006.
Abbreviated Journal Title in JCR: INT J COMPUT COMMUN.
International Standard Serial Number: ISSN 1841-9836.
Publisher: CCC Publications - Agora University of Oradea.
Publication frequency: Bimonthly: Issue 1 (February); Issue 2 (April); Issue 3 (June); Issue 4 (August); Issue 5 (October); Issue 6 (December).
Founders of IJCCC: Ioan DZITAC, Florin Gheorghe FILIP and Mişu-Jan MANOLESCU.
Logo:



Indexing/Coverage:

- Since 2006, Vol. 1 (S), IJCCC is covered by Thomson Reuters and is indexed in ISI Web of Science/Knowledge: Science Citation Index Expanded.
- Journal Citation Reports (JCR - Science Edition), IF = 0.694 (JCR2013).
  Subject Category:
  - Automation & Control Systems: Q4 (46 of 59);
  - Computer Science, Information Systems: Q3 (96 of 135).
- Since 2008, 3(1), IJCCC is covered in Scopus, SJR2013 = 0.231, H index = 13.
  Subject Category:
  - Computational Theory and Mathematics: Q4;
  - Computer Networks and Communications: Q3;
  - Computer Science Applications: Q3.
- Since 2007, 2(1), IJCCC is covered in EBSCO.

Focus & Scope: International Journal of Computers Communications & Control is directed to the international communities of scientific researchers in computer and control from the universities, research units and industry.

To differentiate from other similar journals, the editorial policy of IJCCC encourages the submission of original scientific papers that focus on the integration of the 3 "C" (Computing, Communication, Control).

In particular the following topics are expected to be addressed by authors:

- Integrated solutions in computer-based control and communications;
- Computational intelligence methods (with particular emphasis on fuzzy logic-based methods, ANN, evolutionary computing, collective/swarm intelligence);
- Advanced decision support systems (with particular emphasis on the usage of combined solvers and/or web technologies).

Editorial Address:
Agora University/ R&D Agora Ltd. / S.C. Cercetare Dezvoltare Agora S.R.L.
Piata Tineretului 8, Oradea, jud. Bihor, Romania, Zip Code 410526
Tel./ Fax: +40 359101032
E-mail: ijccc@univagora.ro, rd.agora@univagora.ro, ccc.journal@gmail.com
Journal website: http://univagora.ro/jour/index.php/ijccc/

Gheorghe Păun
Institute of Math. of Romanian Academy
Bucharest, PO Box 1-764, Romania
gpaun@us.es

Mario de J. Pérez Jiménez
Dept. of CS and Artificial Intelligence
University of Seville, Sevilla,
Avda. Reina Mercedes s/n, 41012, Spain
marper@us.es

Dana Petcu
Computer Science Department
Western University of Timisoara
V.Parvan 4, 300223 Timisoara, Romania
petcu@info.uvt.ro

Radu Popescu-Zeletin
Fraunhofer Institute for Open
Communication Systems
Technical University Berlin, Germany
rpz@cs.tu-berlin.de

Imre J. Rudas
Óbuda University
Budapest, Hungary
rudas@bmf.hu

Yong Shi
School of Management
Chinese Academy of Sciences
Beijing 100190, China &
University of Nebraska at Omaha
Omaha, NE 68182, U.S.A.
yshi@gucas.ac.cn, yshi@unomaha.edu

Athanasios D. Styliadis
University of Kavala
Institute of Technology
65404 Kavala, Greece
styliadis@teikav.edu.gr

Gheorghe Tecuci
Learning Agents Center
George Mason University
U.S.A.
University Drive 4440, Fairfax VA
tecuci@gmu.edu

Horia-Nicolai Teodorescu
Faculty of Electronics and
Telecommunications
Technical University "Gh. Asachi" Iasi
Iasi, Bd. Carol I 11, 700506, Romania
hteodor@etc.tuiasi.ro

Dan Tufiş
Research Institute for Artificial Intelligence
of the Romanian Academy
Bucharest, "13 Septembrie" 13, 050711,
Romania
tufis@racai.ro

Lotfi A. Zadeh
Director,
Berkeley Initiative in Soft Computing (BISC)
Computer Science Division
University of California Berkeley,
Berkeley, CA 94720-1776
U.S.A.
zadeh@eecs.berkeley.edu

DATA FOR SUBSCRIBERS
Supplier: Cercetare Dezvoltare Agora Srl (Research & Development Agora Ltd.)
Fiscal code: 24747462
Headquarter: Oradea, Piata Tineretului Nr.8, Bihor, Romania, Zip code 410526
Bank: BANCA COMERCIALA FEROVIARA S.A. ORADEA
Bank address: P-ta Unirii Nr. 8, Oradea, Bihor, România
IBAN Account for EURO: RO50BFER248000014038EU01
SWIFT CODE (eq.BIC): BFER

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, Volume 9, Issue 6, December, 2014.

Contents

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):664-671, December, 2014.

Hadoop Optimization for Massive Image Processing: Case Study Face Detection

İ. Demir, A. Sayar

İlginç Demir
Advanced Technologies Research Institute
The Scientific and Technological Research Council of Turkey
ilginc.demir@tubitak.gov.tr

Ahmet Sayar*
Computer Engineerig Department
Kocaeli University, Turkey
*Corresponding author: ahmet.sayar@kocaeli.edu.tr

Abstract:
Face detection applications are widely used for searching, tagging and classifying people inside very large image databases. This type of applications requires processing of relatively small sized and large number of images. On the other hand, Hadoop Distributed File System (HDFS) is originally designed for storing and processing large-size files. Huge number of small-size images causes slowdown in HDFS by increasing total initialization time of jobs, scheduling overhead of tasks and memory usage of the file system manager (Namenode). The study in this paper presents two approaches to improve small image file processing performance of HDFS. These are (1) converting the images into single large-size file by merging and (2) combining many images for a single task without merging. We also introduce novel Hadoop file formats and record generation methods (for reading image content) in order to develop these techniques.
Keywords: Hadoop, MapReduce, Cloud Computing, Face Detection.

1 Introduction

In the last decade, multimedia usage has increased very quickly, especially as parallel to high usage rate of the Internet. Multimedia data, stored by Flicker, YouTube and social networking sites like Facebook, has reached enormous size. Today search engines facilitate searching of multimedia content on large data sets. So these servers has to manage storing and processing this much data.

Distributed systems are generally used to store and process large scale multimedia data in a parallel manner. Distributed systems have to be scalable for both adding new nodes and for running different jobs simultaneously. Images and videos are the largest set of these multimedia contents. So, image processing jobs are required to run in distributed systems to classify, search and tag the images. There are some distributed systems enabling large scale data storing and processing. Hadoop distributed file system (HDFS) [1] is developed as an open-source project to manage storage and parallel processing of large scale data.

HDFS parallel processing infrastructure is based on MapReduce [2], [3], [4] programming model that is introduced firstly by Google File System (GFS) [5] in 2004. MapReduce is a framework for processing highly distributable problems across huge data sets using a large number of computers (nodes), collectively referred to as a cluster.

The work presented in this paper proposes two optimization techniques to Hadoop framework for processing/handling massive number of small-sized images. This enables extreme parallel processing power of Hadoop to be applied to contents of massive number of image files. However, there are two major problems in doing so. First, image files need to be modeled in an appropriate format as a complete entity, and second, they need to be adapted to the mappers in order to utilize parallelization of Hadoop framework. As a solution to the first problem, an input file format called ImageFileInputFormat is developed. A new record generator class called ImageFileRecordReader is also developed in order to read/fetch the content of image and create whole image pixel data as an input record to MapTask [6].

Regarding the second problem, we develop two approaches. First approach is based on combining multiple small size files into a single Hadoop SequenceFile. SequenceFile is created by ImageFileRecordReader class, which is developed as an extension to Hadoop. Second approach proposes a technique to combine many images as a single input to MapTask without merging. This technique does not require special input file format as SequenceFile, so that images can be used as in their original format. To achieve this, we introduce novel image input format and an image record reader, which is MultiImageRecordReader class, to fetch the image content into image processing library. The architectural details are presented in section 3. These two approaches together with the naive approach are going to be applied on distributed face detection applications on images. The effectiveness of the proposed techniques is proven by the test cases and performance evaluations.

Remaining of this paper is organized as follows. Section 2 presents related works. Section 3 explains the proposed architecture. It covers extending Hadoops application programming interfaces to effectively manipulate images. Section 4 evaluates the performances of techniques on a sample scenario. Section 5 gives the summary and conclusion.

2 Related Work

HDFS is specialized in storing and processing large-size files. Small-size files storage and processing ends up with performance decrease in HDFS. NameNode is the file system manager in HDFS master node which registers file information as metadata. When using massive number of small-size files, the memory usage of Namenode increases so leading master node to be unresponsive for file operation requests from client nodes [7]. Moreover, number of tasks to process these files increases and Hadoop JobTracker and TaskTrackers, which initialize and execute tasks, have more tasks to schedule. In that way, total HDFS job execution performance decreases. For these reasons, storing and processing massive number of images require different techniques in Hadoop. Dong et al. propose two techniques for this problem given in [7] and [8]. In [7] they propose firstly, file merging and perfecting scheme for structurally related small files, and secondly, file grouping and perfecting for logically related small files. Their approach is based on categorization of files based on their logical or structural properties. In [8], they propose another similar approach on the same problem. They introduce a two-level perfecting mechanism to improve the efficiency of accessing small files, and use power point files as a use case scenario. On the other hand, we tackle this problem by introducing a new input file format and a new record generator class in order to read the content of images and create whole image data as an input record to MapTask. Hadoop [3] is based on a parallel programming paradigm MapReduce employing a distributed file system for implementation on very large clusters of low performance processors aimed at text based searching. Although it has been mainly utilized for textual data collections such as crawled web documents and web logs, later it has been adopted in various

types of applications. For example, it has been used for satellite data processing [4], bioinformatics applications [5], and machine learning applications [6].

There are a few example usages of Hadoop in image processing. These are mostly implementations of content-based image searching. Golpayegani and Halem [9] adopted Hadoop in satellite image processing. They propose a parallel text-based content searching. So, each image is annotated with some textual information after fetched by the satellites. Similarly Krishna et al. [10] proposes a Hadoop file system for storage and MapReduce paradigm for processing images crawled from the web. The input to the whole system is a list of image URLs and the contextual information aka the metadata of the image. Searching is done over the metadata of the images. The only challenge in such application is defining key and value pairs and as well as defining map and reduce functions. The other issues are handled by Hadoop core system. The architecture presented in [9] and [10] are called hybrid architectures, because they use text-based annotated data for searching and they access the result images by a URL defined in content data. Afterwards, they can process the image and redefine their metadata and save it with the current version. There is also another type of use cases of Hadoop as in [11]. Kocakulak and Temizel [11] propose a map reduce solution using Hadoop for ballistic image comparison. Firearms leave microscopic markings on cartridge cases which are characteristic to each firearm. By comparing these marks, it is possible to decide whether these two cartridge cases are fired from the same firearm or not. The similarity scores returned by the algorithm included similarity score for each part of the cartridge case and their weighted sum. All correlation results were passed by map tasks to the partitioner as key/value (k, v) pair. The key and the value constituted the id of the cartridge case and the similarity score object respectively. The work presented in this paper is different from [9] and [10] because they use hybrid model to search images, i.e., images are annotated with some textual information enabling content-based searching. In our model, we use images in their pure formats, and instead of text search we utilize advanced face detection algorithm by comparing pixel information in the images. In addition, since we are doing face recognition, we cannot cut the images (mapping) into small parts as in [11]. If we cut the images we can degrade the success of the system. In other words, we might possibly cut the image at a place where a face might be located. So, we keep the images as a whole and propose a different approach as given in Section 3.

3 Architecture: Hadoop Optimization for Massive Image Processing

3.1 Distributed Computing with Hadoop

HDFS is a scalable and reliable distributed file system consisting of many computer nodes. The node running NameNode is the master node and nodes running DataNode are worker nodes. DataNodes manage local data storage and report feedbacks about the state of the locally stored data. HDFS has only one NameNode but can have thousands of DataNodes.

Hadoop uses worker nodes as both local storage units of file system and parallel processing nodes. Hadoop runs jobs parallel by using MapReduce programming model. This model consists of two stages which are Map and Reduce whose input and outputs are records as <key, value> pairs. Users create jobs by implementing Map and Reduce functions and by defining the Hadoop job execution properties. After having defined, jobs are executed on worker nodes as MapTask or ReduceTask. JobTracker is the main process of Hadoop for controlling and scheduling tasks. JobTracker gives roles to the worker nodes as Mapper or Reducer task by initializing TaskTrack-

ers in worker nodes. TaskTracker runs the Mapper or Reducer task and reports the progress to JobTracker.

Hadoop converts the input files into InputSplits and each task processes one InputSplit. InputSplit size should be configured carefully, because InputSplits can be stored more than one block if InputSplit size is chosen to be larger than HDFS block size. In that way, distant data blocks need to be transferred over network to MapTask node to create InputSplit. Hadoop map function creates output that becomes the input of the reducer. So, the output format of the map function is same with the input format of the reduce function. All Hadoop related file input formats derive the FileInputFormat class of Hadoop. This class holds the data about InputSplit. InputSplit does not become input directly for the map function of the Mapper class. Initially, InputSplits are converted into input records consisting of <key, value> pairs. For example, in order to process text files as InputSplit, RecordReader class makes text lines of the file as an input record in <key, value> format where key is the line number and value is the textual data of each line. The content of the records can be changed by implementing another derived class from RecordReader class.

In distributed systems, the data to be processed is generally not located at the node that processes that data and this situation causes performance decrease in parallel processing. One of the ideas behind the development of HDFS is making the data processed in the same node where it is stored. This principle is called data locality which increases the parallel data processing speed in Hadoop [6].

Using massive number of small size files causes shortage of memory in master node due to increasing sizes of Namenode?s metadata file. Moreover, as the number of files increases, the number of tasks to process these files increases and the system ends up with workload increases in Hadoop?s JobTracker and TaskTrackers, which are responsible for initialization, execution and scheduling of the tasks. These might lead master node to be unresponsive for file operation requests from client nodes [7]. In brief, storing and processing massive number of images require different techniques in Hadoop. We propose a solution to this problem of Hadoop by an application of face detection.

3.2 Interface Design

In order to apply face detection algorithm to each image, map function has to get the whole image contents as a single input record. HDFS creates splits from input files according to the configured split-size parameter. These InputSplits become the input to the MapTasks. Creating splits from files causes some files to be divided into more than one split, if their file size is larger than the split-size. Moreover, a set of files can become one InputSplit if the total size of input files is smaller than the split size. In other words, some records may not be represented as the binary content of each file. This explains why new classes for input format and record reader have to be implemented to enable MapTask to process each binary file as a whole.

In this paper, ImageFileInputFormat class is developed by deriving the FileInputFormat class of Hadoop. ImageFileInputFormat creates FileSplit from each image file. Because, each image file is not splitted, binary image content is not corrupted. In addition, ImageFileRecordReader class is developed to create image records from FileSplits for map function by deriving Hadoop?s RecordReader class. In that way pixel data of images are easily fetched from Hadoop input splits into image processing tasks (map tasks). After that point, any image processing algorithm can

be applied to image content. In our case, map function of the Mapper class applies the face detection algorithm to image records. Haar Feature-based Cascade Classifier for Object Detection algorithm defined in OpenCV library is used for face detection [12]. Java Native Interface (JNI) is used to integrate OpenCV into interface. Implementation of map function is presented below. "FaceInfoString" is the variable that contains the information about detection properties such as image name and coordinates where faces are detected.

```
Class:          Mapper
Function:       Map
 Map(TEXT key(filename), BytesWritable value(imgdata), OutputCollector)
        getImgBinaryData_From_Value;
        convertBinaryData_To_JavaImage;
        InitializeOpenCV_Via_JNIInterface;
        runOpenCV_HaarLikeFaceDetector;
        foreach (DetectedFace)
                createFaceBuffer_FaceSize;
                copyFacePixels_To_Buffer;
                create_FaceInfoString;
                collectOutput:
                        set_key_FaceInfoString;
                        set_value_FaceImgBuffer;
        end_foreach
```

Hadoop generates names of output files as strings with job identification numbers (e.g.: part 0000). After face detection, our image processing interface creates output files as detected face images. In order to identify these images easily, the output file names should contain detected image name and detected coordinate information (eg: SourceImageName(100,150).jpg). ImageFileOutputFormat class is developed to store output files as images with desired naming. ReduceTask is not used for face extraction because each MapTask generates unique outputs to be stored in the HDFS. Each task processes only one image, creates output and exits. This approach degrades the system performance seriously. The overhead comes from initialization times of huge number of tasks.

In order to decrease the number of tasks, firstly, converting small-size files into single large-size file and process technique is implemented. SequenceFile is a Hadoop file type which is used for merging many small-size files [13]. SequenceFile is the most common solution for small file problem in HDFS. Many small files are packed as a single large-size file containing small-size files as indexed elements in <key, value> format. Key is file index information and value is the file data. This conversion is done by writing a conversion job that gets small-files as input and SequenceFile as output. Although general performance is increased with SequenceFile usage, input images do not preserve their image formats after merging. Preprocessing is also required for each addition of new input image set. Small files cannot be directly accessed in SequenceFile, whole SequenceFile has to be processed to obtain an image data as one element [14].

Secondly, combining set of images as one InputSplit technique is implemented to optimize small-size image processing in HDFS. Hadoop CombineFileInputFormat can combine multiple files and create InputSplits from this set of files. In addition to that, CombineFileInputFormat selects files which are in the same node to be combined as InputSplit. So, amount of data to be transferred from node to node decreases and general performance increases. CombineFileIn-

putFormat is an abstract class that does not work with image files directly. We developed CombineImageInputFormat derived from CombineFileInputFormat [15] to create CombineFileSplit as set of image. MultiImageRecordReader class is developed to create records from CombineFileSplit. This record reader uses ImageFileRecordReader class to make each image content as single record to map function (see technique in Fig.1). ImageFileOutputFormat is used to create output files from detected face images and stored into HDFS.

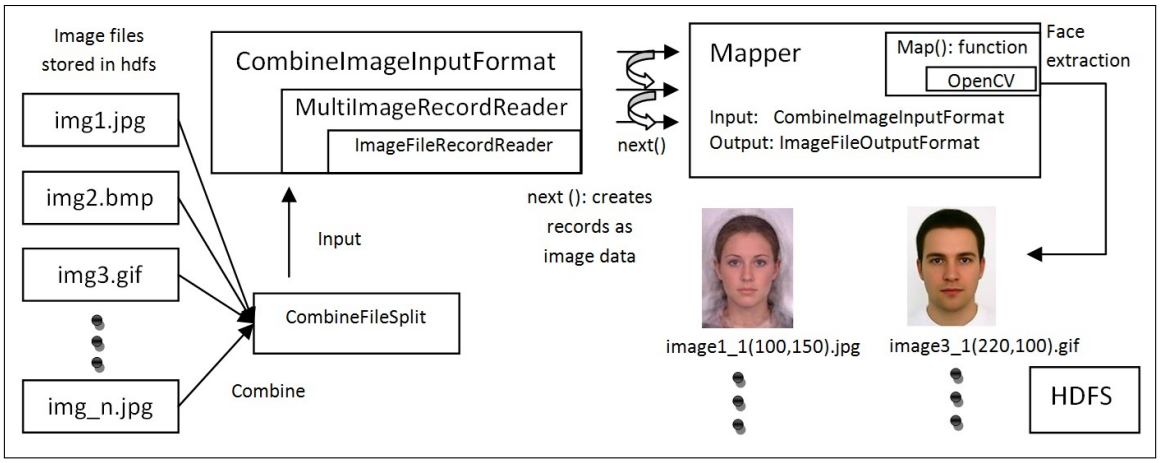


Figure 1: Combine and Process Images Technique

4 Performance Evaluations

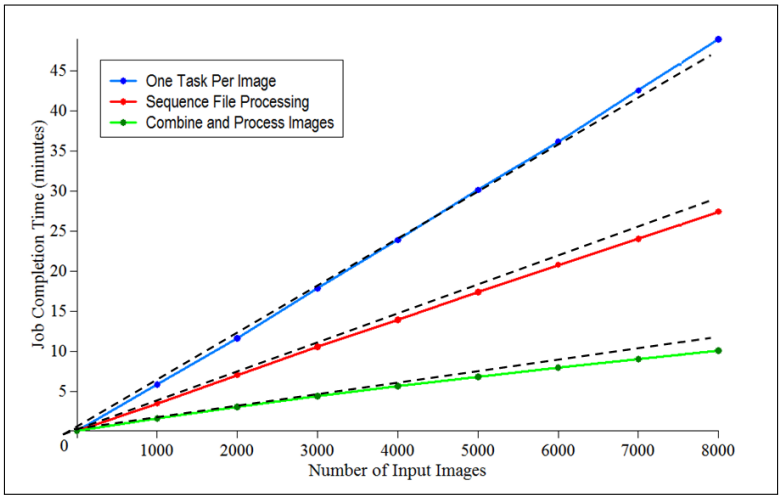


Figure 2: Performance Comparison

To test the system and evaluate the results, we have set up an HDFS cluster with 6 nodes. Face detection jobs are run on a given set of image files on the cluster. HDFS cluster is set up with 6 nodes to run face detection jobs on image sets. Each node has a Hadoop framework installed on a virtual machine. Although virtualization causes some performance loss in total execution efficiency, installation and management of Hadoop become easier by cloning virtual machines. MapTasks require large dynamic memory space when map-function for the image

processing executes. Default Java Virtual Machine (JVM) heap size is not enough for large size images. So, maximum JVM size for Hadoop processes is increased to 600 Mb.

Five different small size images are used as input files. Distribution of the images according to file sizes are preserved in input folders. The images in the input folders went through the face detection job with the three types of approaches in HDFS. These are (1) one task per image brute-force approach (for comparison only), (2) SequenceFile processing approach and (3) combine and process images approach (see performance results in Fig.2).

5 Conclusion

The effectiveness of the proposed technique has been proven by the test cases and performance evaluations. As Figure 2 shows, the proposed approach, combine images and then process, has become the most effective method in processing image files in HDFS. The SequenceFile processing is slower than the combining technique due to the fact that CombineImageInputFormat enforces creation of InputSplits by combining images in the same node. Additionally, in SequenceFile approach, InputSplits to be processed in MapTask does not always consist of datablocks in the same node. So some datablocks may be transferred from other storage node to MapTask node. Extra network transfer causes performance loss in total job execution. On the contrary, SequenceFile approach has better performance against the task per image approach, because small number of input files decreases number of created tasks. In that way, job initialization and bookkeeping overheads of tasks are decreased.

The slope of the job completion time curve for Task per image approach has increased as number of input images increases. But slopes of curves of the other two techniques have slightly decreased by increasing number of input images, because task per image approach causes heavy burden on initialization and bookkeeping by increasing number of tasks. On the contrary, number of tasks is not increased as proportional to the number of images in SequenceFile and combine images techniques. A small number of tasks has been able to process more images when the number of input images is increased.

Consequently, image processing like face detection on massive number of images can be achieved efficiently by using the proposed I/O formats and record generation techniques for reading image content into map tasks are discussed in this paper. We also explained the inner structure of map tasks to read image pixel data and process them. In the future, we plan to enhance and apply the proposed technique on face detection in video streaming data.

Bibliography

[1] http://hadoop.apache.org/.

[2] Berlinska, J.; M. Drozdowski. (2011); Scheduling Divisible MapReduce Computations, *Journal of Parallel and Distributed Computing*, 71(3): 450-459.

[3] Dean, J.; S. Ghemawat. (2010); MapReduce: A Flexible Data Processing Tool, *Communications of the ACM*, 53(1): 72-77.

[4] Dean, J.; S. Ghemawat. (2008); MapReduce: Simplified Data Processing on Large Clusters, *Communications of the ACM*, 51(1): 1-13.

[5] Ghemawat, S.; H. Gobioff.; S. T. Leung.(2003); The Google File System, *Proceedings of the 19th ACM Symposium on Operating System Principles*, NY, USA: ACM, DOI:10.1145/945445.945450.

[6] White, T. (2009); The Definitive Guide. 2009: O'Reilly Media.

[7] Dong, B.; et al. (2012); An Optimized Approach for Storing and Accessing Small Files on Cloud Storage, *Journal of Network and Computer Applications*, 35(6): 1847-1862.

[8] Dong, B.; et al. (2010); A Novel Approach to Improving the Efficiency of Storing and Accessing Small Files on Hadoop: a Case Study by PowerPoint Files, *IEEE International Conference on Services Computing (SCC)*, Florida, USA: IEEE, DOI:10.1109/SCC.2010.72.

[9] Golpayegani, N.; M. Halem. (2009); Cloud Computing for Satellite Data Processing on High End Compute Clusters, *IEEE International Conference on Cloud Computing*, Bangalore, India: IEEE, 88-92, DOI:10.1109/CLOUD.2009.71.

[10] Krishna, M.; et al. (2010); Implementation and Performance Evaluation of a Hybrid Distributed System for Storing and Processing Images from the Web, *2nd IEEE International Conference on Cloud Computing Technology and Science*, Indianapolis, USA: IEEE, 762-767, DOI:10.1109/CloudCom.2010.116.

[11] Kocakulak, H.; T. T. Temizel. (2011); MapReduce: A Hadoop Solution for Ballistic Image Analysis and Recognition, *International Conference on High Performance Computing and Simulation (HPCS)*, İstanbul, Turkey, 836-842, DOI:10.1109/HPCSim.2011.5999917.

[12] http://opencv.org

[13] http://wiki.apache.org/hadoop/SequenceFile

[14] Liu, X.; et al. (2009), Implementing WebGIS on Hadoop: A Case Study of Improving Small File I/O Performance on HDFS, *IEEE International Conference on Cluster Computing and Workshops*, Louisiana USA: IEEE, 1-8, DOI:10.1109/CLUSTR.2009.5289196.

[15] https://hadoop.apache.org/docs/current/api/org/apache/hadoop/mapred/lib/ CombineFileInputFormat.html

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):672-685, December, 2014.

Auto Adaptive Identification Algorithm Based on Network Traffic Flow

S. Dong, X. Zhang, D. Zhou

Shi Dong*
1. School of Computer Science and Technology, Zhoukou Normal University
Zhoukou, 466001, China
2. School of Computer Science & Technology, Huazhong University of Science and Technology
Wuhan, 430074, China
*Corresponding author: njbsok@gmail.com

Xingang Zhang
School of Computer and Information Technology, Nanyang Normal University
Nanyang, 473061, China
zxg@nynu.edu.cn

Dingding Zhou
Department of Laboratory and Equipment Management, Zhoukou Normal University
Zhoukou, 466001, China
zdd@zknu.edu.cn

Abstract: Traffic identification is a key task for any Internet Service Provider (ISP) or network administrator. Machine learning method is an important research method on traffic identification, while impact of the asymmetry router on the traffic identification is considered, so this paper analyzes the impact of asymmetry routing on traffic identification, and proposes an effective method to decrease the impact, and experimental results show the auto adaptive algorithm can improve the traffic identification.
Keywords: Traffic identification, Internet Service Provider (ISP), Auto Adaptive algorithm (AA), asymmetry routing.

1 Introduction

Traffic identification play an important in many fundamental network operations and maintenance activities to detect invade and malicious attacks forbid applications, bill on the content of traffics and ensure quality of service. It increasingly becomes one of the most interesting topics in network science and technology fields, especially in recent years. The current network traffic identification methods roughly five categories: (1) port-based method; (2) based on deep packet inspection (dpi) methods; (3) based on the network flow characteristic; (4) based on host behavior [1]; (5) based on machine learning methods.

The machine learning methods are divided into supervised and unsupervised machine learning. These are the more classic identification method; of course, there is also individual QOS quality of service features for identification [2]. Many share a naive assumption about the Internet that traffic on a given link is approximately symmetric, meaning that both directions of a conversation flow across the same physical link. Many developers even embed this assumption in their traffic classification tools [3, 4]. In fact, except at network edges, Internet traffic is often routed asymmetrically [5], which will impair or invalidate the results of tools and models that assume otherwise. An important cause of this asymmetry is "hot-potato routing" [6], the business practice of configuring traffic crossing one's network to exit as soon as possible, minimizing resource consumption, and thus cost, of one's own infrastructure. Particularly common in commercial settlement-free peering agreements, hot-potato routing implies that the network on the

receiving side of a packet will bear higher cost per received packet. The underlying assumption is that if both networks in a settlement-free peering agreement follow this practice, it will even out, and both sides will share evenly in carrying traffic exchanged by their customers. Another cause of asymmetric traffic is link redundancy, or alternative paths within networks. Since routing decisions occur independently for each packet, load-balancing algorithms may cause packets destined to the same endpoint to follow different paths. Other traffic engineering techniques, e.g., policy-based SPF (Shortest Path First), may also induce asymmetry in internal routing state of large provider networks, through studying on asymmetric routing, we found it had some impacts on traffic identification, and we propose auto adaptive (AA) method to improve traffic identification. Experiments results show that the AA method can achieve better accuracy than others.

The paper is structured as follows: Section 2 introduces related work of traffic identification; Section 3 proposes AA algorithm and evaluation method; in Section 4, at last, we list the proportion results which are classified by our identification algorithm, and analyze the impact of ε on traffic identification; Section 5 concludes the paper.

2 Related work

The application identification problem has been changing due the efforts of two factors that are in a continuous competition. On the one hand, the applications, and especially those that do not want to be detected (e.g., P2P applications), in order to use the network resources without control. On the other hand, a group of network operators, investigators and even ISPs who need to know the traffic characteristics of their networks to manage the resources or even charge the users depending on their consumption.

2.1 Research on traffic identification

It has become a hot research between domestic and foreign experts who take the traffic identification as research direction, which proceed distinguish, QOS, intrusion detection, traffic monitoring, billing and management. From the beginning of the study on port-based method, this method is the use for marking and identifying the traffic type by fixed port which supplied by the IANA, the other method is aim at P2P and some certain protocols, which adopt method based on deep packet detection methods, but this method has defect that can't get some encrypted information and can't get the new service type. Recently traffic identification has new method with a number of new applications come out. With appearance of the new service, the method of machine learning has been applied to the traffic identification. Identify fields on the flow, roughly divided into three research directions: one is the feature selection algorithm [7, 8], the other is identification algorithm [1,2,9], another is a category for different types of data sets, for example, all packets can be divided into flows [10–14] that are sampling NETFLOW [15]. Complementary information about related work in the field of traffic identification can be found in the survey of traffic identification techniques using machine learning in [16], in the comparison of contemporary classification methods in [13], the survey on Inter- net traffic identification in [17] and the research review on traffic identification in [18]. A critical but constructive analysis of the field of Internet traffic identification is proposed in [19], focusing on major obstacles to progress and suggestions for overcoming them. Although some articles have been studied on the identification algorithm, but the identification algorithm still exist some problems to be needed to solve, such as the neural network identification algorithm is one point worthy of study. All previous research studies in traffic identification either use insufficient network data, usually non-public, or use very few/meaningless metrics for evaluation, making it impossible to compare results shown in

different papers [17]. In addition to features selection based on flow, especially the impact of the size of packet traffic is always to be concerned. Therefore, in this article we propose AA method, and we analyze different feature metric set (bidirection feature or unidirection feature) cause different identification results.

2.2 Asymmetry routing

For a pair of hosts A and B, if the path from A to B (forward direction) is different from the path from B to A (reverse direction), we say that the pair of paths between A and B exhibit routing asymmetry. This scenario can be very common in the Internet core where asymmetric routing is an usual practice [20, 21], this asymmetry in the Internet can appear on both as level and router level paths. In fact, the path followed by packets exchanged between end points along one direction can be different from the one followed by packets going in the opposite direction. Recent reports suggest that asymmetrical routing might be moving closer to the edge of the internet than one might expect. For example, the analysis presented in [22] argues that this practice is nowadays quite common even in ISPs directly serving campus-wide networks.

2.3 Flow metric

Definition 1. The definition of flow metric, which is composed with traffic statistical feature such as flow length, flow during etc. These features have high correlation with application type. So considered as flow metric to classify traffic by machine learning. While nowadays there are two kinds of flow metric, one is unidirectional flow metric, and the other is bidirectional flow.

Unidirectional flow metric

Uniflow (Unidirectional flow)(or one-way) within your network is most likely the result of an incorrect configuration, but may also be symptomatic of a larger problem related to your overall routing architecture. Since network communications are bi-directional in nature, unidirectional traffic patterns on your network mean that the traffic flow in one direction is not following the same path as the other. By design, the least cost route to a destination should also be the desired return path. Uniclassifier (Unidirectional classifier) is classifier which use unidirectional flow metric for training set. Where unidirectional flow metric is adopted as table 1 in this paper.

Bidirectional flow metric

Biflow(Bidirectional flow): A biflow is a Flow as defined in the IPFIX Protocol document [RFC5101], composed of packets sent in both directions between two endpoints. A biflow is composed from two uniflows such that:
1.the value of each Non-directional Key Field of each Uniflow (Unidirectional flow) is identical to its counterpart in the other, and
2.the value of each Directional Key Field of each uniflow is identical to its reverse direction counterpart in the other. Biclassifier(bidirectional classifier) is classifier which use bidirectional flow metric for training set. Where bidirectional flow metric is adopted as table 2 in this paper.

Table 1: unidirectional flow feature

Feature	Feature Description
lport	low port number
hport	high port number
duration	Flow duration
Transproto	Stream transport protocol used (TCP / UDP)
TCPflags	TCP header flag,transport layer protocol is UDP,the feature is 0
pps	Packets/duration
bps	bytes/duration
Mean packets arrived time	duration/packets
tos	TOS from NETFLOW
Mean packet length	bytes/packets

Table 2: bidirectional flow feature

Feature	Feature Description
lport	low port number
hport	high port number
duration	Flow duration
Transprotocol	Stream transport protocol used (TCP / UDP)
TCPflags1	TCP header flag,transport layer protocol is UDP,the feature is 0
TCPflags2	TCP header flag,transport layer protocol is UDP,the feature is 0
pps	Packets/duration
bps	bytes/duration
Mean packets arrived time	duration/packets
Bidirectional Packets ratio	Forward packets/ backward packets
Bidirectional Bytes ratio	Forward bytes/ backward bytes
Bidirectional Packet length ratio	Bidirectional packets length ratio
Bidirectional packets	Forward packets + backward packets
Bidirectional bytes	Forward bytes + backward bytes
tos	Bidirectional TOS OR from NETFLOW
Mean packet length	Bidirectional bytes/Bidirectional packets

3 Methodology

3.1 Auto Adaptive algorithm (AA)

In this paper, we propose an algorithm which can auto adjust the flow metric to adapt the traffic identification. The algorithm is called auto adaptive algorithm(AA). The algorithm's core thought is that different traffic can select different classifier with different flow metric (unidirectional flow or bidirectional flow).
Suppose there are n flow samples, each sample has p features, then construct the n*p flow matrix, as follows:

$$A = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ \vdots & & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix} \tag{1}$$

When features number p of the samples are very large which enlarge dimensions of the sample, theoretically, having more features should result in more discriminating power. However, practical experience with machine learning algorithms has shown that this is not always the case. Many learning algorithms can be viewed as making an (biased) probability estimate of a set of features with the class label. This is a complex, high dimensional distribution. Asymmetric routing existing will impact on the traffic identification. So we can consider to adopt auto adaptive method to do with it. In order to depict the method, we have to introduce the H which represent the threshold.

$$H = \frac{Bidirection\_flow\_number}{total\_flow\_number} \tag{2}$$

Definition 2. Optimal threshold: which is used to evaluate the traffic accuracy, it is minimum threshold. When the traffic accuracy is maximum. H is optimal threshold ε.

According to different H, and select H as optimal threshold to enable to obtain the best traffic results, where H is random variable. When $H < \varepsilon$, it will choose unidirectional flow and generate the unidirectional classifier, conversely, it will choose directional flow and generate the directional classifier.

Algorithm 1: AA algorithm

```
// Initialize in the network
A = 0;
for each flowi ∈ [flow1, ..., flown] do
    if H < ε then
        choose unidirectional flow;
    if H >= ε then
        choose bidirectional flow;
        Return the network;
    else
        Goto exit
```

Algorithm AA presents the two kinds of flow metric. The sequence of steps that we show in Figure 1. The procedure mainly set two kinds of dataset for training and testing data set. With these data, we choose AA algorithm to train and test data. The process of machine learning identification is shown in Figure 2:

1.Collecting traffic(Input): Collecting network data from network traffic
2.Selecting traffic features and training data for building traffic classification model(Data Processing): Optimal selecting the known traffic features through the traffic feature selection algorithms. In this paper we only adopt two kinds of feature metric(unidirectional metrics and bidirectional metrics), so extra feature selection method is not added. The traffic classification model is built by training data.
3.Classified the traffic by machine learning algorithm (Output): Using the machine learning identification algorithm to classify network traffic data and generate flow with label.

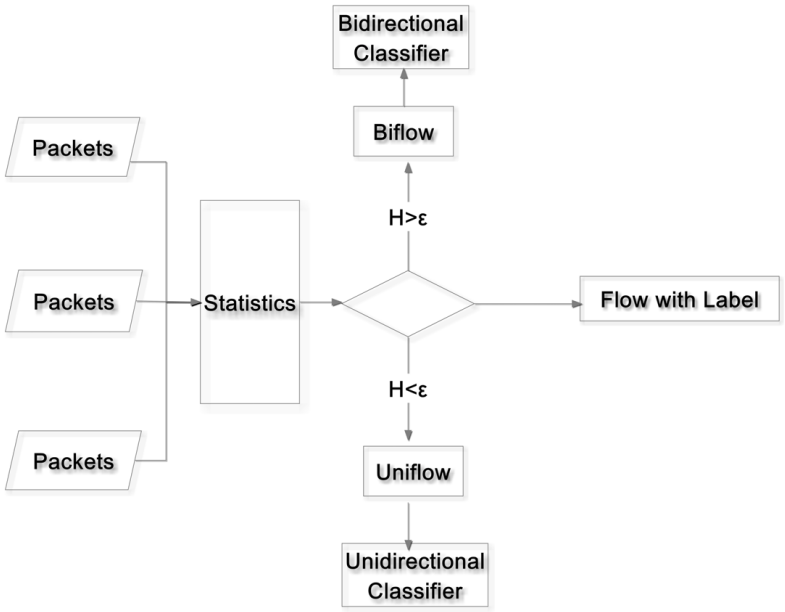


Figure 1: Traffic identification process of AA method

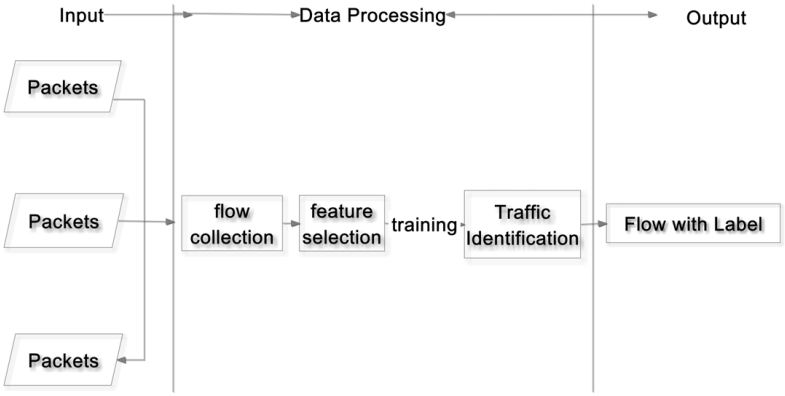


Figure 2: Process of Machine learning, traffic identification

3.2 Algorithm Evaluation

In this paper, we use the routine evaluation standard for verifying the effectiveness of our identification algorithm. The effectiveness of the current flow identification algorithm has the

Table 3: NOC_SET dataset

AppID	Application	Protocal	Flow number	Proportion(%)
1	WWW	HTTP	4943	64.6
2	Bulk	FTP	39	0.5
3	Mail	IMAP,POP3,SMTP	91	1.19
4	P2P	BitTorrent,eDonkey,Gnutella,XunLei	1414	18.5
5	Service	DNS,NTP	433	5.7
6	Interactive	SSH, CVS, pcAnywhere	6	0.08
7	Multimedia	RTSP,Real	20	0.3
8	Voice	SIP,Skype	276	3.6
9	Others	games, attacks	431	5.6

following three concepts evaluation criteria. And the concepts involved are as follows:
-TP (true positive): The flows of application A are classified as A correctly, which is a correct result for the identification;
-FP (false positive): The flows not in A are misclassified as A. For example, a non-P2P flow is misclassified as a P2P flow. FP will produce false warnings for the identification system;
-FN (false negative): The flows in A are misclassified as some other category. For example, a true P2P flow is not identified as P2P. FN will result in identification accuracy loss.

The calculating methods are as follows:
1. Precision: The percentage of samples classified as A that are really in class A

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

2. Recall: The percentage of samples in class A that are correctly classified as A

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

3. Overall accuracy: The percentage of samples that are correctly classified

$$Overallaccuracy = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} (TP_i + FP_i)} \tag{5}$$

4 Experiment

4.1 Dataset

NOC_SET dataset

In order to validate the method and analyze the impact factor,we adopt NOC_SET as dataset.as shown from table 3. We collected data at southeast university,and the collecting site is a 10G backbone channel on Jiangsu Province border of CERNET. We adopt DPI method to mark flow and generate NOC_SET dataset,and use ourself l7_filter_modify software to label the flow.l7_filter_modify is developed based on L7filter [23], at last, we generate NOC_SET dataset.

LBNL_SET dataset

Table 4: LBNL_SET dataset

AppID	Category	flow number	Proportion
1	80	15000	47.69%
2	110	1400	4.45%
3	25	1350	4.29%
4	139	3300	10.49%
5	993	400	1.27%
6	443	10000	31.8%

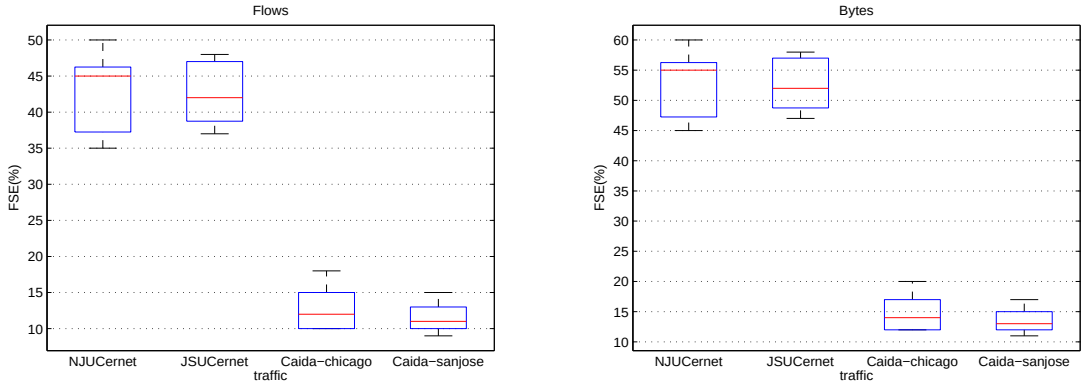
This LBNL_SET data is randomly sampled in several different periods from one node on the internet. The LBNL traffic traces are collected at the Lawrence Berkeley National Laboratory under the enterprise tracing project [24]. The packet traces are obtained at the two central routers of the LBNL network and they contain more than one hundred hours of traffic generated from several thousand internal hosts. The traffic traces are public, but they are completely anonymized, so ascertaining the "ground truth" on the application behind each recorded flow is not possible. Therefore, for this set, we built protocol sets according to the TCP destination port number of each flow, an accepted practice in these cases [25]. We use the traffic traces captured on January 6 and 7, 2005 to obtain the training and the optimization sets. Once again we perform the training by using the most frequently used port numbers in the dataset. Detail $LBNL\_SET$ dataset is shown in table 4.

CAIDA dataset

We built this data set starting from three hour long traces obtained by the Cooperative Association for Internet Data Analysis (CAIDA) [26], and collect at the AMES Internet Exchange (AIX) along an OC48 link on Mar 24, 2011. We use flows extracted from the first hour (corresponding to the interval 16:15-17:00 UTC) to build the training set the optimization set and from the third hour (18:00-18:10 UTC) to buld the evaluation set. As for the previous set, these traces are also anonymized, so port numbers are used as indicators of each protocol. The selection of flows composing the training, optimization and evaluation sets.

Table 5: CAIDA_SET dataset

AppID	Category	flow number	Flow(%)	packets(%)	bytes(%)
1	80	328091	84.69	81.74	81.58
2	110	11539	0.6	0.24	0.25
3	21	28567	3.32	0.03	0.09
4	25	2648	4.57	2.47	2.72
5	4662	2099	0.79	1.34	1.35



(a) Flows (b) Bytes

Figure 3: Comparison of FSEs for traffic

Table 6: the identification Overall accuracy rate AA, Biclassifier, Uniclassifier

Identification	Overall accuracy
AA	99.6742%
Biclassifier	88.2%
Uniclassifier	89.2%

4.2 Impact of asymmetry router on traffic identification:

In this paper, we adopt experimental data based on the NOC-SET data set and CAIDA datas set, use MATLAB tools, WEKA tools and the corresponding algorithm to identify network traffic data [27]. NOC-SET data firstly divided into two test data were 20% and 80% of the test data, and we compared our method that is AA with Biclassifier and Uniclassifier. In order to evaluate and analyze effectiveness of the method about AA. We study traffic identification distribution. In order to analyze asymmetry router, firstly we should remove from the traces any traffic that is inherently asymmetric, such as UDP and ICMP that do not always expect packet recipients to reply, and which would mislead symmetry comparisons if they appear in different magnitudes across networks. TCP background radiation, such as network scanning and probing, can also be a substantial fraction of total inherently asymmetric flows on some links, although it is usually a much lower proportion of bits. We adopt Flow-based Symmetry Estimator(FSE) [28] to evaluate impact degree on traffic, which is a simple method estimate the level of routing symmetry from passively measured flow data. From Figure 3 and Figure 4 we can see different traffic have different FSE, and CAIDA traffic is less. It indicated asymmetry router of CAIDA traffic were more obvious than NOC-SET.

From Table 6 we can see that overall accuracy of AA method traffic is better than biclassifier and uniclassifier,we adopt AA method to classify traffic based NOC-SET data,and select parameter ε=0.5(detailed analysis shown in session F). The data is divided into 9 categories, respectively, WWW, Mail, Bulk, Service, P2P, Interactive, Voice, Multimedia, Others

Table 6 indicates the AA algorithm achieved better result than Biclassifier and Uniclassifier method, moreover. P2P can be seen from Table 7 and the voice of the precision and the recall has greatly improved. The reason for high accuracy is that the proportion of P2P and voice

Table 7: Identification performance for NOC_SET(Precision and Recall)

Category	Algorithm					
	biclassifier		uniclassifier		AA	
	Precision	Recall	Precision	Recall	Precosiin	Recall
WWW	98%	100%	99%	100%	98.5%	99.2%
P2P	58%	100%	75%	100%	93.7%	91.2%
Mail	83%	91.3%	90%	99%	100%	100%
Service	58.90%	100%	70%	99%	90%	90.4%
Inter	84.5%	100%	87%	100%	80%	100%
Multimedia	100%	75%	90%	80%	60%	100%
Voice	35%	50%	45%	55%	37%	50%
Others	44%	46%	48%	77%	45%	60%

account for set of the total is relatively small,the impact of the identification results reduce to a minimum due to the collection of the specimen Caused by imbalance in the ratio.This paper also build NOC_SET dataset which is constructed by bidirectional flow characteristic.

4.3 Comparison of identification algorithm with NOC-SET dataset

Experimental data for the NOC_SET data set (Table 3 as fellows) The analysis data are actual measured IP trace [29], while the traffic flow exits about 40% biflow. NOC_SET dataset is composed by biflow feature.biflow have more information for traffic identification.if use biclassifier to classify the traffic, then the identification result will be improved. In this section, we compare AA algorithm with biclassifier and uniclassifier. Traffic identification result is shown in Table 7. As shown in Table 7, identification result indicates that AA could achieve better accuracy compared with Biclassifier and Uniclassifier.But observing from Inter and Service, identification accuracy of AA is lower than the other method. From Service to Inter types, precision of biclassifier and uniclassifier method is reduced, while the AA is in increments, so that biclassifier and uniclassifier method is easily affected by the number of training samples, while the AA is not vulnerable to the impact of the training Sample dataset. Among three identification algorithm AA, biclassifier and uniclassifier, the overall accuracy of the AA algorithm is highest.

4.4 Comparison of identification algorithm with CAIDA_SET dataset

The data set used in experimental platform: Experimental data for the CAIDA_SET data set (Table 5 as fellows). The analysis data are actual measured IP trace [29]. The two core links are part of an OC192 Tier1 backbone operated by a commercial ISP in the U.S. The first link connects Chicago and Seattle, monitored at an Equinix data center in Chicago. The other one connects San Jose and Los Angeles, monitored at a datacenter in San Jose. On those links, TCP is responsible for about 50% of flows, which was 85% of packets and 93% of bytes on average.UDP carried about 45% of flows (13% of packets and 6% of bytes). We adopted port-based method to mark Flow and generated CAIDA_SET dataset.while the traffic flow exits about 10% biflow. CAIDA_SET dataset is composed by uniflow feature. Biflow have more information for traffic identification. If use biclassifier to classify the traffic, then the identification result will be improved. In this section, we compare AA algorithm with biclassifier and uniclassifier. Traffic identification result is showed in Table 8.

Table 8: Identification performance for CAIDA_SET(Precision and Recall)

Category	Algorithm					
	biclassifier		uniclassifier		AA	
	Precision	Recall	Precision	Recall	Precision	Recall
80	92%	98%	98%	97%	96.5%	98.2%
110	63%	97%	83%	99%	95.7%	92.2%
21	82%	88.3%	92%	98%	99%	99%
25	60.80%	99%	72%	98%	92%	92.4%
4662	82.4%	99%	89%	98%	82.9%	99.2%
Overall Accuracy	65.72%		94.1342%		95.8921%	

Table 9: Identification performance for LBNL_SET(Precision and Recall)

Category	Algorithm					
	biclassifier		uniclassifier		AA	
	Precision	Recall	Precision	Recall	Precision	Recall
80	96%	98%	97%	93%	96.5%	98.2%
110	78%	90%	85%	90%	92.5%	83.2%
25	88%	82.7%	89%	87%	97%	99%
139	59.80%	98%	78%	92%	93%	91.6%
993	86.5%	99%	79%	99%	87%	99%
443	88.5%	99%	89%	99%	84%	99%
Overall Accuracy	68.83%		93.237%		95.861%	

As shown in Table 8, identification result indicates that AA could achieve better accuracy compared with biclassifier and uniclassifier. According to analysis of 4.4 section on traffic result, we can see CAIDA exists the same phenomena which is unbalance sample data. So that biclassifier and uniclassifier method is easily affected by the number of training samples, while the AA is not vulnerable to the impact of the training Sample dataset. Among three identification algorithm AA, biclassifier and uniclassifier, the overall accuracy of the AA algorithm is highest.

4.5 Comparison of identification algorithm with LBNL_SET dataset

We obtained LBNL data from the Lawrence Berkeley National Laboratory, and construct the bidirectional and unidirectional flow metric. We respectively train the two metrics and generate biclassifier and uniclassifier. We compute H value the formula 2 in section 3, and adopt AA method to select classifier which is uniclassifier or biclassifier. The experimental results is shown in table 9. From the results we can see uniclassifier and uniclassifier method is affected by unbalance sample data, while AA method can overcome the problem and improve traffic identification results.

4.6 Impact of ε on traffic identification

In this paper we propose AA method to auto adaptive select classifier(biclassifier or uniclassifier), while threshold ε is a parameter of AA method.ε decide classifiers which were selected, so it is very important for traffic identification. In this section, we will analyze the impact of ε on traffic identification. Detailed experiment method is adopting AA method proposed by varying from ε[0.1,1] based on three dataset(NOC_SET, CAIDA, LBNL_SET). From Figure 4 we can

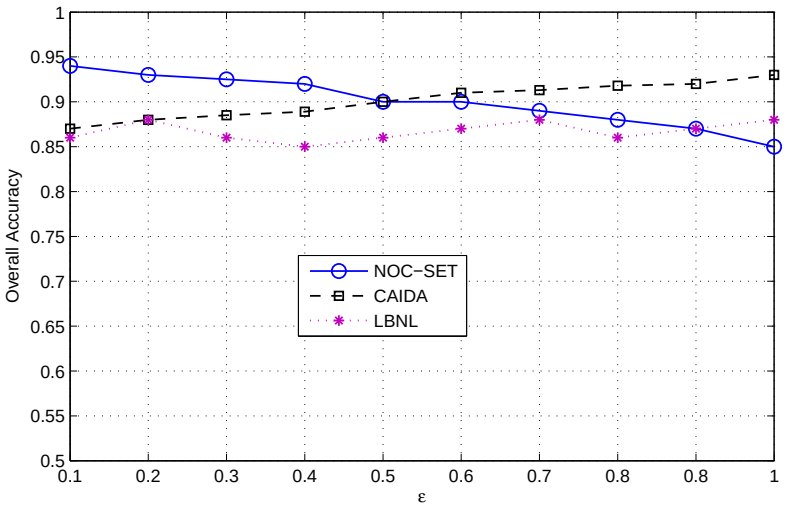


Figure 4: The identification results with ε

see overall accuracy of CAIDA and NOC_SET have biggest change happened when ε vary from 0.1 to 1. Overall accuracy of CAIDA shows an increasing tendency, while NOC_SET is descending. The possible reasons why is that CERNET network contain more symmetry routing, while asymmetry routing is less. Collection point of CAIDA data exist more asymmetry routing. Thus when threshold ε is very small, more opportunity will be selected by biclassifier. Just as mentioned that collection point of NOC_SET is CERNET network containing more symmetry routing, which will have more bidirectional flow metrics, so NOC_SET showed an descending tendency and when ε =0, overall accuracy is maximum.ε =0.5, overall accuracy of CAIDA and NOC_SET is equal. LBNL have not obvious asymmetry routing. So overall accuracy is gentle.

5 Conclusion

In this paper we propose auto adaptive algorithm, and on this basis, the introduction of biclassifier and uniclassifier, and adopt the improved AA method to classify traffic for MOORE_SET as data set, moreover, compare with two other methods which is the biclassifier and uniclassifier method, the results show that, AA method are greatly improved on identification accuracy, to further prove AA method is effective, this paper collect the data in Jiangsu provincial network border and organize trace into flow record such as data sets NOC_SET, the experimental results show that: AA method has high identification accuracy,and we analyze the impact of ε on traffic identification and find ε=0.5 which can be considered as the fixed value, traffic results will be better.

Acknowledgments

This paper is supported by Education Department of Henan Province Science and Technology Key Project Funding (14A520065) and Research Innovation of Zhoukou Normal University (zknuA201408).

Bibliography

[1] T. Karagiannis, K. Papagiannaki, M. Faloutsos (2005); Blinc: multilevel traffic classification in the dark, in: *ACM SIGCOMM Computer Communication Review*, ACM, 35: 229–240, DOI:10.1145/1080091.1080119.

[2] A. Moore, K. Papagiannaki (2005); Toward the accurate identification of network applications, *PAM'05 Proceedings of the 6th international conference on Passive and Active Network Measurement*, 41–54.

[3] A. Moore, D. Zuev (2005); Internet traffic classification using bayesian analysis techniques, in: *ACM SIGMETRICS Performance Evaluation Review*, ACM, 33:50–60, DOI:10.1145/1064212.1064220.

[4] L. Bernaille, R. Teixeira, K. Salamatian (2006), Early application identification, in: *Proceedings of the 2006 ACM CoNEXT conference*, ACM, DOI:10.1145/1368436.1368445.

[5] Wolfgang John, Sven Tafvelin (2007); Differences between in- and outbound internet backbone traffic, in: *Proceedings of Terena Networking Conference*, TERENA, 1-14.

[6] Hotpotatorouting, http://en.wikipedia.org/wiki/Hot-potato_routing.

[7] N. Williams, S. Zander, G. Armitage, Evaluating machine learning algorithms for automated network application identification, Center for Advanced Internet Architectures, CAIA, *Technical Report* 060410B, DOI:10.1.1.84.7170.

[8] N. Williams, S. Zander, G. Armitage (2006), A preliminary performance comparison of five machine learning algorithms for practical ip traffic flow classification, *ACM SIGCOMM Computer Communication Review* 36(5):5–16, DOI: 10.1145/1163593.1163596.

[9] Z. Li, R. Yuan, X. Guan (2007), Accurate classification of the internet traffic based on the svm method, in: *Communications, 2007. ICC'07. IEEE International Conference on*, IEEE, ,1373–1378, DOI: 10.1109/ICC.2007.231.

[10] P. Teufl, U. Payer, M. Amling, M. Godec, S. Ruff, G. Scheikl, G. Walzl (2008), Infect-network traffic classification, in:*Networking, 2008. ICN 2008. Seventh International Conference on*, IEEE, 439–444, DOI: 10.1109/ICN.2008.42.

[11] T. Kiziloren, E. Germen (2007), Network traffic classification with self organizing maps, in: *Computer and information sciences*, 2007. iscis 2007. 22nd international symposium on, IEEE, 1–5, DOI: 10.1109/ISCIS.2007.4456852.

[12] Y. Lim, H. Kim, J. Jeong, C. Kim, T. Kwon, Y. Choi (2010), Internet traffic classification demystified: on the sources of the discriminative power, in: *Proceedings of the 6th International COnference*, ACM, DOI: 10.1145/1921168.1921180.

[13] H. Kim, K. Claffy, M. Fomenkov, D. Barman, M. Faloutsos, K. Lee (2008); Internet traffic classification demystified: myths, caveats, and the best practices, in:*Proceedings of the 2008 ACM CoNEXT conference*, ACM, DOI: 10.1145/1544012.1544023.

[14] J. Erman, M. Arlitt, A. Mahanti (2006), Traffic classification using clustering algorithms, in: *Proceedings of the 2006 SIGCOMM workshop on Mining network data*, ACM, 281–286, DOI: 10.1145/1162678.1162679.

[15] V. Carela-Espanol, P. Barlet-Ros, J. Solé-Pareta (2009), Traffic classification with sampled netflow, DOI:10.1.1.390.5780.

[16] T. Nguyen, G. Armitage (2008), A survey of techniques for internet traffic classification using machine learning, *Communications Surveys & Tutorials*, IEEE, 10(4):56–76.

[17] A. Callado, C. Kamienski, G. Szabó, B. Gero, J. Kelner, S. Fernandes, D. Sadok (2009), A survey on internet traffic identification, *Communications Surveys & Tutorials*, IEEE, 11(3):37–52.

[18] M. Zhang, W. John, K. Claffy, N. Brownlee (2009), State of the art in traffic classification: A research review, in:*PAM '09: 10th International Conference on Passive and Active Measurement, Student Workshop*, Seoul, Korea.

[19] A. Dainotti, A. Pescape, K. Claffy (2012), Issues and future directions in traffic classification, *Network*, IEEE, 26(1):35–40.

[20] Z. Mao, L. Qiu, J. Wang, Y. Zhang (2005), On as-level path inference, in: *ACM SIGMETRICS Performance Evaluation Review*, ACM, 33:339–349.

[21] Y. He, M. Faloutsos, S. Krishnamurthy (2004), Quantifying routing asymmetry in the internet at the as level, in: *Global Telecommunications Conference*, GLOBECOM'04. IEEE, 3: 1474–1479.

[22] W. John (2008), On measurement and analysis of internet backbone traffic, Thesis for the degree of Licentiate of Engineering, a Swedish degree between M.Sc. and Ph.D., Chalmers University of Technology.

[23] J. Levandoski, E. Sommer, M. Strait, et al.(2008), Application layer packet classifier for linux, http://l7-filter.sourceforge.net/.

[24] *** Lbnl/icsi enterprise tracing project, http://www.icir.org/enterprisetracing.

[25] T. Karagiannis, A. Broido, M. Faloutsos, et al. (2004), Transport layer identification of p2p traffic, in: *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, ACM, 121–134, DOI: 10.1145/1028788.1028804.

[26] *** The cooperative association for internet data analysis(caida), http://www.caida.org.

[27] T. Nguyen, G. Armitage (2006), Training on multiple sub-flows to optimise the use of machine learning classifiers in real-world ip networks, in: *Local Computer Networks, Proceedings 2006 31st IEEE Conference on*, IEEE, 369–376, DOI: 10.1109/LCN.2006.322122.

[28] W. John, M. Dusi, K. Claffy (2010), Estimating routing symmetry on single links by passive flow measurements, in: *Proceedings of the 6th International Wireless Communications and Mobile Computing Conference*, ACM, , 473–478, DOI: 10.1145/1815396.1815506.

[29] *** IP Trace Distribution System, http://iptas.edu.cn/src/system.php.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):686-693, December, 2014.

Mathematical Decision Model for Reverse Supply Chains Inventory

L. Duta, C.B. Zamfirescu, F.G. Filip

Luminita Duta
Automation and Computer Science Department
Valahia University of Targoviste
Romania, 130083, Targoviste, 24, Unirii Ave.
duta@valahia.ro

Constantin-Bala Zamfirescu
Dept. of Computer Science and Automatic Control
Lucian Blaga University of Sibiu
Romania, 550024, Sibiu, 10, Victoriei Ave.
zbc@acm.org

Florin G. Filip
The Romanian Academy
Romania, Bucharest, 010071, 125 Calea Victoriei
ffilip@acad.ro

Abstract: In the reverse supply chain inventory theory, inventory models are concerned with the demand of reusable parts, stock replenishment, ordering cycle, delivery lead time, number of disassembled products, ordering costs. The particularity of these models consists in the occurrence of high uncertainties of the quantity and quality of the returned products and resulting parts. To overcome the problem, an inventory model that incorporates decision variables at proactive and reactive levels is derived and discussed in this paper.
Keywords: Reverse supply chains, decision aid, inventory models, Bayesian networks

1 Introduction

A reverse supply chain is the process of moving End of Life (EOL) products from customer to distributor or manufacturer for the purpose of capturing the market leftover value, or for proper disposal. Reverse supply chains operations have a strong impact on forward supply chain stocks and transportation capacities. Customers act as suppliers in reverse supply chains. After sorting, the collected items will be moved through the reverse supply chain. Some used products are sold for recycling, usually after destroying their physical structure. Others are used as source of spare parts on the secondary market. Another common use of returns is as spare parts for warranty claims in order to reduce the cost of providing these services to customers. Defective and used products are moved to manufacturer for reusing or remanufacturing or for disposal. Remanufactured or refurbished products are sold for additional revenue to a secondary market.

Part of the reverse supply chain, the reverse logistics has to deal with product transportation, production planning and inventory management. In reverse supply chain inventory, two problems might occur: a) a high level of uncertainty provided by the product end-of-life state of the EOL product and b) the irregular supply for remanufacturing or refurbishing. In this respect, two types of inventory replenishment models have been identified in literature: a deterministic model with stationary or dynamical demand and a stochastic model with continuous or periodical review [1]. This paper addresses the stochastic models with variability of the demand.

In the sequel the paper is organized as it follows. The state of art from the second section of this paper describes various reverse supply chain inventory models found in literature. In the third section we present a brief review of supply chain inventory policies and the positioning of our work. In the fourth section, an approach based on a mathematical decision model to deal with the stochastic aspect of the reverse supply chain inventory is described. Decision variables are introduced on two levels: a proactive level and a reactive level. Results and discussions are highlighted in the fifth section. Conclusions and future work perspectives are presented in the last part of the paper.

2 Literature review

Early researches in the domain of interest of this paper concerned the economic order quantity logic which was exploited by using deterministic models with stationary demand. In [2] a survey on quantitative models for reverse logistics is given. In such systems, authors show that the objective of the inventory management is to control the component recovery process to guarantee a required level of maintenance service and to minimize fixed and variable costs.

In [3] Kiesmuller and Sherer determinate, for the first time, an optimal policy for a remanufacturing system with dynamic demand, by assuming that there are no backorders and lead times. In [4], authors develop two heuristic procedures to investigate the effect of stochastic yields on a disassembly to order system. In [5], Imtavanich and Gupta use heuristics to address different stochastic parameters of a *disassembly to order* (DTO) system and they propose a goal programming procedure to calculate the number of returned products that satisfy various goals. A DTO system is concerned with the process of finding how many products have to be disassembled in order to fulfil the demand of reusable parts or subassemblies that will be used in remanufacturing. Bayindira *and et al.* [6] investigate the desired level of recovery under various inventory control policies when the success of recovery is probabilistic. All used and returned items go into a recovery process that is modelled as a single stage operation. The recovery effort is represented by the expected time spent for it. Using brand new parts may represent an alternative or/and complementary solution to ensure the necessary stocks for satisfying the orders received in case the recovered parts do not satisfy the demand. Four inventory control policies that differ in timing of and information used in purchasing decision could be envisaged. The objective is to find the recovery level together with inventory control parameter that minimizes the long-run average total cost. A numerical study covering a wide range of system parameters is carried out. Finally computational results are presented with their managerial implications. A DTO model with deterministic yields is presented in [7]. The DTO system is represented by an integer programming model. The objective function is to minimize the total costs. The authors propose three decision variables: one v_1 for the amount of the end of life EOL products to be acquired in view of disassembling, one v_2 for the amount of new parts to be acquired in order to meet the overall demand and one v_3 for the number of the parts that need to be disposed.

The authors made several assumptions: there is no lead time for acquisition or disassembly, products are completely disassembled, there is a continuous supply of EOL products and the demand for parts is completely fulfilled. In [8], a decisional DTO decision model is presented. The author separated decomposes the decision process into two stages. In the first stage one the goal is to minimize the costs by deciding how many items must be disassembled to fulfil the reusable parts demand. In the second stage the goal is to minimise the number of disposed parts and the procurement costs of the new parts. Another model for DTO systems with stochastic yields is presented by Kongar and Gupta in [9]. The quantity of reusable parts is obtained from the number of EOL products disassembled multiply by a weight coefficient. The method uses models in which linear programming encompasses a degree of stochasticity. Decisions are split

into two groups: proactive decisions and reactive decisions. While proactive decisions determine the number of EOL products to disassemble, reactive decisions take into account the uncertain variables (as the quality of the disassembled part, the lead time or the disassembly time) and their probability of occurrence. Kongar and Ilgin propose a linear physical programming solution based methodology which deals with tangible and intangible financial, environmental and performance measures of DTO systems [10]. Two types of decision depending on the disassembly operations are taken into consideration. If the item (the EOL product) is subjected to resale or storage, a non-destructive disassembly process is performed. Otherwise, if the item is subjected to recycling or disposal, a destructive disassembly is applied [9]. In 2010, Ahiska and King [11] extend Kiesmullerand Sheres' approach [3] by considering different lead times for manufacturing and remanufacturing. Recently, Godichaud [12] demonstrates the relevance of *Bayesian Networks* (BN) in a spare parts inventory model which deals with costs and time uncertainties. The same tool (BN) is used in [13] to highlight the temporal dependences between variables in a model for optimal disassembly policy.

3 Inventory policies for reverse supply chain

The economic order quantity model (EOQ) is one of the most widely known inventory control methods [14]. It is used to set the quantities to order in replenishing inventories so that a trade-off between inventory holdings and ordering/set up costs is achieved. Some assumptions are associated with this model: a) constant and known demand, instantaneous receipt of inventory, b) constant and known time intervals between order placement and receipt of the order, stock outs are avoided by placing orders at the right time. It is also assumed that the unit procurement price remains constant irrespective of the number of units purchased. This model induces *four inventory basic policies* according to the moment of the inventory position review: with continuous review or with periodical review. Used notations: Q = order quantity, S = order-up-to level, s = reorder point, T = review *period* [15]:

- The (s, Q) Policy: Whenever the *inventory position* (inventory level plus quantity on order) drops to a given level limit, s, or below, an order is placed for a fixed quantity, Q.
- The (s, S) Policy: Whenever the inventory position (inventory level plus quantity on order) drops to a given level s, or below, an order is placed for a sufficient quantity to bring the inventory position up to a given level, S.
- The (T, S) Policy: Inventory position is reviewed at regular discrete time moments spaced at intervals of length T time units. At each review, an order is placed for a sufficient quantity to bring the inventory position up to a given level, S.
- The (T, s, S) Policy: Inventory position is reviewed at regular instants spaced at time intervals of length T. At each review, if the inventory position is at level s or below, an order is placed for a sufficient quantity to bring the inventory position up to a given level S. If the inventory position is above s, no order is placed.

Remarks:

- When *T=0* in the *(T, s, S)* policy, one obtains the *(s, S)* policy. So, the *(T, s, S)* policy can be regarded as a periodic version of the *(s, S)* policy, which, in turn, may be viewed as a special case of the *(T, s, S)*.
- The *(T, S)* policy represents a special case of the *(T, s, S)* policy in which s = S

4 The decision model

Since the demand for spare parts is variable and is depending on the needs and the time when an order for replenishment is placed until the replenishment arrives (the lead time), the (T, s, S) policy was considered in this paper. In such a system, the period of review is fixed and the ordered quantity changes as per demand or rate of consumption. The period of review T is decided such as the ordered quantity is economical for purchasing the items. The problem is to coordinate the two processes - disassembly and remanufacturing - so as to meet the demand of items. Another problem is the disassembly depth that deals with how completely a product should be disassembled. In this context, the researcher must weigh not only the costs of the process, whether is destructive or not, but also consider which reusable parts are already in stock, how many will be obtained through disassembly and will be accumulated in inventory and how many parts will have to be disposed of. It is obvious that a reverse chain inventory model has to include different decision variables.

4.1 Notations

N_i^d = Number of items i to acquire and disassemble

P_{ki} = Amount of part k in item i

P_{ki}^r = Amount of part k in item i to reuse

P_{ki}^d = Amount of part k in item i to dispose

P_{ki}^h = Amount of part k in item i to hold (to stock)

C_i^a = Acquisition cost of the item i

C_i^d = Disassemble cost of the item i

C_k^r = Reusing cost of the part k of the item i

C_k^d = Disposal cost of the part k of the item i

C_k^h = Holding cost of the part k of the item i

P_{sc} = Probability of scenario sc occurring

TC = total inventory cost

To simplify the analytical model, the following assumptions are made:
- Only a single type of product to disassembly is considered;
- There is no lead times for acquisition or disassembly;
- EOL items to disassembly are always available;
- A single disassembly scenario ($P_{sc} = 1$) is occurring;
- Products are completely disassembled;
- There are only two types of EOL options: reusing and disposal;
- Two types of disassembly operations are considered: destructive and non-destructive;
- All costs are deterministic and constant;
- The interval to acquire is deterministic (at the first slice of time t);
- The amount of reusable, disposable and holding parts is subjected to uncertainties;
- The number of items to acquire at the second slice of time have a probabilistic distribution;
- The model is periodically reviewed [8].

4.2 The Bayesian network model

To implement the correct model, Bayesian Networks were used so as to determinate all influences and causalities between decision variables. The results will show how information influences decisions and how these decisions cause the change of information.

Bayesian networks (BN) have the ability of capturing both qualitative knowledge through their network structure, and quantitative knowledge through their parameters [14]. A static Bayesian Network can be extended to a Dynamic Bayesian Network (DBN) by introducing relevant temporal dependencies to capture the dynamic behaviors of the system at different moments.

To validate the model, the BayesiaLab® software is used ([16] and [17]). The software is able to seize degrees of probability. Once validated, probabilities are used jointly with the probability distribution for giving a new Probability distribution. BayesiaLab® allows the temporal dimension integration in a Bayesian Network. Thus, a BN can be easily transformed into a DBN. Temporal nodes at instants t and $t+1$ can be represented and connected by temporal arcs. The parameters evolution of the DBN nodes can be so tracked in time.Decision nodes are marked by squares. The amount of part k found in item i influences the amount to reuse, to dispose or to hold. Further, these decisional variables change the number of products to be acquired at the next slice of time (moment $t+1$). The two temporal nodes *Nid(t)* and *Nid(t+1)* are linked by a temporal arc. Figure 1 shows background calculation of the total cost. The objective function is included in the *utility node* TC. In the figure, two decision nodes are represented: Da - decision to acquire used products, Dd - decision to disassembly.

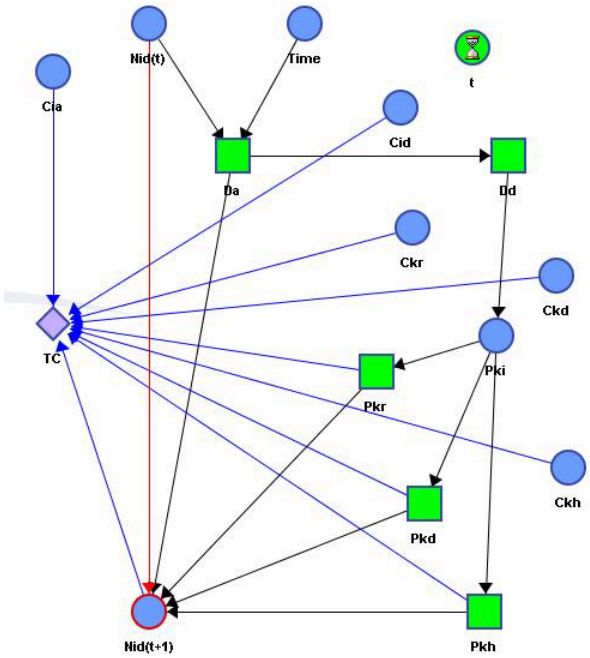


Figure 1: Dynamic Bayesian Network Model

4.3 The mathematical model

The total cost of the disassembly phase (TC) is composed of product acquisition cost, disassembly operations costs, EOL options costs and inventory cost. The value of the total cost is given by the equation (1). The first term is the cost of product acquisition and disassembly so

as to meet the demand of parts for remanufacturing, while the second term is the inventory cost of the disassembled parts. This function is a multi-objective one, since the aim is to find not only the optimal quantity of products to acquire to meet the demand of parts, but also to find the best EOL strategy or scenario so as to optimize the number of parts to reuse in a remanufactured product. For the moment we are interested on quantity to acquire to meet the demand, while scenarios probabilities are introduced by the software to reduce the objective function to a mono-objective one.

The objective function to be minimized is:

$$Min(TC) = Min\left(\sum_i (C_i^a + C_i^d).N_i^d + \sum_i \sum_{sc} P_{sc}(C_k^r.P_{ki}^r + C_k^d.P_{ki}^d + C_k^h.P_{ki}^h)\right) \quad (1)$$

In equation 1 the unknown variables are N_i^d, P_k^r, P_k^d, P_k^h.

This objective function is subjected to the following constraints:

$$d_k \leqslant \sum_i \sum_{sc} P_{sc}.(P_{ki} - P_{ki}^d) \quad (2)$$

$$N_i^d \geq 0 \text{ and integer} \quad (3)$$

$$P_{ki}, P_k^r, P_k^d, P_k^h \geq 0 \text{ and integer} \quad (4)$$

$$\sum_{sc} P_{sc} = 1 \quad (5)$$

Where d_k is the demand (the number of k parts needed). One can note that equations (1) to (5) form a linear integer mathematical model where decision variables can be treated as continuous in order to satisfy the integer value of the demand.

5 Results

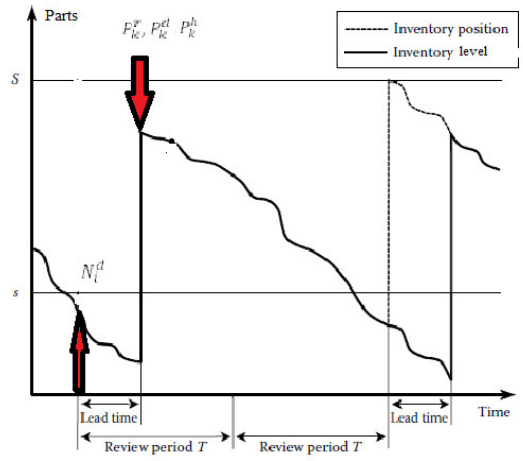


Figure 2: (T, s, S) policy decisions([13], p 117)

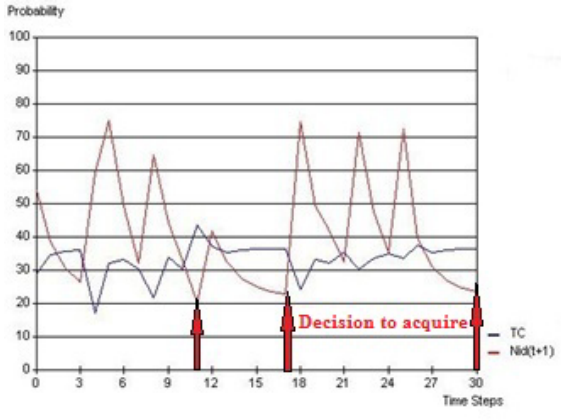


Figure 3: Reactive decisions

Computations were performed with the help of FICO XPRESS® optimization tool [18] which provides a mixed integer solver and framework for constraint integer programming. Supposing a periodic review inventory with a security stock at the beginning of the period, the classical (T, s, S) policy is modeled in figure 2. The planning horizon was fixed to a month, and the review period is established to ten days. Results show that at the end of each review period, the number of disassembled products arrives to a minimal value and the TC to a maximal one. For the EOQ model with constant demand, a decision to acquire products to disassemble is taken whenever the inventory level reaches the reorder point.

Running simulations on the previous model, the decision to acquire is taken before the beginning of each new review period (fig. 3). In other words, starting with a periodical review inventory model, which usually provides orders at constant periods of time, we have reached to a mixed model where orders are given in accordance with the reactive decisions (i.e. in real time).

6 Conclusions and Future Works

An inventory model for reverse supply chains was presented. This model is deduced from the classical inventory model with constant review period and variable demand, and it encompasses decision variables structured on two decisional levels: the proactive level and the reactive level. To implement the approach, DBN was found to be an appropriate decision aid tool. Using a DBN, one can determine the optimal disassembly inventory policy dealing with stochastic aspects of the system.

Proactive decisions aim to determine the initial number of items to disassembly to fulfill the demand of parts. Reactive decisions take into consideration disassembly scenarios and the end-of-life options of the disassembled parts. The model optimizes the quantity of used products to acquire so as to minimize the total inventory cost. Future work will integrate the results above in a decision support system [19]. This issue needs further investigation in real-world settings where the increased cognitive complexity of using different models (i.e. DBN, integer programming), most likely by a collective decision-maker [20], will play a major role in adopting the proposed solution.

Bibliography

[1] Gupta S. M. (2013); *Reverse Supply Chains: Issues and Analysis*, CRC Press, Taylor&Francis

[2] Fleischmann M, Bloemhof-Ruwaard J. M., Dekker R. (1997); Quantitative Models for Reverse Logistics: A Review, *European Journal of Operational Research* , 103: 1-17.

[3] Kiesmuller G P, Scherer C W (2003); Computational issues in a stochastic finite horizon one product recovery inventory model, *European Journal of Operational Research* 146(3): 553-579.

[4] Inderfurth K, Langella I. M. (2006); Heuristics for solving Disassembly to order Problems with Stochastic Yields, *OR Spectrum*, 28 (1): 73-99

[5] Imtanavanich P, Gupta S. M. (2006); Calculating Disassembly Yields in a Multicriteria Decision Making Environment for a Disassembly to Order System, *Application of Management Science*, Elsevier Science, Amsterdam, 12: 109-125

[6] Bayindira Z.P., Dekkerb R, Porrasb E. (2006); Determination of recovery effort for a probabilistic recovery system under various inventory control policies, *The International Journal of Management Science*, 34: 571 - 584.

[7] Inderfurth K, Langella I. M., (2008); *Planning Disassembly for Remanufacture to Order Systems, Environment Conscious Manufacturing*, Gupta and Lambert eds.,CRC Press, Boca Raton, Fla.

[8] Langella I. M (2007), *Planning Demand Driving Disassembly for Remanufacturing*, Deutscher Universitäts-Verlag.

[9] Kongar E, Gupta S M, (2009) Solving the Disassembly to Order Problem Using Linear Physical Programming, *International Journal of Mathematics in Operational Research*, 1(4): 504-531.

[10] Gupta , S.M., Ilgin, M. I. (2012), Physical Programming; A review of the state of the art, *Studies in Informatics and Control*, 21(4): 349-366.

[11] Ahiska S S., King R E (2010); Inventory optimization in a one product recoverable manufacturing system, *International Journal of Production Economics*, 124(1): 11-19.

[12] Godichaud M. (2010); *Outils d'aide à la décision pour la sélection des filières de revalorisation des produits issus de la déconstruction des systemes en fin de vie*. Thèse de doctorat, Université de Toulouse.

[13] Duta L., Addouche S.A. (2012); Dynamic Bayesian Network for Decision Aided Disassembly Planning, *Studies in Computational Intelligence*, Springer, 402: 143-154.

[14] Blumenfeld, D.E. (2008), Operations Research Calculation Handbook, CRC Press, Taylor and Francis Group.

[15] Ghorbel N., Duta L., Addouche S.A., El Mhamedi A. (2011), Decision aided tool for sustainable inventory control, *The 21th International Conference on Production Research*, Sttutgart, Germany.

[16] Conrady S, (2011), *Introduction to Bayesian Networks*, Conrady Applied Science, LLC - Bayesia's North American Partner for Sales and Consulting.

[17] http://www.bayesia.com/en/products/bayesialab.php (last consulted in April 2014)

[18] FICO XPRESS Optimization Suite (http://www.fico.com/en/products/fico-xpress-optimization-suite/)

[19] Filip F.G. (2008), Decision support and control for large-scale complex systems. *Annual Reviews in Control*, 32(1):61-70.

[20] Zamfirescu C.B., Duta L., Iantovics B. (2010),On Investigating the Cognitive Complexity of Designing the Group Decision Process, *Studies in Informatics and Control*, 19 (3):263-27.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):694-702, December, 2014.

Detecting Emotions in Comments on Forums

D. Gîfu, M. Cioca

Daniela Gîfu
"Alexandru Ioan Cuza" University of Iaşi
16 General Berthelot St., Iaşi, 700483, România
daniela.gifu@info.uaic.ro

Marius Cioca
"Lucian Blaga" University of Sibiu
10, Victoriei Bd., Sibiu, 550024, România
marius.cioca@ulbsibiu.ro

Abstract: The paper presents one of the most important issues in Natural Language Processing (NLP), emotion identification and classification to implement a computational technology based on existing resources, open-source or freely available for research purposes. Furthermore, we are interested to use it for establishing Gold standards in sentiment analysis area, such as SentiWordNet. In this sense, we propose to recognize and classify the emotions (sentiments) of the public consumer from the written texts which appeared on the various Forums. We analyse the writing style which refers to how consumers construct sentences together when they write comments to indicate their passion about an entity (persons, brand, location, etc.). We present in this paper a method for integrating Romanian lexical resources from emotional perspective, in developing, which can be used in sentiment analysis. This study is intend to help direct beneficiaries (public consumer, marketing managers, PR firms, politicians, investors), but, also, specialists and researchers in the field of natural language processing, linguists, psychologists, sociologists, economists, etc.
Keywords: sentiment analysis, language resources, emotions levels, semantic classes, Forums.

1 Introduction

In our context, emotion in writing refers to how public consumers express a personal opinion of their experience about entities (products, persons, tourism objectives, etc.). When we say public consumer, actually, we say any commentator who is interested in a range of information about a particular entity.The option for such a topic, known as sentiment analysis (SA) or *opinion mining*[1], encountered in texts circulated on different *Forums*, and comes from the need to clarify descriptive consumer behavior, affected by the amount of promotional messages, regardless of their nature and purpose. At the present time, sentiment analysis is one of the most studied natural language processing (NLP) issues.

The hypothesis of this paper is that by observing the emotional orientation of the commentators over time (visible in writing style) on Forums can help us to build a database with information on topics, services, products, etc. for the public interest, which can serve to implement a NLP tool, useful to predict potential consumer needs.

The paper is structured in five sections. After a brief introduction about the importance of this study, the section 2 mentions some important works focused on SA. The section 3 describes

[1]Opinion Mining originates from the Information Retrieval (IR) community, and aims at extracting and processing users' opinions about entities (products, movies, etc.). Sentiment analysis was initially formulated as the NLP task of retrieval of sentiments expressed in texts. Looking closely, these two issues are similar in their own essence and fall under the area of Subjectivity Analysis.

four units of sentiment analysis some of the most coomonly used in SA, and section 4 describes the our tool functionality. The last section highlights conclusions and mentions the future work, one of the projects of NLP-Group@UAIC-FII.

2 State of the art

Nowadays, Forum becomes a long-term instrument that can consolidate the public sphere, Habermas's concept [9] and civil society. In opposite to the instrumental view of *liberalization* of the Internet, the new dimension can be classified as *environmental*. The ubiquity of Forums affects the marketing mechanisms to respond to the challenges imposed by it. If the landscape of communication becomes denser, more complex and more participative, then the *network population* gets increased access to information, achieving multiple opportunities by engaging in public speech and putting in motion collective actions. But, a problem appears. More information, more opinions reflected mostly in writing style. In fact, any difference in writing reflects the heterogeneity in reviewers culture, education, occupation and so on. This heterogeneity can be quantified in sentiments.

The sentiment is the overall emotion towards the subject matter expressed by the reviewer. In general terms, SA consists of extracting opinions from text. It is assimilated as *subjectivity analysis* [2] or *evaluating affection* [1]. SA defines the processing search results from an article, generating a list of attributes product (quality, characteristics, etc.) and aggregating opinions for each of them (e.g. poorly, good). Moreover, SA has been interpreted as including various types of analysis and evaluation [14], [15], [17], [18].

Another important dimension of SA is researching objectivity in a text, finally resulting a text classification into two classes - objective and subjective -, frequently more difficult to undertake than for a polarity one [16]. In 2001, sentiment analysis was the subject of two researches by Das and Chen [3], and Tong [1], concerned on the opinions on the market sales. Out attention is also take up by the classification of the degree of positivity of a text (document, sentence/clause, etc.), consisting in opinion words (e.g. angry, happy). For instance, in elections, we established two classes, positive and negative, each of them with other three subclasses for determining the intensity of sentiment [7]. Moreover, in the sentiment analysis area there are approaches that consider, also, the neutral class (value 0), assigning words with one value from -5 to +5, with two classes more than the first author [8]. This paper describes a method with a shorter scale of values, from -1 to +1, as the authors are interested to discover the sentiment extracted from their comments.

3 Units of sentiment analysis

SA offers organizations the possibility to monitor opinions about products/ services and their reputation (e.g. measuring feedback with statistical software packages SAS - *Statistical Analysis System*, SPSS - *Statistical Package for the Social Sciences* or *Superior Performing Statistical Software*), on various Forums platforms in real time and to act accordingly.

We describe below four lexical units for SA.

3.1. Document as the unit of analysis

It is the simplest form of SA and assumes that the document contains an opinion on one main message expressed by the commentator. We will stop at two approaches of sentiment analysis from the document.

a) *Supervised* the document must be classified in a finite set of classes, the training data are assigned to each class. This is for the simple case, when there are two classes: positive and

negative. Also, a neutral class can be added or a numeric scale can be considered from which the document has to be reported (for instance, SentiWordNet). Esuli and Sebastiani [6] reports three sentiment scores: positivity, negativity and objectivity. The system learns a classification model based on the training data, using an algorithm of classification, such as SVM (Support Vector Machines) or KNN (K-Nearest Neighbors). Then, this classification is used for mapping new documents in their different sentiment classes. Good precision is achieved even when each document is represented as a bag of words [13].

b) *Unsupervised* the document is based on determining the semantic orientation (SO) of specific phrases. If the average SO of these phrases is above a predefined threshold, the document is classified as positive. Otherwise, it is considered negative. For instance, a set of predefined part-of-speech (POS) models can be used to select those sentences [21] approach taken into consideration in this study - or to create an opinion lexicon structured in words and syntagmas used by the first author since 2009.

3.2. Sentence as the unit of analysis

For a more refined analysis of opinions about an entity (organization, product, political actor, etc.) we must move to the sentence level. It is assumed that there is only one opinion (sentiment) in each sentence. To prove it, each sentence is splitted in clauses (a fragment with a predicative verb) and every clause contains only one opinion which we classified it in subjective or objective. Only the subjective clauses will be analyzed. For instance, the approach is based on minimal reductions [19], as the premise is that the neighboring clauses should have the same subjective classification. Then the sentences can be classified as either positive or negative.

3.3. Comparative sentiment analysis

In many cases, users do not offer a direct opinion about a product, preferring instead comparable opinions such as:

Dacia Logan arată ***mult mai bine*** *decât Dacia Solenza*[2].

In this case, the purpose of the sentiment analysis system is to identify opinions of the sentence containing the comparative views, as well as to extract there from the preferred entity. Authors like Jindal and Liu [12] describe this analytical method. Using a relatively small number of words as comparative adverbial adjectives *mai mult, mai puţin, uşoare*[3], superlative adjectives and adverbs *mai, cel puţin, cele mai bune*[4], additional clauses *favoare, mare, preferă, decât, superioară, inferior, numărul unu, împotriva*[5], we can cover 98 % of the comparative opinions.

For these words/groups of words which frequently appear in texts, but with low precision, a classifier[6] can be used to filter phrases that do not contain comparative views. Ding, Liu and Zhang [4] present a simple algorithm for identifying preferred entities relating to the type of comparisons used and the presence of negation.

3.4. Sentiment lexicon

As we have seen so far, the lexicon is the most important resource for the majority of the sentiment analysis techniques. There are three options in order to create a lexicon of sentiments:

a) *manual approaches*, when researchers create a manual lexicon, consisting of a set of words selected from explanatory dictionaries that will be subsequently extended by using existing lexical resources (synonyms and antonyms for enrichment). We have already mentioned WordNet. This process requires a laborious effort, especially that each domain needs its own lexicon. A handy algorithm is proposed by Kamps, J., Marx, M., Mokken, R.J. and de Rijke, M. (2004).

[2]En. - *Dacia Logan looks much better than Dacia Solenza.*

[3]En. - *more, less, easy.*

[4]En. - *more, at least, the best, etc.*

[5]En. - *favour, high, prefer, rather than, superior, inferior, the number one, against.*

[6]For example, Naive Bayes classifier, a statistical method for forms classification and recognition, where each document represents a collection of words and word order is considered irrelevant.

b) *corpus-based approaches*, in which a set of words/phrases extracted from a relatively small corpus is extended by using a large corpus of documents of a single domain.

The main disadvantage of any dictionary-based algorithm (a) is that the acquired lexicon is too general and therefore does not capture the specific features of a particular area. Advanced approaches based a lexicon are reported in Dragut et al. [5].

If we want to create a specific sentiment lexicon, we have to use a corpus-based algorithm. A classical work in this area [10] highlights the concept of sentiments consistency allowing the identification of complex polar adjectives. In other words, a set of linguistic connectors *şi, sau, nici, fie, sau*[7] has been used to find the adjectives that are connected to the adjectives with well-known polarity.

For example: *bărbat puternic şi armonios*[8].

If we admit that puternic is a positive word, we can assume that the word armonios is also positive thanks to the use of the connector *şi*.

4 The tool description

This version of our tool[9] is able to detect and to explain the appreciations about some entities (persons, products, brands, etc.). This tool is based on information like labeling of parts of speech (e.g. the XML example), extracting of interest nominal groups, automatic extracting of entities and anaphoric connections.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<DOCUMENT>
<P ID="1">
<S ID="1">
<W EXTRA="NotInDict" ID="11.1" LEMMA="" MSD="Vmip3s" Mood="indicative"
Number="singular" POS="VERB" Person="third" Tense="present" Type="predicative"
offset="0"></W>
<NP HEADID="11.2" ID="0" ref="0">
<W Case="direct" Gender="masculine" ID="11.2" LEMMA="nimic" MSD="Pz3msr"
Number="singular" POS="PRONOUN" Person="third" Type="negative"
offset="1">Nimic</W>
<W ID="11.3" LEMMA="mai" MSD="Rg" POS="ADVERB" offset="7">mai</W>
<W Case="direct" Definiteness="no" Gender="masculine" ID="11.4" LEMMA="odios"
MSD="Afpmsrn" Number="singular" POS="ADJECTIVE" offset="11">odios</W>
<W ID="11.5" LEMMA="," MSD="COMMA" POS="COMMA" offset="16">,</W>
<W ID="11.6" LEMMA="mai" MSD="Rg" POS="ADVERB" offset="18">mai</W>
<W ID="11.7" LEMMA="oribil" MSD="Rg" POS="ADVERB" offset="22">oribil</W>
<W Case="direct" Definiteness="no" EXTRA="NotInDict" Gender="masculine"
ID="11.8" LEMMA="decat" MSD="Afpmsrn" Number="singular" POS="ADJECTIVE"
offset="29">decât</W>
</NP>
<NP HEADID="11.9" ID="1" ref="1">
<W Case="direct" Definiteness="yes" Gender="masculine" ID="11.9" LEMMA="pantof"
MSD="Ncmpry" Number="plural" POS="NOUN" Type="common" offset="35">pantofii</W>
<NP HEADID="11.10" ID="2" ref="2">
<W Case="direct" Definiteness="no" Gender="masculine" ID="11.10" LEMMA="sport"
```

[7] En. - and, or, not, either.

[8] En - strong and harmonious man.

[9] The version previous of this tool, called EAT (Emotional Analysis Tool), is still in testing phase.

```
MSD="Ncmsrn" Number="singular" POS="NOUN" Type="common" offset="44">sport</W>
<W ID="11.11" LEMMA="cu" MSD="Sp" POS="ADPOSITION" offset="50">cu</W>
<NP HEADID="11.12" ID="3" ref="3">
<W Case="direct" Definiteness="yes" Gender="feminine" ID="11.12"
LEMMA="platform" MSD="Ncfsry" Number="singular" POS="NOUN" Type="common"
offset="53">platforma</W>
</NP>
</NP>
</NP>

</DOCUMENT>
```

Moreover it was developed an important ontology of entities, categories and values. In figure 1 we have the interface of our tool. We describe briefly work methodology:

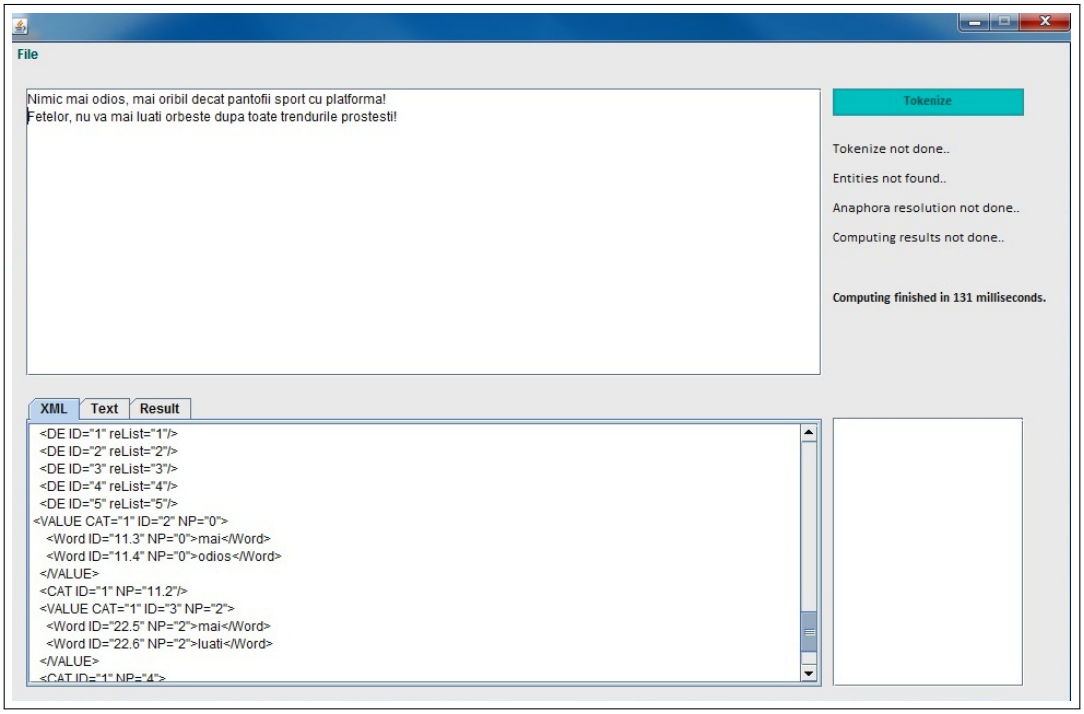


Figure 1: The interface of the computational tool

1. A corpus of texts (50 texts) is manually annotated using PALinka[10], in order to build triplets of the form: `<entitate><categorie><valoare>`.

2. The text is preprocessed using UAIC Romanian Part of Speech Tagger[11] [20]. This tagger combines a statistical model to one based on rules. The morphological dictionary was largely extracted from DexOnline and contains 1.25 milion distinct words. The result is an XML file, each word has been tockenized and annotated according to the POS that it represents.

3. Noun phrases are detected and annotated with NP-chunker[12] [20]. This chunker is used in

[10]http://clg.wlv.ac.uk/trac/palinka/

[11]POS tagger has a precision of 96,6%8, considered on the corrected version of the novel "1984" (George Orwell).(http://instrumente.infoiasi.ro/WebPosRo/).

[12]Chunker receives as input the tokenized text, in XML, formed by suitable groups in text, and the output is another XML file where each nominal interest group will be annotated XML with NP label (http://instrumente.infoiasi.ro/WebPosRo/).

many applications to resolve the ambiguities or to extract information. For example, the newest work studies based on machine translation use texts in two languages (parallel corpora) to derive the appropiate transfer models.

4. Proper names of entities are automatically extracted using a named entity recognizer techonology GATE[13] open source (ANNIE)[14].

5. Anaphoric links (especially, pronouns) are extracted from the text using RARE (*Robust Anaphora Resolution Engine* implemented by Eugen Ignat [11].). This process makes appreciations that the text expresses about those entities (coreferences) to be aggregated to the same entity (reference).

6. Entities, categories and values from the ontologies that have been already created are recognized in the text using NER (Named Entity Recognition) which extracted the entities automatically. NER recognizes entities such as persons, organizations or geographic locations, receiving as input a natural language text and the output is a text file which contains entities as a string that uses separators to delimit named entities.

7. A set of rules is written for the recognition of values and the connections such as `<entity><category><value>` are established.

8. Graphical inteface reveals the extracted information and global scores.

Of the recorded, our tool is able to detect and explain qualitative appreciations about entities. In figure 2 is profiled the architecture of this software as follows:
- *building an anthology* of entities, categories and values, useful to obtain a correct and complete result;
- *preprocessing text*, meaning annotation, splitting text into entities (words, symbols or tokens);
- *noun phrase chuncking* (NP-chunk), meaning splitting text into sequences of syntactically correlated words (nominal groups);
- *recovering anaphoric connections*, important not to lose any reference to a particular entity, using RARE.
- *extracting entities*, using NER module. It receives a file .txt (*input*). The output file contains only the entities mentioned in the analyzed text.

For instance: "[Vodafone] [România] oferă cea mai bună conectivitate pentru serviciile de date dintre toate reţelele mobile [GSM]/ [UMTS]/ [CDMA] din România".

The output file contains the following entities: Vodafone, România, Vodafone România, GSM, UMTS, CDMA. If an entity appears more than once, it will be found only once in the output file.

As an exemplification, here is a part of the XML output-file:

```
<entity type="company">Vodafone România</entity>
<category>conectivitate pentru serviciile de date</category>
<value ="1">bună</value>
```

- *recognizing categories, values and relationships with entities.* Considering the resulting files, once the previous phases have been completed, it will automatically extract the categories, values and relationships with entities using a set of rules (*regular expression*). These regular expressions use parentheses (round, square brackets) that form rules for constructing words. The most frequent use of regular expressions consists in recognizing if a string contains or not words or sub-string, that can be formed by that regular expression.

For instance: the string `p[oa]t` can be interpreted as *pot* and *pat*.

[13] http://gate.ac.uk/
[14] http://services.gate.ac.uk/annie/

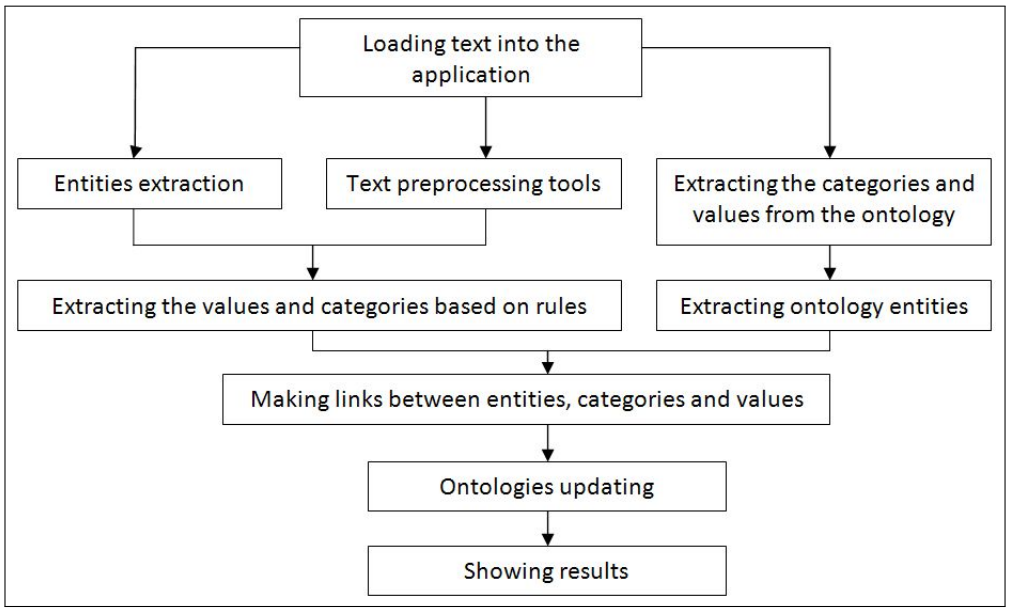


Figure 2: The architecture of the computational software

Basically, the tool completes the following steps:
- it identifies opinion words and phrases;
- it assigns to every positive or negative word a value (+1) for the positive one and (-1) for the negative one;
- the words which depend on context get also a value (0).

For instance: [Dacia] [Logan] este mai fiabilă decât orice [Opel].

```
<entity type="brand">Dacia Logan</entity>
<category>capacitatea sistemelor tehnice de a funcţiona </category>
<value ="1">fiabilă</value>
```

5 Conclusions and future work

This paper presents an automatic method able to detect and explain opinions on certain entities (peoples, companies, products, etc.) identified in a text, regardless of its nature (advertising, political, journalistic, etc.) based on a lexicon of opinions resulted from manual annotation (presented in other papers) of an initial corpus (consisting of opinion words and syntagmas). Moreover, in addition to this lexicon, we focused on the semantic role of negations and pragmatic connectors like "dar" ("*but*"). This application seeks to support the development of a complex lexical resource, necessary to interpret qualitative assessments found in any text. We are convinced that this analyze manner may be an important support for marketing managers, PR firms, politicians, online buyers, but, also, for specialists in NLP, linguistics, etc. Until now, we observed the fact that when a variable of neutralizing sentiments appears, it is not enough to cover only the summarizing operation of values for each opinion sentence. Because of that, we propose to add degrees of intensity and power in expressing opinions. In Romanian language, the superlative amplify semantically the convictions of the person who opines on an issue.

In the sentence - *Vodafone România oferă cea mai bună conectivitate pentru serviciile de date dintre toate reţelele mobile GSM/ UMTS/ CDMA din România.* - the word *bună* gets +1. The

superlative *cea mai* expands the scale of values. It can get the degree of positivity (or negativity). It depends on which word follows. So, *cea mai bună* gets (+2).

Also, due to pragmatic connectors, we have to give up on summarizing values.

Acknowledgments

In order to perform this research the first author received financial support from the Erasmus Mundus Action 2 EMERGE Project (2011 2576 / 001 001 - EMA2). I am also grateful to the NLP-Group@UAIC-FII for offering me support in using some tools for automatic interpretation of Romanian language.

Bibliography

[1] Ardeleanu, I. (2013); Extragerea de opinii din texte, lucrare de licenta coord. de prof.univ.dr. Dan Cristea, Universitatea Alexandru Ioan Cuza din Iasi.

[2] Dave, K.; Lawrence, S. and Pennock D.M. (2003); *Mining the peanut gallery: Opinion extraction and semantic classification of product reviews* in Proceedings of WWW.

[3] Das, S.; Chen, M. (2001); Yahoo! For Amazon: *Extracting market sentiment from stock message boards* in Proceedings of the Asia Acific Finance Association Annual Conference (APFA).

[4] Ding, X., Liu, B. and Zhang, L. (2009): Entity discovery and assignment for opinion mining applications. In Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

[5] Dragut, E.C., Yu, C., Sistla, P. and Meng, W. (2010): Construction of a sentimental word dictionary. In Proceedings of ACM International Conference on Information and Knowledge Management.

[6] Esuli, A.; Sebastiani, F. (2006); *Determining term subjectivity and term orientation for opinion mining* in Proceedings of EACL-06, 11th Conference of the European Chapter of the Association for Computational Linguistics, Trento, IT. Forthcoming.

[7] Gifu, D. (2012); Political Text Categorization in *Humanities and Social Sciences Review*, Vol. 1, No. 3, University Publications.net, USA, part of the paper presented in The International Journal of Arts and Sciences' (IJAS) International Conference for Academic Disciplines, Harvard University, Cambridge, Massachusetts, 27-31 May 2012.

[8] Gifu, D. (2013); *Temeliile Turnului Babel. O perspectiva integratoare asupra discursului politic*, Ed. Academiei Romane, Bucuresti.

[9] Habermas, J. (1962); Strukturwandel der Ofentlichkeit: Untersuchungen zu einer Kategorie der burgerlichen Gesellschaft. Neuwied, Luchterhand. [Trad. rom.: *Sfera publica si transformarea ei structurala*, Bucuresti, CEU, 1989.]

[10] Hatzivassiloglou, V. and McKeown K. R. (1997): Predicting the semantic orientation of adjectives. Proceedings of ACL-97, 35th Annual Meeting of the Association for Computational Linguistics, Madrid, ES, Association for Computational Linguistics.

[11] Ignat, E. (2011); RARE-UAIC (*Robust Anaphora Resolution Engine*), resursa gratuita pe META-SHARE, Universitatea "Alexandru Ioan Cuza" din Iasi, 2011.

[12] Jindal, N. and Liu, B. (2006): Identifying comparative sentences in text documents. In Proceedings of ACM SIGIR Conf. on Research and Development in Information Retrieval.

[13] Kamps, J., Maarten, M., R. ort.Mokken and Maarten de Rijke. (2004): Using WordNet to measure semantic orientation of adjectives in Proceedings of LREC-04, 4th International Conference of Language Resources and Evaluation, vol. IV.

[14] Liu, B. (2010); *Sentiment analysis and subjectivity.* Handbook of Natural Language Processing. N. Indurkhya and F.J. Damerau, eds.

[15] Liu, B. (2012); *Sentiment analysis and opinion mining.* Synthesis Lectures on Human Language Technologies, Morgan Claypool Publishers.

[16] Mihalcea, R.; Banea C.; Wiebe, J. (2007); *Learning Multilingual Subjective Language via Cross-Lingual Projections* in 45th Annual Meeting of the Association for Computational Linguistics (ACL-2007).

[17] Pang, B.; Lee, L. (2008); Opinion mining and sentiment analysis in *Foundations and Trends in Information Retrieval*, 2.

[18] Pang, B.; Lee, L.; Vaithyanathan, S. (2002); *Thumbs up? Sentiment Classification using machine learning techniques* in Proceedings of EMNLP-02, 7th Conference on Empirical Methods in Natural Language Processing (Philadelphia, PA). Association for Computational Linguistics, Morristown, NJ.

[19] Pang, B.; Lee, L. (2004); *A Sentimental Education: Sentiment Analysis using Subjectivity Summarization based on minimum cuts* in Proceedings of the Association for Computational Linguistics.

[20] Simionescu, R. (2011); *POS-tagger hibrid*, lucrare de disertatie coord. de prof.univ.dr. Dan Cristea, Universitatea "Alexandru Ioan Cuza" din Iasi.

[21] Turney, P. (2002); *Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification* of reviews in Proceedings of the Association for Computational Linguistics.

[22] Tong, R.M. (2001); *An operational system for detecting and tracking opinions in on-line discussion* in Workshop note, SIGIR 2001 Workshop on Operational Text Classification.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):703-710, December, 2014.

Colony of Robots for Exploration Based on Multi-Agent System

I. Hughes, G. Millán, C. Cubillos, G. Lefranc

Ian Hughes, Claudio Cubillos, Gastón Lefranc*
Pontificia Universidad Católica de Valparaíso
Escuela de Ingeniería Eléctrica
Avda. Brasil #2147
Valparaíso - Chile
*Corresponding author: glefranc@ucv.cl

Ginno Millán
Universidad Católica del Norte
Escuela de Ingeniería
Larrondo #1281
Coquimbo - Chile
gmillan@ucn.cl

Abstract: In this paper a colony of robots for closed environments exploration is presented. This small colony of robots, conformed by mobile robots and a quadcopter, is based on heterogeneous Multi-Agent System (MAS). The objective of the system is to quickly recognize a closed three-dimensional environment, without access to references such as a Global Positioning System (GPS), to perform exploration of each unit with different characteristics and perform a joint recognition. All communications work wirelessly with a system responsible of data collection, tracking and managing all collected information. Finally, it provides a basis for multi-agent robots which allow recognition, mapping and information gathering in places where units are efficiently deployed the entire colony's abilities.
Keywords: Colony of robots, Multi-Agent Systems (MAS), robotics.

1 Introduction

The distribution of tasks among different units incorporated into engineering areas is an alternative to reduce costs in contrast with one large unit to which all tasks are assigned. Distributing several tasks significantly improve performance, increase fault tolerance, among other qualities, especially when these tasks are inseparable by a single unit. In a colony of robots using multi-agent robotics, a task's distribution can have several different solutions to a variety of problems based on behaviour control by cooperation and collaboration [1], [2].

A typical task requires distribution to increase performance in spatial exploration, which has problems such as maximizing every unit and proper path planning to avoid intersections. Added problems are presented with heterogeneous units, moving capabilities, sensor type and range. Processing power increase the challenge of control due different conditions that have to be applied to each unit.

Tasks and scenarios which are assigned to the MAS will first undergo some standardization of structures and specific protocols cite3. Some of these tasks in the colony of robots are: communication, architecture, planning and control, location, map display and exploration strategies, interaction with the environment, multi-robot for air and ground coordination adaptive exploration and generation of maps [4].

The communication task in MAS has been built maintaining standardized protocols such as FIPA (Foundation for Intelligent Physical Agents) [4] that provide complete and sufficient ontological structure for virtually any application in multi-agents for both the software and mobile

robotics. Communication with one specific robot or the group, sharing particular portions of the gathered information or simply the update rate, improves the unit or group performance [5]- [7].

Also, a control procedure is needed to verify that the global mission objectives are met and that the electronic and software architecture is capable of supporting these control mechanisms. Thus, it is possible to relegate tasks and share goals, using heterogeneous agents efficiently, capable of producing a consistent behaviour [8].

For map representation and exploration, a number of approaches and methodologies for general and specific situations are implemented. An exploration, mapping and localization are always relative to the robots and subject to inaccuracies which can be partially corrected through redundancy or loop closure algorithms [4], [7], [10].

Coordination of multiple agents has to consider each unit independently but also collectively. It has been suggested in several studies that the joint work of robots that require a high level of coordination and cooperative action, in which they must carry loads or move objects exceed the capacity of action of a single unit [4], [10].

Coordination for multi-robot exploration and mapping generation with MAS can create maps through collaborative exploration, with a technique similar to SLAM (Simultaneous Localization And Mapping) with multiple reference points added by different units and perspectives [6], [10], [12].

In this paper, the implementation of a small experimental heterogeneous colony of robots is presented. This heterogeneous colony consists of mobile robots and a quadcopter, based on MAS capable of performing exploration in closed environments, to obtain 3D maps. The system executes scanning in closed environments, without access to external global references such as GPS. The recognition, mapping and information gathering takes place in the colony.

2 The Colony of Robots

A small colony of robots conformed by mobile robots and a quadcopter and based on a heterogeneous robotics MAS is implemented to perform exploration in closed environments, and to obtain maps. This system works by collaboratively scanning closed 3D environments to browse, navigate and communicate information using only the sensors available on each unit. To support the entire software structure, ROS (Robot Operating System) is implemented for its easy reconfiguration [11].

Figure 1 shows an overview of simple configuration of the MAS working on a bi-dimensional plane. In this case, robotic units use their sensors to identify obstacles and communicate with a central station. This communication takes place via TCP/IP messaging structure and wireless links, to share information and generate maps allowing browse previously visited places. All this is on top of a software architecture based on ROS that providing a 3D simulation platform [3].

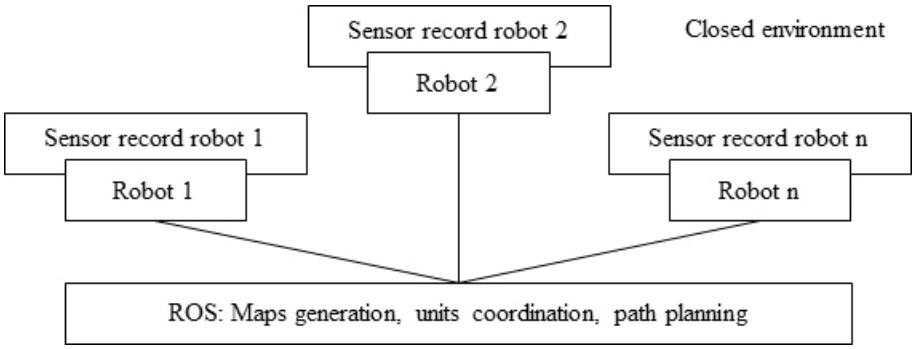


Figure 1: ROS communication with robotic units within a closed environment

This platform is capable of representing different types of sensors, robotic units and conditions before the implementation of the system results in real conditions. The simulation layers of ROS accept data from real sensors to control an output link.

The MAS easily integrates with ROS since the internal communication methods are similar to those set by FIPA. This allows complete integration for MAS to interact with ROS for specific tasks, facilitating exploration and an efficient use of the units to explore various places simultaneously while sharing information. The ROS interface show the state of the agent sensors to measure distances to tha walls, translation and rotation in all axes.

Two real units, the terrestrial mobile robots and air quadcopter (Figure 2), are used to provide real-time information displayed on ROS with respect to all sensors, inertial measurement units and distance sensors. Wireless communication is possible at approximate distance among units of 50 m and 150 m in closed environments such as the inside of a conventional house. Navigation data allow mapping environment.

Figure 2: Robotics unit used

3 Multi-Agent System (MAS)

A multi-agent system is characterized by software with a robust architecture, adaptable and operates in different environments. The different objectives can be achieved in an intelligent and autonomous fashion, by exchanging information from the environment or with other agents [9].

A "Blackboard" type structure used as a workspace to which all agents that request access in order to share information in a bidirectional manner, considers various agent writers and readers. Each agent can see the data status, update their own, and write the task information results to make them accessible to everyone else. To have a better solution that facilitates cooperation, collaboration and communication among agents, high level moderators in charge of supervision are introduced. They monitor and evaluate the situation using the knowledge base to select the most qualified agents to solve the sub problem. A "dispatcher" is also included as responsible for informing of any new situation available on blackboards (Figure 3). Messages between among agents need to comply with a collaborative communication protocol. In this case, it complies with FIPA protocol. This protocol must specify the type of communication process and message format together with processing the semantics of agent communication language. In this case, it complies with FIPA protocol.

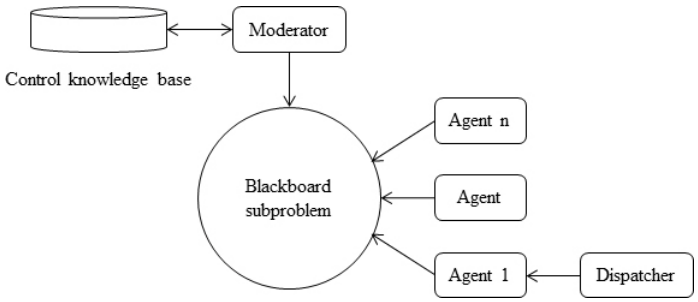


Figure 3: Model of blackboard information exchange system with modifications for optimization

4 Multi-Agent Structure

To carry out the proposed mission of exploration, different types of agents working collaboratively are required to carry out the task of navigating a complex environment while maintaining the integrity of the robots. The MAS structure is shown in Figure 4.

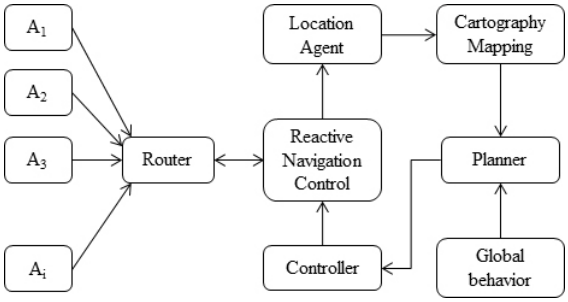


Figure 4: Control based on multi-agent for heterogeneous robot navigation

Reactive Agents: These agents are known for being the simplest and rely solely on the environment variables to give an immediate response. These reactive agents used are: Agents of perception: agents obtain environmental information from robot internal parameters (battery charge level, system status, etc.); Router Agent: it keeps track of the agents and makes the link between the control electronics and a radio transmitter to software agents; and Reactive Navigation Control Agent: this agent makes decisions to meet the requirements and requests of the highest level agents.

Deliberative Agents: These agents are in charge of generating the maps, making control decisions and produce high-level strategies to meet the requirements of exploration, planning and control. These deliberative agents are: Planner Agent: this agent makes decisions to meet the requirements and requests of the highest level agents; Mapping Agent: it provides a system for capturing and storing navigation data collected by the sensors of the robots; Driver Agent: It flows of information from the planner agent to reactive control agent, transforming them into readable data in terms of direct control of robots; Agent interface for overall performance: This agent simply provides a query method for multi-agent system to determine the type of map to build; and Agent Location: Since the generation of maps and the location of several agents are needed localization algorithms.

5 Evaluations

A 2D Simulation with individual units and a test with real data, directly from the sensors of each agent, has been done. The mobile robots and the air quadcopter, provide real-time information displayed on the display of ROS with respect to the sensors and operational inertial measurement unit. In order to test a multi-agent configuration with heterogeneous units, the problem is minimized to a closed environment exploration with only a restricted amount of obstacles for the units to map. Communication between the two units (extensible to more units) establishes a framework that can be used for various types of tasks and processes for collaborative loop closure. Wireless communication allows an approximate distance between units of 150 meters and 50 m in closed environments, such as the inside of a house. With well-established communication, MAS travelled in a controlled fashion through the inside of a simple room. The navigation data is successfully recorded from all the incoming information and compiled into one single 3D map. For the robotic test platforms used in this research, communication is able to address every robotic unit individually, by group or by general broadcast. On top of this, the protocol using MAS strategies needs to take into account the various characteristic of each heterogeneous unit to optimize data flow.

For evaluation purposes, objects representing all possible obstacles (under the sensors minimal specifications for detectability) or errors in sensor readings are removed from the areas to be explored leaving only larger walls. Both, aerial and terrestrial units, have exactly the same capacity in sensors to generate maps and internal representations of their surroundings. After several runs all movement and rotation decisions are made according to the distance measured with three sensors, away from the nearest wall.

The ranges measured empirically with ultrasound can be displayed in a Table 1 and refer to minimum and maximum resolutions that robotic units are able to register.

Table 1: Distance and size respectively maximum and minimum detectability with the maximum opening degree sensor

Maximum distance to the obstacle (meters)	**Minimum size obstacle (meters)**	**Sensor opening (degrees)**
1.8	0.006	< 36
3	0.03	36
3.6	0.083	40
> 6	> 0.1	50

The sensors arrangement, from the front and two opposite sides, has a total of 234 degrees of coverage to the front with two blind spots of 34 degrees centered at 45 degrees from the

central axis of each robotic unit. In the case of aerial units has maximum range sonar 6 meters vertically pointing towards the ground, to keep a record high. Figure 5 shows the reading of the three sensors captured directly from a real drive and after moving interpretation.

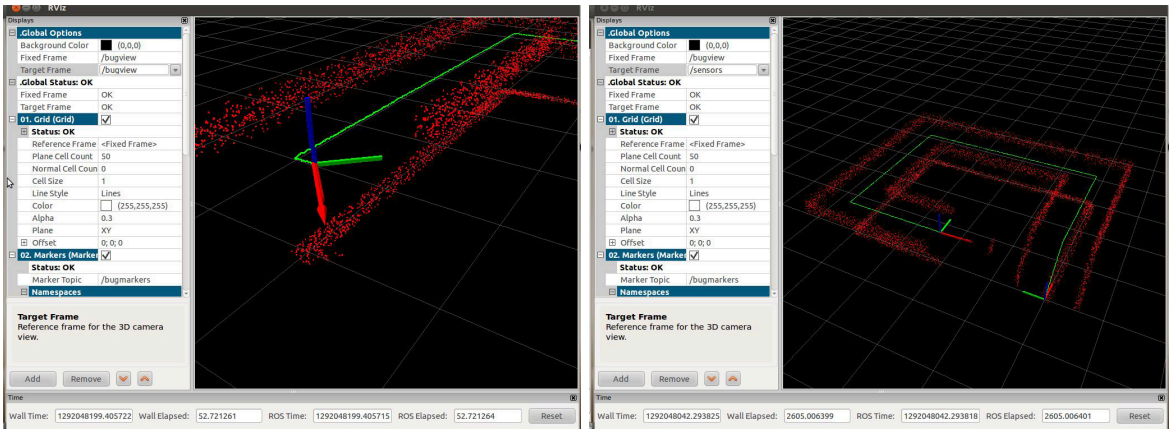


Figure 5: Data received from one of the perception agent in ROS

By removing small objects below the maximum resolution of the sensors, the environment becomes easily navigable. Obviously, the replacement of sensors by other higher resolution can also be considered as an alternative.

For the same scenario, in 10 opportunities are similar times to the end of the path indicated by the arrest of the units having made the assumption of having entire area. Since for this case are two units that perform work at different levels in a three-dimensional plane, it is assumed that a single unit would have required twice the time to explore two levels. Table 2 yields the following:

Table 2: Travel times for attempted per unit and the sum of both

Try	Terrestrial time (min)	Aereal time (min)	Addition (min)
1	1:15	0:45	2:00
2	1:20	0:46	2:06
3	1:16	0:47	2:04
4	1:22	0:46	2:08
5	1:14	0:45	1:59
6	1:19	0:48	2:07
7	1:16	0:44	2:00
8	1:15	0:45	2:00
9	1:19	0:45	2:05
10	1:16	0:46	2:02

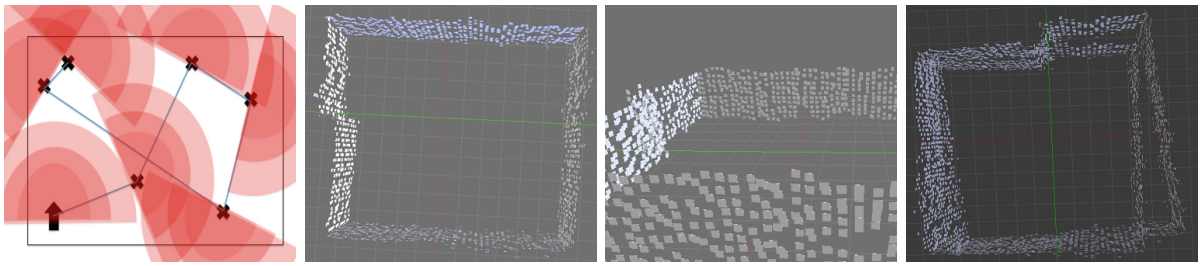
This test provides a comparison of the times it would take a single unit versus the two used in operating scheme and heterogeneous multi-agent to travel and create a map of an environment. By having multiple units, not only reduce the times, but also produces greater redundancy in the data recorded by the sensors which could lead to fewer errors, reduced costs in a single unit of sensory precision greater speed in capturing information. This is being administered by MAS, allowing the inclusion of algorithms that make searches more efficient and avoiding routes for example, visiting a point repeatedly.

During each run, the necessary data is captured to reconstruct a 3D environment and generate

a map. In the times indicated in the above table was obtained real-time reconstruction from the data collected from the sensors, which when viewed by RVIZ of ROS, as in the simulations, can form a complete environment. In the following sequence of figures shows the map after completion of a course of 1:16 minutes of a land agent. For better representation and using the constant redundant sensor readings, each marked on the map represents average readings closest. With this you get a flatter surface and representative mapped with a maximum error of about 15 centimeters per point generated on the map.

The arrow, in Figure 6(a), represents starting point, the crosses are points of decision-making, where it alters the path avoiding obstacles. The three media circles around the starting arrow represent what a unit would be able to see at 1 meter for the inner circle, 1.5 for the next, and 2 meters for the outer circle. The sensors have a maximum range of 6 meters.

During each run, the necessary data is captured to reconstruct a three-dimensional environment and generate a map (Figure 6(d)). Times obtained in real-time reconstruction from the data collected from the sensors, takes 1:20 min to mobile robot (Figure 6(b)) and 0:46 min to quadcopter (Figure 6(c)) to get maps. In total is 2:06 min to have 3D map (Figure 5(d)), with small error.



(a) (b) (c) (d)

Figure 6: (a) Robots using its ultrasonic sensors, (b) Map obtained by mobile robots, (c) Map obtained by quadcopter, (d) Overlay maps generated by two units

6 Conclusions

This paper the implementation of a small Colony of Robots, conformed by mobile robots and quadricopter and based on heterogeneous MAS has been presented. This system is capable of performing exploration, scanning of closed three-dimensional environments, without GPS and only using collaborative techniques for global and local orientation. The system has wireless communication for data collection, track and manage all information collected. A multi-agent robot permits the recognition, mapping and information where units are deployed.

The heterogeneous MAS is capable of distributing among agents and respond appropriately, with ROS, to robot applications. MAS can control several units with different capacities and characteristics. Tests with a mobile robot and aerial unit show MAS is able to manage navigation and communicate both data and actions and commands in a scalable way.

The maps obtained match the simulated, demonstrates that communication, data transfer and displacement by mechanical agents, in real working environments, perform in accurate way with heterogeneity and cooperation among different units.

Bibliography

[1] Lefranc, G. (2008); Colony of Robots: New Challenge, *International Journal of Computers Communications & Control*, ISSN 1841-9836, 3(S):92-107.

[2] Lefranc, G. (2008); Colony of robots, in Lotfi A. Zadeh, Dan Tufis, Florin Gheorghe Filip, Ioan Dzitac (eds.), *From Natural Language to Soft Computing: New Paradigms in Artificial Intelligence*, Editing House of Romanian Academy, ISBN:978-973-27-1678-6.

[3] FIPA TC Architecture, FIPA Abstract Architecture; Specification, Foundation for Intelligent Physical Agents, (http://www.fipa.org), 2002.

[4] Rojas, D. et al (2013); Integration of Algorithms for Maps Construction and Simultaneous localization in a Mobile Robot. *IFAC International Conference on Management and Control of Production and Logistics*, Brazil, 129-134, DOI: 10.3182/20130911-3-BR-3021.00098.

[5] Roth, M.; Simmons, R.; Veloso, M. (2004); *Decentralized Communication Strategies for Coordinated Multi-Agent Policies*, Robotics Institute, Carnegie Mellon University.

[6] Latorre, H.; Harispe, K.; Salinas, R.; Lefranc, G. (2011); Ontology Model of a Robotics Agent Community, *International Journal of Computers Communications & Control*, ISSN 1841-9836, 6(1):125-133.

[7] Latorre, H.; Harispe, K.; Salinas, R.; Lefranc G. (2010); Proposed Model of Behavior for a Community of Robotic Agents, Congreso Infonor, Chile, 168-171, DOI: 10.1109/SCCC.2010.8.

[8] Pechoucek, M.; Marík, V. (2008); Industrial deployment of multi-agent technologies: review and selected case studies, *Auton Agent Multi-Agent Syst*, ISSN: 1387-2532, 17(3):397-431.

[9] Bermes, C. et al (2008); New Design of the Steering Mechanism for a Mini Coaxial Helicopter, IEEE Int. Conf. Intelligent Robots and Systems, Nice, France, 1236-1241, DOI: 10.1109/IROS.2008.4650769.

[10] Fredes, D.; Cubillos, C.; Lefranc, G. (2012); Mobile Robot with Multi Agent Architecture, *IEEE Conference International on Engineering and Systems Applications*. Chile.

[11] Quigley, M. et al (2009); ROS: an open-source Robot Operating System, In ICRA Workshop on Open Source Software.

[12] Cheein, F. A. A.; Tobeiro, J. M., di Sciascio, F.; Carelli, R.; Lobo Pereira, F. (2010); Monte Carlo Uncertainty Maps-based for Mobile Robot Autonomous SLAM Navigation, *IEEE International Conference on Industrial technology*, Viña del Mar, Chile, 1433-1438, DOI: 10.1109/ICIT.2010.5472495.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):711-720, December, 2014.

An Algorithm for Production Planning Based on Supply Chain KPIs

D. Makajić-Nikolić, S. Babarogić, D. Lečić-Cvetković, N. Atanasov

Dragana Makajić-Nikolić*, Sladjan Babarogić
Danica Lečić-Cvetković,Nikola Atanasov
University of Belgrade, Faculty of Organizational Sciences
Serbia, 11000, Belgrade, Jove Ilića 154
E-mail: gis@fon.bg.ac.rs, sladjan@fon.bg.ac.rs
danica@fon.bg.ac.rs, atanasovn@fon.bg.ac.rs
*Corresponding author

Abstract: This paper observe multi-period multi-product production planning problem in make-to-stock production environment with limited production capacity. Such problem is identified in Fast Moving Consumer Goods industry. The goal was to develop an algorithm for supporting dynamic production triggering decisions in relation with two supply chain key performance indicators: stock cover and customer service level. The presented approach is applied to a real example in several scenarios based on different decision criteria.
Keywords: production planning, stock cover, customer service level, heuristic algorithm.

1 Introduction

Manufacturing companies that operate in markets with changing demand are often faced with the problem of insufficient supplies of finished products. Constant fluctuations in demand and the required financial investments are influencing the decision concerning the expansion of the production capacity. Until the production capacity is actually increased, the manufacturing company has to meet the growing demand of the market with its existing production capacity.

As customer orders are received periodically and production capacities are often insufficient, it is necessary to make a choice of products which will be produced in each period. In circumstances of reduced uncertainty, it is possible to use exact methods for planning customer satisfaction by cycle, when trends in demand are predictable over a longer period of time. However, in real life this is not the case, as demand is a weekly phenomenon which requires dynamic decision-making. Therefore, in this paper we propose an algorithm for production planning in which decision on triggering new production is based on two supply chain key performance indicators (KPI): customer service level and stock cover.

This paper is organized in six sections. Next section describes related work with conceptual foundations. Problem description with relevant notation is presented in section tree. Heuristic algorithm for inventory planning is given in fourth section. In fifth section, numerical results with case study are described. Last section is dedicated to conclusions of the research.

2 Related Work

According to [5] uncertainty in production companies is categorized into environmental uncertainty - based on demand and supply uncertainty and system uncertainty (within the production process) - mainly related to production lead time, quality or failure of production process. Uncertainty depends on level of information required to perform relevant business activity based on efficient and effective management decision [4]. Responsive production planning and control

system according to [7] is the most important factor in achieving good delivery performance and demand satisfaction in supply chain. This fact represents one of important reason to focus more on customer service as external performance, once when internal performance is already achieved on certain level.

As recognized by Shen and Daskin in [13] major cost factors associated with designing and managing a supply chain are the facility location costs, the inventory management costs, and the distribution costs, and always should be considered jointly and integrated with customer service goals. Customer service was recognized as key measure of performances within production companies according to [9] and very well described by [11]. Overall managerial question in supply chain is to determine a cost-effective customer-service level in correlation with profits and associated costs, what lead to question: Which service level will satisfy customers and what level of inventory is required? Jeffery et al. identified a range of models for determining service level and the appropriate level of inventory, process was carried out based on logistic regression to understand how performance of delivery are dependent on three independent variables: order lead time, errors in forecast, and variation in demand [6]. Further development of service level and customers selection in make-to-stock production environment was evaluated by [8], while authors in [1] evaluated possibilities for maximization of customer service with limited production capacity and customer classification.

Stock cover is key performance indicator measuring length of time that available finished goods will last if forecasted consumption happens. Available finished goods ready to be delivered to customer according to identified demand are in direct correlation with customer service level. Dellaert and Jeunet [2] evaluated stock cover in relation to behavior of lot-sizing rules in a multilevel context, when forecast demand is subject to changes within the forecast window and relevant lead time. According to [12], supply system needs to ensure adequate stock level to satisfy customers need, despite that additional stock only generates unnecessary costs, which customer has to absorb at the end. Managing customer service level and stock cover represents highly complex problem of supply chain, taking into account that these two KPIs are leading to opposite directions - high stocks assume high customer service level, and, at the same time, stock need to be minimized to deliver working capital reduction and overall company efficiency. Working capital reduction coupled with increase of sales and certain service level was evaluated by [14] through results of horizontal collaboration between supply chain members. Combined approach of managing in parallel customer service level and stocks cover was done by [3] who evaluated main obstacles in increasing pressure to reduce working capital, growing variety of products and the fulfillment of a demanding service level. Petri nets model of production planning system based on supply chain KPIs: customer service level and stock cover was presented in [10].

3 Problem Description

In this paper we observe multi-period multi-product inventory planning problem in make-to-stock production environment with limited production capacity. We started from a real example of Fast Moving Consumer Goods (FMCG) in Serbia. Choice of products than will be produced should be made in each period (cycle). This decision is based on two key performance indicators: Customer service level (CSL) and Stock cover (SC). The basic assumptions of the observed problem can be divided into three groups as follows.

- *Customers orders assumptions*:
  - Customers place orders in all or almost all of the cycles;
  - Demand for each product is uneven and is known only for one cycle in advance;

  - Demand for each product represents the sum of all customers orders for delivery in given cycle;
  - Decision about fulfillment is done in given cycle when all orders are received;
  - Demand is fulfilled from the stock, entirely or partially, depending on the inventory level;
  - Orders that have not been fully met in the reporting cycle shall not be compensated in the subsequent cycles - no reordering policy;

- *Inventory assumptions*:
  - If the incoming customer orders in a single cycle do not exceed the available stock of finished goods, the allocation is complete and all customer orders are fulfilled, while any surplus products are stored for the next cycle;
  - Inventory holding costs are neglected;
  - Total inventory capacity is not limited. Therefore, inventory planning problem becomes production planning problem;

- *Production assumptions*:
  - The production capacity is limited and constant in the entire period;
  - There is no possibility for production extension in medium term planing horizon;
  - Due to specific production technology requirements, outscoring with acceptable costs is not possible;;
  - Lot sizes of products are different and fixed;
  - Production time and costs are neglected due to homogeneity of the products;

In order to formulate the algorithm and based on the problem assumptions, the following notation will be used in the remaining of the paper:

- n - number of products;
- m - number of periods in the observed time horizon;
- $mcsl$ - minimally acceptable customer service level;
- msc - minimally acceptable stock cover;
- l_i - lot size of i-th product, $i = 1, \ldots, n$;
- C - available production capacities in each period;
- S_i - inventory level of i-th product at the beginning of the observed time horizon, i=1,. . . ,n;
- t_{ij} - demand for i-th product in j-th period, i=1,. . . ,n, j=1,. . . ,m;
- p_{ij} - forecast for i-th product in j-th period, i=1,. . . ,n, j=1,. . . ,m;

Forecasts are calculated using k-periods moving average:

$$p_{ij} = \frac{1}{k} \sum_{u=1}^{k} t_{ij-u} \tag{1}$$

4 Algorithm

The goal of the algorithm (Fig.1) is to support two-phased decision making process. The result of the first phase is a list of products that should be produced in observed period, where decision is made based on calculated KPIs (CSL and SC). In the second phase, the algorithm forms a list of products that will be produced, applying one of four defined criteria: minimal CSL, minimal SC, maximal capacity utilization and maximal number of products and taking into account the available capacities. The proposed algorithm has polynomial complexity.

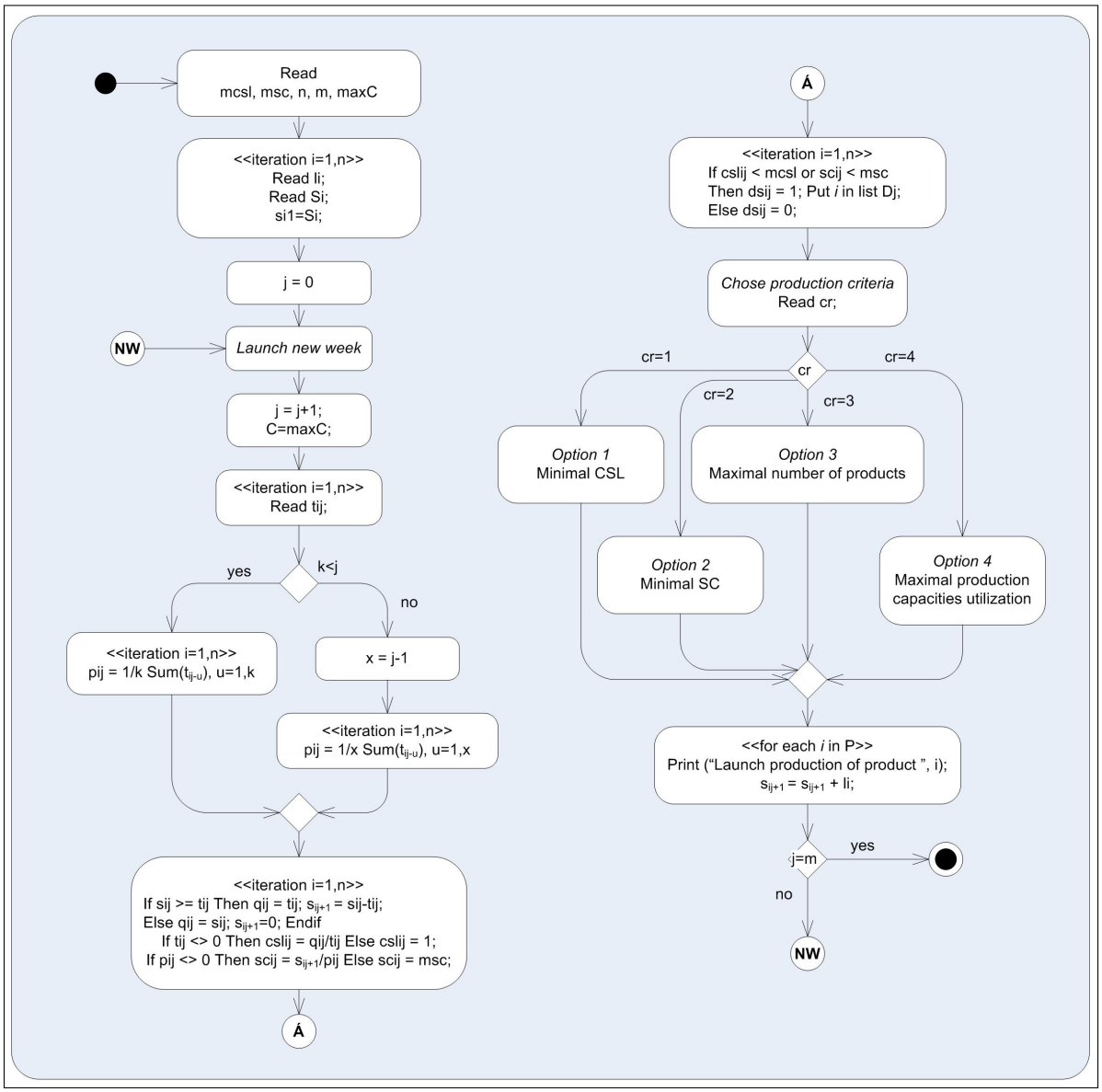


Figure 1: Algorithm represented as UML 2.0 Activity Diagram

Using parameters defined in problem description section, the following variables are calculated in each period.

- s_{ij} is a stock level of i-th product in j-th period, i=1,...,n, j=1,...,m.
- q_{ij} - delivered quantity of i-th product in j-th period, i=1,...,n, j=1,...,m. The delivered

quantity of each product depends on demand and stock level as follows:

$$q_{ij} = \begin{cases} t_{ij}, & s_{ij} \geq t_{ij} \\ s_{ij}, & \text{otherwise} \end{cases}, i = 1, \ldots, n, j = 1, \ldots, m$$

- $csl_{ij} = q_{ij}/t_{ij}$ - customer service level of i-th product achieved in j-th period, i=1,...,n, j=1,...,m, observed as fill rate, indicates ratio between delivered and ordered quantity;
- $sc_{ij} = s_{ij}/p_{ij}$ - stock cover for i-th product provided in j-th period, i=1,...,n, j=1,...,m.

Based on variables values, two decisions must be made sequentially for each product in each period.

1. The first decision refers to the request for the production of the i-th product in j-th period ($ds_{ij}, i = 1, \ldots, n, j = 1, \ldots, m$). Request for production is initiated if any of the indicators (csl_{ij} or sc_{ij}) falls below the minimum value, i.e:

$$ds_{ij} = \begin{cases} 1 \text{ (production requested)} & , \quad csl_{ij} < mcsl \text{ or } sc_{ij} < msc \\ 0 \text{ (production not requested)}, & \text{otherwise} \end{cases}$$

   for $i = 1, \ldots, n, j = 1, \ldots, m$.

   As a result, a set of all products that should be produced during j-th period is obtained: $D_j = \{i | ds_{ij} = 1\}$.

2. Since the available production capacities are limited without possibilities of extension in short term and often insufficient, the second decision is related to a choice of products that will be produced. This choice can be made using several criteria: minimal CSL, maximal number of product and maximal production capacities utilization. Let P be a set of chosen product. In the following, each of the above criteria for generating the set P will be described in detail.

Options 1 and 2 - minimal CSL and minimal SC

According to the first two criteria, the higher priority is given to the products with smaller demand satisfaction or with smaller stock cover in the previous period. These criteria should be used in the case of large variations in the customer service level or stock cover among products. For each time period j, the following procedure is applied.

Initialization: $P = \emptyset$.
Do
  Find $i*$ such that $csl_{i*j} = min\{kpi_{ij} | i \in D\}$.
  If $l_{i*} \leq C$ then $i* \rightarrow P$, $C := C - l_{i*}$ endif.
  $D_j := Dj\ i*$
until $C = 0$ or $D_j = \emptyset$.

Variable kpi_{ij} represents key performance indicator csl_{ij} (Option 1) or sc_{ij} (Option 2) depending on chosen criteria.

The output of the procedure is the set P which contains the indexes of the products that will be produced.

Options 3 and 4 - maximal number of products and maximal production capacities utilization

Maximal number of products can be used as criterion when company wants to cover the market with wide variety of products. When there is a large lack of capacity, maximal production capacities utilization should be used as a criterion. For each period j, these two criteria can be modeled as following knapsack problem:

$$max \sum_{u \in D_j} a_i \cdot x_i$$
s.t.
$$\sum_{u \in D_j} l_i \cdot x_i \leq C$$

where:

$$x_i = \begin{cases} 1, & \text{if the product} i \text{ is chosen to be produced} \\ 0, & \text{otherwise} \end{cases}, i = 1, \ldots, n$$

$$a_i = \begin{cases} 1, & \text{if the criterion is number of products } (\textit{Option 3}) \\ l_i, & \text{if the criterion is capacities utilisation } (\textit{Option 4}) \end{cases}, i = 1, \ldots, n$$

After obtaining the optimal results, set of products that will be produced, P, contains all products i such that $x_i = 1$.

5 Computational Results and Discussion

The algorithm has been applied on real data calculation based on 28 weeks observation, made for four products of real, medium sized Fast Moving Consumer Goods (FMCG) company. Installed production capacity is 290 units per period (week), while lot sizes are 120, 110, 170 and 50 units for products p1, p2, p3 and p4, respectively. Customers' orders are shown in Table 1. Forecast is calculated based on three-week moving average (equation 1).

Table 1: Customers' orders for 28 weeks

Product	w1	w2	w3	w4	w5	w6	w7	w8	w9	w10
p1	35	75	29	48	40	52	29	59	67	82
p2	50	122	55	129	40	346	70	102	112	16
p3	68	112	48	94	62	357	75	98	124	18
p4	6	23	27	57	0	30	45	38	56	69
Product	w11	w12	w13	w14	w15	w16	w17	w18	w19	w20
p1	96	88	45	23	16	31	166	16	34	137
p2	83	28	40	28	53	100	70	36	49	292
p3	11	25	49	34	63	117	112	48	69	325
p4	14	67	171	27	4	66	5	8	15	110
Product	w21	w22	w23	w24	w25	w26	w27	w28		
p1	83	23	110	156	109	102	34	69		
p2	95	119	447	56	154	34	112	107		
p3	162	128	314	93	156	88	115	128		
p4	3	86	153	1	1	1	0	25		

Based on descriptive statistical analysis for the observed period, it can be concluded that products p2 and p3 had the highest average demand (105.18 and 110.46, respectively) but also the highest standard deviation of demand (99.81 and 87.69, respectively). In addition, these two products had three major demand peaks in the same periods (w6, w20 and w23). These facts lead us to the conclusion that production capacity will remain mean issue in the future.

Developed algorithm can be used at operational and strategic level. Operational level is related to the decision of production requesting and launching. Table 2 illustrates the operational decisions. It shows the decisions for the product p1 in the entire period when minCSL criterion is used. The first column represents the weeks of the observed period. The second and third columns give the orders and the forecasts per weeks for product p1. Columns labeled as SB and SA shows the stock level before and after product delivery while the next column gives the amount of delivered quantities. Columns CSL and SC show calculated values for customer service level and stock cover per periods. The last three columns represent the decisions about requested, confirmed and missed production, respectively. In the last row of the Table 2, the values of total delivered quantities, average SCL and SC, and total number of production request, conformation and missing are given.

Table 2: Production decisions for 28 weeks for p1 and minCSL criterion

Time	Order	Forecast	SB	SA	delivered	CSL	SC	PR	PC	PM
w1	35	36	110	75	35	1	2.083			
w2	75	66	75	0	75	1	0	1	1	
w3	29	95	120	91	29	1	0.958	1	1	
w4	48	46	211	163	48	1	3.518			
w5	40	51	163	123	40	1	2.428			
w6	52	39	123	71	52	1	1.821	1		1
w7	29	47	71	42	29	1	0.9	1	1	
w8	59	40	162	103	59	1	2.554			
w9	67	47	103	36	67	1	0.771	1		1
w10	82	52	36	0	36	0.439	0	1	1	
w11	96	69	120	24	96	1	0.346	1	1	
w12	88	82	144	56	88	1	0.686	1	1	
w13	45	89	176	131	45	1	1.477	1	1	
w14	23	76	251	228	23	1	2.987			
w15	16	52	228	212	16	1	4.077			
w16	31	28	212	181	31	1	6.464			
w17	166	23	181	15	166	1	0.643	1	1	
w18	16	71	135	119	16	1	1.676	1		1
w19	34	71	119	85	34	1	1.197	1	1	
w20	137	72	205	68	137	1	0.944	1		1
w21	83	62	68	0	68	0.819	0	1	1	
w22	23	85	120	97	23	1	1.146	1		1
w23	110	81	97	0	97	0.882	0	1		1
w24	156	72	0	0	0	0	0	1	1	
w25	109	96	120	11	109	1	0.114	1		1
w26	102	125	11	0	11	0.1080	0	1	1	
w27	34	122	120	86	34	1	0.703	1		1
w28	69	82	86	17	69	1	0.208	1	1	
Total					1533	0.902	1.346	21	13	8

Table 3 shows requesting for productions and decisions about confirmation or missing the production for all four products in the entire observed period. Abbreviation used in the Table3 are the same as in the Table 2. In the last row total numbers of production request, conformation and missing are given for all the products. Production requests for all four products appeared in 9 out of 28 weeks and in 5 of them only two product were produced.

Table 3: Production decisions for 28 weeks for all products and minCSL criterion

	p1			p2			p3			p4		
Time	PR	PC	PM	PR	PC	PM	PR	PC	PM	PR	PC	PM
w1				1	1		1	1				
w2	1	1		1	1							
w3	1	1		1		1	1	1				
w4				1	1					1	1	
w5				1	1		1	1		1		1
w6	1		1	1	1		1	1				
w7	1	1		1	1		1		1	1	1	
w8				1	1		1	1		1		1
w9	1		1	1		1	1	1		1	1	
w10	1	1		1	1					1	1	
w11	1	1		1	1					1	1	
w12	1	1		1	1					1	1	
w13	1	1								1	1	
w14										1	1	
w15							1	1		1	1	
w16				1	1		1		1	1	1	
w17	1	1		1		1	1	1				
w18	1		1	1	1		1	1				
w19	1	1		1	1							
w20	1		1	1		1	1	1		1	1	
w21	1	1		1	1		1		1	1	1	
w22	1		1	1	1		1	1		1		1
w23	1		1	1	1		1	1		1		1
w24	1	1		1	1		1		1	1	1	
w25	1		1	1	1		1	1		1		1
w26	1	1		1	1		1		1	1	1	
w27	1		1	1	1		1	1				
w28	1	1		1		1	1	1				
Total	21	13	8	25	20	5	20	15	5	19	14	5

At strategic level, a decision about the most appropriate criterion can be made using the proposed algorithm. Computational results for all observed products and all four decision criteria are given in Table 4. Column "Delivered" represents quantities which are delivered during entire period per product. Columns "csl" and "sc" represents KPIs. The last column represents the percent of launched production requests.

The percentages of capacity utilizations for options 1, 2, 3 and 4, are 0.863, 0.857, 0.872 and 0.817, respectively. After algorithm results presentation, management of the observed company considers option 1 the most appropriate. Applying the first option (min CSL) in algorithm, the highest level of CSL and the largest quantity of delivered products are provided. As positive side effect, high level of capacity utilization is achieved even that was not included as criterion

Table 4: Summary results

		Delivered	csl	sc	card(P)/card(D)
Option 1 (min csl)	p1	1533	0.90	1.35	0.714
	p2	2263	0.91	0.98	0.800
	p3	2538	0.89	1.29	0.750
	p4	778	0.83	7.36	0.737
		total 7112	*avg 0.88*	*avg 2.75*	*avg 0.753*
Option 2 (min sc)	p1	1533	0.90	1.40	0.619
	p2	2240	0.93	1.05	0.833
	p3	2488	0.89	1.03	0.682
	p4	728	0.77	7.30	0.684
		total 6989	*avg 0.87*	*avg 2.70*	*avg 0.709*
Option 3	p1	1823	0.99	1.78	1
(max capacity utilization)	p2	1687	0.67	0.67	0.600
	p3	2940	0.98	2.44	1
	p4	575	0.59	2.08	0.591
		total 7025	*avg 0.81*	*avg 1.74*	*avg 0.765*
Option 4	p1	1823	0.99	1.78	1
(max number of product)	p2	1690	0.73	0.76	0.625
	p3	2263	0.80	1.32	0.619
	p4	879	0.91	11.20	1
		total 6655	*avg 0.86*	*avg 3.76*	*avg 0.782*

in option 1. Although the average results in the last column indicate the similar percentages of launched production requests for all four options, detailed analysis by products shows that even nearly 40% production requests for some products remain unrealized in options 3 and 4. Considering this indicator in the last column, option 1 gives the most balanced values.

6 Conclusions

The aim of this paper was to develop a heuristic algorithm for multi-period production planning based on supply chain KPIs: customer service level and stock cover. Analyzing FMCG company with limited production capacities, two important decisions are recognized in each period: which products should be produced (production requesting) and which product can be produced (production launching). The proposed algorithm provides support for both decisions. The first decision is based on two used KPIs, while the second decision can be made by using one of four criteria: minCSL, minSC, maximal capacity utilization and maximal number of products.

Developed algorithm was applied in real FMCG where option 1 was chosen as the most appropriate according to their business policy. However, the main advantage of the algorithm is the fact that it offers a choice among four different decision criteria based on company's business policies. It can be extended in order to generate demand forecast based on different forecasting techniques and adding new KPI: demand forecast accuracy, which will be used for evaluation of the impact of forecast accuracy on customer service level and stock cover variations.

Bibliography

[1] Babarogić, S.; Makajić-Nikolić, D.; Lečić-Cvetković, D.; Atanasov N. (2012); Multi-period Customer Service Level Maximization under Limited Production Capacity, *International Journal of Computers Communications & Control*, ISSN 1841-9836, 7(5): 798-806.

[2] Dellaert N.P., Jeunet J. (2003); *Demand forecast accuracy and performance of inventory policies under multi-level rolling schedule environments*, Research at International Institute of Infonomics, Heerlen, The Netherlands.

[3] Fernandez, R.; Gouveia, J. B.; Pinho, C. (2010); Overstock - A Real Option Approach, *Journal of Operations and Supply Chain Management*, ISSN 1984-3046, 3(1): 98-107.

[4] Galbraith, J. R. (1973); *Designing Complex Organizations*, Reading, MA: Addison-Wesley.

[5] Ho, C. (1989); Evaluating the Impact of Operating Environments on MRP System Nervousness, *Int J Prod Res*, ISSN 0020-7543, 27(7): 1115-1135.

[6] Jeffery M.M., Butler J.R., Malone C.L. (2008); Determining a cost-effective customer service level, *Supply Chain Management: An International Journal*, 13: 225-232.

[7] Lane, R.; Szwejczewski, M. (2000); The Relative Importance of Planning and Control Systems in Achieving Good Delivery Performance, *Prod Plan Control*, ISSN 0953-7287, 11(5): 422-433.

[8] Lečić-Cvetković, D.; Atanasov, N.; Babarogić, S. (2010); An Algorithm for Customer Order Fulfillment in a Make-to-Stock Manufacturing System, *International Journal of Computers Communications & Control*, ISSN 1841-9836, 5(5): 983-791.

[9] Lin, J.; Chen, J.H. (2005); Enhance Order Promising with ATP Allocation Planning Considering Material and Capacity Constraints, *Journal of the Chinese Institute of Industrial Engineers*, ISSN 1017-0669, 22(4): 282-292.

[10] Makajić-Nikolić, D.; Lečić-Cvetković; Atanasov, N.; Babarogić, S. (2013); An Approach to Production Planning for Supply Chain Performance Improvements, *Proceedings of XI Balkan Conference on Operational Research*, ISBN 978-86-7680-285-2, 357-366.

[11] Meyr, H. (2009); Customer Segmentation, Allocation Planning and Order Promising in Make-to-Stock Production, *OR Spectrum*, ISSN 0171-6468, 31(1): 229-256.

[12] Okulewicz, J. (2009); Verification of a Service Level Estimation Method, *Total Logistics Management*, ISSN 1689-5959, 2: 67-78.

[13] Shen Z-Y.M., Daskin M. (2005); Trade-offs Between Customer Service and Cost in Integrated Supply Chain Design, *Manufacturing and Service Operations Management*, 7(3): 188-207.

[14] Wadhwa S., Kanda A., Bhoon K.S. (2006); Bibhushan Impact of Supply Chain Collaboration on Customer Service Level and Working Capital, *Global Journal of Flexible Systems Management*, 7(1-2): 27-35.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):721-729, December, 2014.

Framework for Automated Reporting in EU funded Projects

A. Mihăilă, D. Benţa, L. Rusu

Alin Mihăilă*, Lucia Rusu
Faculty of Economics and Business Administration
Babes-Bolyai University of Cluj-Napoca
alin.mihaila@econ.ubbcluj.ro, lucia.rusu@econ.ubbcluj.ro
*Coresponding author: alin.mihaila@econ.ubbcluj.ro

Dan Benţa
Faculty of Law and Economics
Agora University of Oradea
dan.benta@univagora.ro

Abstract: Reporting processes in EU funded projects involve a huge amount of financial operations. This time consuming procedure can be easily automated as it involves repetitive operations. The novelty of the paper consists in the presentation of AutoFiState software application for automatic financial data capture in projects financed through ESF (European Social Fund) in Romania. The AutoFiSate application was developed on the basis of results obtained in the fields of automated testing and scripting languages. The use of the AutoFiState software application leads to maximum effectiveness in how ESF funds are used by reducing the time needed to draw up the financial reports and the related labor costs. Using scripting languages to develop such reporting-support programs we can improve reporting and save employees effort and time.
Keywords: automatic financial data capture, automated testing, scripting languages

1 Introduction

It is widely known that Romania has a very low degree of EU funds absorption. There are many reasons for the failure to absorb the EU's so-called structural funds and it is not the objective of this paper to discuss them. We will like to address just one of the problems and to suggest a possible solution in order to have projects with greater capacity of absorption. From our own experience with these kinds of projects, one of the main problems in EU funded projects conducted by the universities is the reporting procedure of financial evidence when the requests for reimbursement are made. This is a time consuming activity that needs high cognitive skills such attention to small details. In general, in a reporting period, several EU funded projects are handled by a single person, the financial official from projects' beneficiary. Moreover, financial evidence and reporting should be done by people with great expertise and most of the time these types of tasks cannot be transferred to another team member. We had to experience all this problems and that is why we were interested in this field and finally we managed to develop a software solution to divide reporting tasks in order to facilitate the work of financial officials in beneficiary offices.

Test automation [1], [2], [3], [4] is a significant area of investment and the market awareness of highly automated testing is very high. As some white papers present [5], [6] *you can get the most benefit out of your automated testing efforts by automating: repetitive tests that run for multiple builds, tests that are highly subject to human error, tests that require multiple data sets, frequently-used functionality that introduces high risk conditions, tests that are impossible to perform manually, tests that run on several different hardware or software platforms and*

configurations, tests that take a lot of effort and time when doing manual testing. Moreover, authors [6] describe best practices for script automation or for cross-site scripting attacks [7].

As reporting operations are repetitive tasks, using scripting languages to develop reporting-programs proved to be a great solution for many other businesses, inclusive for our own projects. In a market research report made in 2013 by Musier & Javed [8], automated testing delivers business benefits in multiple areas for most business companies. More than 4 out of 5 firms using test automation identified business benefits of test automation in multiple areas (86%), with most respondents identifying 3 to 6 different areas of benefits. The 5-top ranked areas of value were: 1) greater staff efficiency and time savings; 2) Early identification of defects before business users are impacted; 3) Higher quality in business processes and the software that supports them; 4) Greater accuracy in catching more defects; 5) Faster deployment of innovation and new features for business users.

The authors also mention that testing is increasingly seen as an essential competency for most of the global companies. More than that, harvesting economic benefits will continue to drive the industry's shift toward highly automated testing and away from manual approaches, as companies continue to push for higher quality execution and greater business agility at lower cost.

Beside the aspects mentioned, an important aspect for any work field is the dynamics of the group. This concept is an essential factor in long term projects as it can affect the project performance. Because some staff is short time employed and the same task may be allocated to several employees during the implementation period, aspects of group dynamics become even more important [9].

The reason tackled in this paper is the fulfilment of financial statement demanded when applications for reimbursement (refund) are submitted. The current legislation requires that, for each project implemented by a public institution, each application for reimbursement (refund) should be submitted no later than three months after the first payment is made [10]. Drawing up a funding application is a complex and time-consuming activity. In big organizations, and not only, where ESF-funded projects are implemented, the job of financial officials who draw up funding applications is very difficult and demanding. To support them during the stage of financial recording, the present paper proposes the automation of the process of data input in the reporting system of the managing authority in Romania for the Sectorial Operational Programme Human Resources Development. The reporting system used in Romania by the Managing Authority for the Sectorial Operational Programme Human Resources Development is an online software application called Action Web. Consequently, the automation of the process of data input into the reporting system represents the automation of the process of filling in a web form which, in turn, is similar to automated testing that is appropriate for [5] tests that are highly subject to human error and/or tests that take a lot of effort and time when performed manually. The main advantage of automatizing the financial reporting operations lies in greater staff efficiency and time saving [8].

The novelty of the paper consists in the presentation of AutoFiState software application for automatic financial data capture in projects financed through ESF in Romania. To develop the AutoFiState software application for the automatic data input into the Action Web to prepare the financial statement, use was made of results in the fields of automated testing [1], [2], [3], [4], [5], [6], [8] and scripting languages [6], [7], [11]. The structure of the present paper is as follows: Section 2-Problem statement; Section 3-Presentation of the solution to the problem and of the AutoFiState software application; Section 4-Presentation of tests and results; Section 5-Conclusions.

Within this framework and taking into consideration that a project can be very difficult to manage without proper instruments, we present our solution for data validation and reporting

process in order to avoid human reporting mistakes and to reduce data input time in EU projects. We also consider that our solution helps with the problems that can arise with the short time employment, helps to improve the dynamic of employees and to build trust between partners in EU funded projects.

2 Problem statement

In the case of ESF-funded projects implemented by public institutions, the lead partner must submit, no later than three months after the first payment is made during the reimbursement period, an application for reimbursement consisting of several documents among which is the financial statement. The financial statement contains data about all the expenditures made by all partners in the project during the reimbursement (refund) period. The data obtained from the financial officials of all partners are then entered by the project coordinator into the Action Web software which automatically generates the financial statement in PDF file format (Figure 1). For each type of expenditure made in the project (human resources, participants, other costs, indirect expenses), data must be entered for the documents certifying the commitment of expenditures (date of issue, number of documents, supplier, tax identification number, description, annual budget, budget line - expenditure category, payer, currency, VAT-inclusive price, VAT-exclusive price, VAT-exclusive project costs, VAT) and for the documents certifying the payment was made (date of payment, number of payment document, currency, VAT-inclusive price, VAT-exclusive price).

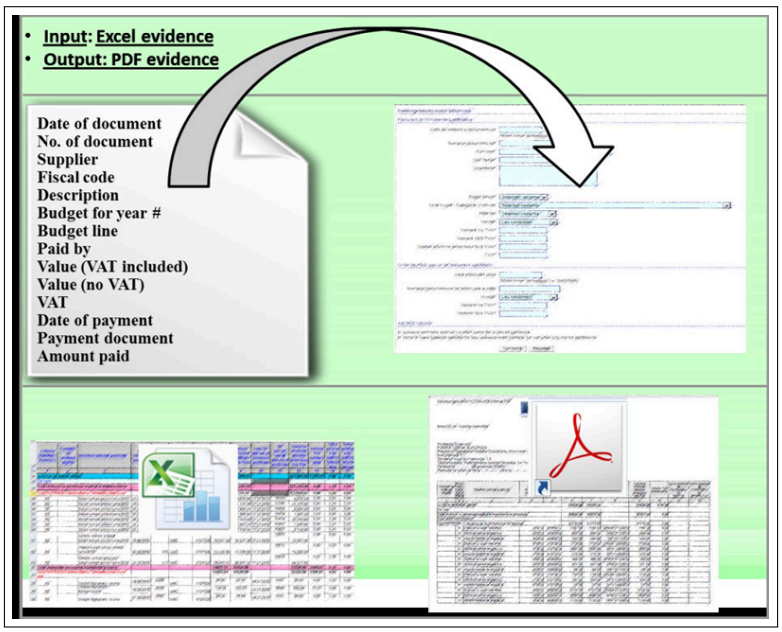


Figure 1: Input and output for reporting

Given that:

• data input into the Action Web application is a time-consuming operation,

• the data to be entered into Action Web follow the same structure/format for all types of expenditures, and

• the Action Web application does not feature the automatic data capture from a file of a particular format/structure, the opportunity was opened up to develop a software application entitled AutoFiState that will be used to automatically enter data into Action Web.

By using the AutoFiState software application:

• the time needed to enter data into Action Web is reduced, and

• the time needed to identify and correct errors made when entering data into Action Web is also reduced.

As a direct consequence thereof, the labor costs for drawing up each financial statement are reduced.

3 Software Design and Implementation

The preparation of a financial statement using the AutoFiState software application is accomplished in three stages (Figure 2):

• The preparatory stage during which the financial officials of all partners in the project prepare the financial data related to all expenditures made by partners and the project coordinator centralizes financial data collected from the financial officials of the partners;

• The AutoFiState stage during which the project coordinator automatically enters the centralized financial data into the Action Web application;

• The final stage during which the financial official of the lead partner (beneficiary) generates the financial statement for the entire project and prepares it to be approved by the legal representative of the project.

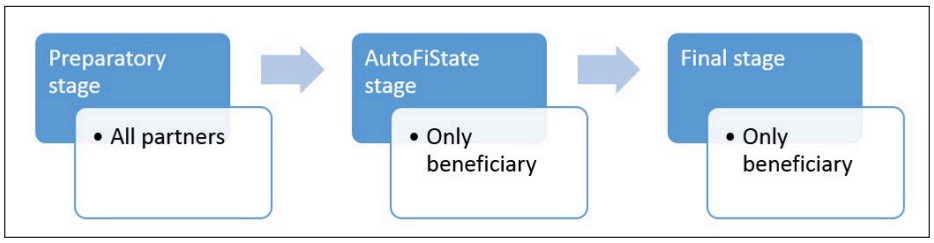


Figure 2: Financial statement generation phases

All mentioned steps involve more than one employee and the beneficiary financial officials' job is to check the whole evidences and costs asking for supporting documents. In our case, we have developed and tested this solution, for the first time, in an EU funded project with five partners (our university as beneficiary and four other partners). We are describing in detail the performed operations in a reporting period.

a) During **the preparatory stage** (Figure 3), the financial officials of each partner draw up their own financial statements by entering the data about each category of project expenditures into an Excel template file. Some data must be entered into the Excel template in the format required by the Action Web application. For instance, calendar dates must be entered in the "DD/MM/YYYY" format and the numeric data should not use any separator except for the dot (".") character. The financial official of the beneficiary receives the financial statements from each partner and checks the eligibility of all project expenditures on the basis of supporting documents. After the eligibility of expenditures for all partners has been checked, the financial official of the beneficiary forwards the Excel templates containing the related financial statements to an operator who automatically enters the data into Action Web using the AutoFiState application.

b) During **the AutoFiState stage** (Figure 4), an operator of the beneficiary checks whether the data in the Excel template file containing the financial statements of each partner comply with the formats of the Action Web application. If some data are entered not according to the required format, the operator processes the data so that they may comply with the format required by the Action Web application. After the data in all Excel template files have been found to comply

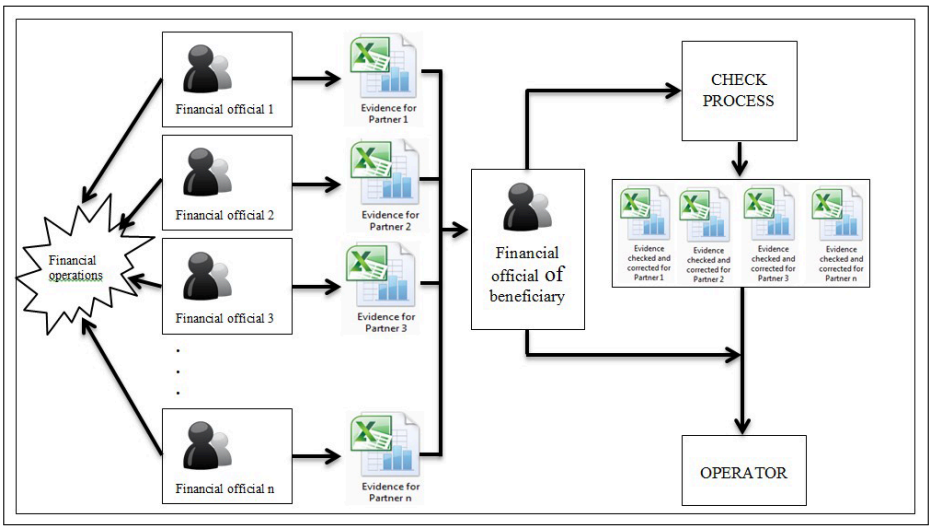


Figure 3: The preparatory stage

with the format required by the Action Web application, they then are transferred to a central Excel file. All fields of the central Excel file are set at identical height and width because the AutoFiState script uses the screen coordinates to copy data from the central Excel file into the Action Web application. While the operator inputs all the records, beneficiary financial official can perform other mandatory operations and prepare the requested for the reimbursement (for instance, documents for the external auditor).

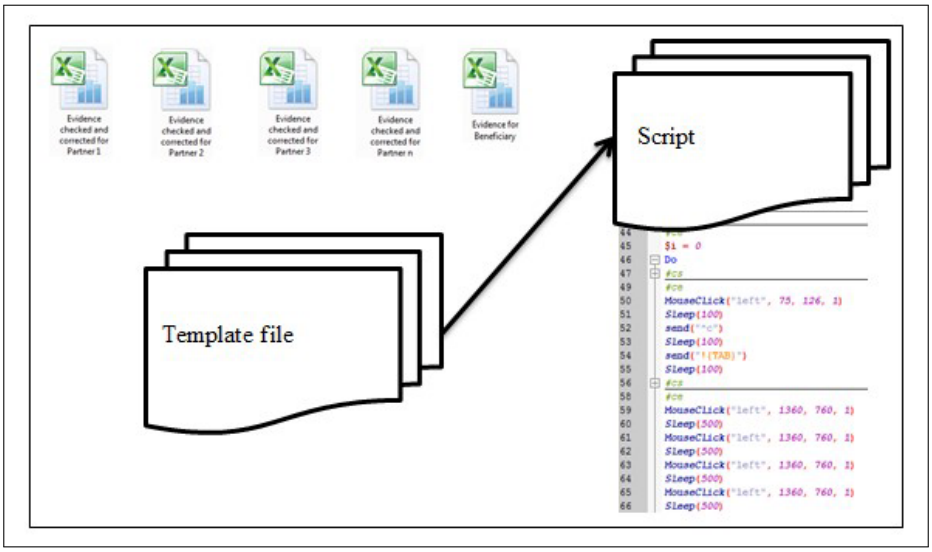


Figure 4: The AutoFiState stage

The AutoFiState application reads the central Excel file row by row so that at a given moment a single row is active or current. Initially, the current row is represented by the first row of the central Excel file. The AutoFiState application copies/captures the data from the current row into the Action Web software. The data from the current row are copied cell by cell, from the left to the right of the row. After having copied the data from the current row, the AutoFiState application waits until the data are saved on the server of the managing authority and sets the next row of the central Excel file as the current row. The new current row is set by clicking on the

vertical scroll bar so that the next row of the central Excel file becomes the first row displayed on the screen. This process is repeated until all rows of the central Excel file are copied into the online application of the managing authority. The AutoFiState application should be run under supervision because either the computer or the Action Web application or both are sometimes slow in responding. In this case, the AutoFiState application will be closed and then reopened from the record that failed. Closing and reopening the AutoFiState application is made using the hotkeys/predefined keys from AutoIT scripts.

To develop and use the AutoFiState application, the following software applications were used:

• Microsoft Office Excel, 2010 version, for data preparation by the financial officials of all partners to be entered into the Action Web software;

• Mozilla Firefox, version 15.0.1, to access the Action Web application;

• Auto IT scripting package, version v3.3.8.1, to develop and use effectively the AutoFiState application (during the development stage, Window Info Cursor Position Tool proved extremely helpful).

In order to properly use the AutoFiState application, several prerequisites/conditions must be met:

• Only Microsoft Excel file and reporting platform should be opened - for our test we used Office 2010 package;

• Use Mozilla Firefox to open reporting platform because Full Screen result differs when IE or Google Chrome browsers are used;

• Reporting platform should run in Full Screen (clicking coordinates were defined in an Full Screen operation);

• Minimize the Ribbon should be activated in Microsoft Excel 2010;

• Auto-hide the taskbar option should be checked in Taskbar and Start Menu Properties.

Those restrictions assure a proper script running and reporting according to Excel file records. With this pre-settings made, the script runs.

Main operations performed by reporting script (Figure 5) are coordinated by hotkeys in AutoIT scripts. A hotkey press for such scripts calls a user function that may pauses or interrupts current AutoIT function. Most used and defined hotkeys in AutoIT scripts are for pause: HotKeySet("PAUSE", "TogglePause") and for terminate: HotKeySet("ESC", "Terminate"). In our case, we have used "ESC" and terminate function to stop script running (Figure 5a). In case of processing delays or server slow response, the script can be stopped and restarted from the last selected record. A message window with some information is defined, then, the script reporting phase is initialized (from 0 to n-1, where n is the number of records from Excel file). Further, the script runs and each record from Excel file is inputted in the reporting web form (Figure 5b). After last record, an increased waiting time (sleep time) is defined in order to secure the web server response time and loading reporting form to enter the next record (Figure 5c). The application crosses financial evidence file in row order. Then, all specific operations for the first row are applicable to the next one. To assure this, after finishing a row, the next row should be repositioned. For this to happen, the application clicks once the roll down bar from Excel so that second row became first displayed row. This process repeats until the last record is inputted in the reporting web form.

c) The automatic financial data capture from the central Excel file into the Action Web application is followed by the final stage (Figure 6) during which the financial official of the beneficiary generates the financial statement from the online application of the managing authority and submits it for approval by the legal representative. Before report generation, incomes must be inputted. Later, along with other documents, the reimbursement (refund) request is send to authorities.

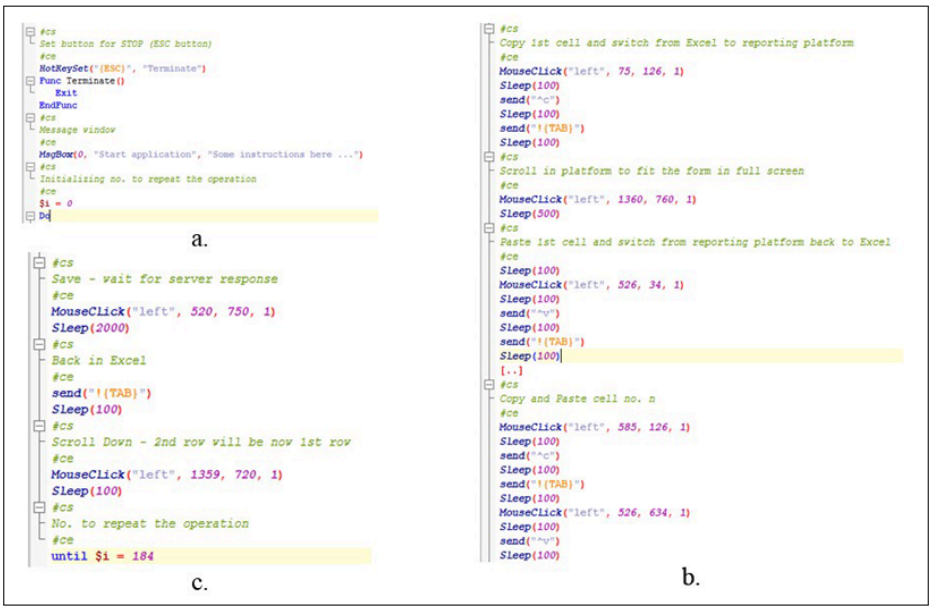


Figure 5: AutoIT script: a. Safety settings; b. Automated reporting; c. Save and repeat.

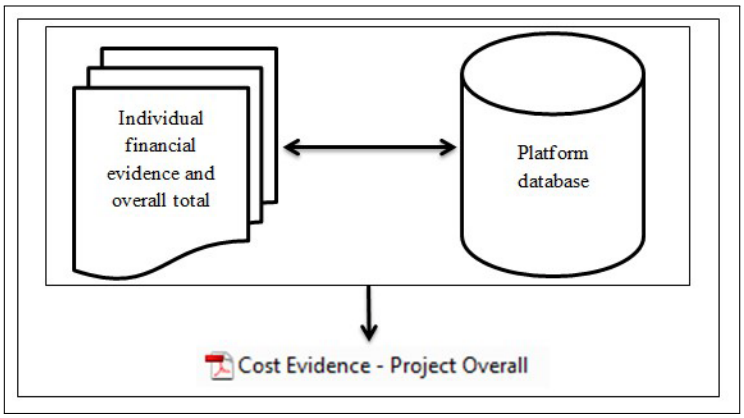


Figure 6: The final stage

4 Tests Results

The AutoFiState application has been used with four ESF projects financed through the Sectorial Operational Programme Human Resources Development:

Only the AutoFiState software application was used to enter the data because, for each project, the amounts of data were relatively large and the available time needed to draw up a funding application and prepare a financial statement was not sufficient as regards entering the data into the Action Web application both manually and automatically.

Leaving aside the time needed to discover possible mistakes due to manual data entry and correct them, it has been noticed that 142 hours have been saved by using the AutoFiState application. Since the gross hourly salary of a financial official amounts to 150 lei, then the saving amounts to 21300 lei.

Table 1: Data test

Project ID	No. of partners	No. of months	Budget (in million Euros)	Status
1	5	36	5	Closed
2	2	29	4,5	Ongoing
3	2	36	3,5	Closed
4	9	36	5	Closed

Table 2: Test results

Project ID	Manual time* (in hours)	AutoFiState time (in hours)	Manual error	AutoFiState error	No. of records
1	111	40	$\geq 0\%$	0%	4767
2	30	11	$\geq 0\%$	0%	1301
3	23	8	$\geq 0\%$	0%	978
4	57	20	$\geq 0\%$	0%	2435

5 Conclusions

Automated reporting shortened our reporting period and improve the quality of reporting tasks. The current version we developed is good enough to make the reporting procedure of all stored records in an efficient and clear manner.

There are various testing and automation software solutions that may improve such repetitive reporting tasks. The implemented solution constitutes a solid foundation for our future work.

As web browsers update is regular made, sometimes defined coordinated doesn't perfect match and should be redefined. A solution to resolve this problem may be implemented. This solution have two major advantages: saved financial official time with more than 50% and offers dynamic collaborative framework for projects' partners. It can be applied to all EU funded projects that need this reporting procedure and can also be applied successfully in national research projects (ANCS, CNCSIS etc.).

Reporting procedures in EU funded projects needs careful planning and design work as a huge amount of time and effort is involved. Using this reporting-support application, a full range of requests is considered. The software enhances the reporting process by increasing data validation and efficiency, removing reporting complexity and lowering costs. It can be adapted and customized for different reporting procedures in projects when repetitive data input operations are needed.

The use of the AutoFiState software application leads to maximum effectiveness in how ESF funds are used by reducing the time needed to draw up the financial reports and the related labor costs. The AutoFiState software application can be improved to the point where it no longer depends on the regular updates of the web browser. The AutoFiState application can be used for all ESF-funded projects in Romania, whose financial reports are forwarded to the managing authority via the Action Web application. The AutoFiState software can also be slightly adjusted so that it can be used successfully in other EU-funded or government-funded projects.

Acknowledgement

The work presented has been founded by the research Grant FP7 AAL NITICS and was financed by IT Center for Science and Technology, Romania, as a partner in this program.

Bibliography

[1] Jureczko, M. (2008). The Level of Agility in the testing process in a large scale financial software project. Software engineering techniques in progress, Oficyna Wydawnicza Politechniki Wroclawskiej, 139-152.

[2] Li, K., & Wu, M. (2004). Available GUI Testing Tools vs. Proposed Tool. *Effective GUI Test Automation: Developing an Automated GUI Testing Tool*. SYBEX.

[3] Myers, G. J., Badgett, T., Thomas, T. M., & Sandler, C. (2004). *The Psychology and Economics of Program Testing. The Art of Software Testing-Second Edition*, John Wiley & Sons, ISBN 0-471-46912-2..

[4] Dustin, E. (2003). Automated Testing Tools. *Effective Software Testing: 50 Specific Ways to Improve Your Testing*. Pearson Education, ISBN 0-201-79429-2, 137-154.

[5] Juniper Networks (2008), Increasing Network Availability with Automated Scripting, Available on-line http://support.neoteris.com/solutions/literature/white_papers/200252.pdf

[6] SmartBear Software (2013), 6 Tips to Getting Started with Automated Testing, White Paper, Available on-line http://smartbear.com/SmartBear/media/pdfs/6_Tips_for_Automated_Test.pdf

[7] Van Acker, S., Nikiforakis, N., Desmet, L., Joosen, W., & Piessens, F. (2012). FlashOver: Automated Discovery of Cross-site Scripting Vulnerabilities in Rich Internet Applications, ASIACCS '12, May 2-4, 2012, Seoul, Korea, ACM.

[8] Musier, R., Javed, S. (2013). 2013 Trends in Automated Testing For Enterprise Systems, pp. 3-4, Available on-line http://www.worksoft.com/files/resources/Worksoft-Research-Report-2013-Trends-in-Automated-Testing.pdf

[9] Levi, D. (2011). *Group Dynamics for Teams*. Sage Publications, 3rd Edition, ISBN: 9781412977623

[10] Romanian Government Emergency Ordinance no 120 from December 23, 2010, Art. 5^1.

[11] AutoIT, http://www.autoitscript.com/site/autoit

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):730-740, December, 2014.

Uncertain Query Processing using Vague Set or Fuzzy Set: Which One Is Better?

J. Mishra, S. Ghosh

Jaydev Mishra*
Computer Science and Engineering Department
College of Engineering and Management, Kolaghat
West Bengal-721171, India
*Corresponding author:jsm03@cemk.ac.in

Sharmistha Ghosh
Galgotias University, Greater Noida
Uttar Pradesh-201306, India
sharmisthag@yahoo.com

Abstract: In this paper we attempt to make a theoretical comparison between fuzzy sets and vague sets in processing uncertain queries. We have designed an architecture to process uncertain i.e. fuzzy or vague queries. In the architecture we have presented an algorithm to find the membership value that generates the fuzzy or vague representation of the attributes with respect to the given uncertain query. Next, a similarity measure is used to get each tuples similarity value with the uncertain query for both fuzzy and vague sets. Finally, a decision maker will supply a threshold or α-cut value based on which a corresponding SQL statement is generated for the given uncertain query. This SQL retrieves different result sets from the database for fuzzy or vague data. It has been shown with examples that vague sets give more accurate result in comparison with fuzzy sets for any uncertain query.
Keywords: uncertain data, similarity measures, fuzzy/vague interpreter.

1 Introduction

In the real world, vaguely specified data values appear in many applications such as sensor information, expert systems, decision analysis, medical sciences, management and engineering problems and so on. Fuzzy set theory has been proposed to handle such vagueness by generalizing the notion of membership in a set. Essentially, in a fuzzy set each element is associated with a point-value selected from the unit interval [0, 1], which is termed as the grade of membership in the set. A vague set, which is conceived as a further generalization of fuzzy set, uses the idea of interval-based membership instead of point-based membership as in the case of fuzzy sets. The interval-based membership in vague sets is more expressive in capturing vagueness of data.

Relational database systems have been extensively studied worldwide since Codd [1] had proposed the relational data model in 1970. Based on this model, several commercial relational database systems are available (see [2]- [4]). This data model usually takes care of precisely defined and unambiguous data. However, in the real world applications data are often partially known i.e., incomplete or imprecise. For example, instead of specifying that the height of David is 188 cm, one may say that the height of David is around 190 cm, or simply that David is tall. Other examples on uncertain data may be “Salaries of almost equally experienced employees are more or less the same" etc. All these are informative statements that may be useful in answering queries or making inferences. However, such type of data cannot be represented in the classical relational data model. In order to incorporate imprecise or uncertain data, the classical relational data model has been extended by several authors on the mathematical framework of fuzzy set theory which was initially introduced by Zadeh [5] in 1965. Based on this fuzzy set theory, various

fuzzy relational database models, such as similarity-based relational model [6], possibility-based relational model [7] and some types of hybrid data models [8] have been proposed to model fuzzy information in relational databases. However, the most important issue in the utilization of any database system lies in its ability to process information and queries correctly. Several authors [9]- [12] have contributed to provide a theoretical contribution to query language for a fuzzy database model. In particular, Bosc et al. [11] and Nakajima [12] have extended the well known SQL language in the framework of fuzzy set theory and have developed a fuzzy SQL language, called SQLf.

It is believed that vague sets, proposed by Gau et al. [13] in 1993, that use interval-based membership values have more powerful ability to process imprecise information than traditional fuzzy sets. Thus the notion of vague sets has also been incorporated into relations in [14] and a vague SQL (VSQL) has been described. The VSQL allows the users to formulate a wide range of queries that occur in different modes of interaction between vague data and queries. In [15], Zhao and Ma have proposed a vague relational database model which is an extension of the classical relational model. Based on the proposed model and the semantic measure of vague sets, they have also investigated vague querying strategies and have given the form of vague querying with SQL.

In this paper, we have made an attempt to make a theoretical comparison between fuzzy sets and vague sets in processing uncertain queries. Firstly, we have designed an architecture to test uncertain queries. Next, we have presented an algorithm to retrieve membership values for imprecise data represented by fuzzy or vague sets. A similarity measure is then used to calculate each tuple's similarity value with the uncertain query for both fuzzy and vague sets. Finally, the decision maker provides a threshold value or α-cut based on which a corresponding SQL statement is generated for the given uncertain query. This SQL retrieves different result sets from the database for fuzzy data or vague data. In the present study, we have considered an Employee database and processed some uncertain queries using fuzzy data as well as vague data. Each time it has been observed that vague sets give more accurate result in comparison to fuzzy sets.

The rest of the paper is organized as follows. Section 2 presents some basic definitions related to fuzzy and vague sets. Similarity measure between two vague data is also defined in the same section. Section 3 represents an architecture for processing uncertain queries. In Section 4, an algorithm has been designed to get the appropriate membership value and represent domain value of fuzzy or vague attributes into fuzzy form or vague form. Section 5 establishes that a vague set is more appropriate than fuzzy set with real life examples. The concluding remarks appear in Section 6.

2 Basic Definitions

In this section, we introduce some basic concepts related to fuzzy and vague sets and similarity measure of two vague sets which have been utilized throughout the paper. Let U be the universe of discourse where an element of U is denoted by u.

2.1 Fuzzy Set

Definition 1. A Fuzzy set F in the universe of discourse U is characterized by a membership function $\mu_F : U \to [0, 1]$ and is defined as a set of ordered pairs $F = \{\langle u, \mu_F(u)\rangle : u \in U\}$ where $\mu_F(u)$ for each $u \in U$ denotes the grade of membership of u in the fuzzy set F.

2.2 Vague Set

Definition 2. A vague set V in the universe of discourse U is characterized by two membership functions given by:
slowromancapi@. a truth membership function $t_V : U \to [0, 1]$ and
slowromancapii@. a false membership function $f_V : U \to [0, 1]$,
where $t_V(u)$ is a lower bound of the grade of membership of u derived from the 'evidence for u', and $f_V(u)$ is a lower bound on the negation of u derived from the 'evidence against u', and $t_V(u) + f_V(u) \leq 1$. Thus the grade of membership $\mu_V(u)$ of u in the vague set V is bounded by a subinterval $[t_V(u), 1 - f_V(u)]$ of $[0, 1]$, i.e., $t_V(u) \leq \mu_V(u) \leq 1 - f_V(u)$. Then, the vague set V is written as $V = \{\langle u, [t_V(u), 1 - f_V(u)]\rangle : u \in U\}$. Here, the interval $[t_V(u), 1 - f_V(u)]$ is said to be the vague value to the object u and is denoted by $V_V(u)$.

For example, in disease diagnosis process of a medical system, the vague value $[0.3, 0.6]$ can be interpreted as "the report of disease in favour is 30%, against is 40% and another 30% is indeterminable". The precision of knowledge about u is clearly characterized by the difference $(1 - f_V(u) - t_V(u))$. If this is small, then the knowledge about u is relatively precise. However, if it is large, we know correspondingly little. If $t_V(u)$ is equal to $(1 - f_V(u))$, the knowledge about u is precise, and vague set theory reverts back to fuzzy set theory. If $t_V(u)$ and $(1 - f_V(u))$ are both equal to 1 or 0, depending on whether u does or does not belong to V, the knowledge about u is exact and the theory goes back to that of ordinary set. Thus any crisp or fuzzy set may be considered as a special case of vague sets.

2.3 Similarity Measure

There have been some studies in literature which discuss the topic concerning how to measure the degree of similarity between vague sets [16]- [19]. In [19] it was pointed out by Lu et al. that the similarity measures defined in [16]- [18] did not fit well in some cases. They have proposed a new similarity measure between vague sets which turned out to be more reasonable in more general cases. The same has been used in the present work which is defined as follows:

Definition 3. Similarity Measure between two vague values
Let x and y be any two vague values such that $x = [t_x, 1 - f_x]$ and $y = [t_y, 1 - f_y]$, where $0 \leq t_x \leq 1 - f_x \leq 1$, and $0 \leq t_y \leq 1 - f_y \leq 1$. Let $SE(x, y)$ denote the similarity measure between x and y. Then

$$SE(x, y) = \sqrt{(1 - (|(t_x - t_y) - (f_x - f_y)|/2))\,(1 - |(t_x - t_y) + (f_x - f_y)|)}.$$

Definition 4. Similarity Measure between two vague sets
Let $U = \{u_1, u_2, u_3, \cdots, u_n\}$ be the universe of discourse. Let A and B be two vague sets on U, such that $A = \{< u_i, [t_A(u_i), 1 - f_A(u_i)] >, \forall u_i \in U\}$, where $t_A(u_i) \leq \mu_A(u_i) \leq 1 - f_A(u_i)$ and $1 \leq i \leq n$. $B = \{< u_i, [t_B(u_i), 1 - f_B(u_i)] >, \forall u_i \in U\}$, where $t_B(u_i) \leq \mu_B(u_i) \leq 1 - f_B(u_i)$ and $1 \leq i \leq n$. Now, the similarity measure between A and B, denoted by $SE(A, B)$ is defined as:

$$SE(A, B) = \frac{1}{n} \sum_{i=1}^{n} SE\left([t_A(u_i), 1 - f_A(u_i)], [t_B(u_i), 1 - f_B(u_i)]\right) = \frac{1}{n} \sum_{i=1}^{n} \sqrt{P * Q}$$

where $P = (1 - (|(t_A(u_i) - t_B(u_i)) - (f_A(u_i) - f_B(u_i))|/2))$

and $Q = (1 - |(t_A(u_i) - t_B(u_i)) - (f_A(u_i) - f_B(u_i))|)$

3 Architecture for Processing Uncertain Query

Below is the architecture for processing an imprecise or uncertain query.

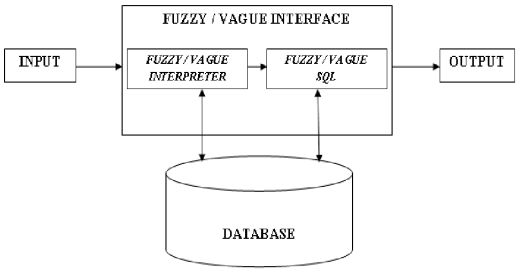


Figure 1: Uncertain Query Processing Architecture

Working principle of all components of the above proposed architecture is given below:

Input: A relational database with fuzzy or vague attributes, uncertain query, threshold value or α-tolerance value given by Decision Maker.
Fuzzy/Vague Interface:
It has two components, namely, **Fuzzy/Vague Interpreter** and **Fuzzy/Vague SQL**.
Fuzzy Interpreter: In this phase, the fuzzy attributes as well as the fuzzy data are identified from the given fuzzy query. Next, the fuzzy interpreter represents all domain values of each of the fuzzy attributes. **Algorithm 1** presented below in section 4 is then used to determine membership value for each domain value of all the fuzzy attributes. This gives us the fuzzy representation of the attributes with respect to the fuzzy data identified from the given query. The above fuzzy representation is then converted to a corresponding vague form whose truth membership value is same as the membership calculated in the fuzzy representation and false membership value is (1-truth membership value). After that a suitable similarity measure formula is to be used to measure the similarity between vague representation of each fuzzy attribute and the corresponding vague representation of the relevant fuzzy data given in the uncertain query. The same method will be applied for all fuzzy attributes appearing in the fuzzy query. If the query has more than one fuzzy attribute, then the similarity measure of tuples is obtained as the intersection of the similarity measures for each attribute.
Vague Interpreter: In this case, the vague attributes and the vague data are identified from the given uncertain query. Next, the vague interpreter will represent all domain values of each of the vague attributes. **Algorithm 1** is now used to get the truth membership values of the vague attributes while the decision maker will supply the false membership values with the condition that the sum of truth and false membership values should not exceed 1. After that, as before, a similarity measure formula is used to measure the similarity between each vague representation of a domain value and vague data given in the query. If the query has more than one vague attribute, once again the intersection of the similarity measures for individual attributes will give the similarity measure of tuples.
Fuzzy/Vague SQL: In this phase, the decision maker will supply a threshold value or α-cut value based on which a corresponding SQL statement for the given uncertain i.e., fuzzy or vague query will be generated.
Output: Finally, the SQL generated above is submitted to the database to get the desired result.

4 Algorithm for finding Membership Value

The following algorithm finds the membership value for each domain value of fuzzy or vague attributes with respect to fuzzy or vague data given in the uncertain query.

Algorithm 1 Membership value calculation

Input: Fuzzy/Vague attributes and Fuzzy/Vague data given in the uncertain query.
Output: A membership value in the interval [0,1].
Method: First find the fuzzy/vague attributes from the fuzzy/vague query.
for each fuzzy/vague attribute **do**
begin
fdata ← data value for the fuzzy/vague attribute of the query
range = maxDomainValue - minDomainValue
avg ← mean value of the domain set of the fuzzy/vague attribute
B ← avg
while(avg ≤ range) **do**
begin
avg = avg + B
end while loop
for each tuple of the relation **do**
begin
tupleValue ← corresponding tuple value from the fuzzy/vague attribute domain
membershipValue = 1-(|fdata-tupleValue| / avg)
end for loop of tuple
end for loop of fuzzy/vague attribute

5 Vague Sets have an extra edge over Fuzzy Sets

In this section, we have experimentally shown with real life examples that vague sets give more accurate result than fuzzy sets. To illustrate this fact, we have considered the following Employee EMP relational database:

Table 1: EMP Relation

Name	Age (yrs)	Exp (yrs)	Sal (Rs)
Prof. Smith	25	1	20000
Prof. Ganguly	52	25	55000
Prof. Roy	38	15	38000
Prof. David	48	23	53000
Prof. Maity	34	10	32000
Prof. Das	30	4	27000
Prof. Ahuja	50	26	55500
Prof. Sharma	51	16	40000
Prof. Kundu	45	22	50000
Prof. Dutta	54	33	80000

Next we consider following uncertain queries to explain that vague sets give better result in

comparison to fuzzy sets.

Uncertain query 1: "Find the details of the Professors whose age is around 50".

i) **Solution with Fuzzy Sets:** In the above *uncertain query 1*, fuzzy attribute is ***Age*** and fuzzy data is ***around 50***. Now, we apply our **algorithm 1** to get the membership value corresponding to each domain value of fuzzy attribute ***Age***.
Input: Algorithm needs the following two inputs: fuzzy attribute ***Age*** and fuzzy data ***around 50***.
Method: Calculation of membership value for each tuple value of fuzzy attribute ***Age*** based on fuzzy data ***around 50***.
dom(Age) = $\{25, 52, 38, 48, 34, 30, 50, 51, 45, 54\}$
given fdata=50
range = 54 - 25 = 29
Avg = 42.7
B = 42.7
Avg$\geq$ range then Avg remain same i.e., Avg = 42.7
Now, we need to find the membership value using the formula specified in the **algorithm 1**:
$membershipValue = 1 - (|fdata - tupleValue|/Avg)$
for the 1st tuple : $membershipValue = 1 - (|50 - 25|/42.7) = 0.41$
for the 2nd tuple : $membershipValue = 1 - (|50 - 52|/42.7) = 0.95$
for the 3rd tuple: $membershipValue = 1 - (|50 - 38|/42.7) = 0.72$
for the 4th tuple: $membershipValue = 1 - (|50 - 48|/42.7) = 0.95$
for the 5th tuple: $membershipValue = 1 - (|50 - 34|/42.7) = 0.63$
for the 6th tuple: $membershipValue = 1 - (|50 - 30|/42.7) = 0.53$
for the 7th tuple: $membershipValue = 1 - (|50 - 50|/42.7) = 1$
for the 8th tuple: $membershipValue = 1 - (|50 - 51|/42.7) = 0.98$
for the 9th tuple: $membershipValue = 1 - (|50 - 45|/42.7) = 0.88$
for the 10th tuple: $membershipValue = 1 - (|50 - 54|/42.7) = 0.91$

The fuzzy representation of the EMP relation w.r.t. **uncertain query 1** is now depicted in Table 2. In particular, the fuzzy representation of the attribute ***Age*** appears in third column of the Table 2. Next in fourth column, we have shown the corresponding vague representation of these fuzzy data. These vague values are then used to find the similarity measures (**S.M.**) with fuzzy data **around 50** whose vague representation is $< 50, [1, 1] >$. The similarity measures have been calculated using the same formula as presented in **definition 3**. For example, consider the following two vague data:
$x =< 50, [1, 1] >$ and $y =< 25, [0.41, 0.41] >$. Here $t_x = 1$, $f_x = 0$, $t_y = 0.41$, $f_y = 0.59$.
Then, $SE(x, y) = \sqrt{(1 - (|(1 - 0.41) - (0 - 0.59)|/2))\,(1 - |(1 - 0.41) + (0 - 0.59)|)}$
$= \sqrt{(1 - 0.59)} = \sqrt{0.41} = 0.64$

Again, for the vague values $x =< 50, [1, 1] >$ and $y =< 52, [0.95, 0.95] >$, $t_x = 1$, $f_x = 0$, $t_y = 0.95$, $f_y = 0.05$.
Then, $SE(x, y) = \sqrt{(1 - (|(1 - 0.95) - (0 - 0.05)|/2))\,(1 - |(1 - 0.95) + (0 - 0.05)|)}$
$= \sqrt{(1 - 0.05)} = \sqrt{0.95} = 0.98$ and so on.
Using the notation: FD = Fuzzy Data, VD = Vague Data, we represent in Table 2:

Now, if the threshold value or α-cut given by the decision maker is 0.95, then the corresponding SQL statement of the uncertain query 1 is generated as below:
Select $*$ from EMP where S.M.(tuple)$\geq$0.95 which retrieves the following resultant tuples given

Table 2: Fuzzy Representation of EMP Relation w.r.t Uncertain Query 1

Name	Age	Fuzzy **Age** with FD **around 50**	Vague FD **Age**	S.M. with VD $< 50, [1,1] >$	Exp	Sal	S.M. (tuple)
Prof. Smith	25	$< 25, .41 >$	$< 25, [.41, .41] >$	.64	1	20000	.64
Prof. Ganguly	52	$< 52, .95 >$	$< 52, [.95, .95] >$	.98	25	55000	.98
Prof. Roy	38	$< 38, .72 >$	$< 38, [.72, .72] >$	.85	15	38000	.85
Prof. David	48	$< 48, .95 >$	$< 48, [.95, .95] >$	.98	23	53000	.98
Prof. Maity	34	$< 34, .63 >$	$< 34, [.63, .63] >$	.79	10	32000	.79
Prof. Das	30	$< 30, .53 >$	$< 30, [.53, .53] >$	.73	4	27000	.73
Prof. Ahuja	50	$< 50, 1 >$	$< 50, [1, 1] >$	1	26	55500	1
Prof. Sharma	51	$< 51, .98 >$	$< 51, [.98, .98] >$	.99	16	40000	.99
Prof. Kundu	45	$< 45, .88 >$	$< 45, [.88, .88] >$	.94	22	50000	.94
Prof. Dutta	54	$< 54, .91 >$	$< 54, [.91, .91] >$	.95	33	80000	.95

in Table 3 from the EMP database in Table 2.

Table 3: Resultant Relation of Uncertain Query 1 for Fuzzy Set at Threshold value α=0.95

Name	***Age***	***Exp***	***Sal***
Prof. Ganguly	52	25	55000
Prof. David	48	23	53000
Prof. Ahuja	50	26	55500
Prof. Sharma	51	16	40000
Prof. Dutta	54	33	80000

ii) **Solution with Vague Sets:** Next, we process the same *uncertain query 1* for vague sets. Here, vague attribute is ***Age*** and vague data is ***around 50***.

It is then necessary to represent all domain values of attribute ***Age*** into vague form whose truth membership values are calculated from the **algorithm 1** and false membership values are provided by the decision maker considering the restriction that sum of truth and false membership values ≤ 1. Similarity measures are then calculated using the same formula as used for fuzzy attributes.

Let us consider the two vague data $x =< 50, [1, 1] >$ and $y =< 25, [0.41, 0.5] >$. Here $t_x = 1$, $f_x = 0$, $t_y = 0.41$, $f_y = 0.5$.

Then, $SE(x, y) = \sqrt{(1 - (|(1 - 0.41) - (0 - 0.5)|/2))\,(1 - |(1 - 0.41) + (0 - 0.5)|)}$

$= \sqrt{(1 - 1.09/2) * (1 - 0.09)} = \sqrt{0.455 * 0.91} = \sqrt{0.41405} = 0.64$

Again, for $x =< 50, [1, 1] >$ and $y =< 52, [0.95, 0.98] >$, $t_x = 1$, $f_x = 0$, $t_y = 0.95$, $f_y = 0.02$.

Then, $SE(x, y) = \sqrt{(1 - (|(1 - 0.95) - (0 - 0.02)|/2))\,(1 - |(1 - 0.95) + (0 - 0.02)|)}$

$= \sqrt{(1 - 0.07/2)(1 - 0.03)} = \sqrt{0.93605} = 0.97$ and so on.

The Table 4 given below shows the complete vague representation of EMP relation w.r.t. the uncertain query 1.

with the same threshold or α-cut value 0.95, the following SQL statement of the uncertain query 1 for vague set will be generated:

Select $*$ from EMP where S.M.(tuple)$\geq$0.95

which now retrieves the following resultant tuples given in Table 5 from the EMP database for

Table 4: Vague Representation of EMP Relation w.r.t Uncertain Query 1

Name	Age	Vague representation of vague data ***Age***	S.M. with vague data $< 50, [1, 1] >$	Exp	Sal	S.M. (tuple)
Prof. Smith	25	$< 25, [.41, .5] >$	.64	1	20000	.64
Prof. Ganguly	52	$< 52, [.95, .98] >$	.97	25	55000	.97
Prof. Roy	38	$< 38, [.72, .8] >$	.84	15	38000	.84
Prof. David	48	$< 48, [.95, .98] >$	.97	23	53000	.97
Prof. Maity	34	$< 34, [.63, .75] >$	.78	10	32000	.78
Prof. Das	30	$< 30, [.53, .7] >$	.71	4	27000	.71
Prof. Ahuja	50	$< 50, [1, 1] >$	1	26	55500	1
Prof. Sharma	51	$< 51, [.98, 1] >$	.98	16	40000	.98
Prof. Kundu	45	$< 45, [.88, .93] >$	.93	22	50000	.93
Prof. Dutta	54	$< 54, [.91, .95] >$	.94	33	80000	.94

vague set in Table 4.

Table 5: Resultant Relation of Uncertain Query 1 for Vague Set at threshold value α=0.95

Name	***Age***	***Exp***	***Sal***
Prof. Ganguly	52	25	55000
Prof. David	48	23	53000
Prof. Ahuja	50	26	55500
Prof. Sharma	51	16	40000

It may be noted from Tables 3 and 5 that vague set gives better solution than fuzzy set since the SQL statement with vague query does not retrieve the tuple of Prof. Dutta with age 54 that has been fetched with the fuzzy query. It may be observed that 54 is less closer to 50 compared to the values of the attribute ***Age*** in all the other tuples retrieved by the SQL statement.

Next, we consider an uncertain query where more than one attribute are fuzzy or vague in nature.
Uncertain query 2: "Find the details of the Professors whose age is **around** 50 and experience is **more or less** 25".

i) Solution with Fuzzy Sets: **Uncertain query 2** has two fuzzy attributes, ***Age*** and ***Experience***. Applying **algorithm 1** and **definition 3** for calculating membership values and similarity measures respectively, we get the following fuzzy representation of EMP relation (FD = Fuzzy Data):

Here, μ_1 denotes the similarity measures of ***Age*** attribute with respect to FD ***Age*** around 50 [for detail calculation **see** Table 2],
μ_2 denotes the similarity measures with respect to fuzzy attribute ***Exp***,
and $\mu = \mu_1 \cap \mu_2$ denotes the similarity measures of tuples.

Then, the result is tested for different **threshold** or α-cut values given by the decision maker.

Table 6: Fuzzy Representation of EMP Relation w.r.t Uncertain Query 2

Name	Age	μ_1	Exp	Fuzzy **Exp** with FD **almost 25**	Vague FD **Exp**	μ_2	Sal	μ
Prof. Smith	25	.64	1	$< 1, .3 >$	$< 1, [.3, .3] >$	.56	20000	.56
Prof. Ganguly	52	.98	25	$< 25, 1 >$	$< 25, [1, 1] >$	1	55000	.98
Prof. Roy	38	.85	15	$< 15, .71 >$	$< 15, [.71, .71] >$	.84	38000	.84
Prof. David	48	.98	23	$< 23, .94 >$	$< 23, [.94, .94] >$	.97	53000	.97
Prof. Maity	34	.79	10	$< 10, .5 >$	$< 10, [.5, .5] >$	.71	32000	.71
Prof. Das	30	.73	4	$< 4, .4 >$	$< 4, [.4, .4] >$	.63	27000	.63
Prof. Ahuja	50	1	26	$< 26, .97 >$	$< 26, [.97, .97] >$	.99	55500	.99
Prof. Sharma	51	.99	16	$< 16, .74 >$	$< 16, [.74, .74] >$	.86	40000	.86
Prof. Kundu	45	.94	22	$< 22, .91 >$	$< 22, [.91, .91] >$	.95	50000	.94
Prof. Dutta	54	.95	33	$< 33, .77 >$	$< 33, [.77, .77] >$	.88	80000	.88

Case a) for α=0.95, the SQL statement is *Select* $*$ *from EMP where* $\alpha \geq 0.95$ which retrieves the resultant Table 7 from Table 6 as follows:

Table 7: Resultant Relation of Uncertain Query 2 for Fuzzy Set at threshold value α=0.95

Name	**Age**	**Exp**	**Sal**
Prof. Ganguly	52	25	55000
Prof. David	48	23	53000
Prof. Ahuja	50	26	55500

Case b) for α=0.87, the SQL statement is *Select* $*$ *from EMP where* $\alpha \geq 0.87$ and the resultant table is shown below in Table 8:

Table 8: Resultant Relation of Uncertain Query 2 for Fuzzy Set at threshold value α=0.87

Name	**Age**	**Exp**	**Sal**
Prof. Ganguly	52	25	55000
Prof. David	48	23	53000
Prof. Ahuja	50	26	55500
Prof. Kundu	45	22	50000
Prof. Dutta	54	33	80000

ii) Solution with Vague Sets:
Again, using **algorithm 1** and **definition 3**, the vague representation of EMP relation for uncertain query 2 may be obtained as follows (VD= Vague Data):

The result is now tested for vague set with the same threshold or α-cut values.

Case a) for α=0.95, the SQL statement is *Select* $*$ *from EMP where* $\mu \geq 0.95$ which retrieves from Table 9 the following resultant table as

Case b) for α=0.87, SQL statement is *Select* $*$ *from EMP where* $\mu \geq 0.87$ and the resultant table is

From Tables 7 and 10 it may be observed that the resultant sets of the uncertain query 2 for both fuzzy data and vague data are same for the threshold value α=0.95.

However, when the same query is tested with α-cut value 0.87, Tables 8 and 11 show that the vague sets certainly gives better result than fuzzy sets because vague SQL has not retrieved the tuple Prof. Dutta with age 54 and experience 33 which is not so closer to age 50 and experience 25.

Table 9: Vague Representation of EMP Relation w.r.t Uncertain Query 2

Name	Age	Vague **Age** with VD **around 50**	μ_1	Exp	Vague **Exp** with VD **almost 25**	μ_2	Sal	μ
Prof. Smith	25	$< 25, [.41, .5] >$	.64	1	$< 1, [.3, .4] >$	.56	20000	.56
Prof. Ganguly	52	$< 52, [.95, .98] >$	.97	25	$< 25, [1, 1] >$	1	55000	.97
Prof. Roy	38	$< 38, [.72, .8] >$	.84	15	$< 15, [.71, .8] >$	.83	38000	.83
Prof. David	48	$< 48, [.95, .98] >$	.97	23	$< 23, [.94, .98] >$	.96	53000	.96
Prof. Maity	34	$< 34, [.63, .75] >$	.78	10	$< 10, [.5, .6] >$	.7	32000	.7
Prof. Das	30	$< 30, [.53, .7] >$	.71	4	$< 4, [.4, .43] >$	.63	27000	.63
Prof. Ahuja	50	$< 50, [1, 1] >$	1	26	$< 26, [.97, 1] >$	.98	55500	.98
Prof. Sharma	51	$< 51, [.98, 1] >$	.98	16	$< 16, [.74, .82] >$	.85	40000	.85
Prof. Kundu	45	$< 45, [.88, .93] >$	.93	22	$< 22, [.91, .96] >$	.94	50000	.94
Prof. Dutta	54	$< 54, [.91, .95] >$	.94	33	$< 33, [.77, .85] >$	.86	80000	.88

Table 10: Resultant Relation of Uncertain Query 2 for Vague Set at threshold value α=0.95

Name	***Age***	***Exp***	***Sal***
Prof. Ganguly	52	25	55000
Prof. David	48	23	53000
Prof. Ahuja	50	26	55500

Table 11: Resultant Relation of Uncertain Query 2 for Vague Set at threshold value α=0.87

Name	***Age***	***Exp***	***Sal***
Prof. Ganguly	52	25	55000
Prof. David	48	23	53000
Prof. Ahuja	50	26	55500
Prof. Kundu	45	22	50000

6 Conclusions

In this paper, we have proposed an architecture to process uncertain queries represented by fuzzy or vague data. We have also presented an algorithm that generates the fuzzy or vague representation of the attributes with respect to the given uncertain query. Similarity measure presented in definition 3 is used to find similarity measure of tuples w.r.t the given uncertain query in fuzzy or vague representation. Then the proposed architecture has been verified for uncertain queries using a real life example, both for the fuzzy as well as vague representation. In each case it has been observed that vague sets have produced more accurate result in comparison to fuzzy sets. Hence a vague relational database model may be more fruitful in processing real life data and queries than the conventional fuzzy data models. A DBMS that implements this vague set theoretic concept can thus become a more powerful software product than those currently available.

Bibliography

[1] Codd E. F. (1970); A Relational Model for Large Shared Data Banks, *Comm. of ACM*, 13(6): 377-387.

[2] Codd E. F. (1990); *The Relational Model for Database Management*, Addison Wesley.

[3] Date C. J. (2004); *An Introduction to Data Base Systems*, 8th ed., Addison Wesley.

[4] Elmasri R.; Navathe S. B. (2010); *Fundamentals of Database Systems*, 6th ed., Pearson.

[5] Zadeh L. A. (1965); Fuzzy Sets, *Information and Control*, 8(3): 338-353.

[6] Buckles P. B.; Petry F. E. (1982); A Fuzzy Representation of Data For Relational Databases, *Fuzzy Sets and Systems*, 7(3): 213-226.

[7] Raju K.V.S.V.N.; Majumdar A.K. (1988); Fuzzy functional dependencies and lossless join decomposition of fuzzy relational database system, *ACM Transactions on Database Systems*, 13(2): 129-166.

[8] Ma Z. M.; Mili F. (2002); Handling fuzzy information in extended possibility-based fuzzy relational databases, *International Journal of Intelligent Systems*, 17(10): 925-942.

[9] Intan R.; Mukaidono M. (2000); Fuzzy functional dependency and its application to approximate data querying, *Proc. of international Database Engineering and Applications Symposium*, 47-54.

[10] Takahashi Y. (1993); Fuzzy database query languages and their relational completeness theorem, *IEEE Transactions on Knowledge and Data Engineering*, 5: 122-125.

[11] Bosc P.; Pivert O. (1995); SQLF: A relational database language for fuzzy querying, *IEEE Transaction on Fuzzy Systems*, 3(1): 1-17.

[12] Nakajima H. et al. (1993); Fuzzy Database Language and Library- Fuzzy Extension to SQL, *Second IEEE International Conference on Fuzzy Systems*, 1: 477-482.

[13] Gau W. L.; D. J. Buehrer (1993); Vague Sets, *IEEE Trans. Syst. Man, Cybernetics*, 23(2): 610-614.

[14] Lu A.; Ng W. (2005); Vague Sets or Intuitionistic Fuzzy Sets for Handling Vague Data: Which one is better?, *Lecture Notes in Computer Science*, 3716: 401-416.

[15] Zhao F.; Ma Z. M. (2009); Vague Query Based on Vague Relational Model, *AISC, Springer-Verlag Berlin Heidelberg*, 61: 229-238.

[16] Chen S. M. (1997); Similarity Measure between Vague Sets and between Elements, *IEEE Trans. Systems. Man and Cybernetics*, 27(1): 153-158.

[17] Hong D. H.; Kim C. (1999); A Note on Similarity Measures between Vague Sets and between Elements, *Information Sciences*, 115: 83-96.

[18] Li F.; Xu Z. (2001); Measures of Similarity between Vague Sets, *Journal of Software*, 12(6): 922-927.

[19] Lu A.; Ng W. (2004); Managing Merged Data by Vague Functional Dependencies, *LNCS, Springer-Verlag Berlin Heidelberg*, 3288: 259-272.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):741-748, December, 2014.

PARMODS: A Parallel Framework for MODS Metaheuristics

E.D. Nino Ruiz, S. Miranda, C.J. Ardila, W. Nieto

Elias D. Nino Ruiz*, Stella Miranda,
Carlos J. Ardila, Wilson Nieto
Universidad del Norte
Computer Science Department
Colombia, Barranquilla
{enino,stellam,carila,wnieto}@uninorte.edu.co
*Corresponding author: enino@uninorte.edu.co

Abstract: In this paper, we propose a novel framework for the parallel solution of combinatorial problems based on MODS theory (PARMODS) This framework makes use of metaheuristics based on the Deterministic Swapping (MODS) theory. These approaches represents the feasible solution space of any combinatorial problem through a Deterministic Finite Automata. Some of those methods are the Metaheuristic Of Deterministic Swapping (MODS), the Simulated Annealing Deterministic Swapping (SAMODS), the Simulated Annealing Genetic Swapping (SAGAMODS) and the Evolutionary Deterministic Swapping (EMODS) Those approaches have been utilized in different contexts such as data base optimization, operational research [1–3, 8] and multi-objective optimization. The main idea of this framework is to exploit parallel computation in order to obtain a general view of the feasible solution space of any combinatorial optimization problem. This is, all the MODS methods are used in a unique general optimization process. In parallel, each instance of MODS explores a different region of the solution space. This allows us to explore distant regions of the feasible solution which could not be explored making use of classical (sequential) MODS implementations. Some experiments are performed making use of well-known TSP instances. Partial results shows that PARMODS provides better solutions than sequential MODS based implementations.
Keywords: MODS, Combinatorial Optimization, Parallel Framework.

1 Introduction

Combinatorial Optimization (CO) is a branch of optimization in which problems can be represented (or reduced) to discrete structures. In this ramification, we find many problems related to operational research and networking fields. Moreover, since the number of possible solutions in this kind of problems increase exponentially with regard to the input parameters, their numerical solution can be very hard (or impossible) to obtain, for instance solving their mathematical formulations. Thus, two important considerations should be taken into in account when we want to solve CO problems: the number of solutions to consider is only a subset of the feasible solution space and the solutions should be obtained in a polynomial time. The first item addresses the necessity of having good solutions and the second one, demands the elapsed time to be small for the proposed implementation. However, those features are opposite, this means, when the number of solutions explored from the feasible solution space is small, the solution is obtained in short time but maybe, the approximated optimal solutions are not good enough. On the other hand, exploring more and more the feasible solution space provides better approximations to the optimal solution but, the performance of the method is affected considerably (long elapsed times). Thus, we need methods which in a polynomial time consider more and more solutions from the feasible solution space. Notice, we are not taking about exhaustive methods such as brute force but, combining information from different metaheuristics in a polynomial time which can be done in parallel.

This paper is organized as follows: in section 2 the TSP problem is introduced and MODS metaheuristics are presented, section 3 describes the proposed implementation and, section 4 and 5 provide the experimental results and conclusions, respectively.

2 Preliminaries

Following the previous section, one of the most widely used CO problems is the Traveling Salesman Probem (TSP) Its importance is derived owing to its application to different branch and fields from optimization. Moreover, some well-known problems such as the Vehicle Routing Problem and the Transportation Problem are derived from the TSP formulation. In general, this problem is defined as follows: we have a set of N cities $\mathcal{C} = \{c_1, c_2, \ldots, c_N\}$, a matrix of weights $\mathbf{W} \in \mathbb{R}^{N \times N}$ whose elements $w_{i,j}$ provides the weight of going from c_i to c_j, for $1 \leq i, j \leq N$ and, the cost function

$$\mathcal{J}(\boldsymbol{\alpha}) = w_{N,\alpha_1} + \sum_{i=1}^{N-1} w_{\alpha_i,\alpha_{i+1}}, \tag{1}$$

which is subjected to

- Visiting each city from $\mathcal{C}$ once.
- Coming back to the initial city once the path $\boldsymbol{\alpha}$ has been completed.

Rudely speaking, we want to find the optimal path $\boldsymbol{\alpha}^*$ which provides the optimal components $w_{\alpha_i^*,\alpha_{i+1}^*}$, for $1 \leq i \leq N-1$, from $\mathbf{W}$ such that (1) is minimized. Note that, the number of possible solutions for the TSP problem increases by $N!$ with regard to the number of cities N.

As we mentioned before, the numerical solution of CO problems can be an exhaustive work making use of numerical methods and the TSP problem is not the exception. Note that, the TSP problem can be seen as a linear programming problem therefore well-known numerical methods based on Integer Programming and Simplex Methods could be used in order to solve (1) but, they have been proved to fail when the number of cities is large, as is usual in practice. On the other hand, the optimal solution of the TSP problem can be approximated making use of metaheuristics, we address one set of them in this paper, those are based on the Metaheursitic Of Deterministic Swapping (MODS) MODS is a metaheuristic inspired on the Automata Theory. Its application ranges from the Operational Research field to the Database Query Optimization area [7]. It is very important to note that, MODS methods are not novel methods, in general, they are nothing but classical combinatorial optimization methods represented on Deterministic Finite Automata structures. This improves the manner to design the solution of the problem since the optimial solution space and the transition between solutions (states of the automata) are defined prior any optimization process. This avoids, for instance, to explore unfeasible regions of the solution space.

MODS considers the next Deterministic Finite Automata (DFA)

$$\mathcal{Q}_{MODS} = \{\mathcal{S}, \Sigma, \delta, \mathcal{S}_0, \mathcal{J}\}\ , \tag{2}$$

where $\mathcal{S}$ is the feasible solution space, Σ is the input alphabet which is utilized by $\delta : \mathcal{S} \to \mathcal{S}$ in order to perturb the solutions, $\mathcal{S}_0$ contains the initial solutions and, $\mathcal{J}$ is the cost function to be optimized. The $\mathcal{S}$ space is unknown since it contains all the possible solutions of the CO problem. Putting all in the TSP context, $\mathcal{S}$ contains all the possible paths, $\mathcal{S}_0$ provides the initial path, $\mathcal{J}$ is the cost function (1) and Σ and δ provides all the possible manners to perturb

a given path, for instance, given the path $\boldsymbol{\alpha}' = [1, 2, 3, 4]$ and the duple $\sigma_1 = (2, 3) \in \Sigma$ then, $\delta(\boldsymbol{\alpha}', \sigma_1) = [1, 3, 2, 4]$. The MODS metaheuristic is defined in Algorithm 2.

Algorithm 2 MODS Metaheuristic

Require: $\Sigma, \delta, \mathcal{S}_0, \mathcal{J}$
Ensure: $\boldsymbol{\alpha}^+ \approx \boldsymbol{\alpha}^*$
1: $\boldsymbol{\alpha}^+ \leftarrow s \in \mathcal{S}_0$
2: **for** $k = 1 \rightarrow M$ **do**
3: $\sigma_k \leftarrow \sigma \in \Sigma$
4: $\boldsymbol{\alpha}^- \leftarrow \delta(\boldsymbol{\alpha}^+, \sigma_k)$
5: **if** $\mathcal{J}(\boldsymbol{\alpha}^-) < \mathcal{J}(\boldsymbol{\alpha}^+)$ **then**
6: $\boldsymbol{\alpha}^+ \leftarrow \boldsymbol{\alpha}^-$
7: **end if**
8: **end for**

Algorithm 3 SAMODS Metaheuristic

Require: $\Sigma, \delta, \mathcal{S}_0, \mathcal{J}, T_0, \rho, L$
Ensure: $\boldsymbol{\alpha}^+ \approx \boldsymbol{\alpha}^*$
1: $\boldsymbol{\alpha}^+ \leftarrow s \in \mathcal{S}_0$
2: **for** $k = 1 \rightarrow M$ **do**
3: **for** $i = 1 \rightarrow L$ **do**
4: $\sigma_i \leftarrow \sigma \in \Sigma$
5: $\boldsymbol{\alpha}^- \leftarrow \delta(\boldsymbol{\alpha}^+, \sigma_i)$
6: **if** $\mathcal{J}(\boldsymbol{\alpha}^-) < \mathcal{J}(\boldsymbol{\alpha}^+)$ **then**
7: $\boldsymbol{\alpha}^+ \leftarrow \boldsymbol{\alpha}^-$
8: **else**
9: Generate (uniformly) $\eta \in [0, 1]$
10: Compute

$$\gamma = \exp\left(-\frac{\mathcal{J}(\boldsymbol{\alpha}^+) - \mathcal{J}(\boldsymbol{\alpha}^-)}{T_k}\right) \tag{3}$$

11: **if** $\eta < \gamma$ **then**
12: $\boldsymbol{\alpha}^+ \leftarrow \boldsymbol{\alpha}^-$
13: **end if**
14: **end if**
15: **end for**
16: $T_{k+1} \leftarrow \rho \cdot T_k$
17: **end for**

SAMODS is a Simulated-Annealing (SA) based MODS method which explores the feasible solution space $\mathcal{S}$ in a more "generous" manner. It allows bad solution to be accepted in small optimization intervals (usually at the beginning of the iterations) Alike MODS, SAMODS makes use of the DFA

$$\mathcal{Q}_{SAMODS} = \{\mathcal{S}, \Sigma, \delta, \mathcal{S}_0, \mathcal{J}, T_0, \rho, L\} \ , \tag{4}$$

where $\mathcal{S}$, Σ, δ, $\mathcal{S}_0$ and, $\mathcal{J}$ remain unchanged, T_0 is the initial temperature, ρ is the cooling factor and, L is the number of refinement iterations. Note that, MODS accepts a new solution

only when its optimal value is better than the current value (from the current path) On the other hand, SAMODS makes use of the Boltzamm distribution (5) in order to give the chance of a bad solution to be improved. This may provides a better solution than the best solution considered so far. Thus, at the beginning of the iterations, the number of solutions accepted as good is large but, this number is decreased when the iterations draws on since the parameter T_k is large and then, the condition of line 11 in Algorithm 3 is almost never satisfied. Following the SA principles, SAGAMODS [5] is defined on the SAMODS method but, when a bad solution is rejected (line 11 in Algorithm 3), the solution is improved making use of Genetic Algorithms. The supporting automata of SAGAMODS method is defined as follows:

$$\mathcal{Q}_{SAGAMODS} = \{\mathcal{S}, \mathcal{S}_0, \mathcal{C}(s, r, k), F(s)\}$$

$\mathcal{S}$ and $\mathcal{S}_0$ remain unchanged from the previous methods. In addition, $\mathcal{C}(s_1, s_2, k)$ is the crossover operator where $s_1 \in \mathcal{S}$ and $s_2 \in \mathcal{S}$ are parents solutions. Likewise, k provides the cross point. SAGAMODS method is presented in the Algorithm 4.

Algorithm 4 SAGAMODS Metaheuristic

Require: $\Sigma, \delta, \mathcal{S}_0, \mathcal{J}, T_0, \rho, L$
Ensure: $\boldsymbol{\alpha}^+ \approx \boldsymbol{\alpha}^*$
1: $\boldsymbol{\alpha}^+ \leftarrow s \in \mathcal{S}_0$
2: **for** $k = 1 \rightarrow M$ **do**
3: **for** $i = 1 \rightarrow L$ **do**
4: $\sigma_i \leftarrow \sigma \in \Sigma$
5: $\boldsymbol{\alpha}^- \leftarrow \delta(\boldsymbol{\alpha}^+, \sigma_i)$
6: **if** $\mathcal{J}(\boldsymbol{\alpha}^-) < \mathcal{J}(\boldsymbol{\alpha}^+)$ **then**
7: $\boldsymbol{\alpha}^+ \leftarrow \boldsymbol{\alpha}^-$
8: **else**
9: Generate (uniformly distributed) $\eta \in [0, 1]$
10: Compute

$$\gamma = \exp\left(-\frac{\mathcal{J}(\boldsymbol{\alpha}^+) - \mathcal{J}(\boldsymbol{\alpha}^-)}{T_k}\right) \tag{5}$$

11: **if** $\eta < \gamma$ **then**
12: $\boldsymbol{\alpha}^+ \leftarrow \boldsymbol{\alpha}^-$
13: **else**
14: Generate integer number (uniformly distributed) $\beta \in [1, modelsize]$
15: **call** $\mathcal{C}(\boldsymbol{\alpha}^+, \boldsymbol{\alpha}^-, \beta)$
16: **end if**
17: **end if**
18: **end for**
19: $T_{k+1} \leftarrow \rho \cdot T_k$
20: **end for**

EMODS [6] is an evolutionary MODS method which improves the solutions making use of evolutionary techniques (crossover and mutation). A complete taxonomy of SAGAMODS and EMODS methods can be read in [4, Chapter 4].

Now we are ready to present our parallel approach of MODS based methods.

3 Proposed Implementation

To start, consider an array of processors available at time t:

$$\mathbf{P} = [p_1, p_2, \ldots, p_n], \tag{6}$$

where n is the number of processors. For simplicity, we avoid the use of time indexes. Moreover, we consider that the metaheuristics MODS, SAMODS, SAGAMODS and EMODS can be run independently at different processors. Thus, we want to split the number of available processors per the number of metaheuristics, this is:

$$jobs_{proc} = \frac{n}{4}. \tag{7}$$

Consider the initial solution $s \in S_0$, then we denote the next DFAs based on the index $1 \leq i \leq n$:

$$\mathcal{Q}_i = \begin{cases} \mathcal{Q}_1 = \mathcal{Q}_{MODS} = \{\ldots, s\} & \text{for } i = 1, 5, \ldots \\ \mathcal{Q}_2 = \mathcal{Q}_{SAMODS} = \{\ldots, s\} & \text{for } i = 2, 6, \ldots \\ \mathcal{Q}_3 = \mathcal{Q}_{SAGAMODS} = \{\ldots, s\} & \text{for } i = 3, 7, \ldots \\ \mathcal{Q}_4 = \mathcal{Q}_{EMODS} = \{\ldots, s\} & \text{for } i = 4, 8, \ldots \end{cases}, \tag{8}$$

and then, we are ready to launch different processes based on the next rule

$$job_i = \begin{cases} \mathcal{P}(\mathcal{Q}_1, s) & \text{for } i = 1, 5, \ldots \\ \mathcal{P}(\mathcal{Q}_2, s) & \text{for } i = 2, 6, \ldots \\ \mathcal{P}(\mathcal{Q}_3, s) & \text{for } i = 3, 7, \ldots \\ \mathcal{P}(\mathcal{Q}_4, s) & \text{for } i = 4, 8, \ldots \end{cases}, \tag{9}$$

where the i-th process job_i ($\mathcal{P}(.,.)$) is executed in the processor p_i of (6), for $1 \leq i \leq n$. Note that, in (8) we choose the automata to be utilized and in (9), we launch the process. For instance, MODS metaheuristic is executed on processors 1,4,..., likewise, SAMODS is executed on processors 2,5,... and so on.

Denote by s_1, s_2, s_3, and s_4 the approximated optimal solutions provided by MODS, SAMODS, SAGAMODS and EMODS among processors, respectively, this is

$$\begin{aligned} s_1 &= \underset{s^{(i)}_{MODS}}{\arg\min} \left\{ \mathcal{J}\left(s^{(i)}_{MODS}\right), \text{ for } i = 1, 4, \ldots \right\} \\ s_2 &= \underset{s^{(i)}_{SAMODS}}{\arg\min} \left\{ \mathcal{J}\left(s^{(i)}_{MODS}\right), \text{ for } i = 2, 5, \ldots \right\} \\ s_3 &= \underset{s^{(i)}_{SAGAMODS}}{\arg\min} \left\{ \mathcal{J}\left(s^{(i)}_{MODS}\right), \text{ for } i = 3, 6, \ldots \right\} \\ s_4 &= \underset{s^{(i)}_{EMODS}}{\arg\min} \left\{ \mathcal{J}\left(s^{(i)}_{MODS}\right), \text{ for } i = 4, 7, \ldots \right\} \end{aligned}$$

where, for instance, $s^{(1)}_1$ is the approximated optimal solution of MODS from the processor 1. Then, we choose the best approximation,

$$s^+ = \underset{s_i}{\arg\min} \left\{ \mathcal{J}(s_k), \text{ for } 1 \leq k \leq 4 \right\} \tag{10}$$

which will serve as the new initial solution in $\mathcal{S}_0$. This iterative process is called PARMODS (Parallel MODS) and it is summarized in Algorithm 5. Note that, the required components of this metaheuristic varies from the definitions of the automatas, that is why the common components are shown in the inputs and the optional parameters are expressed by dots.

Algorithm 5 PARMODS Metaheuristic

Require: $\Sigma, \delta, \mathcal{S}_0, \mathcal{J}, \ldots$
Ensure: $\boldsymbol{\alpha}^+ \approx \boldsymbol{\alpha}^*$
1: $\boldsymbol{\alpha}^+ \leftarrow \mathcal{S}_0$
2: **for** $t = 1 \rightarrow M$ **do**
3: **for all** $i = 1 \rightarrow n$ **do**
4: Launch job_i according to (9).
5: **end for**
6:

$$\boldsymbol{\alpha}^- = \underset{s^{(j)}_{*MODS}}{\arg\min} \left\{ \mathcal{J}\left(s^{(j)}_{*MODS}\right) \text{ , for } 1 \leq j \leq n \right\}$$

7: **if** $\mathcal{J}(\boldsymbol{\alpha}^-) < \mathcal{J}(\boldsymbol{\alpha}^+)$ **then**
8: $\boldsymbol{\alpha}^+ \leftarrow \boldsymbol{\alpha}^-$
9: **end if**
10: **end for**

Notice, the computational cost of the method per iteration will be given by the number of iterations of PARMODS times the upper bound

$$\mathcal{O}\left(A_{PARMODS}\right) = \max\left(\mathcal{O}(A_{MODS}), \mathcal{O}(A_{SAMODS}), \mathcal{O}(A_{SAGAMODS}), \mathcal{O}(A_{EMODS})\right) \text{ ,}$$

where the letter A counts for “Algorithm". Note that, since all the methods are executed in parallel, the computational effort of PARMODS is provided by the largest upper bound, which, in general, is provided by SAGAMODS.

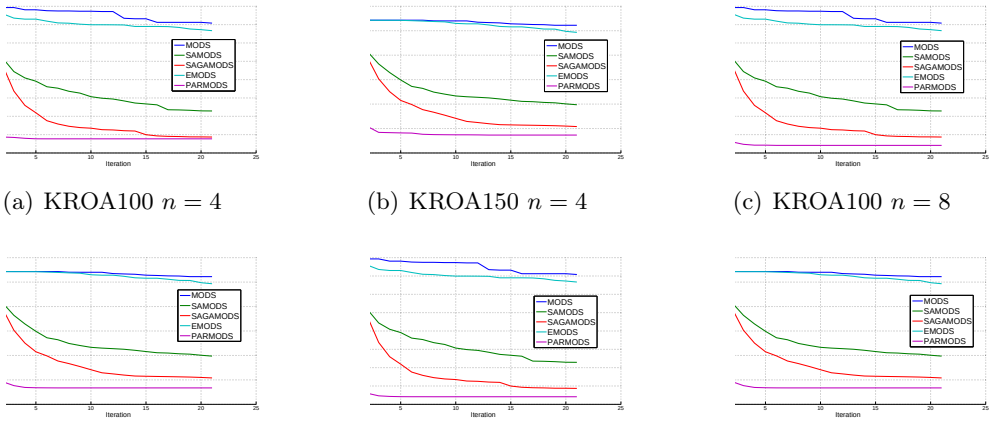
4 Experimental Results

We study the performance and efficiency of PARMODS making use of TSP instances from the TSPLIB. The selected TSP instances are KROA100 and KROA150 which contain 100 and 150 cities, respectively. The solutions obtained by the methods are presented in Table 1 and figure 1.

Metaheuristic	Processors	$\mathcal{J}$ KROA100	$\mathcal{J}$ KROA150
MODS	N/A	1.6165	2.6122
SAMODS	N/A	0.6583	0.9865
SAGAMODS	N/A	0.3739	0.5399
EMODS	N/A	1.5340	2.4663
PARMODS	4	0.3545	0.3664
PARMODS	8	0.2827	0.3359
PARMODS	12	0.2827	0.3359

Table 1: Cost function values $\times 10^5$ for different MODS implementations.

In figure 1 can be seen how PARMODS outperforms the other MODS implementations (sequential MODS, SAMODS, SAGAMODS and EMODS) in terms of accuracy. Moreover, the MODS implementations are divided evenly onto the number of processors available (n) For instance, four processors means one instance of MODS, SAMODS, SAGAMODS and EMODS are used at each processor when PARMODS is executed. Notice, PARMODS do not make use of parallel resources in order to split the domain but to obtain information about the feasible solution space. Since PARMODS spread MODS instances among processors, the best solution is used in the next generation of each MODS implementation (initial state of each Automata).



(a) KROA100 $n = 4$ (b) KROA150 $n = 4$ (c) KROA100 $n = 8$

(d) KROA150 $n = 8$ (e) KROA100 $n = 12$ (f) KROA150 $n = 12$

Figure 1: Graphical comparison of the cost function values per iteration for the different MODS implementations.

5 Conclusions

We propose a novel parallel method based on MODS theor5522894y. The proposed implementation exploits the attractive features of each MODS implementation. Initial results show that PARMODS provides better results among the compared methods. Moreover, when the number of processors is increased, the results are improved. However, we note that the results obtained for 8 and 12 processors are the same. This motivates to study theoretical bounds regarding the number of processors and the percentage of improvement on the solutions.

Bibliography

[1] Anonnimus (1964); Operational research studies in inventory sequencing simulation, *Production Engineer*, 43(9):437–438, DOI: 10.1049/tpe.1964.0060.

[2] Anonnimus (1964); Operational research studies. project a-inventory. *Production Engineer*, 43(9):438–447, DOI: 10.1049/tpe:19640061.

[3] Junyi Chen and Pingyuan Xi (2010); Simulation and application on modern operational research. In *Computer and Automation Engineering (ICCAE), 2010 The 2nd International Conference on*, 4: 118–121.

[4] Elias D. Niño (2012); *Real-World Applications of Genetic Algorithms*, chapter Evolutionary Algorithms Based on the Automata Theory for the Multi-Objective Optimization of Combinatorial Problems. InTech, Oxford, 2012. Book edited by Olympia Roeva.

[5] Elias D. Nino, Carlos J. Ardila, and Anangelica Chinchilla (2012); A novel, evolutionary, simulated annealing inspired algorithm for the multi-objective optimization of combinatorial problems. *Procedia Computer Science*, 9(0):1992 – 1998.

[6] Elias D. Nino-Ruiz (2012); Evolutionary Algorithm based on the Automata Theory for the Multi-objective Optimization of Combinatorial Problems. *International Journal Of Computers Communication & Control*, 7(5):916–923.

[7] Miguel Rodríguez, Daladier Jabba, Elias D. Niño, Carlos J. Ardila, and Yi-Cheng Tu (2013); Automata theory based approach to the join ordering problem in relational database systems. In Markus Helfert, Chiara Francalanci, and Joaquim Filipe, editors, *DATA*, pages 257–265. SciTePress.

[8] Li Zhengfeng and Ye Jinfu (2010); Study on the evolutionary mechanism from operational research activities to sustainable competitive advantage. In *Intelligent Computation Technology and Automation (ICICTA), 2010 International Conference on*, 3: 580–584.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):749-757, December, 2014.

An Online Load Balancing Algorithm for a Hierarchical Ring Topology

C.I. Paduraru

Ciprian I. Paduraru
Computer Science Department
University of Bucharest
ciprian.paduraru2009@gmail.com

Abstract: Ring networks are an important topic to study because they have certain advantages over their direct network counterparts: easier to manage, better bandwidth, cheaper and wider communication paths. This paper proposes a new online load balancing algorithm for distributed real-time systems having a hierarchical ring as topology. The novelty of the algorithm lies in the goal it tries to achieve and the method used for load balancing. The main goal of the algorithm is to correctly utilize the computing resources in order to satisfy the average response time of clients. The secondary goal is to ensure fairness between the numbers of requests solved per client with respect to the average response time. A request from a client is moving through the network until a node considers that it can solve the request in the promised average time for that client or until it seems like the best opportunity to avoid any additional delays in solving it. A performance analysis and motivation for the proposed algorithm is given with respect to the goals it tries to achieve. The results show that the proposed algorithm satisfies its goals.
Keywords: ring; hierarchical; distributed; balancing; algorithm; fairness

1 Introduction

Today, the common approach for processing user requests sent to a web or network-based service is to handle them using a distributed architecture of computers. In this context, the performance of the processing system is closely related to user experience and service availability, and can therefore play an important role in the success or failure of the respective service on the market. As sufficient hardware resources for processing a large number of requests are generally expensive, a good algorithm for the distribution of load - between the processing units in the distributed system - is necessary to save costs in addition to increase client's satisfaction.
This paper presents a load balancing algorithm for hierarchical ring network. A hierarchical ring (Figure 1) is an alternative to 2D meshes or tori [5]. Hierarchical type was chosen to show the generality of the algorithm. Instead, we can have rings combined with other network topologies. In a ring network every node has exactly two neighbors: P_i is connected P_{i+1} to and P_{i-1}. In this paper, if we consider that n is the number of processors in the ring, then we assume that all additions on processors indices are done modulo n. In the hierarchical ring network considered, each sub-network has a leader which is responsible to store information and coordinate some activities. In the continuation, when we refer to a sub-network of a leader node then this includes only the direct nodes under the leader's level.

Requests are received by a web service which coincide with the leader node of the network - and are considered to have an estimated average time to complete. The goals of the presented algorithm are the most significant for services provided in the present. The main goal is to ensure a certain average response time given for each client, depending on what we call a user license. The user license can be interpreted as a contract between the service provider and the user, where parameters referring to the delivery of the service are specified. These include parameters

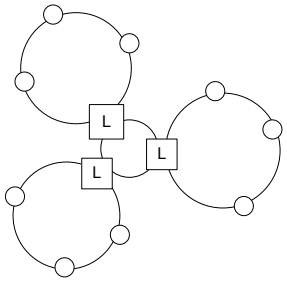


Figure 1: Two levels ring hierarchy.

relevant for load distribution, like the average response time of the system for certain types of requests. The secondary goal is to ensure fairness between the numbers of requests solved per client with respect to the average response time. A simulator has been created to demonstrate how the algorithm succeeds to satisfy the desired goals.
The rest of the paper is organized as follows: In Section 2 there is a discussion about research made on load balancing for ring topologies or other network types but appropriate to our goals. In Section 3 the design of the load balancing algorithm is discussed. If first starts with the assumptions made over the proposed algorithm then it describes the main ideas and pseudocode behind the decision making process. Section 4 shows the simulation results compared with a general load balancing for a hierarchical ring network . Conclusions are given in the last section.

2 Related work

A description of hierarchical ring networks is given in [5]. They are presented as an interesting alternative to popular direct networks such as 2D meshes or tori. Advantages of using them are also described here: simple router designs, wider communications paths and faster networks than their direct network counterparts. However the paper is not dealing with load balancing algorithms. Its a study to determine how large hierarchical ring networks can become before their performance deteriorates due to their bisection bandwidth constraints.
There are not many papers discussing about real time load balancing for ring networks. The most appropriate paper for our presentation is [1]. In comparison with [1], which is a general load balancer for rings, the new proposed load balancing algorithm has another two goals: satisfy the average response time specified in the owners license type and ensure some fairness between the requests with respect to their specified response times. Other papers, like [6], are performing a statical load balancing of requests on a ring network. Paper [2] presents a load balancing algorithm for distributed systems having the same two goals. However, the algorithms presented there are inapplicable to the ring networks. It uses the fact that nodes can communicate directly with a master and it would be too much overhead to simulate the same implementation algorithm on a ring. Both the requests and results are exchanged directly from master to workers.

3 Design of the algorithm

3.1 Assumptions

It is assumed that when a node finishes a request, the results are sent back to clients directly from that node. The algorithm allows for the system to be heterogeneous, workstations may differ in processing capacity. The processing time of a request is expressed as the "request's length" and can be predetermined. We assume that **GetEstTimeToCompute(request)** returns the

estimated processing time of a request on any node and its time complexity is constant. One simple way to do this is to benchmark how fast each node can execute different requests length intervals, then group and store these results in a data structure on the node. It is considered that request processing is workstation independent (all types of requests can be processed by any of the nodes). Requests are independent (the order in which requests are processed does not affect the correctness of the result) and indivisible (can only be handled by a single worker at one moment). The communication time is not generally important for the algorithm. The reason is that while a request spends time moving through the network its priority increases.

3.2 High level implementation and the communication protocol

The main responsibilities of nodes are to take decisions, solve requests and communicate with neighbors. The communication and request's solving should run in different threads to avoid communication blocking. Requests are received by the leader of the ring and send further until a node can execute it in the required time or when that node is a good opportunity to save additional delay in response time. Nodes evaluates if a request can be executed by them or not depending on the time needed to complete all other requests waiting there and having a higher priority than the considered request. Also, in the case of requests that are close to their deadline (or already passed deadline), if they have a higher priority than all other requests waiting on a node then we choose that node to minimize the additional delays in the response time.
By using the above two conditions there is a possibility that all nodes to decline solving a new request. In this case measures need to taken in order to avoid affecting the performance of the system with the new request running too many times through the ring. The method used is to have a variable on each request that represents a bonus time considered when a node evaluates if it has enough time to execute the request. Each time the request goes back through the node that initiated it, the leader of the ring, this variable is incremented by some value determining the nodes to accept it faster.
In the continuation of this section these ideas are presented in more details. It starts with the high level operations and messages exchanged between nodes, then it continues to explain the implementation of data structure and decision making in more details using pseudocode and complexity analysis.

To send data between nodes, two functions are used: **Send**(data) used to send data to the next node on the same sub-network and **SendToSubNetwork(data)** which sends a message from a leader to its coordinated sub-network (this helps moving a message from a higher level network to a sub-network). Another important function is **IsLeaderNode** which has two prototypes. The first one doesn't have any parameters - tests if the node is the leader of a sub-ring - and the second one with a parameter representing a message - tests if the node is a leader and the one who created/added that message in its sub-network.
There are two types of messages used in the communication protocol: **Gather** and **Request**. A Request message is used for sending requests between nodes and contains the following: data context for request execution, the average response time specified in the owners license, timestamp when created, the time when should ideally finish and the current bonus time. The code below shows the high level implementation of the decision making when a Request message is received by a node. A node that is not the leader at the level where a request message is sent can either store the request for later execution or send it further in the same network level. Additionally, a leader node has the option to send the request down in its sub-network.
Users might also want to relax the conditions and not decrease the bandwidth performance with requests that are travelling the ring many times in order to find a node that accepts them

(BONUS_STEP variable is considered as input given by user). To make this possible, when a leader receives back a request that it previously sent to its sub-network, the bonus time variable on the request will be increased. An interesting property of this communication protocol is that if a request travels again back to the leader node of a sub-network because of the high workload, it can eventually get to another sub-network, if the leader evaluates that it is better to do so.

```
OnRequestReceived(request)
   if (CanExecuteRequest(request))
      AddRequest(request);
   else if (IsLeaderNode(request) AND CanExecuteOnMySubNetwork(request))
   {
      if (already received this request)
         request.bonusTime += BONUS_STEP
      SendToSubNetwork(request)
   }
   else
      Send(request);
```

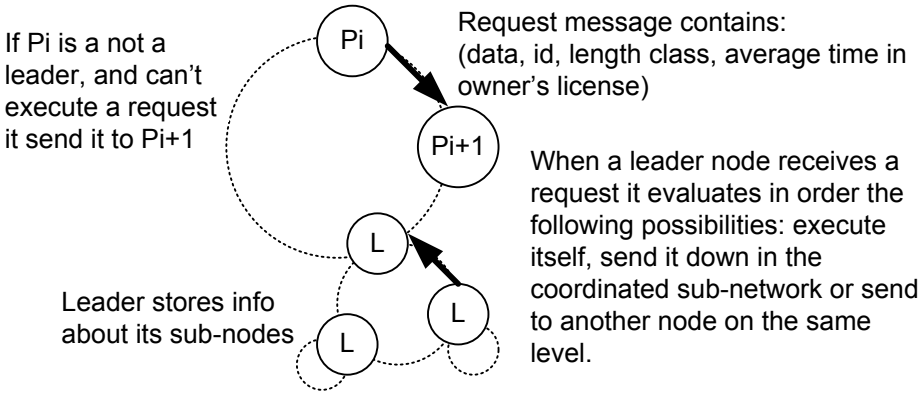


Figure 2: High level decision making for a new request.

A **Gather** message is initiated by leaders of each sub-network at fixed time periods and sent only to the nodes on the same level. The role of this message is to have a snapshot of the current load inside nodes. This information will be used to take a decision if a leader node should accept a request or not to be executed in its sub-network. The gather messages are asynchronous between different sub-networks. When a node receives a gather message, it adds its local load information (like how much load is in there) to the message and sends it further. When the message is received by the leader it updates its load information table. Figure 3 is representative for this flow.

The code below presents the action code of every node and the handler function for receiving Gather messages. *lastTimeGatherSent* is a variable where we store the timestamp of the last Gather message sending occured. T is threshould value set by the user, depending on how often he wants to send the Gather message. *Solve* function is supposed to run the effective job on the request. Function *ExtractNextRequest* selects the task with the highest priority from the local list of tasks.

```
OnUpdate()
   if (IsLeaderNode() AND (GetCurrentTime() - lastTimeGatherSent) > T)
```

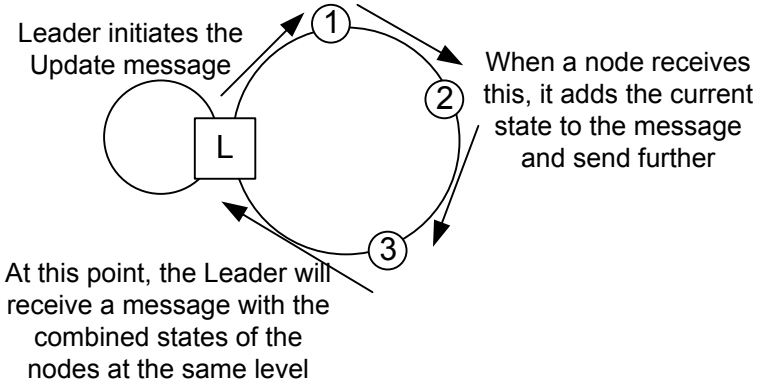


Figure 3: An update message in the ring.

```
   {
      lastTimeGatherSent = GetCurrentTime()
      Gather msg
      SendToSubNetwork(msg)
   }
   request = ExtractNextRequest()
   if (request != null)
      Solve(request)

OnGatherReceived(msg)
   if (IsLeaderNode(msg))
      UpdateLocalState(msg)
   else
   {
      AddLocalState(msg)
      Send(msg)
   }
```

3.3 Decision making to execute a request locally on a node (leaf or leader)

To make computations easier, the average response times specified in the clients licenses are normalized. If $t_1, t_2, t_3, ..., t_n$ are the average response times and $t_{max} = \max t_i$, then $\overline{t_k} = \frac{t_k}{t_{max}}$. Requests are stored and evaluated based on their priorities. The priority of a request is defined as the waiting time for the request to be solved divided by the inverse of the normalized average response time specified in the owners license. If we denote with *WaitingTime(request)* the waiting time of the request to be solved, and *Owner(request)* the index of the owner license then the priority computation can be written as $Priority(request) = \frac{WaitingTime(request)}{\frac{1}{\overline{t_{Owner(request)}}}}$, where *WaitingTime(request)=CurrentTime()-request.createTime*. Priorities of requests waiting on a node are dynamic and could modify in time. This is a key point of the algorithm which gives the fairness between the clients with respect to their average response times. The formula used also helps in the case of requests that travel in the network for a long time. The priority of a request increases with its waiting time regardless of the license's specifications. If a requests travel a long time then it has a bigger priority and its chances to be added in a node are increased. Requests are stored in a node using a linked list. At each query of function CanExecuteRequest(request) the algorithm iterates over the existing items and sum up the time needed to compute all requests

that have a priority higher than the new request. Using the bonus modifier and comparing the result with the average response time of the request's owner we can find out if the request can be executed on that node or not. Also, in the case of requests that are close to ideal execution deadline or should have been executed until now, we check if their priorities are higher than all other priorities waiting in the node. If this is valid then this node is good fit for the request because it will be the first request selected for execution, thus minimizing the delays in response time. A pseudocode for this is given below. *requestsList* is storing requests of a node. *request.bonus* represents the bonus time given by the leader, while *request.idealTimeOfFinish* is the precomputed time when the request should be solved in order to satisfy the owner's license specfication. T is a threshold value defined by user. It could be either the average time for moving data between consecutive nodes, the average time needed for processing it, or a heuristics combining these.

```
CanExecuteRequest(request)
   totalEstTime = 0;
   newEstTime = GetEstTimeToCompute(request) / request.bonus
   bestPriority = null;
   foreach req in the requestsList
   {
      if (Priority(req) > Priority(request))
      totalEstTime = totalEstTime + GetEstTimeToCompute(req);
      if (Priority(req) > bestPriority)
      bestPriority = Priority(req)
   }
   remainingTime = request.idealTimeOfFinish - (GetCurrentTime() + totalEstTime)
   isCloseToDeadline = (request.idealTimeOfFinish - GetCurrentTime()) <= T
   return (remainingTime >= 0 OR
            (isCloseToDeadline AND bestPriority < Priority(request)*request.bonus)
```

3.4 Decision making to execute a request on a sub-network

This type of decision is valid only for leader nodes. In order to make this possible the gather messages are sent in the sub-network in order to collect workload information. In an ideal case, a leader would know informations about all waiting requests in its sub-network nodes and run the same CanExecuteRequest function. But such a message would be too large creating a bandwidth and processing time overhead. A tradeoff solution between performance and quality of the decision result is to gather statistics on how much time the current requests would take to execute on different intervals of priorities. If P_{high} and P_{low} are estimated bounds of the priorities, then splitting on N equal intervals would result in $TimeSum_i$ storing the sum of times to solve the requests with priorities in interval $\left[P_{low} + \frac{P_{high}-P_{low}}{N} * i, P_{low} + \frac{P_{high}-P_{low}}{N} * (i+1) - 1\right]$. The tradeoff can be adjusted using variable N.

The gather message will contain the *TimeSum* array. When adding the local state to a gather message, a nodes responsibility is to iterate through all its waiting requests and for each one to add in the corresponding array index (the correct priority interval) the time needed by the node to solve it. The leader will keep the final *TimeSum* array and use it for decision making. To find out if a request can be executed by the leaders sub-network we need to sum up the values of all intervals of greater priority than the considered request. Then, we divide this sum to the number of nodes in the sub-network to find out the average time needed to finish all higher priority requests. The close to ideal deadline test is used here too, but this time we check if there

are any values bigger than zero on intervals with greater priority than the considered request. Below is presented the pseudocode for adding a local state to the gather message and the decision making of a leader if it should accept or not a request in its sub-network. *GetPriorityInterval* does simple math to get the interval index from the priority of a request.

```
AddLocalState(message)
   foreach req in the requestsList
      message.TimeSum[GetPriorityInterval(req)] += GetEstTimeToCompute(req)

CanExecuteOnMySubNetwork(request)
   P = Priority(request)
   totalTime = 0;
   for i = P +1 to N
      totalTime += TimeSum[i]
   averageTotalTime = totalTime / NumNodesInSubNetwork
   remainingTime=(request.idealTimeOfFinish - (GetCurrentTime() + averageTotalTime))
   isCloseToDeadline = (request.idealTimeOfFinish - GetCurrentTime()) <= T
   return remainingTime>= 0 OR
           (isCloseToDeadline AND there is no TimeSum[k]>0 with k from P+1 to N)
```

The complexities of the operations used here are linear which can be good or bad depending on request's granularity. If there are generally very small requests to execute then this linear time might affect the global performance. An idea to solve this case would be to use a heap tree data structure (which provides logarithmic time for operations) and to rebuild the tree at different time intervals considering the newest priorities.

4 Simulation results

To demonstrate that the algorithm satisfies the proposed goals, a simulator in MPI has been created. The nodes are processes on different machines connected in a network. The test implies a total of 64 processes over 8 machines. Random requests where continuously generated with a normal distribution in length classes. The estimated times to compute the requests were between $[10, 500]$ milliseconds (depending on the computing power of the nodes). Same interval was used for the average response times in the clients licenses. Two relevant tests are used to show how the load balancer works. The results are compared to the results of the algorithm in [1].

Test 1: Check the response times with different workloads.
For this test, the simulator created random requests considering the total computing power of the system. Table 1 shows a comparison between both algorithms in terms of response time delays, given as a percentage value from the value promised in the owner's license. Final results were obtained by averaging multiple simulation results. In the proposed algorithm the average response time goal is satisfied, with important delays appearing just when the workload was too high.

Test 2: Check the fairness between requests with respect to the average response time when the available hardware resources are not enough for satisfying the requests. The simulator creates random requests to simulate a high workload then checks how many of them were solved per interval of average response time. The initial interval of average response times [10,500] was split in 5 intervals as the Table 2 shows. Ideally, the number of requests solved per each interval should be inverse proportional to the average value of the interval.

System workload	Average delays in response time - proposed algorithm	Average delays in response time algorithm in [1]
20 %	1.23 %	15 %
50 %	1.45 %	26.4 %
100 %	1.72 %	54.8 %
200 %	103.14 %	104.2 %

Table 1: Shows average response times with different system workloads.

Intervals of average response times	Average number of requests solved in proposed algorithm	Average number of requests solved in [1]
10-100	4233	1651
101-200	2397	1675
201-300	1683	1649
301-400	779	1693
401-500	541	1680

Table 2: Number of requests solved for different average response time intervals.

The results show that the second goal is satisfied too in the proposed algorithm, while in the other load balancer there is no fairness between the clients.

The proposed algorithm performs much better when comparing the maximum waiting times of requests thanks to the priority formula. Because of the overhead needed to satisfy the goals, the proposed algorithm had a throughput with 2.11% smaller than the reference algorithm. With a proper tuning of the BONUS_STEP, the number of intervals for splitting the local state data and the time to initiate a new Gather message the algorithm can obtain peak performance with minimizing the overhead. These variables should be tuned considering the granularity of the nodes and the available bandwidth. In the simulation, BONUS_STEP was equal to 2, the number of intervals was 5 and the time to initiate Gather messages was 300 milliseconds. As a recommendation, these variables values should actually represent a percentage value of real input data.

5 Conclusion

This paper presented a load balancing algorithm for distributed real-time systems which have a hierarchical ring topology. The algorithm has two proposed goals: satisfy the average response time if the computing power allows this and keep the fairness between clients with respect to the response times specified in their license. The results presented in Section 4 demonstrate that the algorithm satisfies the proposed goals.

Bibliography

[1] Oguz AKAY, Kayhan ERCIYES, *A Dynamic Load Balancing model for a distributed system*, Mathematical & Computational Applications, 8(3):353-350, 2003.

[2] Ciprian Paduraru, *A New Online Load Balancing Algorithm in Distributed Systems*, Symbolic and Numeric Algorithms for Scientific Computing, SYNASC 14th edition, Pages:327-334, 2012.

[3] Andrew S. Tanenbaum, *Modern Operating Systems (3rd Edition)*, Prentice Hall, December 2007.

[4] Kwang Soo Cho, Un Gi Joo, Heyung Sub Lee, Bong Tae Kim, and Won Don Lee, *Efficient Load Balancing Algorithms for a Resilient Packet Ring Using Artificial Bee Colony*, Applications of Evolutionary Computation, LNCS, 6025:61-70, 2010.

[5] G. Ravindran and M. Stumm, *Hierarchical Ring Topologies and the Effect of their Bisection Bandwidth Constraints*, Proc. Intl. Conf.Parallel Processing, I:51-55, 1995.

[6] Perry Fizzano and Clifford Stein, *Scheduling on a Ring with Unit Capacity Links*, Proceedings of the sixth annual ACM symposium on Parallel algorithms and architectures, Pages:210-219, 1994.

[7] Johannes E. Gehrke , C. Greg Plaxton and Rajmohan Rajaraman, *Rapid Convergence of a Local Load Balancing Algorithm for Asyncronous Rings*, Distributed Algorithms, LNCS, 1320:81-95, 1997.

[8] Young-Soo Myung, Hu-Gon Kim, Dong-Wan Tcha, *Optimal Load Balancing on Sonet Bidirectional Rings*, Operations Research, 45(1):148-152, 1997.

[9] Dekel Tsur, *Improved scheduling in rings*, Journal of Parallel and Distributed Computing, 67(5):531-535, 2007.

[10] Amir Gourgy, Ted H. Szymanski, *Cooperative Token-Ring Scheduling For Input-Queued Switches*, Journal of Parallel and Distributed Computing, 58(3):351-364, 2009.

[11] Leonidas Georgiadis, Wojciech Szpankowski, Leandros Tassiulas, *A scheduling policy with maximal stability region for ring networks with spatial reuse*, Queueing Systems (Springer), 19(1-2):131-148, 1995.

[12] Joseph (Seffi) Naor, Adi Rosen, Gabriel Scalosub, *Online time-constrained scheduling in linear and ring networks*, Journal of Discrete Algorithms, 8(4):346-355, 2010.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):758-767, December, 2014.

Representing IT Performance Management as Metamodel

A. Pajić, O. Pantelić, B. Stanojević

Ana Pajić*, Ognjen Pantelić
Department of Information Systems
Faculty of Organizational Sciences, University of Belgrade
Jove Ilica 154, 11000 Belgrade, Serbia
ana.pajic@fon.bg.ac.rs, pantelico@fon.bg.ac.rs
*Corresponding author

Bogdana Stanojević
Mathematical Institute of the Serbian Academy of Sciences and Arts
Kneza Mihaila 36, 11000 Belgrade, Serbia
bgdnpop@mi.sanu.ac.rs

Abstract: Many empirical studies have shown that the business value from investment in IT projects can be greater than the one being currently achieved. Thus it calls for specific focus on IT governance in order to reach fusion between business and IT goals. Good IT performance management should enable the business and IT executives to understand how IT is contributing to the achievement of business goals. The paper addresses the issue of representing IT governance best practice frameworks as ontological metamodels. Special attention is dedicated to VAL IT framework, which represents a comprehensive framework to maximize business value from IT investments. The paper points out the necessity of analyzing, comparing and integrating IT governance frameworks in order to complement different knowledge and generate ontological metamodel of IT performance management. Scope of our work is in the static aspect of the framework and as the metalanguage Extended Entity/Relationship model is used.
Keywords: IT governance, ontology, metamodels, IT investment, IT performance management.

1 Introduction

In today's changing and competitive business environment, organizations are in a constant struggle to obtain a dominant role in the market. Information technology (IT) has become a key enabler of business process reengineering if an organization is to survive and continue to prosper. They invest substantial financial resources for delivering quality software as their competitive advantage. The increasing use of information technology (IT) has resulted in a need for evaluating the productivity impact of IT. Many empirical studies have shown that the business value from investment in IT projects can be greater than the one being currently achieved [8]. More investments do not mean by design achieving desired goals and better business results. It refers to productivity paradox, initially formulated by Solow [20] in 1987, who pointed out "You can see the computer age everywhere but in the productivity statistics". Thus it calls for specific focus on IT governance in order to achieve fusion between business and IT.

Several studies reported rather low success rates in achieving successful business and IT alignment on the organization level, as perceived by business and IT executives [18]. Figures run so low that they are ranked as one of the top concerns for the upper management, in the last few years. One way to reach the strategic alignment and to bridge the gap between business and IT is through the way companies govern their IT. Effective IT governance is seen as a critical element to ensure returns on IT investment and improved organizational performance [12].

This can be achieved through creation of organizational structure with clearly defined roles and responsibilities regarding information, business processes, applications and IT infrastructure. It can contribute to higher returns on assets with the main goal to provide the support for conducting business in a good manner [19]. Strong emphasis is now placed on developing IT governance frameworks to help the management to ensure that organizations realize optimal value from IT business investments at an affordable cost with a known and acceptable level of risk. ITIL (Information Technology Infrastructure Library) and COBIT (Control Objectives for Information and related Technologies) are recognized in the business and academic world as the most used and adopted frameworks ([4], [7]). However, there are a lot of concerns in regard to adaption and integration of different best practice frameworks, which subsequently bring the logical structures and semantic of the frameworks in forefront.

Achieving IT business value and measuring that value are important governance domains. Good IT performance management should enable the business and IT executives to understand how IT contributes to the achievement of business goals, in the past and in the future. As Brandabur [3] pointed out “the IT capability of organization exceeds many concepts like strengths or competitive advantage and had become an absolute necessity". A focus of IT performance management should be the removal of non-value adding activities and processes. According to Haanappel et al. [9], organizations had very different IT performance management approaches and maturity levels. IT performance measurement framework needs to be balanced, comprehensive and adopted as the tool for evaluation and assessment of IT investments. Therefore, the research will address this issue. Paper goal is to provide insight into semantically rich structure of best practice frameworks for supporting management and governance of IT. Special attention is dedicated to Val IT framework, which represents a comprehensive framework to maximize business value from IT investments. Moreover, the integration of different IT best practice frameworks will result in generating ontological metamodel of IT performance management framework.

In Section 2 we discuss the concept of metamodel and ontology from the academic perspective, and elaborate the IT performance management concept. A new ontological metamodel of VAL IT framework, with rich logical structure and semantics of its relationships, is introduced in Section 3. An ontological metamodel for monitoring IT performance is presented in Section 4 to complete the models of IT governance best practices. In Section 5, we discuss research outcomes and ideas for future work.

2 Related Works and Basic Concepts

2.1 Metamodels and Ontologies

Models as the main instruments of enterprise architecture (EA) have been very useful in addressing IT/business alignment problems [17]. The objective of EA is to enable organization to align business goals and IT investment plans. EA models are striving to be executable in order to enable enterprise to adapt IT and enterprise models to change situations and to increase business opportunities.

Models on different levels of abstraction are in existence today. At a higher-level of abstraction “model of model" is called metamodel. Metamodels play an important role in EA by ensuring semantic consistency and a common language for the enterprise [6]. Atkinson and Kühn [2] have come to the conclusion of two dimensions of metamodeling: linguistic and ontological. The difference between them is in the forms of instancing the objects of the metamodel. Linguistic metamodeling deals with the definition of the language and relationships between its elements. On the other hand, ontological metamodels are related to the classification of model elements according to their content. They cope with “instance of" relationships between concepts and

their types. Ontological metamodels are more oriented to the users, focusing on content instead of forms.

Keeping in mind a note of Karagiannis [13] that the combination of metamodels and ontologies provides a solution of fully semantic integration, we present in this paper a VAL IT framework and IT performance management framework as a semantic enrichment metamodels. The scope of our work is in static aspect of the frameworks and as the metalanguage the Extended Entity/Relationship model is used [5]. The data is presented in following types: entity, relationship, attribute and constructor.

2.2 IT Governance Frameworks

The problem of implementing the IT best practice frameworks is receiving growing attention from scientists. IT governance has emerged as an answer to the problem of identifying the business value derived from investments in IT. In order to provide the business value, investment in information technology and systems has to be closely aligned to the corporate strategy. “By creating the necessary structures and processes around IT investments, management can ensure that only those IT projects that are aligned with strategic business objectives are approved, funded, and prioritized" highlighted Symons in [21]. Webb et al. [22] underlined performance and risk management as essential part of IT governance. Moreover, their study has proven that companies with good IT governance model generate higher results than competitors.

Nowadays, COBIT and ITIL are the best known and widely accepted best practices. ITIL represents set of practices focusing on aligning IT services with the needs of businesses. It has become the standard for IT service management (ITSM), providing “a detailed description of a number of important IT practices, with comprehensive checklists, tasks, procedures, and responsibilities, which can be tailored to any IT organization" [15]. On the other hand, COBIT pays more attention on audit and control perspective. In addition it provides maturity models, Critical Success Factors and different metrics [11]. Pereira and Mira da Silva [16] presented in their work the conceptual models of both frameworks, highlighting the fact: “a very complex framework with several dependencies between processes". Moreover, the COBIT metamodel can be found in the literature [7] and in the following we will focus on it. At the heart of COBIT are 34 processes which are grouped into four life cycle domains: Plan and Organize (PO), Acquire and Implement (AI), Deliver and Support (DS), Monitor and Evaluate (ME). Each of these 34 processes has goals, divided into activity, process and IT goals. It is well-structured and therefore applicable for semantic metamodeling.

The main contribution of the presented model is reflected in identifying the possibilities of framework improvement. For example, the authors realize that the component’s activity and the control objective are both connected to the process, but not related to one another. They consider that control objectives have significant overlapping with activities and it can be eliminated by dropping the relationship between activity and process. Furthermore, new processes can be integrated in the framework, which has substantial impact on the model flexibility.

The further subject of this research will be VAL IT framework, developed and maintained by the IT Governance Institute. It sets good practices for the process of value creation. Val IT framework is primarily designed to complement COBIT, but it can be used without a prior implementation of COBIT framework.

2.3 IT Performance Management

Measuring IT performance ensures that organizations maximize the business value of their IT investments. IT performance management can be defined as the area of setting goals, responsibility accounting and monitoring and improving the performance of IT [9]. Four key areas of

IT performance management are recognized by Andra [1]. The two most important areas are IT efficiency and effectiveness. According to Kifor and Tudor [14], IT effectiveness and efficiency represents today some of the most important Key Performance Indicators and a permanent concern for every organization. Many methods, tools and best practices exist to demonstrate the added business value of IT. Problems in measuring IT performance emerge when traditional financial methods are applied to information systems, which often generate intangible benefits. Despite a number of well-established methods available in the market, they are not widely adopted among IT and business executives in evaluation of IT performance. A limited amount of literature is available on how an organization can apply and improve their IT performance management. It is necessary to understand in depth framework's complex structure and purpose with the aim to adopt framework and using it in practice.

3 The Metamodel of VAL IT Framework

In the age of information and knowledge, it is no longer sufficient to measure only financial performance, but instead it is necessary to determine the value of intangible assets. From an IT governance point of view, evaluating the business value of IT investments is of high importance in order to control, govern and manage IT functions. As a comprehensive framework, VAL IT is dedicated to answer this issue with the primary goal to maximize business value from IT investments. According to the ITGI, VAL IT "adds best practices for the end, providing the means to unambiguously measure, monitor and optimize the realization of business value from investment in IT". It is systematic and comprehensive approach for measuring and delivering value, which represents one of the five focus areas of IT governance. Val IT supports organization by providing clear and consistent policy to improve IT investments decisions and returns on investments.

The framework covers the value governance processes and management practices, portfolio and investment management, with ongoing measurements. They represent the three main domains. Each domain includes a number of processes, key management practices and activities that need to occur in order to select the investments with the highest potential to create value and to manage them. Twenty two processes are contained in these domains. Each process produces specific outputs and delivers it to other processes as their inputs. Therefore, both of them are categorized as results and presented through specialization relationship type. Two types of output benefits are distinguished in VAL IT framework. The first are business benefits which influence value directly to. On the contrary, intermediate benefits do not create value despite they might fulfill stakeholders' needs. Thus, we link outputs to the concept stakeholder. Furthermore, process is decomposed to activities and for each activity it is indicated who should be responsible, accountable, consulted and informed (RACI chart). Responsibilities and accountabilities are defined for typical roles. The stated categories might be assigned to one or more of these roles, which might be undertaken by one person or single organization entity in smaller enterprises. Categories are identified as entity and its relationship with role and activity entities is treated as higher-level entity, assignment aggregation.

The VAL IT management guidelines illustrate possible assignments of responsibilities to different roles. Furthermore for each VAL IT process, guidelines include key activities that need to be assumed, activity description, inputs and outputs. Besides, VAL IT processes are collection of practices, including activities and procedures. Within the processes, a set of key management practices are introduced as main characteristics which lead to success. Hence, the process entity is in relationship with management guidelines and management practice entities. It contains management practices and is supported by management guidelines. There are three types of goals and metrics defined for corresponding levels for each of the processes. Domain

goals and metrics describe what has to be done in order to deliver optimal value from IT-enabled business investments. In addition, they were enabled to achieve process goals which are measured by process metrics. The process goals and metrics have been influenced by the process key management practices. Activity goals and metrics are established by process goals and they need to occur inside each process. It is important to measure what has actually been achieved, before outcome is met and afterwards. Maturity level is assigned to each of three domains, based on particular maturity model. Maturity model is given to help enterprise identify its current state and possible future states. The point is to set the priorities for further improvements.

The most important component of Val IT is business case, which is essential for ensuring additional value from business changes. Selecting the right investments closely depends on well defined and comprehensive business case. Business case includes different business processes and a set of assumptions how value will be created. They guarantee expected outcomes based on the major IT input recourses. In addition, business case should be based on key indicators, both financial and non-financial.

Figure 1 shows our ontological metamodel of VAL IT.

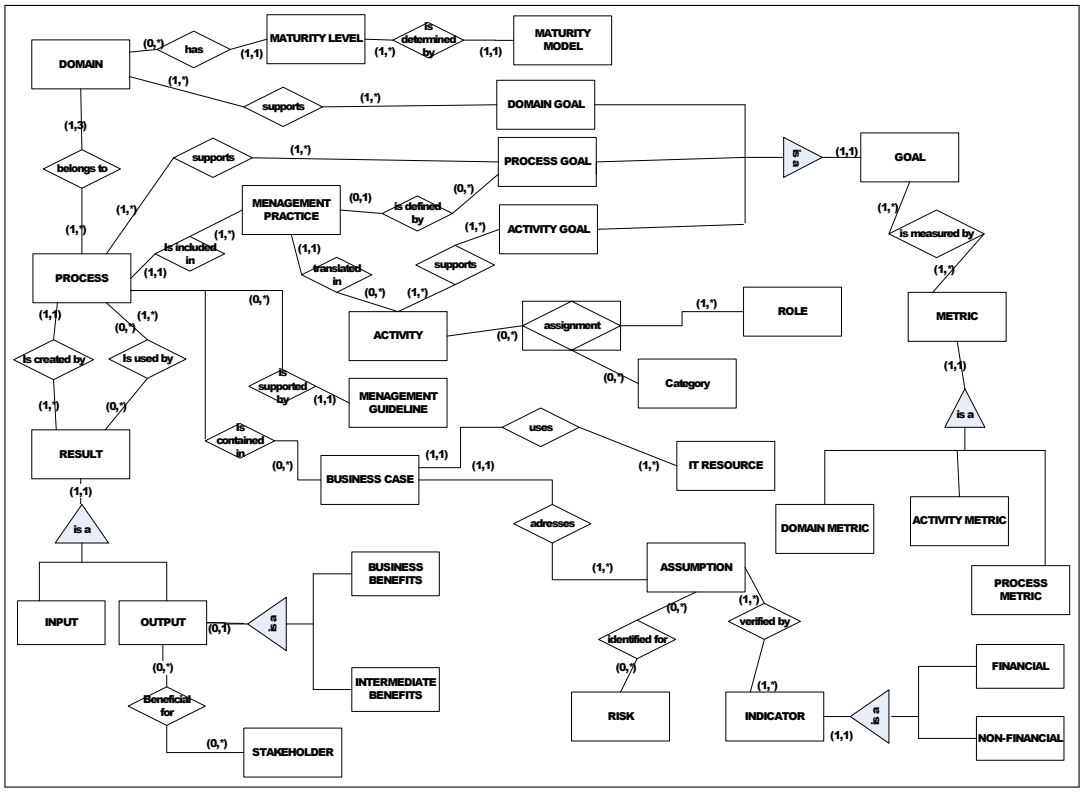


Figure 1: Ontological metamodel of VAL IT

4 IT Performance Management Metamodel

Although there are a great number of methodologies, from quantitative metric ROI, to higher qualitative metric of the IT Balanced Scorecard, there is not the best possible solution or a standard for measuring the value of IT. Simple ROI or other financial metrics are not good enough. Therefore, a comprehensive IT performance management framework will ensure that organization significantly improve their IT investment returns based on projected business value as well as the actual value delivered.

As we mentioned before, the ontological metamodels represent an adequate tool for the analysis, adaptation and integration of best practices in governance of IT. We analyzed the best known and widely accepted IT governance frameworks with focus on COBIT framework in Section 2 of the paper. Different kinds of metrics, such as Key Performance Indicators (KPI), Key Goal Indicators (KGI), and Critical Success Factors (CSF), are suggested in widely accepted IT governance framework COBIT in order to monitor the implementation of each process [7]. VAL IT framework complements the earlier established and used COBIT, going one step forward in setting good practices for the process of value creation. It includes portfolio and investment management processes with ongoing measurements. Therefore, we presented VAL IT framework as ontological metamodel in Section 3, with the idea to analyze and compare these two frameworks in order to generate one comprehensive ontological metamodel of IT performance management framework. Despite different focuses and logical structures, these two frameworks are enough compatible to be compared and integrated in order to generate one comprehensive ontological IT performance management metamodel. Connections between the components of the frameworks are found after performing in-depth analysis of the metamodels. Entity types, such us Process, Activity, Goal, Metric and Maturity model, have similar meanings and attributes in these frameworks. The study covers only the components in the IT performance management domain. Furthermore, the new elements are integrated in the metamodel as the improvement of IT governance best practice frameworks.

The starting point of metamodel is the entity type Business case, essential for ensuring additional value from IT investments. Business case includes different processes and assumptions of how value will be created. For IT performance management it is important to evaluate expected outcomes of business case through different milestones. A milestone is a significant event, which belongs to a business case and is used to monitor the progress in achievement of a particular outcome. The relationship between entity types Milestone and Expected outcome implies that a milestone is used as the checkpoint only for one outcome. Reading the relationship the other way around shows that one outcome has been evaluated through one or more milestones. COBIT is frequently used as the standard for IT governance maturity assessment. In order to apply it as a tool to assess IT performance a lot of expert knowledge is needed. There are lots of metrics, but little support for improving decision-making process. Combining maturity models elements of both frameworks, we have defined elements Maturity model and Maturity level for business case. The reason for assessing maturity of business case lies in the fact that it is active during the whole economic life cycle of investments.

The following IT performance management concepts are defined: Critical Success Factor (CSF), Key Performance Indicator (KPI) and Key Risk Indicator (KRI). They are defined by management guidelines of COBIT and Val IT frameworks. CSFs are elements vital for business case to be successful. Each CSF refers to the milestone and is supported by KPIs and KRIs. The concept of key risk indicator is introduced and it supports process of risk assessment. Relationship between entity types CSF and KPI is treated as higher level entity, aggregation with Rating criteria as attribute. The same type of relationship is defined among entity types CSF and KPI. In addition, business case should be based on KPIs, both financial and non-financial. The relationship between employee and KPI implies that an employee is responsible for zero or more Key Performance Indicators. Reading the relationship the other way around shows that a KPI can belong to only one employee. KPI owner is responsible for proactive monitoring of results progress and creation of KPI evaluation report, which can lead to actions of improvement. One action can be caused by only one KPI evaluation report according to metamodel. It is important to emphasize the difference between metric and KPI concepts. KPI is a metric, but metric is not necessarily a KPI. The cardinality of relationship defines this limitation. There are three types of goals and metrics defined for corresponding level in business case. Process goals and

metrics describe what has to be done in order to deliver optimal value from IT-enabled business investments. The Activity goals and metrics are established by process goals and they need to occur inside each process. In addition, IT goal is introduced with the aim to succeed IT and business alignment. Both concepts are presented through specialization relationship type. The relationship between entity types Goal and Metric is treated as higher level entity, aggregation with Score as attribute. A metamodel of IT performance management framework is presented in Figure 2.

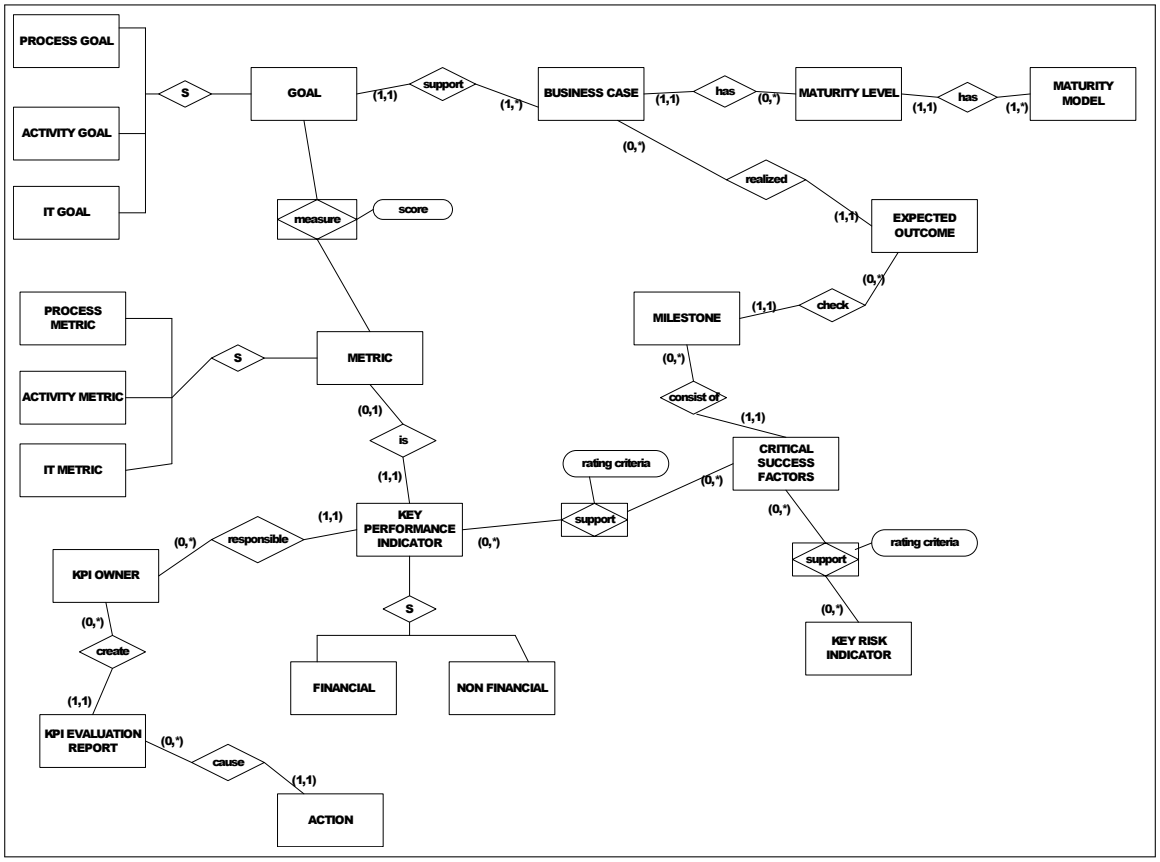


Figure 2: IT performance management ontological metamodel

5 Conclusion and Future Works

Information technologies and systems take a key role within an organization. Organizations are more dependent on IT than ever before and different challenges in management and governance of IT functions have emerged. The goal of achieving a high degree of IT/business alignment has been one of the top priorities for IT professionals in academic and industry world for several years. Therefore, there are a lot of concerns in regard to the IT governance frameworks. These frameworks should support the contribution of IT to the overall value of the enterprise.

There is no single IT governance model that fits all businesses. The choice of model depends on multiple factors. Many of the existing frameworks are complementary, with strengths in different areas. They overlap each other and an organization will probably use more than one framework to achieve a complete model. Effective implementation of the frameworks demands significant changes in the organization and in its processes. It is necessary to understand in depth framework's complex structure and purpose with the aim to analyze, adapt, compare and

integrate different frameworks of IT best practice. Hence, metamodeling can be a good starting point for enterprise specific governance model adaptation and configuration.

We used metamodeling as methodology for adaptation and customization of frameworks on a specific IT performance management domain. IT performance management requires careful preparation and planning, by following set of rules and best practices. It starts by defining what is important to measure and what the goals to be achieved are. Afterwards, it is important to continue with monitoring of progress towards defined goals for permanent improvement.

In this paper, we analyzed IT governance best practice frameworks and their semantic metamodels based on the existing literature. We provided same initial observation on conceptual models and ontologies, underlying the importance of understanding the logical structure of IT governance best practice frameworks for planning successful model integration. The approach of creating ontological metamodels supports comparison and integration of different IT governance frameworks in order to meet the semantic integration challenges. VAL IT and COBIT frameworks are closely related, having the different purposes. VAL IT is addressing strategic and evaluation questions, while COBIT is more oriented on IT architecture and the delivery of high quality IT services. By exhibiting the entity types and associated relationship types of VAL IT framework, we captured points of overlapping among these frameworks and moreover the key areas where they complement each other in order to use strengths of both models.

The main contribution of this work is reflecting through the creation of ontology metamodel of IT performance management framework. Through the model definition the underlying logical and semantically rich structure of the framework was represented. It was examined from different viewpoints, adjusted to conceptual level and improved through the elimination of substantial overlapping, supporting complex structures and relationships amongst entities. Our work captured the knowledge of best practice guidance in creating the business value of information technologies investments focusing on their performance. The resulting metamodel is seen as the first step in building specific enterprise governance model and it is essential to complement it with the knowledge of other frameworks.

In future studies, the methamodel will be adapted on the processes and structures of an organization. It will be used in developing application for monitoring IT performance. It should complete models of IT governance best practices to address in a comprehensive way all critical questions regarding the IT performance.

Acknowledgements

This research was partially supported by the Ministry of Education and Science, Republic of Serbia, Project number TR36006.

Bibliography

[1] Andra, S. (2006); Action-Oriented Metrics for IT Performance Management, *Cutter IT Journal*, 19(4): 17-21. http://www.cutter.com/content-and-analysis/journals-and-reports/cutter-it-journal/sample/itj0604d.html

[2] Atkinson, C.; Kühne T. (2003); Model-Driven Development: A Metamodeling Foundation, *In IEEE Software*, 20(5): 36-41.

[3] Brandabur, R.E. (2013); IT Outsourcing - A Management-Marketing Decision, *International Journal of Computers Communication and Control*, ISSN 1841-9836, 8(2): 184-195.

[4] Broussard, F.W.; Tero, V. (2007); Configuration and Change Management for IT Compliance and Risk Management: The Tripwire Approach, White Paper.

[5] Engels, G.; Gogolla, M.; Hohenstein U.; Hülsmann, K.; Löhr-Richter, P.; Saake, G.; Ehrich, H.-D. (1992); Conceptual Modelling of Database Applications Using an Extended ER Model, *Data & Knowledge Engineering*, ISSN 0169-023X, 9(2): 157-204.

[6] Franke, U. et al. (2009); Decision Support oriented Enterprise Architecture Metamodel Management using Classification Trees, *Proceeding of Enterprise Distributed Object Computing Conference Workshops*, EDOCW, Auckland, New Zealand, 328-335.

[7] Goeken, M.; Alter, S. (2009); Towards Conceptual Metamodeling of IT Governance Frameworks Approach-Use-Benefits, *Proceeding of 42nd Hawaii International Conference on System Sciences HICSS, Hawaii, USA*, ISSN 1530-1605, 1-10.

[8] Gu, B.; Xue, L.; Ray, R. (2008); IT Governance and IT Investment Performance: An Empirical Analysis, *Proceedings of International Conference on Information Systems* ICIS, http://aisel.aisnet.org/icis2008/30

[9] Haanappel, S.; Drost R.; Harmsen, F.; Brinkkemper, S.; Versendaal, J.M. (2011); A framework for IT performance management, http://www.cs.uu.nl/research/techreps/repo/CS-2011/2011-006.pdf

[10] Information Technology Governance Institute (2005); Measuring and Demonstrating the Value of IT, USA, White paper.

[11] Information Technology Governance Institute (2007); COBIT 4.1 Edition, USA, White Paper.

[12] Jacobson, D.D. (2009); Revisiting IT Governance in the Light of Institutional Theory, *Proceeding of 42nd Hawaii International Conference on System Sciences*, HICSS, Hawaii, USA, 1-9.

[13] Karagiannis, D.; Höfferer, P. (2008); Metamodeling as an integration Concept, Software and Data Technologies, In: Springer Berlin Heidelberg, 37-50.

[14] Kifor, V.C.; Tudor, N. (2013); Quality System for Production Software as Tool for Monitoring and Improving Organization KPIs, *International Journal of Computers Communication & Control*, ISSN 1841-9836, 8(2): 235-246.

[15] OGC 2007, The official introduction to the ITIL service lifecycle, London, UK: The Stationery Office, White Paper.

[16] Pereira R.; Mira da Silva M. (2012); A Literature Review: Guidelines and Contingency Factors for IT Governance, *Proceeding in Mediterranean & Middle Eastern Conference on Information Systems*, EMCIS, Munchen, Germany, 342-360.

[17] Saat, J. et al (2010); Enterprise Architecture Meta Models for IT/Business Alignment Situations, *Proceeding of 14th IEEE International Enterprise Distributed Object Computing Conference*, EDOC, Vitoria, Brazil, ISSN 1541-7719, 14-23.

[18] Silvius, A. J. G. et al (2013); The Relationship between IT Outsourcing and Business and IT Alignment: an Explorative Study, *Computer Science and Information Systems*, ISSN 1820-0214, 10(3): 973-998.

[19] Simonsson, M.; Johnson, P. (2008); The IT Organization Modeling and Assessment Tool: Correlating IT Governance Maturity with the Effect of IT, *Proceedings of 41st Annual Hawaii International Conference on System Sciences*, HICSS, Hawaii, USA, pp. 431.

[20] Solow, R. (1987); "We'd better watch out", *New York Times Book Review*, pp. 36.

[21] Symons, C. (2005); IT Governance Framework, Forrester, White Paper.

[22] Webb, P. et al (2006); Attempting to Define IT Governance: Wisdom or Folly?, *Proceeding in 39th Annual Hawaii International Conference on System Sciences*, HICSS, Hawaii, USA, pp. 194a.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):768-775, December, 2014.

GLM Analysis for fMRI using Connex Array

A. Ţugui

Andrei Ţugui
Politehnica University of Bucureşti
Romania, 061071 Bucureşti, Splaiul Independenţei, 313
andrei.tugui@yahoo.com

Abstract: In the last decades, magnetic resonance imaging gained lot of popularity, and also functional magnetic resonance imaging (fMRI), due to the fact that MRI is a harmless and efficient technique for human cerebral activity studies; fMRI aims to determine and to locate different brain activities when the subject is doing a predetermined task. In addition, using fMRI analysis, nowadays we can make prediction on several diseases. This paper's purpose is to describe the General Linear Model for fMRI statistical analysis algorithm, for a 64 x 64 x 22 voxels dataset on a revolutionary parallel computing machine, Connex Array. We make a comparison to other computing machines used in the same purpose, in terms of algorithm time execution (statistical analysis speed). We will show that by taking advantage on its specific parallel computation each step in GLM analysis, Connex Array is able to answer successfully to computational challenge launched by fMRI computation: the speed-up.
Keywords: Connex Array, Functional magnetic resonance imaging, Image reconstruction, Parallel algorithms, Parallel processing.

1 Introduction

Nowadays, neurological activity can be studied using several investigation techniques, each of them having its own advantage and disadvantage, by studying human brain from several perspectives. Although other techniques like EEG (ElectroEncephaloGraphy) or MEG (Magneto Encephalography) have a satisfactory temporal resolution (milliseconds), when it comes to spatial resolution, PET (Positron Emission Tomography) and fMRI are much more indicated to use [1] - [4]. Those investigation techniques pick information from the blood flow changes. As for fMRI statistical analysis on which we will focus in this paper, far away from the biological and functional characteristics, from computational point of vue it is very time consuming. Processing dataset is basically MRI images acquired voxel by voxel; once the blood flow changes, MRI signal strength changes also, this way one can analyze these changes using the statistical data analysis.

Typically, input data acquired one time can be in range of 100 000 voxels, but the investigation's main problem remains the fact that this data is repeatedly acquired by 100 up to 2000 times [5] (usually because of head moving, to compensate discrete acquisition in time, slice by slice, or because most of the time we acquire a lot of noise which must be filtered). A typical fMRI computational chart is presented in Fig. 1. Usually, fMRI dataset acquired during one observation consist of 8 volumes 64 x 64 x 22 voxels each, having a 3.75 x 3.75 x 3.75 mm spatial resolution. If we represent this data as simple precision floats, the data requires about 29 MB memory space. Obviously, we deal to a high enough spatial and temporal memory space, in fact the acquisition process itself is not time-consuming (usually 1 volume/s), but the processing data is, which bottlenecks a lot nowaday's computers tasks, especially when large amount of data must be computed in real time [6]- [7]. Estimated, today we can process a volume dataset using a dedicated processor during about 5 s, but if we use a GPU, the processing time decreases to just 0.5 s! So, this is the high importance that we assign to processing speed in fMRI computation.

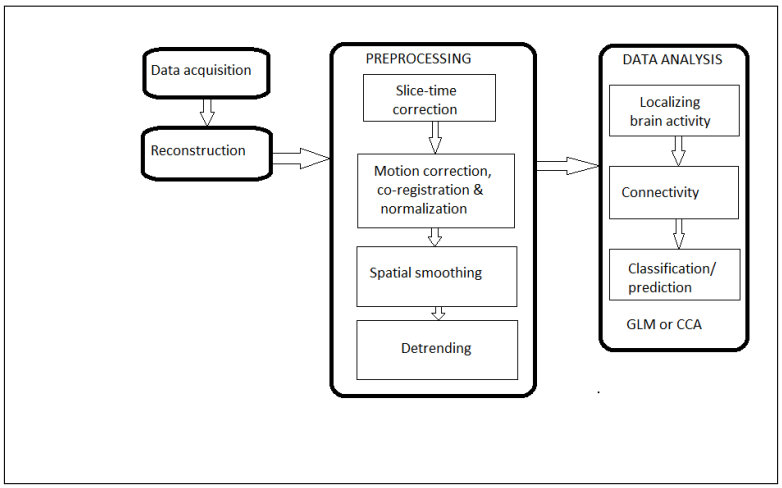


Figure 1: fMRI pipline

In addition, many times fMRI data reconstruction and analysis are made in real time, like BCI (Brain Computer Interface) —a cooperation method between the PC and the subject aiming to solve a given task.
As one can see in Fig.1, fMRI data analysis requires two steps:

- Preprocessing input dataset
- Statistical dataset analysis

In the following, we will present a typically 3D fMRI volume computation using input data from [8], meaning first the preprocessing step and then the statistical data analysis using the GLM (General Linear Model), on a revolutionary parallel computing machine, Connex Array. A full description of this parallel machine's architecture and processing manner is given in papers [9]-[10]. We introduce here only the data vector definition, being in fact the cell that Connex operates, a new N-length data type containing fMRI samples, modeling the vectors from Connex Array [10]. Once we understand the vectorial architecture, we will describe much better this machine's parallelism. Thereby, if we have a C language instruction to be executed as the following one:

```
for( ind = 1;ind <= SIZEOF_VECTOR;ind ++ )
    vect3[ind] = vect1[ind] + vect2[ind]
```

then Connex will sum in one step vect1 and vect2 in the sum vector vect3, whereas a sequential processor would normally do the addition element by element in SIZEOF_VECTOR steps. Here, SIZEOF_VECTOR is a constant holding the vector's length we operate with, by default. Thereby, by typical operators' overloading, using a special C++ library named CVector [11], Connex can sum, multiply, decrease or shift vectors of data. All cycling instructions like *for* are executed using a new CVector instruction like the following:

```
    WHERE (vect1 %2 == 0) {vect1 = 0}
ELSEWHERE {vect1 = 1} ;
```

2 fMRI data preprocessing

Preprocessing usually requires signal filtering. The acquired signal is exposed to a lot of perturbation as head moving (physiological noise), which can be eliminated using Motion Cor-

rection filtering step, or even the acquisition process itself, which is done in discrete steps (once a section) can generate noise: this noise is filtered using the Slice Time Correction filtering step. We show this discrete time slice acquisition of one voxel in Figure 2.

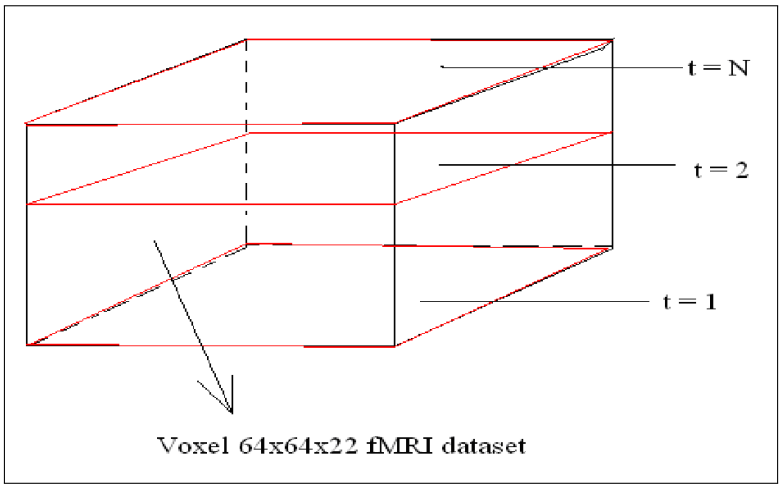


Figure 2: fMRI slice time acquisition

3 Slice time correction

During this preprocessing step, slice time correction is made using *sinc* interpolation. From computational point of view, this step involves:

- one 1D FFT (one Fast Fourier Transform)
- one point-wise multiplication
- one 1D IFFT (Inverse Fast Fourier Transform)

Loading dataset into Connex's memory was made thereby in 64 vectors, 64 floating point elements each (one slice at the time), this way:

```
    vector<float>X[64];
X[0] = [-0.00416487...0.09823550 ];
X[63] = [-0.00687657...0.09823550];
```

1DFFT computation took 6 steps using Cooley-Tuckey algorithm [12], each step the output vector is loaded with the old sample and summed with the new resulted sample, like this:

$$X[0] += (c_i * X[0] + x_{12}) \tag{1}$$

where c_i is the coefficient vector at current step i and x_{12} is a temporary vector obtained from successive input vector shifts, like:

```
    i = 0;
temp1 = shiftLeft(X[0],32);
WHERE (INDEX <32)
{
```

$$x_{12} = temp1;$$

```
}
temp1 = shiftRight(X[0],32);
i += 32;
where (INDEX >= i && INDEX <(i+32))
{
```

$$x_{12} = temp1;$$

} [13]

For *sinc* function computation we used this formula to multiply each pixel:

$$sinc[(\pi/TR)(r - iTR)] \tag{2}$$

where TR = repetition time for each pulse sequence, resulting a coefficient vector c[INDEX] like this:

```
    WHERE (X[INDEX] == 0) {c[INDEX] = 1;}
ELSEWHERE { c[INDEX]=(sin(3.14*(1-INDEX*2)/2))/ (3.14*(1- INDEX*2)/2);}
```

For the reverse Fourier transform computation, the algorithm is similar to 1DFFT algorithm. It worth nothing to see that for this preprocessing step, the total computation time used by SPM (Statistical Parametric Mapping Software) is about 32 s [14], but the same algorithm implementation on Connex Array took only about 100 ms. It worth nothing to say that the same algorithm implementation using fixed-point representation is much faster [15].

4 Image registration

This preprocessing step filters the noise generated by head movement. It involves the alignment of all voxels to a reference voxel (still state). For fMRI, it is sufficient one resolution range with 3 iterations per volume. In each iteration, three quadrature filters are applied on the x, y and z directions, the filters having each 7 x 7 x 7 voxels, complex elements and not Cartesian separable. From the filter response we compute three phase shifts, three gradient shifts and the statistical certainty. Then it results an equation system by adding all voxels and all filters to find the optimization parameter vector. From the optimization parameter vector we compute then a movement vector for each voxel and we apply trilinear interpolation using a 3D texture to rotate and translate the volume. Paper [16] describes the complete image registration algorithm. Head movement field is modeled using a 12—length parameter vector p, like:

$$p = [p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9, p_{10}, p_{11}, p_{12}]^T \tag{3}$$

p vector is computed by solving the 12–length linear equation system with:

$$p = A^{-1}h \tag{4}$$

In this paper we used FFT convolution to compute head movement, although there is also the possibility to use spatial convolution in this purpose. It is worth nothing to follow the inverse matrix computation with Connex, which reveals the parallelism used once the data is loaded into memory. One may consult papers [10] and [13] for further information on Connex architecture and its vectorial computation. We mention that image registration is the most time consuming preprocessing step (about 95% from total time computation time used for a single volume), but using Connex's computational power, we showed once again (see Table 1)

that time bottleneck can be eliminated using Connex Array. If we compare image registration algorithm implementation on Connex Array to the same algorithm implemented on an usual PC using Matlab testing environment, in Matlab we solve this problem during about 14 s, whereas Connex uses only half of this time (Table 1). A closer look on matrix computation with Connex, intense used in this paper, can be found in reference [17].

5 GLM smoothing algorithm

Smoothing dataset is an additional filter step applied before statistical fMRI analysis. Typically, this filter is done for each voxel, except for those voxels near the edge of dataset, and it involves one 3D convolution. Basically, this is done by using a Gaussian lowpass Cartesian separable filter, so the smoothing algorithm becomes in fact a 1D convolution applied in all the three directions (x, y, z). For fMRI smoothing, we use typically 9 x 9 x 9 filters. Thus, for an entire 1D slice we define:

- the input vector:
  - vector<float >X[64];
- the output vector:
  - vector<float >Y[64];
- the impulse response vector:
  - vector<float >H;

By applying smoothing to each line (finally to the entire slice and then to the entire volume), the output vector is processed by multiplying the input vector with the impulse response vector:

$$Y = XH \tag{5}$$

6 Detrending

The final step in preprocessing fMRI data is a special filtering step called detrending. Thus, papers [18] and [19] show how exactly detrending is eliminating the drifts caused by physiological noise and scanner imperfections. Typically, detrending is applied to each slice of the voxel and it searches for the best linearity fit between the slices on one hand, and a particular polynomial on the other hand, fit which is eliminated by the filtering process (polynomial detrending). This filtering procedure uses in fact linear regression, an algorithm similar to statistical GLM analysis, which we will describe in the following.

7 Statistical fMRI analysis using GLM model

After preprocessing, fMRI dataset is subjected to a statistical analysis which can be made using two approaches [20]:

- General Linear Model (GLM)
- Canonical correlation Analysis (CCA)

In this paper we approached General Linear Model (GLM) analysis on each slice (the whole volume), where observation matrix is computed like this:

$$Y = X\beta + \epsilon \tag{6}$$

where:

- Y = the observations (all samples from the slice)
- β = optimization parameters
- X = design matrix given by stimulus paradigm
- ϵ = the errors can not be explained by the model

By minimizing $\| \epsilon \|^2$ it can be shown that the best parameters [14] are given by:

$$\hat{\beta} = (X^T X)^{-1} X^T Y \tag{7}$$

The term $(X^T X)^{-1} X^T$ (we note it with C) is slice independent and can be precomputed and loaded into memory. The only thing we must compute for each voxel in order to find the estimated parameters is the product between the constant C and the current slice Y. If we define the contrast as a column vector (e.g. $[1\ 0]^T$), the test value t [14] is given by:

$$t = \frac{c^T \hat{\beta}}{\sqrt{var(\hat{\epsilon})} c^T (X^T X)^{-1} c} \tag{8}$$

The matrix product is computed fast, as the error and its variance. Then we load again the slice time-series into memory to compute the mean error:

$$\bar{\epsilon} = \frac{1}{N} \sum_{t=1}^{N} (Y(t) - X(t)\hat{\beta}) \tag{9}$$

N = number of time-points and X(t) = design matrix values corresponding to time-point t. The third memory load is made this time to compute error variance an then the test value t. The term $c^T (X^T X)^{-1} c$ is a scalar in fact, which can be precomputed. On Connex, parameter matrix is loaded into 64 vectors 64-length pixels. After the data load into the memory and the matrix computation, we compute matrix product $\hat{\beta} = c^T Y$ for each pixel this way:

$B_{ij} = addAll(Y[i] * C^T[j])$,

`WHERE (INDEX == i) {B[i] =` B_{ij}`; }`

addAll adds all products from elements located at the same positions:

$M_i = addAll(e[i])$

`WHERE (INDEX == i) { M =`M_i`; }`

Finally, the test value t is computed with:

$T[0] = B[0] * (C^T[0] * \sqrt{(variance)})$

where *variance* is the statistical variance (the difference between the mean's square and the mean, computed prior).

8 Conclusions

If on a sequential processor like SPM, GLM analysis would take about 33 s, GLM algorithm implementation on Connex Array requires only 1.7 ms (see Table 1). We demonstrate in this paper that the speed-up in fMRI reconstruction is a challenge that Connex Array can easily overcome.

Table 1: fMRI GLM data analysis on different processors

GLM step	SPM	Matlab	OpenMP	Matlab CUDA	Connex Array
Slice time correction	32 s	280 ms	235 ms	7.8 ms	100.4 ms
Motion compensation	28 s	13.7 s	4.6 s	650 ms	7.857 s
Smoothing	32 s	1.5 s	195 ms	10.4 ms	0.05 ms
Detrending	-	8.6 ms	4.6 ms	0.37 ms	1.73 ms
GLM	33 s	16.6 ms	5.8 ms	0.38 ms	1.7 ms
Total time for GLM	125 s	15.51 s	5.04 s	0.43 s	7.96 s

Bibliography

[1] Tong, S.; Alessio, A.M. (2010); Noise properties in PSF—based fully—3D PET image reconstruction: an experimental evaluation, *Physics in Medicine and Biology*, 55: 1453—1473.

[2] Chen, C.M.; Lee, S.Y. (1990); A parallel implementation of 3—D CT Image reconstruction on hypercube multiprocessor, *IEEE Transactions on Nuclear Science*, 37(3): 1333-1346, DOI: 10.1109/23.57385.

[3] Nishimoto, S.; Vu, A.T.; Naselaris, Th.; Benjamini, Y.; Yu, B.; Gallant, J.L. (2011); Reconstructing Visual Experiences from Brain Activity Evoked by Natural Movies, *Current Biology 21*, 1641—1646.

[4] Holland, D.; Liu, J.; Song, C.; Mazerolle, X. et al. (2013); Compressed sensing reconstruction improves sensitivity of variable density spiral fMRI, *Magnetic Resonance in Medicine*, 70.

[5] Lindquist, M.A. (2008); The Statistical Analysis of fMRI Data, *Statistical Science*, 23(4): 439—464.

[6] Cohen, M.S. (2001); Real—Time Functional Magnetic Resonance Imaging, *Methods*, 25(2): 201—220.

[7] Bernstein, M.A.; King, K.F.; Zhou, X.J. (2004); Handbook of MRI Pulse Sequences, *Elsevier Academic Press.*

[8] http://v04.pymvpa.org/examples.html

[9] Maliţa, M.; Ştefan, Gh. M. (2010); Many-processors & KLEENE's model, *UPB Scientific Bulletin Series C*, 72.

[10] Ştefan, Gh. M. (2010); Integral Parallel Architecture In System—On—Chip Designs, *The 6th International Workshop on Unique Chips and Systems*, Atlanta, USA, 23—26. http://www.arh.pub.ro/gstefan/2010ucas.pdf

[11] Mîţu, B. (2008); C Language Extension for Parallel Processing. http://arh.pub.ro/gstefan/VectorC.ppt

[12] Cooley, J.W.; Tuckey, J.W. (1965); An Algorithm for the Machine Calculation of Complex Fourier Series. Math. Computation, *JSTOR Mathematic of Computation*, 19(90):297-309.

[13] Ţugui, A. (2012); FFT Parallel Implementation for MRI Image Reconstruction, *U.P.B. Scientific Bulletin Series C*, 74: 229-244.

[14] Eklund, A.; Anderson, M.; Knutsson, H. (2012); fMRI Analysis on the GPU Possibilities and Challenges, *Computer Methods and Programs in Biomedicine*, 145—161.

[15] Ţugui, A. (2013); Fixed—point real time MRI reconstruction using Connex Array, *Proceedings of the Romanian Academy Series A*, 14(3): 255—258. http://www.acad.ro/sectii2002/proceedings/doc2013-3/11-Tugui.pdf

[16] Eklund, A.; Andresson, M.; Knutsson, H. (2010); Phase based volume registration using CUDA, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Dallas, USA, 658—661.

[17] Calfa, A.M.; Ştefan, Gh. M. (2010); Matrix Computation on Connex Parallel Architecture, *ICES 2010 —The International Conference on Signals and Electronic Systems*, Gliwice, Poland, 375—378.

[18] Friman, O.; Borga, M.; Lundberg, P.; Knutsson, H. (2004); Detection and detrending in fMRI data analysis, *NeuroImage*, 22(2): 645—655.

[19] Tanabe, J.; Miller, D.; Tregellas, J.; Freedman, R.; Meyer, F.G. (2002); Comparison of Detrending Methods for Optimal fMRI Preprocessing, *NeuroImage*, 15(4): 902—907.

[20] Poldrack, R.H.; Mumford, J.A. (2011); Handbook of Functional MRI Data Analysis, *Cambridge University Press*, New York, USA.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):776-785, December, 2014.

Dissonance Engineering: A New Challenge to Analyse Risky Knowledge When using a System

F. Vanderhaegen

Frédéric Vanderhaegen
[1] Univ Lille Nord de France, F-59000 Lille, France
[2] UVHC, LAMIH, F-59313 Valenciennes, France
[3] CNRS, UMR 8201, F-59313 Valenciennes, France
frederic.vanderhaegen@univ-valenciennes.fr

Abstract: The use of information systems such as on-board automated systems for cars presents sometimes operational risks that were not taken into account with classical risk analysis methods. This paper proposes a new challenge to assess risks by implementing an automated tool based on the dissonance engineering principle. It consists in analysing knowledge in term of dissonances. A dissonance is defined as a knowledge that sounds wrong, or in other words that may present conflicts. The paper focuses on two kinds of dissonances: erroneous affordances when events can be related to erroneous actions and contradictory knowledge when the application of knowledge relates to opposite actions. The proposed automated tool analyses the knowledge base content in order to detect possible dangerous affordances or contradictory knowledge. An example of application is given by using a limited number of simple rules related to the use of an Automated Speed Control (ASC) system for car driving.
Keywords: dissonance engineering, erroneous affordance, contradictory knowledge, risk analysis, car driving system.

1 Introduction

Is there something wrong when engineers or researchers design walking robots directly with two legs without copying the learning process of the human walking that begins initially with the legs and the hands, then with the use of supports and finally with both legs after the complete control of the equilibrium? To control such a process, undesirable events such as lack of knowledge to control equilibrium or breakdown of equilibrium or fatal fall should be studied in order to design algorithms or other devices that are able to prevent the walking robots from a loss of their equilibrium.

Classical risk analysis focuses on the identification and the control of such undesirable events and aims at providing the human-machine systems with barriers in order to protect them from the occurrence or the impact of these events [1]. Despite these barriers, accidents remain and retrospective analyses can help the designers to identify what was wrong. Safety based analysis can apply different methods. The RAMS based methods (Reliability, Availability, Maintainability and Safety based analyses) treat about technical failures. The methods from cindynics treat about organizational dangers Human reliability or human error based analyses focus on the success or the failure of human behaviours respectively. Resilience or vulnerability based methods consider the analysis of the success or the failure of the control of the system stability respectively. This paper proposes on a new way to analyse risks: the use of the dissonance concept to assess conflicts between knowledge.

The dissonance engineering is the engineering science that treats on dissonance [2]. A cognitive dissonance is defined as an incoherency between individual cognitions [3]. Cindynics dissonance is a collective or an organizational dissonance related to incoherency between persons or between groups of people [4]. The occurrence of these dissonances can relate to individual

or collective knowledge when something sounds wrong, i.e. will be, is, maybe or was wrong. A dissonant cognition is linked with contradictory information and a dissonance may produce discomfort due to the occurrence of conflicting cognition or knowledge that controls or affects behaviors, attitudes, ideas, beliefs, viewpoints, competences, etc. Dissonant knowledge of a person or of several persons can explain such conflicts. Nevertheless, dissonances can also be due to the occurrence of important or difficult decisions involving the evaluation of several possible alternatives [5], to divergent viewpoints on human behaviours [1], to the occurrence of failed competitive or cooperative activities [6], [7], to organisational changes that produce incompatible information [8], [9]. Then, the updating or the refining of a given knowledge due to new feedback from field is required but this can also generate dissonances [2].

The more difficult the learning process is to face a dissonance, the less acceptable this dissonance is. Therefore, human operators aim at reducing any occurrence or the impact of a dissonance because it produces discomfort. This activity leads to maintain a stable state of knowledge without producing any effort to change it [3]. Despite this reduction, a breakdown of this stability is sometimes useful in order to facilitate the learning process and refine, verify or confirm knowledge [10]. Such knowledge reinforcement improves the learning abilities. Finally, dissonance can also be seen as a feedback of a decision: dissonance occurs after a decision and this requires a modification of knowledge [9].

Therefore, a discomfort can be a dissonance or can be due to the production of a dissonance, and the detection or the treatment of a dissonance can also produce discomfort. Discomfort can also occur if this dissonance is over the control of the human operators or because the treatment of a detected dissonance increases the human workload or the human error for instance [11], [12]. Such an activity involves a minimum learning process in order to improve the human knowledge and to control such a discomfort. There are then positive and negative feedbacks from the dissonance management. Negative feedbacks relate to discomfort and positive ones to the learning aspect for instance.

Different structures to share tasks between human and machine such as those developed on [13], [14] can be applied for dissonance management and a learning process is usually required to facilitate the control of the knowledge content. Several models exist for self-learning, autolearning, co-learning or cooperative learning [6]. Main of them is based on the reinforcement principle by taking into account previous knowledge and integrating new or future knowledge, in order to create, modify or delete data from the knowledge base and to make it more coherent [15], [16], [17], [18].

This paper proposes an original knowledge analysis based tool to support the control of dissonances into human knowledge. It focuses on two kinds of dissonances: erroneous affordances and contradictory knowledge. This knowledge analysis is based on the knowledge modelling presented on [19]. The concept of affordances was firstly presented by Gibson [20]. Affordances can be defined as invariant relationships between direct perception and possible opportunities for action. The concept is used for different research applications related to human-machine system knowledge management [21]. For instance, the perception of a chair relates directly to the action of sitting. Regarding other human experiences, a chair can also be related to the actions of climbing when a chair is used as a ladder, or for moving or transporting when the action concerns disabled persons. The paper focuses on particular affordances: erroneous affordances when links between occurred or desired events and actions can be erroneous. The last section of this paper gives a practical example of application related to the use of an Automated Speed Control (ASC).

2 The knowledge analysis based tool to control dissonances

The proposed tool analyses the content of a given knowledge base. Two modules are integrated: the module that detects possible erroneous affordances and the module that detects possible contradictory knowledge, Figure 1. The knowledge modelling and reinforcement to create, modify or delete some knowledge are allocated to the users. An interface gives the results of these detections that are validated by the users to reinforce the knowledge modelling and then the knowledge content. The knowledge analysis is based on the knowledge modelling presented on [19]. Therefore, the set of knowledge K contains a list of rules $R(i)$ with a condition of activation noted $Condit(R(i))$ and a conclusion noted $Conclu(R(i))$:

$$R(i) \in K \rightarrow R(i) = (Condit(R(i)) \rightarrow Conclu(R(i))) \tag{1}$$

If $Condit(R(i))$ is to be achieved or realized then $Conclu(R(i))$ has to be realized. For instance, $Condit(R(i))$ can relate to a goal to be achieved or realized and $Conclu(R(i))$ can correspond to the actions to be done on the process by using specific supports of the system or its environment to achieve this goal. These actions can be realized by the users of the system or by an automated module implemented into this system. Some rules are then the procedures to apply in order to achieve a goal and other rules concern the different steps required to achieve a goal. The affordance detection and the contradictory knowledge detection use the function $K\_Analysis$ to identify the rules that can generate possible erroneous affordances or that can present incoherency.

The $K\_Analysis$ function is defined as follows:

$$\begin{aligned} K\_Analysis: \quad & K \rightarrow K \\ & R \rightarrow R^{+} = K\_Analysis(R), \\ & \forall R(i) \in R, Condit(R(i)) \cap BE \neq \{\emptyset\}, R^{+} \leftarrow \cup R(i) \end{aligned} \tag{2}$$

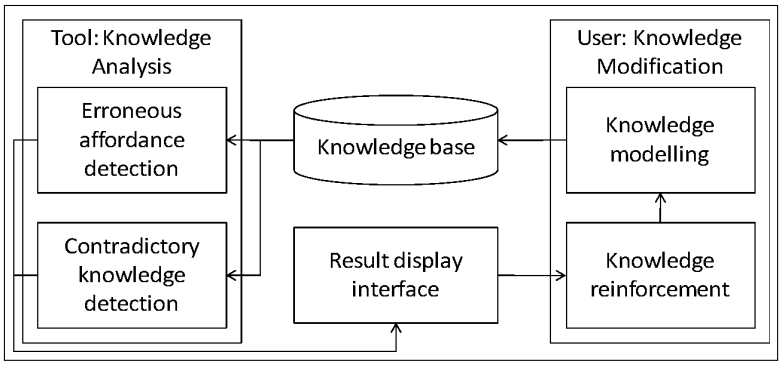


Figure 1: The automated tool modules.

The set K is the set of all the possible rules. For a given rule base noted R of K containing a limited number of rules, the $K\_Analysis$ gives a reduced rule base noted R^{+} of K. R^{+} contains the rules related to the inputs noted BE. BE contains the events that occur or the goals to be achieved. When the condition $Condit(R(i))$ of a rule occurs on BE entirely or partially, then this rule is integrated into R^{+}.

The $K\_Affordance$ function aims at identifying possible new rules combining the condition

and the conclusion of existing rules. It is defined as follows:

$$\begin{aligned} K\_Affordance: \quad & K \rightarrow K \\ & R \rightarrow R^a = K\_Affordance(R), \\ & \forall R(i) \in R, \forall R(j) \in R, i \neq j, Condit(R(i)) \subset Condit(R(j)), \\ & R^a = \cup((Condit(R(i)), Conclu(R(j))), (Condit(R(j)), Conclu(R(i)))) \end{aligned} \quad (3)$$

The result of this function is a new rule base noted R^a of K that combines conditions and conclusions of some rules of R. This function proposes new rules based on the affordance application concept in order to list possible new rules taking into account possible relationships between the condition of a given rule with the conclusion of another one. If a condition of a rule is included into to the condition of another one, then both rules can be used to create new rules. This process is limited to the rules identified by the $K\_Analysis$ function. Then:

$$R^a = K\_Affordance(K\_Analysis(R)) \quad (4)$$

The $K\_Contradictory$ function aims at listing the contradictory rules, i.e. rules that present opposite behaviours. It is defined as follows:

$$\begin{aligned} K\_Contradictory: \quad & K \rightarrow K \\ & R \rightarrow R^c = K\_Contradictory(R), \\ & \forall R(i) \in R, \forall R(j) \in R, i \neq j, \\ & (Conclu(R(i)) \subset \neg Conclu(R(j))) \; or \\ & \qquad (\neg(Conclu(R(i))) \subset Conclu(R(j))), \\ & R^c = \cup(R(i), (R(j)) \end{aligned} \quad (5)$$

When opposite conclusions appear on R, i.e. when both $Conclu(R(i))$ and $\neg(Conclu(R(i)))$ exist, then a possible incoherency occurs. The result of this function is a new rule base noted R^c that contains possible conflicts between rules of R. This process is limited to the rules identified by the $K\_Analysis$ function. Then:

$$R^c = K\_Contradictory(K\_Analysis(R)) \quad (6)$$

This formalism was applied to car driving domain by integrating into the initial rule base the rules related to the use of a Cruise/Speed Control (ASC) system. If the ASC system is activated and if an initial setpoint value is given by the car driver, the ASC has in charge the regulation of the car speed by maintaining this setpoint speed. The "+" and the "-" buttons are used for giving the initial setpoint speed or to modify this setpoint, Figure 2. The "+" button aims at increasing the setpoint value whereas the "-" button at decreasing it.

Several dissonances can generate a possible evolution of the car driver knowledge. The next two sections presents some examples of the detection of possible erroneous affordances and contradictory rules linked to the use of such a system. The wording of the rule condition and conclusion are voluntarily simple in order to illustrate the feasibility of such a knowledge analysis to detect dissonances. Perspectives will consist in using more complex rules with a numerical model integrating for instance belief on rules, belief on condition occurrence or belief on conclusion occurrence.

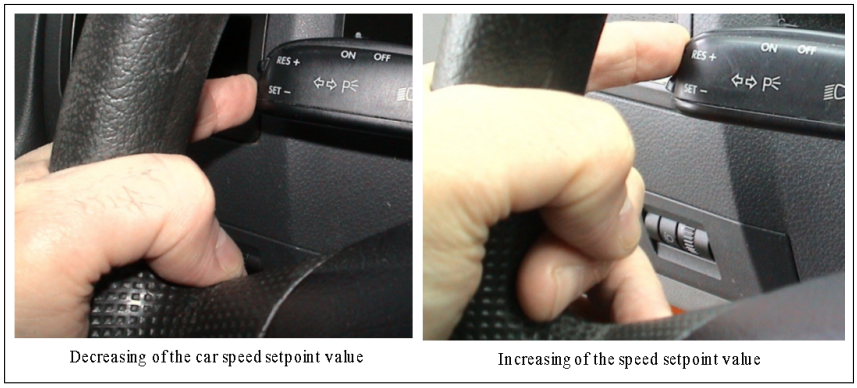


Figure 2: The "+" and "-" buttons of an ASC system of a car.

3 Example of possible erroneous affordance detection

Suppose that a knowledge modeling process produced a knowledge base containing these rules:

- $R(1)$: (the use of the ASC system → turn the activation button on "on")
- $R(2)$: (the deactivation of the ASC system→ brake with the braking pedal)
- $R(3)$: (the increasing of the car speed setpoint → push the "+" button)
- $R(4)$: (the decreasing of the car speed setpoint → push the "-" button)
- $R(5)$: (the increasing of the car speed → push the gas pedal)
- $R(6)$: (the decreasing of the car speed → release the gas pedal)

For instance, whatever the context of the ASC system use, if BE contains initially increasing of the car speed, R^a will list several dissonances to be tested or validated by the car driver. It will contain two possible new rules:

- (the increasing of the car speed setpoint → push the gas pedal)
- (the increasing of the car speed → push the "+" button)

If BE contains initially "decreasing of the car speed", R^a will then contain other new possible dissonances:

- (the decreasing of the car speed setpoint → release the gas pedal)
- (the decreasing of the car speed → push the "-" button)

10 subjects who usually used an ASC system were invited to evaluate the proposed erroneous affordances, Table 1. The outputs of the automated systems were presented to these subjects who have to make comments about them. They have to give their own point of view about the dissonances.

All of them considered that the rules related to the management of the car speed setpoint value are erroneous affordances. However, 8 of them do not consider the other rules as problems and decided to integrate the proposed rules into the knowledge base content. Therefore, these subjects consider that they can manage the car speed without taking into account the management of

Table 1: Subjective evaluation of the proposed erroneous affordances.

Proposed erroneous affordance	Real erroneous affordance?	Consequences
(the increasing of the car speed setpoint → push the gas pedal)	Yes (10 upon 10 subjects)	No modification of the knowledge base
(the increasing of the car speed → push the "+" button)	No (8 upon 10 subjects)	Creation of this rule into the knowledge base
(the decreasing of the car speed setpoint → release the gas pedal)	Yes (10 upon 10 subjects)	No modification of the knowledge base
(the decreasing of the car speed → push the "-" button)	No (8 upon 10 subjects)	Creation of this rule into the knowledge base

the pedals anymore. This reduces their workload. The 2 subjects who did not accept these new rules consider these rules as dangerous because this can lead to adapt a possible body position that can generate problems in case of emergency stop for instance. Figure 3 gives an example of such a body position into the car when managing the car speed only with the "+" and "-" buttons of the ASC system: the legs are crossed because the car speed is managed by a finger that activates the "+" or ""-" buttons, and the position of the legs can therefore become an obstacle or a discomfort in case of emergency stop that may require a quick press on the brake pedal!

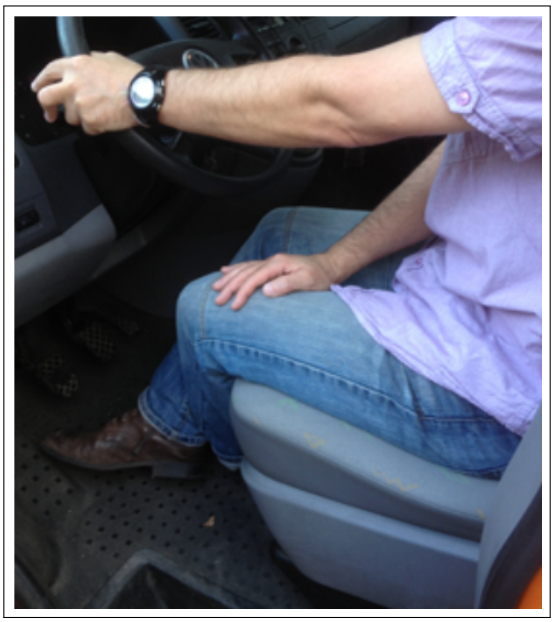


Figure 3: Example of a possible dangerous body position when applying the proposed rules for regulating the car speed.

4 Example of possible contradictory knowledge detection

Suppose that, for another use context, the knowledge modelling process produced another knowledge base combining rules related to the aquaplaning control and some rules related to the ACS control:

- $R(1)$: (the use of the ASC system → turn the activation button "on")
- $R(2)$: (the deactivation of the ASC system → brake)
- $R(3)$: (the reduction of the current car speed that becomes under the setpoint managed by the ASC → accelerate automatically to reach the setpoint value)
- $R(4)$: (the increasing of the current car speed that becomes over the setpoint managed by the ASC → decelerate automatically to reach the setpoint value)
- $R(5)$: (the control of an aquaplaning → ¬ (brake))
- $R(6)$: (the control of an aquaplaning → ¬ (accelerate))

Knowing that the ASC is activated, the car speed setpoint is high and the current car speed is equal to the required setpoint, the occurrence of an aquaplaning may reduce the current car speed due to the natural braking and friction related to the water level on the road. Suppose that in this case, the initial BE content is (control of aquaplaning, deactivation of the ACS, reduction of the current car speed). The contradictory knowledge module identifies some couples of possible dissonant rules. R^c will contains this list of couples of rules:

- ((the deactivation of the ASC system → brake), (the control of an aquaplaning → ¬(brake)))
- ((the control of an aquaplaning → ¬(accelerate)), (the reduction of the current car speed that becomes under the setpoint managed by the ASC → accelerate automatically to reach the setpoint value))

10 subjects were invited to assess these contradictory knowledge proposals. They use an ACS system and are aware about the behaviour to follow and about the car behaviour when an aquaplaning occurs. All of them are agree with the contradictory rules proposed by the automated system, Table 2.

Table 2: Subjective evaluation of the proposed contradictory knowledge.

Proposed contradictory knowledge	Real contradictory knowledge?	Consequences
((the deactivation of the ASC system → brake), (the control of an aquaplaning → ¬(brake)))	Yes (10 upon 10 subjects)	Modification of the current rules or creation of new rules
((the control of an aquaplaning → ¬(accelerate)), (the reduction of the current car speed that becomes under the setpoint managed by the ASC→ accelerate automatically to reach the setpoint value))	Yes (10 upon 10 subjects)	Modification of the current rules or creation of new rules

The contradictory actions (brake, ¬(brake)) and (accelerate, ¬(accelerate)) are then solved by reinforcing the knowledge in different possible ways such as the modification of some current rules or the creation of new rules. They recognize that in emergency situations, it is more natural to use the braking pedal for stopping the car or deactivating a system instead of using the clutch pedal or the "off" button of the ASC system as it is noted on the ASC user manual. Some user manual recommends using the system in particular conditions. For instance, it indicates not to use the system when it is raining. However, water can sometimes occur under bridges for instance even if it is not raining and this can generate aquaplaning.

5 Conclusion

This paper is an original contribution on risk analysis based on dissonance engineering. It proposes a knowledge analysis based tool composed by two main modules: an erroneous affordances detection module, and a contradictory knowledge detection module.

The knowledge analysis consists in identifying possible dissonances into a knowledge base composed by rules. Rules contain conditions of activation and conclusions when they can be activated. The conditions relate to occurred or desired events and the conclusions to the associated actions to be achieved. The erroneous affordances module treats particular dissonances that the users may create. They are new rules for which desired or occurred events may be related to wrong actions. These possible erroneous relations between events and actions are obtained by using the existing rules and initial events. The contradictory knowledge module manages the rules for which the activation presents opposite actions regarding initial inputs.

A practical example is then proposed to study the feasibility of use of such a knowledge analysis based tool. It relates to the use of an Automated Speed Control systems dedicated to car driving. Rules associated to its use and other rules applied for controlling events such as an aquaplaning were proposed. Erroneous affordances and contradictory knowledge are then given by the proposed automated modules. 10 subjects were invited to make comments about these outputs from the proposed tool.

Among the erroneous affordances proposed by the system, there is a rule that was not considered as wrong and that was accepted by 8 subjects upon 10. It concerns the use of the "+" and "-" buttons of the ASC in order to increase or decrease respectively the current car speed on demand instead of using these buttons to control a stable car speed setpoint. The acceleration and deceleration on road or motorway can then be done with such new procedures without involving the legs anymore. The 2 last subjects that agree with the system consider this new rule as dangerous because problems may occur if an emergency stop is required for instance. Problems can be related to the position of the legs or to the body into the car due to the new function allocated to the "+" and "-" buttons for controlling the car speed.

Regarding the contradictory knowledge, all the subjects were agree with the proposals they consider as very dangerous. Indeed, they considered that, in emergency case, people may activate the braking pedal instead of the "off" button of the ASC system or the clutch pedal to deactivate the ASC system. Then, the rules associated to the control of aquaplaning that required no action on the braking pedal and no action on the speed control are contradictory with rules related to the use or behavioural model of the ASC system.

This simple example has shown the interest of such a new approach to analysis risks involving rule based knowledge. Future work research will then focus on more complex applications implementing numerical models of knowledge and will integrate criteria such as uncertainties, beliefs or preferences on knowledge [22], [23]. Future applications will connect automated reinforced learning systems to assist the knowledge reinforcement process. Finally, this paper is a call to future designers of car driving systems such as ASC systems to use a dissonance engineering based risk analysis for designing system functions and user manuals in order to control possible dangerous dissonances and recover side effects of automation.

Acknowledgments

The present research work has been supported by the International Research Group on Human-Machine Systems in Transportation and Industry: the author gratefully acknowledges the support of this institution. The author thanks also P. Richard for his support to format this paper.

Bibliography

[1] F. Vanderhaegen (2010), Human-error-based design of barriers and analysis of their uses *Cognition Technology & Work*, 12: 133-142

[2] F. Vanderhaegen (2013), A Dissonance Management Model for Risk Analysis *Proceedings of the 12th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design, and Evaluation of Human-Machine Systemsl*, Las Vegas, USA, August, 11-15.

[3] L. Festinger (1957), A theory of cognitive dissonance. *Stanford, CA: Stanford University Press.*

[4] G.-Y. Kervern (1995), Eléments fondamentaux des cindyniques (Fondamental elements of cindynics) *Economica Editions, Paris.*

[5] T.-Y. Chen (2011), Optimistic and pessimistic decision making with dissonance reduction using interval-valued fuzzy sets *Information Sciences*, 181(3): 479-502.

[6] F. Vanderhaegen (2012), Cooperation and learning to increase the autonomy of ADAS *Cognition, Technology & Work*, 14: 61-69.

[7] F. Vanderhaegen, S. ChalmĂŠ, F. Anceaux, P. Millot (2006), Principles of cooperation and competition - Application to car driver behavior analysis *Cognition, Technology & Work*, 8(3): 183-192.

[8] O. Brunel, C. Gallen (2011), Just like cognitive dissonance *Proceedings of the 27th International Congress of French Association of Marketing*, 8-20 May 2011, Brussels.

[9] E. E. Telci, C. Maden, D. Kantur (2011), The theory of cognitive dissonance: A marketing and management perspective *Procedia Social and Behavioral Sciences*, 24: 378-386.

[10] E. AĂŻmeur (1998), Application and assessment of cognitive dissonance Theory in the learning process *Journal of Universal Computer Science*, 4(3): 216-247.

[11] F. Vanderhaegen (1999), Cooperative system organisation and task allocation: illustration of task allocation in air traffic control *Le Travail Humain*, 63(3): 197-222.

[12] F. Vanderhaegen (1999), Multilevel allocation modes - Allocator control policies to share tasks between human and computer *System Analysis Modelling Simulation*, 35: 191-213.

[13] F. Vanderhaegen (1997), Multilevel organization design: the case of the air traffic control *Control Engineering Practice*, 5(3): 391-399.

[14] S. Zieba, P. Polet, F. Vanderhaegen (2011), Using adjustable autonomy and human machine cooperation to make a human machine system resilient Application to a ground robotic *Information Sciences*, 181(3): 379-397.

[15] F. Vanderhaegen, S. Zieba, S. Enjalbert, P. Polet (2011), A Benefit/Cost/Deficit (BCD) model for learning from human errors *Reliability Engineering & System Safety*, 96(7): 757-76.

[16] P. Polet, F. Vanderhaegen, S. Zieba (2012), Iterative learning control based tools to learn from human error *Engineering Applications of Artificial Intelligence*, 25(7): 1515-1522.

[17] M. Ercan, T. Acarman (2013), Processing capacity and response time enhancement by using iterative learning approach with an application to insurance policy server operation *International Journal of Computers Communications & Control*, 8(4): 514-524.

[18] C.K. Ang, S.H. Tang, S. Mashohor, M.K.A.M. Arrn (2014), Solving continuous trajectory and forward kinematics simultaneously based on ANN *International Journal of Computers Communications & Control*, 9(3):253-260.

[19] F. Vanderhaegen, P. Caulier (2011), A multi-viewpoint system to support abductive reasoning *Information Sciences*, 181(24): 5349-5363.

[20] JJ Gibson (1986), The ecological approach to visual perception *Lawrence Erlbaum Associates, Hillsdale* (Originally published in 1979).

[21] S. Zieba, T. Inagaki, F. Vanderhaegen (2010), Resilience engineering by the management of affordances Application to intelligent transport system. *Proceedings of the 11th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design, and Evaluation of Human-Machine Systems*, Valenciennes, France, August 31 September 3, 2010.

[22] F. Aguirre, M. Sallak, F. Vanderhaegen, D. Berdjag (2013), An evidential network approach to support uncertain multiviewpoint abductive reasoning *Information Sciences*, 253:110-125.

[23] F. Vanderhaegen, P. Polet, S. Zieba (2009), A reinforced iterative formalism to learn from human errors and uncertainty *Engineering Applications of Artificial Intelligence* , 22(4-5): 654-659.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):786-799, December, 2014.

Application of Fuzzy Reasoning Spiking Neural P Systems to Fault Diagnosis

T. Wang, G. Zhang, H. Rong, M.J. Pérez-Jiménez

Tao Wang, Gexiang Zhang*, Haina Rong
School of Electrical Engineering, Southwest Jiaotong University
Chengdu, 610031, China
wangtaocdu@gmail.com, zhgxdylan@126.com
ronghaina@126.com
*Corresponding author: wangtaocdu@gmail.com

Mario J. Pérez-Jiménez
Research Group on Natural Computing
Department of Computer Science and Artificial Intelligence
University of Sevilla, Sevilla, 41012, Spain
marper@us.es

Abstract: This paper discusses the application of fuzzy reasoning spiking neural P systems with trapezoidal fuzzy numbers (tFRSN P systems) to fault diagnosis of power systems, where a matrix-based fuzzy reasoning algorithm based on the dynamic firing mechanism of neurons is used to develop the inference ability of tFRSN P systems from classical reasoning to fuzzy reasoning. Some case studies show the effectiveness of the presented method. We also briefly draw comparisons between the presented method and several main fault diagnosis approaches from the perspectives of knowledge representation and inference process.
Keywords: fuzzy reasoning spiking neural P system with trapezoidal fuzzy number, fuzzy reasoning, fault diagnosis, trapezoidal fuzzy number, linguistic term.

1 Introduction

Membrane computing, introduced by Gh. Păun in [1], is an attractive research field of computer science aiming at abstracting computing models, called membrane systems or P systems, from the structures and functioning of living cells, as well as from the way the cells are organized in tissues or higher order structures. In recent years, much attention is paid to the spiking neural P systems (SN P systems, for short), an important class of P systems introduced in [2] and investigated in a series of papers (see [3]- [12]), which can be described as a directed graph. An SN P system is a kind of distributed and parallel computing model inspired by the neurophysiological behavior of neurons sending electrical impulses (spikes) along axons from presynaptic neurons to postsynaptic neurons. The features of SN P systems, such as inherent parallelism, understandability, dynamics, synchronization/asychronization, non-linearity and nondeterminism [3], [4], are suitable for solving various engineering problems.

Until now, only a few investigations have focused on the use of SN P systems to solve engineering problems. In [3], a fuzzy reasoning spiking neural P system with real numbers (rFRSN P system) was presented to perform diagnosis knowledge representation and reasoning. In [13], an rFRSN P system was used for fault diagnosis of power systems and three examples were used to verify its effectiveness. The studies in [3, 13], dealing with the fault diagnosis problem, used certainty factors and truth degree values, which are described by real numbers obtained from the frequency of occurrences in historical data, It is known how difficult it is to obtain and process real-time statistical data from power network data and the knowledge of dispatchers and experts in electrical power systems as they usually contain linguistic terms with

some degree of uncertainty. So, using various models, including rFRSN P systems to solve fault diagnosis problems with a certain degree of uncertainty, represents a way to tackle this difficult problem.

This paper discusses the extended version of rFRSN P systems, i.e., fuzzy reasoning spiking neural P systems with trapezoidal fuzzy numbers (tFRSN P systems), and its application to fault diagnosis of power systems. To adapt tFRSN P systems to solve fault diagnosis problems, a matrix-based fuzzy reasoning algorithm (MBFRA) is used inspired by the dynamic firing mechanism of neurons. Given initial pulse values of all input neurons of a tFRSN P system, MBFRA can perform fuzzy inference to obtain the pulse values contained in other neurons and export reasoning results represented by trapezoidal fuzzy numbers. To make MBFRA suitable for multiple faults diagnosis of power systems, a defuzzification method is applied for processing the reasoning results in order to obtain crisp numbers corresponding to them. Some case studies show the effectiveness of the presented method. We also briefly draw comparisons between tFRSN P systems and several other fault diagnosis approaches.

The remainder of this paper is organized as follows. Section 2 introduces concepts and notations used in this work. Section 3 provides the definition of tFRSN P systems and MBFRA. Section 4 discusses the application of tFRSN P systems to fault diagnosis of power systems. Discussions on several fault diagnosis methods are made in Section 5. Conclusions are finally drawn in section 6.

2 Preliminaries

A trapezoidal fuzzy number can be characterized as a 4-tuple of real numbers $\tilde{T_f} = (a, b, c, d), a < b < c < d$, shown in Fig. 1, where a and d represents the left hand and right hand width of the trapezoidal distribution, (b, c) denotes the interval in which the membership value is equal to 1 and $H_{\tilde{T_f}}(x)$ represents the membership function of $\tilde{T_f}$ defined as follows [14]:

$$H_{\tilde{T_f}}(x) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a < x \leq b \\ 1, & b < x \leq c \\ \frac{d-x}{d-c}, & c < x \leq d \\ 0, & x > d \end{cases} \tag{1}$$

Let $\tilde{A}$ and $\tilde{B}$ be two trapezoidal fuzzy numbers, $\tilde{A} = (a_1, b_1, c_1, d_1)$ and $\tilde{B} = (a_2, b_2, c_2, d_2)$. The arithmetic operations of $\tilde{A}$ and $\tilde{B}$ are defined as follows [15]:

1. Addition $\oplus$: $\tilde{A} \oplus \tilde{B} = (a_1, b_1, c_1, d_1) \oplus (a_2, b_2, c_2, d_2) = (a_1 + a_2, b_1 + b_2, c_1 + c_2, d_1 + d_2)$;
2. Subtraction $\ominus$: $\tilde{A} \ominus \tilde{B} = (a_1, b_1, c_1, d_1) \ominus (a_2, b_2, c_2, d_2) = (a_1 - a_2, b_1 - b_2, c_1 - c_2, d_1 - d_2)$;
3. Multiplication $\otimes$: $\tilde{A} \otimes \tilde{B} = (a_1, b_1, c_1, d_1) \otimes (a_2, b_2, c_2, d_2) = (a_1 \times a_2, b_1 \times b_2, c_1 \times c_2, d_1 \times d_2)$;
4. Division $\oslash$: $\tilde{A} \oslash \tilde{B} = (a_1, b_1, c_1, d_1) \oslash (a_2, b_2, c_2, d_2) = (a_1/a_2, b_1/b_2, c_1/c_2, d_1/d_2)$.

We define four logic operations, where $\tilde{A}$ and $\tilde{B}$ are trapezoidal fuzzy numbers, and a, b are real numbers:

1. *Minimum operator* $\wedge$: $a \wedge b = min(a, b)$;
2. *Maximum operator* $\vee$: $a \vee b = max(a, b)$;

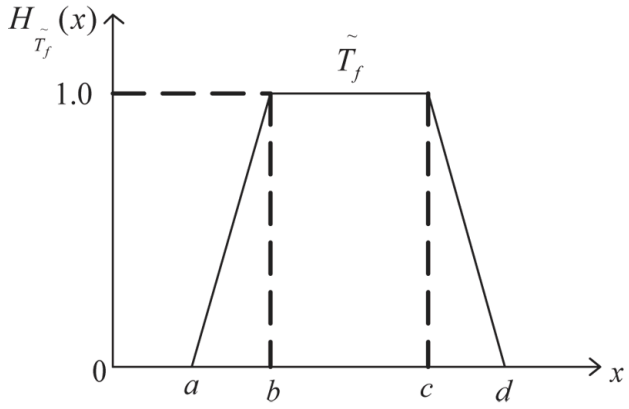


Figure 1: A trapezoidal fuzzy number.

3. *and* $\textcircled{\wedge}$: $\tilde{A} \textcircled{\wedge} \tilde{B} = (a_1, b_1, c_1, d_1) \textcircled{\wedge} (a_2, b_2, c_2, d_2) = ((a_1 \wedge a_2), (b_1 \wedge b_2), (c_1 \wedge c_2), (d_1 \wedge d_2))$;
4. *or* $\textcircled{\vee}$: $\tilde{A} \textcircled{\vee} \tilde{B} = (a_1, b_1, c_1, d_1) \textcircled{\vee} (a_2, b_2, c_2, d_2) = ((a_1 \vee a_2), (b_1 \vee b_2), (c_1 \vee c_2), (d_1 \vee d_2))$.

We define a scalar multiplication operation as follows:

Scalar Multiplication: $x \cdot \tilde{A} = x \cdot (a_1, b_1, c_1, d_1) = (x \cdot a_1, x \cdot b_1, x \cdot c_1, x \cdot d_1)$,

where A is a trapezoidal fuzzy number and x is a real number:

The defuzzification method in [16] is chosen to obtain a crisp number t_f associated with a trapezoidal fuzzy number $\tilde{T_f}$, it is shown in (2), where e and g are the extreme values of the whole fuzzy set range. In this study, e and g are equal to 0 and 1, respectively.

$$t_f = \frac{(d-e)+(c-e)}{((d-e)+(c-e))-((a-g)+(b-g))} \tag{2}$$

3 tFRSN P systems

A tFRSN P system of $m \geq 1$ is a construct $\Pi = (O, \sigma_1, \ldots, \sigma_m, syn, in, out)$, where:

(1) $O = \{a\}$ is a singleton alphabet (a is called spike);

(2) $\sigma_1, \ldots, \sigma_m$ are neurons, of the form $\sigma_i = (\theta_i, c_i, r_i), 1 \leq i \leq m$, where:

   (a) θ_i is a trapezoidal fuzzy number in [0,1] representing the potential value of spikes (i.e. value of electrical impulses) contained in neuron σ_i;

   (b) c_i is a trapezoidal fuzzy number in [0,1] representing the fuzzy truth value corresponding to neuron σ_i;

   (c) r_i represents a firing (spiking) rule contained in neuron σ_i with the form $E/a^{\theta} \rightarrow a^{\beta}$, where E is the firing condition and its form will be specified below, θ and β are trapezoidal fuzzy numbers in [0,1].

(3) $syn \subseteq \{1, 2, \ldots, m\} \times \{1, 2, \ldots, m\}$ with $i \neq j$ for all $(i, j) \in syn, 1 \leq i, j \leq m$, is a directed graph of synapses between the linked neurons;

(4) $in, out \subseteq \{1, 2, \ldots, m\}$ indicate the input neuron set and the output neuron set of Π, respectively.

In tFRSN P systems, the definition of neurons and pulse values can be extended. Specifically, in tFRSN P systems, the neurons are extended to four types, i.e., proposition neurons and three kinds of rule neurons: *general*, *and* and *or*, and the pulse value contained in each neuron is no longer the number of spikes represented by a real value, but a trapezoidal fuzzy number in [0, 1], which can be interpreted as the potential value of spikes contained in neuron σ_i. It is worth pointing out that the number of spikes in each neuron is determined by the problem to be solved and the pulse value contained in each neuron is different. If neuron σ_i contains no spike, then $\theta_i = 0$; otherwise, if neuron σ_i contains only one spike, then θ_i equals to the pulse value of this spike; in any other case, θ_i equals to the result of a operation on all pulse values received from its presynaptic neurons. For different types of neurons, the operations for pulse values are different. For *proposition* neurons and *and* rule neurons, they use operation $\textcircled{\wedge}$ to handle all the pulse values received from their presynaptic neurons while *or* rule neurons use operation $\textcircled{\vee}$, where symbols $\textcircled{\wedge}$ and $\textcircled{\vee}$ represent the *and* and *or* operators of trapezoidal fuzzy numbers, respectively. The firing condition $E = a^s$ means that the spiking rule, $E/a^\theta \to a^\beta$, contained in neuron σ_i, can be applied if and only if neuron σ_i contains at least s spikes, otherwise, the firing rule cannot be applied. More details about tFRSN P systems can be found to the preliminary work [9].

Fuzzy production rules consist of five types:

Type 1: $R_i(c_i) : p_j(\theta_j) \to p_k(\theta_k)$; $\theta_k = \theta_j \otimes c_i$.

Type 2: $R_i(c_i) : p_1(\theta_1)\textcircled{\wedge}\ldots\textcircled{\wedge}p_{k-1}(\theta_{k-1}) \to p_k(\theta_k)$; $\theta_k = (\theta_1\textcircled{\wedge}\ldots\textcircled{\wedge}\theta_{k-1}) \otimes c_i$.

Type 3: $R_i(c_i) : p_1(\theta_1) \to p_2(\theta_2)\textcircled{\wedge}\ldots\textcircled{\wedge}p_k(\theta_k)$; $\theta_2 = \ldots\theta_k = \theta_1 \otimes c_i$.

Type 4: $R_i(c_i) : p_1(\theta_1)\textcircled{\vee}\ldots\textcircled{\vee}p_{k-1}(\theta_{k-1}) \to p_k(\theta_k)$; $\theta_k = (\theta_1\textcircled{\vee}\ldots\textcircled{\vee}\theta_{k-1}) \otimes c_i$.

Type 5: $R_i(c_i) : p_1(\theta_1) \to p_2(\theta_2)\textcircled{\vee}\ldots\textcircled{\vee}p_k(\theta_k)$.

where R_i represents the ith fuzzy production rule; c_i is the certainty factor of rule R_i; p_i is a proposition appearing in the antecedent or consequence part of a rule, $1 \le i \le k$ (k is the number of propositions in a rule-based system); p_j in *Type 1* represents the jth proposition, $1 \le j \le k-1$; θ_i represents the fuzzy truth value corresponding to the ith proposition [15]. c_i and θ_i are trapezoidal fuzzy numbers defined in the universe of discourse [0, 1]. The causality between a fault on a faulty section in a power system and the status information about protective relays and circuit breakers (CBs) of this section can be described by the aforementioned fuzzy production rules. A simplified transmission network shown in Fig. 2 is used to illustrate the notations in a fuzzy production rule. According to the protection principle, if there is fault on transmission L, then its main protective relays, i.e., MLR_1 and MLR_2 and their corresponding CBs, i.e., CB_1 and CB_2, will operate to protect L, which can be backward described by a fuzzy production rule: $R(1,\ 1,\ 1,\ 1) : MLR_1\ operates\ (0.975,\ 0.98,\ 1,\ 1)\ \textcircled{\wedge}\ MLR_2\ operates\ (0.975,\ 0.98,\ 1,\ 1)\ \textcircled{\wedge}\ CB_1\ trips\ (0.975,\ 0.98,\ 1,\ 1)\ \textcircled{\wedge}\ CB_2\ trips\ (0.975,\ 0.98,\ 1,\ 1) \to L\ has\ a\ fault$. The certainty factor of this rule is (1, 1, 1, 1) which represents the contribution of this rule to the final diagnosis result. In this rule, there are four propositions $MLR_1\ operates$, $MLR_2\ operates$, $CB_1\ trips$ and $CB_2\ trips$, and they have an equal fuzzy truth value (0.975, 0.98, 1, 1) representing the contributions of the propositions to the result $L\ has\ a\ fault$.

Because rules *Type 5* are unsuitable for diagnosis, they are not further described in Section 3. tFRSN P system models for rules *Type 1* to *Type 4* are shown in Fig. 3.

To adapt tFRSN P systems to solve fault diagnosis problems, we describe MBFRA in the following description.

Given initial truth values of propositions corresponding to all input neurons in an tFRSN P system, MBFRA can perform fuzzy reasoning to obtain the fuzzy truth values of other neurons with unknown pulse values and output reasoning results. Let us assume that the tFRSN P system contains l proposition neurons and n rule neurons, each of which may be *general*, *and* or *or* rule neurons, $m = l + n$, where m is the number of all the neurons in this system.

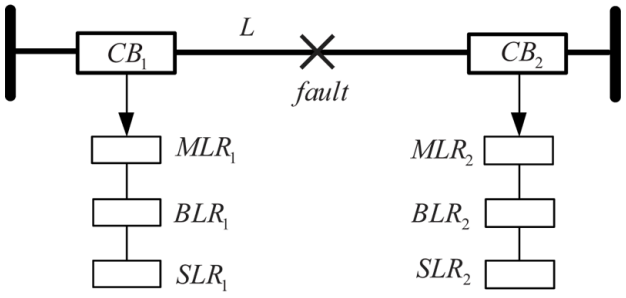


Figure 2: A simplified transmission network.

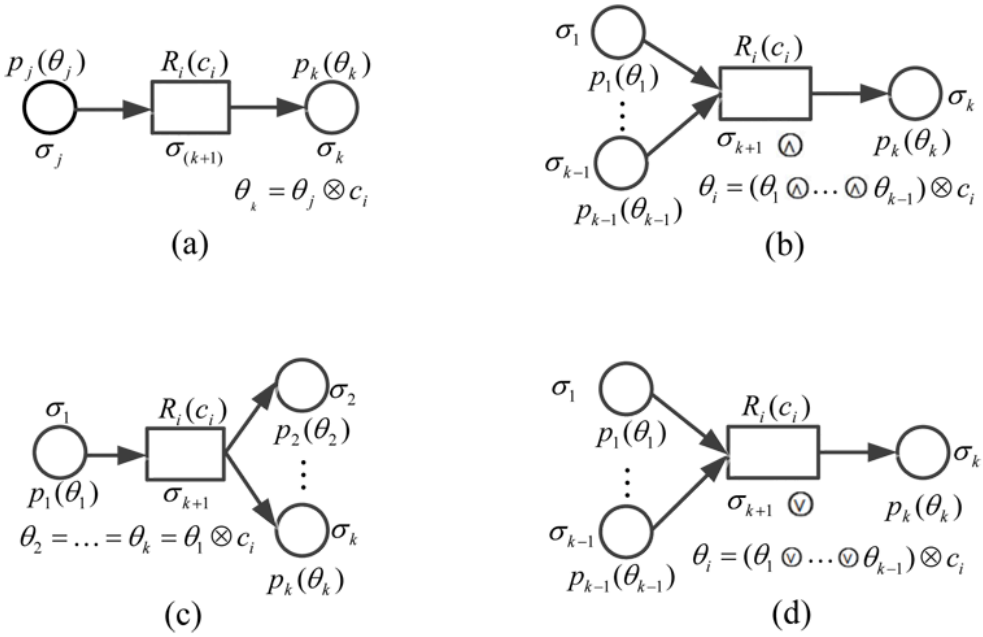


Figure 3: tFRSN P system models for fuzzy production rules. (a) *Type 1*; (b) *Type 2*; (c) *Type 3*; (d) *Type 4*.

In order to clearly present the reasoning algorithm, we first introduce some parameter vectors and matrices as follows.

1) $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_l)^T$ is a fuzzy truth value vector of the l proposition neurons, where θ_i represents the pulse value contained in the ith proposition neuron, $1 \leq i \leq l$, and is expressed by a trapezoidal fuzzy number in [0, 1]. If there is not any spike contained in a proposition neuron, its pulse value is "*unknown*" or $(0, 0, 0, 0)$.

2) $\boldsymbol{\delta} = (\delta_1, \delta_2, \ldots, \delta_n)^T$ is a fuzzy truth value vector of the rule neurons, where δ_j represents the pulse value contained in the jth rule neuron, $1 \leq j \leq n$, and it is expressed by a trapezoidal fuzzy number [0, 1]. If there is not any spike contained in a rule neuron, its pulse value is "*unknown*" or $(0, 0, 0, 0)$.

3) $\boldsymbol{C} = diag(c_1, c_2, \ldots, c_n)$ is a diagonal matrix, where c_j is the certainty factor of the jth fuzzy production rule, $1 \leq j \leq n$, and it is expressed by a trapezoidal fuzzy number.

4) $\boldsymbol{D}_1 = (d_{ij})_{l \times n}$ is a synaptic matrix representing the direct connection between proposition neurons and *general* rule neurons. If there is a directed arc (synapse) from the proposition neuron σ_i to the *general* rule neuron σ_j, then $d_{ij} = 1$, otherwise, $d_{ij} = 0$.

5) $\boldsymbol{D}_2 = (d_{ij})_{l\times n}$ is a synaptic matrix representing the direct connection between proposition neurons and *and* rule neurons. If there is a directed arc (synapse) from the proposition neuron σ_i to the *and* rule neuron σ_j, then $d_{ij} = 1$, otherwise, $d_{ij} = 0$.

6) $\boldsymbol{D}_3 = (d_{ij})_{l\times n}$ is a synaptic matrix representing the direct connection between proposition neurons and *or* rule neurons. If there is a directed arc (synapse) from the proposition neuron σ_i to the *or* rule neuron σ_j, then $d_{ij} = 1$, otherwise, $d_{ij} = 0$.

7) $\boldsymbol{E} = (e_{ji})_{n\times l}$ is a synaptic matrix representing the direct connection between rule neurons and proposition rule neurons. If there is a directed arc (synapse) from the rule neuron σ_j to the proposition neuron σ_i, then $e_{ji} = 1$, otherwise, $e_{ji} = 0$.

Subsequently, we introduce some multiplication operations as follows.

1) $\circledcirc$: $\boldsymbol{C} \circledcirc \boldsymbol{\delta} = (c_1 \otimes \delta_1, c_2 \otimes \delta_2, \ldots, c_n \otimes \delta_n)^T$; $\boldsymbol{D}^T \circledcirc \boldsymbol{\theta} = (\bar{d_1}, \bar{d_2}, \ldots, \bar{d_n})^T$, where $\bar{d_j} = d_{1j}\theta_1 + d_{2j}\theta_2 + \ldots + d_{lj}\theta_l$, $j = 1, 2, \ldots, n$.

2) $\odot$: $\boldsymbol{D}^T \odot \boldsymbol{\theta} = (\bar{d_1}, \bar{d_2}, \ldots, \bar{d_n})^T$, where $\bar{d_j} = d_{1j}\theta_1 \owedge d_{2j}\theta_2 \owedge \ldots \owedge d_{lj}\theta_l$, $j = 1, 2, \ldots, n$.

3) $\circledast$: $\boldsymbol{E}^T \circledast \boldsymbol{\delta} = (\bar{e_1}, \bar{e_2}, \ldots, \bar{e_l})^T$, where $\bar{e_i} = e_{1i}\delta_1 \owvee e_{2i}\delta_2 \owvee \ldots \owvee e_{ni}\delta_n, i = 1, 2, \ldots, l$.

Next, we list the pseudocode of MBFRA.

Algorithm MBFRA

Require: $\boldsymbol{D}_1$, $\boldsymbol{D}_2$,$\boldsymbol{D}_3$, $\mathbf{E}$, $\mathbf{C}$, $\boldsymbol{\theta}_0$, $\boldsymbol{\delta}_0$

Set the termination condition $\boldsymbol{0} = (unknown, unknown, \ldots, unknown)_n^T$
Let $t = 0$, where t represents the reasoning step
while $\boldsymbol{\delta}_t \neq \boldsymbol{0}$ **do**
  for each input neuron ($t = 0$) or each proposition neuron ($t > 0$) **do**
    if the firing condition $E = a^s$ is satisfied **then**
      the neuron fires and computes the fuzzy truth value vector $\boldsymbol{\delta}_{t+1}$ via $\boldsymbol{\delta}_{t+1}$=$(\mathbf{D}_1^T \circledcirc \boldsymbol{\theta}_t) \oplus (\mathbf{D}_2^T \odot \boldsymbol{\theta}_t) \oplus (\mathbf{D}_3^T \circledast \boldsymbol{\theta}_t)$
      if there is a postsynaptic rule neuron **then**
        the neuron transmits a spike to the next rule neuron
      else
        just accumulate the value in the neuron
      end if
    end if
  end for
  for each rule neuron **do**
    if the firing condition $E = a^s$ is satisfied **then**
      the rule neuron fires and computes the fuzzy truth value vector $\boldsymbol{\theta}_{t+1}$ via $\boldsymbol{\theta_{t+1}} = \mathbf{E}^T \circledast (\mathbf{C} \circledcirc \boldsymbol{\delta_{t+1}})$ and transmits a spike to the next proposition neuron
    end if
    $t = t + 1$
  end for
end while

Ensure: $\boldsymbol{\theta}_t$, which represents the final states of pulse values contained in proposition neurons.

4 Application Examples and Results

In this section, a power system with 14-buses, chosen from [17] and as shown in Fig. 4, is applied as an example to describe how to use tFRSN P systems with MBFRA to solve a fault diagnosis problem. The system contains 34 system sections, including 14 buses and 20 transmission lines. The buses are marked as B_{pq} and the transmission lines are represented

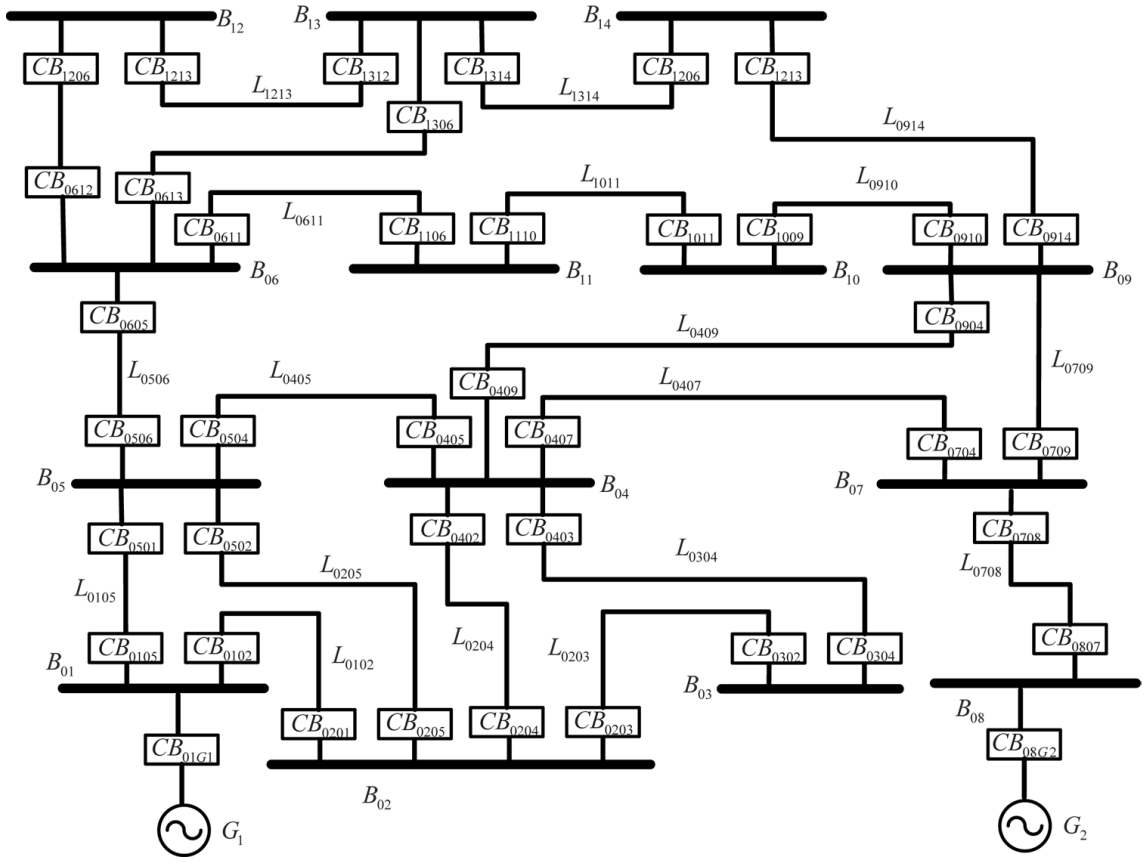


Figure 4: The power system with 14 buses.

as L_{pquv}, where $0 \leq p, q, u, v \leq 9$. The protection system of the 14-bus system contains 174 protective devices consisting of 40 circuit breakers (CBs), 40 main transmission line relays, 40 first backup transmission line relays, 40 second backup transmission line relays and 14 bus relays. A local part, which is composed of a transmission line L_{1314}, its adjoining two buses, B_{13} and B_{14}, and its adjoining three transmission lines, L_{1213}, L_{0613} and L_{0914}, of the protection system is given to describe its structure and symbols of protection devices. The local system is shown in Fig. 5. The operational rules of the protective devices are descried as follows [17].

The main transmission line relay MLR_{1314} protects the entire line L_{1314} and it will operate to trip its associated circuit breaker (CB), i.e., CB_{1314}, to clear a fault on the line L_{1314}. The bus relay BR_{13} protects the bus B_{13} and it will operate to trip the three CBs, i.e., CB_{1312}, CB_{1306} and CB_{1314}, if there is a fault on the bus B_{13}. The first backup transmission line relay BLR_{1314} is a local backup of the relay MLR_{1314} and has the same protection zone as MLR_{1314}. The relay BLR_{1314} will operate to trip CB_{1314} to clear a fault if the fault clearance by the relay MLR_{1314} fails. Secondary backup transmission line relays SLR_{1213} and SLR_{0613} are the remote backups of the relays MLR_{1314} and BLR_{1314}. They will operate to trip their corresponding CBs, i.e., CB_{1213} and CB_{0613}, respectively, to clear a fault if the fault clearance by both MLR_{1314} and BLR_{1314} fails. The relays SLR_{1213} and SLR_{0613} are also two remote backups of the relay BR_{13} and they will operate to trip CBs, i.e., CB_{1213} and CB_{0613}, respectively, to clear a fault if the fault clearance by the relay BR_{13} fails. The functions of the four relays, MLR_{1413}, BLR_{1413}, SLR_{0914} and BR_{14}, and three CBs, CB_{1413}, CB_{1409} and CB_{0914}, in the process of protecting the line L_{1314} and the bus B_{14} are similar and the protection systems for other sections in this 14-bus power system have the same protection rules, so it is not necessary to repeatedly describe

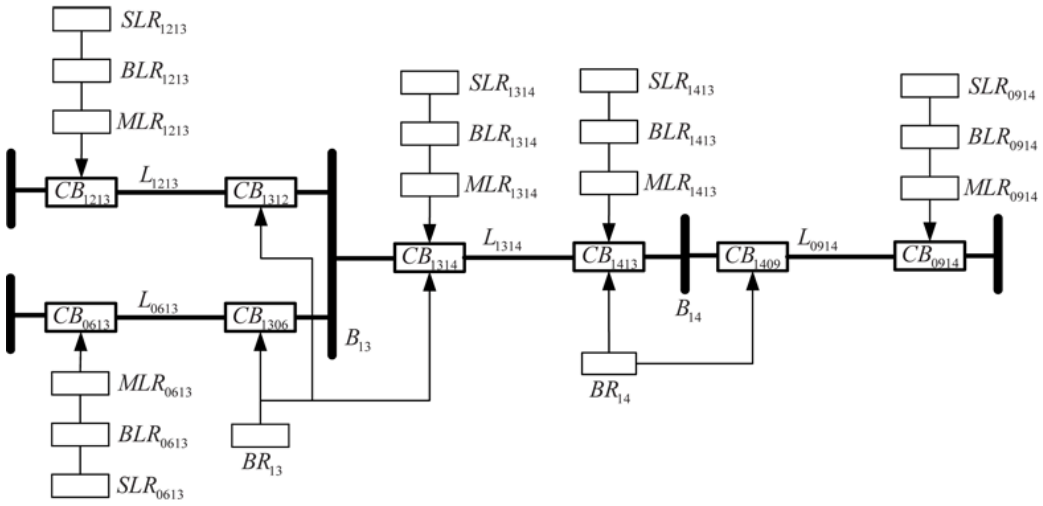


Figure 5: A local part of the protection system of the 14-bus power system.

their operation rules.

The protection rules described above show that when a fault occurs on a certain section of a power system, protection devices will reach certain statuses to protect the section. Meanwhile, the relay trip signals and CBs status signals, used as inputs of fault diagnosis models of sections, can be obtained from remote terminal units (RTUs) of supervisory control and data acquisition (SCADA) systems. The diagnostic strategy in this study is to build one tFRSN P system diagnosis model for each candidate fault section of a power system and each model performs MBFRA by using SCADA data, i.e., relay trip signals and CBs status signals, to get a trapezoidal fuzzy number which represents the fault confidence level of this section. In a single fault case, the section with the highest fault confidence level is the faulty section. In multiple faults cases, several sections with fault confidence levels which are greater than a threshold, which is set as real number 0.5 in this study, are regarded as faulty sections. Thus, to obtain real numbers for easily comparing the fault confidence levels with the threshold, a defuzzification method shown in (2) is used to process the reasoning results represented by trapezoidal fuzzy numbers. In addition, the fault confidence levels of faulty sections in multiple faults cases are ranked from high to low to help operators to decide a repair order of the sections.

Fig. 6 and Fig. 7 show the tFRSN P system diagnosis models for L_{1314} and B_{13}, respectively. It is worth noting that there are several assistant arcs (synapsises) with different arrow endings in the figures. For illustration purposes, we take arcs, from σ_2 to σ_{25} and from σ_2 to σ_{26}, as examples. The meanings of the two arcs are that if CB_{1314} opens, the operation of its corresponding second backup protective devices, including relays (SLR_{0613} and SLR_{1213}) and CBs (CB_{0613} and CB_{1213}), is invalid and then the values of these relays and CBs are set as $(0, 0, 0, 0)$; otherwise, the operation of the second backup protective devices is valid. In what follows we take transmission line L_{1314} as an example to show the fuzzy reasoning process of MBFRA based on tFRSN P systems.

Case 1: A single fault. Transmission line L_{1314} has a fault.

Operated relays: MLR_{1314}, MLR_{1413}, BLR_{1314}. Tripped CBs: CB_{1314}, CB_{1413}.

A tFRSN P system for L_{1314} is Π_1 and its corresponding tFRSN P system diagnosis model is shown in Fig. 6.

$$\Pi_1 = (O, \sigma_1, \sigma_2, \ldots, \sigma_{36}, syn, in, out)$$

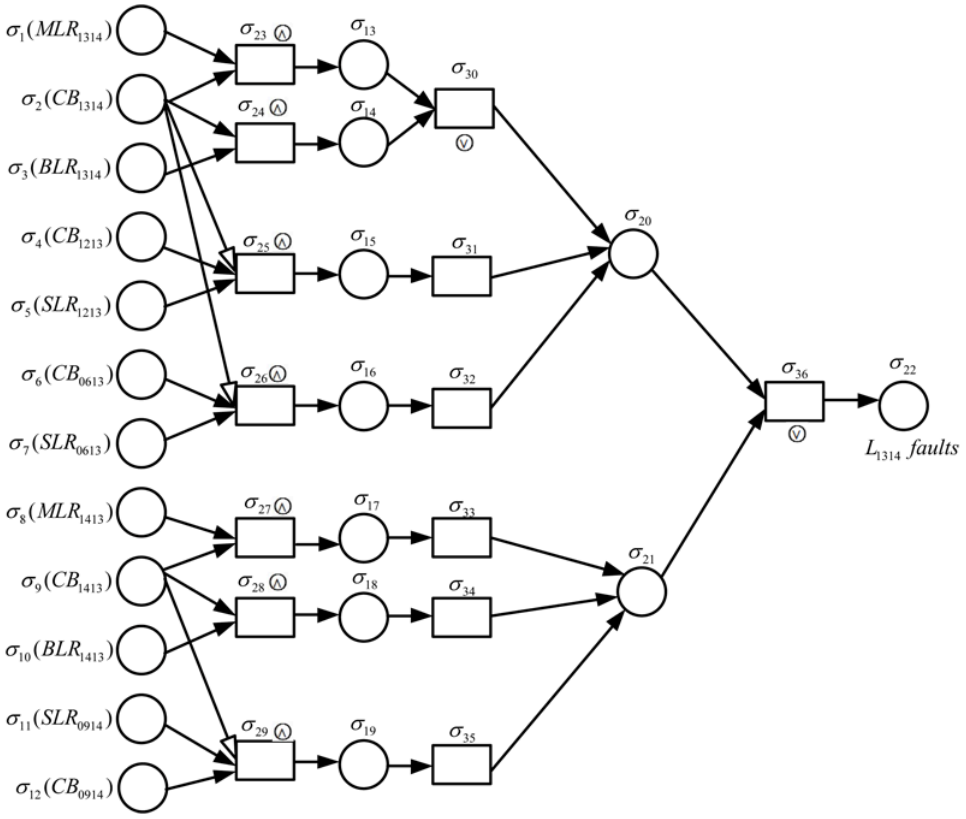


Figure 6: Fault diagnosis model of transmission line L_{1314} based on a tFRSN P system.

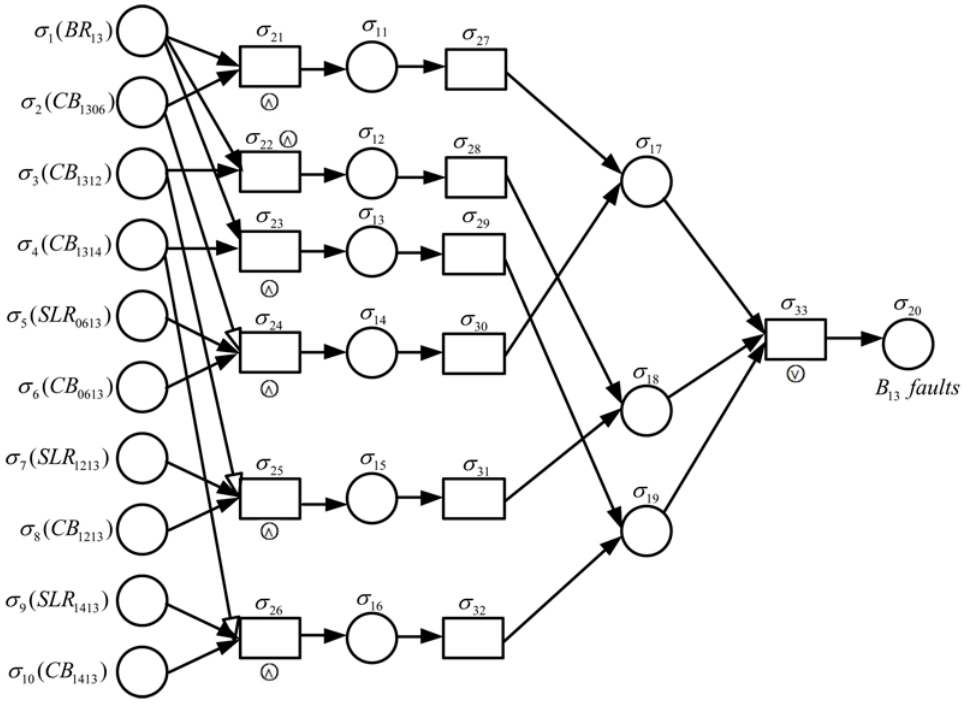


Figure 7: Fault diagnosis model of bus B_{13} based on a tFRSN P system.

Table 1: Linguistic terms and their corresponding trapezoidal fuzzy numbers

Linguistic Terms	Trapezoidal Fuzzy Numbers
absolutely-false (AF)	(0, 0, 0, 0)
very-low (VL)	(0, 0, 0.02, 0.07)
low (L)	(0.04, 0.1, 0.18, 0.23)
medium-low (ML)	(0.17, 0.22, 0.36, 0.42)
medium (M)	(0.32, 0.41, 0.58, 0.65)
medium-high (MH)	(0.58, 0.63, 0.80, 0.86)
high (H)	(0.72, 0.78, 0.92, 0.97)
very-high (VH)	(0.975, 0.98, 1, 1)
absolutely-high (AH)	(1, 1, 1, 1)

where

1) $O = \{a\}$ is the singleton alphabet (a is called spike).

2) $\sigma_1, \ldots, \sigma_{22}$ are proposition neurons corresponding to the propositions with fuzzy truth values $\theta_1, \ldots, \theta_{22}$; that is, $l = 22$.

3) $\sigma_{23}, \ldots, \sigma_{36}$ are rule neurons, where $\sigma_{23}, \ldots, \sigma_{29}$ are *and* rule neurons, σ_{30} and σ_{36} are *or* rule neurons and $\sigma_{31}, \ldots, \sigma_{35}$ are *general* rule neurons; that is, $n = 14$.

4) $syn = \{(1,23)\ ,(2,23),\ (2,24),\ (2,25),\ (2,26),\ (3,24),\ (4,25),\ (5,25),\ (6,26),\ (7,26),$ $(8,27),\ (9,27),\ (9,28),\ (9,29),\ (10,28),(11,29),(12,29),(13,30),(14,30),(15,31),\ (16,32),\ (17,33),$ $(18,34),\ (19,35),\ (20,36),\ (21,36),\ (23,13),\ (24,14),\ (25,15),\ (26,16),\ (27,17),(28,18),(29,19),(30,$ $20),\ (31,20),\ (32,20),\ (33,21),\ (34,\ 21),\ (35,\ 21),\ (36,22)\}$.

5) $in = \{\sigma_1, \sigma_2, \ldots, \sigma_{12}\}$, $out = \{\sigma_{22}\}$.

The knowledge of dispatchers in power systems may contain linguistic terms and the statuses of devices may have a certain degree of uncertainty. Table 1 shows an example of linguistic terms and their corresponding trapezoidal fuzzy numbers. In the tFRSN P system Π_1, input neurons $\sigma_1, \ldots, \sigma_{12}$ are assigned as the empirical values $VH, VH, H, AF, AF, AF, AF, VH, VH, L,$ AF, AF, respectively. Certainty factors corresponding to rule neurons $\sigma_{23}, \ldots, \sigma_{36}$ are given values $VH,\ VH,\ VH,\ H,\ VH,\ VH,\ H,\ VH,\ VH, VH, VH, VH, VH, VH$, respectively.

According to Table 1, we obtain the trapezoidal fuzzy numbers $\boldsymbol{\theta_0}$ and $\boldsymbol{\delta_0}$. In order to succinctly describe the matrices, let us denote $\boldsymbol{O_r} = (x_1, \ldots, x_r)^T$, where $x_i = (0,0,0,0), 1 \leq i \leq r$.

$$\boldsymbol{\theta}_0 = \begin{bmatrix} (0.975, 0.98, 1, 1) \\ (0.975, 0.98, 1, 1) \\ (0.72, 0.78, 0.92, 0.97) \\ \mathbf{O}_4 \\ (0.975, 0.98, 1, 1) \\ (0.975, 0.98, 1, 1) \\ (0.04, 0.1, 0.18, 0.23) \\ \mathbf{O}_{12} \end{bmatrix}_{22\times 1}, \quad \boldsymbol{\delta}_0 = \Big[\mathbf{O}\Big]_{14\times 1}$$

When $t = 0$, we get the results

$$\boldsymbol{\delta}_1 = \begin{bmatrix} (0.9506, 0.9604, 1, 1) \\ (0.702, 0.7644, 0.92, 0.97) \\ \mathbf{O}_2 \\ (0.9506, 0.9604, 1, 1) \\ (0.39, 0.098, 0.18, 0.23) \\ \mathbf{O}_8 \end{bmatrix}_{14\times 1}, \quad \boldsymbol{\theta}_1 = \begin{bmatrix} \mathbf{O}_{12} \\ (0.9268, 0.9412, 1, 1) \\ (0.6845, 0.7491, 0.92, 0.97) \\ \mathbf{O}_2 \\ (0.9268, 0.9412, 1, 1) \\ (0.2808, 0.0764, 0.1656, 0.2231) \\ \mathbf{O}_4 \end{bmatrix}_{22\times 1}$$

When $t = 1$, we obtain the results

$$\boldsymbol{\delta}_2 = \begin{bmatrix} \mathbf{O}_7 \\ (0.9268, 0.9412, 1, 1) \\ \mathbf{O}_2 \\ (0.9268, 0.9412, 1, 1) \\ (0.2808, 0.0764, 0.1656, 0.2231) \\ \mathbf{O}_2 \end{bmatrix}_{14\times 1}, \quad \boldsymbol{\theta}_2 = \begin{bmatrix} \mathbf{O}_{19} \\ (0.9268, 0.9412, 1, 1) \\ (0.9268, 0.9412, 1, 1) \\ \mathbf{O}_1 \end{bmatrix}_{22\times 1}$$

When $t = 2$, we have the results

$$\boldsymbol{\delta}_3 = \begin{bmatrix} \mathbf{O}_{13} \\ (0.9268, 0.9412, 1, 1) \end{bmatrix}_{14\times 1}, \quad \boldsymbol{\theta}_3 = \begin{bmatrix} \mathbf{O}_{21} \\ (0.9036, 0.9224, 1, 1) \end{bmatrix}_{22\times 1}$$

When $t = 3$, we get the results

$$\boldsymbol{\delta}_4 = \begin{bmatrix} \mathbf{O} \end{bmatrix}_{14\times 1}.$$

Thus, the termination condition is satisfied and the reasoning process ends. We obtain the reasoning results, i.e., the fuzzy truth value (0.9036,0.9224,1,1) of the output neuron σ_{22}. The transmission line L_{1314} is a faulty section with a confidence level (0.9036,0.9224,1,1).

tFRSN P systems and MBFRA are also suitable for multiple faults diagnosis problems in power systems. In what follows we take an example of the power system in Fig. 4 to show the effectiveness of the method in diagnosing multiple faults.

Case 2: Multiple faults. Transmission line L_{1314} and bus B_{13} have faults.

Operated relays: MLR_{1314}, MLR_{1413}, SLR_{0613}, SLR_{1213}. Tripped CBs: CB_{1314}, CB_{1413}, CB_{0613}, CB_{1213}.

According to the SCADA data, four candidate fault sections, i.e., L_{1314}, B_{13}, L_{0613} and L_{1213}, are selected. The tFRSN P systems of the four sections are established to perform MBFRA, respectively. After the fuzzy reasoning, fault confidence levels, (0.9036, 0.9224, 1, 1), (0.6673, 0.7341, 0.92, 0.97), (0.2165, 0.299, 0.623, 0.7849) and (0.2165, 0.299, 0.623, 0.7849), represented by trapezoidal fuzzy numbers of sections L_{1314}, B_{13}, L_{0613} and L_{1213} are obtained. According to (2), we obtain their corresponding real numbers, i.e., 0.92, 0.7595, 0.475 and 0.475. Thus, there are two faulty sections, i.e., L_{1314} and B_{13}. The results are summarized in Table 2.

The logic analysis about *Case 2* is described as follows. In this case, information of protective relay BR_{13} is not observed and CB_{1312} and CB_{1306} fail to trip. For transmission line L_{1314}, its main transmission relays, MLR_{1314} and MLR_{1413}, operate to trip their corresponding CBs,

Table 2: Relay trip signals and CBs status signals observed, and diagnosis results

Candidate fault section	Confidence level	Corresponding real number	Ranking	Fault section
L_{1314}	(0.9036,0.9224,1,1)	0.92	1	Yes
B_{13}	(0.6673,0.7341,0.92,0.97)	0.7595	2	Yes
L_{0613}	(0.2165,0.299,0.623,0.7849)	0.475	-	No
L_{1213}	(0.2165,0.299,0.623,0.7849)	0.475	-	No

CB_{1314} and CB_{1413}, to clear a fault. So it is a faulty section. Although main protective relay BR_{13} of bus B_{13} fails to clear a fault, its remote backup protective relays, SLR_{0613} and SLR_{1213}, operate to trip their corresponding CBs, CB_{0613} and CB_{1213}, to clear this fault. So B_{13} is a faulty section. For transmission lines L_{0613} and L_{1213}, only their single-ended remote backup protective relays SLR_{0613} and SLR_{1213} operate to trip their corresponding CBs, CB_{0613} and CB_{1213}, respectively. Actually, SLR_{0613} and SLR_{1213} and their CBs act as remote backup protections of B_{13}. So, L_{0613} and L_{1213} are not faulty sections. Therefore, according to the logic analysis and Table 2, we can know that the presented method can obtain correct results in multiple fault situations.

5 Discussions

A tFRSN P system is a novel graphical model for representing fuzzy knowledge and information. This study employs it to diagnose the faults of power systems. The fault diagnosis ability of a method is usually associated with the knowledge availability and the reasoning process. Thus, in what follows, we make a comparison between tFRSN P systems and several fault diagnosis approaches regarding the aspects of knowledge representation and inference process.

(1) Expert systems (ESs). Both ES and the fault diagnosis method based on tFRSN P systems (FDM-tFRSNP) can make full use of experts' knowledge. The differences are: an ES needs long response time and the maintenance of its knowledge base is difficult [18]; FDM-tFRSNP possesses parallel reasoning ability and adopt graphical knowledge representation and reasoning, which can avoid the main limitation of ES.

(2) Fuzzy set theory (FST). FST is an effective way to represent uncertain information but the definition of membership function is a hard job [18]. The FST-based method and FDM-tFRSNP both possess the ability to deal with uncertain information of protective devices. In addition, linguistic terms used in both methods make them closer to the human thinking compared with the methods using crisp numbers. The main differences between them are that FDM-tFRSNP has a fast reasoning speed and the matrix reasoning process is easier to describe diagnostic process as well as its programming.

(3) Artificial neural networks (ANNs). ANNs can be regarded as opaque black boxes and can be easily used. The main problems of ANNs lie in the difficult acquisition of a complete sample set and a tedious training process needing extra time consumption. In addition, premature convergence is also a problem. FDM-tFRSNP neither needs a training process with a set of comprehensive training data nor has a premature convergence problem [13]. Besides, FDM-tFRSNP can intuitively represent the relationships between faults and operations of protection devices. This feature is very helpful for operators to analyze and summarize failure processes.

(4) Fuzzy Petri nets (FPNs). Both FPNs and tFRSN P systems have graphical knowledge representation and parallel computing ability. However, the mechanism of tFRSN P systems is originated from neurophysiological behavior of neurons or/and living cells. Thus, the working principle of different types of neurons or/and cells may provide new inspirations for extending SN

P systems (or tFRSN P systems), which can increase the ways of knowledge representation and reasoning to solve new problems in power systems [4]. In addition, tFRSN P systems with trapezoidal fuzzy numbers have three rule neurons types, i.e., *general*, *and* and *or*, and one proposition neuron type, while FPNs only contain same places and transition types. Thus, different types of neurons make FDM-tFRSNP have better flexibility and trapezoidal fuzzy numbers (linguistic terms) make tFRSN P systems more understandable to operators of power systems.

6 Conclusions

In this study, tFRSN P systems and a matrix-based fuzzy reasoning algorithm, MBFRA, for fault diagnosis are discussed to extend the application area of SN P systems in fault diagnosis of power systems. MBFRA is based on the dynamic firing mechanism of neurons. Given initial pulse values of all input neurons of a tFRSN P system, MBFRA can obtain the pulse values of other neurons by performing fuzzy reasoning. To make MBFRA suitable for fault diagnosis in power systems, a defuzzification method is employed to treat reasoning results represented by trapezoidal fuzzy numbers. Application examples show that tFRSN P systems with MBFRA is effective in diagnosing faulty sections of power systems. Besides, a comparison between tFRSN P systems and different fault diagnosis approaches is made.

The aim of this study is to construct a tFRSN P system diagnosis model for each candidate fault section. The scale of each diagnosis model depends on protective devices connections of the candidate fault section rather than the scale of power systems. Thus, the presented method can be used for large-scale power systems. This study focuses on the effectiveness and correctness of the fault diagnosis method and the results of application examples are obtained by manual computation. To test the speed, convergence and accuracy of MBFRA and to explore automatical generation of tFRSN P systems in diagnosing faulty sections in power systems, our future work will simulate them on MATLAB, P-Lingua or MeCoSim [19]- [21]. Moreover, how to verify and realize the parallelism of tFRSN P systems and MBFRA on hardware such as FPGA and CUDA is also our further task.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (61170016, 61373047, 61170030), the Program for New Century Excellent Talents in University (NCET-11-0715) and SWJTU supported project (SWJTU12CX008).

Bibliography

[1] Păun, G. (2000); Computing with Membranes, *J Comput. Syst. Sci.*, ISSN 0022-0000, 61(1): 108–143.

[2] Ionescu, M.; Păun, G.; Yokomori T. (2006); Spiking Neural P Systems, *Fund. Inform.*, ISSN 0169-2968, 71(2-3): 279–308.

[3] Peng, H.; Wang, J.; Pérez-Jiménez, M.J.; Wang, H.; Shao, J.; Wang, T. (2013); Fuzzy Reasoning Spiking Neural P System for Fault Diagnosis, *Inform. Sciences*, ISSN 0020-0255, 235(20): 106–116.

[4] Wang, J.; Shi, P.; Peng, H.; Pérez-Jiménez, M.J.; Wang, T. (2013); Weighted Fuzzy Spiking Neural P Systems, *IEEE Trans. on Fuzzy Syst.*, ISSN 1063-6706, 21(2): 209–220.

[5] Păun, G.; Pérez-Jiménez, M.J.; Rozenberg, G. (2006); Spike Trains in Spiking Neural P Systems, *Int. J. Found. Comput. S.*, ISSN 0129-0541, 17(4): 975–1002.

[6] Cavaliere, M.; Ibarra, O.H.; Păun, G.; Egecioglu, O.; Ionescu, M.; Woodworth, S. (2009); Asynchronous Spiking Neural P Systems, *Theor. Comput. Sci.*, ISSN 0304-3975, 410(24-25): 2352–2364.

[7] Păun, G.; Rozenberg, G.; Salomaa A. (ads.)(2010); *The Oxford Handbook of Membrance Computing*, Oxford Unversity Press, New York.

[8] Pan, L.Q.; Zeng, X.X. (2011); Small Universal Spiking Neural P Systems Working in Exhaustive Mode, *IEEE Trans. on Nanobiosci.*, ISSN 1536-1241, 10(2): 99–105.

[9] Wang, T.; Wang, J.; Peng, H.; Wang, H. (2011); Knowledge Representation and Reasoning Based on FRSN P System, *In: Proc of the 9th World Congress on Intelligent Control and Automation*, pp. 849–854.

[10] Zhang, X.Y.; Luo, B.; Fang, X.Y.; Pan, L.Q. (2012); Sequential Spiking Neural P Systems with Exhaustive Use of Rules, *BioSystems*, ISSN 0303-2647, 108(1-3): 52–62.

[11] Francis G.C.; Henry N.A. (2012); On Structures and Behaviors of Spiking Neural P Systems and Petri Nets, *Lecture Notes in Computer Science*, ISSN 0302-9743, 7762: 145–160.

[12] Song, T.; Pan, L.Q.; Păun, Gh. (2013); Asynchronous Spiking Neural P Systems with Local Synchronization, *Inform. Sciences*, ISSN 0020-0255, 219: 197–207.

[13] Xiong, G.J.; Shi, D.Y.; Zhu, L.; Duan, X.Z. (2013); A New Approach to Fault Diagnosis of Power Systems Using Fuzzy Reasoning Spiking Neural P Systems, *Math. Probl. Eng.*, ISSN 1024-123X, vol. 2013: Article ID 815352, 13 pages.

[14] Chen, W.H. (2011); Fault Section Estimation Using Fuzzy Matrix-based Reasoning Methods, *IEEE Trans. on Power Deliver.*, ISSN 0885-8977, 26(1): 205–213.

[15] Chen, S M. (1996); A Fuzzy Reasoning Approach for Rule-based Systems Based on Fuzzy Logics, *IEEE Trans. Syst., Man, Cybern., Syst.*, ISSN 1083-4427, 26(5): 769–778.

[16] Liu, H.C.; Liu, L.; Lin, Q.L.; Liu, N. (2013); Knowledge Acquisition and Representation Using Fuzzy Evidential Reasoning and Dynamic Adaptive Fuzzy Petri Nets, *IEEE Trans. on Cybern.*, ISSN 1083-4419, 43(3): 1059–1072.

[17] Luo, X.; Kezunovic M. (2008); Implementing Fuzzy Reasoning Petri Nets for Fault Section Estimation, *IEEE Trans. on Power Syst.*, ISSN 0885-8950, 23(2): 676-685.

[18] Chen, W.H.; Tsai, S.H.; Lin, H.I. (2011); Fault Section Estimation for Power Networks Using Logic Cause-effect Models, *IEEE Trans. on Power Deliver.*, ISSN 0885-8977, 26(2): 963–971.

[19] The Matlab Website. http://www.mathworks.es/products/matlab/.

[20] Research Group on Natural Computing, University of Seville: The P-Lingua Website. http://www.p-lingua.org.

[21] Research Group on Natural Computing, University of Seville: The MeCoSim Website. http://www.p-lingua.org/mecosim.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 9(6):800-810, December, 2014.

Optimization Scheme of Forming Linear WSN for Safety Monitoring in Railway Transportation

N. Zhang, X. Zhang, H. Liu, D. Zhang

Ning Zhang*, Xuemei Zhang,
Haitao Liu, Dengfeng Zhang
School of Computer and Information Technology
Beijing Jiaotong University, Beijing ,China
Beijing Haidian District, ShangYuanCun Number 3
Beijing, 100044, China
nzhang1@bjtu.edu.cn, 12120477@bjtu.edu.cn,
10120486@bjtu.edu.cn, 11120502@bjtu.edu.cn
*Corresponding author: nzhang1@bjtu.edu.cn

Abstract: With the development of wireless sensing network, more and more applications have been deployed in the safety monitoring and in natural disasters prevention. The safety and disaster prevention systems have usually been laid out in linear network architecture, such as those of railway, motorway transportation and pipes. The background of the article is railway hazard goods transportation safety surveillance. The paper discusses about the network architecture of linear wireless sensor networks with multiple sink nodes, and proposes the grouping method of sink nodes and the formation scheme of networks. The scheme can re-establish a monitoring network when the train on the way is disconnected and re-grouped. The switching algorithm of group head nodes is put forward, so that the energy consumption of each node in the group is even. The optimal switching parameters for group head nodes are suggested by the simulation. Compared with the usual monitoring network, the method proposed enables the system life expectancy to prolong more than five times, and meets the monitoring requirements simultaneously.
Keywords: Wireless Sensor Network (WSN), linear network, strategy of forming the network, energy balance algorithms.

1 Introduction

The wireless sensor network has been more and more widely used in safety and disaster prevention area, because of its sensing capability in physical world, the adaptive capability and its flexibility in terms of deployment. In many sensor network applications, the energy consumption has become a limitation of long working hours of the system when performing the tasks like monitoring and data transmission, without outside electricity supply in the system. This is especially the case for the sink node working as gateway; the gateway not only needs to receive data from monitoring nodes but also needs to send data to the monitoring center through the internet [1]- [2]. These nodes have to consume a large amount of energy either in sending the date through the wireless mobile network or the satellite network. Because those nodes are essential for the whole network system, it has special significance to optimize the work of those nodes and to minimize their energy consumption [3].

It has been a major research topic to longer the lifetime of the wireless sensor networks and to lower the energy consumption whether from the aspect of sensing the physical world, the scheme of forming the network and data processing [4]. Because of the limitation imposed by the energy consumption, the computation resource and the storing capability and transmission capability are also limited. For the same reason, the gateways' storing and transmission capability are also limited. In today's researches of wireless sensor networks with multiple sink nodes, the

sensor network's coverage is usually a plane, which means the sensor nodes are located in a plane area [5]. Hence, most research interests are focused on the placement of sensor nodes, topology of the sensor networks and transmission strategies [6]- [7]. Hoping by placement of sink node in a reasonable way, the power consumption of the system will be lowered, the lifetime of the network will be increased. Those researches usually share an assumption that the sink node itself is a rich power node, hence the energy minimization and optimization of important sink node is less researched [8]. In many practical applications, sink node is usually the bottleneck of increasing the lifetime of the whole system. In some situation, when multiple sink nodes are presented in the system, the data exchange between sink nodes is also possible [9]- [10]. For instance, in the safety monitoring of railway goods transportation, the topology of sensor network is linear, the network in each section of the train send data to the sink node. The sink node of each section has the potential of exchange data. Other then that, same structure can also been seen in the monitoring system of fallen rocks on the sides of highway and railway slops [11]- [12].

2 Raise of the question

The hazard goods safety monitoring system based on wireless sensor network is shown in figure 1. The sensor nodes located in each section monitor the status of hazard goods, and send the data to the sink node through the sensor network. The sink node acts as gateway, as a result, it is called gateway node. On one hand, the gateway node manages the sensor network in the car, and collects the monitoring data. On the other hand, it sends the merged data to the ground control center through the mobile network. To realize the monitoring of hazard goods in the entire transporting process, the system requires the gateway nodes to send data constantly to the ground control center through mobile network or satellite network, so that a large amount of energy in those nodes is consumed. The wireless sensor networks work in the harsh environment and usually a long time. And each node has only one battery as its power supply. Hence, how to reduce the energy consumption of gateway nodes and longer their lifetime are vital to realize whole time the hazard goods transporting monitoring. Fig. 1 shows the situation described above.

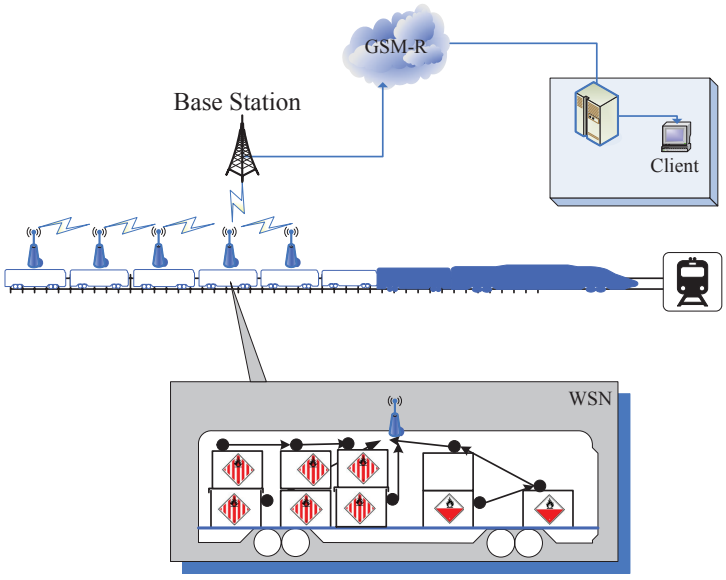


Figure 1: Hazard goods safety-monitoring system in railway transportation

In the situation showed in Fig. 1, if the gateway nodes send the data to the ground through mobile network or satellite network, it can guarantee that the information of each section of the train will be send to the ground timely. But it can easily cause the gateway nodes stop working by exhausting the power. Take GPRS (general packet radio service) as a way of sending the information, for instance, before gateway nodes send the data, they need to perform a series of test of parameters. It includes: signal strength of BCCH (Broadcast Control CHannel) or PDCH (Packet Data CHannel) received by the node, the variance of signal strength of downlink PDCH received signal, downlink PDCH signal strength, downlink PDCH signal interference and so on. On one hand, when sending the data, it takes a large amount of energy when establishing the connection and sending the data. On the other hand, it also takes a large amount of energy to measure the incoming and outgoing of message. Because the monitoring data is usually a small amount, comparing with pure data transmission, establishing connection and measuring process take a large proportion in the energy consumption. As a result, if multiple sink node can be grouped together and form a high network, only one node takes a task of sending data in a certain period of time. This will reduce multiply sink node GPRS measuring signal's sending and receiving; hence the total amount of energy consumption will be reduced. Besides this, because of the disconnection and regrouping of the train, the original network structure will be altered. As a result, it requires reforming the monitoring network with minimum energy consumption and satisfies the monitoring requirement.

This article focuses on the above situation, suggests the scheme of forming the linear wireless sensor network with sink node of each section of the train and the energy balance algorithm, so that those sink nodes energy can be shared to achieve lower energy consumption and longer the system lifetime. At the mean time, the scheme suggested can support the disconnection and regrouping of the train in the trip and reforming the monitoring network according to the regrouping of the train.

3 Reforming the network strategy

The normal trains car has length between 10 meter and 20 meter. The basic idea of forming the network can be seen as follows: connecting sink node in each car with the sink node in neighbor car/cars, so as to form new networks, and the monitoring data are sent to one of the sink nodes. Through that sink node, the data will be sending to the ground by mobile network. By applying reasonable method to balance the energy consumption, the lifetime of the network can be extended. For linear structure wireless sensor networks, it is very common that the multi-hop method is used to realize the monitoring data gathering. However, normal hazard transportation train contains more then ten cars, or even the cars of whole train. Due to the electrical and magnetic interference, if multi-hop method is applied to the whole train, the interference can easily affect the transmission, and causing high hop count and too much data stored on one node, in term causing the data package lost or over time. As a result, grouping cars so that hop count can be reduced, hence better transmission efficiency. Now we assume:

1) Each car has a sink node which is uniquely numbered, the node number is a value;

2) Each car has a sink node with same initial energy;

3) All sink node has the same distance with its neighboring sink node.

We take N_s as the sink node number; N_g as sink node number in each group, hence the group number: $N = N_s/N_g$, where N_g can be decided using the following method:

We assume the channel error as e_c, average data length as D_L(bit), sending data from arbitrary node to sink node has a successful rate no smaller than r_d, we can easily derive that the maximum hop count h_{MAX}:

$$h_{MAX} = \frac{\lg r_d}{\lg(1 - D_L e_c)} \tag{1}$$

Under the worst case, sink node as the border node of the group, and the node on the other side of the group takes $N_g - 1$ hop to reach sink node. Hence node number in each group N_g should be:

$$N_g \leq h_{MAX} + 1 \tag{2}$$

Grouping algorithm:

Idea of grouping algorithm: first decide two border nodes. Then take the both border nodes as starts respectively. Group every N_g node from the starts towards the middle nodes. The process is shown in Fig 2.

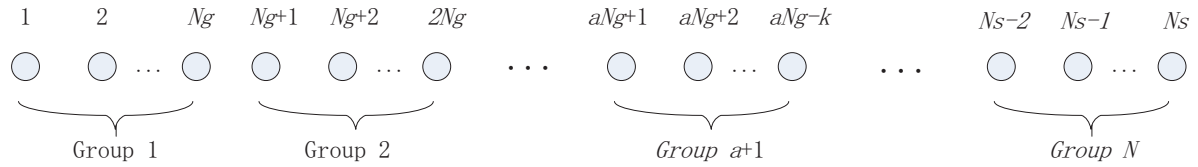


Figure 2: Nodes grouping

The detailed steps are as follows:

(1) Confirmation of the border node

1) After each node switched on and activated, a detection package is send with minimum power. At the meantime, the node listens to the neighboring node for response. If there are no responses, then increase the sending power by one level and send the detection package again. If response is received, goes to next step. If there is still no response even with maximum sending power, then end the process. It shows that two ends of the node have no monitored car, the node works alone.

2) Send the response package at once, when receive the detection package.

3) If two response packages are received, then the node is not a border node, and the node goes into idle.

4) If the node only receives one response package, it shows that this node is a border node.

(2) Algorithm of forming the network:

The grouping of the nodes starts from the border node. After the border nodes have confirmed their positions, it goes into forming network phase of the algorithm. In this phase, each node sends the message with small power so that it only covers the neighboring nodes (the power is determined by previous step). The detailed steps are as follows:

1) The border node send invitation package to neighboring node. The invitation package includes: group number (using the node number), member number (initialize to be 1) and so on;

2) When the node receives the invitation package, it firstly decides if itself has been given grope number. If so goes to step 4. Otherwise, increase the member number by 1, and set the group number to be received group number, and check if the member number is N_g. If the member is less than N_g then send invitation package with member number increase by 1. If the member number is bigger or equal to N_g then send the regroup package to the new neighboring node, and send confirmation package to its own group nodes.
3) When the regrouping package is received, the node checks itself if the group number is given yet. If the group number is given then no response is needed. The forming network is ended. Otherwise, the node creates a new invitation package. Within the invitation package, the nodes number serves as the new group number, the member number is 1. And then send invitation package to neighboring nodes.
4) When the node whose group number has been decided receives the invitation package, if the group number in the invitation package is the same with the nodes group number, then no response is needed. Otherwise, compare the sum of member number of the group and the member number of the invitation package, if the sum is less than N_g, then compare the group number. If the group number is smaller, then send invitation packages to node of the same group. Otherwise, replace the group number with new group number and send altered invitation package. If the sum of member number of the group and the member number in the invitation package is equal or larger than N_g, then responds with reject package, and send confirmation package to nodes of own group.
5) When invitation sender receives reject package, send confirmation package to own group and end the forming network.

Performing grouping based on the algorithm may create at most 2 groups with member number less than N_g . Each group works independently and that won't affect the whole system.

After the grouping process, each group needs to choose its own group head which transmits monitored signal to the surveillance center through mobile network. In the grouping process, the information of remaining energy of each node is exchanged. Hence, the node with maximum remaining energy can be picked as head. When more then one node has the maximum energy, the node closest to the middle will be picked as head.

The sink node of each car will send data to the group head through several hops while fulfilling the monitoring requirement. Group head will merge all the monitoring data and then send it to the ground surveillance center through mobile network. During the transportation, the train might go through multiple splitting and regrouping. If the car in same group is separated, and the head node has not received information from certain node sometimes, it assume those node has leave the group and the rest of the nodes will reform the network. On the other hand, if some node cannot send information to the group head, those nodes will reform the network according to the above method. In this process, due to the uniqueness of the node number, those group numbers will not be duplicated. The separated group will form two new group and two new group head.

4 Energy balance scheme

After grouping using the above algorithm, the head node needs to handle the transmitting information to the ground surveillance center through the mobile network and it has large energy consumption. Hence, a scheme is necessary so that the group head will be performed in turns

by certain rules. So the energy consumption of each node is balanced, and the lifetime of the system is maximized.

Scheme of group head rotation:

After the group head is firstly settled, the group head record the remained energy of itself, which is called the initial energy, denoted by P_γ. After each transmission of data, the head checks its remaining energy. When the energy drops to $\alpha\%$ $(0 \leq \alpha \leq 100)$ of P_γ , the head sends repacking message to the group nodes through broadcast. The repacking message includes node number, the remaining energy. The group nodes send its own remaining energy information to other nodes, record the group nodes' remaining energy information and compare the information with themselves. Finally, pick the node with largest remaining energy as new group head. When equal energy is seen, the system picks the node with smallest node number. The new group head generate process is shown in Fig. 3.

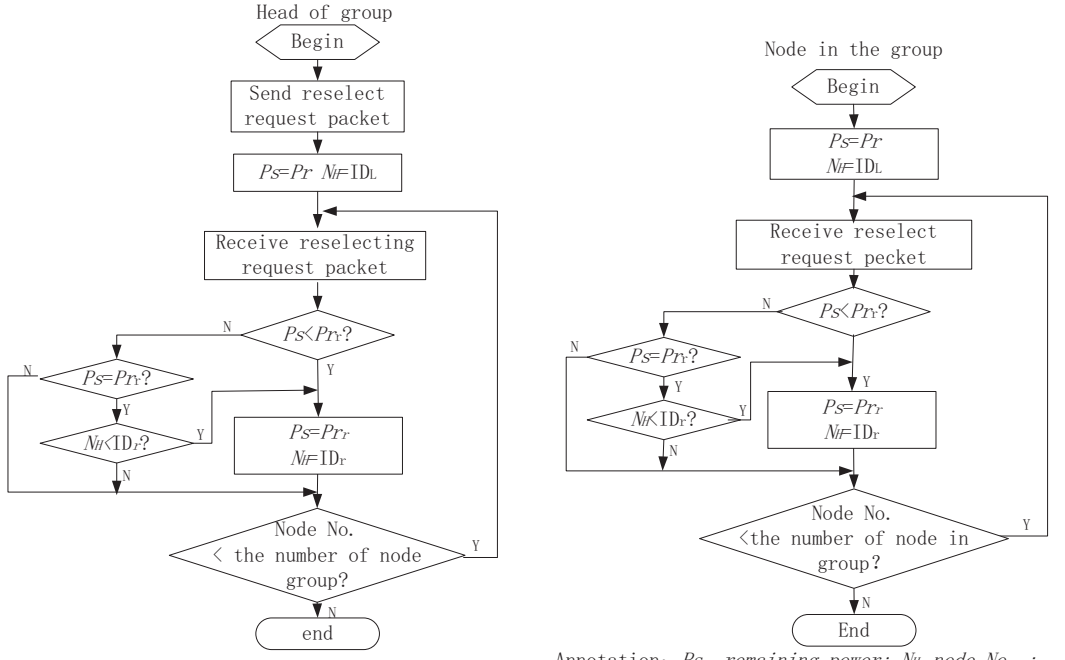


Figure 3: The new group head generation procedures

After the process shown above finished, all the node number stored in the node N_H are same, this number is the node number for the new group. This node is automatically the new group head; the new group will work around the head node. The node will record its remaining energy and when the energy drops to $\alpha\%$ of the remaining energy, the next rotation process is triggered. The whole network works as shown above and the energy of each node is balanced.

5 Performance measurement of the algorithm

Assume the minimum successful rate of send data from arbitrary node to sink node r_d as 0.9. The average data package length D_L is 200 bits, the channel error rate is 10^{-4} , by formulation(1)

we can get:

$$h_{MAX} = \frac{\lg r_d}{\lg(1 - D_L e_c)} = \frac{\lg 0.9}{\lg(1 - 200 \times 0.0001)} \approx 5 \tag{3}$$

By formulation(2) we can get:

$$N_g \leq h_{MAX} + 1 = 5 + 1 = 6 \tag{4}$$

Hence, under worst case, the data transmission success rate r_S :

$$r_S = (1 - D_L e_c)^h = (1 - 200 \times 0.0001)^5 \approx 0.904 > 0.9 \tag{5}$$

We can know that have 6 nodes in one group can satisfy the reliability requirement.

We assume the sum of time used between nodes' transmission t_S as 0.1s, the maximum system data transmission delay:

$$t_{MAX} = t_S \times h_{MAX} = 0.1 \times 5 = 0.5s \tag{6}$$

5.1 GPRS consumption analysis

To save energy, the GPRS module will work in an intermittent way, which means cut off the electricity right after the transmission. The voltage will be applied when the next transmission is ready. The time interval will be determined according to real monitoring requirement, usually ranges from several minutes to several tens of minutes, we pick 10 minutes in this case. According to real test, GPRS receiving and sending power $\bar{P}_{send}$ has its peak value around of 80mw. Taking the channel interference into account, the speed can be picked at 56kbps. We assume GPRS module takes about $t_{Link} = 20s$ to connect with system (including the GPRS ground system), the node monitoring data package has a average length of 200 bytes. The transmission time is:

$$t_{send} = \frac{1600}{56000} = 0.028s \tag{7}$$

After transmission, the node controls GPRS module to cutoff the electricity.

In this situation, each sink node takes more than $t_{Total} = t_{Link} + t_{send} = 20.028s$ to transmit data. The energy consumption of sending the data is:

$$A_{SUN} = \bar{P}_{send} \times t_{Total} = 0.08 \times 20.028 \approx 1.60448J \tag{8}$$

There is not data exchange between sink nodes; hence the energy consumption of data transmission is 0. We can know from this that the total energy consumption of one transmission of 6 sink nodes is:

$$A_{Total} = A_{SUN} \times 6 = 1.60448 \times 6 = 9.62688J \tag{9}$$

Using the method given in this paper, 6 nodes' information is sent by one node which acts as a gateway. This node takes same amount of time to send the data and establish connection, the data sending time is:

$$t_{SUN} = t_{send} \times 6 = 28 \times 6 = 168ms \tag{10}$$

Now,

$$t_{Total} = t_{Link} + t_{SUN} = 20 + 0.168 = 20.168s \tag{11}$$

5.2 Computation of energy consumption of data transmission between nodes

The data transmission between sink nodes uses zigbee protocols. Under the presumption of maintaining the channel transmission quality, system should send data with minimum power. Ordinary sensor node has a very small sending power and a high sending rate. Take CC2430 for instance; when it performs sending and receiving data, the typical working current is 27mA, the supply voltage is 3.3V, the sending rate can achieve 250kbps. If the average data is 200 bytes long, even under the worst case: it takes 15 hops for 6 nodes to send data to sink node. Hence, the total energy consumption is:

$$A_{ST} = 0.0891 \times 15 \times 1600 \div 250000 = 0.00855J \tag{12}$$

Hence,

$$A_{Total} = \bar{P}_{send} \times t_{Total} + A_{ST} = 0.08 \times 20.168 + 0.00855 \approx 1.623J \tag{13}$$

As we can see, every time those 6 nodes send their data, the energy consumption using the method of this paper is just 17% of the original method. The effect of saving the energy is significant. When the node is not sending data, the node is in sleep mode with minimum power consumption. That minimum energy consumption has little effect in analyzing the power consumption. Hence the above analysis ignores the consumption of sleeping node.

5.3 Energy balance simulation

Assume a set of 6 sink nodes uses 3.3V, 30mA battery as power supply. Each node has same initial remaining energy:

$$A = 3.3 \times 3600 \times 0.03 \approx 356J \tag{14}$$

From above computation: It takes about 1.6J of energy to send data to the ground surveillance center every time; each sending or receiving data between each node takes about 0.0086J. When there is no grouping, the system working cycle are shown in Fig. 4.

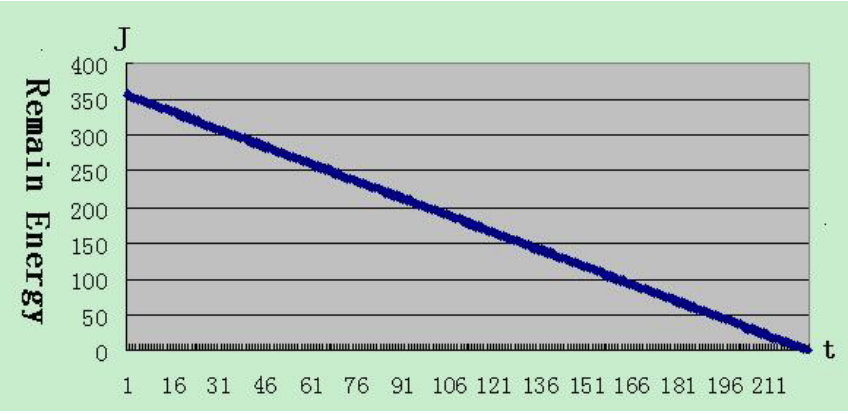


Figure 4: System lifetime when one node is working

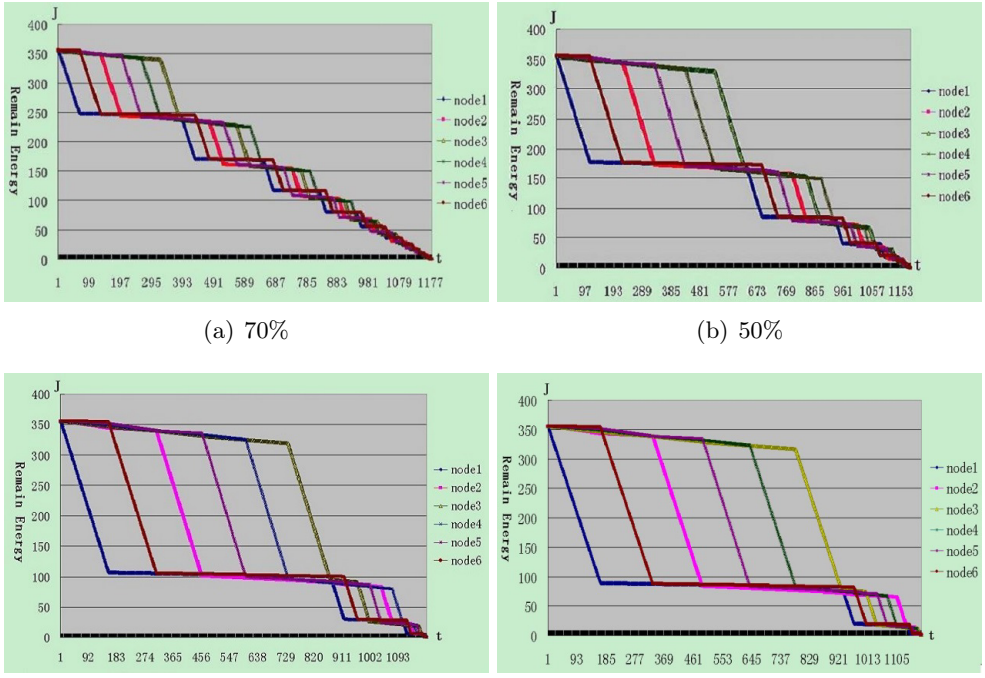
Under grouping situation, each node has different energy consumption when sending and receiving due to the difference of location of head node. The sending and receiving count of each node under different situations is given below in Table 1.

Node heads switching strategy is set as: when the remaining energy is 70%, 50%, 30% and 20% of the initial energy. When any nodes remaining energy is less then energy needed for one

Table 1: Count of each node under different situations

Node1	Node2	Node3	Node4	Node5	Node6
*5	9	7	5	3	1
1	*5	7	5	3	1
1	3	*5	5	3	1
1	3	5	*5	3	1
1	3	5	7	*5	1
1	3	5	7	9	*5

Note: * means the node is head node



(a) 70% (b) 50%

(c) 30% (d) 20%

Figure 5: System lifetime after grouping.

Table 2: The Simulation Result

Forming network	Switching strategy	Simulation steps	Remaining total energy J
Grouping (one group contains 6 nodes)	70%	1179	1.6
	50%	1182	2.74
	30%	1182	6.35
	20%	1182	7.38
Single Node	None	222	0.768

transmission, or the energy of every node is less than 1.6J, the system can no longer send the monitoring data, this ends the simulation. The simulation result is shown below both in Fig. 5 and in Table 2.

As it is shown in the simulation, the system lifetime after grouping is far better than the lifetime of single node mode (5.3 times over). After grouping, the steps are minimum when switching strategy is 70%. When the switching strategy is 50%, 30%, 20%, the step count is 1182, but the remaining energy is different, maximum at 20%, minimum at 50%. The reason is that the head node switching requires certain energy and in the whole working time, less switching time is healthy for the lifetime of the system.

6 Conclusion

Linear sensor network as a wide application, it especially has significant meaning to transportation safety monitoring. Due to the special node allocation, how to forming the network to maximize system lifetime is still a great challenge. This paper discussed about the question of sink node working together in a linear sensor network, and proposed a method to determine the sink node number in a group and network forming scheme. The experimental results and analysis can support the proposed method perfectly. The author hopes this can contribute to the optimization of this kind of sensor network.

Acknowledgment

The study is supported by National High Technology Research and Development Program of China (2008AA040207).

Bibliography

[1] Yu Gu; Yusheng Ji; Jie Li; Hongyang Chen; Baohua Zhao; Fengchun Liu. (2010); Towards an Optimal Sink Placement in Wireless Sensor Networks, *Communications (ICC), 2010 IEEE International Conference on*, ISSN 1550-3607: 1-5.

[2] Fengchao Chen; Ronglin Li. (2011); Single Sink Node Placement strategy in Wireless Sensor Networks, *Electric Information and Control Engineering (ICEICE), 2011 International Conference on*, ISBN 978-1-4244-8036-4: 1700-1703.

[3] Rossi, L.A.; Krishnamachari, B.; Kuo, C.-C.J.. (2007); Optimal Sink Deployment for Distributed Sensing of Spatially Nonstationary Phenomena, *GLOBECOM 2007 - IEEE Global Telecommunications Conference*, ISBN 978-1-4244-1042-2: 1124 - 1128.

[4] Shucheng Dai; Changjie Tang; Shaojie Qiao; Kaikuo Xu; Hongjun Li; Jun Zhu. (2010); Optimal Multiple Sink Nodes Deployment in Wireless Sensor Networks based on Gene Expression Programming, *Communication Software and Networks, 2010. ICCSN '10. Second International Conference on*, ISBN 978-1-4244-5726-7: 355-359.

[5] Flathagen, J.; Kure; Engelstad, P.E. (2011); Constrained-based Multiple Sink Placement for Wireless Sensor Networks, *Mobile Adhoc and Sensor Systems (MASS), 2011 IEEE 8th International Conference on*, ISSN 2155-6806: 783-788.

[6] Jaewan Kim; Sungchang Lee. (2009); Spanning Tree Based Topology Configuration for Multiple-Sink Wireless Sensor Networks, *Ubiquitous and Future Networks, 2009. ICUFN 2009. First International Conference on*, ISBN 978-1-4244-4215-7: 122-125.

[7] Vincze, Zoltan; Vida, Rolland; Vidacs, Attila. (2007); Deploying Multiple Sinks in Multi-hop Wireless Sensor Networks, *Pervasive Services, IEEE International Conference on*, ISBN 1-4244-1325-7: 55-63.

[8] Rui Teng; Bing Zhang. (2010); Distribution of Sink-Node's Operation for On-Demand Information Retrieval in Wireless Sensor Networks, *Wireless Advanced (WiAD), 2010 6th Conference on*, ISBN 978-1-4244-7069-3: 1-6.

[9] Xuguang Sun; Jingsha He; Yunli Chen; Shunan Ma; Zhen Zhang. (2011); A New Routing Algorithm for Linear Wireless Sensor Networks, *Pervasive Computing and Applications (ICPCA), 2011 6th International Conference on*, ISBN 978-1-4577-0209-9: 497-501.

[10] Ping Dong; Suixiang Gao. (2008); Adjustment of Transmission Radius in Linear Wireless Sensor Networks, *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on*, ISBN 978-1-4244-2107-7: 1-4.

[11] Malik, N.N.N.A.; Esa, M.; Yusof, S.K.S.; Hamzah, S.A. (2010); Optimization of Linear Sensor Node Array for Wireless Sensor Networks Using Particle Swarm Optimization, *Microwave Conference Proceedings (APMC), 2010 Asia-Pacific*, ISBN 978-1-4244-7590-2: 1316 - 1319.

[12] Gernot Fabeck; Rudolf Mathar. (2010); Optimization of Linear Wireless Sensor Networks for Serial Distributed Detection Applications, *Optimization of Linear Wireless Sensor Networks for Serial Distributed Detection Applications*, ISSN 1550-2252: 1-5.

Author index