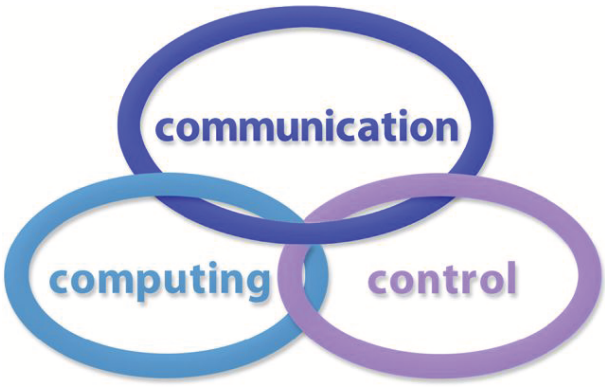
INTERNATIONAL JOURNAL

of

COMPUTERS COMMUNICATIONS & CONTROL

ISSN 1841-9836



A Bimonthly Journal
With Emphasis on the Integration of Three Technologies

Year: 2016 Volume: 11 Issue: 4 Month: August

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

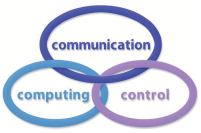
CCC Publications - Agora University

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.
ISO: Int. J. Comput. Commun. Control; **JCR Abbrev:** 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 Misu-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.

  2016 Journal Citation Reports® Science Edition (Thomson Reuters, 2016):
  Subject Category: (1) Automation & Control Systems: Q4(2009,2011,2012,2013,2014,2015), Q3(2010); (2) Computer Science, Information Systems: Q4(2009,2010,2011,2012,2015), Q3(2013,2014).

  Impact Factor/3 years in JCR: 0.373(2009), 0.650 (2010), 0.438(2011); 0.441(2012), 0.694(2013), 0.746(2014), 0.627(2015).
  Impact Factor/5 years in JCR: 0.436(2012), 0.622(2013), 0.739(2014), 0.635(2015).

- Since 2008 IJCCC is indexed by Scopus (SNIP2014= 1.029):
  Subject Category: (1) Computational Theory and Mathematics: Q4(2009,2010,2012,2015), Q3(2011,2013,2014); (2) Computer Networks and Communications: Q4(2009), Q3(2010, 2012, 2013, 2015), Q2(2011, 2014); (3) Computer Science Applications: Q4(2009), Q3(2010, 2011, 2012, 2013, 2014, 2015).

  SJR: 0.178(2009), 0.339(2010), 0.369(2011), 0.292(2012), 0.378(2013), 0.420(2014), 0.319(2015).

- Since 2007, 2(1), IJCCC is indexed 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: (1) Integrated solutions in computer-based control and communications; (2) Computational intelligence methods (with particular emphasis on fuzzy logic-based methods, ANN, evolutionary computing, collective/swarm intelligence); (3) 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 11, Issue 4, August, 2016.

Contents

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(4):457-471, August 2016.

DTN Routing Algorithm for Networks with Nodes Social Behavior

A.M. Dziekonski, R.O. Schoeneich

Andrzej Marek Dziekonski, Radoslaw Olgierd Schoeneich*
Institute of Telecommunications, Warsaw University of Technology
ul. Nowowiejska 15/19, Warsaw, 00-665, Poland
a.m.dziekonski@stud.elka.pw.edu.pl
*Corresponding author: rschoeneich@tele.pw.edu.pl

Abstract: This article presents routing algorithm in Delay and Disruptive Tolerant Networks (DTN). The main idea of this work is routing method that is based on information about nodes social behavior and their social relations in sparse structure of network. The algorithm takes advantage of friendship relationships between nodes and uses historic information to create groups of friends for each node, which is used in buffer management and forwarding phase of routing. Beside the routing method, mechanisms of collecting and exchanging of maintenance information between nodes is described. The algorithm was tested using The ONE simulation tool especially designed for DTN scenario and compared with miscellaneous popular solutions.
Keywords: DTN, routing algorithm, social behavior.

1 Introduction

Communication has always been important part of different communities lives, both short and long distance. One of dynamically developing wireless networks are ad-hoc networks, that create connections directly between networks nodes - Mobile Ad-hoc Networks (MANET) [1]. This is a type of network, in which its nodes are only users and also they are the only elements required for networks existence and operating. One kind from MANET networks family are Delay Tolerant Networks (DTN).

DTN [2] is a kind of wireless networks that enables communication in sparse and disrupted mobile ad-hoc networks. There are not any central or privileged elements in the means of network operations. Depending on networks purpose, there might be nodes collecting data from others, or controlling them, but this is ensured simply by distribution of proper messages, not by other networks mechanisms. DTN network is generally very dynamic - its elements can freely move, so constant connections between nodes exist very seldom. Also periods between subsequent nodes contacts is not deterministic and usually is very long.

One of the essential parts of telecommunication, and also very important in DTN networks, is routing. Routing is the decision making process of finding the best paths that messages should follow to reach destination in networks. Without this mechanism nodes would not know which messages should be passed to which nodes to provide good network operation. In DTN networks routing is especially important, because contacts between nodes are rare and short lasting, so every opportunity should be perfectly used. Unfortunately, for the same reason, it is more complex than in traditional networks, where all connections are known for long periods and it is easier to find proper paths for messages.

Nodes in DTN networks are devices that move during network operation. Some nodes might be means of public transport like communication equipment connected to buses or trams, smartphones, small devices connected to animals etc. - there is no limitation of DTN application. Still the most interesting and with the biggest possible usage are networks based on smartphones or other equipment held by people. That is why there is a big need for development of solutions

directed for networks with human mobility patterns, since people movement and behavior is not random, but more-or-less predictable.

2 State of the art

Routing protocols for DTN networks are specific solutions due to the necessity of dealing with complex requirements set by the network conditions, mostly in the means of disruption and long message delivery latencies. That is the reason why traditional algorithms from common MANET networks, like [3]: Ad-hoc On-Demand Distance Vector (AODV) or Dynamic Source Routing (DSR) cannot be used.

The easiest solution for delivering messages in DTN networks is Epidemic routing algorithm [4]. It floods the network with data without making decisions based on any criteria. Node is transmitting all of the messages from his buffer to any met node. The order of choosing messages is random. According to the fact, that it creates many copies of the same message in order to distribute them further, this algorithm belongs to the routing solutions group based on message replication. It is a very good reference point while comparing other solutions efficiency, because it chooses randomly proxy nodes and uses all available network resources. In perfect conditions, that are unlimited buffer capacities and infinite connection bandwidth, Epidemic routing acquire biggest possible number of successfully delivered messages to the destination node.

The next solution from replication-based types of routing algorithms is Spray and Wait [5]. It works much alike Epidemic routing, but it does not flood the network with unlimited amount of data. Each message can be replicated only in specified number of copies, which are transferred to other nodes. After creating and transmission of the last copy of selected message, all of the copies are kept in the buffers and transferred only while meeting the message destination node. Spray and Wait can work in two modes - normal and binary. In normal mode only the node that created the message is distributing copies to other nodes, which later can transfer them only directly to the destination node. In binary mode the node transmitting the message passes half of its lasted copies to the other node, and all the nodes that keep in their buffers more than one copy can pass them to any node until they have only one copy left.

Prophet algorithm [6] in contrast to Spray and Wait and Epidemic solutions is based on predictions of future nodes contacts. This is done using historic information collected during existence of the network. The main part of this algorithm is calculation of probability of meeting each of the other nodes from the network. In the first phase it calculates the new probability for the node that currently the connection is held. Second phase is actualization of the transitive probabilities, that is probability with the use of a proxy node (non-direct probability). Prophet solution implements also a mechanism that is prioritizing newest contacts before oldest ones. Nodes that are more active in the network, that establish more connections, have higher probabilities and higher priority in other nodes routing data collections. This algorithm creates many copies of the same message, theoretically infinite - there are no strict limits, but the network is not flooded with messages to the full extent - during contact only those messages are exchanged, that have higher probability of reaching destination while held by the node on the other end of the connection.

The algorithm that is best fitted for DTN networks based on complex, natural mobility models is MaxProp [7], despite the fact that it was developed for vehicle-based networks. In this solution each node keeps information about the network as a graph, which edges weights are calculated probabilities of the contact two chosen nodes. Then algorithm looks for the paths and makes decisions regarding order of the messages in the buffer. The creators did an observation based on many simulations, that transmission of newer messages, so the ones which passed by fewer nodes, in the first place can increase efficiency of routing. That is why in Maxprop algorithm

they developed a mechanism, that messages having hop count less than some calculated border value are transmitted in the first place, and those which passed by more nodes are ordered based on the probabilities calculated with the use of Dijkstra algorithm. One of the important elements of the solution is dealing with successfully delivered messages, that enables clearing nodes buffers from those messages and prevents the unneeded its further exchange.

Many research teams focused on analyzing human mobile patterns for the needs of telecommunication [8–12]. Therefore there are also few algorithms that were developed for networks which nodes are behaving like people. One of the algorithms of this type is Label Routing [13]. It is based on an assumption that people belong to some communities, so they meet with some of the nodes regularly. Authors developed a solution that create a label for each node that tells other nodes about his community. When message is supposed to be forwarded, it is passed directly to the destination node or to the node which is in the same community (has the same label) as the destination node. The problem is that messages are exchanged only if labels of potential next-hop and destination nodes match. In other cases messages are stored and waiting for a proper possibility to forward the message. This increases delivery delay and in some cases, for example with low and restricted node mobility, can have a big influence on routing efficiency.

Another interesting solution focused on social-based DTN networks is Bubble Rap Forwarding [14]. This algorithm takes advantage of two social characteristics - community and centrality. They assume that nodes belong to different communities and are active at different level. The algorithm allows each node to belong to more than one group - one group can be family, another colleagues at work, another friends from high school etc. In each of those groups nodes are prioritized by calculation of centrality of the node among others. This value is kind of popularity of the node and shows which ones have the highest probability to meet all others from that group. Since this value is calculated inside single group, each node has many values of centrality - one for each community to which it belongs. This parameter helps to forward message when it is already held by a node from the message destination node community. Another problem is to exchange message between different groups. For this purpose another centrality is calculated - global one, that takes into account whole network as one group. When new message is created, the first phase is forwarding the message to the destination node community using the nodes global centrality values and then the second phase is exchanging message using local centralities from current group.

Social Based Multicasting [15] is a multicast approach, that is creating many copies of the same massage, to DTN routing that uses two social characteristics - centrality and community. In this solution, the centrality of the nodes and the cumulative probability of future contacts is calculated based on Poisson modeling of social networks. Then it uses unified knapsack problem to select relay nodes to assure proper delivery ratio. The main issue is the computational complexity.

On the other hand, the authors of SANE routing [16] focus on completely different social characteristics - interest and similarity. This solution comes from the assumption, that people with similar interests meet each other more often, than the ones that have nothing in common. The interests are represented as a vector, and messages are forwarded to nodes which interest profiles are close to the destination node one.

There are few other algorithms that focus on nodes social behavior and its application in order to develop an efficient routing [17–19]. Most of them focus on three social characteristics: community, centrality, friendship [20]. First of those assume that nodes can be divided into groups (communities) that has many regular contacts between each other and that will continue in further network operation. Thanks to that algorithms can divide its operation into two phases - between communities and inside destination group. Examples of solutions using community characteristic are described above Label and Bubble Rap, and also Friendship Based Routing

[21]. Second of frequently used social characteristics is node centrality, which is a measure of activity and importance of the selected node. This helps to differ nodes inside the network and find most important nodes in the means of potential better forwarding efficiency. The main algorithms taking advantage of centrality is described above Bubble Rap as well as SimBet [22] algorithms. The last social characteristic is friendship, which is measure of the relationship between two nodes. The number of contacts, the periods between consecutive connections, duration of contacts etc. - all of those parameters that can help calculate the grade of relation between two nodes can be taken into account and considered as friendship measure. The example of use of this characteristic is Friendship Based Routing algorithm.

3 The social based algorithm

This routing algorithm is based on information of nodes social behavior collected during network existence. Beside the main algorithm, also mechanisms of collecting and exchanging control information are important for making proper routing decisions.

The main assumptions while developing the new solution were that the algorithm should be aimed for networks with human-mobility patterns, but still work well in other types of networks. It should predict future nodes behavior based on historic data collected during network operation. Since in that type of networks resources should not be a big problem - smartphones and similar devices have large amounts of memory and communication interfaces with fast transmission speed - and the main goal is maximizing the number of delivered messages, while minimizing the delivery latency, the proposed algorithm is a full-replication-based type, so there is no limits of created messages copies. Secondary goal is also to minimize the length of the paths that messages follow, since it is proven that it helps the network to operate more efficiently [23].

3.1 Information collected and exchanged for routing purposes

The key element of the solution are historic information regarding the network needed for making routing decisions. This data can be collected by the nodes themselves, or received from other nodes. Each node holds two information regarding whole network. (a.) First of them is maximum popularity value from all network nodes. It is used in mechanism that orders messages to be transfered to other nodes. (b.) Second stored data is the time of last reset of nodes popularity values. This helps to determine time, when table aging mechanism, that is the decrease of number of nodes contacts and groups popularities, should be launched.

For proper network operation it is needed to handle successfully delivered messages. For this purposes each node stores list of already delivered messages and updates and sends it during each new contact. After such an update node deletes from its buffer all messages that identification numbers are included in the received list. Essential for making routing decisions are two tables. First of them stores number of contacts with each node from the network. Based on it, the node decides which nodes belong to its friends group. The table is updated after each contact by increasing the number of connections with currently met node. Then the border value for belonging to nodes friends group is calculated. Identification numbers of the nodes that exceeds this parameter, so the ones that are chosen to friends group, and the popularity of the group, which is defined as calculated border value, are passed to each met node. Second essential information is table containing definitions of other nodes friends groups. For each node, with which at least once connection was established, the list of its friends identification numbers and the group popularity value is stored. This data is used in buffer management mechanism and during making decisions related to transferring messages to other nodes.

Except mentioned above, nodes store also information about the time of the last contact and the average period between those contacts for each node from the network. We use that data as one criteria in memory buffer management.

Nodes gets maintenance information in two ways. Some are collected by the nodes themselves during network operation - for example connection times with other nodes, number of meetings etc. Other information are collected from met nodes.

Due to limited node communication capabilities it is very important to assure effective and compact way of exchanging maintenance data. After establishing a new connection first and significant for solution effectiveness element of the algorithm is the exchange of successfully delivered messages identification numbers. This data is exchanged in both ways. According to them, nodes update their lists and clears their buffers from appropriate messages. Next the information about friends groups are exchanged. After update and rebuilding of nodes friends groups, each of the sides of the connection passes its friends identification numbers and popularity of his group to other node. This helps reduce amount of sent data in this phase of the algorithm. The node that receive this data refresh in its memory information about friends of the node on the other end of the contact. The last collected data from other nodes is information about most active router in the network - hub node. The activeness is measured as the number of contacts. Identification number of the node is not exchanged, only the value of the popularity of this biggest hub node.

3.2 Memory buffer management

After establishing a connection between two nodes, exchanging and processing maintenance information, the solution proceeds to the key phase of the algorithm - making routing decisions. It is divided into two stages. The first one concerns the management of buffer, the actions that need to be done when the memory is overloading. The second is directly connected with routing - it makes decisions in what order messages should be transmitted to connected nodes. This order has essential significance because of limited contact times and bandwidth speed.

The first stage is memory management. Nodes have limited space in buffers to store messages. It is highly probable situation, when a new message is created or received from other node, but there is no free space in the buffer. In this case the algorithm is launching decision-making procedure to free some space in the memory by choosing older message to be deleted. In the solution there is separate algorithm to compare messages for buffer management purposes. As a result of its operation all messages are sorted and placed in order to be removed, to free enough space for new message. It is done by directly comparing two messages and deciding which of them is less important and should be deleted before the second one. This algorithm is presented on the diagram below. In the buffer management decision-making process there are few criteria implemented. First of them, while choosing messages to delete, is the number of nodes that message already went through - number of hops. In case of its different values, the one which went through more nodes is chosen to be deleted before the other one. When compared messages have the same hop count, the algorithm checks predicted time of next contact with destination node. Prediction is based on historic information about the time of last contact and average period between them. This predicted time is not very precise, but taking into account non-deterministic nodes movement and the need to minimize resources, that are memory and computation time, it is sufficient to decide which messages should have higher priority to stay in memory. Additionally, to compensate inaccuracy, that difference of two predicted contact times has to exceed some border value. In other case the algorithm passes to the last criteria, which is comparison of paths through friends groups. In the case of memory management the distance is calculated and the message which distance is longer is chosen to be deleted. There is no need

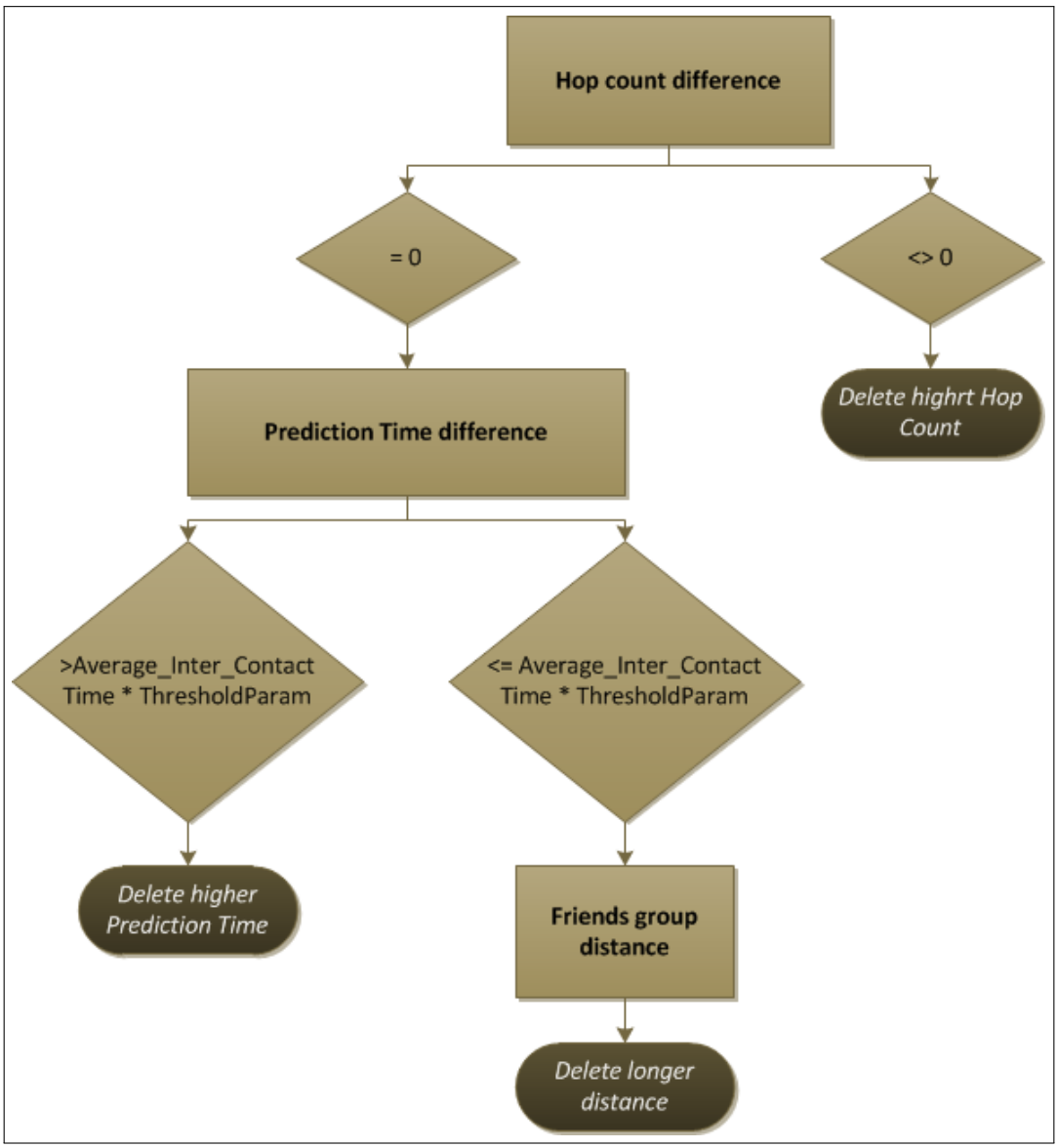


Figure 1: Algorithm to sort messages for buffer-management purposes.

to add any extra criteria - if all of described above steps do not differentiate priority of staying in buffer for two messages, it can be assumed that they are equally important.

3.3 Data exchange

The main part of developed routing solution is the way of passing messages to other nodes - methods of making the decisions about the order of messages to be exchanged with the nodes with whom connection is currently established. The primary assumption in passing the messages is the routing algorithm type, that is replication-based what means that many copies of the same message are created. While spreading many copies it is essential to give messages priorities, that tell which ones should be transmitted in the first place. For this purposes the sorting algorithm is developed.

Characteristic distinction between sorting method in this stage of routing algorithm compared to the one in buffer management are the items being sorted. In this case not only messages are ordered, but pairs of message-established connection. It helps better estimate probability of delivering data to destination, depending on which node will be next hop on for the message. Similarly to buffer management stage, in this phase also few criteria are implemented in order

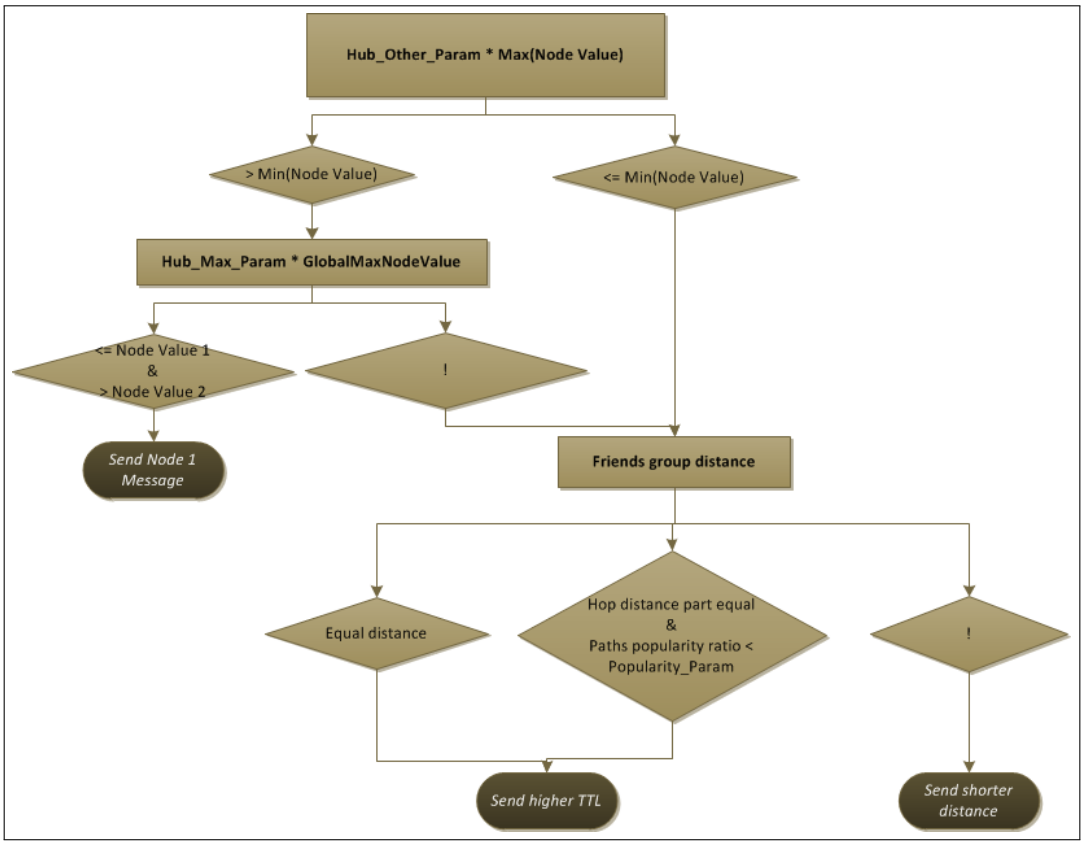


Figure 2: Algorithm to sort pairs messages-connection for data exchange purposes.

to sort elements. First of the criteria is detection, if the node being compared is a hub or not. Before checking that, to eliminate situations when both compared nodes are hubs or just the difference between them is very insignificant, the ratio of popularities of both nodes is calculated (bigger value to smaller) and it has to be bigger than border value set as a solution parameter.

In case, when this ratio is bigger than the border value, both of the compared nodes are checked if they can be classified as hubs. If only one is successfully classified, his pair, which is

the node and compared with it message, is receiving higher priority in sorting process. In other case, when both or none of them are classified, the algorithm advances to next criteria.

The next criteria in routing part of the algorithm is the most important one and makes decisions in most cases - distance based on friends groups. The main goal of the solution is working in networks, which nodes are personal equipment held by people, so the mobility is not random. People belong to different communities, have their own friends with whom they meet most often etc. Those characteristics are mostly constant or at least long lasting. The algorithm draws conclusions based on observations made during network existence - nodes spending most of the time in some location are establishing connections with chosen group of nodes. Additionally for each node those groups are different - family members spend a lot of time together, but they work in different places, take different routes using public transport or car, have different friends. That is why the creation of common groups is very complex and inefficient. In the solution each node create his own independent friends group and pass its description, containing friends identification numbers and group popularity, to other network elements. Group popularity is calculated based on the number of connections required by nodes to belong to the friends group.

Calculation of the distance based on collection of friends groups is done by looking for a path through those groups to the destination node. In data exchange stage of the solution, the node that currently has an established connection is considered next hop for the message and based on its friends group next nodes on the message path are searched. After finding a path, the number of proxy groups is considered the calculated distance value. Furthermore friends groups has different importance - active nodes has higher border value that recognizes network elements as its friends, so they are more active and this suggest that this group should have higher priority. Popularities of the groups on the predicted path are not added, but averaged geometrically to prevent situations that would preference of the paths with bottleneck group inside, so one group with very small popularity that could decrease efficiency of the path. Still the length of the path calculated just as number of proxy groups is more important. That is why the popularity of the path is considered only when the length of compared paths is equal.

There is an additional limitation of maximum path length, that the search process is finished after reaching that value and if path to destination is not found, the distance is set to infinity. This limitation was introduced for two reasons - (a) to decrease computational complexity, usage of node resources and time needed to find paths, and (b) to introduce next criteria to compare pair message-neighbor node that can have better efficiency than comparison of very long potential message paths. This maximum path length is set as a variable parameter to the algorithm.

If any of mentioned above criteria do not resolve which compared pair should have priority, the algorithm passes to final criteria - comparison of TTL (Time To Live) parameter of the message. Justification is that messages that will not expire in short future have better probability of reaching the destination. It is worth noticing, that in this criteria only messages are compared - the connections in corresponding pairs are not influencing the result.

4 Simulations

Simulations were performed to check the efficiency of the solution using The ONE Simulator [24] - a tool developed on Aalto University in Helsinki. It is used for examination of DTN networks operations, mainly for checking network efficiency with the use of different routing algorithms and different nodes mobility patterns. This tool has many advantages that simplify simulations - good programming interfaces to easily add new algorithms, GUI that helps finding problems during network operation and implementation of few mobility patterns and most popular existing routing algorithms, i.e. MaxProp, PRoPHET, Spray&Wait, Epidemic.

Table 1: Values for parameters set for the simulations

Parameter name	*Value*	*Parameter name*	*Value*
Popularity	1,25	Hub Other	0,85
Hop Limit	4	Reset Seconds	21600
Border Value	1,4	Reset Divisor	2,0
Hub Max	0,95	Predict. Time Threshold	1,25

The efficiency of routing algorithms was determined by analyzing few parameters. The main ones are:

- *Delivered messages* - defined as ratio of number of successfully delivered messages to all generated messages.
- *Message delivery latency* - average time between generation of the message and time of its delivery to destination node. Only successfully delivered messages are taken into calculation. Dropped, expired or non-delivered are omitted.
- *Message hop count* - the number of nodes that message went through before reaching destination node. Rejected, expired and non-delivered messages are not included in calculation.
- *Message buffer time* - average time of holding each copy of the message in nodes buffers.
- *Overhead ratio* - represents the level of flooding network with messages. It is calculated as a ratio of number of all copies of all messages exchanged by nodes, reduced by number of delivered messages, to number of delivered messages. In case of Direct Delivery algorithm (that passes meessags only when nodes meet destination node) overhead ratio reaches its minimum value - 0.

4.1 Comparison with chosen existing algorithms

In order to compare efficiency of developed solution few simulations were performed, that differs mostly in used mobility patterns and simulation duration. They were executed using five different routing algorithms - proposed new solution, MaxProp, PRoPHET, Spray and Wait and Epidemic. To compare general efficiency of the new algorithm, values of its parameters that have influence on algorithm operation were set top-down, without looking for their optimal values, the same for all simulation scenarios. This way comparison with other solutions is fair and objective. They are presented in Table 1.

The values of the parameters were chosen to be universal and work well in different network types. We consider a node to be a hub if its popularity is at least 95% of the largest encountered popularity in the whole network (Hub Max parameter). This is important only if the difference between two nodes being compared in the message exchange process is larger than 15% (Hub Other parameter). Also each node consider other nodes to be their friends if they meet them at least 40% more often than the average number of all its encounters (Border Value parameter). The aging of the number of encounters mechanism is set to be executed each 6 hours (Reset Seconds parameter) and is done by dividing the number by 2 (Reset Divisor parameter). Maximum length of the path through friends groups is set to be 4 (Hop Limit parameter) and calculated difference in paths popularities must be at least 25% (Popularity parameter). The difference of predicted time of the next encounters for two pairs of nodes must be larger than 25% (Pred. Time Threshold parameter).

The two node mobility patterns that were used are Shortest-Path Map-Based Movement (SPMBM) and Working Day Movement (WDM). The second one is a pattern that simulates in the closest way the behavior of people during normal days of their life, so the one that algorithm was aimed for.

Delivery ratio

Delivery ratio, that is number of messages that were successfully delivered to destination, is the most important criteria while comparing routing algorithms. On the figure 3 the results of

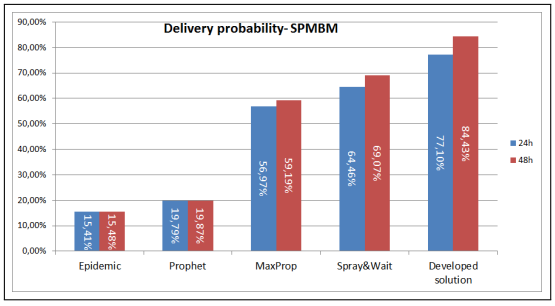


Figure 3: Delivery ratio for Shortest Path Map Based Movement scenario.

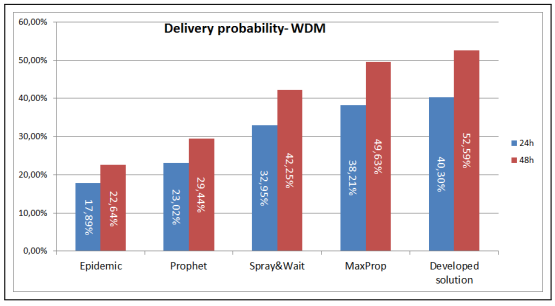


Figure 4: Delivery ratio for Working Day Movement scenario.

simulations for SPMBM mobility pattern scenario are presented. The developed solution reaches the best results from all tested algorithms, regardless of the simulation period. The important observation is that the longer simulation lasts, the greater increase in delivery probability this algorithm reaches from all others. Also it is easy to observe, that the difference between the best and the worst, which is Epidemic routing, is substantial.

On the figure 4 the results for WDM scenario are shown. In this case also the best delivery results are reached by described in the article new solution, but in this case the difference is not that significant. The important thing to observe is that in this scenario, so the one with human mobility patterns, with the longer simulation time the results are improving notably. It shows that nodes are making better routing decisions when they have more data collected, so they learn during network operation.

Delivery latency

The second most important criteria while checking efficiency of routing algorithms is delivery latency, because in many situations it is important to deliver messages quick, in other case they can be useless. In the SPMBM scenario (Fig. 5) the new solution gets the longest average time

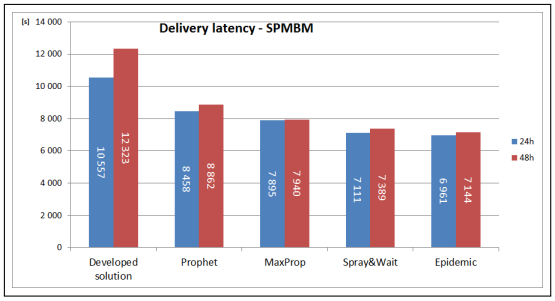


Figure 5: Mean delivery latency for Shortest Path Map Based Movement scenario.

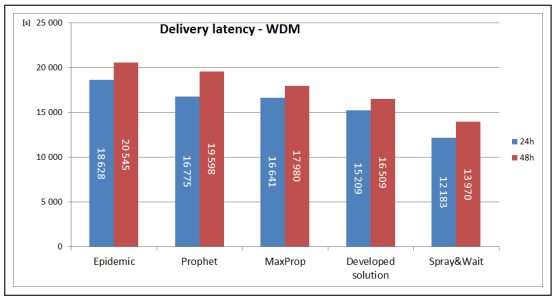


Figure 6: Mean delivery latency for Working Day Movement scenario.

for the messages to reach destination. It is not expected result, since one of the goals was to minimize latency. But it is important to look at this result together with the acquired delivery

ratios. In other case the conclusion would be that the Epidemic routing is the most efficient, since the delivery latency is the smallest. The developed solution has big latency values, because it successfully delivered messages, that other algorithms dropped. This messages often take a lot of time to reach destination, so it overstates the measured average delivery latency.

In WDM scenario the results look even better, than in SPMBM. Despite the fact, that new developed solution has the best delivery ratio, the average latency has quite similar values that all other algorithms. This means that not only it delivers more messages than other algorithms, but also average time of the same messages delivered by different algorithms is smaller for described in the article solution.

Average message hop count

The figures 7 and 8 present the results of the average number of proxy nodes on the path (hops)

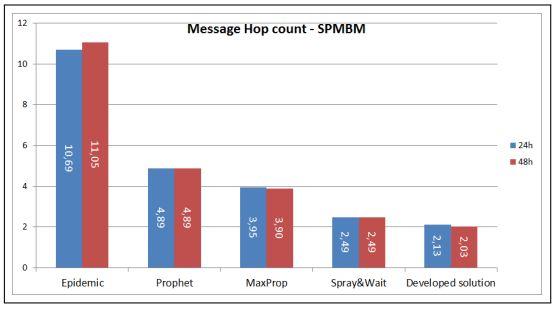


Figure 7: Mean message hop count for Shortest Path Map Based Movement scenario.

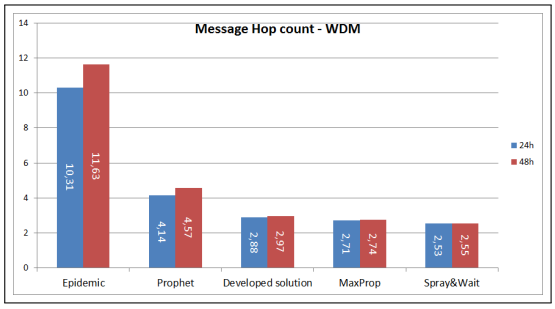


Figure 8: Mean message hop count for Working Day Movement scenario.

that messages followed before reaching destination. In networks using Epidemic routing the messages passed many nodes before getting to the destination node. This is caused by the fact, that there is no decision-making process and messages are transferred in random order. The best results for WDM scenario, so the ones when messages go through a small number of nodes, are reached by new developed solution, MaxProp and Spray&Wait. They are very close to each other, so the conclusion is that is almost optimal value. In SPMBM scenario also described in this article algorithm and Spray&Wait allow messages to reach destination in shortest paths calculated in number of proxy nodes.

4.2 Different network conditions

Each algorithm works the best in certain environment. Different conditions in network can have significant influence on the obtained results. Even if all of the parameters of the network and its elements change the way that network operates, there are few characteristics that have the biggest influence. The main are mobility patterns, that were described in previous chapter, number of hosts, transmission speed and range and buffer sizes.

Number of nodes in DTN network

We performed the simulation with the use of two algorithms - the new described in the article and Spray&Wait - to have a reference point to existing solutions. The results are presented on the figure 9. In the small network (100 nodes) both algorithms obtain the same delivery ratio, but Spray&Wait need less time for that. With the increase of the network size, the developed algorithm is gaining much better results. The delivery ratio is increasing more dynamically and saturates at higher level than the other algorithm. The delivery latency is also changing in the good way - Spray&Wait needs more and more time to deliver messages with the growth of the size of network, while new solution is obtaining smaller delivery latency in bigger networks. It

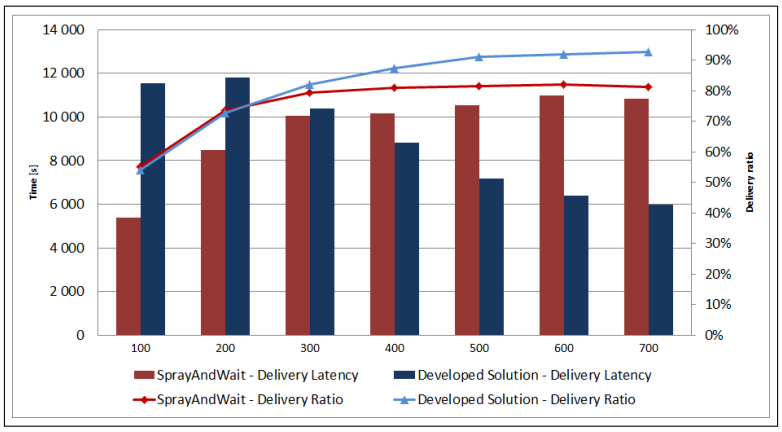


Figure 9: Impact of number of hosts in network.

shows that the algorithm described in the article works best in big networks, but still get good results in smaller ones.

Transmission speed and range

On the figure 10 and 11 the influence of changes in transmission are shown - change in transmis-

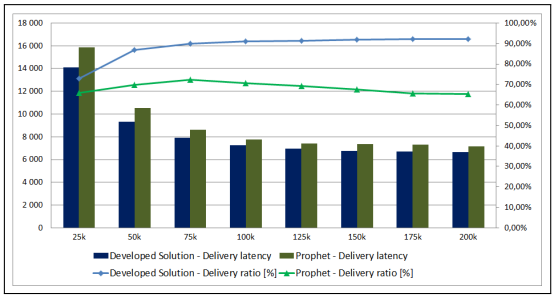


Figure 10: Impact of transmission speed.

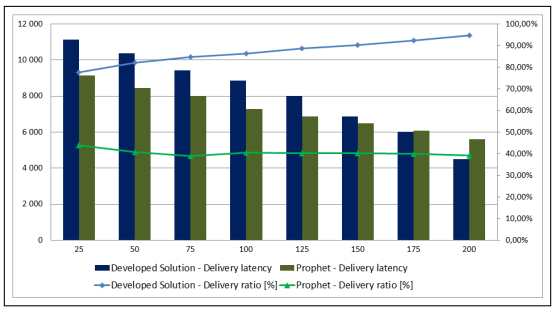


Figure 11: Impact of transmission range.

sion speed, that allows to exchange more data during contact, and change in transmission range, that causes more contacts between nodes. It is clear that with the growth of these parameters, the conditions are becoming better, so the algorithms should work more efficiently. The charts show that new developed solution works exactly as expected - with the growth of transmission speed or range, the delivery ratio is increasing and the latency is decreasing. The improvement in the routing efficiency is better in the new developed solution in comparison to Prophet algorithm.

Buffer size

The last examined network nodes characteristic that has big influence on the results is the buffer size. The increase of this parameters allows routers to hold more messages without the need to drop some of them. The figure 12 shows that the described in the article algorithm is more efficient when buffers are smaller and reaches its efficiency maximum for the much smaller buffer size than Prophet algorithm. This means that the buffer management mechanisms and correlated with it decision making process is very efficient.

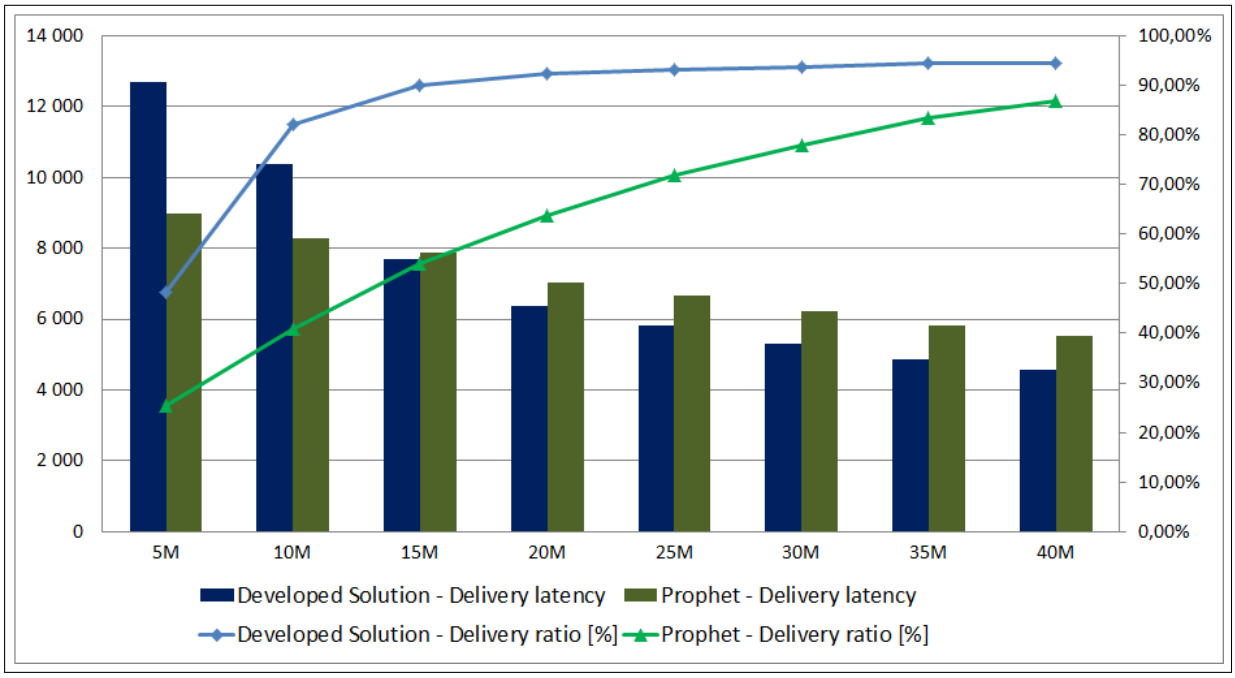


Figure 12: Impact of buffer size.

5 Conclusion

DTN networks are a young field of study and are not standardized. That leaves plenty of space for development new solutions for them. One of the very important part of network operation is routing and in case of DTN networks, in which nodes are connecting with others seldom and for short times, its efficiency is essential.

The algorithm that we developed aims for the networks, where nodes follow human mobility patterns, so the ones where networks are made of devices held by people. Our algorithm can be divided into three stages. The first one is the exchange and collection of historic control data needed for predicting future contacts and making decisions in further stages of the solution. The second phase is buffer management - this consists of processing control data and making decisions of releasing space in buffer when it is overflowed. The last stage is strictly making decision process about the messages to be transferred other nodes. Contacts are rare and short-lasting, so it is important to choose which messages to transfer and in what order, because in most cases there is no possibility of exchanging all the messages between nodes during contact. In case of developed solution, which is full-replication based, nodes try to exchange all the messages with other nodes - they do not choose which ones - but focus on the order in which messages are tried to be send.

The developed algorithm gets very good results, especially in the most important criteria, that is successful delivery probability. Compared to existing algorithms (Spray&Wait, Epidemic, MaxProp, Prophet), our new solution if very efficient, especially in the networks with human-mobility patterns.

Bibliography

[1] T.Omari; G.Franks; M.Woodside (2005); On the effect of traffic model to the performance evaluation of multicast protocols in MANET, *Electrical and Computer Engineering, Canadian Conference on,* IEEE: 404-407.

[2] L. Pelusi; A. Passarella; M. Conti (2006); Opportunistic networking: data forwarding in disconnected mobile ad hoc networks, *Communications Magazine*, IEEE 44(11):134-141.

[3] S. Corson; J. Macker (1999); Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations, *IETF RFC 2501*, 1-11.

[4] A. Vahdat; d. Becker (2000); *Epidemic routing for partially connected ad hoc networks*, Technical Report CS-200006, Duke University.

[5] T. Spyropoulos; K. Psounis; C.S. Raghavendra (2005); Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks, *Proc. of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*: 252-259

[6] A. Lindgren; A. Doria; E. Davies; and S. Grasic (2012); Probabilistic routing protocol for intermittently connected networks, *IETF RFC 6693*, 1-8.

[7] J.Burgess; et al. (2006); MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks, *INFOCOM*, 1-11, DOI: 10.1109/INFOCOM.2006.228.

[8] D. Karamshuk; C. Boldrini; M. Conti; A. Passarella (2011); Human Mobility Models for Opportunistic Networks, *IEEE Communications Magazine*, 49(12):157-165.

[9] D. Karamshuk; C. Boldrini; M. Conti; A. Passarella (2012); An Arrival-based Framework for Human Mobility Modeling, *Proceedings of the IEEE International Symposium WoWMoM*: 1-9.

[10] A. Passarella; M. Conti; C. Boldrini; R. I.M. Dunbar (2011); Modelling Inter-contact Times in Social Pervasive Networks, *Proceedings of the ACM MSWiM*: 333-340.

[11] C. Boldrini; M. Conti; A. Passarella (2007); Users Mobility Models for Opportunistic Networks: the Role of Physical Locations. *Proceedings of the WRECOM07*: 1-6

[12] C. Boldrini; M. Conti; A. Passarella (2009) The Socialble Traveller: Human Travelling Patterns in Social-Based Mobility, *Proceedings of the MobiWAC*: 34-41.

[13] P. Hui; J. Crowcroft (2007); How small labels create big improvements, *Procedeengs ot the IEEE PerCom*: 65-70.

[14] P. Hui; J. Crowcroft; E. Yoneki (2011); Bubble rap: Social-based forwarding in delay-tolerant networks, *Mobile Computing, IEEE Transactions*, 10(11): 1576-1589.

[15] W. Gao; Q. Li; B. Zhao; G. Cao (2009); Multicasting in delay tolerant networks: a social network perspective networks, *Proceedings of the ACM MobiHoc*: 299-308.

[16] A. Mei; G.Morabito; P. Santi; J.Stefa (2011); Social-aware stateless forwarding in pocket switched networks, *Proceedings of the IEEE INFOCOM*: 251-255.

[17] Y. Zhang; J. Zhao (2009); Social network analysis on data diffusion in delay tolerant networks, *Proceedings of the ACM MobiHoc*: 345-346.

[18] W. Gao; G. Cao (2011); User-centric data dissemination in disruption tolerant networks. *Proceedings of the IEEE INFOCOM*: 3119-3127.

[19] F. Fabbri; R. Verdone (2011); A sociability-based routing scheme for delay-tolerant networks, *EURASIP Wireless Communications and Networking*: 1-13.

[20] Y. Zhu; B. Xu; x. Shi; Y. Wang (2013); A survey of social-based routing in Delay Tolerant Networks: positive and negative social effects, *IEEE Communications Surveys and Tutorials* , 15(1):387-401.

[21] E.Bulut; B. K. Szymanski (2010); Friendship based routing in delay tolerant mobile social networks, *Proceedings of the IEEE GLOBECOM*, 10.1109/TPDS.2012.83, 23(12): 2254-2265.

[22] E. M. Daly; M. Haahr (2007); Social networks analysis for routing in disconnected delay-tolerant manets, *Proceedings of the MobiHoc*: 32-40.

[23] C. Boldrini; M. Conti; A. Passarella (2012); Less is More: Long Paths do not Help the Convergence of Social-Oblivious Forwarding in Opportunistic Networks. *Proceedings of the ACM/SIGMOBILE MobiOpp*: 1-8.

[24] A. Keranen; J. Ott; T.Karkkainen (2009); The ONE simulator for DTN protocol evaluation, *Proc. of the SimuTools*: DOI: 10.4108/ICST.SIMUTOOLS2009.5674.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(4):472-479, August 2016.

Direct Evolutionary Search for Nash Equilibria Detection

R.I. Lung

Rodica Ioana Lung
Babeş-Bolyai University of Cluj-Napoca
Faculty of Economics and Business Administration
T. Mihali 58-60, Cluj-Napoca
rodica.lung@econ.ubbcluj.ro

Abstract: A Direct method of computing mixed form Nash equilibria of a normal form game by using a simple evolutionary algorithm is proposed. The Direct Evolutionary Search algorithm (DES) uses a generative relation for Nash equilibria with binary tournament selection and uniform mutation. Numerical experiments are used to illustrate the efficiency of the method.
Keywords: Mixed form Nash equilibria, Evolutionary algorithms, generative relation

1 Introduction

The problem of computing Nash equilibria of normal form games is one of the most challenging problems faced by computer scientists. The complexity of this problem - varying hugely from a type of game to another - is still studied.

Normal form games can be solved by modern heuristics by transforming them into an optimization or fixed point problem [2]. An interesting challenge consists on designing a method that solves the game directly, as a *game* and not as a corresponding optimization problem.

The main problem in directly approaching the Nash equilibria is caused by the fact that there does not exist an order (or even preorder) relation defined for game situations that can guide a search operator towards the Nash equilibrium. While any kind of optimization endeavor is driven by a corresponding order (or preorder) relation defined in the objective space, in the case of Nash equilibria the lack of such a relation limits the design possibilities of computational heuristics.

However, recently [1] a generative relation for Nash equilibria - called the Nash ascendancy relation - has been proposed. The Nash ascendancy relation is defined on strategy profiles. Even if it does not induce an actual order, numerical results indicate that it is capable of guiding a search operator towards Nash equilibria.

Nash equilibria - definition and generative relation

Nash equilibrium [4] is the most popular solution concept in noncooperative game theory. A finite strategic game is defined by a set o players, a set of strategies available to each player and a set of payoff functions for each player and denoted by $\Gamma = (N, S, U)$ where:

- N represents the set of players, $N = \{1, ..., n\}$, n is the number of players;
- for each player $i \in N$, S_i represents the set of actions available to him, $S_i = \{s_{i_1}, s_{i_2}, ..., s_{i_{m_i}}\}$ where m_i represents the number of strategies available to player i and $S = S_1 \times S_2 \times ... \times S_N$ is the set of all possible situations of the game;
- for each player $i \in N$ denote by p_{ij} the probability that he selects its j-th action, $j \in \{1, ..., m_i\}$. Then $P_i = (p_{i1}, ..., p_{im_i})$ represents a probability distribution over the set of actions of player i and $P = (P_1, ..., P_n)$ represents a mixed strategy profile for the game, where $p_{ij} \in [0, 1]$ and $\sum_{j=1}^{m_i} p_{ij} = 1$, $\forall i = \overline{1, N}$ and $\forall j = \overline{1, m_i}$;

- for each player $i \in N$, $u_i(P)$ represents the expected payoff for the mixed strategy P;

Denote by (Q_i, P^*_{-i}) the strategy profile obtained from P^* by replacing the probability distribution of player i with Q_i i.e.

$$(Q_i, P^*_{-i}) = (P^*_1, P^*_2, ..., P^*_{i-1}, Q_i, P^*_{i+1}, ..., P^*_n).$$

A strategy profile $P \in S$ for the game Γ represents a Nash equilibrium [2, 4] if no player has anything to gain by unilaterally changing his own strategy while the others do not modify theirs.

Several methods to compute NE of a game have been developed. For a review on computing techniques for the NE see [2] and [6].

Nash ascendancy relation

A generative relation for Nash equilibria is a relation between two strategy profiles that enables their comparison with respect to the Nash solution concept, i.e. it evaluates which one is "closer" to equilibrium. In [1] such a generative relation has been introduced and shown that solutions that are non-dominated/ascended with respect to this relation are exactly the Nash equilibria of the game.

Consider two mixed strategy profiles P and Q and the operator κ that associates the cardinality of the set

$$\kappa(P, Q) = |\{i \in \{1, ..., n\} | u_i(Q_i, P_{-i}) \geq u_i(P), Q_i \neq P_i\}|$$

to the pair (P, Q), i.e. the number of players i that would benefit from unilaterally switching their strategies from P_i to Q_i.

The operator κ can be used to induce a relation on the set of mixed strategy profiles in the following manner: we can say that the strategy profile P *Nash ascends* the strategy profile Q in and we write $P \prec_N Q$ if the inequality

$$\kappa(P, Q) < \kappa(Q, P)$$

holds, i.e. there are les players that can increase their payoffs by switching their strategy from P to Q than vice-versa. It can be said that strategy profile P is more stable (closer to equilibrium) then strategy Q.

Regarding the Nash ascendancy relation, two strategy profiles P and Q may be in the following situation:

1. P ascends Q: $P \prec_N Q$ $(\kappa(P, Q) < \kappa(Q, P))$
2. Q ascends P: $Q \prec_N P$ $(\kappa(P, Q) > \kappa(Q, P))$
3. or $\kappa(P, Q) = \kappa(Q, P)$ and P and Q are considered *indifferent* (neither P ascends Q nor Q ascends P).

The strategy profile P^* is called non-ascended in Nash sense (NAS) if there does not exist any mixed strategy profile Q such that

$$Q \neq P^* \text{ and } Q \prec_N P^*.$$

A very important result in [1] shows that for pure strategies all non-ascended strategies are NE and also all NE are non-ascended strategies. The proof in the case of mixed strategies is

similar and direct. Thus the Nash ascendancy relation can be used to characterize the equilibria of a game and can be considered as a generative relation for NEs.

The Nash ascendancy concept was introduced with the purpose to compare two strategy profiles [1] during the search of an evolutionary algorithm in order to compute NEs of a game.

Consider two strategy profiles P^* and P from Δ. Then $k : \Delta \times \Delta \to N$ associates the pair (P^*, P) the cardinality of the set

$$k(P^*, P) = card\{i \in \{1, ..., n\} | u_i(P_i, P^*_{-i}) > u_i(P^*), P_i \neq P^*_i\}.$$

This set is composed by the players i that would benefit if - given the strategy profile P^* - would change their strategy from P^*_i to P_i.

It is obvious that for any $P^*, P \in S$, we have

$$0 \leq k(P^*, P) \leq n.$$

Definition 1. Let $P, Q \in \Delta$. We say the strategy profile P Nash ascends Q and we write $P \prec Q$ if the inequality

$$k(P, Q) < k(Q, P),$$

holds.

Remark 1.1. Two strategy profiles $P, Q \in \Delta$ can have the following relation:

1. either P Nash ascends Q,
2. either Q Nash ascends P,
3. if $k(P, Q) = k(Q, P)$ then P and Q are *indifferent*

Definition 2. A strategy profile $P^* \in S$ is called non-dominated with respect to the Nash ascendancy relation (NNS) if

$$\nexists Q \in \Delta, Q \neq P^* \text{such that } Q \prec P^*.$$

Definition 3. The set of all Nash nondominated strategy profiles with respect to the Nash ascendancy relation is the set containing all nondominated strategies i.e.

$$NND = \{s \in S | s \text{ Nash non-dominated with respect to the Nash ascendancy relation}\}$$

Proposition 1. A strategy profile $P^* \in \Delta$ is a NE iff the equality

$$k(P^*, Q) = 0, \forall Q \in \Delta,$$

holds.

Proof: Let $P^* \in \Delta$ be a NE. Suppose there exists $Q \in \Delta$ such that $k(P^*, Q) = w$, $w \in \{1, ..., n\}$. Therefore there exists $i \in \{1, ..., n\}$ such that $u_i(Q_i, P^*_{-i}) > u_i(P^*)$ and $Q_i \neq P^*_i$, which contradicts the definition of NE.

For the second implication, let $P^* \in \Delta$ such that $\forall Q \in \Delta, k(P^*, Q) = 0$. This means that for all $i \in \{1, ..., n\}$ and for any $Q_i \in \mathcal{P}_i$ we have $u_i(Q_i, P^*_{-i}) \leq u_i(P^*)$. It follows that P^* is a NE. □

Proposition 2. All NE are Nash nondominated solutions (NND) i.e.

$$NE \subseteq NND.$$

Proof: Let $P^* \in \Delta$ be a NE. Suppose that there exists a strategy profile $P \in \Delta$ such that $P \prec P^*$. It follows that $k(P, P^*) < k(P^*, P)$. But $k(P^*, P) = 0$, therefore we must have $k(P, P^*) < 0$ which is not possible since $k(P, P^*)$ denotes the cardinality of a set. □

Proposition 3. All Nash nondominated solutions are NE, i.e.

$$NND \subseteq NE.$$

Proof: Let P^* be a nondominated strategy profile. Suppose P^* is not NE. Therefore there must exist (at least) one $i \in \{1, ..., n\}$ and a strategy $P_i \in \mathcal{P}_i$ such that

$$u_i(P_i, P^*_{-i}) > u_i(P^*),$$

holds. Let's denote by $Q = (P_i, P^*_{-i})$. It means that $k(P^*, Q) = 1$. But $k(Q, P^*) = 0$. Therefore $k(Q, P^*) < k(P^*, Q)$ which means that $Q \prec P^*$ thus the hypothesis that P^* is nondominated is contradicted. □

Using propositions 2 and 3 it is obvious that the next result holds:

Proposition 4. The following relation holds:

$$NE = NND,$$

i.e. all NE are also Nash nondominated and also all Nash nondominated strategies are NE.

Direct evolutionary search

The Direct Evolutionary Search (DES) algorithm is a simple evolutionary algorithm based on tournament selection and uniform mutation designed for Nash equilibria detection.

Individuals represent strategy profiles of the game. Real valued encoding is used. Each individual is represented as a vector composed of $(\sum_{i=1}^{n} m_i)$ components between 0 and 1. For each individual n corresponding payoffs are computed.

Selection Binary tournament selection is used in the following manner: For each individual i, another one k is selected randomly. If individual k Nash ascends individual i, k is selected and copied in a separate population of children $P\prime$.

Mutation Uniform mutation with probability p_m is applied to all children in $P\prime$. With probability p_m all strategies are modified ($\pm$) with ε. If the resulting value is lower than 0, it is set to 0. If it is higher than 1, it is set to 1.

Termination condition DES runs either a maximum number of generations which is a parameter of the algorithm and depends on the problem, either until no child can replace a parent for 100 generations. For this the variables *CountReplacements* and *control* in Algorithm 1 are used.

Algorithm 1 Direct Evolutionary Search algorithm

Randomly generate population;
Evaluate population;
$CountReplacements = 1$;
$control = true$;
for $nrgen = 0; ((nrgen < MaxNoGenerations)$ and $(control)); nrgen++$ **do**
  Apply Selection;
  Apply Mutation(p_m);
  Replace Children;
  $CountReplacements+ =$no of children that replace a parent;
  if $nrgen = 100$ **then**
    if $CountReplacements = 0$ **then**
      $control = false$;
    else
      $CountReplacements = 0$;
    end if
  end if
end for
Return Non-Ascended Solutions;
=0

Table 1: Description of the seven games tested

Name of game	No. of players	No. of strategies	Type of NE
GAME1	2	2,2	totally mixed
GAME2	2	2,2	totally mixed
Matching pennies	2	2,2	totally mixed
G1	3	2,2,2	totally mixed
G2	3	3,3,3	mixed
O'Neill	2	4,4	totally mixed
Poker	2	4,2	totally mixed

Table 2: Descriptive statistics of numerical results obtained by DES using the following parameters: population size 50, $p_m = 0.5$, $\varepsilon = 0.5$.

GAME1			
Avg. dist:	0	St dev:	0
Avg. no. gen.:	844.33	St dev:	476.57
Avg. eval.:	447,117.47	St dev:	271909.88
GAME3			
Avg. dist:	7.75094E-17	St dev:	3.13E-17
Avg. no. gen.:	501.00	St dev:	141.42
Avg. eval.:	243,736.93	St dev:	74416.31
Matching	pennies		
Avg. dist:	0	St dev:	0
Avg. no. gen.:	497.67	St dev:	125.12
Avg. eval.:	233,854.53	St dev:	56641.84
G1			
Avg. dist:	4.04061E-11	St dev:	2.25E-17
Avg. no. gen.:	11,941	St dev:	13424.39
Avg. eval.:	15,541,134	St dev:	17317631
G2			
Avg. dist:	4.04061E-11	St dev:	6.98E-18
Avg. no. gen.:	931	St dev:	324.70
Avg. eval.:	1,076,041	St dev:	374858.2
Oneill			
Avg. dist:	2.79146E-05	St dev:	0.0001
Avg. no. gen.:	8,217.63	St dev:	26628.89
Avg. eval.:	6,556,652.7	St dev:	21598601
Poker			
Avg. dist:	3.70074E-18	St dev:	1.99E-17
Avg. no. gen.:	794.33	St dev:	240.73
Avg. eval.:	419,827.93	St dev:	126713.48

2 Numerical results

DES was first tested on a set of simple well known 2 or 3-players with up to 4 actions available to them. The games have been chosen from the GAMBIT [3] set of normal form games. The characteristics of the games are presented in Table 1. Table 2 presents descriptive statistics of the results obtained using DES for the seven games: average and standard deviation values for the minimum distance to the NE of the game for the Non Ascended solutions in the final population, number of evaluations and of generations until DES stopped the search.

The second set of games that was used to test if DES was generated by using the GAMUT distribution [5]. The distribution names, characteristics and rates of success of DES are presented in Table 3.

Table 3: GAMUT distributions and results obtained by DES. Rate of success represents percent of runs in which DES detected a NE. Parameters of DES: population size 100, $p_m = 0.1$

Distribution	No. of players	No. of actions	Rate of success
Bertrand	2	10/player	100%
Ologopoly	2	20/player	100%
	2	50/player	100%
	2	100/player	100%
	5	2/player	100%
	5	4/player	100%
	5	6/player	100%
Bidirectional LEG	2	10/player	100%
Complete Graph	2	20/player	100%
	2	50/player	100%
	2	100/player	100%
	5	2/player	100%
	5	4/player	100%
	5	6/player	100%
Bidirectional LEG	2	10/player	100%
Random Graph	2	20/player	100%
	2	50/player	100%
	2	100/player	100%
	5	2/player	100%
	5	4/player	100%
	5	6/player	100%
Bidirectional LEG	2	10/player	100%
Star Graph	2	20/player	100%
	2	50/player	100%
	2	100/player	100%
	5	2/player	100%
	5	4/player	100%
	5	6/player	100%
Covariance Game	2	10/player	100%
$\rho = 0.9$	2	20/player	100%
	2	50/player	100%
	2	100/player	100%
	5	2/player	100%
	5	4/player	100%
	5	6/player	100%

Conclusion

The presented results indicate that it it possible for an evolutionary algorithm to directly detect Nash equilibria of normal form games. A population composed of situations of the game is randomly generated; using selection and mutation operators and the Nash ascendancy relation individuals of the populations are guided towards the Nash equilibrium of a normal form game.

Two sets of problems have been chosen to test the method: the first one consists of games presenting a mixed Nash equilibrium. Results indicate that DES is capable of locating mixed equilibria for the selected games. The second group of games is used to test the scalability of the method: the number of actions available to each player, as well as the number of players are increased for five distributions available in the GAMUT package. Results also indicate the potential of the method, inspite of well known scalability issues associated with evolutionary algorithms.

Acknowledgment

This work was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS - UEFISCDI, project number PN-II-RU-TE-2014-4-2560.

Bibliography

[1] Rodica Ioana Lung and D. Dumitrescu (2008); Computing nash equilibria by means of evolutionary computation, *Intrenational Journal of Computers Communications & Control*, Suppl. issue, 3(5):364-368.

[2] Richard D. McKelvey and Andrew McLennan (1996); Computation of equilibria in finite games. In H. M. Amman, D. A. Kendrick, and J. Rust, editors, Handbook of Computational Economics, volume 1 of Handbook of Computational Economics, chapter 2, pp. 87-142, Elsevier, 1996.

[3] Richard D. McKelvey, Andrew M. McLennan, and Theodore L. Turocy (2010); *Gambit: Software tools for game theory*, Technical report.

[4] John F. Nash (1951) Non-cooperative games, *Annals of Mathematics*, 54:286-295.

[5] Eugene Nudelman, Jennifer Wortman, Yoav Shoham, and Kevin Leyton-Brown(2004); Run the gamut: A comprehensive approach to evaluating game-theoretic algorithms, *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, Washington, DC, USA, 2004. IEEE Computer Society, AAMAS04*, 2: 880-887.

[6] Ryan Porter, Eugene Nudelman, and Yoav Shoham (2008); Simple search methods for finding a nash equilibrium, *Games and Economic Behavior*, 63(2): 642-662.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(4):480-492, August 2016.

Modeling Mobile Cellular Networks Based on Social Characteristics

J. Ma, W. Ni, J. Yin, R.P. Liu, Y. Yuan, B. Fang

Ji Ma*
1. Beijing University of Posts and Telecommunications
Beijing, China, 100876
2. Digital Productivity Flagship, CSIRO, Australia, 2122
*Corresponding author: maji@bupt.edu.cn

Wei Ni, Jie Yin, Ren Ping Liu
Digital Productivity Flagship, CSIRO, Australia, 2122
Wei.Ni@csiro.au, Jie.Yin@csiro.au, Ren.Liu@csiro.au

Yuyu Yuan, Binxing Fang
Beijing University of Posts and Telecommunications
Beijing, China, 100876
yuanyuyu@bupt.edu.cn, fangbx@bupt.edu.cn

Abstract: Social characteristics have become an important aspect of cellular systems, particularly in next generation networks where cells are miniaturised and social effects can have considerable impacts on network operations. Traffic load demonstrates strong spatial and temporal fluctuations caused by users social activities. In this article, we introduce a new modelling method which integrates the social aspects of individual cells in modelling cellular networks. In the new method, entropy based social characteristics and time sequences of traffic fluctuations are defined as key measures, and jointly evaluated. Spectral clustering techniques can be extended and applied to categorise cells based on these key parameters. Based on the social characteristics respectively, we implement multi-dimensional clustering technologies, and categorize the base stations. Experimental studies are carried out to validate our proposed model, and the effectiveness of the model is confirmed through the consistency between measurements and model. In practice, our modelling method can be used for network planning and parameter dimensioning to facilitate cellular network design, deployments and operations.
Keywords: social characteristics, mobile networks, spectral clustering, energy efficiency, traffic model.

1 Introduction

Due to the increasing popularity of smart phones, mobile data has been exponentially increasing [1]. The radio base stations (RBS), e.g., Node B, act an important role in cellular networks. When mobile users move between radio base stations and access data network, those activities cause RBS to present significant social features. First, the spatial resources are limited since a certain space has its maximum capacity of people. Second, and temporal resources are restricted because of human physiology reasons, using mobile is only a portion during users' day life-cycle. Third, users' behaviors lead a social pattern of radio base stations. Understanding those characteristics is important for data traffic forecasting, network optimization, energy saving and delivering of service.

Previous works has explored that data traffic has significant spatial-temporal pattern in access network. Paper [6] characterize users' activity patterns and find significant temporal and spatial variations in different parts of the network. Traffic spatial distribution has been studied

in paper [7], which propose a spatial model of traffic density to simulate log-normal distribution. The reason of the traffic phenomena is mobility of mobile users, while people move between RBSs and use their equipment to access data network. González et al. [8] have studied the human mobility patterns on mobile call and SMS transactions with a six-month period data. However, human mobility study on mobile data has its limits, since only traffic data are captured while users using their mobile equipments. Song et al. [9] pointed out a 93% potential predictability in user mobility across the whole user base, by measuring the entropy of each individual's trajectory. In [21], the authors proposed a feedback simulation model to analyse the interaction between urban densities and travel mode split.

Human activities has inherent social characteristics, thus recent papers pay more attentions to the traffic social patterns and its impact to cellular networks. In [23], colored petri net (CPN) models are proposed to specify citizen's preferences and affinities in front of an urban contextual change, while multiple agent system is proposed to evaluate the social dynamics by translating the CPN semantic rules into agent's rules. The authors of papers [10,11] use the Gini coefficient to measure user social pattern of a certain area or period, and apply it to spectrum and energy efficiency in heterogeneous cellular networks. In [22], a social similarity-aware multicast routing protocol is proposed for delay tolerant networks. The social similarity is quantified as the probability of the encounter of two nodes in the future based on the number of common neighbors in the history. Social informations can also be used to detect community structure [12] and to match interests [13] in heterogeneous networks. However, the above works were focused only on certain parts of social characteristics, and are unable to characterize the social behaviours of RBSs in a comprehensive way.

In this paper, we develop a new model which categorizes base stations based on multiple important social characteristics of the base stations (more specifically, the traffic at the base stations). Four key properties are taken into account to indicate the social characteristics of RBSs, namely, traffic fluctuation, user nondeterminacy, temporal homogeneity and usage diversity. Based on these, we apply multi-dimensional spectral clustering techniques and categorize RBSs with similar social patterns into groups. For each of the categories, we establish a temporal traffic model and derive important model parameters. Our proposed model is able to characterize different cellular scenarios and terrains with an emphasis on the diversity of user mobility and traffic fluctuation. This is of practical value. Specifically, the model can refine network simulation and network planning parameters for a range of practical network environments. Users social behaviours and mobility patterns can be captured in those parameters. In this sense, our proposed model can facilitate designing, simulating, and evaluating network deployment.

The rest of the paper is organized as follows. In Section 2, we describe the actual mobile data, based on which our modelling study was conducted. In Section 3, the four key characteristics, i.e., traffic fluctuation, user nondeterminacy, temporal homogeneity and usage diversity, are defined. In Section 4, we elaborate on the clustering based social behaviour modelling method. The application scenarios on energy efficiency are given in Section 5, followed by a conclusion in Section 6.

2 Data preliminary

Our study in this work is based on a real-world mobile data set comprised of user transaction logs, which are collected from the biggest city, Chongqing, in China. This data set records 1.6 million anonymous mobile users' data traffic on a Saturday, which has 38,000 RBSs within the city over 2,000 square kilometres. Each time a user initiates a package data request, the location of the associated RBS and traffic summary are recorded. We extract the data schema as a map of points of interest (POI), while each RBS is a POI. Mobile users generate data traffics when

they check these points.

Each record of our data set includes a list of attributes, including subscriber identity (SID), local area code (LAC), cell identity (CI), coordinate, total traffic volume, time-stamp, type and etc. SID is the anonymous mobile number hashed by SHA-1 algorithm. A RBS can be identified by the composite of LAC and CI. Coordinate includes the latitude and longitude of a RBS. We consider the total traffic to calculate measurements in general instead of up-link and down-link traffic. The attribute type presents the protocol codes of transactions. For example, HTTP requests are recorded as code 204 of POST and 205 of GET; SMTP, POP3, and IMAP are represented by 500, 501 and 502 respectively. To make this attribute more representative, We categorize those 51 codes into eight types, namely Web, Stream Media, Instant Messaging (IM), File Sharing, E-mail, Multimedia Messaging Service (MMS), Voice over Internet Phone (VoIP), and Miscellaneous traffic.

These traffic data show nonuniform patterns in spatial-temporal dimensions. In temporal dimension, total traffic presents dual peak like fluctuation, and peak hours are about 8:00 AM and 19:00 PM. There are also little burst around noon and 14:00 PM. Web surfing, which contributes 73% traffic, has the similar peak hours. Other types have little differences in usage hour. The second contribution is a bundle of miscellaneous traffics, gives a portion of 11%, which includes several signaling and unrecognized traffics such as DNS query, Apple push, Multi-cast, Session Initiation, and etc. Stream media and IM generate volumes of 7% and 6%. The others share the rest about 3% of total.

3 Social characteristics extracted from traffic data

We study the characteristics of cellular networks from perspectives of individual RBS, with an emphasis on user activity patterns and social behaviour. We consider two types of characteristics. One is traffic fluctuation, and the others are entropy based characteristics, including user nondeterminacy, temporal homogeneity, and usage diversity.

3.1 Traffic fluctuation

Traffic fluctuation indicates the fluctuation of traffic volume at a RBS. Considering one day, the traffic fluctuation can be expressed as a time sequence of 24 hours. We use Spearman's rank correlation coefficient to measure the similarity between two sequences. It is defined as the Pearson correlation coefficient between the ranked variables. The value is between -1 and 1. A perfect correlation of $+1$ or -1 occurs when each of the variables is a perfect monotone function of the other, and 0 indicates there is no correlation. We rank the hourly traffic volume of each RBS as descending order, and each hour has a rank number in a day. For given two rank sequences P and Q, the fluctuation similarity can be calculated by:

$$\rho(P,Q) = \frac{\sum_{i=1}^{n}(P_i - \bar{P})(Q_i - \bar{Q})}{\sqrt{\sum_{i=1}^{n}(P_i - \bar{P})^2}\sqrt{\sum_{i=1}^{n}(Q_i - \bar{Q})^2}}, n = 24,$$

where P_i or Q_i is the traffic rank at hour $i = 1, 2, \cdots, 24$.

3.2 Entropy based characteristics

RBSs can have non-uniform traffic density in spatial-temporal dimensions [6] or application usage [14]. We introduce Shannon entropy to generalize nonuniform traffic characteristics from RBSs' angle.

User nondeterminacy

User nondeterminacy indicates human mobility within each cell, which further indicates the social function of the cell. For instance, a residential zone has a stable group of users, while users at a transport junction are fast changing. This property shows the uncertainty of user traffic volumes. For each RBS, we define the user nondeterminacy as:

$$S^u = -\sum\nolimits_i u(i) \log u(i),$$

where $u(i)$ is the proportion of traffic generated by user i. A greater value means RBS serves more users, and the users generate traffics more evenly. S^u can be as small as 0, if only one user uses this RBS in a whole day. In our work, we use SID as an aggregate group to calculate user nondeterminacy.

Temporal homogeneity

Another important aspect of RBSs is temporal homogeneity, which is the distribution of time intervals between traffic generations. Usually, a single function area has a sharp temporal pattern of traffic, while a mixed function area has more stable traffic in temporal dimension. To capture such patterns, we aggregate the traffic volume on an hourly basis. The temporal homogeneity of each RBS can be defined as:

$$S^h = -\sum\nolimits_i h(i) \log h(i),$$

where h is the proportion of traffic generated within an hour i. A lower value of S^h shows a concentration on a few hours. This property has the maximum value of $\log 24 \approx 4.585$, under the condition that traffic is evenly divided into 24 hours. The attribute timestamp is divided into 24 hours for producing temporal homogeneity.

Usage diversity

Users' usage behaviour is another key indicator of the traffic pattern, which reflects different application preferences and access habits of users. We use traffic protocols as service identification. To extract those protocols, we categorize traffics into eight major types. Usage diversity of data services can be described by the type entropy, which is defined as:

$$S^v = -\sum\nolimits_i v(i) \log v(i),$$

where $v(i)$ is the proportion of traffic by the above categories. This property shows the diversity of traffic volume of the eight types, which would have a larger value if a RBS resource is more evenly distributed among different services. Our analysis shows that there is no obvious correlation between usage diversity and user nondeterminacy, or temporal homogeneity. The Pearson's correlation coefficients are only 0.117 and 0.182 respectively.

4 Clustering based social modeling and analysis

The characteristics of RBSs can identify the function and the population density of the coverage areas of the RBSs, since these characteristics are strongly related to human mobility and social activities. Modelling the realistic RBS characteristics can be useful for network designing and simulating with social behaviours. Clustering analysis is an unsupervised technique to study laws without prior knowledge. In this section, we propose to model RBS social characteristics using spectral clustering [15]. Spectral clustering has many advantages compared with traditional

clustering algorithms. Affinities between every pair of RBSs can be used. And there is no strong assumption on the statistics of clusters, such as the number of clusters or feature selection.

We would like to highlight that our key contribution is the social characteristics modeling for mobile networks. We extract four properties to examine social characteristics of RBSs, and all these characteristics are strongly related to human mobility and social activities. Our model is able to be applied to characterize different cellular scenarios and terrains. This application is enabled by our new designed spectral clustering method.

Our new design of spectral clustering method is able to implement the area functions distinguish. By inspecting the gap of an eigenvalue and the average with the previous value we can identify how many clusters needed to be grouped. It is worth pointing out that the application of the spectral clustering to areas function discovery is new, although the concept itself is not.

Our new design of spectral clustering method is also necessary to implement the area functions distinguish. In fact, we cannot exactly know how many cellular scenarios and terrains existed in current network before we really find them out. None of the existing works are able to address the issue, though cells clustering have been extensively studied.

4.1 Spectral clustering

The main idea of spectral clustering is using the spectrum of the similarity matrix of the data to perform dimensionality reduction before clustering. A typical spectral clustering process has these following steps:

Constructing affinity matrix

We construct a n dimensional distance matrix $\mathbf{M}$, where n is the number of RBSs. Each element m_{ij} represents the distance between RBS i and RBS j. Distance matrix $\mathbf{M}$ can be transformed to affinity matrix $\mathbf{W}$. $\mathbf{W}$ is weighted adjacency matrix, and each edge is weighted by pairwise vertex affinity. $\mathbf{W}(i,j)$ is equal to 0 if i and j are not connected, and $\mathbf{W}(i,j)$ gives positive value if i,j are connected. A greater value indicates two RBSs are closer in characteristic space. We use Gaussian similarity function $w(x_i, x_j) = \exp(-m(i,j)^2/(2\sigma^2))$ to form matrix $\mathbf{W}$, where the parameter σ controls the width of neighbourhood.

Building Laplacian

Given a graph with affinity matrix $\mathbf{W}$, the way to construct k partitions is to solve min-cut problem. To avoid over cutting, the normalized cut is introduced [16]. For a given number k of subsets, the min-cut approach simply consists in choosing a partition $A_1, ..., A_k$ which minimizes:

$$N_{cut}(A_1, \ldots, A_k) = \frac{1}{2}\sum_{i=1}^{k} \frac{\mathbf{W}(A_i, \bar{A}_i)}{vol(A_i)}$$

where $\bar{A}_i$ is the complement of A_i ,$vol(A_i)$ is the weights of edges of A_i. According to the normalized cut problem, the normalized graph Laplacian [17] can be used as given by:

$$\mathbf{L}^{norm} = \mathbf{D}^{-1/2}(\mathbf{D} - \mathbf{W})\mathbf{D}^{-1/2}$$

where $\mathbf{D} = \mathrm{diag}(d_1, \ldots, d_n)$, and $d_i = \sum_j \mathbf{W}_{ij}$.

Clustering with reduced dimensions

After Laplacian matrix $\mathbf{L}$ is constructed, we find the first k generalized eigenvectors $u_1, u_2, \ldots, u_k$, corresponding to the k smallest eigenvalues of $\mathbf{L}$, and form a matrix $\mathbf{U} = [u_1, u_2, \ldots, u_k] \in R^{n \times k}$ by stacking the eigenvalues in columns. we normalize each row i of matrix $\mathbf{U}$ so that they have unit Euclidean norm by:

$$t_{ij} = \frac{u_{ij}}{\sqrt{\sum_{g=1}^{k} u_{ig}^2}}, u_{ij} \in U,$$

where t_{ij} is the element of matrix $\mathbf{T}$. We then treat each row of matrix $\mathbf{T}$ as a point in $R^{n \times k}$ and cluster them into k clusters via the k-means clustering algorithm [18].

Analyzing and characterizing

To characterize each cluster, we first use kurtosis and skewness to estimate the data distribution of each feature. Kurtosis is a descriptor of the shape of a probability distribution. A higher kurtosis means more of the variance is the result of infrequent extreme deviations. Skewness is a measure of the asymmetry of the probability distribution. Negative skew indicates that the tail on the left side of the probability density function is longer than the right side. Conversely, positive skew has the longer right tail. For example, normal distribution has kurtosis of 3 and skewness of 0; kurtosis of log-normal is greater than 3, and skewness is positive. After the function of probability distribution is determined, we apply maximum likelihood estimation (MLE) to estimate the corresponding parameters.

4.2 Clustering with traffic fluctuation

We first study on clustering traffic fluctuation similarity. However, traffic fluctuation similarity is a relative property, but not an absolute coordinate. In order to keep coherence with other characteristics, we transform the correlation coefficient to Euclidean distance. According to alternative definition of $\rho_{x,y}$:

$$\rho_{x,y} = \frac{\sum_i x_i y_i - n\mu_x \mu_y}{n\sigma_x \sigma_y}$$

where μ_x and μ_y are the means of X and Y respectively, and σ_x and σ_y are the standard deviations of X and Y, if X and Y are standardized, they will each have a mean of 0 and a standard deviation of 1, we can get the distance expression from correlation coefficient [19]:

$$d(X^{\ddagger}, Y^{\ddagger}) = \sqrt{2n(1 - \rho(X^{\ddagger}, Y^{\ddagger}))} \tag{1}$$

The codomain of function 1 is from 0 to $2\sqrt{n}$. The normalized distance is shown as:

$$d^{norm}(X^{\ddagger}, Y^{\ddagger}) = \sqrt{\frac{1 - \rho(X^{\ddagger}, Y^{\ddagger})}{2}} \tag{2}$$

We use this normalized formula and standardized traffic ranks to calculate the distance on traffic fluctuation similarity.

Fig.1 shows the smallest 20 eigenvalues of the Laplacian matrix with 2-phase simple moving average. We can identify gaps by inspecting the difference between an eigenvalue and the average with the previous value. We can see that from the figure, the major gaps appear between the first three values. There are minor gaps after the fourth and fifth eigenvalue. According to those gaps, it is reasonable to group the data into 2, 4, 6 or 7 clusters.

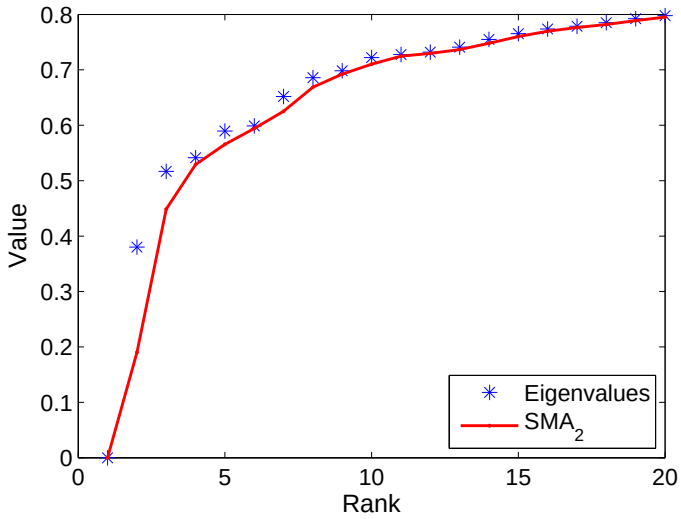


Figure 1: The smallest 20 eigenvalues of the Laplacian generated from traffic fluctuation.

We proceed to show the results of grouping our data into four clusters. The generated clusters exhibit obvious patterns of distinct traffic fluctuations as shown in Fig.2, in which there is a mixed cluster (cluster 1), one busy afternoon cluster (cluster 2), a evening time cluster (cluster 3), and cluster 4 has two peak time at 1:00 AM and 10:00 AM. By clustering RBSs with traffic fluctuation, the results can be used to operate the complementary cells to improve spectrum and energy efficiency.

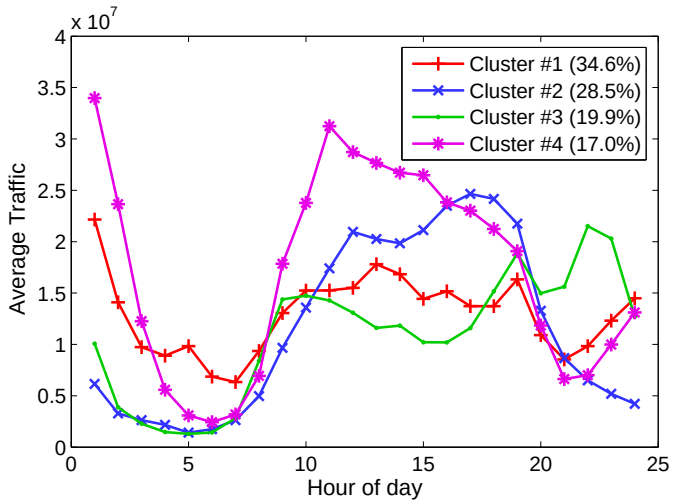


Figure 2: The average traffic of four clusters generated from traffic fluctuation similarity.

4.3 Clustering on full dimensions

We inspect the distribution of user nondeterminacy, temporal homogeneity, and usage diversity within the traffic data. User nondeterminacy follows normal distribution in this city ($\mu = 3.69, \sigma = 1.63$). The expectation is 3.69 and standard deviation is 1.63. The density histogram of temporal homogeneity has negative skew and peaks are greater then 4. It follow truncated Laplace distribution ($\mu = 4.11, \sigma = 0.79$), where μ is a location parameter and σ is a

scale parameter. Usage diversity is also well fitted by normal distribution. A city range have a expectation of 1.08 and a standard deviation of 0.38.

We combine all characteristics, including user nondeterminacy, temporal homogeneity, usage diversity, and traffic fluctuation, to investigate the traffic patterns of individual RBS. The full dimensional distance between RBS i and RBS j can be written as:

$$m_{i,j} = \sqrt{(S_i^u - S_j^u)^2 + (S_i^h - S_j^h)^2 + (S_i^v - S_j^v)^2 + \frac{1-\rho_{i,j}}{2}}$$

Entropy based properties are normalized in distance calculation.

We proceed to show the results of grouping our data into three clusters. Fig. 3 illustrates the characteristics of the three clusters that are generated using our spectral clustering method. Fig. 3(a) shows the three entropy based social aspects of the three clusters, i.e., user instability, temporal homogeneity, and usage diversity. We can see that each cluster has an ellipse-like distribution of points. The clusters are visually separable in the three-dimensional space. Figs. 3(b), 3(c), and 3(d) show the traffic fluctuation of the three clusters along with the time. The three clusters exhibit distinct patterns of their average traffic fluctuation, as highlighted in the figures, whereas cells of the same clusters show strong similarity of the patterns, as respectively shown in each of the three figures.

Table 1: Statistical results of three clusters

Cluster	User Nondeterminacy		Temporal Homogeneity		Usage Diversity	
	mean	std	mean	std	mean	std
1	4.705	1.104	3.923	0.371	1.227	0.344
2	6.089	0.741	4.090	0.197	1.259	0.230
3	3.422	1.012	3.206	0.618	0.944	0.298

Table 1 further gives the mean and standard deviation of each property for each cluster. Since the entropy based properties have no obvious groups, a clustering with only these properties will cause an average separation. In other words, traffic fluctuation represents human mobility and social activity patterns. The other three characteristics indicate the community variety. Full dimensional clustering can recognize different function areas and communities.

Further more, we have analyzed nine clusters result in detail. Fig. 4 shows the average traffic fluctuation of the generated nine clusters and Table 2 reports the characteristics of the other three properties (user nondeterminacy, temporal homogeneity and usage diversity).

Table 2: Statistical analysis of nine clusters

No.	Prop.	User Nondeterminacy				Temporal Homogeneity				Usage Diversity			
		mean	std.	kurt.	skew.	mean	std.	kurt.	skew.	mean	std.	kurt.	skew.
1	25.7 %	4.31	1.20	3.45	-0.67	3.87	0.43	12.52	-2.28	1.20	0.33	2.97	0.02
2	16.3 %	5.51	0.72	3.01	-0.30	4.09	0.24	3.91	-1.03	1.20	0.19	2.85	-0.19
3	15.0 %	6.69	0.57	3.42	0.65	4.19	0.14	4.11	-0.94	1.23	0.22	3.040	0.07
4	14.2 %	4.79	0.93	3.12	-0.34	3.85	0.32	3.50	-0.79	1.55	0.21	2.74	0.31
5	13.5 %	4.94	0.88	2.57	-0.31	3.82	0.29	2.52	-0.26	0.83	0.19	2.23	-0.01
6	8.1 %	5.33	0.71	3.18	-0.41	3.93	0.28	3.77	-0.94	1.22	0.18	2.80	0.06
7	5.3 %	2.78	0.77	2.50	-0.03	2.90	0.40	2.63	0.13	0.96	0.30	2.35	0.05
8	1.4 %	1.68	0.60	1.87	-0.09	1.78	0.55	3.27	-0.90	0.95	0.45	1.93	-0.02
9	0.5 %	5.85	0.57	2.08	-0.36	4.15	0.14	2.92	-0.86	0.98	0.27	1.96	-0.33

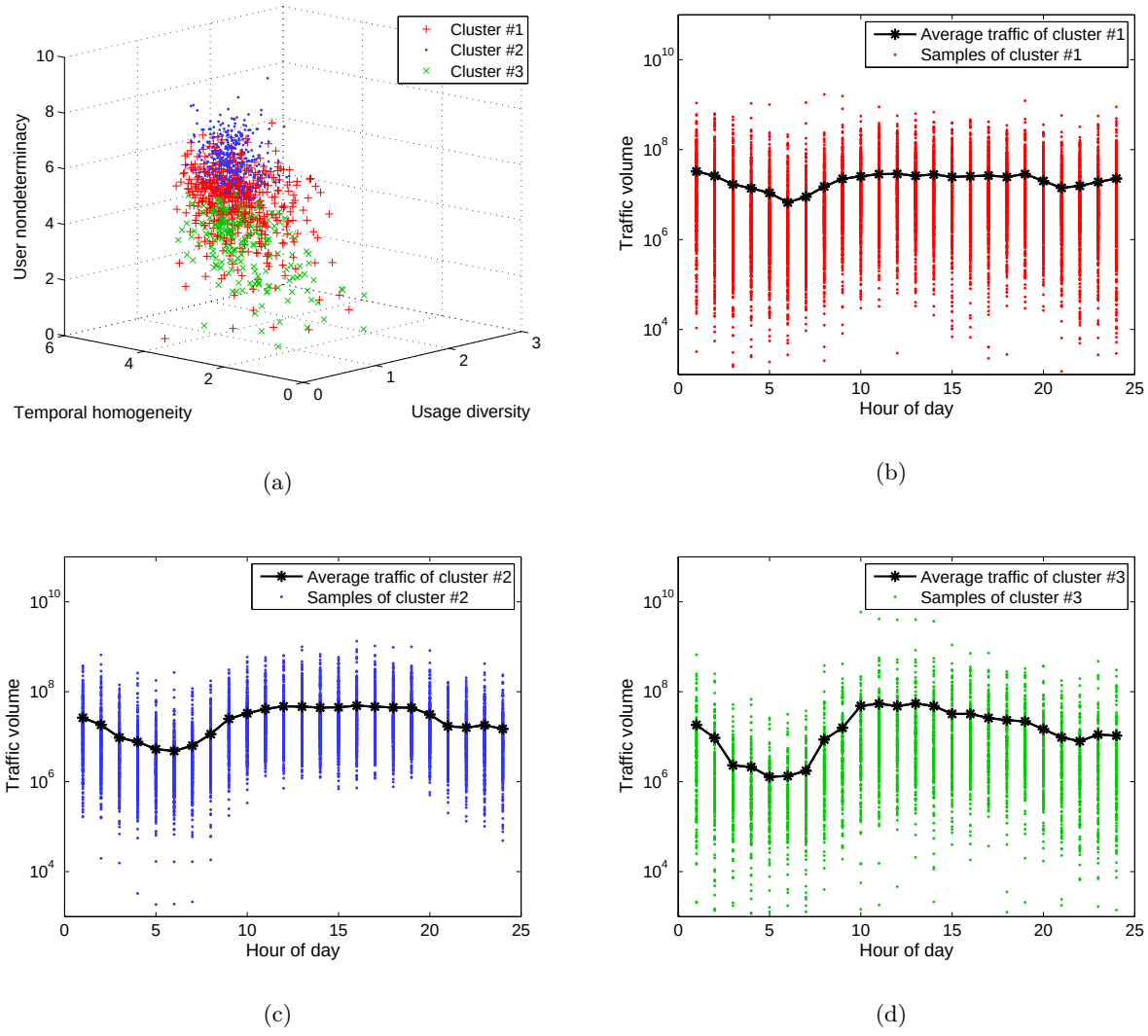


Figure 3: The characteristics of three clusters: (a) The clustering result of the entropy based social characteristics, user instability, temporal homogeneity, and usage diversity, where the three clusters are produced using our spectral clustering method. (b), (c), and (d) The traffic fluctuation of the three clusters.

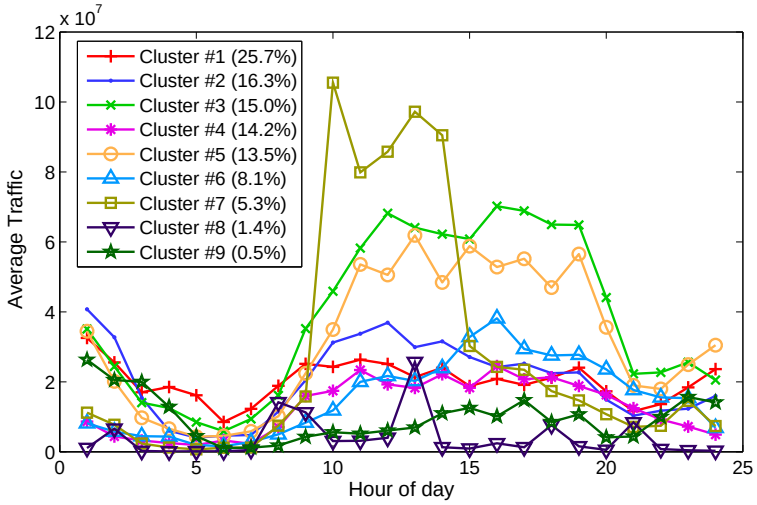


Figure 4: Average of the measured traffic for the nine clusters that we generate using our multi-dimensional spectral clustering method.

Distributions of the other three entropy based social characteristics are provided in Table 2. Using the table, we can derive models for each social aspect of a cluster. Specifically, we can use

1. a normal distribution or its truncated variations to model the social characteristics with kurtosis of around 3 and skewness of around 0. Examples include the user instability of cluster 2.

2. Weibull, gamma and log-normal distributions to model the social characteristics with kurtosis of between 3.3 and 3.5. Examples include the truncated Weibull distribution for the user instability of cluster 1.

3. truncated Laplace distributions to model the social characteristics with kurtosis of greater than 3.5 and skewness of less than -0.7. Examples include the temporal homogeneity of clusters 1 and 2.

In addition, the Weibull distribution can also be used to model the social characteristics with kurtosis of less than 2.7. Examples include the three entropy based characteristics of cluster 7.

5 Applications to energy-efficient wireless networks

Mechanisms that adapt the cellular layout to the traffic density distribution can be used to improve the energy efficiency of a cellular network, including heterogeneous cellular deployment, cell breathing and relay network [20]. The social characteristics of RBS can be used as adjusting factors or constraints in these approaches to wireless networks.

In heterogeneous cellular networks, base stations can be switched off if a decrease of traffic is predicted. The combination of multi-layer RBSs is designed to meet non-uniform traffic demand. The social characteristics are concise properties to indicate human activity pattern in RBSs' effective coverage. In heterogeneous RBS deployment, user nondeterminacy of overlapped cells illustrates user mobility pattern in a certain area. In low user nondeterminacy area, micro cells tend to keep active since they are more energy efficient at the same traffic level. And to avoid frequent switching, the RBSs with high temporal homogeneity have high potential to keep active. Cell breathing is a coverage optimization mechanism to reduce energy consumption. The RBSs

are switched off to reduce their energy consumption and their neighboring cells are zoomed in to meet the traffic demand. Clustering RBSs with the combination of traffic fluctuation correlation and geographic distance can be used to identify the cells that are to zoom out, and the cells that are to zoom in or can be switched off. Fig.5 shows cells are clustered to four groups according traffic fluctuation within a district of Chongqing. RBSs of different groups, which have complementary traffics, can be operated together to improve energy efficiency. Another approach of cell layout adaptation is relay network. Relaying can be performed by using repeater stations or mobile devices as relays. A relay network with low user nondeterminacy is stable and more efficient.

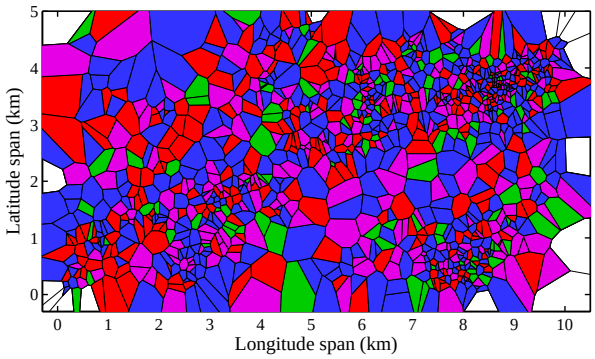


Figure 5: Cells are clustered to four groups according traffic fluctuation within a district of Chongqing

Social characteristics are interactive with multi-layer RBSs operations. The dynamics of RBSs' operation (e.g., switch on/off, zoom in/out) can affect social characteristics in spatial-temporal dimensions, which in turn improve the accuracy of modeling wireless network and refine the energy efficient designs of the networks.

Conclusions

In this paper, we proposed a new model to characterize the social features of individual RBSs. Key measures of traffic fluctuation, user nondeterminacy, temporal homogeneity and usage diversity, are defined.

Multi-dimensional clustering can be carried out based on these social measures to categorize cellular base stations. We also established social models for each of the categories and derived important model parameters.

The applications to energy efficient wireless networks were studied. The proposed models are of practical value and can facilitate designing, simulating, and evaluating network deployment.

Acknowledgment

This work was supported by the National Basic Research Program of China under the Grant No.2013CB329606 and the China Scholarship Council.

Bibliography

[1] Index, Cisco Visual Networking (2014), Global mobile data traffic forecast update, *White Paper, February*, 2013-2018.

[2] P. Zerfos, X. Meng, S. HY Wong, V. Samanta, S. Lu (2006), A study of the short message service of a nationwide cellular network, *Proc. of the 6th ACM SIGCOMM conference on Internet measurement*, 263-268.

[3] D. Willkomm, S. Machiraju, J. Bolot, A. Wolisz (2008), Primary users in cellular networks: A large-scale measurement study, *New frontiers in dynamic spectrum access networks, 2008. DySPAN 2008. 3rd IEEE symposium on*, 1-11.

[4] A. Klemm, C. Lindemann, M. Lohmann (2001), Traffic modeling and characterization for UMTS networks, *Global Telecommunications Conference, 2001. GLOBECOM'01. IEEE*, 3: 1741-1746.

[5] Y. Zhang, A. Årvidsson (2012), Understanding the characteristics of cellular data traffic, *ACM SIGCOMM Computer Communication Review*, 42(4): 461-466.

[6] U. Paul, A. P. Subramanian, M. M. Buddhikot, S. R. Das (2011), Understanding traffic dynamics in cellular data networks, *INFOCOM, 2011 Proceedings IEEE*, 882-890.

[7] D. Lee, S. Zhou, X. Zhong, Z. Niu, X. Zhou, H. Zhang (2014), Spatial modeling of the traffic density in cellular networks, *Wireless Communications, IEEE*, 21(1): 80-88. 80-88.

[8] M. C. González, C. A. Hidalgo, A. Barabási (2008), Understanding individual human mobility patterns, *Nature*, 453(7196): 779-782.

[9] C. Song, Z. Qu, N. Blumm, A. Barabási (2010), Limits of predictability in human mobility, *Science*, 327(5968): 1018-1021.

[10] X. Zhang, Y. Zhang, R. Yu, W. Wang, M. Guizani (2014), Enhancing spectral-energy efficiency forLTE-advanced heterogeneous networks: a users social pattern perspective, *Wireless Communications, IEEE*, 21(2): 10-17.

[11] Y. Huang, X. Zhang, J. Zhang, J. Tang, Z. Su, W. Wang (2014), Energy Efficient Design in Heterogeneous Cellular Networks Based on Large-Scale User Behavior Constraints, *IEEE Transactions on Wireless Communications*, 13(9): 4746-4757.

[12] D. Hu, B. Huang, L. Tu, S. Chen (2015), Understanding Social Characteristic from Spatial Proximity in Mobile Social Network, *International Journal of Computers Communications & Control*, 10(4): 539-550.

[13] Z. Qiao, P. Zhang, Y. Cao, C. Zhou, L. Guo, B. Fang (2014), Combining Heterogenous Social and Geographical Information for Event Recommendation, *The Twenty-eighth AAAI Conference.*

[14] I. Trestian, S. Ranjan, A. Kuzmanovic, A. Nucci (2009), Measuring serendipity: connecting people, locations and interests in a mobile 3G network, *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, 2009: 267–279.

[15] U. Von Luxburg (2007), A tutorial on spectral clustering, *Statistics and computing*, 17(4): 395-416.

[16] J. Shi, J. Malik (2000), Normalized cuts and image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8): 888-905.

[17] A. Y. Ng, M. I. Jordan, Y. Weiss (2002), On spectral clustering: Analysis and an algorithm, *Advances in neural information processing systems*, 2: 849-856.

[18] J. A. Hartigan, M. A. Wong (1979), Algorithm AS 136: A k-means clustering algorithm, *Applied statistics*, 100-108.

[19] M. Greenacre (2010), Chapter 6 Measures of distance and correlation between variables, *Correspondence analysis in practice*.

[20] L. Suarez, L. Nuaymi, J. Bonnin (2012), An overview and classification of research approaches in green wireless networks, *EURASIP Journal on Wireless Communications and Networking*, 142, DOI: 10.1186/1687-1499-2012-142.

[21] X. Feng, et al. (2015), Feedback analysis of interaction between urban densities and travel mode split, *International Journal of Simulation Modelling*, 14(2): 349-358.

[22] X. Deng, et al. (2013), A social similarity-aware multicast routing protocol in delay tolerant networks, *International Journal of Simulation and Process Modelling*, 8(4):248-256.

[23] M. A. Piera, R. Buil, M. M. Mota (2014), Specification of CPN models into MAS platform for the modelling of social policy issues: FUPOL project, *International Journal of Simulation and Process Modelling*, 9(3):195-203.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(4):493-506, August 2016.

A Dimension Separation Based Hybrid Classifier Ensemble for Locating Faults in Cloud Services

M.J. Peng, Y. Yue, B. Li, C.Y. Wang

Min-jing Peng*, Bo Li
Institute of E-commerce and Public Informational Services, Wuyi University
Jiangmen 529020, Guangdong, China
2586961312@qq.com, 15819748999@139.com
*Corresponding author: 15819748999@139.com

Yun Yue, Chun-yang Wang
School of Economics and Management, Wuyi University
Jiangmen 529020, Guangdong, China
1146942842@qq.com, 243717652@qq.com

Abstract: Cloud services provide Internet users with various services featured with data fusion through the dynamic and expandable virtual resources. Because a large amount of data runs in different modules of the cloud service systems, it will inevitably produce all kinds of failures when the data is processed in and transferred between modules. Therefore the job of rapid fault location has an important role in improving the quality of cloud services. Because of the features of large scale and data fusion of data in the cloud service system, it is difficult to use the conventional fault locating method to locate the faults quickly. Taking the requirements on the speed of locating faults into account, we will make a clear division to all possible failure causes according to the business phases, and quickly locate the faults by implementing a cascading structure of the neural network ensemble. At last, we conducted an experiment of locating faults in a cloud service system runned by a telecom operator, comparing the proposed hybird classifier ensemble with neural networks trained by separated data subsets and a conventional neural network ensemble based on bagging algorithm. The experiment proved that the neural network ensemble based on dimension separation is effective for locating faults in cloud services.
Keywords: fault locating, hybird classifier ensemble, dimension separation, cloud service, data fusion, neural network.

1 Introduction

Cloud service is a mode of increasing, applying and delivering service provided through the Internet with dynamic and virtualized resources [1]. Cloud services are easily expandable and delivered in accordance with users' requirements [2]. These services are typically related with the fields of IT, software and Internet.

In the general cloud service systems, the data of account information, user resources, credit status, billing data, order information and other transaction data needing to be fused. In spite of the convenience of business support, data fusion brings problems of oversize of data storage, complexity of data processing and frequent data transferring [3]. These problems will inevitably lead to more failures of cloud service systems [4].

The speed of locating fault is an important measure to maintain and improve the quality of cloud services [5]. However, due to the large data size and complexity of fused data, conventional methods of customer feedback, regular inspection, data audit and other methods delay the progress of locating faults. As for common intelligent classification algorithms, the whole cloud system is taken as a black box, even if the fault can be detected, it would be very hard to locate it.

Taking into account the requirement of rapid locating, we make a clear division to all kinds of possible failure causes according to the business phases, and quickly locate faults by implementing a cascading structure of the classifier ensemble. At last, we conducted an experiment of locating faults in a cloud service system running by a telecom operator in China, comparing the proposed ensemble with neural networks trained with data subsets of dimension separation and a conventional neural network ensemble. The experiment proved that the proposed hybrid classifier ensemble based on dimension separation is effective for the rapid location of cloud service failure.

2 Related works

There are two kinds methods of locating faults in information systems: (1) Regular methods; (2) Pattern recognition methods of artificial intelligence.

2.1 Regular methods

Conventional methods of locating faults include customer feedbacks, regular inspection and data auditing.

Despite the fact that the customer feedbacks are helpful in locating the faults, the fault location based on customer feedback has three drawbacks: (1) time delaying; (2) hurt on user experience; (3) high costs. The negative sides hinder the application of this method in cloud service systems [4].

Regular inspection is another way to find the fault in the cloud service system. For cloud service system operators, regular inspection on the important system module is a must. There are two the problems for regular inspection: (1) the boring work of inspection leads to low efficiency and high error rate; (2) it is hard to find the complex problems.

Data auditing is a method of verifying the failure based on the data generated by the system. This method is generally based on the reports and statistics services. It can only be used to find the faults meeting with two conditions: (1) the faults and the causes are acknowledged; (2) data auditors are experienced with the faults. It is not effective in using the method in locating faults caused by the uncertain problems, despite that facts that this method is simple, accurate and speedy in application.

2.2 Pattern recognition based on artificial intelligence

The problem of locating faults using pattern recognition method is eseentially a problem of classification. There are many classification algorithms, such as artificial neural network, support vector machine and wavelet analysis [6]. In general, for the problem of locating faults in cloud service systems, neural network (shown in Fig.1) is commonly used [7].

In Fig.1, there are following problems existing in the general model of locating faults based on neural network:

- Requiring large number of data samples. In the task, all types of faults are located using the same neural network, so it is necessary to provide a large number of training samples, which hinders the application of the model in practical applications. In order to train a neural network for locating faults in cloud services, the input data should include all kinds of negative samples and the positive samples, without which the appropriate neural network parameters could not be trained during the training process. Therefore, we must obtain a large number of samples to locate the fault types in the system, which leads to a great increase in the operating costs of cloud services.

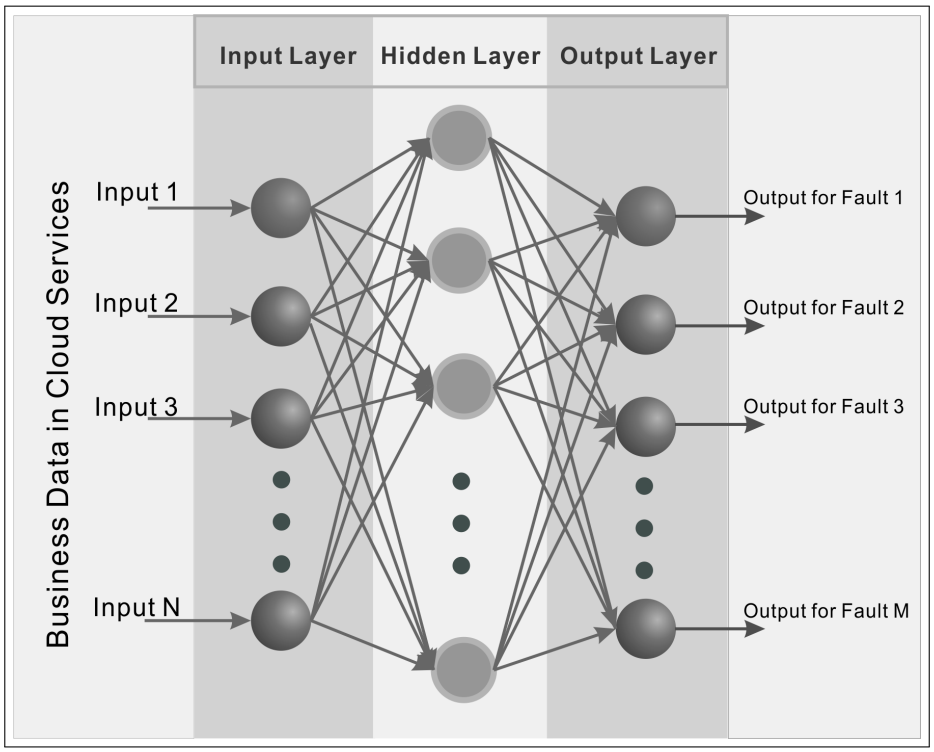


Figure 1: The general model for locating faults based on neural network

- Increasing the complexity of computation. As each cause of system failure is represented by one or serveral data dimension, the application of locating faults based on neural network requires high dimensional data. The neural network trained by the high dimensional data will have a very complex structure, which leads to a substantial increase in the amount of computation, and greatly reduces the accuracy of locating faults. Consequently, it will increase costs of subsequent failure recognition and troubleshooting.

- Lacking of explanation for the causes of system faults. It is hard to establish a clear causal relationship between the failure phenomenon (the output of the neural network) and the cause of the fault (the input of the neural network) through the fault locating neural network. Therefore it is very difficult to accurately locate the system failure because that we can't confirm the relationship between the output data and the input data.

3 Framework

In order to reduce the training sample size, simplify the computational complexity and improve the explanation of locating faults using neural networks, we need to reduce the dimension of data. Dimension reduction is based on the correlation of different data dimensions. When the correlation between the data dimensions is low, simple dimension reduction will lead to the loss of information supported by the data.

The approach used in this research is to divide the original high dimensional data set into several low dimensional data sets according to the business relationship. Then, a neural network is allocated for each of the low dimensional data sets separated dimensionally. And the corresponding neural network is trained and tested with the low dimensional data set. In order

to reduce the computational complexity and improve the recognition efficiency, we use the cascading structure to integrate the various neural networks to form the neural network ensemble based on dimension separation as shown in Fig. 2.

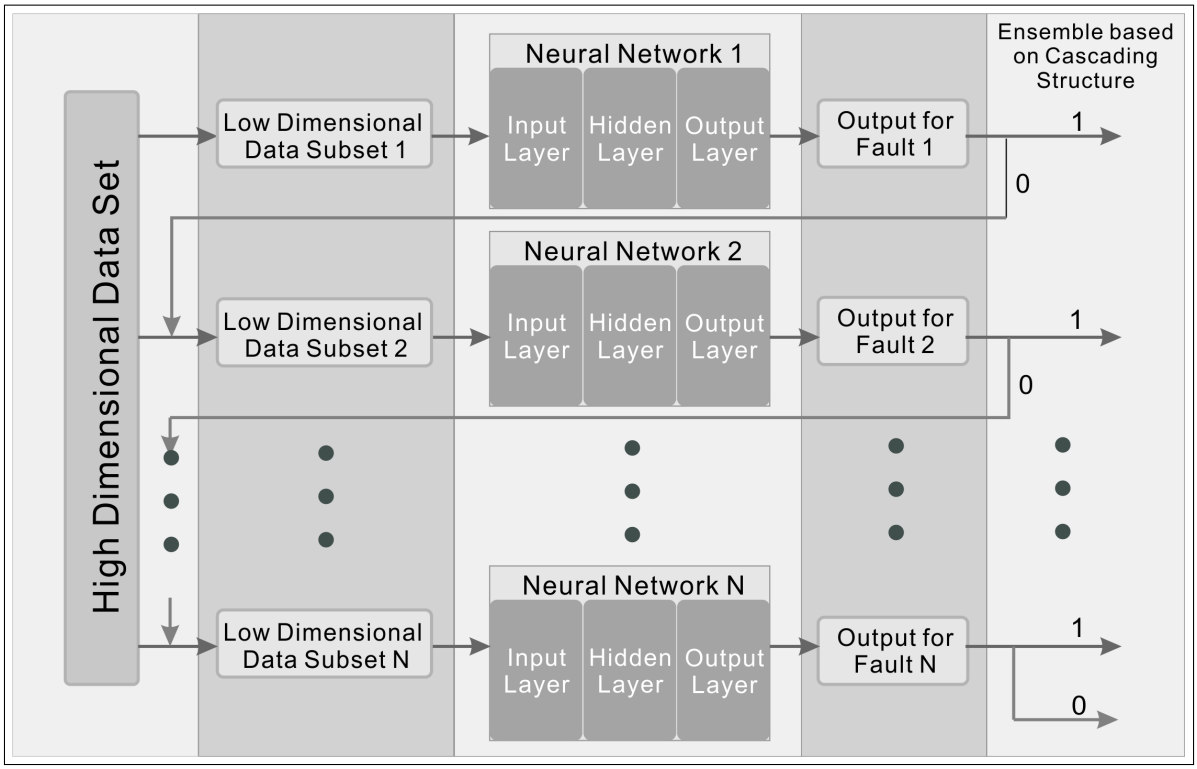


Figure 2: Schematic diagram of fault locating neural network ensemble based on dimension separation

In the cascading ensemble structure, if the output of the current neural network is 1, which indicates that there is an error with the transaction corresponding to the neural network. Therefore, there is no need to carry out the further fault location. Otherwise, the next neural network and the corresponding low dimensional data subset will be set for locating further faults.

Compared with the single neural network, there are three advantages in the proposed ensemble based on dimension separation:

- The work efficiency of the proposed ensemble is higher. Because that the high dimensional data is divided into low dimensional data subsets, each neural network in the ensemble needs only be trained and tested by a data subset, which greatly improves the work efficiency of the classifier based on the ensemble.

- The reason causing the fault is clearer. Because different neural networks in the proposed ensemble are built according to different fault types, the fault reason can be determined according to the neural network corresponding to the output data.

- It is easier for the ensemble than the single neural network to expand. The neural network ensemble could be added with more neural networks without changing the fundamental structure, which means it can be used to locate a new fault when it is needed.

The defect of this study is that we need to determine the fault types and to train the corresponding artificial neural networks in order to setup the proposed neural network ensemble,

which requires the application supporters to understand the business logics of the system and corresponding failure reasons.

4 Hybrid classifier ensemble based on dimension separation

The speeding requirement of fault locating requires that we should locating the fault occurring frequently first. The premise of satisfying this requirement is that all kinds of faults can be clearly divided, so we adopt the method of dimension separation, based on the cascading structure, to integrate the neural networks and other classifiers, in which the fault causes and fault phenomena are correlated to achieve the goal of locating faults [8].

The algorithm of hybrid classifier ensemble based on dimension separation involves two steps: dimension separation and neural network ensemble.

4.1 Dimension separation

In this research, various neural networks are integrated in ensemble structure. These different neural networks have their own responsibilities. They only receive data subsets with dimensions related to the corresponding transaction failures. The original high dimensional samples are separated into the low dimensional data. The separated dimensions of the data subsets are determined based on different transaction phases.

The N-dimensional data $A = \{B_1, B_2, B_3, \cdots, B_N\}$ is separated into $A_1, A_2, A_3, \cdots, A_m$ as shown in equation (1).

$$\begin{aligned} A_1 &= \{B_i, B_j, B_k, \cdots, B_l\} \\ A_2 &= \{B_m, B_n, B_o, \cdots, B_p\} \\ &\cdots \\ A_M &= \{B_i, B_k, B_m, \cdots, B_n\} \end{aligned} \tag{1}$$

In equation (1),$\{i, j, k, l, m, n, o, p\} \in \{1, 2, 3, \cdots, N\}$. The data dimension subscripts in the data subsets are not necessarily continuous relative to the original N-dimensional data set. And the dimension subscripts of different subsets allow to be repeated.

In this research, we divided the related transaction process in the cloud service system into five phases: the ordering phase, the using phase, the billing phase, the accounting phase and the grading phase. For transaction data, we can separate the related dimensions of the original business data into corresponding data subsets in accordance with these five phases as shown in Fig. 3.

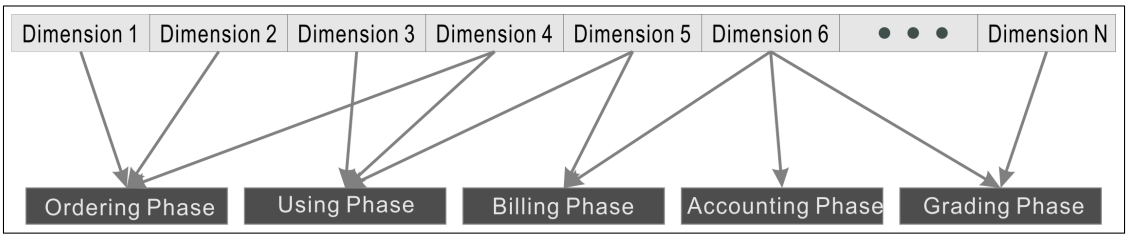


Figure 3: Dimension separation diagram

4.2 Neural network ensemble

The general understanding of the theory of neural network ensemble is: on the premise that the generalization errors of the neural networks in the ensemble are stable, it can effectively enhance the generalization ability of neural networks, and reduce the errors of the neural networks [9].

In general, there are two ways to deversify the neural networks in ensemble: one way is to increase the differences in the structure of neural networks, for example, set different numbers of hidden layers or choose different objective functions; the other way is to provide completely different training data for different neural networks in the ensemble based on the weak learning theory and the boosting techniques.

Neural network ensembles are different with different functions. When the neural network is used to solve the classification problem, the output result of the neural network can be processed by simple average or weighted average. Perrone's research suggests that the weighted average can promote the generalization ability of the neural network ensemble [10], but some studies also show that the adjustment of weights in the process may negatively affect the generalization ability of neural network [11].

In a dynamic weighted ensemble neural network with n neural networks, the training dataset is $A = \left\{\left(x_1, y_1\right), \left(x_2, y_2\right), \cdots, \left(x_n, y_n\right)\right\}$. And the output value of one neural network in the ensemble is $y = f_i\left(x\right)$. The output of the ensemble is given in equation (2).

$$f_o = \sum_{i=1}^{n} w_i f_i\left(x\right) \tag{2}$$

In equation (2), the value of weight w_i could be set to $1/n$ by simple average.

The absolute majority voting method is usually used in the classification of the neural network ensemble, which means the decision of a classification result is determined equally by every neural network in the ensemble. If more than half the neural network classifiers are prone to a result, the result would be used as the classification result of the ensemble. If the probability of voting for the right result for each neural network classifier is $1-p$, and the result of each neural network is not affected by other neural networks, the probability of voting for the right results for the ensemble consisting of N neural networks would be given in equation (3).

$$P_{true} = 1 - \sum_{k>N/2}^{N} \binom{N}{k} p^k \left(1-p\right)^{N-k} \tag{3}$$

Obviously, when $p < 1/2$, N and P_{true} are positively correlated with each other, which means the probability of voting for the right result for the ensemble increases with the increase of the number of neural networks.

4.3 Hybrid classifier ensemble

After the separation of dimensions and the training of neural networks, the neural networks can be integrated into an ensemble structure. Considering the computational accuracy and complexity in locating faults, we choose the cascading structure as the ensemble structure. The cascading structure consists of many layers, where each independent neural network can be put. The neural networks can only be trained and tested with the subsets with fewer dimensions separated from the original data. The output of the neural network in the nst layer can be the input parameters of the $n+1$st layer. The independence of the layers in ensemble structure determines the degree of freedom of selecting algorithms put into the layers in the ensemble.

Besides neural networks, other forms of classification algorithm can also be integrated into the ensemble. Here, according to the features of the large transaction volume and simple standard of classification for some faults locating in the cloud services, the algorithm of binary tree is introduced into the ensemble structure, forming a hybrid classifier ensemble of binary trees and neural networks [12].

In the task of locating faults in cloud services, part of the transaction logic is very simple, or can be completed according to the existing disciminant rules. Therefore, there is no need to use the neural networks. For this type of fault locating with simple transaction logic, classification method based on binary tree can be employed [11].

The algorithm of binary tree is very simple. There is no need for complex training process in the binary tree. The application of the tree only needs a discriminant function. For example, for the input data subset $A = (x_1, x_2, x_3, \cdots, x_n)$, if the computational result of the function satisfies the condition of K, we would be able to locate the fault in the cloud service system as shown in equation (4).

$$y = \begin{cases} 1 & \text{if } f(A) \text{ satisfies } K \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

For some other nonlinear and complex samples not suitable for neural network or binary tree, we can use other classifiers to integrate the classification results as a layer in the ensemble.

The ensemble integrating binary trees, neural networks and other classifier can locate various faults in cloud service systems [13], as shown in Fig. 4.

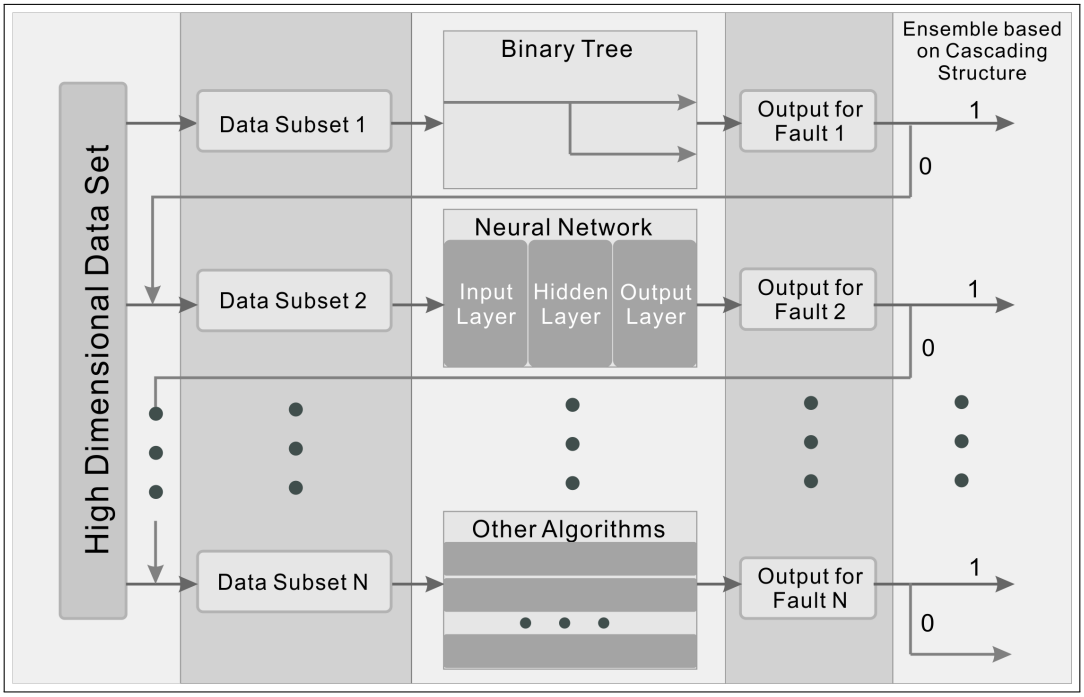


Figure 4: Hybrid Classifier Ensemble

5 Experiment

5.1 Dimension selection

There are a wide variety of transactions in cloud service systems. In this experiment, we monitor and obtain the data from the ordering phase, the using phase, the billing phase, the accounting phase and the grading phase of a transaction for locating the faults.

The basis for selecting what kind of business data is the transactional features. For example, there are many fields involving in the ordering phase of a transaction in a cloud service. These fields involve the process and its details. Therefore, we need to obtain the fields related with locating faults in transaction from the database of the cloud service based on the transactional features for locating faults.

Data values of obtained fields involved in related aspects of the transaction process are listed in Tab. 1. In the table, the letters O, U, B, A and G respectively represent the ordering phase, the using phase, the billing phase, the accounting phase and the grading phase of a transaction.

Table 1: Related fields of five phases of a transaction

Field Name	O	U	B	A	G	Field Name	O	U	B	A	G
Field 1	■					Field 21			■		
Field 2	■	■				Field 22			■		
Field 3		■	■			Field 23			■		
Field 4		■	■			Field 24			■	■	
Field 5		■	■			Field 25			■		
Field 6		■	■			Field 26				■	
Field 7		■	■			Field 27				■	
Field 8		■	■			Field 28				■	
Field 9		■	■			Field 29				■	
Field 10		■	■			Field 30				■	
Field 11		■				Field 31				■	
Field 12		■				Field 32				■	
Field 13		■				Field 33				■	
Field 14		■				Field 34					■
Field 15						Field 35					■
Field 16						Field 36					■
Field 17		■				Field 37					■
Field 18		■				Field 38					■
Field 19		■				Field 39					■
Field 20		■									

In the process of data collection, the user identity is the unique identifier. The related data of the user in every phase are collected with comments and foreign keys excluded.

In this experiment, 39 fields in Tab. 1 are collected. The dimensions corresponding to the fields are separated according to the transaction phases. Then the separated dimensions are used in the training and testing steps.

Some setup values are given in Tab. 2.

5.2 Neural network ensemble with high dimensional data

The common neural network training and testing methods are used in the ensemble for the high dimensional data. And the majority voting method is used to integrate the results of

neural networks to get the ensemble results to locate the faults in the cloud service system. 28 neurons are determined to setup in the hidden layers after "trial and error" tests. And four neural networks are created to be layers in the ensemble. Therefore, the four neural networks are trained based on the high dimension samples respectively. And the test data is used to test the validation of the neural networks respectively. The classification results are integrated to obtain fault locating results by the majority voting method.

In order to diversify the four neural networks to be integrated, the 6001 training samples are tripled to 18003 samples. Then 6001 samples were randomly selected to train each neural network. After the completion of the training, 900 test samples were tested on each neural network respectively. And test confusion matrices of the four neural networks are shown in Fig. 5. Correct rates of each phases of the ensemble are 0.278, 0.833, 0.882, 0.955 and 0.962. The overall correct rate of the ensemble is 0.679.

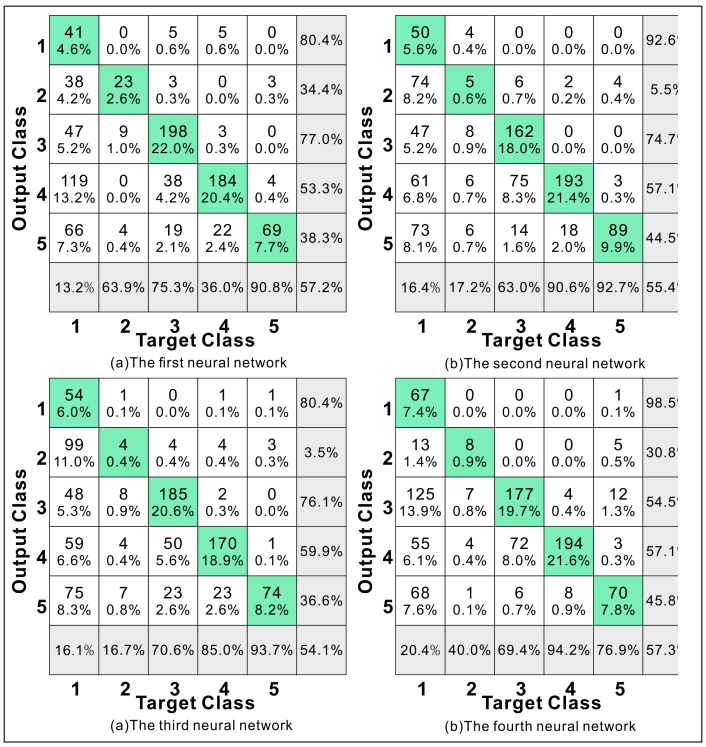


Figure 5: Test confusion matrices of four neural networks for ensemble with high dimensional data

Table 2: Setup values of the experiment

Item	Value
Platform	MatLab 2011a
Neural network tool	patternnet()
trainFcn	'trainscg'
performFcn	'crossentropy'

5.3 Neural networks with low dimensional data

As the rule for locating faults in the ordering phase is very simple, a binary tree is used to locate fault by implementing the rule on the data subset related with the ordering phase.

Four classification neural networks are created by using the function "patternnet()" corresponding to the other four data subsets after dimension separation according to the dimension relations in Tab. 5. The dimension numbers of the data subsets corresponding to the four phases are 17, 13, 9 and 6 respectively. After some " trial and error", 10 neurons are determined to be in the hidden layers in the four neural networks, in order to obtain expected training and testing results.

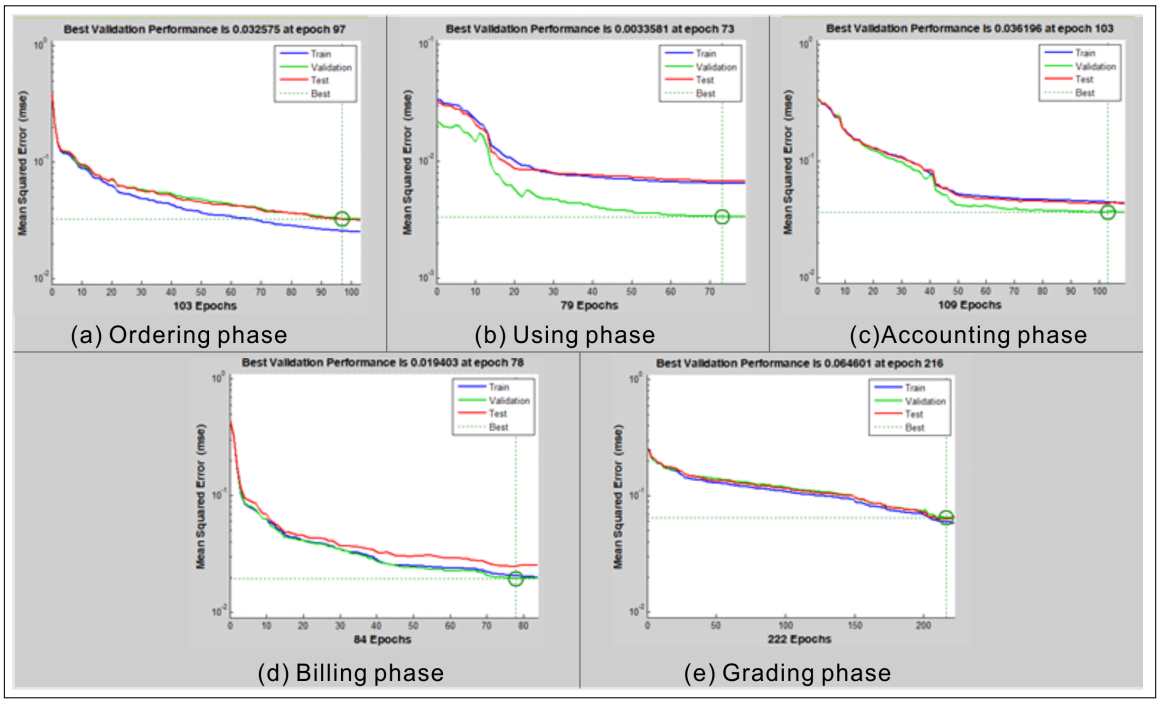


Figure 6: The training processes of neural networks for five business phases

The four created neural networks are respectively trained on the corresponding data subsets separated dimensionally. And the training processes in Fig. 6 are obtained by using the "plotperform()" function.

900 records of the data subsets being dimensionally separated are respectively used to test the binary tree and the four trained neural networks. The test confusion matrices are given in Fig. 7. According to the test results, the accuracy of the grading phase is about 92

5.4 Analysis of experiment results

In this experiment, we compared the common neural network ensemble with the dimension separation based hybrid classifier ensemble in the task of locating faults in a cloud service system. It was found that the dimension separation based hybrid classifier ensemble performances better than the common neural network ensemble, and the experimental results are shown in Tab. 3.

Test confusion matrices in Fig. 5 and Fig.7 give the numbers of true positive (TP), false positive (FP), false negative (FN) and true negative (TN) samples of different classifiers and different layers in a classifier. The sensitivities of TP, FP, FN and TN are respectively true positive rate (TPR), false positive rate (FPR), false negative rate (FNR) and true negative rate

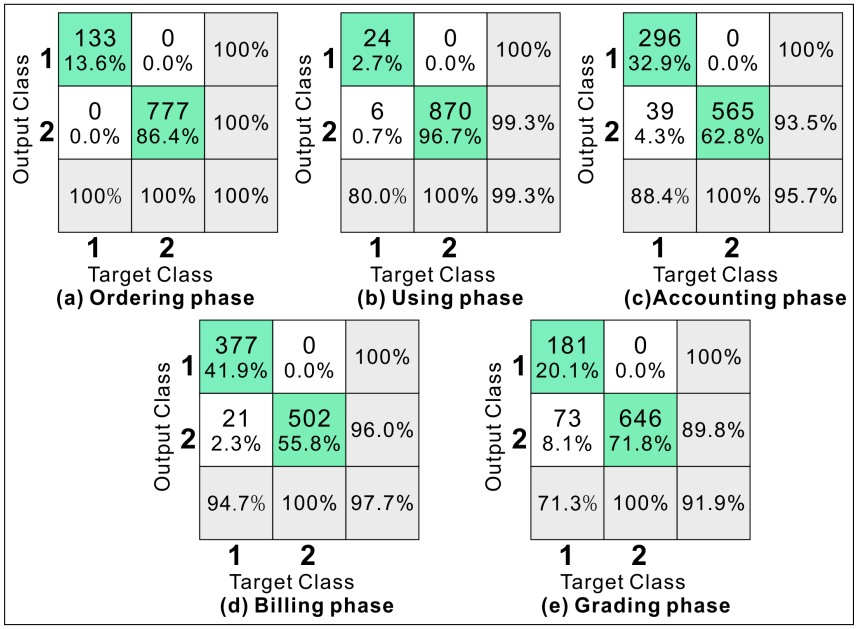


Figure 7: Test confusion matrices of the binary tree and the neural networks trained with low dimensional data subsets

Table 3: Comparison of common neural network ensemble,dimension separation based neural networks and dimension separation based neural network ensemble

	common neural network ensemble	dimension separation based classifiers	dimension separation based hybird classifier ensemble
Accuracy of Ordering Phase	0.278	1.000	
Accuracy of Using Phase	0.833	0.993	
Accuracy of Accounting Phase	0.882	0.957	
Accuracy of Billing Phase	0.955	0.977	
Accuracy of Grading Phase	0.962	0.919	
Overall Accuracy	0.679	0.969	0.979

(TNR). The values of TP and TN of the layers in hybrid classifier ensemble of cascading structure are given in equation (5). In the equation, i is the number of the layer containing the classifier.

$$TP_{i+1} = TPR_{i+1} \cdot (FN_i + TN_i) \tag{5}$$

And the overall accuracy of the proposed ensemble is given in equation (6). In the equation, n is the total number of layers of the propose ensemble.

$$A_{Overall} = \frac{(TN_1 + TP_1) + (TN_2 + TP_2) + ... + (TN_n + TP_n)}{(TN_1 + TP_1 + FN_1 + FP_1) + (TN_1 + FN_1) + ... + (TN_{n-1} + FN_{n-1})} \tag{6}$$

High accuracy of classification is the premise of troubleshooting of cloud services. The experimental results in Tab. 3 prove that the proposed hybrid classifier ensemble based on dimension separation is more suitable for locating faults in cloud service systems than the common neural network ensemble.

5.5 Applications

The application framework for the proposed hybrid classifier ensemble is given in Fig. 8.

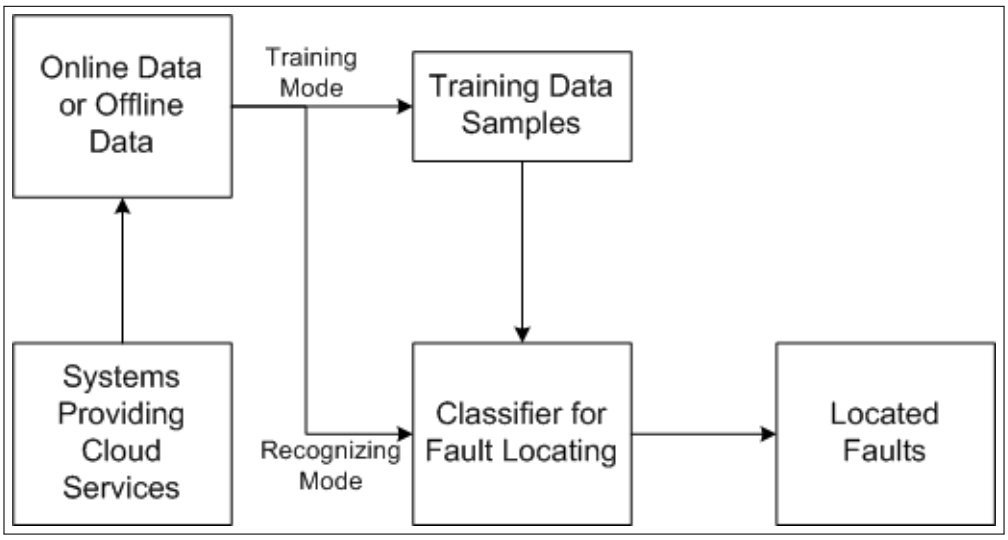


Figure 8: The application framework of the hybrid classifier for locating faults in cloud services

The data are collected from the cloud service systems. These data may be real-time online business data, or be offline business data stored in the data warehouse. In the training mode, data is organized into training sample sets for training the proposed classifier of fault locating. After training, the online and offline data collected from the cloud systems are to be used to locating faults of the cloud services by the proposed classifier.

According to the real-time feature of input data for the proposed classifier, the faults can be divided into two categories: (1) when the input is the real-time data, based on the input real-time data, the running cloud services and supporting systems are real-time monitored to detect the system faults; (2) when the input data are offline, we can audit the businesses through checking the logical relationship between data representing the businesses to find out the problems in business.

Conclusions and future work

Fault locating is an important job for supporting the operations of cloud services because of large data scale and data fusion. The key to fault location is to accurately locate the faults and clearly explain the causes. On the premise that the causes of the faults are complex, the neural network becomes an important method for fault location. Common neural network has two problems: (1) the low accuracy of fault location; (2) unexplainable fault locating results.

Considering that the neural network has higher classification accuracy for the data with the fewer dimensions, we divide business data into data subsets according to the business phases through dimension separation. And then a binary tree and four neural networks corresponding to the business phases are created, trained and integrated into the cascading ensemble structure for locating faults. Experiments show that the proposed hybrid classifier ensemble based on dimension separation is more accurate and explainable in locating faults in cloud service systems than common neural network ensemble.

The problem of this research is the architecture of the proposed ensemble is more dependent on the familiarity level of the business supporters of the cloud service system, because the works of obtaining data obtain and separating dimensions are both dependent on the transaction environments in real world.

Due to the openness of cascading ensemble structure, we could integrate binary trees or other suitable classification algorithms into the proposed ensemble based on dimension separation by replacing existing neural networks with other algorithms, according the real environments of the cloud service system.

The proposed fault location classifier can be used in two aspects: (1) real-time monitoring of the cloud service system for locating faults in the operation; (2) logic matching on the business data for detecting business problems of cloud services.

Acknowledgment

This research is supported by National Science Foundation of China (Grant No. 71203162), Science and Technology Planning Project of Guangdong Province, China (Grant No. 2014B040404072), Natural Science Foundation of Guangdong Province, China (Grant No. 2015A030313642) and Innovation Project of Wuyi University (Grant No. 2014KTSCX128 and 2015KTSCX144).

Bibliography

[1] Sun, L., Dong, H., Hussain, F.K., Chang, E. (2014); Cloud Service Selection: State-of-the-art and Future Research Directions, *Journal of Network and Computer Applications*, ISSN 1084-8045, 45: 134-150.

[2] Noor, T.H., Sheng, Q.Z., Ngu, A.H.H., Dustdar, S. (2014); Analysis of Web-Scale Cloud Services, *IEEE Internet Computing*, ISSN 1089-7801, 18(4): 55-61.

[3] Breiter, G., Behrendt, M. (2009); Life Cycle and Characteristics of Services in the World of Cloud Computing, *IBM Journal of Research and Development*, ISSN 0018-8646, 53(4): 1-8.

[4] Gu, Y., Wang, D.S., Liu, C.Y. (2014); DR-Cloud: Multi-Cloud Based Disaster Recovery Service, *Tsinghua Science and Technology*, ISSN 1007-0214, 9(1): 1-13.

[5] Chauvel, F., Song, H., Ferry, N., Fleurey, F. (2015); Evaluating Robustness of Cloud-Based Systems, *Journal of Cloud Computing: Advances, Systems and Applications*, ISSN 2192-113X, 4(18): 1-17.

[6] Ren, J. (2012); ANN vs. SVM: Which One Performs Better in Classification of MCCs in Mammogram Imaging, *Knowledge-Based Systems*, ISSN 0950-7051, 26(1): 144-153.

[7] Ahn, B.S., Cho, S.S., Kim, C.Y. (2000); The Integrated Methodology of Rough Set Theory and Artificial Neural Network for Business Failure Prediction, *Expert Systems with Applications*, ISSN 0957-4174, 18(2): 65-74.

[8] Webb, G.I., Zheng, Z. (2004); Multistrategy Ensemble Learning: Reducing Error by Combining Ensemble Learning Techniques, *IEEE Transactions on Knowledge and Data Engineering*, ISSN 1041-4347, 16(8): 980-991.

[9] Zhou, Z.H., Wu, J.X., Tang, W. (2002); Ensembling Neural Networks: Many Could Be Better Than All, *Artificial Intelligence*, ISSN 0004-3702, 137(1-2): 239-263.

[10] Perrone, M.P., Cooper, L.N.(1993); When Networks Disagree: Ensemble Method for Neural Networks. In: *Mammone, R.J. (ed.) Artificial Neural Networks for Speech and Vision*, Chapman & Hall, New York, ISBN 978-041-25-4850-5.

[11] Mui, J.K., Fu, K.S.(2014); Automated Classification of Nucleated Blood Cells Using a Binary Tree Classifier, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, ISSN 0162-8828, 2(5): 429-443.

[12] Hosseini, K.(2015); Accurate Hybrid Method for Rapid Fault Detection, Classification and Location in Transmission Lines using Wavelet Transform and ANNs, *International Journal of Scientific Engineering and Technology*, ISSN 2277-1581, 4(5): 329-334.

[13] Dounias, G., Linkens, D. (2004); Adaptive Systems and Hybrid Computational Intelligence in Medicine, *Artificial Intelligence in Medicine*, ISSN 0933-3657, 32(3): 151-155.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(4):507-521, August 2016.

An Ontology to Support Semantic Management of FMEA Knowledge

Z. Rehman, C. V. Kifor

Zobia Rehman*
1. Faculty of Engineering and Management
Lucian Blaga University of Sibiu, Romania
2. Department of Computer Science
COMSATS Institute of Information Technology, Abbottabad, Pakistan
*Corresponding author: zobia.rehman@gmail.com

Claudiu V. Kifor
Faculty of Engineering and Management
Lucian Blaga University of Sibiu, Romania
claudiu.kifor@ulbsibiu.ro

Abstract: Risk mitigation has always been a special concern for organization's strategic management. Various tools and techniques have been developed to manage risk in an effective way. Failure Mode and Effects Analysis (FMEA) is one of the tools used for effective assessment of risk. It analyzes all potential failure modes, their causes, and effects on a product or process. Moreover it recommends actions to mitigate failures in order to enhance product reliability. Organizations spend their resources and domain experts make their efforts to complete this analysis. It further helps organizations identify the expected risks and plan strategies in advance to tackle them. But unfortunately the analysis produced after spending a lot of organizational assets and experts' struggles, is not reusable due to its natural language text based description. Information and communication technology experts proposed some solutions but they are associated with some deficiencies. Authors in [13] proposed an ontology based solution to extract and reuse FMEA knowledge from the textual documents, and this article is the first step towards its implementation. In this article we proposed our ontology for Process Failure Mode and Effects Analysis (PFMEA) for automotive domain, along with its implementation, reasoning, and data retrieval through it.
Keywords: Ontology, FMEA, Knowledge Management, OWL, Protégé, SPARQL.

1 Introduction

Since ever we have been managing our knowledge as best as we could. In beginning the acquisition and storage of knowledge were the biggest issues but Information Technology turned the tables and now just one click on Google search button brings us pools of knowledge we desire. Ordinary gadgets can store Gigabyte to Terabytes of information and cloud storage offers access to logical storage of a physical storage spanning over multiple locations. On one hand these advances in technology are making storage and acquisition of knowledge easier and on the other hand it has become awfully essential for organizations to capture, store, apply and share their knowledge in a collective and systematic way, so they could survive and flourish in this knowledge based economy era. Thus the knowledge management has become a dire need of organizations and industry. They need to structure their knowledge and transform it into valuable competencies, products, and services for effective and well-timed utilization in order to make fruitful decisions. Risk management for an organization is a set of strategic level activities that maximizes the chances of objectives being achieved, by systematically understanding and

evaluating the project level risks. It has become a core part of organization's strategic management. Organizations spent the significant share of their investments in handling expected and unexpected risks on a project. Risk management is a process of risk identification, risk assessment, risk response and control development [6]. Risk assessment is a critical phase after risk identification as outcome of this phase leads to develop appropriate response and control for a risk. There are different tools available for risk assessment, e.g., scenario analysis for event probability and impact, risk assessment matrix, FMEA, probability analysis, and semi-quantitative scenario analysis. FMEA is a systematic tool to assess the risks associated to a product or process. It highlights all potential failure modes and their impacts on a product in advance so that they could be fixed timely in order to achieve desired goals. In 1960's for the very first time aerospace industry brought it into use during Apollo mission. In 1974, the US Navy developed MIL-STD-1629 about its use. Since late 1970's, it is in use in automotive industry for safety and reliability analysis. Nowadays this inductive reasoning tool is extensively being used in almost every engineering sector [14]. It finds all possibilities in which a product or process might go wrong (failure modes), determines the effect(s) of each failure mode along with severity, cause(s) of failure mode along with frequency(occurrence) and also the level of difficulty in order to detect a failure (detection). Further it calculates the Risk Priority Number (RPN) by multiplying magnitude values of severity, occurrence, and detection. Depending on the value of RPN, recommendations are determined and executed along with newly predicted values of severity, occurrence, detection, and RPN [15]. The knowledge produced during the process of

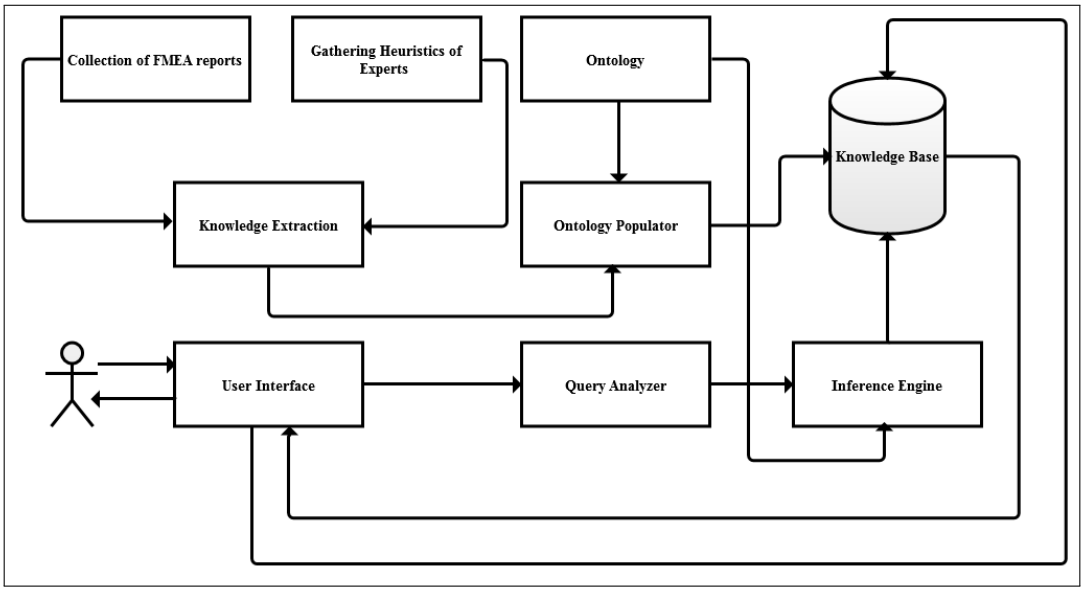


Figure 1: Conceptual architecture of ontology based knowledge management system for FMEA [13]

FMEA is really valuable. If it is adequately managed and re-utilized, it can reduce cost and effort. Although organizations spend huge cost and effort to apply FMEA method but knowledge acquired during this analysis is neither reused nor shared. Since it is not semantically organized, its interpretation varies from person to person and from situation to situation. It is usually found incomplete but larger in size due to its redundant production and that larger size makes it imprecise as well [9]. Artificial Intelligence proposed different solutions for this problem, e.g., rule based expert systems, case based reasoning, and knowledge based systems with ontological support. Among all these, knowledge based systems with ontological support are found most appropriate as rule based systems are not suitable if domain knowledge is larger enough, because

Table 1: FMEA Worksheet [16]

Process Failure Mode and Effects Analysis FMEA Number:_____ Team Leader:_____ Process Responsibility:_____ Prepared By:_____ FMEA Date (Orig.):_____ FMEA Date (Rev.):_____																	
Process Name	Potential Failure Mode	Effect(s) of Failure	SEV	Causes	OCC	Current Controls		DET	RPN	Recom. Action	Resp.& Comp. By	Action Result				
						Prevention	Detection					Action	SEV	OCC	DET	NRPN

its coding, verification, validation, maintenance, and inference through it becomes complex and time consuming [2]. In case based reasoning all probable cases are stored in case library with additional overhead of their attributes and references, moreover the information returned by such engines is not well formatted [8]. Available ontology based approaches to support FMEA also lack some significant aspects, e.g., in (Lee, 2001) authors presented FMEA only as a conceptual model without any inference and rules. Authors in [1] considered the discrepancies left by [7] and proposed a system, based on the combination of knowledge management and quality management concepts but it lacks functional taxonomy. In [10] authors discussed a better ontology based approach for FMEA procedure representation in lead free soldering but this proposal is still being worked on. Moreover no specific ontology is found to address the FMEA knowledge sharing and reuse for automotive domain. To address all these issues authors in [13] presented a conceptual architecture of their proposed system as given in figure 1 and this article in first step towards its implementation. This article is about the ontology we developed for representation and retrieval of PFMEA knowledge. Knowledge of interest used in this ontology is from automotive domain. Automotive engineering (vehicle engineering)deals with the design, manufacturing, and operation of vehicles(automobiles, buses, motorcycles etc.). It assimilates various components from different engineering sectors, e.g., mechanical, electrical, electronics, software, safety and quality engineering. From designing to production there are different kind of activities (belonging to different fields of engineering)are involved. Productivity of each activity heavily relies on past experiences, expertise of concerned people, and customer feedback. Better management of all this knowledge helps organizations improve time to market and customer satisfaction which not only helps earn better profit but brings sustainability for an organization in the market. In section 2 and 3, tools used to develop and query the ontology, are discussed. Section 4 describes the process of ontology development in detail, section 5 illustrates some query examples, and section 6 concludes the discussion with highlights of future work.

2 Protégé

Protégé is an open source software tool by Stanford University to develop domain models and knowledge based systems with ontology. It facilitates with both of the foremost means to model an ontology, e.g., frames and OWL (Web Ontology Language). It has built-in reasoning support for computing the inferred ontology class hierarchy and ontology consistency checking [13]. We used Protégé 5.0 beta version. Using it we developed our OWL ontology with RDF/XML format. RDF/XML format is a syntax defined by W3C to present RDF (Resource Description Format) graph by defining triples of subject, predicate, and object in XML (EXtensible Markup Language) format [12]. We developed ontology in OWL, the language beyond RDF schema that allows machines to perform more useful reasoning on its documents, and created its individuals (instances) in RDF. We got reasoning support from Protégé's built-in reasoner HermiT version 1.3.8.3. HermiT is an efficient reasoner based on "hypertableau"calculus that takes a few seconds to reason complex ontologies that are written in OWL. It reads an OWL file and determines the consistency of classes and their properties [4]

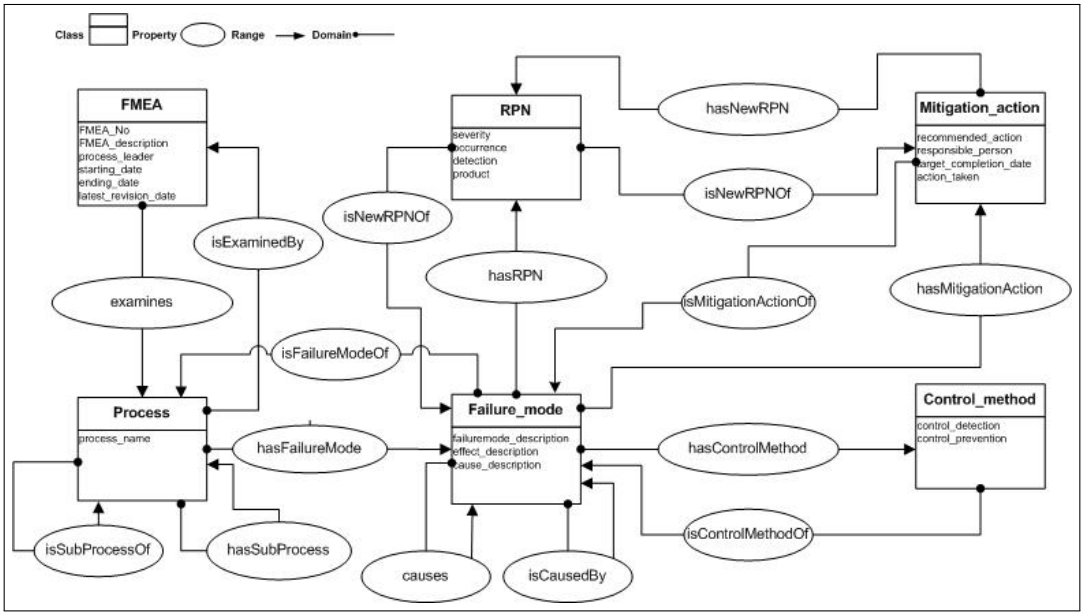


Figure 2: PFMEA ontology

3 SPARQL and Jena Fuseki Server

Jena is an open source Semantic Web framework for Java. It facilitates with an API (Application Program Interface) for extraction and writing data to/from RDF graphs. The graphs are represented as an abstract model, which can be sourced with data from files, databases, URLs or a combination of these. Fuseki is an http interface to RDF data that supports SPARQL (W3C recommended language to query directed labeled RDF graphs) for querying and updating RDF graphs. We used Apache Jena Fuseki server version 1.1.0. It provides REST-style SPARQL HTTP Update, SPARQL Query, and SPARQL Update by using the SPARQL protocol over HTTP [5]. It can be freely downloaded and runs on port 3030 in web browser.

4 Development of FMEA ontology

According to classic definition by [3] the ontology is an explicit specification of conceptualization. In information science perspective the ontology is a formal representation of the knowledge of a specific domain as a set of concepts within a domain, and the relationship between those concepts. It provides shared vocabulary and common (unambiguous) understanding of a domain and supports reasoning about the concepts. According to W3C definition, ontology is a vocabulary that defines concepts, relationship between concepts, and constraints on their usage in order to define and represent domain of discourse [11]. An ontology consists of different components, e.g., classes, properties (relationship of classes), and individuals (instances of the domain of discourse). In this section all these terms will be discussed in PFMEA perspective. In table 1 a PFMEA worksheet is shown. Using the PFMEA attributes given in that worksheet we designed an ontology given in figure 2.
This ontology is based on five different classes whereas all these classes are logically subsumed

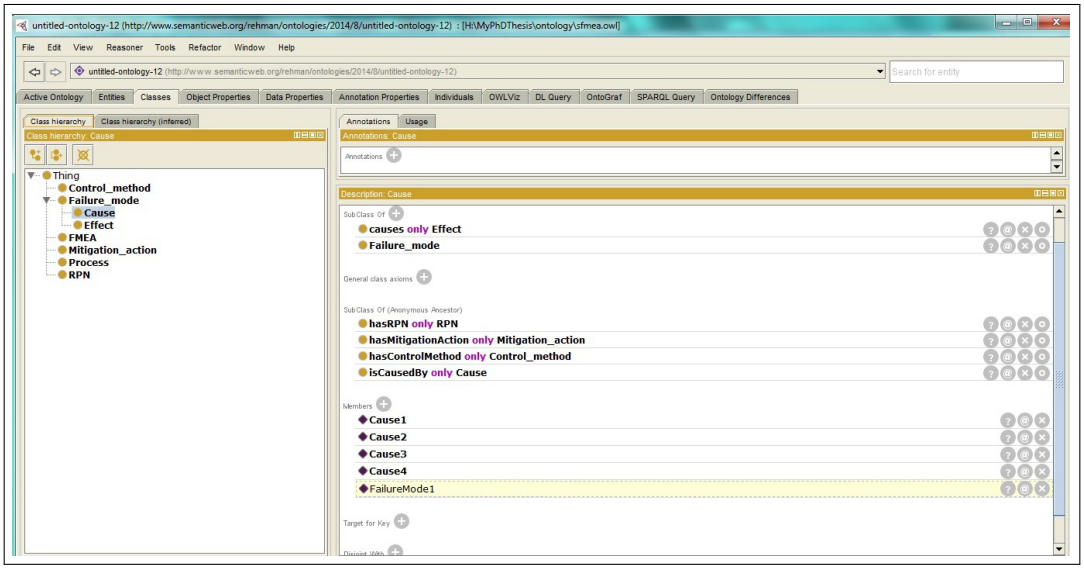


Figure 3: PFMEA classes and sub-classes in Protégé

sub-classes of the root concept "Thing". According to ISO-15926 "Thing"is a top-level ontology (collection of general concepts same in all knowledge domains) that subsumes abstract object and possible individual classes. Any immaterial object (which exists only as a concept) can be said an abstract object, e.g., information; and any material object (which exists in time and space) is known as possible individual, e.g., a pen.

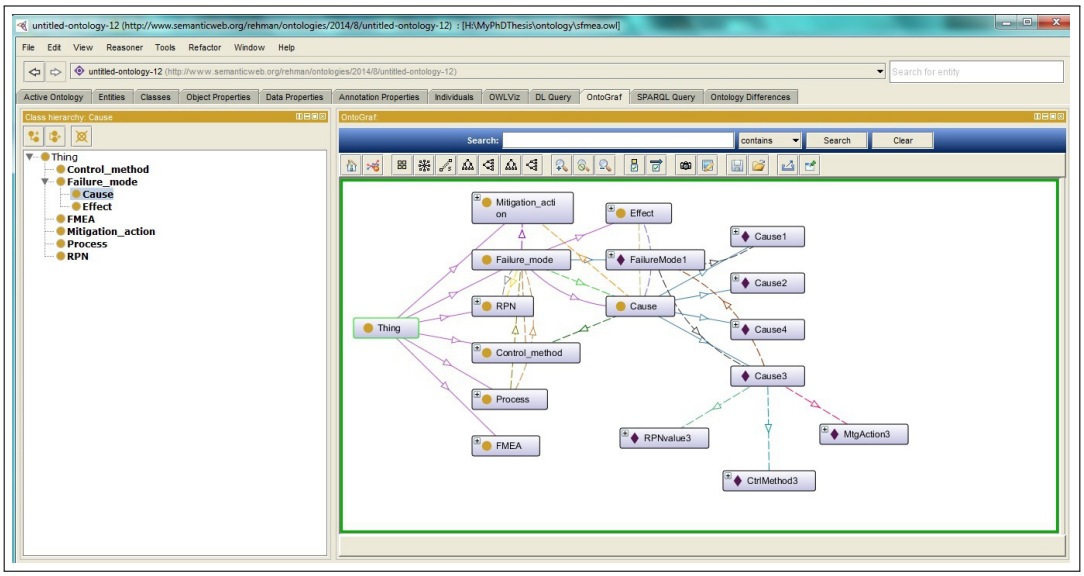


Figure 4: OntoGraph in Protégé

Rest of the classes, their attributes with data types, and relationships with one another are described following.

- FMEA class represents the header information of FMEA worksheet. Its attributes are `FMEA_No` (string), `FMEA_description` (string), `process_leader` (string), `starting_date` (dateTime), `ending_date` (dateTime), and `latest_revision_date` (dateTime). Object

property examine (inverse: isExaminedBy) connects FMEA class to Process class.

- Process class describes a process or a sub-process under analysis. It has a single attribute `process_name` (string). Object property hasSubProcess connects it to itself, whereas the property hasFailureMode (inverse: isFailureModeOf) connects it to class `Failure_mode`.

- `Failure_mode` class represents a failure mode, its cause(s) and effect(s). It has three attributes `failuremode_description` (string), `effect_description` (string) for sub-concept Effect, and `cause_description` (string) for sub-concept Cause. Different object properties are part of the concept `Failure_mode`. The hasControlMethod (inverse: isControlMethodOf) connects it to class `Control_method`, hasMitigationAction (inverse: isMitigationActionOf) connects it to concept `Mitigation_action`, it is related to class RPN through property hasRPN (inverse: isRPNOf). As a failure causes another failure and inversely a failure is the effect of another failure, therefore the concept `Failure_mode` is divided into two sub-concepts Cause and Effect. Object properties causes and isCausedBy (inverse of one another) make it distinguished if a failure is a cause or an effect. A failure that demonstrates root cause nature is only responsible to bear the relation with concepts `Control_method`, `Mitigation_action` and RPN.

- RPN class represents the magnitude impacts of a failure mode, its effect and analysis. Its attributes are severity (integer), occurrence (integer), detection (integer), and their product (integer).

- `Control_method` class describes the controls for detection and prevention of a failure. It has two attributes `control_detection` (string) and `control_prevention` (string).

- `Mitigation_action` class describes recommendations and actions taken in order to combat a failure. Its attributes are `recommended_action` (string), `responsible_person` (string), `target_completion_date` (dateTime), and `action_taken` (string). Object property hasNewRPN (inverse: isNewRPNOf) relates it to the concept RPN and on the basis of new RPN value effectiveness of a mitigation action is evaluated.

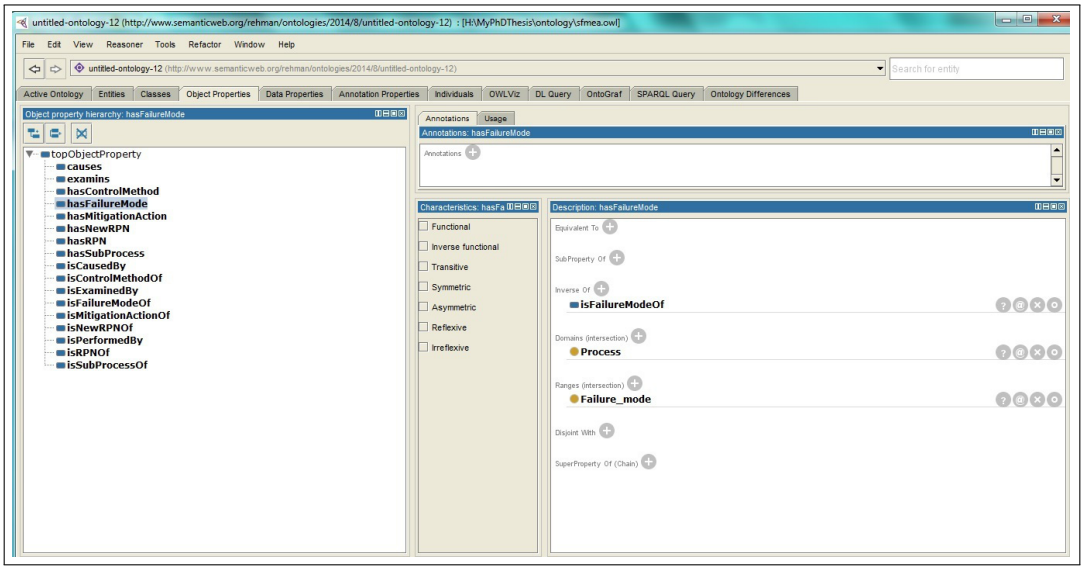


Figure 5: Object properties in Protégé

Figure 4 represents the ontology graph (OntoGraph) of proposed ontology as perceived in Protégé. In figure 5, Protégé is displaying the list of object properties. Each object property is a sub-property of topObjectProperty. Each property connects two classes, known as domain and range. For example the object property hasFailureMode connects two classes the Process and the `Failure_mode`. It is a relationship from Process to `Failure_mode`, thus the concept Process is domain of the property and concept `Failure_mode` is its range. Its inverse property isFailureModeOf would be a relation from concept `Failure_mode` to concept Process. Figure 6 shows the list of data properties, each data property is sub-property of topDataProperty. Data properties are attributes of class which are further used to create variables of instances in order to store some values. Each data property has a domain (the class name it belongs to) and a range (the data type, the type of data it allows to be stored). For example `action_taken` is an attribute of the concept `Mitigation_action` and its assertions can have only String data type.

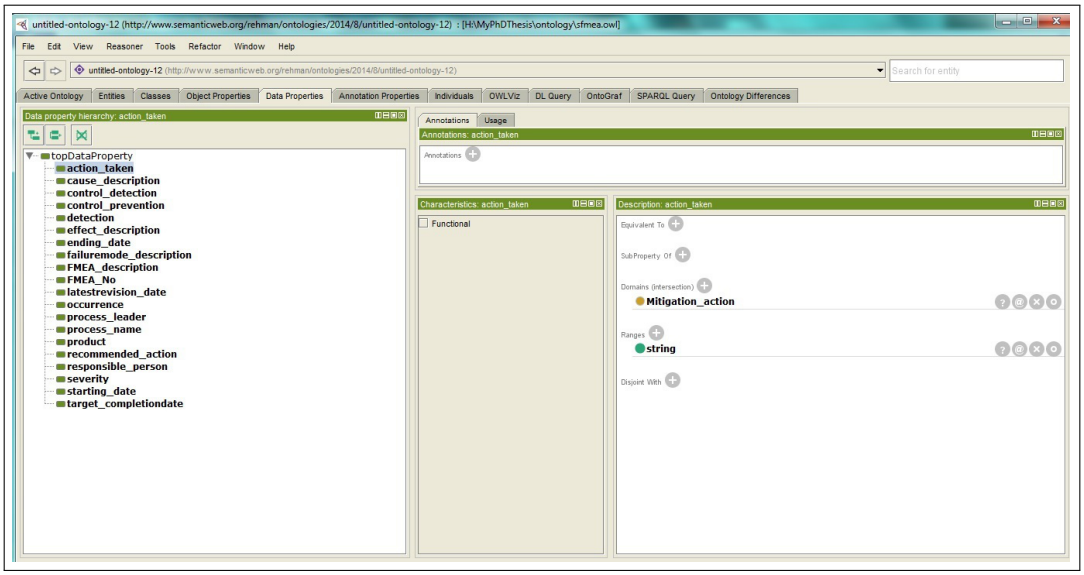


Figure 6: PFMEA data properties in Protégé

Figure 7 shows an inferred individual of concept Cause in Protégé. It clearly shows that a failure mode is caused by a cause and it further causes an effect, consequently any cause which causes a failure mode is actually responsible for its effect too. Figure 8 shows an instance of concept Effect. Figure 9 is also about an inferred individual of concept Mitigation_action. In figure 10 a complete FMEA ontology instance is shown. Whereas figure 11 shows a few instances of FMEA ontology in RDF form.

5 SPARQL queries to retrieve FMEA information from ontology

We used SPARQL with Apache Jena Fuseki server in order to access information from our ontology. Jena Fuseki server provides user friendly Graphical User Interface (GUI) to mount ontologies on server and allows retrieving query results in multiple formats. We chose CSV (Comma-Separated Values) format as original FMEA files are in the same format. Query results in CSV format can be downloaded and saved for further use and can be easily viewed in any CSV viewer. Here are some queries used to extract information from ontology.

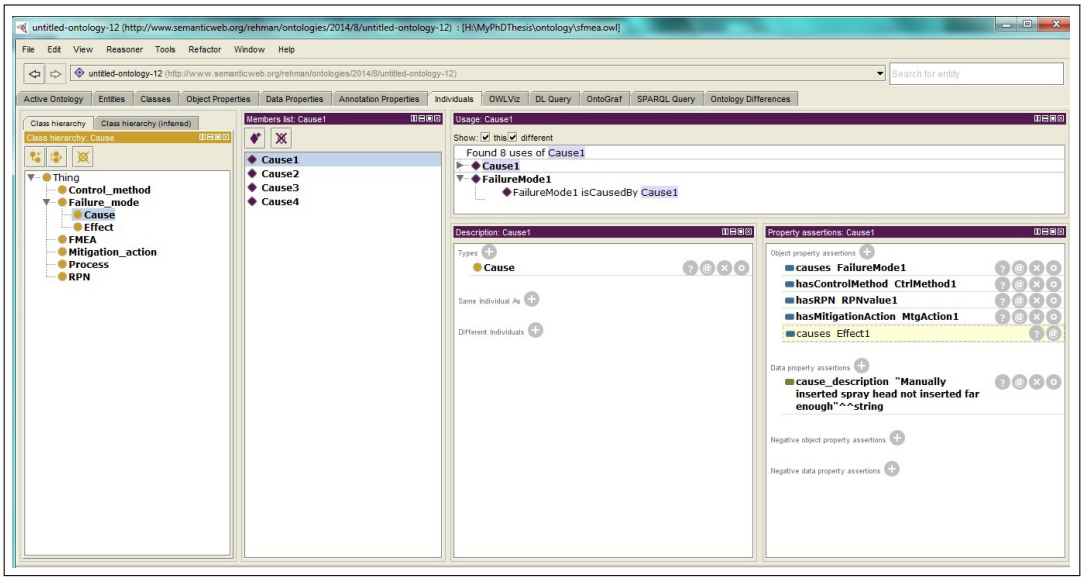


Figure 7: PFMEA instances of a sub-class cause in Protégé

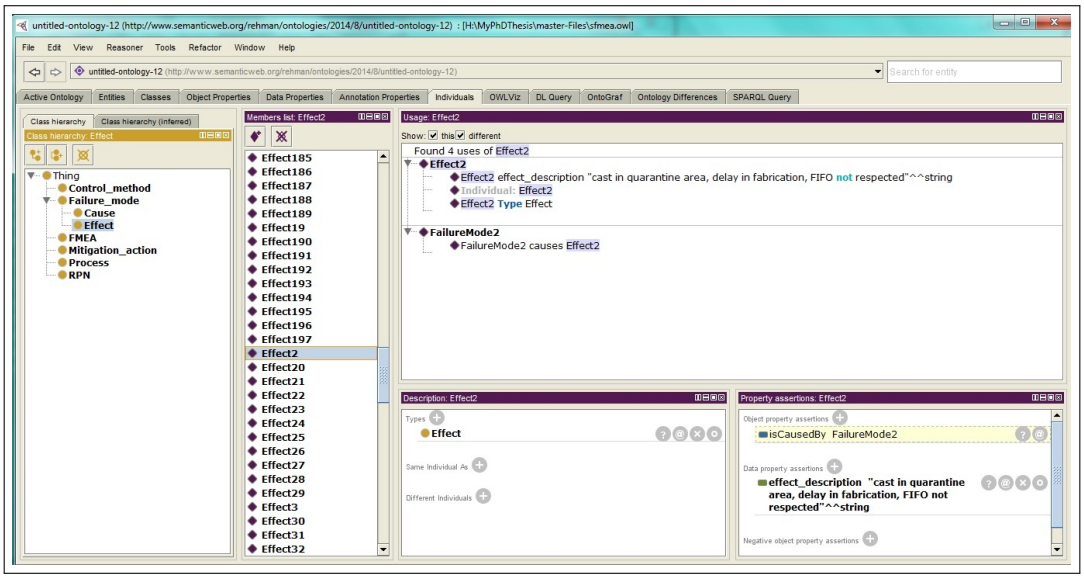


Figure 8: PFMEA instances of a sub-class effect in Protégé

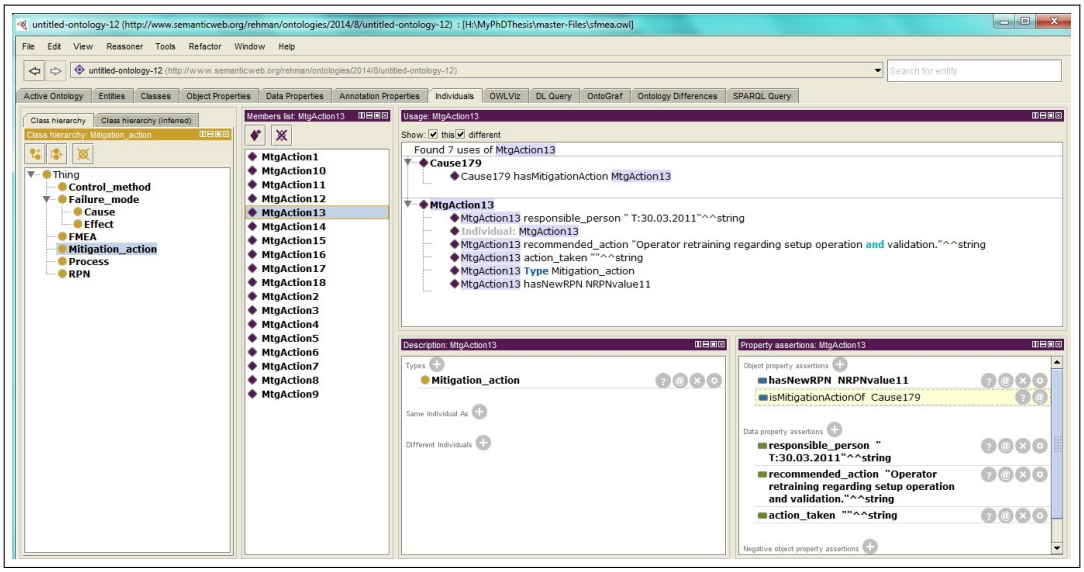


Figure 9: PFMEA instances of a sub-class mitigation_action in Protégé

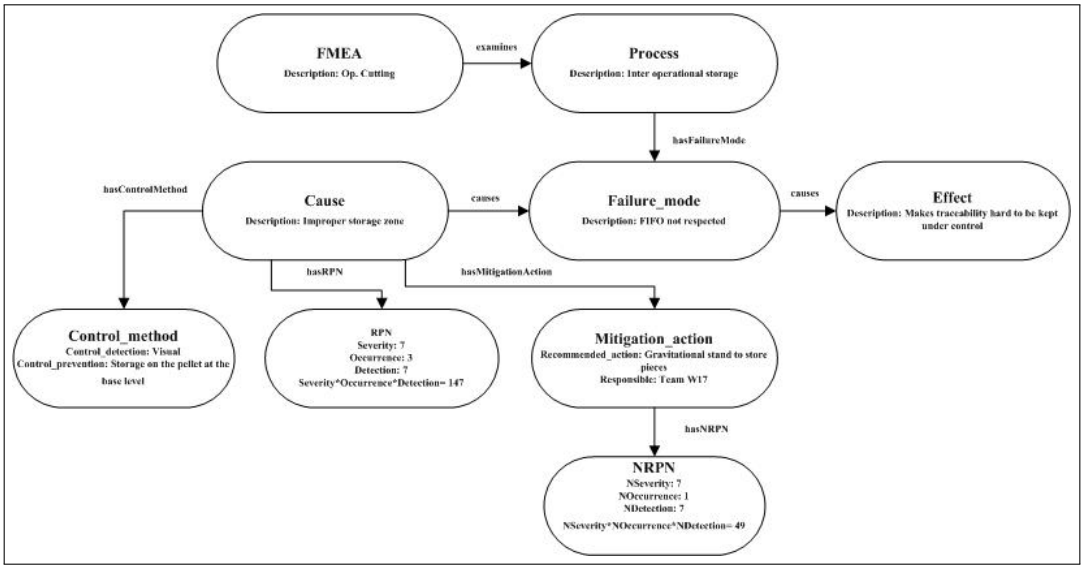


Figure 10: Example of ontology based FMEA instances

```
<!-- http://www.semanticweb.org/rehman/ontologies/2014/8/untitled-ontology-12#Cause1
-->
<owl:NamedIndividual rdf:about="http://www.semanticweb.org/rehman/ontologies/2014/
8/untitled-ontology-12#Cause1">
<rdf:type rdf:resource="http://www.semanticweb.org/rehman/ontologies/2014/8/untitled-
ontology-12#Cause"/>
<cause_description rdf:datatype="&xsd;string">Manually inserted spray head not inserted
far enough</cause_description>
<hasControlMethod rdf:resource="http://www.semanticweb.org/rehman/ontologies/2014/
8/untitled-ontology-12#CtrlMethod1"/>
<causes rdf:resource="http://www.semanticweb.org/rehman/ontologies/2014/8/untitled-
ontology-12#FailureMode1"/>
<hasMitigationAction rdf:resource="http://www.semanticweb.org/rehman/ontologies/
2014/8/untitled-ontology-12#MtgAction1"/>
<hasRPN rdf:resource="http://www.semanticweb.org/rehman/ontologies/2014/8/untitled-
ontology-12#RPNvalue1"/>
</owl:NamedIndividual>
```

Figure 11: Example of an FMEA ontology instance in RDF

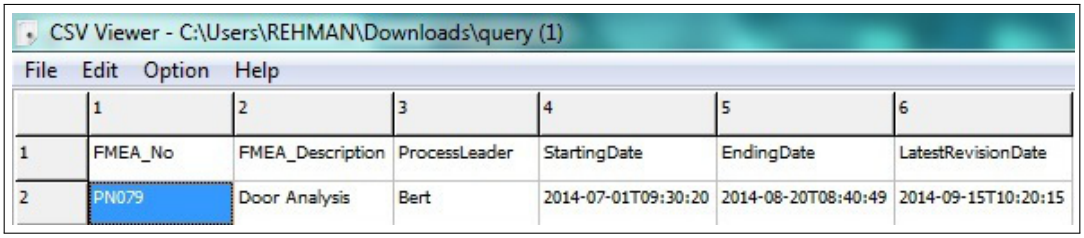
5.1 Prefixes

Following prefixes are a must for the queries to execute on Jena Fuseki sever, these are used to declare the ontology being used, the language it is developed in and its syntax and schema.
PREFIX owl: <http://www.w3.org/2002/07/owl#Ontology>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX fmea:<http://www.semanticweb.org/rehman/ontologies/2014/8/untitled-ontology-12#>

5.2 Query to display FMEA worksheet header information

SELECT (STR(?fno) as ?`FMEA_No`) (STR(?fd) as ?`FMEA_Description`) (STR(?pl) as ?ProcessLeader)
(STR(?sd) as ?StartingDate) (STR(?ed) as ?EndingDate) (STR(?ld) as ?LatestRevisionDate)
WHERE
{
?x fmea:`FMEA_No` ?fno;
fmea:`FMEA_description` ?fd;
fmea:`process_leader` ?pl;
fmea:`starting_date` ?sd;
fmea:`ending_date` ?ed.
OPTIONAL
{
?x fmea:`latestrevision_date` ?ld.
}
}

Output of this query is shown in figure 12. This query helps know very basic information about an FMEA report. For example the description of the process the report is about, its leader,and all important dates about its initiation, completion, and revision.



	1	2	3	4	5	6
1	FMEA_No	FMEA_Description	ProcessLeader	StartingDate	EndingDate	LatestRevisionDate
2	PN079	Door Analysis	Bert	2014-07-01T09:30:20	2014-08-20T08:40:49	2014-09-15T10:20:15

Figure 12: FMEA worksheet header information in CSV viewer

5.3 Query to display complete details of FMEA process

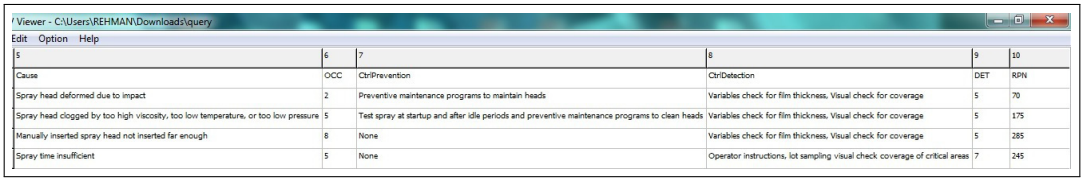
SELECT (STR(?pn) as ?Process) (STR(?fmd) as ?FailureMode) (STR(?ed) as ?Effect) (STR(?sv1) as ?SEV) (STR(?cd) as ?Cause) (STR(?oc1) as ?OCC) (STR(?cp) as ?CtrlPrevention) (STR(?cdt) as ?CtrlDetection) (STR(?d1) as ?DET) (STR(?p1) as ?RPN) (STR(?ra) as ?RecommendedAction)(STR(?at) as ?ActionTaken) (STR(?rp) as ?Responsible) (STR(?tcd) as ?TargetCompletedBy) (STR(?sv2) as ?PSEV) (STR(?oc2) as ?POCC) (STR(?d2) as ?PDET) (STR(?p2) as ?PRPN)
WHERE
{
?x fmea:`process_name` ?pn;
fmea:hasFailureMode ?fm.
?fm fmea:`failuremode_description` ?fmd;
fmea:causes ?failureEffect;
fmea:isCausedBy ?failureCause.
?failureEffect fmea:`effect_description` ?ed.
?failureCause fmea:`cause_description` ?cd;
fmea:hasControlMethod ?ctrlMethod;
fmea:hasRPN ?RPN1;
fmea:hasMitigationAction ?mgt.
?ctrlMethod fmea:`control_detection` ?cdt;
fmea:`control_prevention` ?cp.
?RPN1 fmea:severity ?sv1;
fmea:occurrence ?oc1;
fmea:detection ?d1;
fmea:product ?p1.
?mgt fmea:`recommended_action` ?ra;
fmea:`action_taken` ?at.
OPTIONAL
{
?mgt fmea:`responsible_person` ?rp;
fmea:`target_completiondate` ?tcd;
fmea:hasNewRPN ?RPN2.
?RPN2 fmea:severity ?sv2;
fmea:occurrence ?oc2;

```
fmea:detection ?d2;
fmea:product ?p2.
}
}
```

Result was spanning over large window this is why we split it into three as shown in figure 13, 14, and 15. Purpose of this query is to extract all information related to a process. For example its probable failure modes, their causes and effects,magnitude impact of the failure, recommendations and actions taken to reduce impact of failure and the magnitude impact of mitigation made.

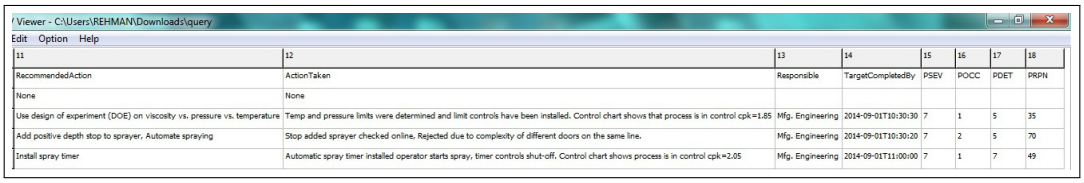
	1	2	3	4
1	Process	FailureMode	Effect	SEV
2	op.70 Manual application of wax inside door panel	Insufficient wax coverage over specified surface	Allows integrity breach of inner door panel, Corroded interior lower door panels, Deteriorated life of door leading to rust through paint and impaired function of inner door hardware.	7
3	op.70 Manual application of wax inside door panel	Insufficient wax coverage over specified surface	Allows integrity breach of inner door panel, Corroded interior lower door panels, Deteriorated life of door leading to rust through paint and impaired function of inner door hardware.	7
4	op.70 Manual application of wax inside door panel	Insufficient wax coverage over specified surface	Allows integrity breach of inner door panel, Corroded interior lower door panels, Deteriorated life of door leading to rust through paint and impaired function of inner door hardware.	7
5	op.70 Manual application of wax inside door panel	Insufficient wax coverage over specified surface	Allows integrity breach of inner door panel, Corroded interior lower door panels, Deteriorated life of door leading to rust through paint and impaired function of inner door hardware.	7

Figure 13: FMEA Process details in CSV Viewer



5	6	7	8	9	10
Cause	OCC	CtrlPrevention	CtrlDetection	DET	RPN
Spray head deformed due to impact	2	Preventive maintenance programs to maintain heads	Variables check for film thickness, Visual check for coverage	5	70
Spray head clogged by too high viscosity, too low temperature, or too low pressure	5	Test spray at startup and after idle periods and preventive maintenance programs to clean heads	Variables check for film thickness, Visual check for coverage	5	175
Manually inserted spray head not inserted far enough	8	None	Variables check for film thickness, Visual check for coverage	5	285
Spray time insufficient	5	None	Operator instructions, lot sampling visual check coverage of critical areas	7	245

Figure 14: FMEA Process details in CSV Viewer



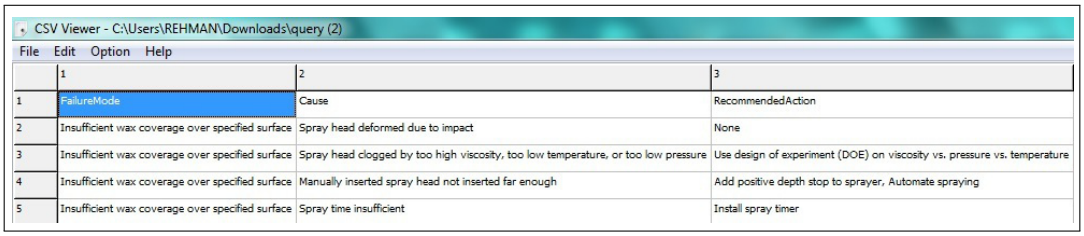
11	12	13	14	15	16	17	18
RecommendedAction	ActionTaken	Responsible	TargetCompletedBy	PSEV	POCC	PDET	PRPN
None	None						
Use design of experiment (DOE) on viscosity vs. pressure vs. temperature	Temp and pressure limits were determined and limit controls have been installed. Control chart shows that process is in control cpk=1.85	Mfg. Engineering	2014-09-01T10:30:30	7	1	5	35
Add positive depth stop to sprayer, Automate spraying	Stop added sprayer checked online, Rejected due to complexity of different doors on the same line.	Mfg. Engineering	2014-09-01T10:30:20	7	2	5	70
Install spray timer	Automatic spray timer installed operator starts spray, timer controls shut-off. Control chart shows process is in control cpk=2.05	Mfg. Engineering	2014-09-01T11:00:00	7	1	7	49

Figure 15: FMEA Process details in CSV Viewer

5.4 Query to display causes and recommendations for each failure mode

SELECT (STR(?fmd) as ?FailureMode) (STR(?cd) as ?Cause) (STR(?ra) as ?RecommendedAction)
WHERE
{
?fm fmea:`failuremode_description` ?fmd;
fmea:isCausedBy ?failureCause.
?failureCause fmea:`cause_description` ?cd;
fmea:hasMitigationAction ?mgt.
?mgt fmea:`recommended_action` ?ra;
fmea:`action_taken` ?at.
}

Output of this query is shown in figure 16. This query helps extract all causes and recommendation for each cause for a specific failure mode.


CSV Viewer - C:\Users\REHMAN\Downloads\query (2)

File Edit Option Help

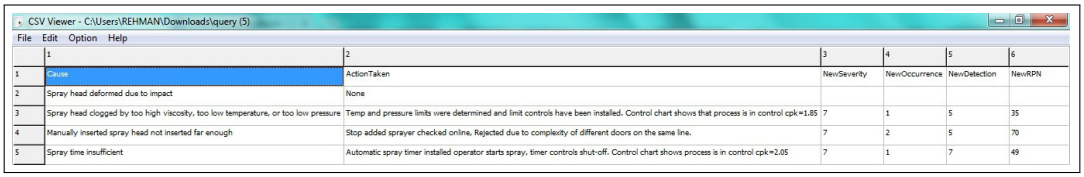
	1	2	3
1	FailureMode	Cause	RecommendedAction
2	Insufficient wax coverage over specified surface	Spray head deformed due to impact	None
3	Insufficient wax coverage over specified surface	Spray head clogged by too high viscosity, too low temperature, or too low pressure	Use design of experiment (DOE) on viscosity vs. pressure vs. temperature
4	Insufficient wax coverage over specified surface	Manually inserted spray head not inserted far enough	Add positive depth stop to sprayer, Automate spraying
5	Insufficient wax coverage over specified surface	Spray time insufficient	Install spray timer

Figure 16: Causes and recommendations for each failure mode in CSV Viewer

5.5 Query to display causes, effects and recommendations for a specific failure mode

SELECT (STR(?cd) as ?Cause) (STR(?at) as ?ActionTaken) (STR(?sv2) as ?NewSeverity) (STR(?oc2) as ?NewOccurrence) (STR(?d2) as ?NewDetection) (STR(?p2) as ?NewRPN)
WHERE
{
?x fmea:hasFailureMode ?fmd.
?fmd fmea:`failuremode_description` ?fd.
FILTER regex(?fd,"Insufficient wax coverage over specified surface")
?fmd fmea:isCausedBy ?failureCause.
?failureCause `fmea:cause_description` ?cd;
fmea:hasMitigationAction ?mgt.
?mgt fmea:`action_taken` ?at.
OPTIONAL
{
?mgt fmea:`responsible_person` ?rp;
fmea:`target_completiondate` ?tcd;
fmea:hasNewRPN ?RPN2.
?RPN2 fmea:severity ?sv2;
fmea:occurrence ?oc2;
fmea:detection ?d2;
fmea:product ?p2.
}
}

Output of this query is shown in figure 17. This query extracts causes, mitigation actions, and their magnitude impacts for a given failure mode.


CSV Viewer - C:\Users\REHMAN\Downloads\query (5)

File Edit Option Help

	1	2	3	4	5	6
1	Cause	ActionTaken	NewSeverity	NewOccurrence	NewDetection	NewRPN
2	Spray head deformed due to impact	None				
3	Spray head clogged by too high viscosity, too low temperature, or too low pressure	Temp and pressure limits were determined and limit controls have been installed. Control chart shows that process is in control cpk=1.85	7	1	5	35
4	Manually inserted spray head not inserted far enough	Stop added sprayer checked online, Rejected due to complexity of different doors on the same line.	7	2	5	70
5	Spray time insufficient	Automatic spray timer installed operator starts spray, timer controls shut-off. Control chart shows process is in control cpk=2.05	7	1	7	49

Figure 17: Causes and mitigation action(s) for a specific failure mode in CSV viewer

6 Conclusion and future work

Nowadays risk management has become a vital part of an organization's strategic management. To achieve the organizational objectives it is mandatory to increase the probability of success and decrease the probability of failure for a product or process. It is only possible when an organization knows about all expected risks and has planed policies and actions for its timely avoidance and mitigation. FMEA is one of the tools available for risk assessment. Because of its effectiveness organizations spend a lot to complete its studies. Due to some reasons as mentioned in introduction section, the valuable information produced by FMEA is not reusable. To combat this problem authors in [13] proposed a system which would be capable enough to extract information from FMEA documents, store it in a knowledge repository, and help retrieving the required information. In this article we presented an important component of that proposed system, the ontology and retrieval of information through it. As we want to develop a system which should be capable of disseminating information in a domain of experts unambiguously, for this we need a common vocabulary, understanding and structure of the specified domain knowledge, machine interpret-able descriptions of concepts and their relations, and a barrier between domain knowledge and operational knowledge; therefore we developed and FMEA ontology that qualifies all these specifications. This article not only presents the ontology but also the semantic ways to retrieve information through it. Our next step is to use this ontology for auto-population of a knowledge base from CSV format FMEA worksheets and then we will measure its effectiveness (in terms of completeness and conciseness) by deploying it, so that domain experts could interact with it for required knowledge.

Bibliography

[1] Dittmann, L. et al (2004); Performing FMEA using ontologies, *18th international workshop on qualitative reasoning*, 209-216.

[2] Fernandez, B.I.; Saberwal, R. (2010); *Knowledge Management: Systems and Processes*, M.E. Sharpe.

[3] Gruber, T. R. (1993); A translation approach to portable ontology specifications, *Knowledge Acquisition*, 5(2): 199-220.

[4] Horrocks, I. et al (2012); The HermiT OWL Reasoner, emphOWL Reasoner Evaluation Workshop, Manchester.

[5] http://jena.apache.org/documentation/`serving_data`/

[6] Larson, E. W.; Gray, C. F. (2011); *Project Management: The Managerial Process*, 5th Edition, McGraw Hill.

[7] Lee, C.F.; (2001); Using FMEA models and ontologies to build diagnostic models, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 281-293.

[8] Mansouri, D.; Hamdi-Cherif, A. (2011); Ontology-oriented case-based reasoning (CBR) approach for training adaptive delivery, *15th WSEAS Int. Conf. on Computers (CSCC'11)*, 328-333.

[9] Mikos, W.L. et al (2011); A system for distributed sharing and reuse of design and manufacturing, *Elsevier Journal of Manufacturing Systems*, 133-143.

[10] Molhanec, M. et al (2010); The Ontology based FMEA of Lead Free Soldering Process, *International Spring Seminar on Electronics Technology - ISSE*, DOI: 10.1109/ISSE.2009.5206998, 1-4.

[11] http://www.w3.org/standards/semanticweb/ontology

[12] http://www.w3.org/TR/rdf-syntax-grammar/

[13] Rehman, Z.; Kifor, S. (2014); A Conceptual Architecture Of Ontology Based KM System For Failure Mode And Effects Analysis, *International Journal of Computers Communications & Control*, 9(4): 463-470.

[14] Stamatis, D.H. (2003); Failure mode and effect analysis: FMEA from theory to execution, USA: ASQ Quality Press.

[15] Tay, K. M.; Lim, C. P. (2006); Fuzzy FMEA with a guided rules reduction system for prioritization of failures. International Journal of Quality & Reliability Management , 23(8): 1047 - 1066.

[16] Wheeler, D. J.; Chamber, D. S. (2013); *Understanding Statistical Process Control Wheeler and Chambers*, 3rd Edition, SPC Press.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(4):522-537, August 2016.

Data-driven Control of the Activated Sludge Process: IMC plus Feedforward Approach

J.D. Rojas, O. Arreta, M. Meneses, R. Vilanova

J.D. Rojas, O. Arrieta
Escuela de Ingeniería Eléctrica, Universidad de Costa Rica,
San José, 11501-2060 Costa Rica.
Tel: +506-2511-3892, fax: +506-2511-3920
{jdrojas, oarrieta}@eie.ucr.ac.cr

M. Meneses, R. Vilanova*
Departament de Telecomunicacióó i d'Enginyeria de Sistemes, Escola Tècnica Superior d'Enginyeria,
ETSE, Universitat Autònoma de Barcelona,
08193 Bellaterra, Barcelona, Spain
{montse.meneses, ramon.vilanova}@uab.cat
*Corresponding author: ramon.vilanova@uab.cat

Abstract: In the work presented in this paper, data-driven control is used to tune an Internal Model Control. Despite the fact that it may be contradictory to apply a model-free method to a model-based controller, this methodology has been successfully applied to a Activated Sludge Process (ASP) based wastewater treatment. In addition a feedforward controller over the influent substrate concentration was also computed using the virtual reference feedback tuning and applied to the same wastewater process to see the effect over the dissolved oxygen and the substrate concentration at the effluent.
Keywords: Activated sludge process, data-driven control, internal model control, wastewater treatment plants

1 Introduction

Data-Driven Control is a rather new control approach that does not attempt to find the model of the plant to control, instead, it uses experimental data to directly find a controller, which, generally, is meant to minimize some control performance criterion. Some of the most remarkable methods within this control approach are the Iterative Feedback Tuning (IFT) [1, 2], the Windsurfer Approach [3, 4], the Correlation Approach [5, 6] and the Virtual Reference Feedback Tuning (VRFT) [7–9]. While the IFT and the Windsurfer Approach are iterative methods (that is, several experiments on the plant have to be performed in order to find the controller) the Correlation approach and the VRFT are one shot methods (only one set of data is needed to find the controller).

IFT computes an unbiased gradient of a performance index to iteratively improve the tuning of the parameters of a reduced order discrete time controller, at each iteration three different experiments are performed on the system and based on this data, the gradient is computed; in the Windsurfer approach the objective is to find a better model for the plant (and subsequently a better controller) using closed-loop data and Internal Model Control (IMC) design [10] in such a way that, with every iteration, the closed loop bandwidth can be increased; Data-Driven control using the correlation approach is a one-shot methodology that attempts to find the values of a restricted order controller that tries to minimize the correlation between the closed-loop error of the system (based in a desired closed-loop behavior) and the reference for the process output, and the VRFT translates the model reference control problem into an identification problem,

being the controller the transfer function to identify based on some "virtual signals" computed from a batch of data taken directly from an open-loop experiment.

In this work, the VRFT approach is used and extended in order to be applied to a Wastewater Treatment Plant (WWTP). WWTP are an important case of study within the process control area, while an active research area that involves other disciplines as for example chemistry, biology, and instrumentation. Moreover, by it self, WWTPs have deep impact in the quality of live in big cities. That is why the constraint on the level of pollution of the treated water before discharging it into the receiving waters, is becoming more stringent [11] and because of that, a correct control and operation of WWTP is one of the top priorities for both, industry and academics.

Among the types of WWTP, the Activated Sludge Process (ASP) is one of the most popular and more studied [12, 13]. This is also true from the automatic control perspective: for example in [14] a parameter and state non-linear estimator is used in an adaptive linearizing control of the dissolved oxygen and substrate concentration of an ASP but under the assumption that only the dissolved oxygen is available for measurement. In [15], several multivariable PI control method are applied to the ASP by linearizing the nonlinear model and the results are presented, as well as the combination of some of these methods. In [16], predictive control is used to maintain a low concentration of substrate at the output by controlling the dissolved oxygen using the dilution rate. The internal model of the predictive control is a three layer neural network. In [17] the control of the substrate concentration is achieve using an estimation based on the dissolved oxygen measurements, a dynamic controller that cope with the change in reference and a PID controller that corrects the steady state error produced by the use of a linearized model in the first controller. In [18] a decentralized PI approach is presented to show that simple well tuned PI controllers can achieve a similar performance than more complex methodologies for the ASP case. Some other strategies have been proposed recently such as in [19] where an I-P controller control system with pole-placement design is proposed.

In all the cases, some sort of model (non-linear o linearized) is used to computed the controller. In several cases it is supposed that some parameters are known which may no be the case for a real plant. The contribution of this paper is to apply a data driven approach to the tuning of discrete-time restricted-order linear controller in a decentralized approach to have good performance for both reference tracking and disturbance rejection in an ASP based WWTP. Without explicitly computing a model of the process an Internal Model Control approach is used in conjunction with the VRFT methodology. Even more, the effect of the influent concentration disturbance is taken into account by computing a feedforward control using the VRFT as well. It was found that this methodology provide excellent results when compared with a PI approach.

The rest of the paper is divided in two parts, in section 2, a short overview on VRFT is presented as well as the mentioned extensions for the IMC control. In section 3 the results of the application of this data-driven method is presented and compared with a two-degrees of freedom PI controller. The conclusion are presented in section 3.

2 Virtual reference feedback tuning extensions

In this section, an overview on the VRFT is presented as well as some results that extend the capacity of the VRFT for different control strategies and structure of controllers is presented.

2.1 Virtual Reference Feedback Tunning overview

The Virtual Reference Feedback Tuning (VRFT) is a one-shot data-based method for the design of feedback controllers. The original idea was presented in [7], and then formalized by

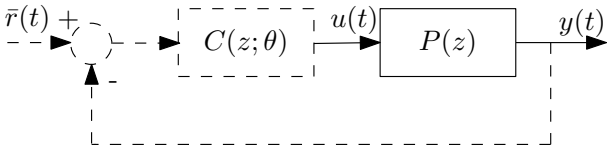


Figure 1: The VRFT set up. The dashed lines represent the "virtual" part of the method

Lecchini, Campi, Savaresi and Guardabassi (see [8, 9]).

In [8], the method is presented for the tuning of a feedback controller. If a the controller belongs to the controller class $\{C(z;\theta)\}$ given by $C(z;\theta) = \beta^T(z)\theta$, where $\beta(z) = [\beta_1(z) \ \cdots \beta_n(z)]^T$ is a known vector of transfer functions, and $\theta = [\theta_1 \ \theta_2 \ \cdots \theta_n]^T$ is the vector of parameters, then the control objective is to minimize the model-reference criterion given by:

$$J_{MR}(\theta) = \left\| \left(\frac{P(z)C(z;\theta)}{1 + P(z)C(z;\theta)} - M(z) \right) W(z) \right\|_2^2 \tag{1}$$

Starting from a batch of open-loop data $\{u(t), y(t)\}$, a "virtual" signal is computed in such a way that, if the closed-loop system is feed with this virtual signal and the controllers in the loop were the ideal controllers that would achieve a predefined target transfer function, then the input and output signals of the plant in closed-loop would be the same than the batch of open-loop data. The output of the controller should be equal to $u(t)$ and then, this controller can be found by *identifying* the transfer function which yields the output $u(t)$ when the input $\bar{r}(t) - y(t)$ is applied to the input as depicted in Fig. 1

The original VRFT algorithm, as presented by the authors in [8], is as follows: Given a set of measured I/O data $\{u(t), y(t)\}_{t=1,\ldots,N}$

1. Calculate:
   - a virtual reference $\bar{r}(t)$ such that $y(t) = M(z)\bar{r}(t)$, and
   - the corresponding tracking error $e(t) = \bar{r} - y(t)$
2. Filter the signals $e(t)$ and $u(t)$ with a suitable filter $L(z)$:
$$e_L(t) = L(z)e(t)$$
$$u_L(t) = L(z)u(t)$$
3. Select the controller parameter vector, say, $\hat{\theta}_N$, that minimizes the following criterion:
$$J_{VR}^N(\theta) = \frac{1}{N}\sum_{t=1}^{N}(u_L(t) - C(z;\theta)e_L(t))^2 \tag{2}$$
   If $C(z;\theta) = \beta^T(z)\theta$, the criterion (2) can be given by
$$J_{VR}^N(\theta) = \frac{1}{N}\sum_{t=1}^{N}\left(u_L(t) - \varphi_L^T(t)\theta\right)^2 \tag{3}$$
   with $\varphi_L(t) = \beta(z)e_L(t)$ and the parameter vector $\hat{\theta}_N$ is given by
$$\hat{\theta}_N = \left[\sum_{t=1}^{N}\varphi_L(t)\varphi_L(t)^T\right]^{-1}\sum_{t=1}^{N}\varphi_L(t)u_L(t) \tag{4}$$

The authors, also showed that, the filter $L(z)$ should be the one that approximates the criterion (2) to (1). This filter should be designed to accomplish the constraint:

$$|L|^2 = |1 - M|^2 |M| |W|^2 \frac{1}{\Phi_u} \tag{5}$$

where Φ_u is the spectral density of $u(t)$.

The VRFT framework have been used in several applications and even have been extended for the MIMO case and used for PID tuning, for example see [20–24].

2.2 Internal model control using the virtual reference feedback framework

Internal Model Control (IMC) is a popular control method that incorporates the model of the process into the controller [10]. The standard structure is depicted in Fig. 2. $P(z)$ represents the Plant, while $\bar{P}(z)$ is its model. $Q(z)$ is the IMC controller. If the output of the model and the output of the plant are the same, and there is no disturbance, the control system behaves as if it was in open-loop. If this is the case, to have perfect tracking, $Q(z)$ must try to cancel the dynamics of the plant. On the other hand, if there is a mismatch between the plant and its model or if a disturbance acts on the system, the feedback loop enters into play. This characteristics leads to the well know property that an IMC system would be nominally internally stable if $Q(z)$ is stable, in case the model is equal to the plant.

Of course, finding a perfect model is rarely achievable and if it were, $Q(z)$ may not be possible to be equal to the inverse of this model due to physical limitations or because the inverse of the plant may lead to an unstable controller. In [10] a two-step design is proposed for this kind of controller:

1. Solve the nominal performance criterion given, for example, by

$$\min_{\bar{Q}(z)} \left\| \left(1 - \bar{P}(z)\bar{Q}(z)\right) W(z) \right\|_p \tag{6}$$

   Where W is a filter chosen to give more importance in certain frequencies and $\|\cdot\|_p$ is a given norm that defines the performance criterion. The optimal solution to this problem yields to a sensitivity function given by $S^*(z) = 1 - \bar{P}(z)\bar{Q}(z)$ and the complementary sensitivity function given by $M^*(z) = \bar{P}(z)\bar{Q}(z)$, that is, the response to a change in the reference is as if it were in open loop, while the response to a disturbance is in closed-loop. Of course this response is not achievable and therefore, the model $\bar{P}(z)$ should be divided in an invertible and non-invertible part to be able to approximate the optimal controller.

2. To introduce robustness conditions, the complementary sensitivity has to be rolled off at high frequencies, therefore, it is necessary to add a low pass filter $f(z)$ to the controller $\bar{Q}(z)$, to obtain the final controller $Q(z) = \bar{Q}(z)f(z)$. Suppose that the multiplicative uncertainty is bounded by a frequency dependent function $\bar{l}_m(\omega)$,

$$\left| \frac{P(e^{\jmath\omega}) - \bar{P}(e^{\jmath\omega})}{\bar{P}(e^{\jmath\omega})} \right| \leq \bar{l}_m(\omega)$$

   Then the closed-loop system is robustly stable if and only if

$$|f(e^{\jmath\omega})| < \frac{1}{\left|\bar{P}(e^{\jmath\omega})\bar{Q}(e^{\jmath\omega})\bar{l}_m(\omega)\right|} \quad \forall\omega \tag{7}$$

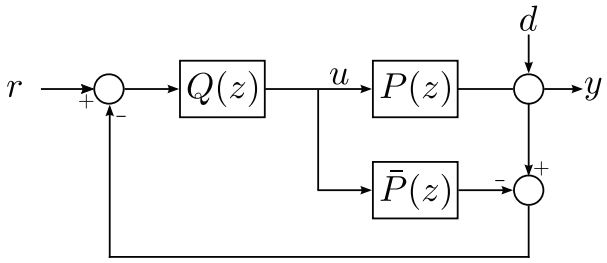


Figure 2: Standard Structure of the IMC. $\bar{P}$ represents the plant model and Q is the IMC controller

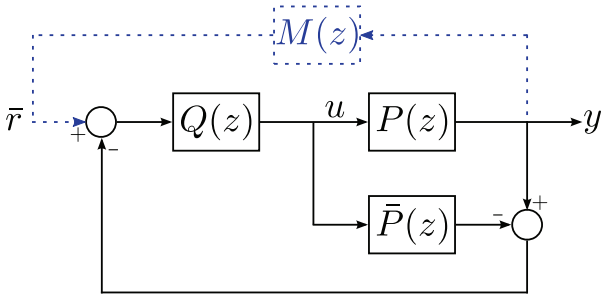


Figure 3: Disposition for the VRFT experiment using the IMC topology. The dashed line represents the virtual signals and components.

The reasons why the IMC control has become very popular is because, finding the controller and the conditions for robust stability can be cast in a very simple form. Using the VRFT framework, this constraint are not really necessary, since the methodology does not need any modeling step. Interested reader on IMC control, can find more information on [10, 25].

It is possible to find an IMC controller using the VRFT framework without concerning about the modeling of the system. In Fig. 3, the experimental setup for the VRFT applied to the IMC topology is depicted. If the target complementary sensitivity function is given by $M(z)$, then the virtual reference $\bar{r}(t)$ is computed as:

$$\bar{r}(t) = M^{-1}(z)y(t) \tag{8}$$

If the ideal controller were in the loop, then one would have $\bar{P}(z) = P(z)$ and the input to the controller $Q(z, \theta)$ would be $\bar{r}(t)$ and its corresponding output would be $u(t)$ in order to have $y(t)$ as the output of the closed-loop system. From Fig 3, it can be found that the ideal controller would be given by

$$\begin{aligned} Q_0(z) &= M(z)P(z)^{-1} \\ \bar{P}_0(z) &= M(z)Q_0(z)^{-1} \end{aligned} \tag{9}$$

$P_0(z)$ would be the ideal plant model that is derived from the ideal controller. This basic idea leads to the following optimization problem which gives the set of optimal parameters θ^* (in a least square sense):

$$\min_{\theta} J(\theta) = \min_{\theta} \sum_{i=1}^{N} \left(u(i) - Q(z, \theta)\bar{r}(i)\right)^2 \tag{10}$$

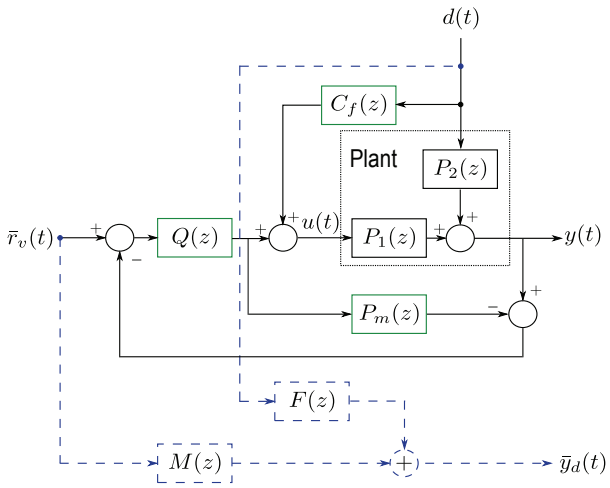


Figure 4: Virtual Reference setup for feedforward plus IMC controller. Solid lines are for "real" components and signals while dashed lines are for "virtual" components and signals

Once $Q(z, \theta^*)$ has been determined, it is easy to compute the approximation of the process model of the plant from (9):

$$\bar{P}(z, \theta) = M(z)Q(z, \theta)^{-1} \tag{11}$$

It is important to note that $\bar{P}(z, \theta)$ is seen just as an "instrumental model", that results from the determination of the optimal controller. In fact, it can be seen just as a part of the IMC controller that results from the optimization. Of course, if a robust check is performed with the obtained controller, this approximation of the plant can be used as if it were the nominal model. In that case, the controller and the nominal model would be found at once. The filter for robust operation presented in (7), is already included in the determination of $Q(z, \theta^*)$ given the desired $M(z)$.

2.3 VRFT approach to feedforward control

Sometimes, it is possible to measure disturbances that affect the process output. In those cases, it is desirable to use a feed-forward controller that acts before the effects of these disturbances reach the output of the plant. In [26], the idea of using the VRFT controller was presented to be used in conjunction with a one-degree of freedom controller. The main difference is that it is assumed that the disturbance is available for measurement and is used in the optimization problem.

In this paper this idea was implemented in conjunction with the VRFT-IMC controller. Suppose that the control system can be represented by the diagram in Fig. 4, where $P_1(z)$ and $P_2(z)$ represent the unknown dynamics of the plant from the input $u(t)$ and the disturbance $d(t)$ to the output $y(t)$, respectively. These three signals are measured from an open-loop experiment. The idea of using the feedforward control plus the IMC controller is to cope with both measurable and non-measurable disturbances. $Q(z)$, $P_m(z)$ and $C_f(z)$ are the controllers to be found. The "virtual" components and signals (which are presented with dashed lines in Fig. 4) are:

- $M(z)$, which is the target closed-loop dynamics from the reference signal to the output of the controlled system.
- $F(z)$, is the target closed-loop dynamics from the measured disturbance to the output.

- $\bar{r}_v$, is the virtual reference computed from the data obtained from an open loop experiment and the closed-loop target functions.
- $\bar{y}_d$, is the ideal disturbed output in closed-loop, if the virtual reference is applied in the closed-loop and the ideal controllers are set in place.
- d, is the measurable disturbance signal that it is suppose to be available in the open loop experiment.

The virtual reference signal $\bar{r}_v$ has to be computed according to the ideal relationships and the measured and virtual signals:

$$\bar{y}_d = M\bar{r}_v + Fd \tag{12}$$

If one is able to find the ideal controllers, then $\bar{y}_d = y$. Since this is exactly what is needed, the virtual signal is computed from (12) as:

$$\begin{aligned} \bar{y}_d &= y \\ \bar{r}_v &= M^{-1}\left(y_d - Fd\right) \end{aligned} \tag{13}$$

The transfer function of the controlled system is

$$y = \frac{P_1 Q}{1 + (P_1 - P_m)Q} r + \frac{(1 - P_m Q)(P_1 C_f + P_2)}{1 + (P_1 - P_m)Q} d \tag{14}$$

Note in (14), that the input signals do not have a bar, denoting that these signals are not virtual, but actually are entering to the system. When comparing (12) and (14), one is able to find the ideal controllers that would, theoretically, drive the system to the desired dynamics (if the transfer functions of the plant were known):

$$\begin{aligned} Q_o &= \frac{M}{P_1 - M(P_1 - P_m)} \\ C_{fo} &= \frac{1}{P_1}\left(\frac{F(1 - (P_1 - P_m)Q)}{1 - P_m Q} - P_2\right) \end{aligned} \tag{15}$$

Once $Q(z)$ and $C_f(z)$ has been obtained by optimization, the best approximation of $P_m(z)$ is $P_m(z) = M(z)Q^{-1}(z)$, just as in (9) where one expects that $P_m(z) \approx P_1(z)$ since the virtual reference was computed to achieve this relationship. This optimization is found by following the paths in the diagram of Fig. 4 that lead from the measured and virtual inputs to the $u(t)$ signal, it is straightforward to find that the cost function to optimize is given by:

$$J(\theta) = \frac{1}{N}\sum_{i=0}^{N-1}\left[u(i) - \left(Q\left(z,\theta\right)\bar{r}_v(i) + C_f\left(z,\theta\right)d(i)\right)\right]^2 \tag{16}$$

Solving this optimization problem, one is able to find directly the two controllers and the instrumental model, using only one batch of input-output data without any iterative scheme.

3 Application to an ASP based WWTP

In this section a practical example of the IMC-VRFT method exposed above is presented. The plant considered in this paper is the WWTP given in [27]. It comprises an aerated tank

Table 1: Initial conditions

Biomass	$X(0)$	= 217.79 mg/l
Substrate	$S(0)$	= 41.23 mg/l
Dissolved Oxygen	$DO(0)$	= 6.11 mg/l
Recycled Biomass	$X_r(0)$	= 435.58 mg/l
Influent Substrate	$S_{in}(0)$	= 200.00 mg/l
Influent Dissolved Oxygen	$DO_{in}(0)$	= 0.50 mg/l

Table 2: Kinetic parameters

$\beta = 0.2$	$K_c = 2$mg/l
$r = 0.6$	$K_s = 100$mg/l
$\alpha = 0.018$	$K_{DO} = 0.5$
$Y = 0.65$	$DO_s = 0.5$mg/l
$\mu_{max} = 0.15$ h^{-1}	

where microorganisms act on organic matter by biodegradation, and a settler where the solids are separated from the wastewater and a proportional part is then recycled to the aerator in order to maintain certain amount of biomass in the system. The layout is shown in Fig. 5. The component balance for the substrate, biomass, recycled biomass and dissolved oxygen provide the following set of non-linear differential equations:

$$\frac{dX(t)}{dt} = \mu(t)X(t) - D(t)(1+r)X(t) - rD(t)X_r(t) \tag{17}$$

$$\frac{dS(t)}{dt} = -\frac{\mu(t)}{Y}X(t) - D(t)(1+r)S(t) + D(t)S_{in} \tag{18}$$

$$\frac{dDO(t)}{dt} = -\frac{K_o\mu(t)}{Y}X(t) - D(t)(1+r)DO(t) + K_La(DO_s - DO(t)) + DO(t)DO_{in} \tag{19}$$

$$\frac{dX_r(t)}{dt} = D(t)(1+r)X(t) - D(t)(\beta+r)X_r(t) \tag{20}$$

$$\mu(t) = \mu_{max}\frac{S(t)}{k_S + S(t)}\frac{DO(t)}{k_{DO} + DO(t)} \tag{21}$$

$$K_La = \alpha W(t) \tag{22}$$

where $X(t)$ - biomass, $S(t)$ - substrate, $DO(t)$ - dissolved oxygen, DO_s - maximum dissolved oxygen, $X_r(t)$ - recycled biomass, $D(t)$ - dilution rate, $W(t)$ - aeration rate, S_{in} and DO_{in} - substrate and dissolved oxygen concentrations in the influent, Y - biomass yield factor, μ - biomass growth rate in a Monod like form [28], μ_{max} - maximum specific growth rate, k_S and k_{DO} - saturation constants, K_La - oxygen mass transfer coefficient, α - oxygen transfer rate, K_o - model constant, r and β - ratio of recycled and waste flow to the influent. The influent concentrations are set to $S_{in} = 200$ mg/l and $DO_{in} = 0.5$ mg/l.

The control strategy is a decentralized control as in [18] where the multivariable process is treated as two separate single variable process. The strategy is depicted in Fig.6. With respect to the control problem definition, it is considered that the dissolved oxygen, $DO(t)$, and substrate, $S(t)$, are the controlled outputs of the plant, whereas the dilution rate, $D(t)$, and aeration rate $W(t)$ are the two manipulated variables. The control of DO provides a method to maintain the necessary amount of biomass in the system while controlling S gives a way to keep the pollution

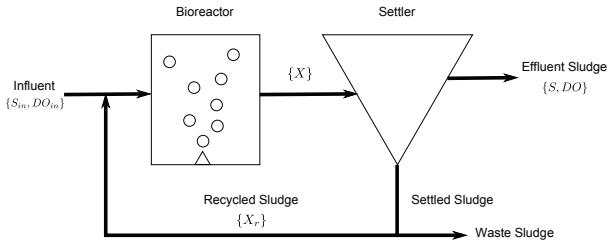


Figure 5: Wastewater Treatment Process

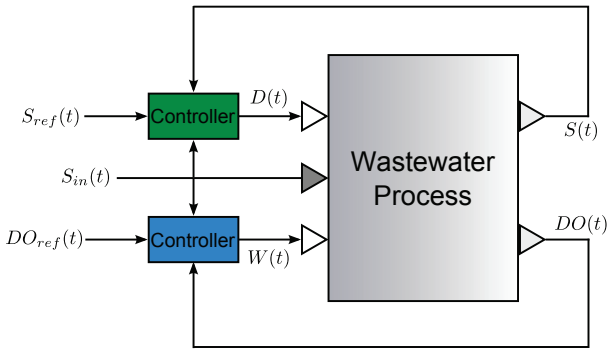


Figure 6: Control Strategy for the WWTP

at the effluent in an acceptable level [14]. The initial conditions and kinetic parameters are taken as in [18, 27] and presented in Table1 and 2.

The settings of the VRFT controller are as follows: For both control loops, the sampling time was selected as $Ts = 0.5$min, the IMC controller $Q(z)$ has the following parameterization:

$$Q(z) = \frac{\alpha_1 + \alpha_2 z^{-1} + \alpha_3 z^{-2}}{\beta_1 + \beta_2 z^{-1} + \beta_3 z^{-2}} \tag{23}$$

the target transfer function for the DO loop is:

$$M_{DO}(z) = \frac{0.02357 z^{-1}}{1 - 0.9764 z^{-1}} \tag{24}$$

which represents a first order transfer function with a constant time of approximately 20min. For the S loop (controlled by manipulating $D(t)$), the target closed-loop dynamics is a first order transfer function with a constant time of approximately 40min given by:

$$M_S(z) = \frac{0.01382 z^{-1}}{1 - 0.9862 z^{-1}} \tag{25}$$

The input-output data was selected as an additive random signal of 0 mean and variance 90 for the $W(t)$ and variance 7.5e-4 for the $D(t)$ around the operation points given in Table 1. The resulting controllers were found as:

$$\begin{aligned} Q_{DO}(z) &= \frac{40.69 - 19.35 z^{-1} - 19.65 z^{-2}}{1 - 0.4683 z^{-1} - 0.4792 z^{-2}} \\ Q_S(z) &= \frac{0.01236 - 0.006155 z^{-1} - 0.006158 z^{-2}}{1 - 0.4863 z^{-1} - 0.4924 z^{-2}} \end{aligned} \tag{26}$$

The IMCFF-VRFT version was also implemented, considering the influent substrate concentration S_{in} as the measurable disturbance. The Q controller has the same parameterization as in (23). For the feedforward controller, the parameterization is:

$$C_f(z) = \frac{\gamma_0 + \gamma_1 z^{-1}}{1 - \sigma_1 z^{-1}} \tag{27}$$

The sampling time is the same as for the Q controllers, and the desired target transfer function is $F(z) = 0$, which is normally what is desired with the feedforward control. The experimental data was slightly changed, because the dynamics from the disturbance to the output are slower: the data changes slowly and it was taken into account that a portion of the output data were affected only by the input, while another portion is affected only by the disturbance and, finally, another portion is affected by both. This is helpful to identify correctly both controllers at the same time. The resulting controllers are:

$$\begin{aligned} Q_{DO}(z) &= \frac{40.57 - 80.37z^{-1} + 39.8z^{-2}}{1 - 1.974z^{-1} + 0.9739z^{-2}} \\ C_{fDO}(z) &= \frac{-0.0002002 + 0.001138z^{-1}}{1 - 0.9976z^{-1}} \\ Q_S(z) &= \frac{0.01233 - 0.02416z^{-1} + 0.01183z^{-2}}{1 - 1.947z^{-1} + 0.9479z^{-2}} \\ C_{fS}(z) &= \frac{-0.0006154 + 0.0004481z^{-1}}{1 - 0.7261z^{-1}} \end{aligned} \tag{28}$$

The results of this controllers are compared to the two-degrees of freedom, continuous time PI controller of [18]. Two different test were performed: a change in the references and a disturbance on the influent substrate S_{in} where it is considered that every 24h, an increase of 10% of the value of S_{in} during 1h takes place.

For the change in reference ($S_{ref}(t)$ for the substrate concentration reference and $DO_{ref}(t)$ for the dissolved oxygen reference), the result is as given in Fig. 7 and 8. A step change of 10mg/l is applied to $S_{ref}(t)$ at time $t = 10$h while a step change of -2mg/l in $DO_{ref}(t)$ is applied at time $t = 100$h. The effect of one loop change in the other loop, due to the process interaction, can be observed as well. In both cases the IMC-VRFT and the IMCFF-VRFT controller achieves a better performance for both the reference tracking as well as the disturbance rejection when the other loop changes. In Table 3 the values of the integral of the squared errors (ISE) and the Total Variation (TV), which measures the aggressiveness of the control effort, are presented. ISE and TV are computed as:

$$\begin{aligned} \text{ISE} &= \int_0^t e(t)^2 \, dt \\ \text{TV} &= \sum_{i=1}^{N} |u(i) - u(i-1)| \end{aligned} \tag{29}$$

$e(t)$ is the error signal (the reference minus the measured output), and $u(i)$ is the output of the controlled sampled every hour and N is the total number of samples. In the column "Reference Tracking" it can be seen that for the S loop with the application of the IMC-VRFT controller the ISE is greatly reduced (near the 57%) but with almost the same TV. The DO loop is improved in both ISE and TV, as can be also seen in Fig. 7 and 8, the section that is zoomed details the change in the DO, it is clear that the response of the PI controller is much worse than the

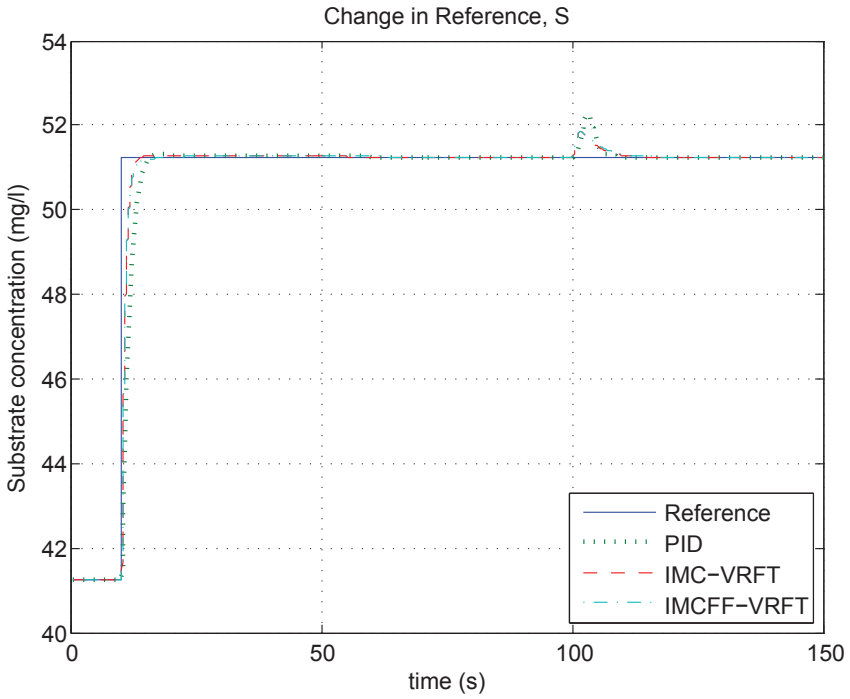


Figure 7: Step change in the S reference at time $t = 10$h

Table 3: Comparison of the results between the IMC-VRFT, IMCFF-VRFT and the PID control

		Reference tracking		Disturbance rejection	
		S	DO	S	DO
ISE	PI	77.23	2.94	12.05	0.0021
	IMC-VRFT	32.85	0.64	5.56	6e-005
	IMCFF-VRFT	33.25	0.65	0.15	0.0083
TV	PI	0.091	89.10	0.15	7.02
	IMC-VRFT	0.083	67.33	0.15	4.69
	IMCFF-VRFT	0.082	65.01	0.16	13.07

response of the IMC-VRFT, which almost has no overshoot. In Fig. 9, the plot of the control signals is presented for both the dilution rate and the air flow rate. As it was expected, the performance of the IMC-VRFT and the IMCFF-VRFT are similar for the reference tracking since no disturbance is present for this simulation, despite the fact that the controllers were found using different optimization methods (in the case of the IMC-VRFT a simple linear least square problem can be casted, while for the IMCFF-VRFT, the output error (OE) method was applied [29]).

For the disturbance in the substrate concentration of the influent, the responses are presented in Fig. 10, Fig. 11 and Fig. 12. PI control is faster to control the disturbance Fig. 10, but the overshoot is larger. The IMCFF-VRFT controller performs much better than the IMC-VRFT controller (the ISE is 97% lower) with almost the same TV. The response of the DO is greatly improved with a reduction of almost 97% of ISE for IMC-VRFT, but the IMCFF-VRFT performs much worse than the PI controller (the ISE is almost 4 times bigger with a TV 2 times grater). It is clear that a feedforward strategy is not adecuate for the DO loop, since the effect of the change in S_{in} is much slower than the effect of the aireation.

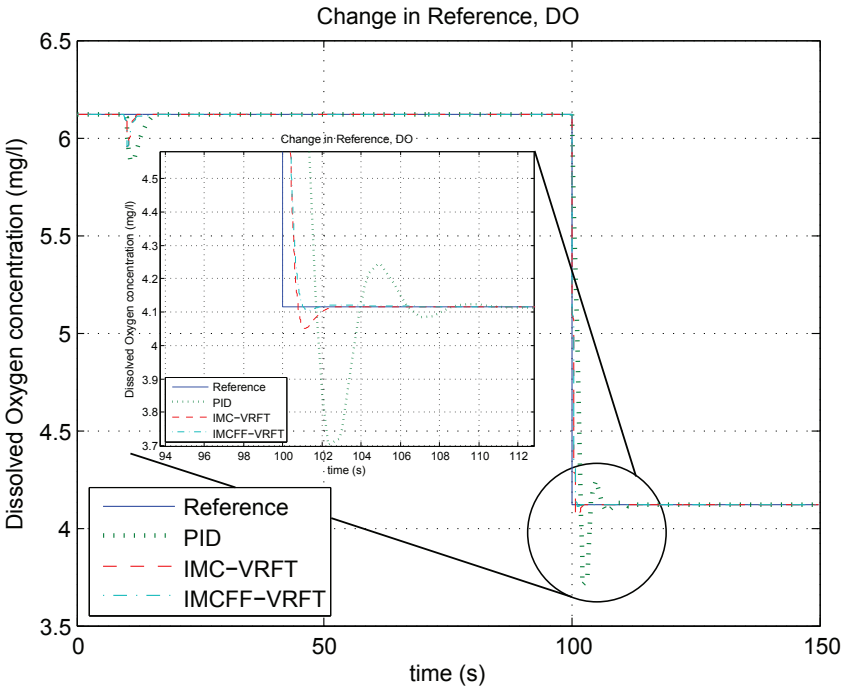


Figure 8: Step change in the DO reference at time $t = 100$h

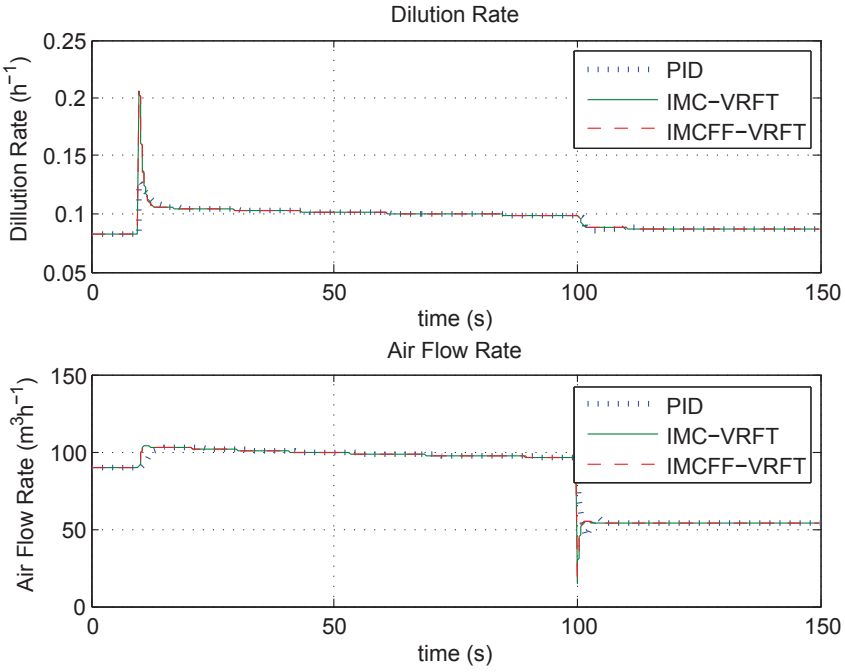


Figure 9: Control effort during the change in the reference

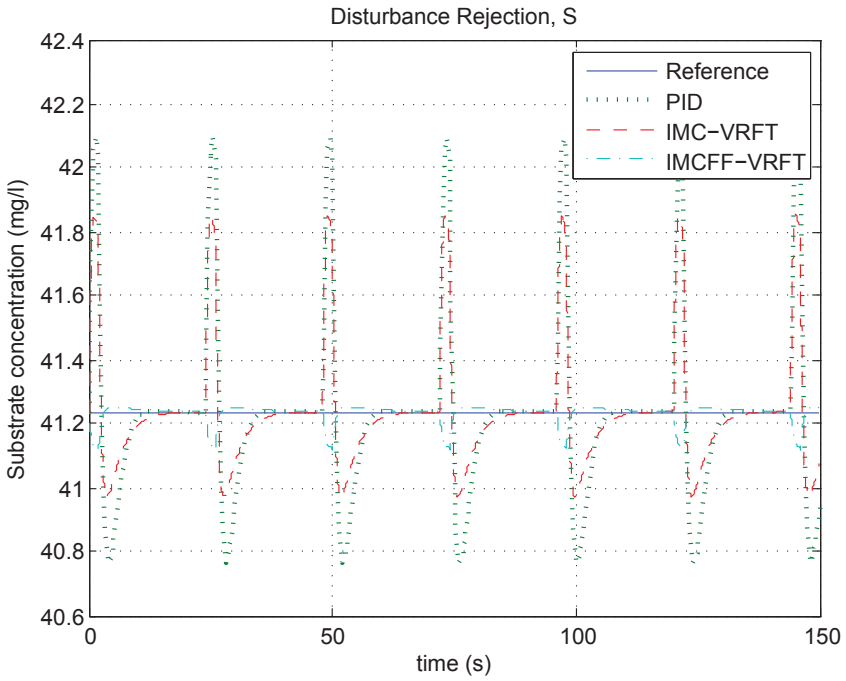


Figure 10: Effect over the substrate concentration when the substrate input is disturbed

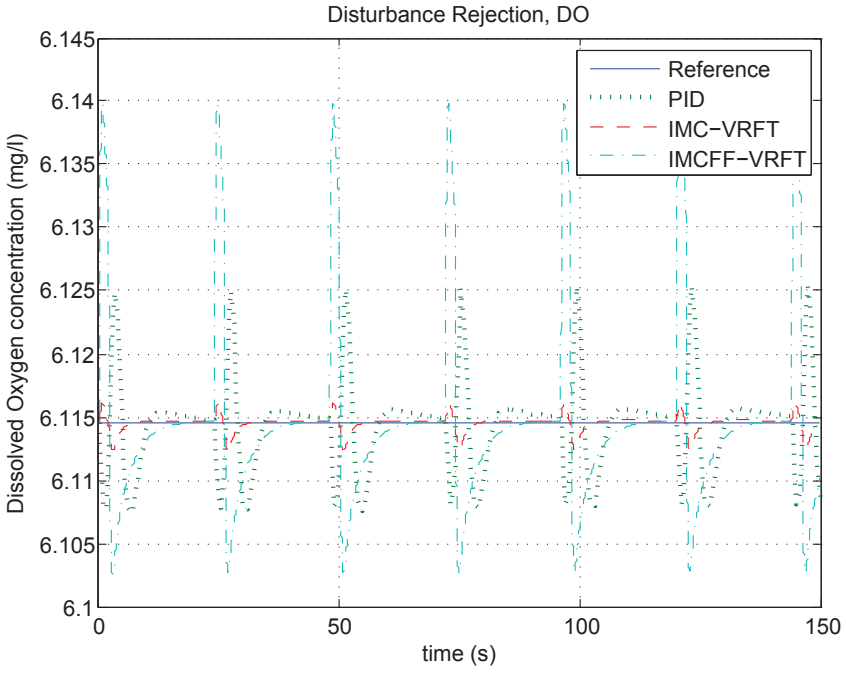


Figure 11: Effect over the dissolved oxygen when the substrate input is disturbed

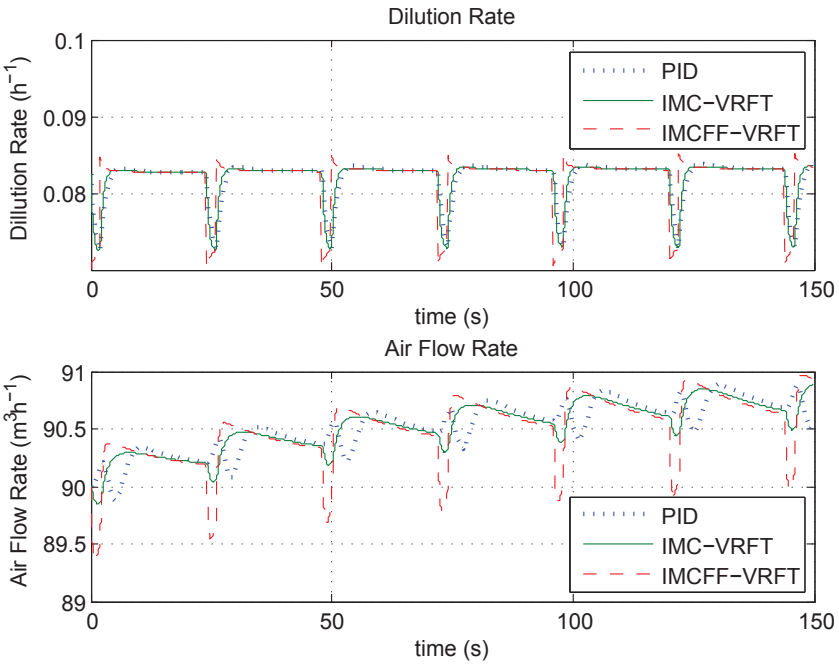


Figure 12: Control effort during the disturbance in the substrate concentration of the influent

Conclusions

In this paper, the VRFT method has been studied and was applied within the IMC framework. Also, a feedforward extension to the IMC-VRFT controller was also presented. Both methodologies were successfully applied to a WWTP process, substantially improving the results of a continuous time two-degrees of freedom PI controller using a restricted order discrete time controller in the case of the reference tracking. For the disturbance rejection, the feedforward controller greatly improved the performance for the substrate loop but for the dissolved oxygen loop, it was found that the feedforward component degrades the performance. The difference in the constant time and the little effect that has the influent substrate concentration over the dissolved oxygen may be the reason of this poor performance. Data-driven control is a powerful tool that can be easily applied to several control problems and that can be extended to several control structures. How to chose the closed-loop target functions without any knowledge of the plant (only data) and how to guarantee stability and robustness when the controller is found, are subject that still need further research in this control area.

Acknowledgment

The financial support from the University of Costa Rica, under the grants 322-B4-218 and 731-B4-213, is greatly appreciated. Also, this work has received financial support from the Ministerio de Economia e Innovacion of Spain under project DPI2013-47825-C3-1-R.

Bibliography

[1] H. Hjalmarsson, M. Gevers, S. Gunnarsson, O. Lequin (1998), Iterative feedback tuning: theory and applications, *Control Systems Magazine, IEEE*, DOI: 10.1109/37.710876, 18(4): 26–41.

[2] M. Gevers (2002), A decade of progress in iterative process control design: from theory to practice, *Journal of Process Control*, DOI: 10.1016/S0959-1524(01)00018-X, 12(4): 519-531.

[3] W. S. Lee, B. D. O. Anderson, I. M. Y. Mareels, R. L. Kosut (1995), On some key issues in the Windsurfer approach to adaptive robust control, *Automatica*, DOI: 10.1016/0005-1098(95)00092-B, 31(11): 1619–1636.

[4] B. Anderson (2002), Windsurfing Approach to Iterative Control Design, in: *A. P., S. P. A. (Eds.), Iterative Identification and Control: Advances in Theory and Applications*, Springer Verlag, 142-166.

[5] A. Karimi, L. Miskovic, D. Bonvin (2003), Iterative correlation-based controller tuning with application to a magnetic suspension system, *Control Engineering Practice*, DOI: 10.1016/S0967-0661(02)00191-0, 11(9): 1069 - 1078.

[6] A. Karimi, K. van Heusden, D. Bonvin (2007), Noniterative Data-driven Controller Tuning using the Correlation Approach, in: *European Control Conference*, Kos Island, Greece.

[7] G. Guardabassi, S. Savaresi (2000), Virtual reference direct design method: an off-line approach to data-based control system design, *Automatic Control, IEEE Transactions on*, DOI: 10.1109/9.855559, 45(5): 954–959.

[8] M. C. Campi, A. Lecchini, S. M. Savaresi (2002), Virtual reference feedback tuning: a direct method for the design of feedback controllers, *Automatica*, DOI: 10.1016/S0005-1098(02)00032-8, 38(8): 1337-1346.

[9] A. Lecchini, M. Campi, S. Savaressi (2002), Virtual reference feedback tuning for two degree of freedom controllers, *International Journal of Adaptative control and Signal Processing*, DOI: 10.1002/acs.711, 16(5):355–371.

[10] M. Morari, E. Zafirou (1989), *Robust Process Control*, Prentice-Hall International.

[11] U. Jeppsson (1996), Modelling aspects of wastewater treatment processes, Ph.D. thesis, Department of Industrial Electrical Engineering and Automation (IEA) Lund Institute of Technology (LTH), http://www.iea.lth.se/ ielulf/publications/phd-thesis/PhD-thesis.pdf

[12] M. Henze, P. Harremoës, E. Arvin, J. la Cour Jansen (1997), Wastewater Treatment, Biological and Chemical Process, 2nd Edition, Enviromental Engineering, Springer Verlag, New York, USA., series Editors: Förstner, U. and Murphy, Robert J. and Rulkens, W.H.

[13] M. Henze, W. Gujer, T. Mino, M. van Loosdrecht (2002), Activated Sludge Models ASM1, ASM2, ASM2d and ASM3, 1st Edition, Scientific and Technical Report, IWA Publishing, London, UK, 2002.

[14] F. Nejjari, B. Dahhou, A. Benhammou, G. Roux (1999), Non-linear multivariable adaptive control of an activated sludge wastewater treatment process, *International Journal of Adaptive Control and Signal Processing*, 13 (5): 347–365.

[15] N. A. Wahab, R. Katebi, J. Balderud (2006), Multivariable pid tuning of activated sludge processes, *Proc. of the International Control Conference (ICC2006)*, 2006.

[16] S. Caraman, M. Sbarciog, M. Barbu (2007), Predictive control of a wastewater treatment process, *International Journal of Computers Communications & Control* 2(2): 132–142.

[17] F. Koumboulis, N. Kouvakas, R. King, A. Stathaki (2008), Two-stage robust control of substrate concentration for an activated sludge process, *ISA Transactions*, 47(3): 267- 278.

[18] R. Vilanova, R. Katebi, V. Alfaro (2009); Multi-loop PI-based control strategies for the Activated Sludge Process, *Emerging Technologies and Factory Automation, IEEE International Conference on*, 2009.

[19] A. D. Kotzapetros, P. A. Paraskevas, A. S. Stasinakis (2015), Design of a modern automatic control system for the activated sludge process in wastewater treatment, *Chinese Journal of Chemical Engineering*, 23(8): 1340-1349.

[20] M. Nakamoto (2004), An application of the virtual reference feedback tuning for a MIMO process, *SICE 2004 Annual Conference*, Sapporo, Japan, 3: 2208-2213.

[21] F. Previdi, T. Schauer, S. Savaresi, K. Hunt (2004), Data-driven control design for neuroprotheses: a virtual reference feedback tuning (VRFT) approach, *Control Systems Technology, IEEE Transactions on*, DOI: 10.1109/TCST.2003.821967, 12(1): 176–182.

[22] F. Previdi, M. Ferrarin, S. M. Savaresi, S. Bittanti (2005), Closed-loop control of FES supported standing up and sitting down using Virtual Reference Feedback Tuning, *Control Engineering Practice*, DOI: 10.1016/j.conengprac.2004.10.007, 13(9): 1173 -1182.

[23] A. Sala, A. Esparza (2005), Extensions to virtual reference feedback tuning: A direct method for the design of feedback, controllers, *Automatica*, DOI: 10.1016/j.automatica.2005.02.008, 41(8): 1473 - 1476.

[24] Y. Kansha, Y. Hashimoto, M.-S. Chiu (2008), New results on VRFT design of PID controller, *Chemical Engineering Research and Design*, DOI: 10.1016/j.cherd.2008.02.018, 86(8):925 - 931.

[25] D. E. Rivera, M. Morari, S. Skogestad (1986), Internal model control: PID controller design, *Industrial & Engineering Chemistry Process Design and Development*, DOI: 10.1021/i200032a041, 25(1): 252-265.

[26] G. Guardabassi, S. Savaresi (1997), Data-based simultaneous design of composite feedback-feedforward controllers: a virtual input direct design approach, *4th European Control Conference (ECC97)*, Brussels, Belgium, 1997.

[27] F. Nejjari, G. Roux, B. Dahhou, A. Benhammou (1999), Estimation and optimal control design of a biological wastewater treatment process, *Mathematics and Computers in Simulation*, DOI: 10.1016/S0378-4754(98)00158-X, 48(3): 269 - 280.

[28] J. Monod (1949), The growth of bacterial cultures, *Annual Review of Microbiology*, 3(1): 371–394.

[29] L. Ljung (1999), *System Identification, Theory for the User*, 2nd Edition, Prentice Hall, 1999.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(4):538-552, August 2016.

Detecting Topic-oriented Overlapping Community Using Hybrid a Hypergraph Model

G.L. Shen, X.P. Yang, J. Sun

Gui-lan Shen*; Xiao-ping Yang
Information School, Renmin University
Beijng, China
guilan.shen@buu.edu.cn; yang@ruc.edu.cn
*Corresponding author: guilan.shen@buu.edu.cn

Jie Sun
Business School, Beijing Union University
Beijing,China
jie.sun@buu.edu.cn

Abstract: A large number of emerging information networks brings new challenges to the overlapping community detection. The meaningful community should be topic-oriented. However, the topology-based methods only reflect the strength of connection, but ignore the consistency of the topics. This paper explores a topic-oriented overlapping community detection method for information work. The method utilizes a hybrid hypergraph model to combine the node content and structure information naturally. Two connections for hyperedge pair, including real connection and virtual connection are defined. A novel hyperedge pair similarity measure is proposed by combining linearly extended common neighbors metric for real connection and incremental fitness for virtual connection. Extensive experiments on two real-world datasets validate our proposed method outperforms other baseline algorithms.
Keywords: information network, overlapping community detection, topic-oriented, hybrid hypergraph model.

1 Introduction

Community is considered to be a fundamental property of complex network. Despite the variety of complex networks, community often accounts for the functionality of the system [1]. Research of recent years shows that the structure of community is not disjoint. Overlapping is an important property of many real-world networks, i.e., they are naturally characterized by multiple community memberships. For example, a person could join in several hobby groups in social networks; one academic paper could cover a number of fields, etc. It is therefore a very essential work to develop approach for efficient overlapping community detection, which will contribute to the links prediction, collaborative recommendation and influence propagation in many application fields.

Although numerous techniques have been developed for overlapping community detection in recent years, most of them only focus on the structure information for real network. It is well understood, however, that there exist a large quantity of real networks with node content or semantic information, which is referred to as information network, such as www, scientific citation network, and social network. The meaningful detected community of the information network should be topic-oriented, which has two characteristics: the nodes inside one community should have dense connections and consistent or similar topics. Communities identified via those topological methods often incorporate different topics since stronger connections represent the interactions that occur across several different topics, which would confuse the meanings of the topic-oriented community [2].

In this paper, we propose a topic-oriented overlapping community method for information network which combines node content information and link information. Firstly, information network is modeled as a hybrid hypergraph composed of hyperedge that features the collection of nodes with common attributes. For different information network, the node attribute could be represented by interest, tab, word or topic. Secondly, a hyperedge pair similarity calculation method is proposed, which combines the content information by calculating common neighbors of hyperedge pair and structure information by measuring the link relationships between the nodes involved into two hyperedges. Finally, an agglomerative hierarchical clustering algorithm is applied to partition the hybrid hypergraph model into different topic-oriented overlapping communities.

Compared with the existing methods, our method can identify communities from the perspective of both content and link structure for information network. From this result, we can easily find more meaningful communities, such as topics, research fields and so forth. Moreover, with the inherent characteristics of hybrid hypergraph model, overlapping of communities could be identified easily.

We proceed to report our work in the rest of the paper as follows. We discuss the related works in Section 2. In section 3, we propose our approach for identifying the meaningful overlapping communities based on hybrid hypergraph model for information network. In order to verify our approach, we conducted extensive experiments. The experimental design and results analysis are given in Section 4. Finally, a conclusion is drawn in Section 5.

2 Related works

Overlapping community detection using topology. Some methods have been proposed to detect overlapping communities in a network.

LFM proposed by Lancichinetti et al [3] is a kind of algorithms utilizing local expansion and optimization. This method relies on a local benefit function that characterizes the quality of a densely connected group of nodes. LFM expands a community from a random seed node to form a natural community until the fitness function

$$f(c) = \frac{k_{in}^c}{(k_{in}^c + k_{out}^c)^\alpha} \tag{1}$$

is locally maximal, where k_{in}^c and k_{out}^c are the total internal and external degree of the community c, and α is the resolution parameter controlling the size of the communities. After detecting one community, LFM randomly selects another node not assigned to any community to expand another new community. This method obviously can identify the overlapping community, since they allow a single node to be put into different community owing to different optimization process.

In real-world network, it's difficult to decide how many communities that a single node should be put in, however it's very clear whether the edge incident on the node is in the community or not. Now, researcher suggests using links to defining community [4], owing to an edge only is in one community, but the nodes connected by edge could be put into different communities. Some methods [5–7] using line graph and link partitioning to detect overlapping community have been proposed. Among them, Ahn [6] partitions links into clusters via hierarchical clustering of edge similarity. Given a pair of links e_{ik} and e_{jk} incident on a node k, the edge pair similarity can be computed via the Jarracd Index defined as,

$$sim(e_{ik}, e_{jk}) = \frac{|Nb_+(i) \cap Nb_+(j)|}{|Nb_+(i) \cup Nb_+(j)|} \tag{2}$$

Where $Nb_+(i)$ is the inclusive neighbors of a node i, which the set contains the node itself and its neighbors. With this similarity, single-linkage hierarchical clustering is then used to build a link dendrogram. Cutting this dendrogram at a special threshold yields link communities. Although the link partitioning method can detect the overlapping communities naturally, there is no guarantee that is provides high quality detection for information network because the method only relies on links of the network while ignores the node content totally.

There are many other methods to detect overlapping community. For example, the ones based on subgraphs, such as CPM [8], CPMw [9] etc.al, treat community structure as the composition of adjacent subgraphs, as one node can belong to several subgraphs. However, these methods are usually considered to solve the pattern matching of complex networks rather than finding communities. In addition, the methods extended Girvan and Newman's divisive clustering algorithm [10], such as CONGA [11], CONCO [12], allow a node to split into multiple copies.

Despite the use of different techniques, the above methods can always detect overlapping dense connections in network. However, they only focus on the topology information but ignore the content information that contributes to improve the quality of the community [13].

Topic-oriented community detection using topology and content. Based on the assumption that the content information can improve the quality of the detected community, various approaches have been combined the links and contents for community detection. Some approaches have combined content information with structure information for community discovery. One of them is generative probabilistic modeling which considers both contents and links as being dependent on one or more latent variables, and then estimates the conditional distributions to find community assignments. PLSA-PHITS [4], Community-User-Topic model [15] and PMC [16] are three representatives in this category. Other fusing the content and structure methods, such as SA-clustering [17] via augmenting the underlying network to take into account the content information, heuristic algorithm CKC [18] to solve the connected k-Center problem, subspace clustering algorithm [19] on graphs with feature vectors.

Different from those methods using topology, these methods account for the content information of the nodes, so the division results for the network is more cohesive in the topics. However, considering the content of nodes, the complexity of the algorithm is greatly increased which will lead to some new challenges, such as how to deal with the high dimensional sparse for node attributes. Furthermore, those methods are not designed for overlapping communities.

3 Methodology

In this section, we present our method for fusing structure and content via hybrid hypergraph to detect the overlapping communities for information networks. Firstly, we present the definitions, and then introduce how to build the hybrid hypergraph model for information network. Thirdly, we give the method that how to measure hyperedge pairs similarity. Finally, we briefly introduce algorithmic details of HLP (Hyper Link Partition), a novel method extended from link graph partition algorithm.

3.1 Definitions

Link partitioning method is a kind of topology methodology based on classic graph theory. It's very simple and distinct to model information networks as simple graphs, in which nodes indicate entity object, and links indicate the binary relationships between node pairs. However, real information networks are characterized by node content attribute, hence simple graph is not suitable for representing the content information. As the generalization of simple graph,

hypergraph can represent the multiple relationships for nodes in finite set, and describe the relationship between general discrete structures, which overcome the defect of the knowledge represented by the simple graph. Hypergraph characterized by one hyperedge incident on any number of nodes, is a graph in generalization. The definition of hypergraph is provided as follows.

Definition 1. A hypergraph [20], $H = (V, E)$ is defined as a set of vertices $V = \{v_1, v_2, \cdots, v_n\}$, and a set of hyperedges $E = \{e_1, e_2, \cdots, e_m\}$, where:

(1) $e_1 \neq \phi (i = 1, 2, \cdots, m)$

(2) $\bigcup_{i=1}^{m} e_i = V$

According to definition, a hyperedge essentially is the set of vertices which are independent. That is, hypergraph cannot represent the original topology of vertices. Thus, we give the definition of hybrid hypergraph.

Definition 2. A Hybrid Hypergraph, $HH = (V, E, \varepsilon, \psi)$ is defined as a set of vertices $V = \{v_1, v_2, \cdots, v_n\}$, a set of hyperedges $E = \{e_1, e_2, \cdots, e_m\}$ and a set of edges ε where:

(1) $e_i \neq \emptyset$ $(i = 1, 2, \cdots, m)$

(2) $\bigcup_{i=1}^{m} e_i = V$

(3) $\psi(\varepsilon_i) = (v_i, v_j)$ $(i, j = 1, 2, \cdots, n)$

3.2 Modeling information network as hybrid hypergraph

We need to extend a structural graph with tuples describing node attributes. This can be formally expressed as a quad $AG = \{V, \varepsilon, F_V, \psi\}$, where each node v is associated with a feature vector $f(v)$. F_V is the set of features for all nodes, where, $f(v) \subseteq F_V, v \in V$. Feature selection is an important issue in system anomaly detection applications [21]. With different node attributes in different information network, the feature vector $f(v)$ can be varied as a topic, a keyword, a place, an author, an activity. The number of features of F_V is m, formally, $m = \left| \bigcup_{v \in V}^{f(v)} \right|$.

So, the question of how to build the information network as the hybrid hypergraph model is simplified as how to map a quad attribute graph AG into a quad HH. Here, we take each feature f_i as a basic unit to build hyperedge e, when $f_i \in f(v)$, the node $v \in e$

We use incidence matrix and adjacent matrix to represent the data structures with related to hybrid hypergraph.

The $N_v \times M_E$ Incidence matrix for a hybrid hypergraph HH, say I, is defined as that

$$I_{ve} = \begin{cases} 1, & if\ v \in e \\ 0, & otherwise \end{cases} \tag{3}$$

The $M_E \times N_V$ transposed matrix for I, say K, is defined as that

$$K_{ev} = \begin{cases} 1, & if\ v \in e \\ 0, & otherwise \end{cases} \tag{4}$$

The $N_V \times N_V$ adjacent matrix for a hybrid hypergraph HH, say A, is defined as that

$$A_{ij} = \begin{cases} 1, & if\ (i, j) \in \varepsilon \\ 0, & otherwise \end{cases} \tag{5}$$

The $M_E \times M_E$ similarity matrix for a hybrid hypergraph HH, say Sim, is defined as that

$$Sim_{ij} = simlarity(e_i, e_j) \quad e_i, e_j \in E \tag{6}$$

3.3 Similarity for hyperedge pairs

In this section, we present the method of how to calculate the similarity of hyperedge pairs in HH. This technique was originally introduced by link partitioning algorithm and local expansion algorithm for the purposes of identifying the overlapping communities in network. However, Link partitioning algorithm and local expansion algorithm both only focus on topological information, but ignore the content information.

We argue the HH model for information network can fuse the content information and structure information naturally. According to the definition of HH, a hyperedge can be regarded as a sub-community characterized by a special feature, because the hyperedge is a set of nodes with the same feature. We know clearly that the sub-community represented by the hyperedge is different from the final detected community. The reason is that one node is usually associated with more than one feature, but a hyperedge only reflect one of the features. Nevertheless, we can determine whether the hyperedge pairs have the same topic or not by exploring the link relationship between them. If the node only has one feature, moreover, such as topic, our work can be simplified as the work presented by zhao [22].

The relationship between two hyperedges is the link between two node sets which is more complicated, rather than just that between the two nodes in the simple graph. The link type for two hyperedges comes into two kinds: one kind is shared common nodes; the other is that the nodes in one hyperedge are connected with those in the other hyperedge. In order to better illustrate this, we offer two formal definitions as follows.

Definition 3. Given two hyperedge e_i and e_j, is real connection, where $e_i \cap e_j \neq \emptyset$.

For instance, two hyperedges e_1, e_2 are shared with two common nodes v_2, v_3 (Fig1.a), that is $e_1 \cap e_2 = \{v_2, v_3\} \neq \emptyset$, therefore the link type of e_1, e_2 is real connection.

Definition 4. Given two hyperedge e_i and e_j, is virtual connection, where

(1) $e_i \cap e_j \neq \emptyset$

(2) $\forall \varepsilon \in \{< v_m, v_n > | \exists v_m \in e_i, \exists v_n \in e_j\} \neq \emptyset$

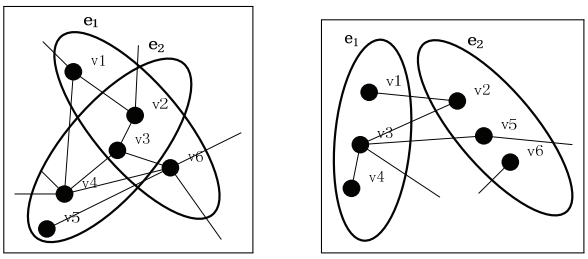
Or

(3) $e_i \cap e_j \neq \emptyset$

(4) $\forall \varepsilon \in \{< v_m, v_n > | \exists v_m \in e_i - e_i \cap e_j, \exists v_n \in e_j - e_i \cap e_j\} \neq \emptyset$

For instance, as shown in Fig1.b $e_1 \cap e_2 = \emptyset$, and, there are links $< v_1, v_2 >$, $< v_2, v_3 >$ between nodes for e_1 and e_2, this is one case of virtual connection. Another case is also shown in Fig1.a, $e_1 \cap e_2 = \{v_2, v_3\} = \emptyset$, considering edge $< v_4, v_6 >$, which match the conditions (4) $v_4 \in e_1 - e_2 \cap e_2$ and $v_6 \in e_2 - e_1 \cap e_2$. Therefore, hyperedge e_1 and e_2 has both real connection and virtual connection.

Obviously, when measuring the hyperedge similarity, both virtual connection and real connection are meaningful. Virtual connection reflects the structure information of the original network essentially, while real connection reflects the number of nodes with common attributes. Intuitively, hyperedge similarity is dependent on the tightness of hyperedge pairs. As mentioned above, hyperedge is the group of nodes; therefore, measuring the tightness of hyperedge pair can be converted into how to measure the tightness of two groups of nodes. If there are a lot of links between two sets of nodes, the two sets are strongly tight. For instance, in citation network, if article a1 with keyword k1 cited article a2 with keyword k2, then k1 and k2 have certain correlation or similarity. Similarly, the more articles involving the keyword k1 are cited by the articles involving k2 keyword, that is, k1 set has strong tightness with k2 set, which turns out that k1 and k2 have higher correlation.



(a) Real connection (b) Virtual connection

Figure 1: Two kinds of link type for hyperedge

To better quantify the tightness between the two groups of nodes, we extend the local fitness function in LFM [3]. Local fitness function for a given local community S, is formally given as

$$f_s = \frac{k_{in}^s}{k_{in}^s + k_{out}^s} \tag{7}$$

Where k_{in}^s and k_{out}^s are the total internal and external degree of the community.

Fitness function f_s can measure the internal and external tightness for a local community. In topology detecting community methods, it has shown a good performance, and was expanded or applied by more researchers [23, 24]. Inspired by this thought, we propose the fitness function for a hyperedge e to be given as

$$f_e = \frac{k_{in}^e}{k_{in}^e + k_{out}^e} \tag{8}$$

Where k_{in}^e and k_{out}^e are the total internal and external degree of the hyperedge. The total of k_{in}^e and k_{out}^e is the total degree of all nodes in this hyperedge. Similar to f_s, f_e can measure the internal and external tightness for hyperedge. Whether to merge two hyperedge into a larger hyperedge depends on how the changed numbers of virtual connection influence the fitness of the combined hyperedge. We define incremental fitness for combined hyperedge as the similarity for virtual connection of two hyperedges. Given two hyperedge e_i and e_j

$$Sim_{virtual}(e_i, e_j) = \Delta f_{ij} = f_{e_i} \cup f_{e_i} = \frac{k_{in}^{e_i e_j}}{k_{in}^{e_i e_j} + k_{out}^{e_i e_j}} \tag{9}$$

Where $k_{in}^{e_i e_j}$ is the numbers of virtual connection between e_i and e_j. The total of $k_{in}^{e_i e_j}$ and $k_{out}^{e_i e_j}$ is the total degree of all nodes in merged hyperedge. For instance, in Fig2, considering two hyperedge e_1 and e_4, $k_{in}^{e_1 e_4} = 4$, $k_{in}^{e_1 e_4} + k_{out}^{e_1 e_4} = 32$, so, $\Delta f_{14} = f_{e_1} \cup f_{e4} = \frac{1}{8}$. Another example in this figure is, $k_{in}^{e_3 e_4} = 3$ and$k_{in}^{e_3 e_4} + k_{out}^{e_3 e_4} = 41$, so, $\Delta f_{34} = f_{e_3} \cup f_{e4} = \frac{3}{41}$.

Whether to merge two hyperedges into a larger hyperedge depends on the value of incremental fitness function.

Virtual connection reflects the tightness of hyperedge pair via the structure information, while, real connection reflects the semantic similarity of hyperedge via the content information. In our method, we do not directly calculate the similarity of the features implied by hyperedges, such as that of keyword or topic, instead, we evaluate the similarity via CN metric. CN(Common Neighbors) [25] is also called structural equivalence, namely, the nodes are similar if they share a lot of common neighbors. CN is one of the most widely used metrics when measuring the similarity in local community detection methods. Therefore, we extend the CN metric to make it suitable for measuring the similarity between two hyperedges, or, two groups of nodes. To define clearly the neighbors set for hyperedges, we give the following definitions.

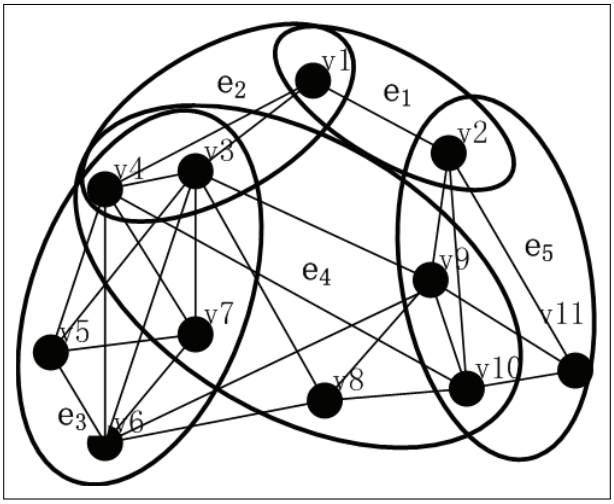


Figure 2: A sample of hybrid hypergraph model

Definition 5 (Inductive nodes set)**.** Given two hyperedge e_i, e_j in HH, where e_i, e_j has real connection, the Inductive nodes set for e_i related to e_j, say $p_j(e_i)$ is formally as $p_j(e_i) = \{v|v \in e_i - e_i \cap e_j\}$.

In order to illustrate the neighbors for hyperedges, we specially designate those neighbors related to Inductive nodes set.

Definition 6 (Extended Common Neighbors of Inductive nodes set)**.** Given two hyperedge e_i, e_j and inductive nodes set $p_j(e_i)$ in HH, the extended neighbors of $p_j(e_i)$ including $p_j(e_i)$, say $n_+(p_j(e_i))$,is formally as $n_+(p_j(e_i)) = \{x|d(v,x) \leq 1, v \in p_j(e_i)\}$

Where $d(x,y)$ is the distance of two nodes, formally as follows:

$$d(x,y) \leq 1 \; if \; x \in e_i, y \in e_j \; and \; e_i \cap e_j \neq \emptyset \tag{10}$$

In Fig2, $e_1 \cap e_2 = \{v_1\}$, $p_2(e_1) = \{v_2\}$, $p_1(e_2) = \{v_3, v_4\}$, so we can calculate the extended common neighbors of inductive nodes sets $n_+(p_1(e_2)) = \{v_1, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}$, $n_+(p_2(e_1)) = \{v_1, v_2, v_9, v_{10}, v_{11}\}$.

Based on Jaccard index, we propose the method that how to calculate the similarity for hyperedge pair e_i, e_j with real connection. Formally as:

$$Sim_{real}(e_i, e_j) = \frac{|n_+(p_j(e_i)) \cap n_+(p_i(e_j))|}{|n_+(p_j(e_i)) \cup n_+(p_i(e_j))|} \tag{11}$$

In fact, this definition is consistent with the similarity for the link pair in Ahn[14_6] research. For instance, in Fig2, $s(e_1, e_2) = \frac{3}{11}$. We combine the Jarracd index and fitness function to compute the similarity for hyperedge pair e_i, e_j, formally as:

$$Sim_{ij} = Sim(e_i, e_j) = \lambda Sim_{real}(e_i, e_j) + (1 - \lambda) Sim_{virtual}(e_i, e_j) \tag{12}$$

We give the detail of how to compute the similarity for Hyperedge pair in Algorithm 1.

3.4 Stop criterion

In our method, we adopt divisive hierarchical clustering to cluster the hyperedges. The results of hierarchical clustering are presented in a dendrogram. One important job in this method is

Algorithm 1: Computing the similarity for hyperedge pair

Input: e_1, e_2, K
Output: simValue

$j = 0$, $f = 0$
check e_1 and e_2 is whether real connection or virtual connection
If $(e_1 \cap e_2 \neq \emptyset)$
  //real connection
  for each $v \in V$
  if $(v \in e_1$ & $\&v \in e_2)$
    $vComm \leftarrow v$
    //get extended neighbors of Inductive nodes set
  $NIV_1 \leftarrow veNeighbors(K, e_1, vComm)$
  $NIV_2 \leftarrow veNeighbors(K, e_2, vComm)$
If $NIV_1 \neq \emptyset$ && $NIV_2 \neq \emptyset$
  interNum=interSection(NIV_1, NIV_2)
  unionNum=unionSection(NIV_1, NIV_2)
  Jaccard = interNum / unionNum
  //fitness
  fin=blink(e_1, e_2)
  ftotal=totalDegree(e_1)+ totalDegree(e_2)
  f=fin/ftotal
else
  fin=blink(e_1, e_2)
  ftotal=totalDegree(e_1)+ totalDegree(e_2)
  f=fin/ftotal
  simValue = λ*Jaccard + (1 - λ)*f

Figure 3: Algorithm Similarity

deciding the stop criterion for clustering. We define partition density D, as a function of the dendrogram cut threshold. The maximum of D indicates the discovered hyperedge communities are well structured.

For a HH, $\{HP_1, HP_2, \cdots, HP_K\}$ is a partition of the hyperedges into K clusters. Cluster HP_K has m_K hyperedges, and $n_K = \left| \bigcup_{e_i \in HP_K} \{v \in e_i\} \right|$ nodes. Then we define this as normalized form:

$$D_K = \frac{m_{HP_K} - 1}{m_K - 1} \tag{13}$$

Where, $m_{HP_k} = \sum_{i=1}^{n_K} m^i_{HP_K} \Big/ n$ is the mean value for the nodes located in a number of hyperedges. $m^i_{HP_K}$ is the number of hyperedges in which node i is located. The larger D_k indicates the probability, at which nodes are clustering into one cluster, the tighter internal connection in the cluster is. The partition density D, is the average of D_k.

$$D = \sum_{i=1}^{K} D_i \Big/ K \tag{14}$$

The goal of hierarchical clustering is to find K clusters when partition density D is maximal. When D = 1, all clusters are merged into one community.

3.5 HLP algorithm

Based on the HH model for information network, we apply the previously-defined similarity algorithm to pairs of all hyperedges, and produce a similarity matrix S. In the process of clustering, the clustered hyperedge contain more than one initial hyperedge. For instance, $e_i\prime$ has n initial hyperedges, $e_j\prime$ has m initial hyperedges. The Jarracd similarity of $e_i\prime$, $e_j\prime$ is computed as:

$$Sim_{real}(e_i\prime, e\prime_j) = \sum_{t}^{n}\sum_{c}^{m} Sim_{real}(e_i^c, e_j^t) \Big/ n * m \tag{15}$$

We give a simple description of HLP in Algorithm 2. We obtain the clusters of merged hyperedges via this algorithm. The nodes located in each clustered hyperedges constitute the communities related to some special topics.

Algorithm2: Clustering for hyperedges

Input: K

Output: $K\prime$

$S \leftarrow 0$, $m = 0$, $n = 0$, $K\prime \leftarrow K$
m = LengthofRow(K)
n = LengthofCol(K)
//Initialize the similarity matrix for K
for(int $i = 0$; $i < m$; i++)
for(int $j = 0$; $j < n$; j++)
  $S(i, j)$ = Similarity(i, j)
  $S\prime \leftarrow S$
While $m > 1$
//find the coordinates for the maximal similarity in $S\prime$
  $\max_i$ = maxI($S\prime$)
  max_j = maxJ($S\prime$)
  $K\prime \leftarrow$ clustering($K\prime$(, $\max_i$), K(, $\max_j$))
If Density($K\prime$) = maxDensity($K\prime$)
  return $K\prime$

Figure 4: Algorithm HLP

4 Experiments

In this section, we present experiments on real datasets to evaluate the performance of our method. We first applied our method to two datasets to choose the optimal Value of λ. Then we compared the performance of our method with two baseline methods. Before going to details, we first describe the datasets and the method to extract the node feature, and introduce the performance metric to be used in our experiments.

4.1 Datasets

Two real datasets used in our experiments are described in the following:

Cora Dataset: The Cora dataset [26] consists of the abstracts and references of about 34000 computer science research papers. Three subfields of Machine Learning (ML), Programming (PL) and Database (DB) are used and those articles without references to other articles in the set are removed. The detailed information about each subfield is shown in Table1.

Table 1: Cora dataset

Area	*#of subfields*	*#of total papers*	*#of used papers*	*#of links*
ML	7	4218	2708	5249
PL	9	4496	3292	7772
DB	9	1396	1060	2522

WebKB Dataset: The WebKB dataset [27] consists of about 6000 web pages from computer science departments of four schools. The web pages are classified into seven categories, including, staff(SA), course(CS),department(DP), faculty(FA), student(SD),project(PJ),other(OT). We select five categories excluding DP and OT, in that the two categories contain only a few pages. The detailed information about these web pages is shown in Table 2.

Table 2: WebKB dataset

School	*#of CS*	*#of FA*	*#of SD*	*#of PJ*	*#of SA*	*#of Total*	*#of links*
CN	44	34	18	128	21	245	304
TX	36	46	20	148	2	252	328
WA	77	30	18	123	10	258	446
WI	85	38	25	156	12	316	530

4.2 Preprocessing

For Cora, we treated each subfield as an independent dataset. After stemming and removing stopwords we were left with vocabulary of stemmed unique words for each subfield respectively. All words with document frequency less than 15 were removed. Taking DB dataset as an example, a vocabulary of size 5911 unique words was reduced to size 1086 words by removing the low document frequency words. The word cloud for preprocessed DB dataset is shown in Fig.5

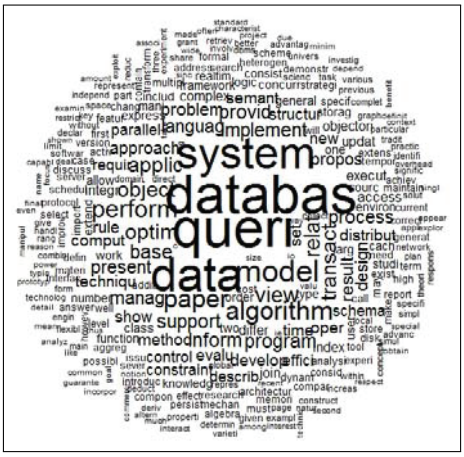


Figure 5: Word cloud for DB dataset

The same preprocessing is handled by WebKB. We extracted words whose document fre-

quency is more than 10.

4.3 Evaluation metrics

In our paper, we focus on the topic similarity of the detected overlapping communities, therefore when the ground-truth community is known, we utilize two measures of purity and NMI to evaluate the quality of overlapping communities detected by different methods. Purity measures the internal topic similarity within the community, and NMI is the most widely used measure to account for overlapping communities.

Given the ground-truth community structure, $G = \{G_1, G_2, \cdots, G_S\}$ where G_S contains the set of nodes that are in the s^{th} community. The community structure given by the algorithms is represented by $C = \{C_1, C_2, \cdots, C_S\}$, where C_k contains the set of nodes that are in the k^{th} community.

The purity of C_i is defined as:

$$Purity(C_i) = \frac{1}{|C_i|} \max_j \{C_i \cap G_j\} \tag{16}$$

Usually, the detected community C_i includes nodes that belong to other G_j in the ground-truth. For C_i, we compute the intersection set with each standard community G_j, then take the maximum as the final purity for it.

The purity of C is defined as:

$$Purity(C) = \frac{1}{K} \sum_{i=1}^{K} Purity(C_i) \tag{17}$$

The higher the purity, the better the communities are partitioned from the perspective of topics.

The mutual information between G and C is defined as

$$MI(G, C) = \sum_{x \in G, y \in C} p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \tag{18}$$

The NMI(normalized mutual information) is defined by

$$NMI(G, C) = \frac{MI(G, C)}{\max(H(G), H(C))} \tag{19}$$

where $H(G)$ and $H(C)$ are the entropies of the partitions G and C. The higher the NMI, the closer the partition is to the ground truth.

4.4 Optimal value of λ

As we discussed in section 3, the parameter λ balances the Jaccard index and fitness function value when compute the similarity of hyperedge pairs. We perform experiments to study how the λ value affects the purity of detected communities. We set the step 0.1, the result is shown in Figure 4.

The result shows that λ value is decided by the structure of network and the number of hyperedges. Jaccard index and fitness function value both affect the purity of the detected communities. It is proved that the structure and content information both have influence on the topical community detection. However, we observe that the characteristics of information networks detemine the value of parameter λ . Different network will lead to different λ . This is

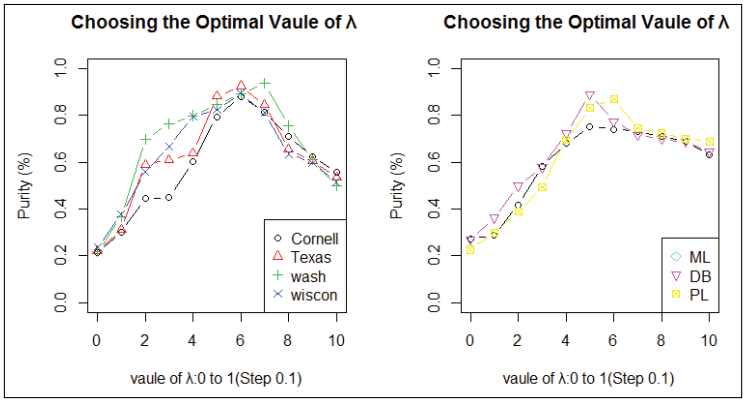


Figure 6: The result of choosing parameter λ

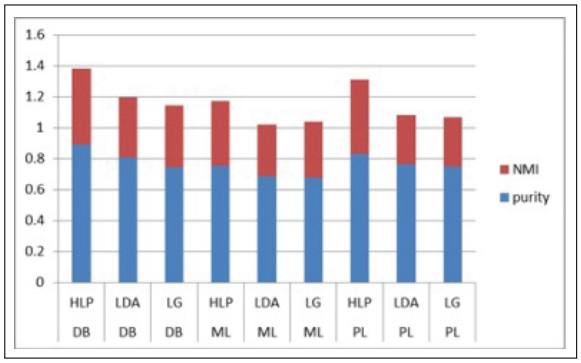
because there are noise problems to various degree, including both links and node content in the information network. For WebKB dataset,the best performace is achieved when "$\lambda = 0.6$", and for Cora dataset, the optimal value "$\lambda = 0.5$" which are used as default settings for the following experiments.

4.5 Results

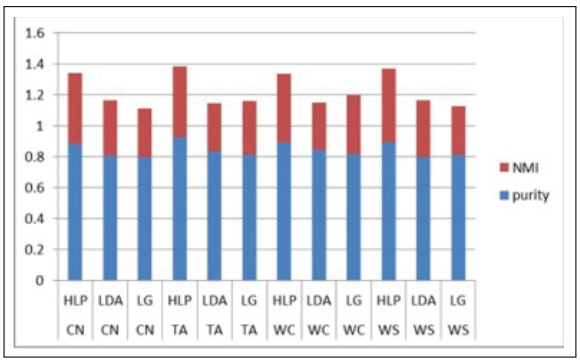
To evaluate the effectiveness of HLP, we compare our method with two baseline methods: one is topology-based method, Line graph partition [6], the other is LDA to cluster the nodes by using content information only.

We use purity and NMI quantifying the performance each algorithm.

The details are shown in Figure 5, which illustrates HLP achieve the best performance in real information networks. From the results, we also can observe some interesting things. In some datasets, such as DB,PL,CN and WS, LDA algorithm can achieve better performance than Line Graph algorithm, In some other datasets, nevertheless, such as ML,TA and WC, the results are opposite. This confirms our assumption again that both node information and link information affect the quality of detected overlapping communities. Therefore, we are sure that the combination of node information and link information can improve the quality of overlapping community detection.



(a) Cora dataset



(b) WebKB dataset

Figure 7: The evaluations of community algorithms over two real information networks.

Conclusion

In this paper, we propose a topic-oriented overlapping community detection approach based on hierarchical clustering for hybrid hypergraph model, which can combine the content and structure information of information network naturally. Considering the complex of the hybrid hypergraph model, we classify the connections of hyperedges into real connection related to content information, and virtual connection related to structure information. We present incremental fitness to evaluate the tightness for hyperedge pairs in virtual connection. Meanwhile, we extend CN metric on hyperedge pairs to conduct the semantic similarity calculation in real connection. In order to balance the influence of two connections, we combine linearly the two measures for similarity of hyperedge pairs. The density function is employed to determine the appropriate number of communities. To evaluate the performance, we conducted experiments on two real datasets. Compared with the benchmark, Line graph partition algorithm focusing on topological detection, LDA focusing on clustering node contents, our approach gained a better performance in information network. Furthermore, the overlapping communities detected by our approach were more meaningful since they are topic-oriented.

Our approach has many potential applications. It can be applied to many kinds of information networks, where nodes contain content. With our method detecting the communities, we are able to improve the efficiency of collaborative scientific research, discover experts for each topic, and analyze topic-oriented influence propagation.

Future work includes qualifying the weight of each node in the hyperedge to improve the purity of detected communities. We also intend to take the time factor into account, so that we can detect the evolution communities.

Acknowledgment

This paper is supported by Natural Science Foundation of China (No.71572015), Scientific Research Project of Beijing Union University (No. Zk10201506), Beijing Higher Education Young Elite Teacher Project (No.YETP1503).

Bibliography

[1] Cobanoglu B, Zengin A, Ekiz H, et al (2014); Implementation of DEVS Based Distributed Network Simulator for Large-Scale Networks [J]. *International Journal of Simulation Modelling* (IJSIMM), 13(2): 147-158.

[2] Ding Y. (2011), Community detection: topological vs. topical, *Journal of Informetrics*, DOI: 10.1016/j.joi.2011.02.006, 5(4): 498-514.

[3] Lancichinetti, Andrea, Santo Fortunato, and János Kertész (2009); Detecting the overlapping and hierarchical community structure in complex networks, *New Journal of Physics*, 11(3): 033015.

[4] Xie J., Kelley S., Szymanski B. K. (2013), Overlapping community detection in networks: The state-of-the-art and comparative study, *ACM Computing Surveys* (CSUR), 45(4): 43-79.

[5] Evans T. S., Lambiotte R. (2009), Line graphs, link partitions, and overlapping communities, *Physical Review E*, DOI:http://dx.doi.org/10.1103/PhysRevE.80.016105, 8(1): 92-105.

[6] Ahn Y. Y., Bagrow J. P., Lehmann S. (2010), Link communities reveal multiscale complexity in networks, *Nature*, doi:10.1038/nature09182, 466: 761-764.

[7] He, C., Ma, H., Kang, S., Cui, R. (2014), An Overlapping Community Detection Algorithm Based on Link Clustering in Complex Networks, *In Military Communications Conference (MILCOM) IEEE*, 865-870.

[8] Palla, G., Derényi, I., Farkas, I., & Vicsek, T. (2005), Uncovering the overlapping community structure of complex networks in nature and society, *Nature*, 435: 814-818.

[9] Farkas, I., Ábel, D., Palla, G., & Vicsek, T. (2007), Weighted network modules, *New Journal of Physics*, 9: 80-198.

[10] Girvan M., Newman M. E. J. (2002), Community structure in social and biological networks, *Proc. of the National Academy of Sciences*, 99: 7821-7826.

[11] Gregory S. (2007), An algorithm to find overlapping community structure in networks, Knowledge discovery in databases: PKDD 2007, *Springer Berlin Heidelberg*, 91-102.

[12] Gregory S. (2008), A fast algorithm to find overlapping communities in networks, Machine learning and knowledge discovery in databases, *Springer Berlin Heidelberg*, 408-423.

[13] T. Yang, R. Jin, Y. Chi, S. Zhu (2009), Combining link and content for community detection: a discriminative approach, *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, Paris*, 927-936.

[14] Hric D., Darst R. K., Fortunato S. (2014), Community detection in networks: structural clusters versus ground truth, *arXiv preprint* arXiv:1406.0146.

[15] Hofmann, David Cohn Thomas (2001), The missing link-a probabilistic model of document content and hypertext connectivity, *Proceedings of the 2000 Conference on Advances in Neural Information Processing Systems*, Vancouver, 430-436.

[16] D. Zhou, E. Manavoglu, J. Li, C. Giles, and H. Zha (2006), Probabilistic models for discovering e-communities, *In Proceedings of the 15th international conference on World Wide Web*, Banff, 173-182.

[17] Yang, B., Di, J., Liu, J., Liu, D. (2013), Hierarchical community detection with applications to real-world network analysis, *Data & Knowledge Engineering*, 83: 20-38.

[18] M. Ester, R. Ge, B. Gao, Z. Hu, and B. Ben-Moshe (2006), Joint cluster analysis of attribute data and relationship data: the connected k-center problem, *Proceedings of the 2006 SIAM International Conference on Data Mining*, Maryland, USA, 25-46.

[19] Günnemann S, B. Boden, and T. Seidl (2011), Db-csc: a density-based approach forsubspace clustering in graphs with feature vectors, Machine Learning and Knowledge Discovery in Databases, *Springer Berlin Heidelberg*, 565-580.

[20] Berge, Claude (1989), *Hypergraphs: Combinatorics of Finite Sets*, North Holland.

[21] Zhou X. et al.(2014); Information-value-based feature selection algorithm for anomaly detection over data streams [J]. *Tehnički vjesnik*, 21: 223-232.

[22] Zhao, Z., Feng, S., Wang, Q., Huang, J. Z., Williams, G. J., Fan, J. (2012), Topic oriented community detection through social objects and link analysis in social networks, *Knowledge-Based Systems*, 26: 164-173.

[23] Darst R. K., Nussinov Z., Fortunato S. (2014), Improving the performance of algorithms to find communities in networks, *Physical Review E*, 89(3): 42-58.

[24] McAuley J., Leskovec J. (2014), Discovering social circles in ego networks, *ACM Transactions on Knowledge Discovery from Data* (TKDD), 8(1): 10-16.

[25] Lorrain F., White H. C. (1971) , Structural equivalence of individuals in social networks, *The Journal of mathematical sociology*,1: 49-80.

[26] Rayid Ghani (2014), CMU World Wide Knowledge Base (WebKB) project, Jan, 2001.[Online]. Available: http://www.cs.cmu.edu/∼webkb. [Accessed: April 9, 2014]

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(4):553-566, August 2016.

A Hybrid Model for Concurrent Interaction Recognition from Videos

M. Sivarathinabala, S. Abirami

M. Sivarathinabala*
Department of Information Science and Technology,
Anna University, Chennai, India.
*Corresponding author: sivarathinabala@gmail.com

S. Abirami
Department of Information Science and Technology,
Anna University, Chennai, India.
abirami_mr@yahoo.com

Abstract: Human behavior analysis plays an important role in understanding the high-level human activities from surveillance videos. Human behavior has been identified using gestures, postures, actions, interactions and multiple activities of humans. This paper has been analyzed by identifying concurrent interactions, that takes place between multiple peoples. In order to capture the concurrency, a hybrid model has been designed with the combination of Layered Hidden Markov Model (LHMM) and Coupled HMM (CHMM). The model has three layers called as pose layer, action layer and interaction layer, in which pose and action of the single person has been defined in the layered model and the interaction of two persons or multiple persons are defined using CHMM. This hybrid model reduces the training parameters and the temporal correlations over the frames are maintained. The spatial and temporal information are extracted and from the body part attributes, the simple human actions as well as concurrent actions/interactions are predicted. In addition, we further evaluated the results on various datasets also, for analyzing the concurrent interaction between the peoples.
Keywords: Pose prediction, interaction recognition, layered HMM

1 Introduction

Human interaction analysis involves human activities that are happening between two or more persons to understand the interaction. The automation is required in video surveillance to detect activities from the videos and infer some useful information without the intervention of human beings. The type of human activities detected mainly depends upon the domain in which surveillance system has been employed. Human activity recognition may be used for behavior pattern observation or for suspicious activity detection. The activities can either be detected or reported during the event when it takes place or it can be predicted in advance. A system or framework to understand human interaction from surveillance videos involves the following key components: a) Low level components for Background modeling, Feature extraction and Object tracking, b) Middle level components for Object classification c) High level components for Semantic interpretations (ie, understanding actions, interactions between two / multiple people). Significant works have been progressing in the literature for each level in this framework. This work mainly focuses on higher level components in order to predict the human interactions. Human Interaction Recognition (HIR) involves activity recognition of humans to understand their behaviors. Human activity recognition has been presented by ([2], [19]) as, single layered and hierarchical approaches like space- time volumes, space- time trajectories, space-time features and state based models such as HMM and DBN. From these approaches, human activity such

as shaking hands, punching, pushing, pointing, picking up the object, throwing are recognized. There exist some limitations such as difficulty in recognizing the interactions that happened between multiple people in varying time difference in when he/she re-enters in the scene, difficulty in distinguishing the poses when transformation and scaling has been performed, variation in the performance of body part detection, difficulty to recognize the interactions in complex environments.

2 Related works

Activity analysis involves two fold actions: (i) Analysis of motion patterns (ii) Understanding of High level descriptions of actions/interactions happening place among humans or in an environment. Activities can be recognized [2] in single layered approaches and in hierarchical approaches. In Single layer approaches human activities could be recognized based on the image sequences and it is suitable for gesture/action recognition. In contrast, Hierarchical approaches recognize high level activities which are complex in nature. It has been observed from the literature that the hierarchical approach well suits to recognize high level activities (Interactions). Thus, many researchers proposed a level of HMM differently as Hierarchical HMM, Semi- HMM, 2D- HMM, factorial HMM, Coupled HMM, Asynchronous IO HMM etc., This paper attempts to analyze the Interactions between two or more persons in a new fashion of Hidden Markov Model .

Hidden Markov Model (HMM) plays a vital role in determining activities which are vision based. The research works ([11], [13], [14], [12], [20])proves that HMM is one of the most appropriate models for person activity prediction. [9] stated that, a layered hidden Markov model (LHMM) can obtain different levels of temporal details while recognizing human activity. The LHMM is a cascade of HMMs, in which each HMM has different observation probabilities processed with different time intervals. [23] has proposed layered hidden Markov model (LHMM) as a statistical model which is derived from the hidden Markov model (HMM).

In HMM, the activities are recognized with less temporal correlated frames and over fitting problem occurs during the calculation of observation probability. The activities that take place in long temporal difference cannot be identified with good accuracy. Moreover, the use of single variable state representation makes more difficult to model the complex activities involving multiple interacting agents. In order to solve the over fitting problem in the HMM, Human action recognition [8] has been done using three layered HMM. Their system was capable of recognizing six distinct actions, including Raising the right arm, Raising the left arm, Stretching, Waving the right arm, Waving the left arm, and Clapping. From the observations made in the survey process, most of the previous work centered on identification of particular activities in the particular scenario and less effort has been done to recognize interactions.

Many research works ([21], [22], [3], [13]) have been done using graphical models such as HMM and CRF. A conventional HMM can have many mathematical structures and has proved its simplicity in recognizing temporal events. Its only limitation in conventional HMM is, it cannot capture high temporal correlated frames since the output depends only on the current states. Our work is different in recognizing interactions (i.e., actions between two or more persons) in the group. When there are multiple persons in the scene, one person may stand still (or) not doing any actions (or) not interacting with any other persons in the group. That particular person's activity may be considered as abnormal activity. In order to identify the abnormal activity in the particular frame, we are interested in concurrent interactions between a group of peoples. As a result, a three layered HMM has been designed in our work to recognize the actions and interactions in the group. In the first layer, pose of the persons has been identified and in the second layer ie., the action layer in which actions of the individual persons has been recognized

and in the third layer, the actions of the first person and the second person are coupled in order to identify the interaction of the people. This three layered HMM able to handle temporal correlations between the frames.

This paper has been motivated to design a human interaction analysis system which can overcome these limitations and to design an automated smart surveillance system which could predict actions/interactions taking place in public environments. As motivated by the above challenges the following contributions arose in our work: a) Learning model has been designed and implemented in order to learn the complex activity/interaction, b) joint form of LHMM and CHMM provides concurrent interactions between the persons c) proposed model has been validated using different interaction recognition datasets and self generated data set.

3 Preprocessing and feature extraction

Pre-processing includes two phases called Background modeling and Foreground segmentation. This phase highlights the portion of the frame which is under motion. Background modeling [5] is a process which tries to model the current background of the scene precisely and aids to segment the foreground objects from the background. Foreground segmentation has been performed to obtain the foreground image from the actual image.

Feature extraction starts with contour extraction [20] and the region properties of the frame have been considered, the number of objects in each frame has been found out. Object centroid and object area have been calculated from the region properties as the next set of features. The human body parts need to be modeled before performing the pose estimation. Here, to employ a silhouette based approach ([6], [15], [16], [14]) in body part modeling, convex hull technique has been adapted. Here, the convex hull points have been obtained for the whole blob to construct the skeleton. A minimum of 5 dominant points such as head, left hand, right hand, left leg and right leg that lie on the convex hull polygon has been chosen in our observation. Body part modelling [4] differs for each and every pose since the pose stand may completely vary from the pose sit. In order to predict every pose precisely, the height of the human is divided into four quadrants, upper most quarter, upper middle quarter, lower middle quarter, and lower most quarters. From the contour (sketch) of the human body, the location of the hands, head and legs have been exactly predicted using the convex hull co-ordinates that lie in the arrangement of the four regions mentioned above. This type of body part modelling can better suit to any kind of pose prediction.

4 Human interaction recognition

In our work, three layers of HMM have been framed in order to identify the group activity in varying time intervals. Figure 1 shows the overview of the Human Interaction Recognition System. This layered representation provides outputs at different levels and decompose the levels in different time granularity. Hidden Markov Model (HMM) [7] is the most successful approach to modeling and classifying dynamic behaviors. Layered Hidden Markov Model (LHMM) and Coupled Hidden Markov Model (CHMM) are combined together to enhance the robustness of the interaction analysis system by reducing the training parameters. Effective learning process has been carried out by combining max-belief algorithm and Baum-Welch algorithm. Max-belief algorithm is used to derive the most likely sequence of results in observation activities and the Baum-Welch algorithm reduces the inference errors.

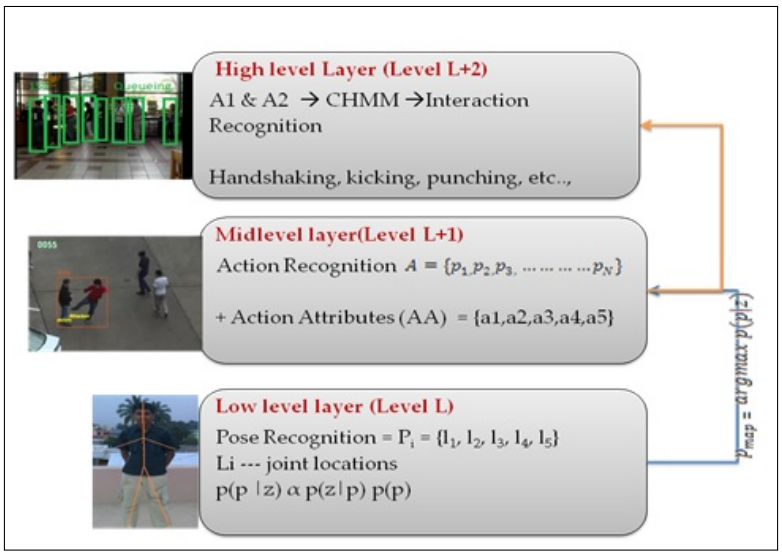


Figure 1: Overview of human interaction recognition system

5 Proposed learning model

The visual features are extracted from the videos and the feature vector has been derived from the observations and shown in figure 2. Feature vector observation includes spatial information $S_i(t)$ and temporal information T_N where t corresponds to the time stamp within the set of frames and T_N represents the trajectory information for N number of frames. Z is the feature set and it is represented as $Z = S_1(t), S_2(t), S_3(t), ...S_N(t), T_1, T_2, T_3, ...T_N$. Here, spatial representations have been given as joint locations.

Let the layered HMM1 models the observations of the person 1 and layered HMM2 models the observation of person 2 respectively. The Observations of the person 1 and 2 have been given as input to the layered HMM1 and layered HMM2 respectively. Let $P_1, P_2, ...P_n$ represents the pose of the person that has been shown in Pose layer (P-HMM) and defined as layer 1i and layer 1j for person 1 and 2 respectively. The Action layer (A-HMM) for person 1 and 2 is defined in layer 2i and layer 2j. Pose as the feature vector has been given as input to this layer and action of the person has been identified. In order to provide high level information action attributes has been added to the action. After identifying the action attributes, it has been manually labeled. Action and attributes of person 1 and action and attributes of person 2 has been coupled together to recognize the new interaction. The Interaction layer (I-HMM) has been considered as layer 3 and interactions are represented as $I_1, I_2, I_3, ...I_n$.Because of coupling the A1- HMM and A2-HMM are coupled so that interactions between two persons have been recognized.

6 Layers in HIR model

6.1 Pose layer (low level layer)

Pose is defined as the preliminary motion sequences that are obtained from the observations. Let the input observations be $o_1, o_2, o_3, ...o_n$as n represents the number of observations. High dimensional pose vector, $P_i = L_1, L_2, L_3, L_4, L_5$ where L_i represents the joint locations of each part of an image frame.$Pi \in \mathbb{R}^2$ where $\mathbb{R}^2$ represents the 2d space. $P_i \in p$ where p represents the pose of the body.$P(p|z)$ represents the inference framework to estimate the posterior probability

where p represents the pose and z represents the feature set. The desired posterior probability has been calculated using likelihood and prior. $P(p|z) \propto p(z|p)p(p)$. Maximum a posteriori solution can give a high likelihood. In the layer 1, the pose of the person in the particular time interval has been identified and the output has been given as input to the next level.

$$p_{map} = argmax(p(p|z)) \tag{1}$$

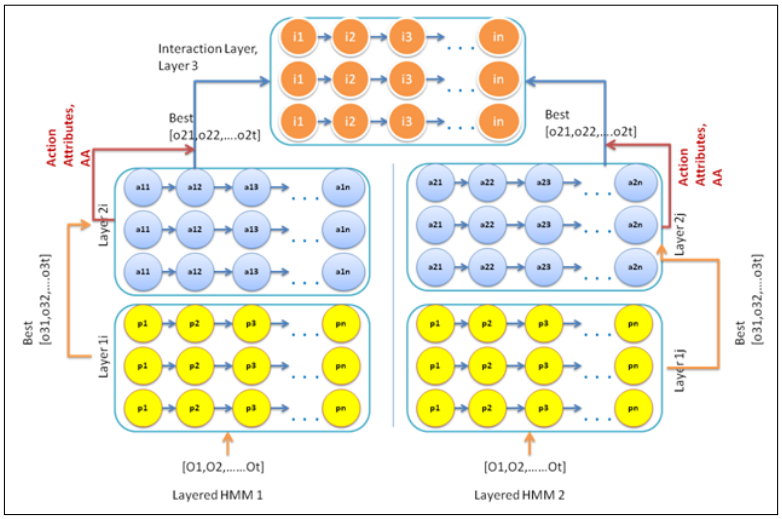


Figure 2: HMM model for HIR system

6.2 Action layer (mid level layer)

Action is defined as the stream of successive pose that happened in the particular time interval. Actions may be defined as the event that takes place for a single person.

$$Action, A = p_1, p_2, p_3, ...p_N \tag{2}$$

The Action of the individual person includes pose and action class labels. ie.,$Action, A \in P, A_i(t)$ Maximizing the log likelihood probabilities, the inferential result has been calculated and the actions such as walking, running, jogging, loiter and get hurt has been identified. After training, the likelihood of the action class in the layer 2i and layer 2j is calculated separately. Let $a_t = a_1^t, a_2^t, ...a_{pn}^t \in \mathbb{R}^{pn}$ denotes the pose vector in a continuous space of dimension equal to the number of individual actions.This layered approach directly outputs the probability p_k^t for each individual action model M_k Where $k = p_1, p_2, p_3, ...p_N$ as input to action HMM where $a_k^t = p_k^t$ for all k. To calculate the probability of model M_k for the given sequence x_1^t is computed in the following manner: $x_1^t = x_1, x_2, ...x_t$ represents the sequence of model.

Let us define forward variable $\alpha(i,t) = P(x_1^t, q_t = i)$ which is the probability of having generated the sequence x_1^t being in state i at time t. In asynchronous HMM,$\alpha(i,t)$ can be replaced by a corresponding factor ([9], [24]). Assume$\sum_{j=1}^{NS} p(q_t = j) = 1$ where NS is the number of states for all models. Probability $p(q_t = i|x_1^t)$ of state i is given as

$$\begin{aligned} p(q_t = i|x_1^t) = \frac{p(q_t = i|x_1^t)}{p(x_1^t)} &= \frac{p(q_t = i|x_1^t)}{p(q_t = j|x_1^t)} \\ &= \frac{\alpha(i,t)}{\sum_{j=1}^{NS} \alpha(j,t)} \end{aligned} \tag{3}$$

From this, the probability of model M_k can be computed as

$$p_{k^t} = \sum_{i \in M_k} (p(q_t = i|x_1^t)) = \frac{\sum_{i \in M_k} \alpha(i,t)}{\sum_{j=1}^{NS} \alpha(j,t)} \tag{4}$$

Where i is the state of the model M_k, i ∈ states of all models and NS denotes the total number of the states. In this work, individual action recognition vectors with the action attributes as observations given to the interaction level HMM.

Attribute selection criteria

Human interactions can be recognized with the help of Action Attributes (AA). The purpose of the attributes is to provide high level knowledge about actions.

$$Action Attribute set, AA = a_1, a_2, a_3, a_4, a_5 \tag{5}$$

Attributes have been manually labeled where low level features are given a class label. Here, in this work, five attributes have been manually labeled. Stretching arm, withdrawing arm, stretching legs, withdrawing leg, hand contact and body contact are defined as the attributes. The presence or absence of each attribute is approximated by the confidence value (0 or 1). Attribute classifiers are learned from training data sets. In the multi-level modeling approach [25] , the Knowledge of actions and attributes gives interaction in a more accurate way.

6.3 Interaction layer (high level layer)

Interaction is defined as successive actions between two persons that are integrated together. I represent the possible interactions between all possible co-existing pairs A1 and A2 where A1 and A2 denote the action of the person 1 and 2 respectively. A1 and A2 are coupled together to identify the new interaction

$$I(1,2) = [((A1 + AA)(t1).....(A1 + AA)(tz))U((A2 + AA)(t1).....(A2 + AA)(tz))] \tag{6}$$

where Iij represents the interactions between 2i layer, 2j layers respectively, and t1,t2...tz represents the time frames from 1 to z. The Interaction layer (I-HMM) is defined as the third layer in which the observations have been done using log-likelihood of the action layer and its inferential results. The interactions such as handshaking, pushing, hugging, fighting and meeting have been recognized. The knowledge of both the layers has been coupled to recognize the interaction between two or more persons. This layer uses spatio-temporal constraints as features. A CHMM model (λ) is defined by the following parameters. [17]

$$\pi_{o^C}(i) = P(q_1^c = S_i) \tag{7}$$

$$a_{i|j,k}^C = P(q_t^C = S_i | q_{t-1}^{A1} = S_j, q_{t-1}^{A2} = S_k) \tag{8}$$

$$b_t^C(i) = P(O_t^C || q_t^C = S_i) \tag{9}$$

Where $C \in (Action1, Action2)$and q_t^C Represent the state of coupling nodes in the C^{th} stream at time T. In Coupled HMM, the output observed from layer 2i and layer 2j are given as inputs. The observed sequence has been given as, $O = A_{1T}, A_{2T}$ Where $A_{1T} = a_{11}, a_{12}, a_{13}, a_{1T}$ are the observations of the first person and $A_{2T} = a_{21}, a_{22}, a_{23}, a_{2T}$ are the observed sequence of second person. The observation a_{11} consists of A1 + AA. Here, the observation sequence consists of actions of each person (A) and action attribute (AA) of each person. The state sequence

has been given us, $S = X_1^T, X_2^T$ where $X_1^T = X_{11}, X_{12}, X_{13}, ...X_{1T}$ $X_1 \in 1, ...M$ are the state sequence of first observations and $X_2^T = X_{21}, X_{22}, X_{23}, ...X_{2T}$ $X_2 \in 1, ...M$are the state sequence of second observations. State Transition probabilities of the first chain of observations have been represented as, $P(X1_{t+1}|X1_t, X2_t)$ and for the second chain as P $(X2_{t+1}|X1_t, X2_t)$. P (X1) and P (X2) are the prior probabilities of first and second chain respectively. $P(A1_t|X1_t)$ and $P(A2_t|X1_t)$ are the observation densities assumed to be multivariate Gaussian with mean vectors μ_x , μ_y and covariance matrices Σx,Σy. Expectation Maximization (EM) Algorithm finds the maximum likelihood that estimates the model parameters by maximizing the following function (10). Parameter λ , contains parameters of transition probability, prior probability, and parameters of observation densities.

$$\begin{aligned} M(\lambda) = P(X1)P(X2)\pi_{t-1}^{N}P(A1_t|X1_t)P(A2_t|X2_t)P(X1_{t+1}|X1_t, X2_t) \\ P(X2_{t+1}|X1_t, X2_t), 1 \leq t \leq N \end{aligned} \tag{10}$$

7 Concurrent interaction recognition

The interactions that happened between groups of people simultaneously at a particular time interval are defined as concurrent interactions. Example of concurrent interaction is hugging and handshaking in a single scenario. Here, in this work four persons have been considered and concurrent interactions between them have been identified.

$$CI(1, 2, 3, 4) = I(1, 2) + I(3, 4) \tag{11}$$

The training of CHMM differs from standard HMM in the expectation step (E) while they are both identical in the maximization step (M) which tries to maximize the equation (10).The expectation step of CHMM is defined in terms of forward and backward recursion. For the forward recursion, we define a variable for both observation chains at t=1,

$$\alpha_{t=1}^{person1(A1)} = P(A1|X1)P(X1) \tag{12}$$

$$\alpha_{t=1}^{person2(A2)} = P(A2|X2)P(X2) \tag{13}$$

Then the variable α is calculated incrementally at any arbitrary moment t as follows.

$$\begin{aligned} \alpha_{t+1}^{person1(A1)} = P(A1_{t+1}|X1_{t+1}) \int\int (\alpha_t^{person1(A1)})(\alpha_t^{person2(A2)} \\ P(X1_{t+1}|X1_t, X2_t), dX1_t dX2_t \end{aligned} \tag{14}$$

$$\begin{aligned} \alpha_{t+1}^{person2(A2)} = P(A2_{t+1}|X2_{t+1}) \int\int (\alpha_t^{person1(A1)})(\alpha_t^{person2(A2)} \\ P(X2_{t+1}|X1_t, X2_t), dX1_t dX2_t \end{aligned} \tag{15}$$

In the backward direction, there is no split in the calculated recursions which can be expressed as:

$$\begin{aligned} \beta_{t+1}^{person1(A1),person2(A2)} = P(O_{t+1}^N|S_t) = \int\int P(A1_{t+1}^N, A2_{t+1}^N|X1_{t+1}, X2_{t+1}) \\ P(X1_{t+1}, X2_{t+1}|X1_t, X2_t)dX1_{t+1}dX2_{t+1} \end{aligned} \tag{16}$$

After combining both forward and backward recursion parameters, an interaction will be tested on the trained model, generating the equivalent interaction that most likely fit the model. The generated interaction sequence is determined when there is a change in the likelihood.

8 Results and discussion

8.1 Datasets and experimental setup

The proposed system has been implemented in MATLAB version13a, with no special requirements in hardware. The proposed work has been analyzed and evaluated using four video datasets. They are UT-Interaction dataset, BEHAVE dataset, Self generated dataset and KTH dataset. The UT-Interaction dataset consists of different two person interaction patterns like shake hands, hug, point, kick, punch and push. The BEHAVE dataset consist of the outdoor environment and it consists of activities like group formation, crossing each other, depart, approach, move closer, move farther, etc. The generated dataset consists of single person activities and interactions that happen between two persons are taken under indoor environment. The activities covered in this dataset are walking, running, jogging, loitering and getting hurt. In Kth dataset, the walking, jogging and running scenarios have been taken for evaluation. In Experiment 1, The UT-Interaction dataset [12] contains videos of continuous executions of 6 classes of human-human interactions: shake-hands, point, hug, push, kick and punch. There are a total of 20 video sequences whose lengths are around 1 minute. The videos are taken with the resolution of 720*480 and 30fps. High level interactions have been recognized using four participants. Results from this dataset has been shown in figure 3,4,5.

Figure 3: Activity recognition - approach and group formation, Departing, Collide and Divert

Figure 4: Interaction recognition (Kick, Hug and Push)

Figure 5: Concurrent interaction recognition (Hug and Punching, Pushing and Approaching, Handshaking and Pushing)

In Experiment 2, training and testing has been carried out using Behave dataset [18] comprises of two views of various scenario's of people acting out various interactions. The data is captured

at 25 frames per second. The resolution is 640 * 480. The following ten interactions have been recognized in the Behave dataset. In Group, Approach, Walk Together, Meet, Split, Ignore, Chase, Fight, Run Together and Following are the interactions. The results from this dataset have been shown in figure 6.

Figure 6: Concurrent interaction recognition
(in group and crossing each other, walk together and walking)

8.2 Performance analysis

In this research, metrics such as Accuracy, Precision, Recall, Positives such as True Positive (TP), False Positive (FP) and Negatives such as True Negative (TN), and False Negative (FN) have been used to measure the performance of the system. The Performance metrics have been calculated for the poses, activity and interactions.

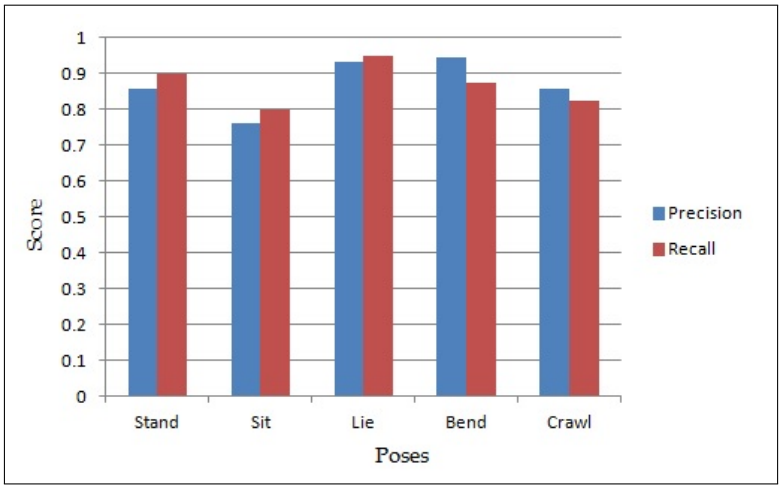


Figure 7: Pose recognition rate

Precision and recall values for each pose such as stand, sit, lie, bend and crawl have been shown in figure 7. Stand and sit poses has more precision and recall values. Lie, bend and crawl poses has more precision values than recall values. Figure 8 shows the precision- recall curve for activity recognition rate. The performance for the activities such as walk, run, jog, loiter and get hurt has been shown. Walk, run and loiter activities have high recall value than precision value. The activities walk, run and loiter are slightly confusing due to temporal difference. The other activities such as jog and get hurt have high precision values.

Figure 9 show the precision-recall curve for interaction recognition rate. The interactions such as handshaking, hugging, kicking, punching and pushing have been predicted. Handshaking, punching and pushing interactions have high precision values. Hugging and kicking have low precision values than other interactions. The recognition rate of poses, activity, interaction

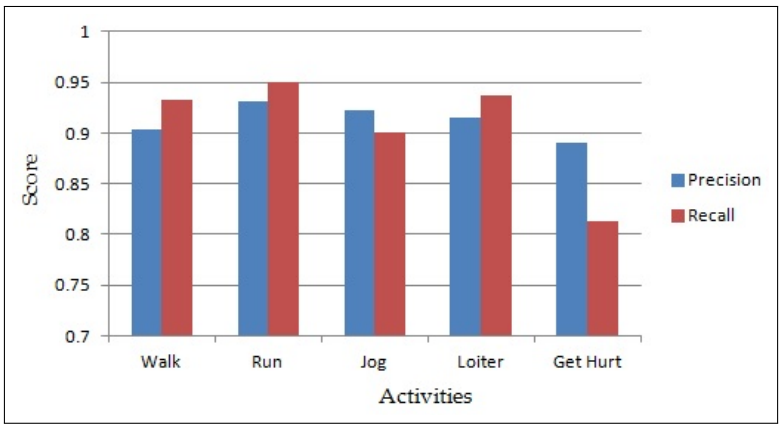


Figure 8: Activity recognition rate

and concurrent interaction in different datasets has been shown in table 1. These concurrent interactions have been identified using the proposed learning model. Single person action has been recognized from Kth dataset and recognition rate has been obtained as 87.62%.

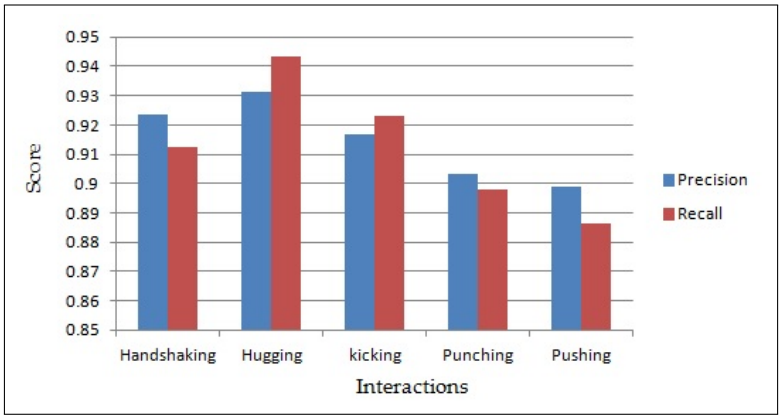


Figure 9: Interaction recognition rate

Table 2 shows the comparison of the previous works that are carried out in this activity recognition. In Learning Methodologies, the joint form of LHMM and CHMM outperforms other learning models. The spatial relations and the levels of temporal granularity have been considered. Usually, the HMM can handle temporal variations in the video. In this paper, the interaction between neighboring states also been considered. CHMM has the capacity to underlying synchronization of two different processes. The two actions have been coupled using coupling probabilities with proper weights and the concurrent interactions have been recognized.

Table 1. Recognition Rate across Datasets (in terms of Accuracy)

Datasets	Poses	Single - person Action	Two- person Interaction	Concurrent Interaction
Kth	87.62	-	-	-
UT-interaction	-	-	83.90	81.54
BEHAVE	-	-	-	71.56
Self-generated	91.55	93.47	84.23	88.36

Table 2. Comparison of Activities recognized with previous works

Previous Works	Learning Model Proposed	Concentrated on	Activities Recognized	Recog. Rate(%)
Sunyoung Cho et al.,2013 [26]	Visual and textual information as features,Graphical model using structured learning with latent variables	Activity Recognition	High five, handshake, hug and kiss.	78.4
M.J.Marin Jimenez et al.,2013 [27]	Dictionary learning, support vector machines with φ^2 Kernel.	Activity recognition	Hug, kiss and hand shake.	78
Hejin Yuan et al.,2015 [28]	Semi supervised learning-k-means clustering, Skeleton features. (Cumulative Skeleton Image(CSI), Silhouette History Image(SHI))	Activity recognition - single person - basic actions	Weizmann dataset (10 basic actions) Bend, jump, jump, jack, wave, wave1, wave 2, side, walk, run.	90
Fadime Sener et al .,2015 [4]	Multiple instance learning process. Shape and motion features	Two person Interactions.	UT Interactions dataset and TV interaction Dataset	75.60
Proposed System	Three layer HMM along with coupled HMM	Concurrent Interaction with four persons - complex activities	In Group, walking together, chase, fight, following, handshaking and hugging.	88.36

8.3 Discussion

Our interaction recognition system is based on Hierarchical Hidden Markov Model (HHMM) which combines layered and coupled HMM that tries to find the concurrent interactions. The three layers such as Pose layer, action layer and interaction layer has been modeled. In order to identify the interaction, action of the two persons has been modeled independently. As a preprocessing step, we have been deployed body part modeling and location based trajectory tracking to aid the localization of the people in frames. The action sequence is composed of parallel states presenting the poses and each pose is composed of the specific number of observations (M= 5 in our case). The action sequence of person 1 and 2 has been modeled individually in the layered fashion. Interaction Sequence is composed of the action sequence of person 1 and the action sequence of person 2. (M=10). After training LHMM, the observation sequence from the databases, the system tries to find a corresponding sequence of poses based on the learning during the training phase. The generated pose sequence is the sequence that achieves the maximum likelihood estimation with the poses. Thus the observed pose is given as input

to the next layer HMM. In the second layer HMM, the system tries to find the corresponding action sequences. The action of person1 is identified from the maximum likelihood estimation of the action sequences. In the same way, the action of person 2 also identified using two layered HMM. Coupled HMM (CHMM) couples both the action sequence of person1 and person2 and the system tries to find the interaction in a better way. CHMM in lag1 condition can couples the observation channels. Each channel has its own action sequence. From both the action sequences the next state emission probability has been generated. Based on all previous works, the specific activity has been recognized using the specific learning model. HMM is the most successful framework in speech and video applications and it is well suited for computing with uncertainties. Here, in this work to demonstrate the concurrent interactions, the HMM learning model has been extended in the joint form of layered HMM and coupled HMM. Layered HMM models the non-causal symmetric influences and CHMM to model the temporal and asymmetric conditional probabilities between observation chains.

Conclusion

In this work, a hybrid learning framework is designed to recognize the concurrent interactions between multiple peoples. The spatial and temporal information and body part attributes are considered as features. The poses and actions are recognized in a layered fashion. The actions of multiple persons are coupled to recognize the interactions and concurrent interactions. The joint form of LHMM and CHMM has been used for providing concurrency. The interactions between neighboring persons has also been recognized. Here, the activity recognition has been done for the continuous events and this could be extended in future to discrete event recognition mechanisms also. Further work will focus on identifying interactions and behavior in different person to person interaction contexts that will allow the system to recognize the interactions under different conditions.This system can act as a smart surveillance to recognize the actions/interactions of multiple people without human intervention in the environments such as meeting hall, discussion groups, public places, banking sectors, where multiple people could interact with each other.

Acknowledgment

The work reported in this paper has been supported by Anna University,Chennai by providing Anna Centenary Research Fellowship. We also acknowledge the anonymous reviewers for comments that lead to clarification of the paper.

Bibliography

[1] Alexandros Andre Chaaraoui, Pau Climent-Perez, Francisco Florez-Revuelta(2013); Silhouette-based human action recognition using sequences of key poses, *Pattern Recognition Letters*, 34(15): 1799-1807.

[2] Aggarwal, J. K.and Ryoo, M. S. (2011); Human activity analysis: A review, *ACM Computing Survey*, 43(3): 16:1–16:43.

[3] Arnold Wiliem,Vamsi Madasu, Wageeh Boles and Prasad Yarlagadda (2012); A suspicious behaviour detection using a context space model for smart surveillance systems, *computer vision and Image Understanding*, 116(2): 194-209.

[4] Fadime sener and Nazli Ikizler-cinbis (2015); Two Person Interaction Recognition via spatial Multiple Instance Embedding, *Journal of Visual Communication and Image Representation*, 32: 63-73.

[5] Gowsikhaa.D, Abirami.S and Baskaran.R. (2012); Automated human behavior analysis from surveillance videos: a survey, *Artificial Intelligence Review* , DOI 10.1007/s10462-012-9341-3, 1-19.

[6] Gowsikhaa.D, Manjunath and Abirami S. (2012); Suspicious Human activity detection from Surveillance videos, *International Journal on Internet and Distributed Computing Systems*, 2(2): 141-149.

[7] Junji Yamato, Jun Ohya and Kenichiro Ishii (1992); Recognizing Human Action in Time-Sequential Images using Hidden Markov Model, *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, doi:10.1109/cvpr.1992.223161, 379-385.

[8] Matthew Brand, Nuria Oliver, and Alex Pentland (1997); Coupled Hidden Markov Models for Complex Activity Recognition, *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, DOI: 10.1109/CVPR.1997.609450, 994 - 999.

[9] Nuria Oliver, Ashutosh Garg and Eric Horvitz (2004); Layered Representations for learning and inferring office activity from multiple sensor channels,*Computer Vision and Image Understanding*, 96: 163-180.

[10] Roberto Melfi, Shripad Kondra and Alfredo Petrosino (2013); Human activity modeling by spatio temporal textural appearance, *Pattern Recognition Letters*, 34(15): 1990-1994.

[11] Ryoo M.S. (2011); Human Activity Prediction: Early Recognition of Ongoing Activities from Streaming Videos, *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, DOI: 10.1109/ICCV.2011.6126349, 1036-1043.

[12] Ryoo, M.S, and Aggarwal, J.K. (2010); UT Interaction Dataset, *Proc. of ICPR Contest on Semantic Description of Human activities.*

[13] Sangho Park and J.K. Aggarwal (2004); Semantic-level Understanding of Human Actions and Interactions using Event Hierarchy, *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshop*, DOI:10.1109/CVPR.2004.160, 1-12.

[14] Sang Min Yoon , Arjan Kuijper (2013); Human action recognition based on skeleton splitting, *Expert systems with Applications*, DOI:10.1016/j.eswa.2013.06.024, 40(17): 6848-6855.

[15] Sivarathinabala M. and Abirami S. (2014); Motion Tracking of Humans under Occlusion using Blobs, *Proceedings of Advanced Computing, Networking and Informatics- Volume 1, Smart Innovation, Systems and Technologies*, 27: 251-258.

[16] Shih-Kuan Liao, Baug-Yu Liu,(2010); An edge-based approach to improve optical flow algorithm, *Proceedings of Third International Conference on Advanced Computer Theory and Engineering*, 6: 45-61.

[17] Shizhong and Joydeep Ghosh (2001); A New formulation of Coupled Hidden Markov Models, doi=10.1.1.607.5700rep=rep1type=pdf.

[18] S. J. Blunsden and R. B. Fisher (2010); The BEHAVE video dataset: ground truthed video for multi-person behavior classification, *Annals of the BMVA*, 4: 1-12.

[19] Teddy Ko (2010); A Survey on Behavior Analysis in Video Surveillance Applications. *Proceedings of IEEE, Applied Imagery Pattern Recognition Workshop*, 1-8.

[20] Thomas Brox, Bodo Rosenhahn, Juergen Gall, and Daniel Cremers (2010); Combined Region and Motion-Based 3D Tracking of Rigid and Articulated Objects, *IEEE Transactions on Pattern Analysis And Machine Intelligence*, 32(3): 402-415.

[21] Weilun Lao, Jungong Han, and Peter H. N. deWith (2010); Flexible Human Behavior Analysis Framework for Video Surveillance Applications. *International Journal of Digital Multimedia Broadcasting*,ID: 920121, 1-9.

[22] Weiyao Lin, Ming-Ting Sun, Radha Poovendran and Zhengyou Zhang (2010); Group Event Detection with a Varying Number of Group Members for Video Surveillance, *IEEE Transactions on Circuits and Systems for Video Technology*, 20(8): 1503.00082.

[23] Weiming Hu, Guodong Tian , Xi Li , Stephen Maybank (2013); An Improved Hierarchical Dirichlet Process-Hidden Markov Model and Its Application to Trajectory Modeling and Retrieval, *Int J Comput Vis*, DOI 10.1007/s11263-013-0638-8, 105:246-268.

[24] Dong Zhang, Daniel Gatica-Perez, Samy Bengio, Iain McCowan, and Guillaume Lathoud (2004); Modeling Individual and Group Actions in Meetings: a Two-Layer HMM Framework, *the Second IEEE Workshop on Event Mining: Detection and Recognition of Events in Video, In Association with CVPR*, 1-8.

[25] Gildas Morvan, Daniel Dupont,Jean-Baptiste Soyez, Rochdi Merzouki (2012); Engineering hierarchical complex systems: an agent-based approach, The case of flexible manufacturing systems, *Chapter - Service Orientation in Holonic and Multi-Agent Manufacturing Control, series Studies in Computational Intelligence*, 402: 49-60.

[26] Cho, Sunyoung and Kwak, Sooyeong and Byun, Hyeran (2013); Recognizing Human-human Interaction Activities Using Visual and Textual Information, *Pattern Recogn. Lett.*, 34(15):1840-1848.

[27] Manuel J. Marin-Jimenez, Enrique Yeguas, Nicolas Perez de la Blanca (2013); Exploring STIP-based models for recognizing human interactions in TV videos, *Pattern Recognition Letters*, 34: 1819 -1828.

[28] Hejin Yuan (2015); A Semi-supervised Human Action Recognition Algorithm Based on Skeleton Feature, *Journal of Information Hiding and Multimedia Signal Processing*, 6(1): 175-181.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(4):567-579, August 2016.

An Abnormal Network Traffic Detection Algorithm Based on Big Data Analysis

H.P. Yao, Y.Q. Liu, C. Fang

Haipeng Yao*
1. State Key Laboratory of Networking and Switching Technology
Beijing University of Posts and Telecommunications
No 10, Xitucheng Road, Haidian District, Beijing, PRC
2. Beijing Advanced Innovation Center for Future Internet Technology
Beijing University of Technology
100 Ping Le Yuan, Chaoyang District, Beijing, PRC
*Corresponding author: yaohaipeng@bupt.edu.cn

Yiqing Liu
State Key Laboratory of Networking and Switching Technology
Beijing University of Posts and Telecommunications
No 10, Xitucheng Road, Haidian District, Beijing, PRC
colin617@qq.com

Chao Fang
1. Beijing Advanced Innovation Center for Future Internet Technology
Beijing University of Technology
100 Ping Le Yuan, Chaoyang District, Beijing, PRC
fangchao.bupt@gmail.com
2. College of Electronic Information and Control Engineering
Beijing University of Technology
100 Ping Le Yuan, Chaoyang District, Beijing, PRC
fangchao.bupt@gmail.com

Abstract: Anomaly network detection is a very important way to analyze and detect malicious behavior in network. How to effectively detect anomaly network flow under the pressure of big data is a very important area, which has attracted more and more researchers' attention. In this paper, we propose a new model based on big data analysis, which can avoid the influence brought by adjustment of network traffic distribution, increase detection accuracy and reduce the false negative rate. Simulation results reveal that, compared with k-means, decision tree and random forest algorithms, the proposed model has a much better performance, which can achieve a detection rate of 95.4% on normal data, 98.6% on DoS attack, 93.9% on Probe attack, 56.1% on U2R attack, and 77.2% on R2L attack.
Keywords: Anomaly Traffic Detection, Big Data, K-means, Decision Tree, Random Forest.

1 Introduction

With the improvement of network, storage, calculation and transmission,the Internet is interacting more closely with people than ever before. While the Internet is making our life more convenient, it brings about some potential risks. For example, malicious attacks involving user privacy and security become more and more frequent.

The changes of how people use the Internet is a new challenge for traditional abnormal network event detection techniques. It is more hard for researchers to get aware of some new kinds of attacks. To resolve these problems, some abnormal network traffic detection methods

have been proposed. Traditional abnormal traffic detection method can be classified into two categories [1–3]. One is misuse detection, and the other is abnormal detection. The two methods have their own pros and cons. Misuse detection has a high accuracy but needs support from known knowledge. Abnormal detection do not need known knowledge, but cannot categorize the type of attacks, the accuracy is also lower. For example, Hari Om [4] designes a hybrid detection system, which is a hybrid anomaly detection system considering k-means, k-nearest neighbor and Naïve Bayes methods.

However, the explosive increase of network traffic has directly or indirectly pushed the Internet into the big data era, which makes anomaly traffic detection more difficult to deal with because of high calculation volume and constant changes of network data distribution caused by big data [5–8]. Because the speed of network data generation is fast, it makes the volume of normal traffic and abnormal traffic differ a lot, and the distribution of the data change. Besides, with big data, the difference between normal traffic and abnormal traffic is increasing. It makes the traditional methods unable to effectively detect abnormal traffic.

Therefore, to increase the accuracy of abnormal traffic and avoid the loose caused by false negative detection, we propose a novel model based on big data analytics, which can avoid the influence brought by adjustment of network traffic distribution, increase dectection accuracy and reduce the false negative rate. The core of the proposed model is not simply combination of traditional detection methods, but a novel detection model based on big data. In the simulation, we use k-means, decision tree and random forest algorithms as comparative objects to vertify the effectiveness of our model. Simulation results reveal that the proposed model has a much better performance, which can achieve a detection rate of 95.4% on normal data, 98.6% on DoS attack, 93.9% on Probe attack, 56.1% on U2R attack, and 77.2% on R2L attack.

The rest of this paper is organized as follows. In Section 2, related work of this paper is presented. The system model is given in Section 3. Simulation results are presented and discussed in Section 4. Finally, we conclude this study in Section 4.3.

2 Related work

2.1 k-means

k-means is a classic clustering algorithm [9,10], which uses simple iteration algorithm to cluster the data set into certain amount of categories.Commonly, the number of clusters is annotated to be K. The four steps of k-means are:

1. Initialization: Randomly select K data points from the data set as the centers of the K-clusters;
2. Distribution: Assign each point in the data set to the nearest center;
3. Update: calculate new centers according to the cluster assignment, the new center is the average point of all the points in a cluster;
4. Repeat: Repeat these steps until no center is updated in this round, and the clustering is converged.

k-means needs the number of classification K to be specified. If K is not chosen properly, it will lead to an improper result of classification. So choose a proper cluster number is crucial to the result of k-means.

Another disadvantage of k-means is that, k-means can only use Euclidean distance. Even though Euclidean distance is convenient to calculate, but it cannot take the difference between two features into consideration, it means it treats all features as same. In the reality, it will sometimes lead to poor performance.

Anyway, k-means has its own advantages when dealing with big data.

1. k-means is simple. The time complexity is $n\left(n^{d*k+1}\log n\right)$, it can be fast when the number of clusters and the number of features are small;
2. k-means can be well adjusted to big data set and has high performance.

2.2 Decision tree

Decision Tree [9] is a common algorithm used in machine learning. A complete decision tree is composed by three kind of elements:
1. Decision Node, indicating which feature is used in split;
2. Chance nodes, indicating possible values of each features;
3. Leaf node, indicating which category is the record in.

There are two steps needed to use a decision tree:
1. Tree generation: Generate a tree according to training set. Need to determine which feature need to use in the split, and determine which category the result is in.
2. Classification: Classify new records from the root of the decision tree, and compare the record with each of the decision node, move to corresponding branch with the result. Repeat this process, and after a data reaches the leaf node, the category of leaf node is the new category of the node.

Quinlan proposed C4.5 algorithm in [11], which is a well known decision tree algorithm.The main method is to generate the decision tree from root to leaf, in order to reduce the level of uncertainty. Therefore, this algorithm can be described as follows.

Gain ratio is the index C4.5 used to select feature. Define a feature in the feature set to be A_k, the training set to be Tď ż˝ď ż˝definition of information gain is defined like this:

$$Gain(T, A_k) = Info(T) - Info_{A_k}(T) \tag{1}$$

where

$$Info(T) = -\sum_{i=1}^{n} \frac{freq(c_i, T)}{|T|} \log_2 \frac{freq(c_i, T)}{|T|} \tag{2}$$

$$Info_{A_k}(T) = -\sum_{a_k \in D(A_k)} \frac{|T_{a_k}^{A_k}|}{|T|} Info(T_{a_k}^{A_k}) \tag{3}$$

$freq(c_i, T)$ means the number of records belongs to c_i in T. $T_{a_k}^{A_k}$ express that subset which A_k is a_k, and domain of A_k is $D(A_k)$.

$SplitInfo(A_k)$ is defined to be:

$$SplitInfo(T, A_k) = -\sum_{a_k \in D(A_k)} \frac{|T_{a_k}^{A_k}|}{|T|} \log_2 \frac{|T_{a_k}^{A_k}|}{|T|} \tag{4}$$

Gain ratio is

$$Gainratio(T, A_k) = \frac{Gain(T, A_k)}{SplitInfo(A_k)} \tag{5}$$

The advantages of decision tree are:
1. The tree generated is easy to generate and easy to explain;
2. Performs well when dealing with large data set.

2.3 Random forest

Random Forest algorithm [9, 12] is a classification algorithm and contains multiple decision trees, where each tree has a vote, and result is the one with highest vote.

When generating decision tree, feature selection and pruning can be used to avoid over fitting. But when the number of features is large, the problems can hardly be avoided. Random forest consists of multiple decision trees, which can effectively avoid those problems.

Random forest has following advantages:

1. It can be used in various situation with a pretty high accuracy on classification;
2. It can effectively support multi-feature situation without feature selection;
3. It can report the importance distribution of features.

3 System model

Influenced by big data, network data distribution is gradually changing. This paper try to solve the problem that caused by the increasing difference between normal traffic and abnormal traffic. Therefore, we proposed a new abnormal traffic detection model based on big data analysis, and this model includes three sub-models.

3.1 Normal traffic selection model

Normal traffic selection model uses classification and clustering algorithm to distinguish normal and anomaly behaviors, rather than involved specific anomaly behaviors. This model includes two stages:

1. Training stage: training model uses data that labeled normal or abnormal, and the model applies in test stage.

2. Test stage: test stage is similar to detection in practice. Using unlabeled date, the model classifies traffic data into normal or abnormal, and labels them.

Normal traffic selection model uses k-means clustering algorithm, KNN, decision tree and random forest classification algorithms. Traditionally, before using k-means algorithms, it is very important to set the number of categories, because we don't know how many categories. But in order to distinguish normal and abnormal behavior, the normal traffic selection model uses k-means as following way.

In training stage, using labeled information classify data into normal and abnormal. These two categories use k-means separately instead of clustering all data at once, getting the center of the data set respectively. Then using the center of the data set, KNN clustering algorithm classifies test data. Decision tree and random forest classification algorithms train with labeled normal and abnormal data.

3.2 Abnormal traffic selection model

The purpose of abnormal traffic selection model is avoid influence caused by too many normal traffic than abnormal traffic. This model classifies anomaly traffic into specific categories, and includes two stage as well:

1. Training stage: this stage only use abnormal data to train classification model, and every data label specific attack group. Using classification algorithms learns classified rules.

2. Test stage: test stage is similar to detection in practice, using unlabeled data (including normal behavior data). The classification model classifies anomaly traffic into specific categories according to the classified rules, and gives specific label to every data.

Table 1: Distribution of KDDCUP99 data set

Data set	Normal	DoS	Probe	R2L	U2R
10 percent of training data set	97278	391458	4107	1126	52
test data set	60593	229853	4166	16189	228

Abnormal traffic selection model uses decision tree and random forest classification algorithms. Abnormal traffic selection model and normal traffic selection model are independent, without order of priority in training stage or test stage.

Mixed compensation model combines the result from normal traffic selection model and abnormal traffic selection model to produce a final result. Although abnormal traffic selection model is more effective because without influence of normal traffic data, the model has high false negative rate due to this characteristic. Therefore use normal set produced by normal traffic selection model to compensate abnormal set $A = \{A_1, A_2, \cdots, A_k\}$ produced by abnormal traffic selection model. $A_i, i \in [1, k]$ denote specific attack category. If c denote detection result, rule of compensation as follow:

$$\begin{cases} \text{if } c \in A_i,\ c \in N, \text{ then } c \in N \\ \text{if } c \in A_i,\ c \notin N, \text{ then } c \in A_i. \end{cases} \tag{6}$$

4 Simulation results and discussions

Before using three sub-models of anomaly detection based on big data analysis, data set needs be preprocessed with label for training model. It should be noted that rightly selecting feature is a good way to reduce dimension and increase efficiency of running. In the simulation, three different algorithms are used to verify validity of the proposed model.

4.1 Data set

In the simulation, we use KDDCUP99 [13] data set to test my model. KDDCUP99 data set is widespread use for testing abnormal detection model, which is obtained and processed from KDDCUP99 [14]. KDDCUP99 data set has 41 features and been sorted into three group: basic feature, content feature and time feature [15].

The distribution of data set is shown as Table 1, where training data has 5 million records, 10 percent of training data has 494021 records, and test data has 311029 records. Every record is labeled to be normal or abnormal, and abnormal data can be classified into four groups: Dos, U2R, R2L and Probe. From Table 1, we find that normal data in training data set is more than abnormal data in test data set. Therefore, this data set can be used to test the performance of the proposed model under different circumstances.

4.2 Simulation results

As shown in Table 2, we have done eight experiments with the model based on big data analysis, and three control experiments which used k-means, decision tree or random forest respectively. In the control groups, training classify model uses all training data set with five categories, then classifying test data into five categories. Another control group is winner of KDDCUP99.

In the simulation, prediction accuracy is used as a simulation metric of detection effect, which is shown in Table 3. Besides, we adopt way of sorting and grading for every type. For example,

Table 2: Number of experiments

No.	Normal traffic selection model	Abnormal traffic selection model	No. of control group	Algorithm
1	k-means1*	Random Forest	9	k-means
2	k-means1*	Decision Tree	10	Decision Tree
3	k-means2*	Random Forest	11	Random Forest
4	k-means2*	Decision Tree	12	Winner of KDDCUP99
5	Decision Tree	Decision Tree		
6	Decision Tree	Random Forest		
7	Random Forest	Decision Tree		
8	Random Forest	Random Forest		

*note: In the normal traffic selection model, the number of cluster of normal and abnormal respectively is 4 and 30 in $k-means1$, and the number of cluster of normal and abnormal respectively is 100 and 300 in $k-means2$.

Table 3: Prediction accuracy

No.	Experiment	Normal	DoS	Probe	U2R	R2L
1	k-means1+Random Forest	0.632	0.814	0.939	0.561	0.679
2	k-means1+Decision Tree	0.656	0.791	0.878	0.500	0.772
3	k-means2+Random Forest	0.945	0.983	0.910	0.513	0.510
4	k-means2+Decision Tree	0.946	0.979	0.852	0.500	0.504
5	Decision Tree+Decision Tree	0.951	0.984	0.829	0.500	0.512
6	Decision Tree + Random Forest	0.951	0.986	0.831	0.550	0.517
7	Random Forest + Decision Tree	0.954	0.980	0.861	0.500	0.521
8	Random Forest + Random Forest	0.952	0.985	0.872	0.520	0.510
9	k-means	0.938	0.968	0.785	0.500	0.528
10	Decision Tree	0.951	0.983	0.793	0.500	0.500
11	Random Forest	0.952	0.985	0.875	0.522	0.507
12	Winner of KDDCUP99	0.995	0.971	0.833	0.132	0.084

all experiments are sorted by prediction accuracy of normal. The first grades 1 point, the second grades 2 points, and so on. Finally, adding grade of five groups is final grade.

As shown in Table 4, the experiment group and winner of KDDCUP99 are sorted by final grade. While the later has high detection rate in normal data, as for four attack types, the result of model based on big data analysis is better than winner of KDDCUP99.

Algorithm of winner of KDDCUP99 is C5 decision tree [16–19]. Training data of winner of KDDCUP99 is a little different with my experiment. Thus for evaluating detection effect of the proposed mode, we did three control experiments with same training data and test data, used with k-means, decision tree or random forest respectively. The number of these experiments is noted as 11, 10 and 9.

Sorting result shows that detection effect of algorithm of the proposed model is better than no use, as shown as Table 5. We will discuss experiments results, compared No.8 with No.11, No.7 with No.5 and No.3 with No.4.

Discussing result of no.8 and no.11

Score of top three are same. Judging No.8 and No.11 with final grade, detection result of two experiments are almost same. And both of them use random forest algorithm. But the difference is:

Table 4: Compared with winner of KDDCUP99

No.	Experiment	Normal	DoS	Probe	U2R	R2L	Final Socre	Rank
8	Random Forest+Random Forest	3	2	2	2	4	13	1
6	Decision Tree+Random Forest	4	1	6	1	2	14	2
7	Random Forest +Decision Tree	2	5	3	4	1	15	3
2	k-means2+Random Forest	7	4	1	3	5	20	4
5	Decision Tree+Decision Tree	4	3	7	6	3	23	5
4	k-means2+Decision Tree	6	6	4	5	6	27	6
12	Winner of KDDCUP99	1	7	5	7	7	27	6

Table 5: Compared with control group

No.	Experiment	Normal	DoS	Probe	U2R	R2L	Final Socre	Rank
6	Decision Tree+Random Forest	4	1	6	1	3	15	1
8	Random Forest + Random Forest	2	2	3	3	5	15	1
11	Random Forest	2	2	2	2	7	15	1
7	Random Forest + Decision Tree	1	7	4	5	2	19	4
3	k-means2+ Random Forest	8	5	1	4	5	23	5
5	Decision Tree + Decision Tree	4	4	7	5	4	24	6
10	Decision Tree	4	5	8	5	9	31	7
9	k-means	9	9	9	5	1	33	8
4	k-means2+ Decision Tree	7	8	5	5	8	33	8

1. Importance of variable used in classifying is different;
2. No.8 has lower false negative rate.

• Importance of variable

As shown as Fig. 1, variables chosen by random forest in No.8 and No.11 are different. Random forest algorithm can output importance of variables, noted Gini index [9]. Fig. 1 shows that top 20 have important variables in comparison with top 1, whose value is higher and more important.

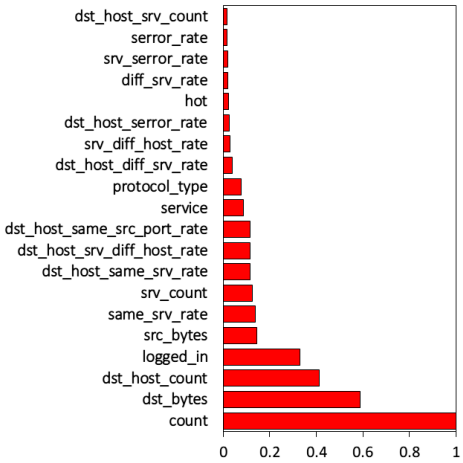
In No.8, rank of variables is different between normal traffic selection model and abnormal traffic selection model. This means that variable used for predicting normal or abnormal and specific attack is different. Therefore, choosing variable in No.11 is influenced by both sides, and output a compromised result when choosing variables, that's why prediction of model in No.11 has deviation.

• Comparison of false negative rate

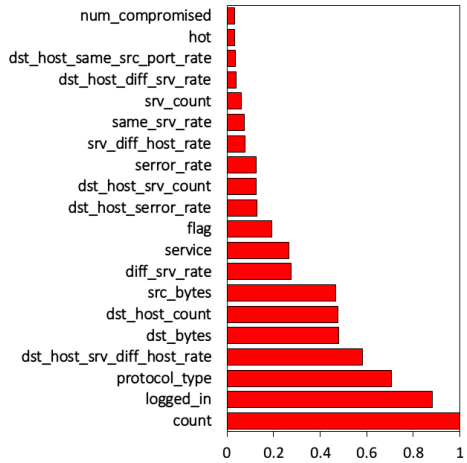
In order to evaluate effect on predicting abnormal behavior, false negative rate is used as an important index, which can measure how many attack events are omitted. Table 6 shows confusion matrix of results of experiments No.8 and No.11 when using random forest. Row express information of prediction, and column express actual information. False negative rate of No.8 in normal type is very low, but high in U2R and R2L type. In No.8, false negative rate of normal selection model in normal is low. Without influence of normal training data, false negative rate of abnormal selection model in four specific attack types are lower than No.11.

Discussing result of no.5 and no.7

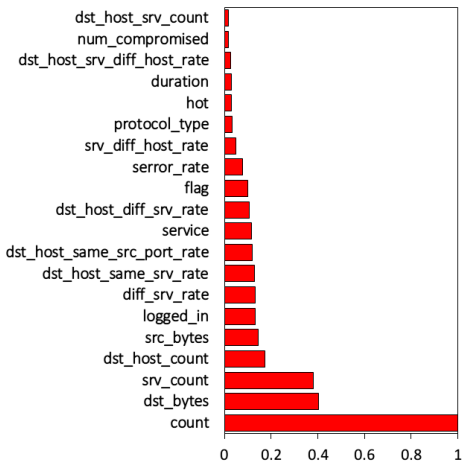
No.5 and No.7 respectively compare with No.6 and No.8 by using same algorithm in normal traffic selection model, and their ranks are lower when using decision tree in abnormal traffic



(a) No.11



(b) No.8 Normal traffic selection model



(c) No.8 Abnormal traffic selection model

Figure 1: Importance of Variables in Random Forest.

Table 6: Confusion matrix

No.11					
Prediction	**Normal**	**DoS**	**Probe**	**U2R**	**R2L**
Normal	60287	5967	847	159	15839
DoS	69	223814	191	8	0
Probe	233	72	3128	50	104
U2R	1	0	0	10	5
R2L	3	0	0	1	241
False Negative	0.00505	0.026273	0.24916	0.95614	0.985113
No.8 **Normal traffic selection model**					
Prediction		**Normal**		**Abnormal**	
Normal		60289		22853	
Abnormal		304		227583	
False Negative		0.005017		0.091253	
No.8 **Abnormal traffic selection model**					
Prediction		**DoS**	**Probe**	**U2R**	**R2L**
DoS		229231	769	20	4693
Probe		297	3393	135	5646
U2R		0	0	39	32
R2L		325	4	34	5818
False Negative		0.002706	0.18555	0.828947	0.64062

Table 7: Confusion Matrix of abnormal traffic selection model with decision tree

Predition	**DoS**	**Probe**	**U2R**	**R2L**
DoS	227792	589	34	6245
Probe	1434	3192	20	283
U2R	0	0	0	0
R2L	627	385	174	9661

selection model.

Table 7 is confusion matrix of abnormal traffic selection model with decision tree algorithm. It shows that U2R can not be detected and false negative rate of R2L is higher. In order to find the reason, classify tree is checked in Fig. 2, where the classification model prefers DoS and Probe attack, then R2L attack, and no result point of U2R attack. Distribution of training data can explain this phenomenon, which can be shown in Fig. 3.

When generating decision tree, the obtained information will cause results in favor of feature which have more samples. Therefore, if the number of training data set in every group is different enough, it cannot get efficient classification model for small samples. Moreover, because the number of between training data is comparatively equal, classification result is better, such as No.6, when normal traffic selection model uses decision tree.

Discussing result of no.3 and no.4

No.3 and No.4 use k-means in normal traffic selection model to choose clustering center. Table 8 shows final prediction accuracies in No.3 and No.4. Because final results are lower than that of normal traffic selection model or abnormal traffic selection model, we find that this problem is caused by using k-means in normal selection model. Table 9 shows confusion matrix

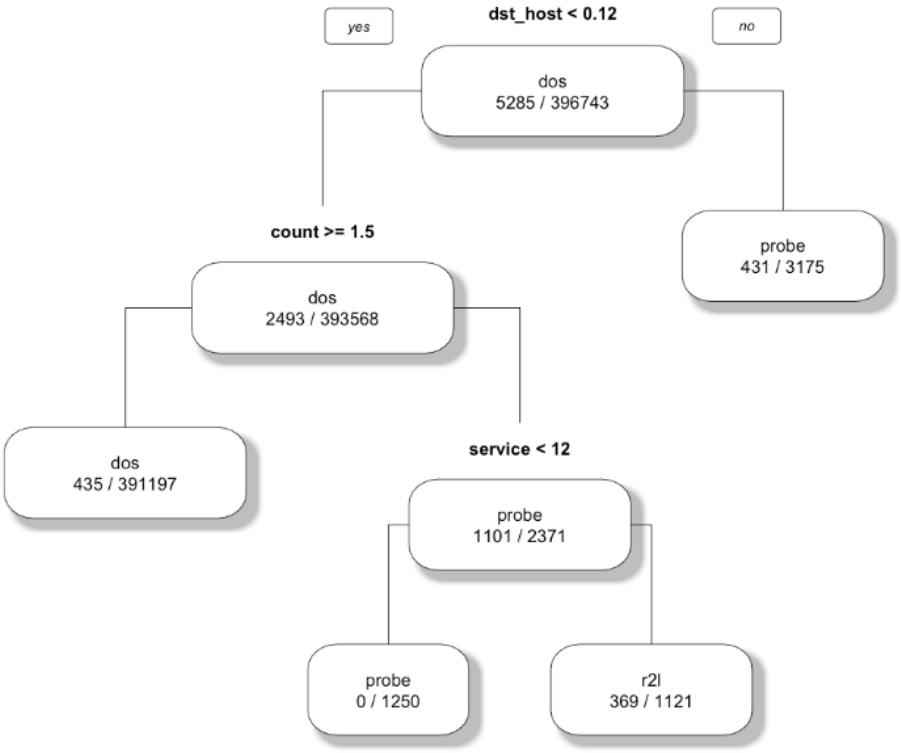


Figure 2: Classify tree of abnormal traffic selection model.

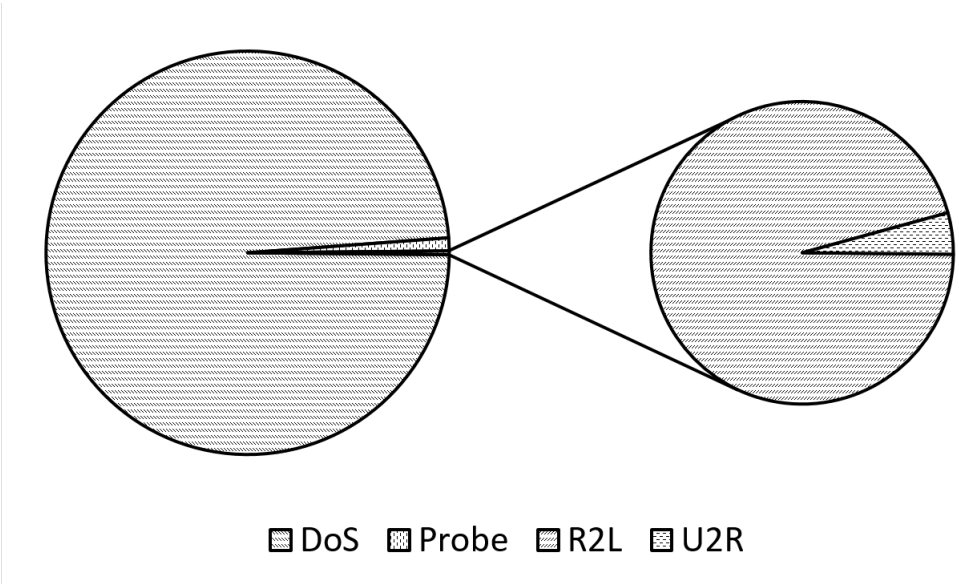


Figure 3: Distribution of training data.

Table 8: Accuracy of no.3 and no.4

No.	Model	Algorithm	Accuracy
No.3	Normal Traffic Selection	k-means	0.926
	Abnormal Traffic Selection	Random Forest	0.948
	Mixed Compensation Model		0.923
No.4	Normal Traffic Selection	k-means	0.925
	Abnormal Traffic Selection	Decision Tree	0.961
	Mixed Compensation Model		0.918

Table 9: Confusion matrix of normal traffic selection model of no.3 and no.4

No.	Prediction	Normal	Abnormal
No.3	Normal	59189	21663
	Abnormal	1404	228773
No.4	Normal	59428	22221
	Abnormal	1165	228215

of normal traffic selection model of No.3 and No.4. Many abnormal records are predicted as normal, which cause high false negative rate. Therefore, many abnormal records predicted by abnormal traffic selection model will be regarded as normal after mixed compensation model.

Nowadays, many novel attacks are unknown to researchers, and many attacks will be disguised as normal. It's very dangerous to have high false negative rate, and it does not fit the proposed model.

Because the effect of k-means has great correlation with the number of centers chosen to cluster, and we can fine tune the strength of clustering, and lower the false negative rate to establish a strict normal selection model.

In No.3 and No.4, the number of centers for normal traffic and attacks is 100 and 300, respectively. Although it can achieve a good overall accuracy, its false negative rate is higher than other model. However, according to Table 10, by choosing 4 and 30 in No.1 and No.2, it has lower false negative rate, and only classify four kinds of attacks. Besides, a strict normal detection model is established.

By adjusting the parameters and reducing false negative rate in No.1 and No.2, we can find that the rank has increased rapidly compared with No.3 and No.4. Especially, when K-means combines with random forest, it has a very high accuracy on Probe, U2R and R2L attack. Therefore, we can draw the conclusion that by adjusting the parameters of K-means, the strength of abnormal traffic detection can be controlled by adjusting the strength of normal traffic identification.

4.3 Summary

Based on the results analyzed above, as shown in Table 11, the following conclusions can be drawn:

1. Random forest classification algorithm can adapt to the change of distribution of network data, and this algorithm by using the proposed model can reduce false negative rate.

2. If the number of training data in different group is largely different with each other, the classify model built by decision tree will prefer to attack types, which have more training data. So we should avoid using decision tree in abnormal traffic selection model. However, in the normal traffic selection model, the difference between different groups is comparatively small. In this

Table 10: Results of experiments

No.	Experiment	DoS	Probe	U2R	R2L	Final	Rank
1	k-means1+Random forest	10	1	1	2	14	1
6	Decision tree+ Random forest	1	8	2	5	16	2
8	Random forest + Random forest	2	5	4	7	18	3
11	Random forest	2	4	3	9	18	3
3	k-means2+ Random forest	5	2	5	7	19	5
2	k-means1+ Decision tree	11	3	6	1	21	6
7	Random forest + Decision tree	7	6	6	4	23	7
5	Decision tree + Decision tree	4	9	6	6	25	8
9	k-means	9	11	6	3	29	9
4	k-means2+ Decision tree	8	7	6	10	31	10
10	Decision tree	5	10	6	11	32	11

Table 11: Summary of model

	Model 1	Model 2	Model 3
Normal traffic selection model	k-means1	Decision Tree	Random Forest
Abnormal traffic selection model	Random Forest	Random Forest	Random Forest

situation, using decision tree can fast get classify model, and the results have higher accuracy. 3. There are more and more unknown abnormal events in the future. In order to avoid loss of false negative prediction, we can change the number of clustering in the normal traffic selection model with k-means algorithm to reduce false negative rate and increase the accuracy of detecting abnormal events.

Conclusion

With the change of distribution of network data, traditional anomaly traffic detection techniques can not fit this situation. In order to solve the problem, we propose an anomaly traffic detection model based on big data analysis. Simulation results show that the proposed model achieves a detection rate of 95.4% on normal data, 98.6% on DoS attack, 93.9% on Probe attack, 56.1% on U2R attack, and 77.2% on R2L attack.Therefore, the model can increase the accuracy of attack behavior, and reduce false negative rate.

Acknowledgment

This work was supported by NSFC (61471056) and China Jiangsu Future Internet Research Fund (BY2013095-3-1, BY2013095-3-03).

Bibliography

[1] Patcha, A.; Park, J.M. (2007); An overview of anomaly detection techniques: Existing solutions and latest technological trends, *Computer Networks*, ISSN 1389-1286, 51(12): 3448-3470.

[2] Lazarevic, A.; Kumar, V.; Srivastava, J. (2005); Intrusion detection: A survey, *Managing Cyber Threats*, ISSN 0924-6703, 5: 19-78.

[3] Axelsson, S. (1998); Research in intrusion-detection systems: a survey, *Department of Computer Engineering, Chalmers University of Technology, Goteborg. Sweden*, Technical Report 98-17.

[4] Om, H.; Kundu, A. (2012); A hybrid system for reducing the false alarm rate of anomaly intrusion detection system, *IEEE 1st International Conference on Recent Advances in Information Technology (RAIT)*, ISBN 978-1-4577-0694-3, 131-136.

[5] Kaisler, S. et al (2013); Big data: Issues and challenges moving forward, *IEEE 46th Hawaii International Conference on System Sciences (HICSS)*, ISSN 1530-1605, 995-1004.

[6] Michael, K.; Miller, K.W. (2013); Big Data: New Opportunities and New Challenges, *Computer*, ISSN 0018-9162, 46(6):22-24.

[7] Russom, P. et al (2011); Big Data Analytics, *TDWI Best Practices Report*, Fourth Quarter.

[8] Fan, W.; Bifet, A. (2013); Mining big data: current status, and forecast to the future, *ACM SIGKDD Explorations Newsletter*, ISSN 1931-0145, 14(2): 1-5.

[9] James, G. et al (2013); An introduction to statistical learning, *Springer*, ISSN 1431-875X.

[10] Guan, Y.; Ghorbani, A.A.; Belacel, N. (2003); Y-means: A clustering method for intrusion detection, *IEEE Canadian Conference on Electrical and Computer Engineering*, ISSN 0840-7789, 2:1083-1086.

[11] Quinlan, J.R. (1993); C4.5: Programs for Machine Learning, *Morgan Kaufmann Publishers Inc.*, ISBN 1558602402.

[12] Elbasiony, R.M. et al (2013); A hybrid network intrusion detection framework based on random forests and weighted k-means, *Ain Shams Engineering Journal*, ISSN 2090-4479, 4(4): 753-762.

[13] KDD Cup 1999, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. May 2015

[14] Lippmann, R.P. et al (2000); Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation, *IEEE Proceedings of DARPA Information Survivability Conference and Exposition (DISCEX)*, ISBN 0-7695-0490-6, 2:12-26.

[15] Tavallaee, M. et al (2009); A detailed analysis of the KDD CUP 99 data set, *Proceedings of the Second IEEE Symposium on Computational Intelligence for Security and Defence Applications (CISDA)*, ISBN 978-1-4244-3763-4, 1-6.

[16] Pfahringer, B. (2000); Winning the KDD99 classification cup: bagged boosting, *ACM SIGKDD Explorations Newsletter*, ISSN 1931-0145, 1(2): 65-66.

[17] Yu, G. D. et al (2014); Multi-objective rescheduling model for product collaborative design considering disturbance, *International journal of simulation modelling*, ISSN 1726-4529, 13(4): 472-484.

[18] Gusel, L. R. et al (2015); Genetic based approach to predicting the elongation of drawn alloy, *International journal of simulation modelling*, ISSN 1726-4529, 14(1): 39-47.

[19] Prasad, K. et al (2016); A knowledge-based system for end mill selection, *Advances in Production Engineering & Management*, ISSN 1856-6250, 11(1): 15-28.

INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 11(4):580-593, August 2016.

A Forward-connection Topology Evolution Model in Wireless Sensor Networks

C. Zhang, C. Li, N. Ning

Changlun Zhang*, Nan Ning
Science School
Beijing University of Civil Engineering and Architecture
Beijing, China
zclun@bucea.edu.cn, ningnan@stu.bucea.edu.cn
*Corresponding author: zclun@bucea.edu.cn

Chao Li
Beijing Key laboratory of Communication and Information Systems
Beijing Jiaotong University
Beijing, China
lichao21261@163.com

Abstract: The stability and reliability of the topology structure play an important role in the efficiency of the data collecting for wireless sensor networks. In this paper, a topology evolution model is proposed. The model considers the directionality of the data flow, and adopts the forward connectionism to ensure the neighbor nodes of each node. Furthermore, the model considers the balanced energy overhead in each communication path, adopts the energy balanced mechanism to compute the connection probability to the neighbor nodes. Meanwhile, the process of topology evolution is distributed and the communication radiuses of all sensor nodes are limited. A theoretical analysis exhibits that the model has power-law distribution of node degrees. Simulation shows that the proposed topology evolution model make energy overhead more balanced, and prolongs the lifetime of the network.
Keywords: wireless sensor networks; topology evolution; energy balanced mechanism; power-law distribution

1 Introduction

Wireless sensor networks (WSNs) are a kind of wireless networks which are constructed by plenty of sensor nodes. WSNs can gather the data from its monitoring physical or environment conditions (e.g. the temperature, the sound etc.) and send their data to the destination (Base Station) directly or via multi-hop [1,2]. WSNs cover a wide range of applications, since it is easily deployed and self-organized, such as environmental monitoring, military target tracking, natural disaster relief and health monitoring, and so on [3,4]. On the other hand, WSNs are weakness in processing capability and storage capacity. Especially, when the WSNs are deployed in a harsh environment that people hardly reach, the nodes are difficultly recharged and replaced, and it leads to the limited energy sensor nodes. Therefore, establishing a model which can prolong the lifetime of WSNs efficiently is a very important issue.

The study of complex networks has become a common focus of many branches of science since the end of last century [5, 6]. The complex networks study the characteristics of the networks, which can describe many systems in nature, such as the cooperative networks, social networks and so on. Most complex networks are scale-free networks, which are robust against random removal or failures of nodes. Recently, complex network is used to study connectivity, fault tolerant and topology evolution of the wireless sensor networks.

In this paper, we proposed a forward-connection topology evolution model for wireless sensor networks. Different from existing schemes, in the proposed model, a node sends the data to those nodes that are nearer to the base station than itself.

The remainder of this paper is organized as follows. In Section 2, the related work is summarized. In Section 3 and 4, an algorithm of forward-connection topology evolution model is proposed and analyzed. In Section 5, the simulation to present the features of the networks generated by the proposed algorithms is provided. Finally, the conclusion of this paper is given.

2 Related works

The topology evolution models can be divided into two types roughly: the cluster-based and the non-cluster-based. In the non-cluster-based topology evolution models, all the nodes transit the data by a chain, a tree and so on to the base station. These models are used to some small WSNs or the single type nodes.

In 2002, S. Lindsey et al. [7] proposed PEGASIS (Power-Efficient Gathering in Sensor Information Systems) model, an optimal chain-based protocol, which can reduce the energy consumption. But if a middle node in the chain is drained, all the nodes behind the drained node $can't$ transmit the data to the base station. In 2003, H. O. Tan et al. [8] improved the PEGASIS Model, and proposed PEDAP algorithm. The algorithm is near to optimal minimum spanning tree based routing schemes.

In 2003, Xiang Y L et al. [9] considered how the transmission range is related with the number of nodes in a fixed area such that the resulted network can sustain k fault nodes with high probability. Then they presented a localized method to control the network topology.

In 2005, Thallner et al. [10] presented an improvement of topology control algorithm for dynamic networks and low power devices, which provided an improvement of topology control algorithm for dynamic networks and low power devices.

In 2006, Abhishek et al. [11] proposed an approximation algorithm, which used additional relay nodes to construct a fault-tolerant backbone network. On the other hand, the cluster-based topology evolution models divide the WSNs into the inner-cluster layer and inter-cluster layer. These two layers usually possess the self-similarity. These models are used to some large or mixed WSNs.

In 2000,W. R. Heinzelman et al. [12] presented an LEACH protocol which is able to distribute energy dissipation evenly throughout the sensors, doubling the useful system lifetime for the networks we simulated. But the LEACH protocol is not fit for the huge WSNs and the WSNs which have the unbalanced energy for each node.

In 2003, S. Bandyopadhyayet al. [13] proposed a distributed, randomized clustering algorithm to organize the sensors into clusters in a wireless sensor network. They then extended this algorithm to generate a hierarchy of cluster heads and observe that the energy savings increase with the number of levels in the hierarchy.

In 2004, Ossama et al. [14] proposed HEED protocol, which selected cluster heads periodically according to the node residual energy. The HEED protocol can prolong the lifetime of WSN.

Recently, the evolution models which combine the complex networks come up.

In 2009, Li J C et al. [15] proposed an evolving model among the cluster heads of WSN. The model is based on the random walker theory, and exhibits a power-law distribution.

In 2009, Zhu et al. [16] presented two self-organized energy-efficient models for WSNs. The first model constructed evolving network considering the connectivity and remaining energy of each node. The second model considered the energy consumption balance of the whole network.

In 2011, Xiao G Q et al. [17] proposed a topology evolution TEBAS, which introduces fitness and local world and more suitable for WSNs than ERW (Topology Evolution by Random Walker).

In 2012, Ya Q W et al. [18] studied the influence of node failure on the performance of WSN, and different immunization strategies were given.

In 2012, Xiao J L et al. [19] considered the energy-aware mechanism, and proposed a new topological evolving model based on the complex network theory. They found that node energy distribution had the weak effect on the degree distribution.

In the practical applications, a node sends the data to the others that are nearer to the base station than itself. This kind of connection reduces the energy consumption and the delay of transmission. Based on this connection, a forward-connection topology evolution model is proposed, which considers both the forward-connection mechanism and the energy-balanced mechanism. Analysis and simulation show that the network exhibits well robustness, a power-law degree distribution and a long lifetime.

3 Forward-connection topology evolution model

The model uses the cluster network which contains three kinds of nodes: base station, cluster heads and cluster nodes. The network is divided into two layers: inner-cluster and inter-cluster. In the inner-cluster, data are sent to the cluster head, and the cluster head verifies the data integrity to restrict the range of compromised node. In the inter-cluster, data are sent to the base station, and the integrity is verified at the base station. Furthermore, a mechanism is proposed to locate the compromised node. SPPDA model can be divided into initialization, key distribution, inner-data aggregation and inter-data aggregation.

In this section, a forward-connection mechanism is considered. Meanwhile, the energy of each node in a path should be kept balance. It means that there is no node which energy is less than others. Then, an energy balanced mechanism is considered, which prolong the lifetime of the path.

In WSNs, most energy of sensor nodes was used in data transmission. So, it ′s assumed that the higher energy a node has, the greater connection probability the node holds in this model. Besides, referred to the evolution principles of complex networks, preferential attachment of the nodes is present in the processes of the topology evolution. Meanwhile, considering the relationship between the distance and the energy consumption, a threshold of communication radius should be set, which decides the communication probability between two nodes. The probability equals zero when the distance of two nodes is larger than the threshold. And the closer of two heads are, the larger probability connects to these heads is. Here, $d_{j,max}$ is used to represented the communication radius of node j, and d_{max} is the communication radius of the node whose energy is E_{max},then $d_{j,max} = \frac{E_j}{E_{max}} d_{max}$.

Since each node in the network are homogenous, the initial node actual does not exist in the theory of distributed mechanism, which is considered in this paper. All nodes will send a message to its surrounding node when the topology evolution begins. According to the time of index, the node sends agreement to the earliest request, and refuses to the rest. The messages that nodes sent are direction in the evolution. Then a directed network is considered, where the direction shows the direction of data flow.

3.1 Forward-connection mechanism

In forward-connection mechanism, the forward neighbors of each node are confirmed. The mechanism is divided into two steps. In the first step, the neighbors of sending node are confirmed. In the second step, if the base station is the neighbor of sending node, the sending node has no forward neighbors, because the sending node transmits the data to the base station directly. If the base station is not the neighbor of sending node, the neighbor nodes whose distance

to the base station is smaller than the sending node are selected. These neighbor nodes are the forward neighbors of sending node. As shown in Figure 1, point O is the base station. Point I is the sending node. Line OA, OI and OB is the distance between sending node and the base station. Lind IA and IB is the communication radius of the sending node. Point P is a neighbor of sending node. Then, a conclusion to confirm the forward neighbors is given as follows.

Conclusion: if $\angle PIO$ is less than 60 degree, P is the forward neighbor of sending node.

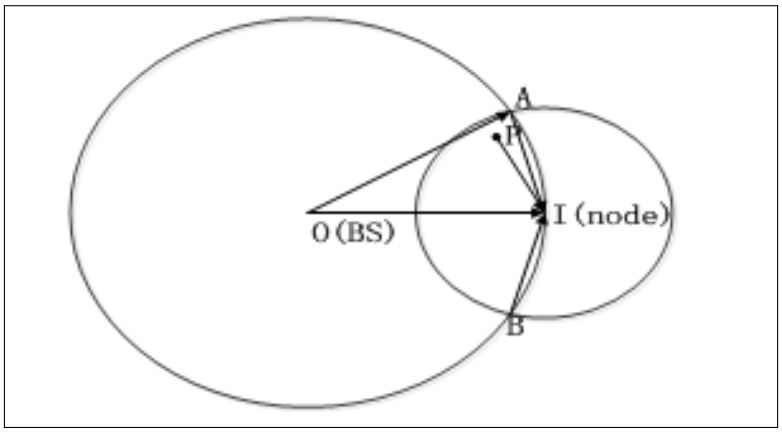


Figure 1: Forward-connection mechanism

Proof: The forward neighbors of sending node exist. So, we have $OA > AI$, then $\angle OIA > \angle AOI$. On the other hand, $OA = OI$, so $\angle OIA = \angle IAO$.In $\triangle\ AIO$, $\angle AIO + \angle IAO + \angle AOI = 180°$. So, $180° = \angle AIO + \angle IAO + \angle AOI < \angle AIO + \angle IAO + \angle AIO = 3\angle AIO$. That is $\angle AIO > 60°$. In the same way, we have $\angle OIB > 60°$. □

According to the conclusion, the forward neighbors of each node can be confirmed easily.

3.2 Energy-balanced path mechanism

In energy-balanced mechanism, the neighbors of sending node in two hops are considered. In this paper, an energy-balanced element is used to judge the level of the energy-balanced. And a threshold ŚĹ is used to distinguish the high energy nodes and the low energy nodes. If the energy of a forward neighbor ϵ is larger than the energy of sending node, this forward neighbor is a high energy node. Node i is one of the forward neighbors of sending node. N is the number of the forward neighbors of node$i$, and N_ϵ is the number of the high energy nodes of node i. Thus, the energy-balanced element of node i can be counted as $\delta_i = \frac{N_\epsilon}{N}$.

3.3 Model of evolution network

In this model, a network is modeled as a directed graph *G(V, E)*, where nodes are represented as the set of vertices V and the links as the set of edges E. The number of sensor nodes is defined as $|V| = N$. In the topology evolution networks, there are two rules of the distributed and local-world topology evolution model which is R-1 (growth) and R-2 (preferential attachment) is given as follows:

R-1:After the node i is selected, this node sends the message to other nodes in its local-world, which means the circle area is within the maximum communication radius $d_{j,max}$ of the head i.

The nodes send agreement to the earliest request, and refuses to the rest.

The evolution ends when each node succeed to connect other nodes actively.

R-2: Until the each node in its local-world returns the agreement to the request node, the request node connects m edges to these nodes with the probability $\Pi_{i\to j}$. If the number of the nodes is less than m, it connects all of the nodes. $\Pi_{i\to j}$ represents the probability of a head j connected to head i. The probability $\Pi_{i\to j}$ depends on the connectivity, the distance of two nodes and the threshold of node i. The form of $\Pi_{i\to j}$ is $\Pi_i=\frac{E_j\delta_j k_j}{\sum_{l\in\Lambda_i} E_l\delta_l k_l}$,where $\bigwedge_i$ is the local-world of node i.

3.4 Model of evolution network

In a network, the energy consumptions in each node are different according to the different distances and the amount of data. The route is constructed by the initial energy. Then, the energy is consumed when the nodes send the data to the base station. And the remaining energy in each node is changed. So, the local-reconstruction is needed when the network runs for a while.

In this section, two local-reconstruction mechanisms are provided which are initiative local-reconstruction mechanism and passive local-reconstruction mechanism. The initiative local-reconstruction mechanism is a centralized reconstruction, and the local-reconstruction require is sent by the node that has a lower energy. The passive local-reconstruction mechanism is a distributed reconstruction. And the local-reconstruction require is sent by the node that whose parent node has a lower energy. These two mechanisms are proposed as follows:

The initiative local-reconstruction

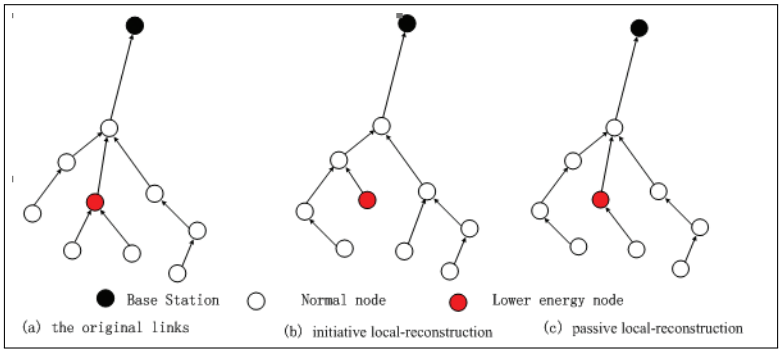


Figure 2: Local-reconstruction mechanisms

In initiative local-reconstruction, when a node that has a lower energy exists in the network, this node determines the nodes that need to change the routes in its neighbors. Then, this node sends the local-reconstruction requests to these nodes. The nodes that receive the requests delete the link from the lower energy node. Finally, these nodes connect to other nodes with the evolution model above. Figure2(b) shows the initiative local-reconstruction.

Algorithm1 shows the algorithm of initiative local-reconstruction. N_i is the lower energy node,NF_i is the parent node of node N_i, NS_i is the set of son nodes with node N_i.

The Initiative Local-Reconstruction

In passive local-reconstruction, each node determines whether it needs reconstruction according to the rate between the node and its parent node. A node need, it sends the local-reconstruction requests to its parent when necessary. Then, the parent node deletes the link between them after receiving the requests. Lastly, this node connects to other nodes with the

Algorithm 1. Algorithm of initiative local-reconstruction

```
Set the threshold δ1 and δ2
Node Ni
For all the nodes in the wireless sensor network
if Ni is a leaf node then
    if E(Ni) < δ1 then
      Delete the connection of its parent node
      Use the evolution model to choose a new parent node
    else
      if E(Ni) < δ2 then
        Delete the connection of its parent node
        Set its energy-balanced element of node Ni to 0
        Use the evolution model to choose a new parent node
        Send the reconstruction message to all of its son nodes NSi
      else
        if δ2 < E(Ni) < δ1 then
          Send the reconstruction message to all of its son nodes NSi
        end if
      end if
    end if
end for
Nodes in NSiSet
For all the nodes in the NSiset
Delete the connection of node Ni
Set its energy-balanced element of node Ni to 0
Use the evolution model to choose a new parent node
```

Figure 3: Algorithm of initiative local-reconstruction

Algorithm 2. Algorithm of passive local-reconstruction

```
Set the threshold δ1
For all the nodes in the wireless sensor network
Each node i
Get the energy of the parent node of node i: E(Pi)
E(i) is the energy of node i
if E(Pi)/E(i) < δ1 then
  Delete the connection of its parent node
  Set its energy-balanced element of node Ni to 0
  Use the evolution model to choose a new parent node
end for
```

Figure 4: Algorithm of passive local-reconstruction

evolution model above. Algorithm 2 shows the passive local-reconstruction.Figure.2(c) shows the algorithm of initiative local-reconstruction. N_i is the lower energy node, NF_i is the parent node of node N_i, NS_i is the set of son nodes with node N_i.

4 Analysis

Degree distribution is an important and useful feature for a given complex networks. The degree distribution of this network is analyzed by the mean field theory.

We assumed that N is the total number of the nodes in the WSNs, and these nodes are uniformly distributed in the WSNs. Some important parameters are defined in Table 1.

Table 1: Definition of important parameters

Parameters	Definition
N	Number of nodes in a network
d_{max}	The maximum communication radius0
M	Numbers of new edges which a new node connect at every time step
k_i	Number of links connected to node i
d_{max}	60 600 150 200
E	Remaining energy of a node
L	Number of nodes in every new comer′s local-area
t_i	Time of node i newly introduced into the network

The evolution process described in this paper adopts the distributed mechanism, which can describe the real evolution processing more efficiently. Actually,the nodes can not begin to send the message at the very same time/concurrently, so it can be approximately regarded the process as follows.

R-1: Starting with one node and m edges, at each step, a new node with m edges was added. The node can connect with all the nodes surrounding.

R-2: When a new node comes into the network, it will choose some nodes in its local-world to connect with the probability $\prod_{i\to j}$. During each unit of time, m new edges are formed.

So we get

$$\frac{\partial k_i}{\partial t} \approx \frac{mL_i \prod_{j\to i}}{N} = \frac{mL_i E_j \delta_j k_i}{N\Sigma_{l\in\Lambda\Lambda_j} E_l \delta_l k_l} \tag{1}$$

Here, L_i is the number of nodes which can connect to node i.

According to the mean field theory [20]:

$$\sum_{l\in\Lambda_j} E_l \delta_l k_l = L_j \bar{E}\bar{\delta}\langle k\rangle \tag{2}$$

Similarly, L_j is the number of nodes in the new comer′s (node j) local-world. $\bar{E}$ is the mean value of the local-world energy, and $\langle k\rangle$ is the average degree of local-world. In a large scale network, the average degree can be calculated as

$$< k >= \frac{1}{t+1} = \varepsilon_j m \tag{3}$$

Where l is the number of edges connected to the node that has joined. it′s not a fixed number while it satisfied the equation:$0 \leq l \leq (t+1)m$, $\varepsilon_j m$ is the mean degree of the network after node j joined. The ϵ_j increases when time t increases constantly.

So it′s simply defined as $\epsilon_j = \frac{t}{N}$.

Combine three equations above, we get

$$\frac{\partial k_i}{\partial t} = \frac{L_i \delta_i E_i}{L_i \bar{e} \bar{\delta}} \frac{k_i}{t} \tag{4}$$

We set

$$\frac{L_i \delta_i E_i}{L_i \bar{e} \bar{\delta}} = g \tag{5}$$

Then we have

$$\frac{\partial k_i}{\partial t} = g \frac{k_i}{t} \tag{6}$$

Solving this differential equation, we get $k_i(t) = Ct^g$. Since $k_i(t_i) = m_i$, thus we have

$$k_i(t) = \frac{m_i}{t_i^g} t^g \tag{7}$$

The probability that a node has connectivity $k_i(t)$ smaller than k is

$$P(k_i(t) < k) = P(\frac{m_i}{t_i^g} t^g < k) = 1 - P(t_i < (\frac{m_i}{k})^{\frac{1}{g}} t) \tag{8}$$

Assume that we add the nodes to the network at equal time intervals, the probability density at the time is $f(t_i) = \frac{1}{t+1}$, therefore, we get

$$P(k_i(t) < k_{in}) = 1 - P(t_i < (\frac{m_i}{k_{in}})^{\frac{1}{g}} t) = 1 - \frac{t}{t+1} (\frac{m_i}{k_{in}})^{\frac{1}{g}} \tag{9}$$

The probability density function of the degree of a node with remaining energy E is

$$P(k_{in}) = \frac{\partial P(k_i(t) < k_{in})}{\partial k_{in}} = \frac{t m_i^{\frac{1}{g}}}{t+1} k_{in}^{-(1+\frac{1}{g})} \tag{10}$$

Here,$g = \frac{L_j E_{\ j} \delta_j}{L_j \bar{E} \bar{\delta}}$

Therefore, the distribution has a power-law form with degree exponent $\gamma = -(1 + \frac{1}{g})$ in the network. Thus, we can organize WSNs with scale-free feature in this energy-aware algorithm. This algorithm can not only make the network evolution in energy-efficient, but also improve the network reliance against random errors, which is an inherent advantage of most scale-free networks. Especially, when the distributions of energy and nodes are uniform distribution, it is got that $g \approx (1 - d(i,j)/d_{j,max})$, which is obvious that $\gamma < -2$, and if d_{max} is big enough, the degree exponent $\gamma \to -2$.

5 Simulations

In the evolution of the network, we assume that the cluster heads have been selected, and the network has 1000 cluster heads. These cluster heads are randomly deployed over a 600 meters ×600 meters. The network parameters are listed in Table 2.

In the simulation of lifetime and network-efficiency, we compare three kinds of aggregation tree: the tag aggregation tree [21] (Tag aggregation tree), the tag aggregation tree with forward-connection evolution model and the FCTag tree with reconstruction.

5.1 The degree distribution

Table 2: Main settings in simulation

Parameters	Setting
Area	600×600
N	1000
m	2 6 15 30
E	Random in [0.8 1]
d_{max}	60 600 150 200

In the analysis, there are two parameters influence the degree distribution: d_{max} and m. The following simulation shows the influence of d_{max} and m on the evolving network.

The influence of d_{max}

In this section, the influence of d_{max} is discussed. Four different value of d_{max} are listed in table 2 (m=3).

As shown in Figure 5, when d_{max} is small,$f < 1$, so the exponent $\gamma < -2$. When d_{max} gets larger, f closes to 1, then the exponent γ closes to -2. Besides, Figure 1 also shows that when the d_{max} increases constantly, the maximum in-degree get larger. This is because the larger the communication radius is, the more likely be connected by other nodes.

The Influence of m

In this section, the influence of m is discussed with different m ($d_{max} = 120$).

As shown in Figure.6, when m is small, the process of preferential attachment works well, and a node is connected randomly, the network exhibits a power-law degree distribution. When m gets larger, the process of preferential attachment is limited. It leads to the reduction of the randomness of the connection. So the degree distribution can′t obey the power-law form well. Especially, when m is larger than the neighbors, the node will connect all nodes surrounding, which is independent of the preferential attachment.

5.2 Lifetime

In the simulation of lifetime, we assume that all sensor nodes have an initial energy which is 0.5J. The data packet size is 1000 bits. There is not an exact definition in lifetime for a WSN. Some papers use the round when the first drained node appears. Some papers use the round when a certain ratio of drained nodes appears. Some papers use the round when the network canĄŻt cover the monitor area. In this paper, we just run the network for a certain round (here

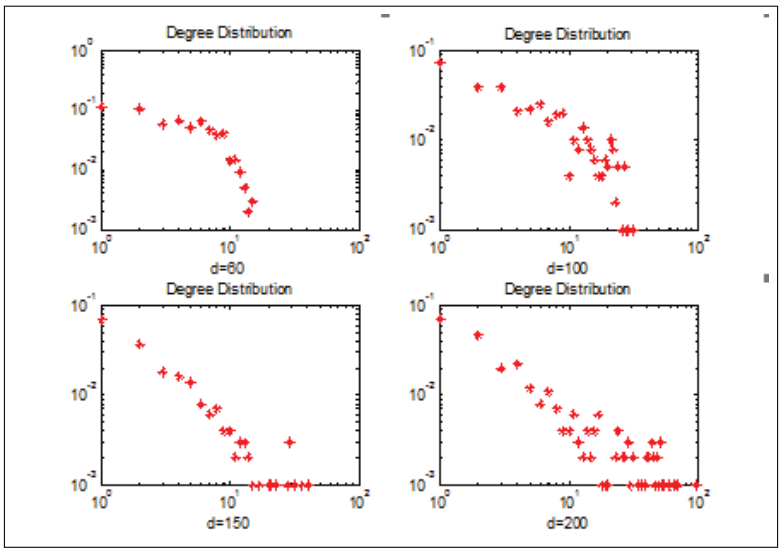


Figure 5: The influence of d_{max}

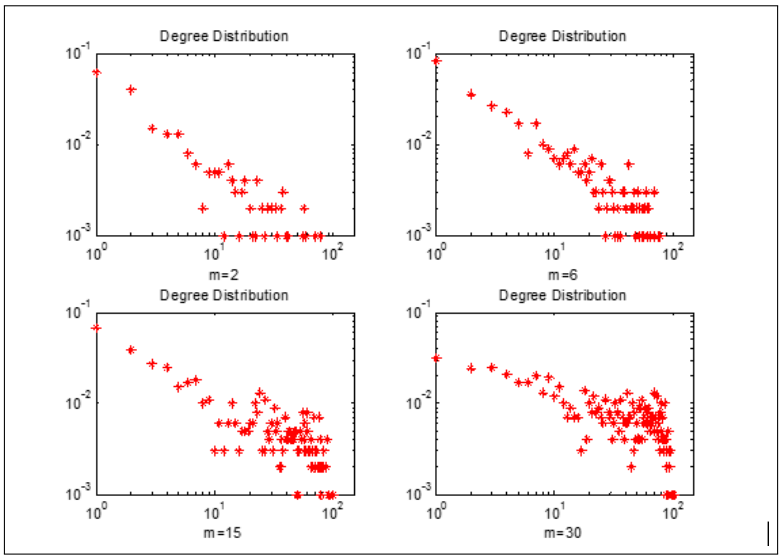


Figure 6: The influence of m

uses the 1500 round), and we observe the changing rule in different aggregation tree. We deem an aggregation tree has a longer lifetime if the drained nodes increase slower. Nodes consume energy both in sending and receiving data according to [22]. In this paper, we use the model that the pass loss exponent is 2. The model is as follows:

A k-bit data packet is transmitted and the energy consumption of sending node is given by $E_t = \varepsilon_1 \times k + \varepsilon_2 \times d^2 \times k$, d is the distance between the two sensor nodes, and $\varepsilon_1 = \frac{50nJ}{bit}, \varepsilon_2 = \frac{100pJ}{bit \cdot m^2}$ .A k-bit data packet is transmitted, and the energy consumption of receiving node is given by $E_\gamma = \varepsilon_1 \times k$

Without Local-Reconstruction

As shown in Figure 7, the dotted line is the lifetime of Tag aggregation tree. The solid line is FCTag aggregation tree. Obviously, in the dotted line, the first drained node appears at the 592^{th} round. In the solid line, it appears at 737^{th} round. This shows that the FCTag aggregation tree consumes the energy more balance than the Tag aggregation tree.

Another key point appears at about the 1280^{th} round. Two lines intersect. And after this point, the number of drained nodes in dotted line is larger than that in solid line. It means that, after some round, the energy consumption in our model is more unbalance than the pure tag aggregation tree.

The point of intersection appears normal and reasonable. There are two reasons for the point. Firstly, the FCTag aggregation tree is constructed according to the energy and the position of nodes at that time. With the network running, the energy decreases which make the FCTag aggregation tree is no longer suit for the network now.

Secondly, the idea of FCTag aggregation tree is that, when the nodes transmit the data, the nodes that have large energy use more energy, and the nodes that have low energy use less energy. So the nodes that have low energy can run more round, but it makes the nodes that have large energy run less round. In fact, the FCTag aggregation tree has large total energy consumption in one round than Tag.

In the first reason, a local-reconstruction mechanism can solve it well. But the second reason is because of the idea of our model. Finally, the point of intersection is difficult to remove, but it can be delayed.

Local-Reconstruction

In this part, we simulated the lifetime with Tag aggregation tree and FCRTag aggregation tree.

In the Figure 8, the dotted line is the lifetime of Tag aggregation tree, the solid line is FCRTag aggregation tree. It shows that when a local-reconstruction mechanism is considered in the FCTag aggregation tree. The lifetime of the network is better. The drained node appears in 737^{th} round too. And obviously, there is no intersected point before the 1500^{th} round. But in fact, these two lines are getting more and more near when the round increasing. So it can be inferred that there is an intersected point after 1500^{th} round.

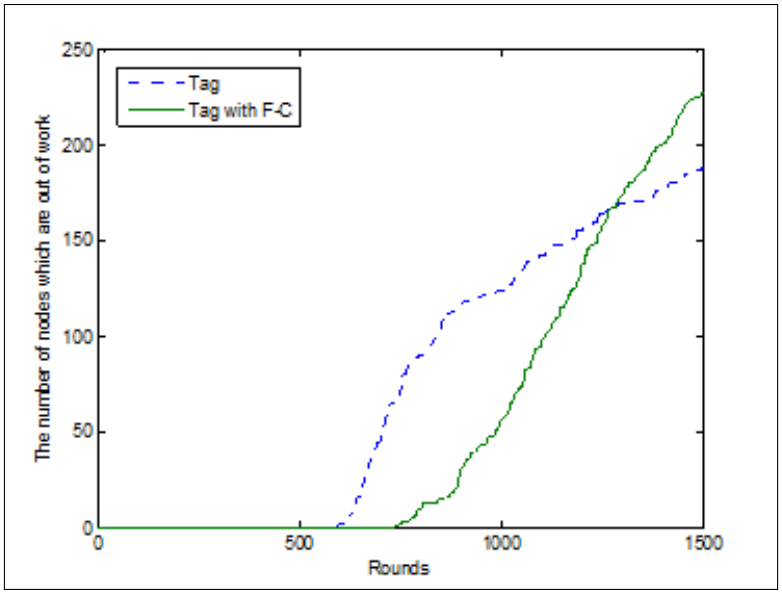


Figure 7: The lifetime without local-reconstruction

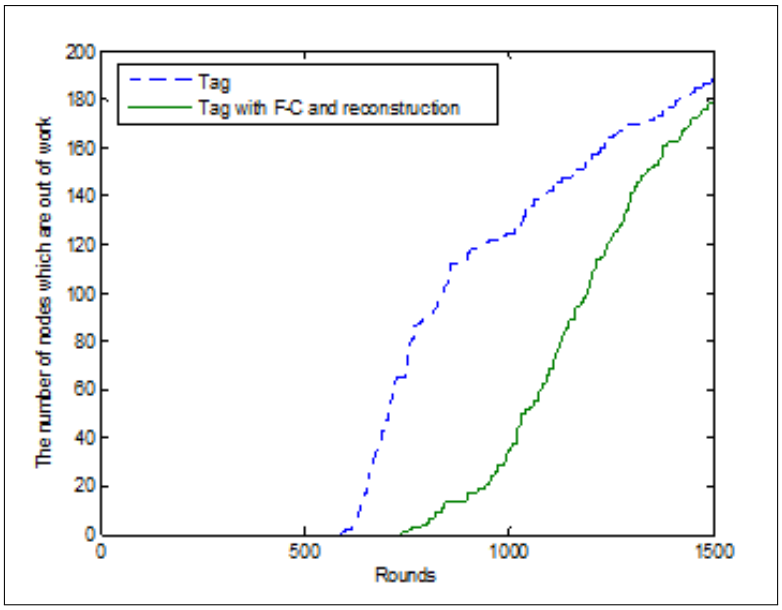


Figure 8: The lifetime with local-reconstruction

Conclusions and future work

In this paper, we present a new secure privacy-preserving data aggregation model, which adopts a mixed data aggregation structure of tree and cluster. The proposed model verifies the data integrity both at the cluster nodes and the base station. Meanwhile, the model gives a mechanism to locate the compromised nodes. Finally, the detail analysis shows that this model is robust to many attacks, and has lower communication overhead.

Acknowledgment

This work is supported by Beijing Natural Science Foundation under Grant (4132057), National Natural Science Foundation of China under Grant 61201159, Beijing Municipal Education Commission on Projects (SQKM201510016013), and Foundation of MOHURD (2015-K8-029).

Bibliography

[1] J. Yick, B. Mukherjee, D. Ghosal (2008); Wireless sensor network survey, *Computer Networks*, 52(12): 2292-2330.

[2] A. Mainwaming, J. Polastre, R. Szewczyk, D. Culler, J. Anderson (2002); Wireless sensor networks for habitat monitoring, *Proc. of ACM International Workshop on wireless serlsor Networks and Applications*, 88-97.

[3] G. J. Pottie, W. J. Kaiser (2000); Wireless Integrated Network Sensors, *Communication of the ACM*, 43(5): 51-58.

[4] C. Rotariu, H. Costin, I. Alexa, G. Andruseac, V. Manta, B. Mustata (2010); E-Health System for Medical Telesurveillance of Chronic Patients, *International Journal of Computers Communications & Control*, 5(5): 900-909.

[5] M. E. J. Newman, D. J. Watts (1999) Renormalization Group Analysis of the Small-World Network Model, *Physics Letters A*, 263(4): 341-346.

[6] A. L. Barabasi, R. Albert (1999); Emergence of scaling in random networks, *Science*, 286(5439): 509-512.

[7] S. Lindsey. C. S. Raghavendra (2002); Pegasis: Power-Efficient gathering in sensor information systems, *Proc of the IEEE Aerospace Conf*, 18(4):305-314.

[8] H. Tan (2003); Power efficient data gathering and aggregation in wireless sensor networks, *Acm Sigmod Record*, 32(4): 66 - 71.

[9] X. Y. Li, P. Wan, Y Wang, C. W. Yi (2003); Fault tolerant deployment and topology control in wireless networks, *Proceedings of the Fourth Acm Symposium on Mobile Ad Hoc Networking and Computing*, 117-128.

[10] T. Bernd, M. Heinrich (2005); Topology control for fault tolerant communication in highly dynamic wireless networks, *Proceedings of the 3rd International Workshop on Intelligent Solutions in Embedded Systems*, 89-100.

[11] A. Kashyap, S. Khuller, M Shayman (2006); Relay Placement for Higher Order Connectivity in Wireless Sensor Networks, *Infocom IEEE International Conference on Computer Communications*, 1-12.

[12] W. R. Heinzelman, A. Chandrakasan, H. Balakrishnan (2000); Energy-efficient communication protocol for wireless microsensor networks, *System Sciences Proceedings of Annual Hawaii International Conference on*, DOI: 10.1109/HICSS.2000.926982.

[13] S. Bandyopadhyay, E. J. Coyle (2003); An energy efficient hierarchical clustering algorithm for wireless sensor networks, *In Proc. of IEEE INFOCOM*, 1713 - 1723.

[14] O. Younis, S. Member, S. Fahmy (2004); HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks, *IEEE Trans. Mobile Computing*, 366–379.

[15] L. J. Chen, M. Liu, D. X. Chen, L. Xie (2009); Topology evolution of wireless sensor networks among cluster heads by random walkers, *Chinese journal of computers*, 32(1): 69-76.

[16] H. Zhu, H. Luo, H. Peng, L. Li, Q. Luo (2009); Complex networks-based energy-efficient evolution model for wireless sensor networks, *Chaos Solitons and Fractals the Interdisciplinary Journal of Nonlinear Science and Nonequilibrium and Complex Phenomenal*, 41(4): 1828-1835.

[17] X. Qi, S. Ma, G. Zheng (2011); Topology Evolution of Wireless Sensor Networks Basedon Adaptive Free-scale Networks, *Journal of Information and Computational Science*, 8(3): 467-475.

[18] Y. Q. Wang, X. Y. Yang (2012); Study on a model of topology evolution of wireless sensor networks among cluster heads and its immunization, *Acta Physica Sinica*, 2012, 61(9): 1321-1323.

[19] X. Luo, H. Yu, X. Wang. Energy-Aware Topology Evolution Model with Linkand Node Deletion in Wireless Sensor Networks, *Mathematical Problems in Engineering*, 55(1): 256-267.

[20] A. Barabasi, R. Albert, H. Jeong (1999); Mean-field theory for scale-free random networks, *Physica A Statistical Mechanics and Its Applications*, 272: 173-187.

[21] S. Madden, M. J. Franklin, J. M. Hellerstein (2002); TAG: A Tiny Aggregation Service for Ad-Hoc Sensor Networks, *Proceedings of the Usenix Symposium on Operating Sysems Design & Implementation*, 14-22.

[22] M. Hussaini, H. Bello-Salau, A. F. Salami, F. Anwar, A. H. Abdalla (2012); Enhanced clustering routing protocol for power-efficient gathering in wireless sensor network, *International Journal of Communication Networks and Information Security*, 18-28.

Author index