The Use of Values for Modeling Social Agents

Conference Paper · August 2017

CITATIONS
0

READS
95

3 authors:

Rijk Mercuur
Delft University of Technology
6 PUBLICATIONS **5** CITATIONS
SEE PROFILE

Virginia Dignum
Delft University of Technology
312 PUBLICATIONS **3,552** CITATIONS
SEE PROFILE

Catholijn M. Jonker
Delft University of Technology
543 PUBLICATIONS **6,380** CITATIONS
SEE PROFILE

The Proceedings of the 3nd International Workshop on Smart Simulation and Modelling for Complex Systems

Quan Bai, Fenghui Ren, Minjie Zhang, Takayuki Ito (Eds.)

Joint with the 26rd International Conference on Artificial Intelligence (IJCAI 2017)

Melbourne, Australia, August 21, 2017

Preface

This volume contains the papers presented at SSMCS2017: The 3nd International Workshop on Smart Simulation and Modelling for Complex Systems held on August 21, 2017 in Melbourne Australia.

Computer based modelling and simulation has become useful tools to facilitate humans to understand systems in different domains, such as physics, astrophysics, chemistry, biology, economics, engineering and social science. A complex system is featured with a large number of interacting components (agents, processes, etc.), whose aggregate activities are nonlinear and self-organized. Complex systems are hard to be simulated or modelled by using traditional computational approaches due to complex relationships among system components, distributed features of resources, and dynamics of environments. Meanwhile, smart systems such as multi-agent systems have demonstrated advantages and great potentials in modelling and simulating complex systems. The 3^{rd} International Workshop on Smart Simulation and Modelling for Complex Systems (SSMCS2017) aims to bring together researchers in both artificial intelligence and system modelling/simulation to discuss research challenges and cutting edge techniques in smart simulation and modelling.

The topics of SSMCS include but are not limited to:

- Agent based simulation for complex systems
- Agent based modelling for complex systems
- Large-scale simulations
- Network simulation and modelling
- Environment and ecosystem simulation and modelling
- Smart Grid/Service simulation and modelling
- Simulation of social and economic organizations
- Simulation of social complexity
- Cooperation, coordination, negotiation and self-organisation in complex systems
- Market-based model and simulation
- Transportation model and simulation
- Crowd model and simulation
- Evacuation model and simulation
- Human behaviour modelling, learning and simulations

Special thanks to all PC members and paper reviewers for their valuable contrbiutions to the workshop. We also thank all supports from all authors and workshop participators.

August, 2017

SSMCS2017 Organisation Committee
Dr. Quan Bai
Dr. Fenghui Ren
Prof. Minjie Zhang
Prof. Takayuki Ito

Table of Contents

Program Committee

Quan Bai	Auckland University of Technology
Fenghui Ren	University of Wollongong
Minjie Zhang	University of Wollongong
Takayuki Ito	Nagoya Institute of Technology
Guandong Xu	University of Technology Sydney
Shantanu Chakraborty	NEC Coporation
Kanamori Ryo	Nagoya Univesity
Xudong Luo	Sun Yat-sen University
Dayong Ye	Swinburne University of Technology
Xing Su	Beijing University of Technology
Iván Marsa Maestre	University of Alcala
Naoki Fukuta	Shizuoka University
Katsuhide Fujita	Tokyo University of Agriculture and Technology
Qing Liu	CSIRO
Xuyun Zhang	University of Auckland
Chao Yu	Dalian University of Technology

Application of particle filtering on short-term urban link travel time prediction with probe vehicle data

Ruotian Tang[1,*], Ryo Kanamori[2], Toshiyuki Yamamoto[3]

[1]Graduate School of Civil Engineering, Nagoya University, Japan

[2]Institute of Innovation for Future Society, Nagoya University, Japan

[3]Institute of Materials and Systems for Sustainability, Nagoya University, Japan

[*]Corresponding author: tang.ruotian@a.mbox.nagoya-u.ac.jp

Abstract. This paper applied particle filtering (PF) method to predict urban link travel time in short term. A multistep non-parametric model using probe vehicle data was developed. In order to evaluate performances of the proposed model, K-nearest neighbour (KNN) method was used as a comparison. Both methods were tested in two scenarios. One is real case where probe data was collected by taxi in Nagoya, Japan. Since the taxi only accounts for a small part of total traffic volume, the other scenario is simulated by computer, where all vehicles are treated as probe cars. Results show that PF has similar performance with KNN and is more computationally efficient. However, there is still a big room for improvement because of the complex traffic condition in urban network, such as the change of traffic signal.

Keywords: travel time prediction, urban network, short-term, particle filtering, taxi probe data

1 Introduction

Short-term travel time prediction in urban area is critical for today's intelligent transportation system (ITS) because of a rising real-time de-

mand for reliability and safety. Besides the ITS, recently there has been a similar requirement for short-term traffic forecasting to continuously update the traffic characteristics of the pre-built digital map in driverless vehicle system [6]. Therefore, short-term traffic forecasting, whose predicting range varies from seconds to even hours, has become more and more important in many areas of transportation research [7]. Many efforts have been made to predict travel time with traditional data collected from fixed devices along the highway or arterial road. Because of its low coverage and high initial expense, it is difficult to apply traditional data to forecasting on urban network. Moreover, this kind of data has a disadvantage of time-lag so it can't catch up with the real-time traffic state.

With the development of information technology, probe data which is collected from vehicles (e.g. taxi, bus, ambulance) equipped with on-board unit has become more and more accessible. Probe data has many advantages over traditional data, such as low cost, real-time update and wide coverage. Many studies have been done to estimate and predict traffic characteristics like travel time with probe vehicle data by parametric models [2,4,8]. However, researches [7] showed that non-parametric models outperformed parametric models under complex, dynamic and non-linear condition and it is relatively easy for non-parametric models to extend from one application to another. For example, the non-explicit state-transition model using particle filtering [1] predicted travel time by using historical trends to model the state-transition trend instead of discussing the specific traffic state and estimating the parameters. The k-nearest neighbour (KNN) method [3] utilized similarity between current traffic state and historical traffic state to forecast traffic flow rate.

So far, most of researches using non-parametric models with probe vehicle data have focused on highway. In our research, we are aimed to apply non-parametric model to urban link travel time prediction. We particularly concentrated on application of PF in this proposed model with probe vehicle data collected from taxi in Nagoya City, Japan. To evaluate the performance of proposed model, we compared it with the well-developed KNN method.

The paper is organized as follows. Section 2 introduces the basic concept of PF and presents the short-term travel time prediction methodology. Section 3 describes the data used in this research. The performance of pro-

posed model is presented in comparison with the KNN method in Section 4. The last section provides the conclusions of this paper and future work.

2 Methodology

The PF is derived from a perspective of Bayesian filter which consists of two stages: prediction (Eq.2.1) and update (Eq.2.2). Bayesian filter is a recursive probabilistic approach to estimate the posterior density function (Eq.2.2) of a target state variable x_t using the given past measurement z_t: $\{x_1, x_2 \dots x_t\}$ at each time point t. [5]

$$p(x_t|z_{t-1}) = \int p(x_t|x_{t-1})p(x_{t-1}|z_{t-1})dx_{t-1} \tag{2.1}$$

$$p(x_t|z_t) = \frac{p(z_t|x_t)p(x_t|z_{t-1})}{p(z_t|z_{t-1})} \tag{2.2}$$

The key of PF is to replace the required posterior density function with a set of random particles with associated weights $w_t^{(i)}$ (Eq.2.3) and to compute estimations based on these particles and weights. The weight is updated using Eq.2.4. Relationship in Eq.2.4 can further simplifies to Eq.2.5, if using sampling importance resampling (SIR) filter which uses the transitional prior pdf $p(x_t^{(i)}|x_{t-1}^{(i)})$ as the importance density $q(x_t^{(i)}|x_{t-1}^{(i)}, z_t)$ and resamples at each step. [5]

$$p(x_t|z_t) \approx \sum_{i=1}^{N} w_t^{(i)} \delta(x_t - x_t^{(i)}) \tag{2.3}$$

$$w_t^{(i)} \propto w_{t-1}^{(i)} \cdot \frac{p\left(z_t \middle| x_t^{(i)}\right) p(x_t^{(i)}|x_{t-1}^{(i)})}{q(x_t^{(i)}|x_{t-1}^{(i)}, z_t)} \tag{2.4}$$

$$w_t^{(i)} \propto p\left(z_t \middle| x_t^{(i)}\right) = p(z_t - z_t^{(i)}) \tag{2.5}$$

$\delta(\cdot)$ represents the Dirac delta measure, $p(z_t - z_t^{(i)})$ represents the similarity between a prediction $z_t^{(i)}$ drawn from historical data and an objective measurement data z_t. There are several methods to define $p(z_t - z_t^{(i)})$, such as normal distribution [1]. The definition of $p(z_t - z_t^{(i)})$ in proposed model is shown by Eq.2.6, where $\sum_{j=1}^{t-1} \alpha_j = 1$ and ΔT_j is the time in-

terval between time point *j+1* and *j*. Since the taxi probe data is not abundant enough, it is difficult to find particles with exactly the same time interval. $\sum_{j=1}^{t-1} \alpha_j \left|\Delta T_j - \Delta T_j^{(i)}\right|$ represents the differences of time intervals and $\sqrt{\sum_{j=1}^{t-1}\left(x_j - x_j^{(i)}\right)^2}$) represents the Euclidean distance between the real measurement and particle.

$$p(z_t - z_t^{(i)}) = 1/((\sum_{j=1}^{t-1} \alpha_j \left|\Delta T_j - \Delta T_j^{(i)}\right|) * \sqrt{\sum_{j=1}^{t-1}\left(x_j - x_j^{(i)}\right)^2}) \quad (2.6)$$

The proposed model used the same main structure as Chen and Rakha [1] used in their research, which consists of three parts—updating, resampling and predicting. The algorithm is shown in Table 2.1. First of all, the time window which is the scale of a particle is set up and a database of particles is built from historical data (Line 1~4). Here, the particle is a continuous temporal sequential measurement of travel time. In order to increase the number of particles a buffer time is added to the standard time interval (Line 2). The whole prediction starts when there is an observed measurement data (Line 4) and ends when it reaches the prediction horizon. In updating process, the time window is shifted one step ahead and if there is an observed real-time data, the measurement data is replaced by real-time data instead of the prediction in the last step (Line 8~12). Then, *N* new candidates are generated randomly from the particle database and their weights are calculated by Eq.2.6 (Line 13~14). Since the candidate is randomly picked up from the database, it is inevitable that candidate with relatively low weight, which should be replaced, will affect the accuracy of prediction. In the resampling process, partial of candidates with higher weight are kept (Line 16~20). Particles with maximum similarity of each historical day that higher weight candidates belong to are selected and used to replace the remaining candidates with lower weight according to the selected particles' similarity (Line 21~24). The number of candidates that needs to be replaced is decided by the resampling rate which is set in advance (Line 5). In predicting process, all predicted travel time candidates $x_t^{(i)}$ are summed up according to their weights and the weighted summation is used as new measurement data (Line 26).

Table 2.1. Multi-step particle filtering method for short-term travel time prediction

1.	Initialize the width of time window *t*, build particle database
2.	$\{(x_t^{(p)}, z_t^{(p)})_{d^{(p)}} \mid \Delta T_j \in (t_{int} - t_B, t_{int} + t_B)\}$, where
3.	$j \in \{1,2 \dots t-1\}, p \in \{1,2 \dots\}$, $d^{(p)}$=date, t_B= buffer time,
4.	t_{int}=standard time interval. Find the observed measurement data
5.	z_t and *l*=*t*. Set the resampling rate *K%*.
6.	
7.	Step 1: Updating
8.	Shift the time window
9.	For *i=2:t*,
10.	$x_{i-1} = x_i$.
11.	End For
12.	If there is an observed data x_l, $x_{t-1} = x_l$
13.	Draw *N* candidates $\{(x_t^{(i)}, z_t^{(i)})_{d^{(i)}}\}_{i=1}^{N}$from $\{(x_t^{(p)}, z_t^{(p)})_{d^{(p)}} \mid \Delta T_j \in$
14.	$(t_{int} - t_B, t_{int} + t_B)\}$ and calculate their weight $w_t^{(i)} = p(z_t - z_t^{(i)})$
15.	Step 2: Resampling
16.	Sort the candidates in decreasing order according to their weights
17.	For j $= 1: N \cdot (1 - K\%)$
18.	Pick up $\{(x_t^{(r)}, z_t^{(r)})_{d^{(r)}}, w_t^{(r)}\}$ from historical database,
19.	where, $z_t^{(r)} = argmax_{d^{(p)}=d^{(j)}} p(z_t - z_t^{(p)})$
20.	End For
21.	For j $= N \cdot (1 - K\%)$: N
22.	Select $(x_t^{(r)}, z_t^{(r)})_{d^{(r)}}$ according to its weight $w_t^{(r)}$,
23.	$(x_t^{(j)}, z_t^{(j)})_{d^{(j)}} = (x_t^{(r)}, z_t^{(r)})_{d^{(r)}}, w_t^{(j)} = w_t^{(r)}$
24.	End For
25.	Step 3: Predicting
26.	*l=l+1*; $\widetilde{x_l} = x_t = \sum_{i=1}^{N} w_t^{(i)} \cdot x_t^{(i)} / \sum_{i=1}^{N} w_t^{(i)}$

3 Data description

The data used in this research was collected by taxi in Nagoya, Japan in February and June, 2015 (58 days in total). Since taxi only accounts for a small part of total traffic volume, not all the links have enough data. Even though we selected links with relatively large amount of data, it is still far from enough. Therefore, a simulation at a normal cross intersection with four 200-meters-long links was constructed by VISSIM (version 7.0).

3.1 Data in real case

Each data represents a taxi go through a link, including information such as the time point when a taxi entered and exited a link, the number and length of the link, the number of next link, the longitude and latitude of end point (node) for each link. Data which implies the taxi stopped in the link to pick up or deliver passengers was discarded. Consequently, each link's travel time includes the delay at the downstream intersection.

Vehicles turning left or right have to give way to vehicles going straight which are usually the main component of the total traffic [2], so only vehicles going straight were considered in proposed model. Moreover, travel time for vehicles only going straight was predicted in order to avoid complexity. Since we restrained the focus on vehicles going straight which have a priority when going through the non-signalized intersection, roads separated by non-signalized intersection are connected as one link which only has signalized intersections at its end-points.

One link with relatively large amount of data was selected from the whole network in this research. It is a south-north 223-meters-long link belonging to Ootsu Street located in downtown. Its sample size is 47001 and average travel time is 64.4s. Even though the selected link has more data than most of other links in the network, the amount of data is still far from enough. The ratio of going straight among all turning choices is 57%. The changes of average number and travel time of taxis going straight per hour during one day in February and June are similar as illustrated in Fig. 3.1.1. As for the signal timing, the length of green phase is about 71s, while the length of red phase is about 76s.

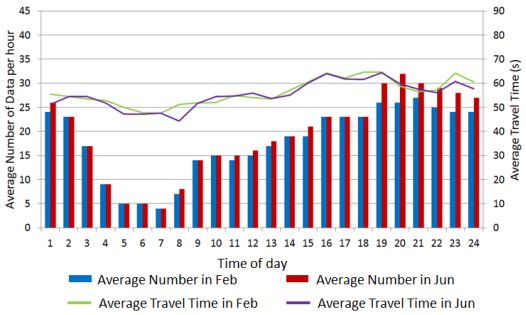


Fig. 3.1.1 Sample distribution and travel time by time of day

3.2 Data in simulation

For simplicity, vehicles in all links only go straight at the intersection and the initial speed varies from 10 to 50km/h randomly. The duration for one simulation is 1 hour and it repeated for 30 times under the same setting. We used one of them as testing data and the remaining 29 simulations make up the historical database. Concerning the traffic volume, since we only focused on one link whose volume is 200, 600 and 100veh/hour in the first 10 minutes, second 10 minutes and third 10 minutes respectively, and this pattern repeats in the later 30 minutes. For the other three links, traffic volume is fixed at 200veh/hour. All the vehicles are delivered randomly based on the default setting. The traffic signal is designed as two phases without the all red phase and the length of each phase is 60 seconds.

Link travel time was recorded, including the stopping time at the intersection. Data with travel time over 170s was discarded. The number of data in testing database is 295, while the number of data in historical database is 8568. The average link travel time in testing database is 90.5s, while in historical database is 85.7s.

4 Case study

First of all, the scale of particle in this research should be determined. As we used the tendency of travel time change, there should be at least three points in a particle. Although we didn't directly reflect the influence of traffic signal on travel time which contained the stopping time at the in-

tersection, it should be taken into consideration. The exit time of a link was used as time point and the standard time interval between each point was 60s with a 15-seconds-long buffer time, which is close to the signal timing, so that it may reflect the influence of signal timing. In addition, the prediction is aimed to apply to traffic control and management system so it is not necessary to predict link travel time before a vehicle enters a link. The number of particles and its corresponding scale in real case is shown in Table 4.1. Even though there are 47001 data collected from different time points, the number of useful particles is limited. We chose the scale of particle which has the largest number of particles. In another word, travel time of the first two time points were used to forecast the third one.

Table 4.1. Number of particles in real case for PF

Scale of particle	3	4	5	6	7
No. of particles	299	64	11	3	1

A KNN method was used to compare with the proposed model. In KNN, the same way of calculating similarity between real-time data and historical data was applied by Eq.4.1. The difference between KNN and PF is that KNN can predict the travel time several intervals later directly as long as there is corresponding historical data by Eq.4.2, while PF has to predict step by step.

$$w_t^{(i)} = 1/((\sum_{j=1}^{t} \alpha_j \left| \Delta T_j - \Delta T_j^{(i)} \right|) * \sqrt{\sum_{j=1}^{t} \left(x_j - x_j^{(i)} \right)^2}) \qquad (4.1)$$

$$\widetilde{x_{t+m}} = \sum_{i=1}^{k} w_t^{(i)} \cdot x_{t+m}^{(i)} / \sum_{i=1}^{k} w_t^{(i)} \qquad (4.2)$$

where *k* is the number of candidates, *m* is the prediction horizon.

The sample size in real case and its corresponding prediction horizon is shown in Table 4.2. Sample's prediction horizon should have been integral time but samples with the same buffer time as PF was also considered in order to increase the sample size.

Table 4.2. Sample size for KNN in real case

Prediction horizon (min)	1	2	3	4	5
Sample size	299	585	386	337	685

As mentioned above, travel time of two continuous time points were used to calculate the weight, so we need to determine *k*, α_0 and α_1 ($\alpha_0 + \alpha_1 = 1$). The mean absolute percentage error (MAPE) for KNN under dif-

ferent combination of k and α_0 is shown in Fig.4.1. It is difficult to identify an optimal combination of k and α_0. We assumed that each time interval has the same influence on the prediction, so α_0 was chosen as 0.5. In general, accuracy became better when k is around 7.

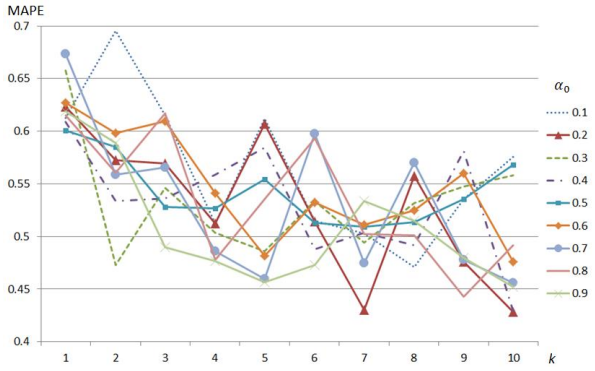


Fig. 4.1 MAPE for KNN

As for PF in proposed model, the same value of α_0 was used but we still have to determine the value of resampling rate and number of candidates. As shown in Fig.4.2, accuracy became better when number of candidates is around 25 and resampling rate is about 30%.

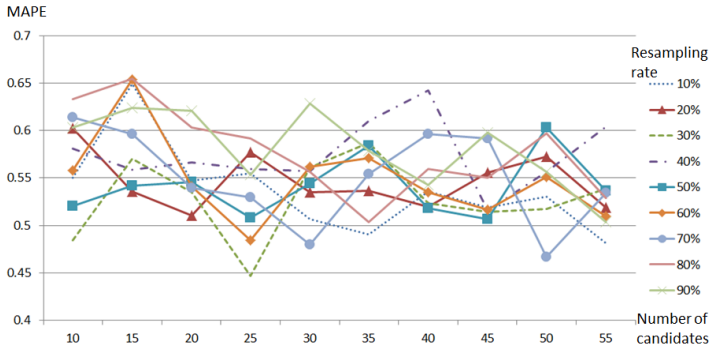


Fig. 4.2 MAPE for PF

For both PF and KNN, a piece of data was selected randomly as testing (real-time) data and the remaining data was used as historical data. In the sake of simplicity, the maximum prediction horizon was chosen as 5min, the computation time is also compared between two methods. The program was built by C++ and run by computer with 2.5 GHz Intel Core i7 processer and 16 GB 1600 MHz DDR3 memory. The results for both methods in real case are shown in Table 4.3.

Table 4.3 Results for both methods in real case

Prediction horizon (min)	1	2	3	4	5
Computation time for PF (10^{-3}s)	2.13	2.25	2.38	2.5	2.75
Computation time for KNN (10^{-3}s)	4.25	10.5	5.63	4.88	13.4
MAPE for PF (%)	54.5	57.3	91.5	49.6	65.6
MAPE for KNN (%)	51.4	61.4	60.8	50.4	63.3

In general, KNN is more accurate than PF but needs more computation time. The computation time of PF is mainly affected by prediction horizon, while the computation time of KNN is mainly affected by the sample size. Even though neither KNN's nor PF's performance is satisfactory, there is still a big room for improvement because the traffic condition in urban link is more complex than highway. In addition, another main reason for the poor performance is that the data can't represent all vehicles on the road because it was only collected by taxi. In order to solve this problem, a simulation was constructed where all vehicles function as probe vehicle. In the simulation, we still used particles with three continuous time points but the buffer time decreased to 2.5s, which means the time interval was nearly 60s. The sample size and its corresponding prediction horizon in simulation are shown in Table.4.4. Since PF and KNN used the same data to build historical database when prediction horizon is 1min, the number of particles was 163. The results for both methods in simulation are shown in Table 4.5. As for computation time, the prediction horizon is the main factor that affects PF's computation time, while scale of database determines the computation time for KNN. PF needs less computation time than KNN. The accuracy of PF deteriorated when prediction horizon increased because it can't catch the change during the long time interval beyond the scale of particle, but its performance was still at the same level with KNN. If both PF and KNN have similar scale of database, it needs less time than KNN, which means PF is more suitable for practical application.

Table 4.4 Sample size for KNN in simulation

Prediction horizon (min)	1	2	3	4	5
Sample size	163	175	158	173	157

Table 4.5 Results for both methods in simulation.

Prediction horizon (min)	1	2	3	4	5
Computation time for PF (10^{-3}s)	0.8	1	1.12	1.25	1.38

Computation time for KNN (10^{-3}s)	1.5	1.6	1.38	1.5	1.5
MAPE for PF (%)	18.4	34.4	34.7	41.8	48.2
MAPE for KNN (%)	16.8	29.2	34.9	48.5	47.5

5 Conclusions

Previous researches used probe data and data-driven models to predict freeway travel time, while we focused on urban link travel time prediction. Compared with freeway, traffic condition in urban link is much more complex because of traffic signal and other unexpected factors along the road. Since we used a data-driven model, it didn't reflect the influence of traffic signal on travel time directly. In the proposed model, we used a particle filtering (PF) method which can predict travel time at multiple steps. The proposed model used the same main structure as Chen and Rakha [1] did, but we added a buffer time to solve the problem of lack of particles and therefore used a different way to calculate similarity. A k-nearest neighbour (KNN) method was applied to make a comparison. We found that if prediction is only required at the next interval and the same calculation of similarity is used, KNN is quite similar with PF. In this case, the only difference is KNN searches among the whole historical database to find qualified candidates, while PF randomly selects candidates and resamples partial of them to ensure the quality of candidates. In another word, PF can be viewed as the combination of several simplified KNN methods which searches part of its historical database. That's why both methods have similar performance. However, PF only needs one database because the scale of particle is determined, while KNN has to change database when prediction horizon changes. It is easier for PF to expand the prediction horizon and make continuous prediction. Both methods were tested using real data and simulated data. In real case, neither KNN nor PF performed well because of traffic signal and fragmentary data. Performance of both methods in simulation was better than real case because all vehicles in the link were treated as probe vehicles. With the expansion of prediction horizon, the accuracy of PF deteriorates because it can't track the continuous change of travel time beyond the scale of particle. Generally speaking, PF has similar performance with KNN but it is more extensible and computationally efficient. Accuracy for both methods can be improved as long as the influence of traffic signal can be fully reflected and PF has an advantage over KNN in practical application.

There is plenty of future work waiting for us. The most urgent task is to take traffic signal into consideration. It may be difficult to reflect the influence of traffic signal only by non-parametric model, so we plan to combine the explicit model with non-parametric model. We will further utilize the information form crossing vehicles at the same downstream intersection to estimate the signal timing. Interactions between vehicles in the same traffic flow will also be used to improve model's interpretation capability. Since the PF in the proposed model is similar with KNN, it didn't outperform KNN concerning accuracy. Thus, we are going to keep the "predicting-resampling" structure but change the method of sampling and resampling. After that, the improved model should be tested on a small urban network. Last but not least, except for KNN, other methods should be used to compare with the proposed method.

Acknowledgments

This research was supported by Grant-in-Aid for Scientific Research (No. 26220906) from the Ministry of Education, Culture, Sports, Science and Technology, Japan (MEXT) and the Japan Society for the Promotion of Science (JSPS).

References

1. Chen, H., Rakha, H.A., 2014. Real-time travel time prediction using particle filtering with a non-explicit state-transition model, *Transportation Research Part C*, 43, 112-126.
2. Feng, Y., Hourdos, J., Davis, G.A., 2014. Probe vehicle based real-time traffic monitoring on urban roadways, *Transportation Research Part C*, 40, 160-178.
3. Habtemichael, G., Cetin, M., 2016. Short-term traffic flow rate forecasting based on identifying similar traffic patterns, *Transportation Research Part C*, 66, 61-78.
4. Hiribarren, G., Herrera, J.C., 2014. Real time traffic states estimation on arterials based on trajectory data, *Transportation Research Part B*, 69, 19-30.
5. Ristic, B., Arulampalam, S., Gordon, N., (2004): Beyond the Kalman Filter: Particle Filters for Tracking Applications, Artech House, p. 1-65.
6. Surden, H., Williams, M.A., (2016): How self-driving cars work, available at SSRN: https://ssrn.com/abstract=2784465.
7. Vlahogianni, E.I., Karlaftis, M.G., Golias, J.C., (2014): Short-term traffic forecasting: Where we are and where we're going, *Transportation Research Part C*, 43, 3-19.
8. Zheng, F., Zuylen, H.V., 2013. Urban link travel time estimation based on sparse probe vehicle data, *Transportation Research Part C*, 31, 145-157.

The Use of Values for Modeling Social Agents

Comparing a Value-based Agent to a Learning *Homo Economicus* Agent in the Context of the Ultimatum Game

Rijk Mercuur, Virginia Dignum and Catholijn Jonker

Abstract This paper aims to contribute to the development of realistic social agents by investigating the relevance of human values, such as privacy, wealth and fairness, in decision making. The concrete question studied in this paper is whether human behavior in the Ultimatum Game (UG) is better modeled with a value-based agent or with a learning *homo economicus* (LHE) agent. Based on requirements extracted from human UG play we model a value-based and LHE agent. We show that with the value-based agent one can better explain various UG scenarios than with the LHE agent. We look in particular at scenarios that vary in the amount of money that is divided. As the value-based agent does not reproduce the exact UG human behavior, we discuss the relevance and limitations of our results for modeling realistic social agents.

Key words: Social Agent, Human Values, Agent-based Simulations, Ultimatum Game

1 Introduction

In the past years a series of papers calls for more realistic social agents [20, 9, 8, 24]. Their central claim is that current agents lack the social aptitude to interact with humans and other agents in a realistic way. Moreover, people do not function as autistic agents filtered according to some 'social awareness' modules to pick socially acceptable actions. On the contrary, our actions are based in the core on social ingredients: e.g. human values, identity, norms and culture. In this paper we investigate the relevance of human values in decision making. Human values can be defined as 'what a person finds important in life', e.g. privacy, wealth and fairness [14]. Values have been shown to correlate

Rijk Mercuur · Virginia Dignum
Faculty of Technology, Policy and Management
Delft University of Technology, The Netherlands
e-mail: r.a.mercuur@tudelft.nl

Virginia Dignum
e-mail: m.v.dignum@tudelft.nl

Catholijn Jonker
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology, The Netherlands
e-mail: c.m.jonker@tudelft.nl

to human actions in a lot of different contexts [22], but there is no consensus on if, and moreover how, values can be useful in modeling agents.

In this paper we provide a partial answer to the question of how informative values are for modeling realistic agents, by comparing the realism of a value-based agent to a learning *homo economicus* (LHE) agent for a tangible setting. The *homo economicus* (HE) agent is the canonical agent in game theory [23] and classical economics [18], that only cares about maximizing its direct own welfare, payoff or utility. The HE agent has been critiqued for not resembling humans in, among many other cases [19], an experiment called the Ultimatum Game (UG) [17]. In the UG two players negotiate over a fixed amount of money ('the pie'). Player 1, the *proposer*, demands a portion of the pie, with the remainder offered to Player 2. Player 2, the *responder*, can choose to accept or reject this proposed split. If the responder chooses to 'accept' the proposed split is implemented. If the responder chooses to 'reject' both players get no money. The HE agent, caring only about its direct own welfare, would accept any positive offer. In contract, empirical research shows that human responders reject offers as high as 40% of the pie [25].

One approach to explaining these findings is by extending the HE agent model to incorporate learning [15, 27]. The core of this explanation is that humans have learned through the feedback of repeated interaction to reject low offers to force the proposer into making higher offers. In this view, humans resemble a LHE agent for which, roughly said, fairness only exist as an instrument for wealth. In this paper we compare such an LHE agent to an agent that uses human values in its decision making. This value-based agent does not learn, but cares about both wealth and fairness. We evaluate these agents on their ability to explain human behavior in the setting of the Ultimatum Game (UG) [17] for which extensive data on human behavior is available. We say an agent model explains human behavior when it resembles human behavior both on the micro as macro level [4, 10].

We approach this problem thus by first reviewing literature on values and the UG in section 2, as to extract realistic micro and macro requirements, to which our models should adhere. Section 3 explicitly lists these requirements and describes *possible* models for the LHE and value-based agent that adhere to (most of) these requirements. Section 4 describes the simulation experiments we use to compare the agent models on their power to explain variations of the UG; specifically UGs varying in pie size. Let us emphasize that we need these simulation experiments as inferring the aggregate behavior for the model specifications is not trivial. First, because the matching of proposer and responder influences the results. Second, because the interaction of the agents influences the learning of the LHE agent. We need agent-based simulations to properly randomize and simulate this matching and interacting, such that, we can compare the resulting aggregate behavior against human behavior. We find that the value-based agent outperforms the LHE agent in both the standard UG as in UGs varying in pie size, but that even the value-based agent cannot produce exact human play. We conclude this paper by discussing what these UG specific results mean for the value of value-based decision making in modeling social agents.

2 Background

In this section we will describe some of the literature that will be useful in extracting requirements for our model. Subsection 2.1 on values is mainly relevant in designing the value-based agent. Subsection 2.2 on UG research is relevant for testing the models against human play.

2.1 Values

Values are notoriously hard to capture. Most scholars seem to agree that values represent 'what a person finds important in life' and function as 'guiding principles in behavior'. In the remainder of this subsection we will describe some of the work on values in psychology, sociology and philosophy.

In the area of psychology Schwartz developed several instruments (e.g. surveys) to measure and categorize values [28]. With these instruments Schwartz shows that, in general, people who give positive answers to survey questions on wealth are more likely to give negative answers to survey questions on fairness. These findings on intervalue comparison have been extensively empirically tested and shown to be consistent across 82 nations representing various age, cultural and religious groups [28, 29, 3, 6, 13]. These measurement on values seem to correlate with actions. Recent work in sociology [22] uses data from the European Social Survey to show that values predict 15 different measured actions over six behavioral domains and in every country included in the study. In line with many other empirical studies (see [16, p. 545-546] for an overview) Miles shows that the value action correlation is consistent, but weak and highly depending on other moderating and mediating variables. They seem to be the abstract fixed points that actions over many context can be traced back to.

In philosophy values are also seen as a starting point, i.e. that what is good in itself or for its own sake. This makes it difficult to make a choice when two different options are selected by two different values as best, i.e. a value conflict [26, p.177-190]. There are different ways to resolve a value conflict, for instance by doing a 'multi-criteria analysis' and by upholding thresholds. In multi-criteria analysis the different options are weighted on the values and compared on a common measure; in threshold comparison an option is good as long as both values are promoted above a certain threshold. If one option upholds both thresholds, while the other one does not the former is chosen.

In arguably the most popular social psychology approach to predicting behavior, the Reasoned Action Approach [12], values are seen as *pre-action* triggers of intentions (that in turn bring about actions). In [21, 31] values are seen as *post-action* standards or criteria to evaluate actions on through which an agent learns and navigates its deliberation. In [1, 2] the authors, just as in this study, use values in the context of the UG. They provide a high level analytical qualitative analyses of how agents could *argue* in favor of all kind of different actions. Our work differs in that we quantitatively compare a value-based agent and LHE agent actions (not arguments) to those of humans. Moreover, we study different scenarios and due to simulating the match up of the agents we can study if individual models lead to realistic aggregate results.

2.2 Ultimatum Game

The Ultimatum Game (UG) has been the subject of many experimental studies since it first appearance in [17]. In [25] the authors compare 75 results of UG experiments and find that on average the proposers demand 60% of the pie and on average 16% of the demands are rejected. In this paper we will look in particular at UG studies varying in pie size. As one of the most consistent results in UG research is that the pie size correlates positively with the size of the demand and negatively with the rejection rate [5, 25].As a result of this extensive studies the UG has been a testbed to design other-regarding preferences [5, 7]. Although the aim of these studies is similar to ours they do not see the criteria they evaluate actions on as human values. In section 5 we will further discuss the relevance of seeing criteria as values.

3 Model

In this section we aim to construct a value-based agent and a LHE agent to model UG behavior. We first describe the formal UG scenario and the requirements following from empirical human UG play. Both agents need to adhere to these requirements as both agents need to be able to reproduce UG play. In subsection 3.2 we describe the requirements for our value-based agent and a formal model that adheres to them. In subsection 3.3 we do the same for the LHE agent.

3.1 Description of Target Scenario

The scenario comprises a one-shot UG as described in the introduction with the following specifics:

- The amount of players N is 32.
- The amount of rounds Z is 1.
- The pie size P is 1000.[1]

We choose this scenario as it represents the average scenario as analyzed in the meta-study of Oosterbeek et al. [25]. We filtered the dataset to only represent data on the first round. Although the differences in first round behavior and subsequent rounds are small, we do not wish to represent them with our data. In the remainder of this paper we will refer to the dataset of Oosterbeek et al. - filtered on the first round - as our adapted dataset.[2]

The agent needs to be able to replicate the following facts that represent average scores in our adapted dataset:

UG.1 On average the agents need to demand 59.2% of the pie when in the role of the proposer,
UG.2 On average the agents need to reject 14.9% of the demands when in the role of the responder.
UG.3 The agents are heterogeneous on their demand and accept rate. The demands are roughly natural distributed, but with no demands under 50%.

3.2 Value-based Agent

In this subsection we aim to model a value-based agent that reproduces UG play (i.e. **UG.1**, **UG.2** and **UG.3**). In addition to UG play we validate the value-based agent model on four extra requirements:

V.1 An agent considers both wealth and fairness as the primary determiner of its actions.
V.2 The higher one values wealth the higher the demands one makes (and expects). The higher one values fairness the more equal the demands one makes (and expects).
V.3 The values of wealth and fairness are negatively correlated.
V.4 Agents should be heterogeneous in the values they find important.

[1] For ease of presentation we choose to ascribe no monetary unit to the pie size. The average pie size in Oosterbeek et al. namely has no standard unit, but is 25 times the GDP of the subjects country.

[2] Due to space limitations we choose not to present the adapted dataset, but refer for this to the representation of the similar original dataset in [25].

Firstly, in this paper the purpose is to study the relation of relevant values to a decision. We thus focus on how humans determine their actions based on their values as opposed to how humans select which values are relevant or how these values evolve over time. We thus postulate (based on literature and our purpose) that the values of wealth and fairness alone are necessarily and sufficient to inform the decision of the value-based agent (**V.1**). Secondly, based on intuition we link the value of wealth to the amount of money obtained and the value of fairness to the equality of the split (**V.2**). Thirdly, from the literature on values we know when people are asked how much they value wealth these answers negatively correlate with how much they value fairness (**V.3**). Furthermore, people differ significantly in the answers they give (**V.4**) [29].

3.2.1 Formal Model & Validation

We postulate that the utility of an action is determined by two variables per value:

- i_v : the importance (or weight) one attributes to a value
- $s_v(r)$: a function that specifies how much a result r satisfies value v

As said our agents consider two values: wealth (w) and fairness (f) (**V.1**). The value weights i_w and i_f are normally distributed and thus differ per agent as required by **V.4**. We also assume a perfect negative correlation between the value weights and thus adhere to **V.3**. Releasing this assumption to a weaker negative correlation did not change our results significantly. As the two value weights now completely depend on each other we can simplify the model to have two parameters μ and σ that specify a normal distribution from which the *difference* (di) in value strengths is drawn, i.e. for every agent

$$i_w = 1.0 + 0.5di \tag{1}$$

and

$$i_f = 1.0 - 0.5di \tag{2}$$

such that, $di = i_w - i_f$, represents how much more an agent values wealth over fairness.
Furthermore we postulate the satisfaction function as follows:

$$s_w(r) = \frac{r}{1000} \tag{3}$$

$$s_f(r) = \frac{r - 0.5P}{P} \tag{4}$$

where r is the resulting money obtained from one round of UG-play and P is the pie size. The satisfaction of wealth thus increases as one gets more money and the satisfaction fairness peaks around an equal split. This allows the model to adhere to requirement **V.2**. Note that we chose to model the denominator as 1000 and not as P; the rationale is that we think the satisfaction of wealth increases absolutely and not relative to the pie size. In further work we should further explore empirical work to vouch for this modeling choices.

As said in the background section there are multiple ways to resolve value conflicts: i.e. determine which action is best when multiple values are at play. Using this available theory we postulate the following 'divide function' to determine the utility (u) of a result (r):

$$u(r) = -\frac{i_w}{s_w(r) + ds} - \frac{i_f}{s_f(r) + ds} \tag{5}$$

where ds is a small number (by default ds=0.5) to avoid division by zero.[3]

The proposer now demands that $d \in [0,P]$ for which the utility (as given by $u(r)$) is maximal. Note that the proposer thus assumes the demand d will be equal to the result r, i.e. it does not take in account the action of the responder. This is a simplification that seems likely in non-anonymous games, but deserves more attention in further work.The responder chooses to accept if (and only if) the utility of what it receives - $u(P-d)$ - is higher than the utility of a reject. We choose to model the utility of rejection by filling in function 5 with $s_w(0)$ and $s_f(0.5P)$, i.e. the agent interprets it as getting maximum fairness (as in the r= 0.5P case), but getting almost no wealth (as in the $r = 1$ case).

It is worth mentioning we tried several combinations of i and s as utility functions. Including a product function:

$$u(r) = i_w * s_w(r) + i_f * s_f(r) \tag{6}$$

and a difference function:

$$u(r) = i_w - s_w(r) + i_f - s_f(r) \tag{7}$$

In figure 1 we compare the three functions. All functions represent the fact that agents rarely demand below 50. Function 7 does not uphold requirement **V.2**: the value strength does not correlate with the demands. Function 6 does not uphold requirement **UG.3**: i.e. the demands will not be normally distributed. Function 5 is the only function of these three that reacts appropriately to a difference in values and upholds all requirements.

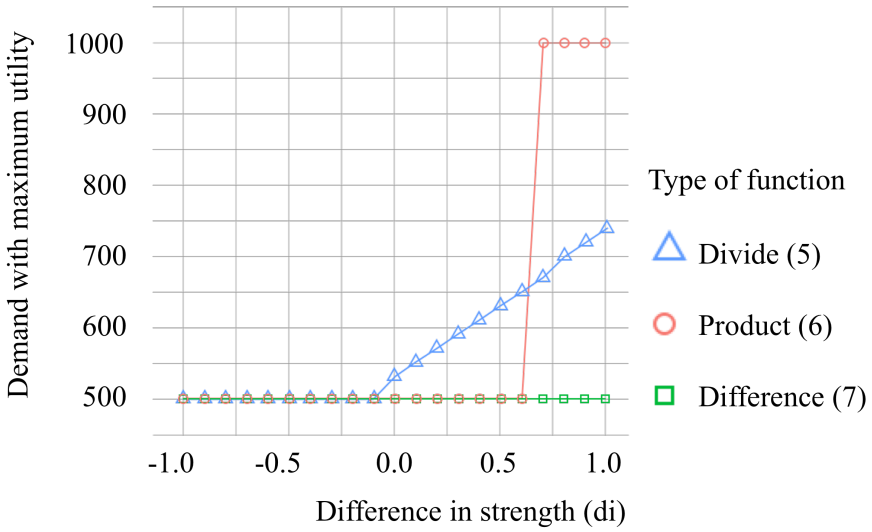


Fig. 1 Three value functions compared on what demand gives maximum utility (y-axis) for different value strengths (x-axis).

[3] In further work we need to study the necessity of the extra parameter ds and the added complexity following from it.

Lastly, we need to set the parameters of the model to tune it to **UG.1** and **UG.2**: an average demand of 59.2% and an average reject rate of 14.9%. We run 50 independent simulations for each $\mu \in [-2,2]$ and $\sigma \in [0,2]$ with stepsize 0.01 and check the performance of these parameter on the average demand and the average reject rate they result in. We find that for $\mu = 0.16$ and $\sigma = 0.73$ there is an average demand of 59.2% and a rejection rate of 14.9%. Under these parameters the model thus adheres to **UG.1** and **UG.2**.

Our model thus shows that the average humans demand in the UG can be sufficiently explained - while adhering to known literature and theory - by the proposition that people value wealth slightly higher than fairness. Furthermore, to explain a rejection rate of 15% people only need to (1) differ in the strength to which they adhere values and (2) be randomly matched so that these differences result in rejections of 'unfair' demands.

3.3 LHE agent

In this subsection we aim to model a LHE agent that reproduces UG play (i.e. **UG.1**, **UG.2** and **UG.3**). We take a different approach than in the previous subsection. Instead of extracting requirements from the literature we take an existing LHE agent and try to find the parameter settings under which it adheres to our UG play requirements. Note that the scenario we study comprises only one round, i.e. humans have no time to learn, but they still demand fair shares. The proponents of the LHE agent might counter that humans have learned to make fair demands in interactions similar to the UG and transpose the utility they attribute to such actions to the UG. In this section we model such an explanation by taking the agent introduced by Roth et al. [27] as exemplary of a such a LHE agent. In the next subsection we introduce the model and validate the model against **UG.1**, **UG.2** and **UG.3**.

3.3.1 Formal model & Validation

A simplification of the LHE agent model in [27] can be presented as follows:

1. At round $z = 1$, each player n attributes an initial utility i to each action $a \in A$, i.e. $u_n(z = 1, a) = i$.
2. The set of possible actions A is $\{0, 0.1P, 0.2P, \ldots, P\}$. For the proposer this number represents the demand it makes. For the responder this number represents a threshold; if the demand is above this threshold it will reject, if the demand is equal or below the threshold it will accept.
3. Each round an agent picks an action according to the distribution of these utilities, i.e. the probability H, to pick an action a, is defined by the following function $H(a) = \frac{u_n(z,a)}{\sum_{a \in A} u_n(z,a)}$.
4. When the round is finished each player n updates the utility u_n of the played action $\hat{a}$ by adding the obtained money r to the previous utility, i.e. $u_n(z+1, \hat{a}) = u_n(z, \hat{a}) + r$. The utility of the other actions remains the same.

There are two versions of the model that differ in their approach to the initial utilities. Before introducing them we first introduce the parameter $s(1)$, the initial strength of the model, defined as the sum of the initial utilities, i.e. $s(1) = \sum_{a_i \in A_i} u(a)$. The initial strength determines the initial learning speed of the agent. The two versions of the model are:

1. The initial utilities are all equal to each other i.e. u(a) = u(b) for all actions $a, b \in A$. (But $s(1)$ is free.)

2. The initial utilities sum to 500, i.e. s(1) = 500. (But are randomly distributed.)

In [11] the authors show that for many games data can be reproduced with the simple reinforcement learning agent introduced and equal initial utilities, but do not treat the UG in this paper. In [27] the authors show that crudely UG results can be reproduced with random utilities and a fixed strength of 500, but do not provide the exact parameter settings nor specifically compare the learned distributions to first round play. In this study, we choose to further explore the first model (with equal utilities) as the search space is much more manageable.

We continue our aim by trying to validate the first model against **UG.1**, **UG.2** and **UG.3**. The results presented are averages over 50 simulation runs taken from $z = 1000$ when demands and accept ratios stablized. In terms of the explanation, the LHE agent learns from $z = 1$ until $z = 999$ in UG like scenarios and then plays its first round at $z = 1000$.

We run simulations for $s \in [0.00005, 8]$ with a logarithmic stepsize as exploration learns that simulations outside this interval are similar to the bounds. We find that:

1. A negative exponential relation between s and the average demand. We can approximately match a demand of 59.2% (**UG.1**) for $s = 0.09$.
2. A negative exponential relation between s and the accept ratio. We find no s for which the reject ratio is 14.9% (**UG.2**). The closest accept ratio is 0.64 for $s = 0.0006$.
3. A negative exponential relation between s and the heterogeneity of the demand. For none of the s the distribution of demands come close to the real distribution (**UG.3**).

We conclude that this model can only explain the average demand by learning what demands result in the highest wealth; but not the average reject and heterogeneity of demands.

4 Explaining multiple scenarios

In the previous section we saw that our value-based model can better reproduce certain UG facts than our LHE model in a fixed scenario. In this section we aim to compare the agent models on their power to explain *multiple scenarios* of the UG: specifically UGs varying in pie size. This is helpful in our main aim: to compare a value-based with a LHE agent on its ability to reproduce human behavior in the UG. If one of the models can explain multiple scenarios than that model is more general and (in that dimension) better. We are interested in the generality of our results as our motivation comes not so much from understanding the ultimatum game as understanding if values are useful in designing social agents. We thus split our adapted dataset on the median pie size and as such obtain data for two more scenarios:

small pie size scenario In this scenario $P = 73$, the average demand is 59.2% and the average rejection rate 17.1%.

large pie size scenario In this scenario $P = 1830$, the average demand is 59.3% and the average rejection rate is 13.2%.[4]

Firstly, we check if, analogous to the previous section, we can reproduce this new small pie scenario and large pie scenario. We present these results in table 1. For now we compare what

[4] Note that although humans demand almost the same in both scenarios, the positive correlation between the pie size and the demand *is* significant (-0.32, $p = 0.02$). In fact, it is this relation between multiple scenarios we are interested in as we will see in a moment. Having said that further work will have to shed light on the exact effect of the pie size until then we adhere to these findings supported by [25, p. 178].

humans do (bottom row of the table) to what agents do when they are calibrated to the same scenario as they play in (these results are in italics). The average demand can be reproduced in both models. In contrast, the reject ratio can be (almost) reproduced by the value-based agent, but is way off for the LHE agent. In the same way as in the previous section our value-based agent outperforms the LHE agent; apparently not only for our target scenario, but also for this small pie and large pie scenario.

Secondly, we calibrate the parameters to one scenario and check if the model is general enough to reproduce the results from the other scenario. There are two ways to do this:

1. we test the model in a large pie size scenario with the parameters obtained in the small pie size scenario
2. we test the model in a small pie size scenario with the parameters obtained in the large pie size scenario

If we are going to transpose an agent from one scenario to the other we will need a method for this. For the value-based agent this is quite straightforward as we can easily transpose it by its defining parameters (di_μ and di_σ). In case of the LHE agent we 'train' the LHE agent in one scenario by letting it play 999 rounds. As in the previous section we try to find the parameter (s) for which this training leads to results that are as close as possible to human play. The LHE agent has now learned the utility of every action in the first scenario and will be tested for one round in the new scenario. To transpose the learned utilities we map actions in the first scenario to the second scenario in ratio to the pie size. For example, if an agent attributed a utility of 100 to demanding 36.5 (i.e $0.5P$) in the small pie scenario, it will attributed a utility of 100 to 915 (i.e. $0.5P$) in the large pie scenario. We take this approach as it seems to match the way the model of the LHE agent explains first round behavior (as explained in section 3.3), i.e. by learning from many 'UG'-like scenarios, and is to our knowledge the standard way to transpose learned rewards (e.g. see [30]).

Table 1 A table representing the results of different scenarios, different agent models and different calibration for the agent models.

		Average Demand		**UG.4**	**Reject Ratio**		**UG.5**
Agent	**Calibration**	Play Small Pie	Play Large Pie		Play Small Pie	Play Large Pie	
Value-based	Small Pie	*59.2%*	78.1%	✓	*16.3%*	0.0%	✓
	Large Pie	53.3%	*59.3%*	✓	81.3%	*13.0%*	✓
LHE	Small Pie	*59.2%*	59.2%	✗	*41.9%*	41.9%	✗
	Large Pie	59.3%	*59.3%*	✗	44.0%	*44.0%*	✗
Human		59.2%	59.3%	✓	17.1%	13.2%	✓

In table 1 we present our results. This time we compare the human play (bottom row) to what agents do when they are calibrated in a different scenario as they play (these results are underlined.) We see that none of the agent play matches the human play. When introduced to a new scenario the LHE agent does not change its behavior at all; the value-based agent makes too large leaps. Having said that the story is different if we look at a *qualitative* instead of a *quantitative* level. In particular can check for the following two qualitative requirements:

UG.4 The demanded share is higher for larger pie sizes.
UG.5 The rejection rate is lower for larger pie sizes.

In table 1 we compare the value-based and LHE agent and see that the value-based agent can reproduce these requirements where the LHE agent cannot. We conclude that the value-based agent thus outperforms the LHE agent in (1) being able to adapt to both scenarios when the parameters are

free and (2) being able to reproduce the relation between scenarios on a qualitative level when the parameters are fixed.

5 Discussion

The previous sections compare a LHE agent model and a value-based agent model on their power to explain human UG play. This section discusses the generalizability of these results. We found that in this study our value-based agent outperformed our LHE agent in several scenarios by reproducing rejection rates (**UG.2**) and the heterogeneity in demands (**UG.3**). Furthermore, the value-based agent could reproduce qualitative relations between the pie size and the demand (**UG.2**); and the pie size and the reject ratio (**UG.3**). We find the fact that our first try on a value-based model outperforms a LHE model very encouraging towards using values as a basis for agent models. This study is of course limited, as any other, in showing a comparison for a specific value model; a specific LHE model; and testing it against specific requirements. The relevance of this study is thus firstly as a positive example of what values can do, but more importantly in getting insight in when and why they are of valuable.

One of the main reason for the performance of the value-based model seems to be the ability to differ in criteria on which an action is evaluated. The model allows us to specify how such criteria are evaluated. The value of wealth is satisfied as money increases, where the value of fairness is satisfied when more equal demands are made. We are aware that this is not the first model that evaluates action on multiple criteria; our model differs in that it interprets this criteria as human values. We argue that in this study having a well-studied semantic to the criteria helped us. Firstly, empirical work on values gave us the requirement that the importance attributed to the value of wealth and fairness are negatively correlated. Secondly, intuitions on values gave us an idea on how to model the relation of the obtained money in the UG to the satisfaction of the values. Thirdly, theory on value conflicts gave us an idea of how to model a preference over actions based on multiple values. Lastly, previous work on values helps us in interpreting the results as we will show now.

In section 4 we saw that although our value-based model outperforms our LHE model it does not reproduce the exact demands and rejection rates when the parameters are fixed. This is not unexpected. As said values are seen as an abstract fixed point in human reasoning. Values are thus a good instrument to model the qualitative relation of actions over many context, but to determine a specific outcome one needs to take into account the moderating and mediating variables that are specific to a context. Even in a scenario as the UG where many of these influences have artificially been stripped we will need a more complex model to reproduce human behavior. That complexity might be caught in a single function, as attempted in this study. But most likely such a function would be difficult to interpret and thus lose meaning. We see more good in a more modular approach where we connect the theory of values with appropriate models of norms, culture and identity. This leaves open at least two directions for further work. Firstly, one could concentrate on values as an instrument to model a rough pattern of behavior over many different sorts of context. Secondly, our current value-based model could be improved (in terms of representing human play) by connecting it with other social concepts. Using the same methodology as in this paper one could show the merit of these concepts by comparing which data the different models can explain. In this way we could build in a stepwise manner to social agents.

As a last note we feel the need to make explicit that we pay little attention to the popular ‘Nash Equilibrium’ as to our knowledge (normal, subgame-perfect or evolutionary) Nash Equilibria have not been shown to be representative of human play in the UG [5].

6 Conclusion

In this paper our aim is to provide partial answer to the question of how informative values are for modeling realistic agents, by comparing the realism of a value-based agent to a LHE agent in the ultimatum game. We found that our value-based model outperforms our LHE model. First, in section 3 by reproducing a specific scenario of the UG and second, in section 4, by reproducing a small pie and large pie scenario and the qualitative relation in demands and rejects between these scenarios. We conclude that these results are very encouraging towards using values in agent models. Having said that, this is only a positive example and no definitive proof; part of the relevance of this research lies in showing when and why values are useful. In Section 5 we argued that values were useful in giving us empirical facts to base our work upon, an intuition on how to model their relation with outcomes of the UG, a theory on value conflicts and lastly in helping us interpret our results. One limitation of this model is namely that it cannot explain the exact average demand and exact reject rate in different scenarios when the parameters are fixed. We argue that this result can be seen as an example of the (in the literature) theorized abstractness of values. One direction for further work would be to improve the model by adding other social concepts as norms, social practices and identity. Another would be to focus on the power of values to model the abstract relation of actions over many different behavioral context. In conclusion, we can say that values have clearly helped improved our modeling of human decision making in the ultimatum game and hope that this paper can form the basis of a stepwise construction of a more general theory of social agents.

References

1. Atkinson, K., Bench-capon, T.: Value Based Reasoning and the Actions of Others. In: Proceedings of ECAI, pp. 680–688 (2016). DOI 10.3233/978-1-61499-672-9-680
2. Bench-Capon, T.J.M., Atkinson, K., McBurney, P.: Altruism and agents: an argumentation based approach to designing agent decision mechanisms. AAMAS '09: Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems pp. 1073–1080 (2009). URL http://portal.acm.org/citation.cfm?id=1558163
3. Bilsky, W., Janik, M., Schwartz, S.H.: The Structural Organization of Human Values-Evidence from Three Rounds of the European Social Survey (ESS). Journal of Cross-Cultural Psychology **42**(5), 759–776 (2011). DOI 10.1177/0022022110362757
4. Boero, R., Squazzoni, F.: Does empirical embeddedness matter? Methodological issues on agent-based models for analytical social science. Journal of Artificial Societies and Social Simulation **8**(4), 1–31 (2005)
5. Cooper, D.J., Kagel, J.H.: Handbook of Experimental Economics, vol. 2 (2013)
6. Davidov, E., Schmidt, P., Schwartz, S.H.: Bringing values back in: The adequacy of the European social survey to measure values in 20 countries. Public Opinion Quarterly **72**(3), 420–445 (2008). DOI 10.1093/poq/nfn035
7. Debove, S., Baumard, N., Andre, J.B.: Models of the evolution of fairness in the ultimatum game: A review and classification. Evolution and Human Behavior **37**(3), 245–254 (2016). DOI 10.1016/j.evolhumbehav.2016.01.001. URL http://dx.doi.org/10.1016/j.evolhumbehav.2016.01.001
8. Dignum, F., Dignum, V., Prada, R., Jonker, C.M.: A conceptual architecture for social deliberation in multi-agent organizations. Multiagent and Grid Systems **11**(3), 147–166 (2015). DOI 10.3233/MGS-150234
9. Dignum, F., Prada, R., Hofstede, G.: From Autistic to Social Agents. In: Proceedings of the 2014 international conference on Autonomous Agents and Multi-Agent Systems, pp. 1161–1164 (2014). URL http://dl.acm.org/citation.cfm?id=2617431
10. Edmonds, B.: Bootstrapping Knowledge About Social Phenomena. Journal of Artificial Societies and Social Simulation **13**(2010), 1–16 (2010)
11. Erev, I., Roth, A.E.: Predicting How People Play Games: Reinforcement Learning in Experimental Games with Unique, Mixed Strategy Equilibria. American Economic Review **88**(4), 848–881 (1998)
12. Fishbein, M., Azjen, I.: Predicting and Changing Behavior: The Reasoned Action Approach. Taylor & Francis (2011)

13. Fontaine, J.R.J., Poortinga, Y.H., Delbeke, L., Schwartz, S.H.: Structural Equivalence of the Values Domain Across Cultures: Distinguishing Sampling Fluctuations From Meaningful Variation. Journal of Cross-Cultural Psychology **39**(October), 345–365 (2008). DOI 10.1177/0022022108318112
14. Friedman, B.: Value-sensitive design. Interactions **3**(6), 16–23 (1996). DOI 10.1145/242485.242493
15. Gale, J., Binmore, K.G., Samuelson, L.: Learning to be imperfect: The ultimatum game. Games and Economic Behavior **8**(1), 56–90 (1995). DOI 10.1016/S0899-8256(05)80017-X
16. Gifford, R.: Environmental psychology matters. Annual review of psychology **65**, 541–79 (2014). DOI 10.1146/annurev-psych-010213-115048. URL http://www.ncbi.nlm.nih.gov/pubmed/24050189
17. Güth, W., Schmittberger, R., Schwarze, B.: An experimental analysis of ultimatum bargaining. Journal of Economic Behavior and Organization **3**(4), 367–388 (1982). DOI 10.1016/0167-2681(82)90011-7
18. John Stuart Mill: On the definition of political economy, and on the method of investigation proper to it. In: Essays on some unsettled questions of Political Economy, p. 326 (1844)
19. Kahneman, D., Tversky, A.: Prospect Theory: An Analysis of Decision under Risk. Econometrica **47**(2), 263 (1979). DOI 10.2307/1914185. URL http://www.jstor.org/stable/1914185?origin=crossref
20. Kaminka, G.: Curing robot autism: A challenge. Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems pp. 801–804 (2013). URL http://dl.acm.org/citation.cfm?id=2485047%5Cnhttp://link.springer.com/content/pdf/10.1007/978-3-319-05044-7.pdf
21. Mercuur, R., Dignum, F., Kashima, Y.: Changing Habits Using Contextualized Decision Making. In: Advances in Social Simulation 2015. Springer Series: Advances in Intelligent Systems and Computing, p. (in press) (2015). URL http://www.springerlink.com/index/10.1007/978-1-4020-9783-6
22. Miles, A.: The (Re)genesis of Values: Examining the Importance of Values for Action. American Sociological Review **80**(4), 680–704 (2015). DOI 10.1177/0003122415591800. URL http://asr.sagepub.com/content/80/4/680.abstract
23. Myerson, R.B.: Game theory. Harvard university press (2013)
24. Norling, E.: Don't Lose Sight of the Forest: Why the Big Picture of Social Intelligence is Essential. In: Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems, May, pp. 984–987 (2016)
25. Oosterbeek, H., Sloof, R., van de Kuilen, G.: Cultural Differences In Ultimatum Game Experiments: Evidence From A Meta-Analysis. SSRN Electronic Journal **188**, 171–188 (2001). DOI 10.2139/ssrn.286428. URL http://www.ssrn.com/abstract=286428
26. van de Poel, I., Royakkers, L.: Ethics, Technology, and Engineering: An Introduction. John Wiley & Sons (2011). URL http://eu.wiley.com/WileyCDA/WileyTitle/productCd-EHEP002302.html
27. Roth, A.E., Erev, I.: Learning in extensive-form games: Experimental data and simple dynamic models in the intermediate term. Games and Economic Behavior **8**(1), 164–212 (1995). DOI 10.1016/S0899-8256(05)80020-X
28. Schwartz, S.H.: An Overview of the Schwartz Theory of Basic Values. Online Readings in Psychology and Culture **2**, 1–20 (2012). DOI http://dx.doi.org/http://dx.doi.org/10.9707/2307-0919.1116
29. Schwartz, S.H., Cieciuch, J., Vecchione, M., Davidov, E., Fischer, R., Beierlein, C., Ramos, A., Verkasalo, M., Lönnqvist, J.E., Demirutku, K., Dirilen-Gumus, O., Konty, M.: Refining the theory of basic individual values. Journal of Personality and Social Psychology **103**(4), 663–688 (2012). DOI 10.1037/a0029393
30. Taylor, M.E., Stone, P.: Transfer Learning for Reinforcement Learning Domains : A Survey. Journal of Machine Learning Research **10**, 1633–1685 (2009). DOI 10.1007/978-3-642-27645-3. URL http://portal.acm.org/citation.cfm?id=1755839
31. Tosto, G.D., Dignum, F.: Simulating social behaviour implementing agents endowed with values and drives. In: Multi-Agent-Based Simulation XIII, pp. 1–12. Springer Berlin Heidelberg (2013)

A self-adaptive approach for mobile wireless sensors to achieve energy efficient information transmission

Xing Su[1], Limin Guo[1], Tong Li[1], and Zhiming Ding[1]

Beijing University of Technology, China
xingsu@bjut.edu.cn, guolimin@bjut.edu.cn, litong@bjut.edu.cn, zmding@bjut.edu.cn

Abstract. Wireless sensor networks have played an important role in many applications, where sensors in the network can automatically collect and transmit information of an environment without much human maintenance. To enhance the adaptability of sensors, people produce mobile wireless sensors through fixing wireless sensors on mobile vehicles or robots. The same as static wireless sensors, since the energy of mobile wireless sensors is still supplied by the battery, their energy management has a great impact on their lifetime. Different with static wireless sensors, the mobility of mobile wireless sensors gives us a new perspective to achieve efficient energy management. In this paper, a decentralized self-adaptive approach is proposed for mobile wireless sensors, which enables sensors in a mobile wireless sensor network to adapt their locations according to information transmission regularity so as to reduce their energy consumption for information transmission. The experimental results indicate that after employed the proposed approach to adapt locations, the energy consumption for information transmission of sensors in a mobile wireless sensor network can be greatly reduced and the lifetime of sensors is extended.

1 Introduction

Nowadays, due to the low infrastructure dependence, adaptability, scalability, etc, wireless sensor networks (WSNs) have played an important role in many applications, such as intelligent transportation, environment monitoring, health care, military, etc [1], [10]. In these applications, a number of sensors are deployed in a target environment, where they can sense to collect the information of the environment. Then, sensors can use wireless transceivers to transmit their collected information to receivers or sinks. Through WSNs, people, such as researchers, managers, etc, can obtain the information of an environment without entering it. However, this working mode means that wireless sensors have to work in an environment for a long period of time without maintenance. Since the energy of sensors is only supplied by the battery, how to effectively manage the limited available energy of sensors for information collection and transmission is a key issue in WSNs.

In WSNs, the energy of sensors are mainly consumed for sensing (information collection) and information transmission. By contrast, the energy consumed for sensing is much less than that consumed for information transmission. At the early stage of the WSN, sensors transmit their collected information directly (or through satellites) to receivers or sinks [2]. This type of WSNs is called unconnected WSNs. Based on the wireless communication technology, the energy consumption for information transmission is proportional to the square of the information transmission distance so that the energy consumption for information transmission in unconnected WSNs is huge. In order to reduce such energy consumption, many approaches were proposed from different perspectives. One kind of approaches is to establish connected WSNs [9] [8] [14]. In connected WSNs, sensors use their neighboring sensors to transmit information to receivers or sinks in a multi-hop way, which can greatly reduce the information transmission distance so as to save energy for information transmission. Rogers et al [7] proposed a self-organized routing approach for WSNs, which enables sensors to automatically choose the most suitable neighboring sensors to transmit information. In the recent years, many relay-based approaches are proposed, which reduce the information transmission distance through deploying relays at suitable locations in a WSN. Relays act as temporary receivers or sinks, which collect information from nearby sensors and transmit to the final receivers or sinks.

All above energy efficient approaches are proposed for static WSNs, where sensors in the WSN cannot move. In order to enhance the adaptability of WSNs, people produce mobile wireless sensors through fixing sensors on mobile vehicles or robots and the network consisting of this kind of sensors is called the mobile wireless sensor network (MWSN). In MWSNs, mobile wireless sensors can dynamically adapt their locations according to the requirement of environments and other sensors in the network, which makes MWSNs more adaptive than static WSNs. In addition, the adaptability of MWSNs create a new perspective for sensors to achieve efficient energy management. However, the research on MWSNs mainly focuses on the use of sensors' mobility to route or self-organize in complex environments and barely pays attention to reduce the energy consumption of sensors. This is because that the energy consumption problem of MWSNs can be partially solved by approaches proposed for static WSNs.

Against this background, the motivation of this research is to use of the mobility of sensors in MWSNs to reduce their energy consumption for information transmission. In this paper, a decentralized self-adaptive approach is proposed for sensors in MWSNs, which enables them to adapt their locations according to the information transmission regularity in the MWSN so as to reduce their energy consumption for information transmission. The contributions of the proposed approach are described as follows.

- The proposed approach considers the characteristics of MWSNs, which enables sensors with local views about the MWSN to adapt their locations in a decentralized manner;
- The proposed approach contains an optimization method, which enables sensors to quickly find their adaptive locations;

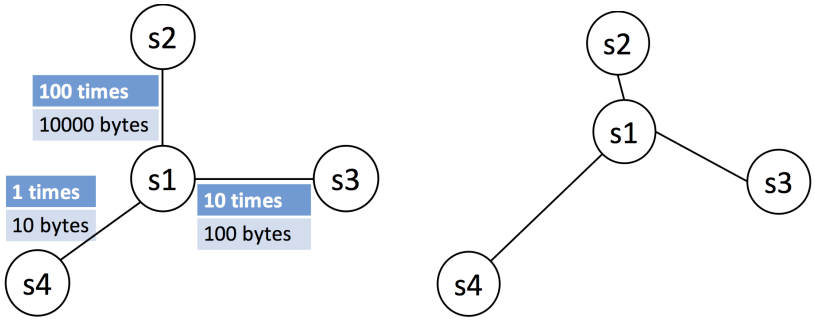
– The proposed approach considers the shrink problem of MWSNs during the location adaption of sensors, which contains a compensation mechanism to handle such problem.

The rest of this paper is organized as follows. Section 2 gives the problem description and definitions of the proposed approach. Section 3 is the basic principle of the proposed approach. Section 4 gives experimental results and analysis. Section 5 introduces the related work of the proposed approach. Section 6 is the conclusion and future work.

2 Problem description and definitions

In a connected MWSN, mobile wireless sensors have their regular information transmission routings, which uses their wireless connections with neighboring sensors to transmit information to receivers or sinks in a multi-hop way. During the information transmission of a sensor, the usages of different connections to different neighboring sensors is not the same, where the frequency and the amount of information transmitted through some connections to some neighboring sensors are much more than that transmitted through other connections to other neighboring sensors. Since the energy consumption for information transmission is proposition to the square of information transmission distance [7], mobile wireless sensors in a MWSN should adapt their locations according to their information transmission regularity, where sensors should reduce their distance to the neighboring sensors that frequently exchange (i.e, transmit or receive) big amount of information so as to minimize their energy consumption for information transmission.

As the example shown in Fig. 1(a), a sensor s_1 has three neighboring sensors s_2, s_3 and s_4. The information transmission regularity between s_1 and s_2, s_3, s_4 is that s_1 exchanges 100 times and 10000 bytes information with s_2, 10 times and 100 bytes information with s_3 and 1 time and 10 bytes information with s_4. To reduce the energy consumption for information transmission, s_1 should adapt its location to reduce its distance with s_2, which is shown as Fig. 1(b)



(a) Information transmission of s_1 (b) Adaptive location of s_1

Fig. 1. The object of the proposed approach

A connected MWSN consists of m number of sensors, which are $s_1, ..., s_i, ..., s_m$, where s_i represents the i^{th} sensor in the MWSN.

Definition 1: A *mobile wireless sensor* (s_i) can be defined as a three-tuple:

$$s_i =< id, loc_i, NSen_i >, \tag{1}$$

where id is the unique identification of s_i in the MWSN, $loc_i = (x_i, y_i)$ represents the current location of s_i and $NSen_i = \{sn_{i1}, ...sn_{ik}, ..., sn_{ip}\}$ represents all neighboring sensors of s_i, with which s_i can exchange (i.e, transmit or receive) information.

In order to record the information transmission regularity to adapt the location, each sensor (e.g. s_i) needs to maintain its information transmission records, which can be described as $ITr_i = \{itr_{i1}, ..., itr_{ij}, ..., itr_{in}\}$, where itr_{ij} represents the j^{th} information transmission of s_i.

Definition 2: An *information transmission record* of s_i (itr_{ij}) can be defined as a three-tuple:

$$itr_{ij} =< from, to, a_{ij} >, \tag{2}$$

where $from$ is the id of the sensor that transmits information, to is the id of the sensor that receives information and a_{ij} represents the amount of information that is transmitted in bytes. Since itr_{ij} is the information transmission record of s_i, s_i is either the information transmitter ($from = s_i$ and $to = sn_{ik}$) or the information receiver ($from = sn_{ik}$ and $to = s_i$).

All of the information transmission records of a MWSN can be described as $\{Itr_1, ..., Itr_i, ..., Itr_m\}$, which have duplicated information transmission records, since for each information transmission record, both the transmitter (i.e, sensor $from$) and the receiver (i.e, sensor to) maintain the same record.

2.1 The object of the proposed approach

The proposed approach aims to reduce the energy consumption for information transmission of sensors through adapting their locations to reduce their distances to the neighboring sensors that frequently exchange (i.e, transmit or receive) big amount of information. The energy consumption for information transmission is proposition to the square of the information transmission distance [7]. To evaluate the energy consumption for information transmission in a MWSN, the object value (i.e, OBJ) of the proposed approach calculated based on the Euclidean distance is described as follows.

$$OBJ = \sum_{i=1}^{m} \sum_{itr_{ij}} ((x_{from} - x_{to})^2 + (y_{from} - y_{to})^2) \cdot a_{ij}, \tag{3}$$

where m is the number of sensors in the MWSN, (x_{from}, y_{from}) is the location of the sensor that transmits the information, (x_{to}, y_{to}) is the location of the sensor that receives the information and a_{ij} is the amount of information that is transmitted in bytes. The object of the proposed approach is to minimize the energy consumption for information transmission in the MWSN (i.e, $min\ OBJ$).

3 The basic principle of the proposed approach

The proposed approach enables sensors with only local view about the MWSN to adapt their locations according to their information transmission regularity so as to reduce their energy consumption for information transmission. To achieve this object, an optimization method based on the object function (i.e, Equation 3) is proposed in Subsection 3.1. Then, the shrink problem caused by the optimization method is analyzed in Subsection 3.2. To handle the shrink problem, a compensation mechanism is proposed in Subsection 3.3. Combining with the optimization method and the compensation mechanism, the proposed approach can generate energy efficient adaptive locations for sensors in a MWSN.

3.1 The optimization method for the sensor location adaption

In the proposed approach, the object function of the MWSN is minimized through minimizing the object function of each sensor, which is describe as follows.

$$min\ OBJ = \sum_{i=1}^{m} min\ obj_{s_i}, \tag{4}$$

where m is the number of sensors in the MWSN. The object function of a sensor s_i in the MWSN is described as follows.

$$obj_{s_i} = \sum_{j=1}^{n}((x_i - x_{ik})^2 + (y_i - y_{ik})^2) \cdot a_{ij}, \tag{5}$$

where n is the number of information transmission records of s_i, (x_i, y_i) is the location of s_i, (x_{ik}, y_{ik}) is the location the neighboring sensor sn_{ik} of s_i, which exchanges (i.e, transmits or receives) information with s_i in the information transmission record itr_{ij} (see, Definition 2) and a_{ij} is the amount of information that is transmitted.

The minimization of Equation 5 can be described as follows.

$$\begin{aligned} min\ obj_{s_i} &= min \sum_{j=1}^{n}((x_i - x_{ik})^2 + (y_i - y_{ik})^2) \cdot a_{ij} \\ &= min(x_i^2 \cdot \sum_{j=1}^{n} \cdot a_{ij} + y_i^2 \cdot \sum_{j=1}^{n} \cdot a_{ij} - x_i \cdot \sum_{j=1}^{n} 2 \cdot x_{ik} \cdot a_{ij} \\ &\quad - y_i \cdot \sum_{j=1}^{n} 2 \cdot y_{ik} \cdot a_{ij} + \sum_{j=1}^{n}(x_{ik}^2 + y_{ik}^2) \cdot a_{ij}), \end{aligned} \tag{6}$$

Since the locations (x_{ik}, y_{ik}) of all neighboring sensors of s_i are known, the objective function of s_i is a quadratic and convex function of (x_i, y_i). To calculate the value of (x_i, y_i) that minimizes the function, the convex optimization is employed in the proposed approach. Based on the convex optimization, the

object function can be minimized when the partial derivatives of x_i and y_i are '0', respectively, which is described as follows.

$$\begin{cases} \dfrac{\partial obj_{s_i}}{\partial x_i} = 2 \cdot x_i \cdot \sum\limits_{j=1}^{n} \cdot a_{ij} - \sum\limits_{j=1}^{n} 2 \cdot x_{ik} \cdot a_{ij} = 0, \\ \dfrac{\partial obj_{s_i}}{\partial y_i} = 2 \cdot y_i \cdot \sum\limits_{j=1}^{n} \cdot a_{ij} - \sum\limits_{j=1}^{n} 2 \cdot y_{ik} \cdot a_{ij} = 0, \end{cases} \tag{7}$$

$$\begin{cases} x_i = \dfrac{\sum_{j=1}^{n} x_{ik} \cdot a_{ij}}{\sum_{j=1}^{n} \cdot a_{ij}}, \\ y_i = \dfrac{\sum_{j=1}^{n} y_{ik} \cdot a_{ij}}{\sum_{j=1}^{n} \cdot a_{ij}}, \end{cases} \tag{8}$$

From Equation 8, it can be seen that the adaptive location of s_i is the weighted average location of its information transmission.

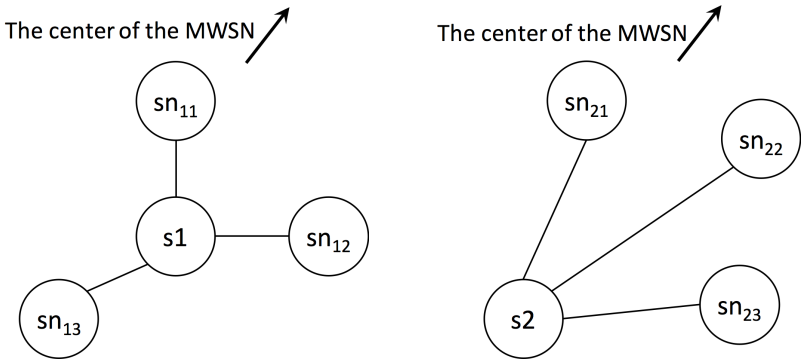
3.2 The shrink problem of the MWSN

According to Equation 8, the adaptive location of s_i based on the information transmission records can be found. However, in the real applications, if sensors in a MWSN adapt their locations according to Equation 8, the shrink problem will occur. The shrink problem means that sensors adapt to concentrate to the center of the MWSN, which will greatly reduce the size and coverage area of the MWSN.

The reason for this problem can be explained as follows. An information transmission record (e.g, itr_{ij}) can be considered as an adaption of s_i. The direction of the adaption is from the location of s_i to the location of sn_{ik} that exchanges (i.e, transmit or receive) information with s_i in itr_{ij}. The adaption distance is proportion to the amount of information (i.e., a_{ij}) that is transmitted in itr_{ij}. Based on this consideration, according to Equation 8, s_i is more likely to adapt to the direction with more neighboring sensors. As the example shown in Fig. 2, the arrows represent the direction of the center of the MWSN. For a sensor (e.g, s_1 in Fig. 2(a)) in the middle of the MWSN, most of time, the distribution of neighboring sensors of s_1 is balance. While for a sensor (e.g. s_2 in Fig. 2(b)) at the edge of the MWSN, the distribution of neighboring sensors of s_2 is more likely to bias to the center of the MWSN. After employing Equation 8, s_2 will definitely adapt to the center of the MWSN, which reduces the size and coverage area of the MWSN.

3.3 The compensation mechanism

To handle the shrink problem, a compensation mechanism is proposed in this paper. In the compensation mechanism, a hypothetical neighboring sensor (hs_i) is



(a) The neighboring sensors of s_1 (b) The neighboring sensors of s_2

Fig. 2. The shrink problem of the MWSN

generated for each sensor (s_i) in the MWSN, which is to balance the distribution bias of all real neighboring sensors of s_i.

To determine the location of hs_i, the distribution bias of all real neighboring sensors of s_i should be calculated. For a real neighboring sensor of s_i (e.g. sn_{ik}), its distribution bias can be considered as a vector v_{ik}, where the direction of v_{ik} is from the location of s_i to the location of sn_{ik}. The length of v_{ik} is the Euclidean distance between s_i and sn_{ik}. Based on this description, the vector v_{ik} can be described as follows.

$$v_{ik} = (x_{ik} - x_i, y_{ik} - y_i), \tag{9}$$

where (x_i, y_i) is the location of s_i and (x_{ik}, y_{ik}) is the location of sn_{ik}.

According to Equation 9, the distribution bias (i.e, $RV_i = (x_{rv_i}, y_{rv_i})$) of s_i can be calculated based on vectors of all real neighboring sensors of s_i (i.e, $\{v_{i1}, v_{i2}, ..., v_{ik}\}$) as follows.

$$\begin{cases} x_{rv_i} = \sum_{k=1}^{p} (x_{ik} - x_i) \\ y_{rv_i} = \sum_{k=1}^{p} (y_{ik} - y_i), \end{cases} \tag{10}$$

where p is the number of real neighboring sensors of s_i (i.e, see Definition 1).

To balance RV_i, the vector (i.e, v_{hs_i}) of the hypothetical neighboring sensor hs_i should be $v_{hs_i} = (-x_{rv_i}, -y_{rv_i})$. The location of hs_i (i.e, (x_{hs_i}, y_{hs_i})) is calculated as follows.

$$\begin{cases} x_{hs_i} = x_i + \sum_{k=1}^{p} (x_i - x_{ik}) \\ y_{hs_i} = y_i + \sum_{k=1}^{p} (y_i - y_{ik}), \end{cases} \tag{11}$$

Except for the location of hs_i, the amount of information (i.e, $a_{(i,hs_i)}$) that is transmitted between s_i and hs_i should also be generated. In the proposed compensation mechanism, the amount of information that is transmitted between

s_i and hs_i is proportion to the average amount of information that is transmitted between s_i and all its real neighboring sensors, which can be calculated as follows.

$$a_{i,hs_i} = \frac{\beta \cdot \sum_{itr_{ij} \in Itr_i} a_{ij}}{p}, \tag{12}$$

where β is the coefficient to control the amount of compensation, itr_{ij} is an information transmission record of s_i, ITr_i is all information transmission records of s_i, a_{ij} is the amount of information that is transmitted in itr_{ij} and p is the number of real neighboring sensors of s_i (i.e, see Definition 1).

Based on all above equations, the compensated adaptive location (i.e, (fx_i, fy_i)) of the sensor s_i based on its information transmission records and the hypothetical neighboring sensor hs_i can be calculated as follows.

$$\begin{cases} fx_i = \dfrac{(x_i + \sum_{k=1}^{p} x_i - x_{ik}) \cdot \frac{\beta \cdot \sum_{itr_{ij} \in Itr_i} \cdot a_{ij}}{p} + \sum_{j=1}^{n} x_{ik} \cdot a_{ij}}{\frac{\beta \cdot \sum_{itr_{ij} \in Itr_i} \cdot a_{ij}}{p} + \sum_{j=1}^{n} \cdot a_{ij}} \\ fy_i = \dfrac{(y_i + \sum_{k=1}^{p} y_i - y_{ik}) \cdot \frac{\beta \cdot \sum_{itr_{ij} \in Itr_i} \cdot a_{ij}}{p} + \sum_{j=1}^{n} y_{ik} \cdot a_{ij}}{\frac{\beta \cdot \sum_{itr_{ij} \in Itr_i} \cdot a_{ij}}{p} + \sum_{j=1}^{n} \cdot a_{ij}} \end{cases} \tag{13}$$

4 Experiments

In this section, the experiment is to evaluate the performance of the proposed approach (i.e, the optimization method and the compensation mechanism) on reducing energy consumption for information transmission of sensors in a MWSN.

4.1 Experimental settings

In the experiment, 30 wireless mobile sensors randomly deployed to form a MWSN in a 30 x 30 environment. The total amount of information that is transmitted between two sensors is between 1000 bytes to 10000 bytes.

The settings of the experiment are shown in Table 1.

Name	Value
The size of the environment	30×30 $(unit\ of\ distance)^2$
The number of sensors	30
Amount of transmitted information	1000 $bytes \sim$ 10000 $bytes$

Table 1. The settings of the experiment

The experiment evaluates the impact of coefficient β, which represent the compensation of the hypothetical sensors on the location adaption of sensors. In the experiment, β is ranged from 0 to 1 with 0.2 per step. To evaluate the performance of the proposed approach, two parameters are calculated and compared in the experiment, which are the object value (OBJ) and the size (S) of the MWSN before and after the location adaption of sensors.

The object value of the MWSN represents the energy consumption for information transmission of sensors in the MWSN, which can be calculated based on Equation 3 as follows.

$$OBJ = \sum_{i=1}^{m} \sum_{itr_{ij}} ((x_{from} - x_{to})^2 + (y_{from} - y_{to})^2) \cdot a_{ij}, \tag{14}$$

The size of the MWSN represents the shrink degree of the MWSN, which can be calculated based on the maximum differences of x and y coordinates of sensor locations in the MWSN.

$$S = (max(x_{i1}) - min(x_{i2})) \times (max(y_{i3}) - min(y_{i4})) \tag{15}$$

where $max(x_{i1})$ and $min(x_{i1})$ are the maximum and minimum values of x-coordinate of sensor locations in the MWSN, respectively. While $max(y_{i3})$ and $min(x_{i4})$ are the maximum and minimum values of y-coordinate of sensor locations in the MWSN, respectively.

4.2 Experimental results and analysis

The experimental results are shown in Fig. 3 and Fig. 4. In Fig. 3, the X-axis is the values of β, while the Y-axis is the object values (i.e, OBJ) of the MWSN. The experimental results are shown as follows.

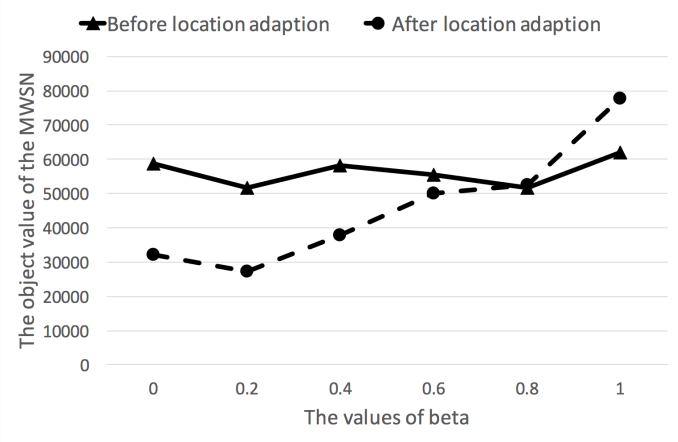


Fig. 3. The object value OBJ of the MWSN

From Fig. 3, it can be seen that when β is 0 (i.e, no compensation to balance the distribution bias of sensors), after employed the proposed approach to adapt the locations of sensors, the object value OBJ of the MWSN can be greatly reduced. With the increase of β (i.e, increase the compensation), after the location adaption of sensors, the reduction of OBJ is becoming smaller. This is because that the compensation of hypothetical sensors prevents the sensors in the MWSN to achieve their object. Therefore, when β is 1 (i.e, the compensation is dominant the location adaption of sensors), OBJ after the location

adaption of sensors is more than OBJ before the location adaption of sensors, which means that after location adaption, sensors have to consume more energy for information transmission. According to the experiment, the high value of β is not applicable in the proposed approach.

In Fig. 4, the X-axis is the values of β, while the Y-axis is the sizes (S) of the MWSN.

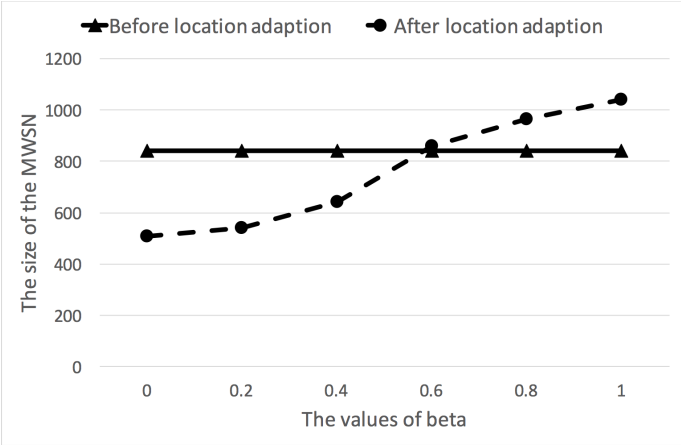


Fig. 4. The size S of the MWSN

From Fig. 4, it can be seen that when β is 0 (i.e, no compensation to balance the distribution bias of sensors), after employed the proposed approach to adapt the locations of sensors, the size S of the MWSN is greatly reduced, which means that after the location adaption, a very serious shrink problem occurs in the MWSN. With the increase of β (i.e, increase the compensation), after the location adaption of sensors, the shrink problem of the MWSN is alleviating. This is because that the compensation of hypothetical sensors prevents the shrink of the MWSN during the location adaption of sensors. Therefore, when β is 1 (i.e, the compensation is dominant the location adaption of sensors), after location adaption of sensors, S is even larger. This experiment indicates that the compensation of hypothetical sensors can effectively prevent the shrink problem of the MWSN during the location adaption of sensors.

In summary, the introducing of the hypothetical sensors can effectively solve the shrink problem when employing the proposed approach to adapt locations of sensors in a MWSN. However, the compensation (i.e, β) cannot be too big so as to enable the sensors in a MWSN to adapt to achieve energy efficient information transmission (i.e, lower object value and higher energy efficiency). According to the experiment, the suitable values of β is $[0.4, 0.5]$.

5 The related work

In the last twenty years, many approaches are proposed for sensors to reduce their energy consumption and extend their lifetime from different perspectives [12], [6], [5], [13], [4].

Yang et al. proposed a relay placement approach for sensor networks [11], which considers the two important issues in long-term sensor networks: the energy and the connectivity. In their approach, the deployment locations of relays are found through a tradeoff between the longest lifetime of relays and the maximum coverage of sensors.

Rogers et al. [7] proposed a self-organized approach to adapt the information transmission routing of sensors in wireless micro-sensor networks. In their approach, to reduce the energy consumption for information transmission, Rogers et al. enable sensors in a WSN to adapt their information transmission routing, which transmit information with assistant of neighboring sensors. By doing so, sensors can greatly reduce their information transmission distances so as to reduce the energy consumption for information transmission.

However, most of these approaches are proposed for static WSNs, where the sensor cannot move. Different with these approaches, the proposed approach aims to use the mobility of the mobile wireless sensors to reduce the energy consumption through adapting their locations to reduce their information transmission distances with neighboring sensors.

Heo et al. [3] proposed a self-spreading approach for the location adaption of mobile wireless sensors which was inspired by the equilibrium of molecules. The strength of interaction forces between two mobile wireless sensors is calculated from the distance between the two sensors. The final adaptive location of each sensor is the balance point of interaction forces. Through controlling the strength of interaction forces, Heo et al.'s approach can adapt the distances between sensors. However, based on their approach, the distances between a sensor and its neighboring sensors are always the same. Different from Heo et al.'s approach, the proposed approach adapts the locations of sensors according to the information transmission regularity in the MWSN, which is more suitable to reduce the energy consumption for information transmission than the molecule equilibrium-based approach.

6 Conclusion

In this paper, a decentralized self-adaptive approach is proposed for sensors in a MWSN to adapt their locations so as to reduce their energy consumption for information transmission. To achieve this object, an optimization method is proposed to adapt sensors' locations according to their information transmission records. In addition, to handle the shrink problem of the MWSN during the sensor location adaption, a compensation mechanism is proposed, which generates hypothetical sensors to balance the distribution bias of real neighboring sensors. The experimental results have shown that through the adaption of the proposed approach (i.e, the optimization method and the compensation mechanism), the energy consumption for information transmission of sensors in a MWSN can be effectively reduced. In the future, we will improve the compensation mechanism of the proposed approach and make it to accurately compensate distribution bias of real neighboring sensors.

Acknowledgment

The work is supported by the National Natural Science Foundation of China (Grants No. 61402449, 91546111, 91646201), and the Key Project of Beijing Municipal Education Commission (Grants No. KZ201610005009).

References

1. L. Azpilicueta, P. L. Iturri, E. Aguirre, J. J. Astrain, J. Villadangos, C. Zubiri, and F. Falcone. Characterization of wireless channel impact on wireless sensor network performance in public transportation buses. *IEEE Transactions on Intelligent Transportation Systems*, 16(6):3280–3293, 2015.
2. W. Guo, X. Huang, and Y. Liu. Dynamic relay deployment for disaster area wireless networks. *Wirel. Commun. Mob. Comput.*, 10(9):1238–1252, September 2010.
3. N. Heo and P.K. Varshney. A distributed self spreading algorithm for mobile wireless sensor networks. In *IEEE Wireless Commun. Networking Conf.*, volume 3, pages 1597–1602 vol.3, March 2003.
4. A. Mai, Y. Liang, and T. Li. Mobile coordinated wireless sensor network: An energy efficient scheme for real-time transmissions. *IEEE Journal on Selected Areas in Communications*, 34(5):1663–1675, 2016.
5. M. Mitici, J. Goseling, M. D. Graaf, and R. J. Boucherie. Energy-efficient data collection in wireless sensor networks with time constraints. *Performance Evaluation*, 102:34–52, 2016.
6. A. K. Nayak, B. B. Misra, and S. C. Rai. Energy efficient adaptive sensing range for sensor network. In *International Conference on Energy, Automation, and Signal*, pages 1–6, 2011.
7. A. Rogers, E. David, and N. R. Jennings. Self-organized routing for wireless microsensor networks. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 35(3):349–359, 2005.
8. Z. Shao, Y. Liu, Y. Wu, and L. Shen. A rapid and reliable disaster emergency mobile communication system via aerial ad hoc bs networks. 34(17):1–4, 2011.
9. A Srinivas, G. Zussman, and E. Modiano. Construction and maintenance of wireless mobile backbone networks. *IEEE/ACM Trans. Networking*, 17(1):239–252, Feb 2009.
10. A. Triantafyllidis, V. Koutkias, I. Chouvarda, and N. Maglaveras. An open and reconfigurable wireless sensor network for pervasive health monitoring. *Methods of Information in Medicine*, 47(3):229–234, 2008.
11. D. Yang, S. Misra, X. Fang, G. Xue, and J. Zhang. Two-tiered constrained relay node placement in wireless sensor networks: Computational complexity and efficient approximations. *IEEE Transactions on Mobile Computing*, 11(8):1399–1411, 2012.
12. D. Ye and M. Zhang. A self-adaptive sleep/wake-up scheduling approach for wireless sensor networks. *IEEE Transactions on Cybernetics*, 2017.
13. Z. Zheng, A. Liu, L. X. Cai, Z. Chen, and X. Shen. Energy and memory efficient clone detection in wireless sensor networks. *IEEE Transactions on Mobile Computing*, 15(5):1–1, 2016.
14. Y. Zhu, C. Xue, H. Cai, J. Yu, L. Ni, M. Li, and B. Li. On deploying relays for connected indoor sensor networks. *Journal of Communications & Networks*, 16(3):335–343, 2014.

A Broker-Based Matching Approach between Trading Agents using Multi-Objective Optimization in an Open E-marketplace

Dien Tuan Le and Minjie Zhang and Fenghui Ren

Abstract A broker in e-marketplaces can provide many value-adding functions that cannot be replaced by direct buyer-seller dealings. A broker acts as a middleman between buyers and sellers in the trading processes to satisfy a buyer's requirements as per a seller's offers. Although a broker plays an important role in e-marketplaces, theory and guidelines for matching between buyers and sellers in multi-attribute trading are limited. This paper proposes a novel approach to match between buyers and sellers through a broker based on a multi-objective optimization model under consideration of price discount for a buyer as per trade volume and a seller's discount policy for a broker. The major contributions of this paper are that (i) a proposed framework is applicable to help a broker to carry out the matching progress in multi-attribute trading; (ii) a formula system is generated to measure a buyer's satisfaction degree as per a seller's offers; and (iii) a multi-objective model is built to maximize a buyer's satisfaction degree, a broker's turnovers and a broker's profits. Experimental results demonstrate the good performance of the proposed approach in terms of maximizing a buyer's satisfaction degree and a broker's turnovers and a broker's profits.

1 Introduction

In the recent years, e-marketplaces become more popular in many organizations and gradually replace more and more from the conventional business [9]. Much trading information between buyers and sellers is exchanged in business environ-

Dien Tuan Le
University of Economics, The University of Da Nang, Viet Nam, e-mail: dtl844@uowmail.edu.au

Minjie Zhang
University of Wollongong, Australia e-mail: minjie@uow.edu.au

Fenghui Ren
University of Wollongong, Australia e-mail: fren@uow.edu.au

ments. Specially, matching processes between buyers and sellers become complex and difficult on last twenty years due to the increasing number of buyers and sellers in e-marketplaces. Facing the inconvenience, people are paying more and more attention to an electronic broker. A broker's purposes are to find the best fitting trading target between buyers and sellers in the time as short as possible in e-marketplaces [6, 4].

Research on brokers or intermediaries in e-marketplaces as a third party of the trading processes between buyers and sellers has been a very active direction in recent years [2]. Tiwari et al. [10] proposed a new approach to select cloud service providers based on users' service requirements through a cloud broker in the cloud environment. They applied the rough set to model the given services of the cloud service providers and users' requirements to find the optimal cloud service providers. Jiang et al. [5] and Li et al. [7] built a multi-objective function to carry out matching progresses through a broker. Srivastava et al. [8] studied modelling and managing attributes in a seller's offers through a broker to select the best seller as per buyers' requirements. Alpár et al. [1] proposed a conceptual framework of matchmaking in a B2B e-marketplace environment. Matchmaker's responsibility includes analysis, modelling, implementation and optimization.

The above approaches have focused on studying brokers as the third party in the trading process between buyers' requirements and sellers' offers in e-marketplaces. However, there is little theory and few guidelines to help a broker to satisfy a buyer's requirements and maximize a buyer's satisfaction degree, a broker's turnovers and a broker's profits based on sellers' discount policy for buyers and a broker. A broker's turnovers are an amount of money that buyers pay to sellers through a broker-based trading processes; a buyer's satisfaction degree is to measure a buyer's happiness as per a seller's offers through multi-attribute trading; and a broker's profits are calculated based on sellers' discount policy for a broker. Therefore, how to maximize a buyer's satisfaction degree, a broker's turnovers and a broker's profits based on sellers' discount policy for a broker and buyers; and multi-attribute trading in a buyer's requirements and a seller's offers is one of the most important challenges for a broker.

In order to solve this challenge, this paper proposes a broker-based matching approach between trading agents using multi-objective optimization. The major contributions of this paper are as follows: (i) a proposed conceptual framework is applicable to help a broker to allocate buyers' requirements to sellers' offers using a multi-objective function; (ii) a novel formula system is proposed to measure a buyer's satisfaction degree, a broker's turnovers and a broker's profits as per sellers' discount policy; and (iii) a multi-objective function and a set of constraints are generated to carry out a broker's matching process. Experimental results show that the proposed matching approach is feasible and efficient to find out a set of Pareto-optimal solutions rather than only one solution for a broker in a business environments.

The rest of this paper is organized as follows. Problem description and definitions are presented in Section 2. The broker-based proposed matching approach is intro-

duced in Section 3. An experiment is presented in Section 4. Section 5 concludes in this paper and points out our future work.

2 Problem description and definitions

A broker acts as a third party in market environments to facilitate interactions between buyers and sellers by satisfying both buyers' and sellers' needs [4, 11]. A broker's main responsibility in this paper is how to allocate buyers' requirements to sellers' offers to seek the optimal allocation pairs in a given time interval so that a broker's turnovers, a broker's profits and a buyer's satisfaction degree are maximized. Before the detail contents of the proposed approach are presented, it is necessary to define the scope of the proposed approach and provide some necessary definitions.

A buyer $B_i(i = 1,\ldots,n)$ is considered as a consumer who would like to find a particular commodity from an e-marketplace to satisfy a buyer's requirements.

Definition 1. B_i's requirements are related to trade volume, namely C_{iv}; expected price, namely C_{ip} and many other attributes. Excepting the attributes of trade volume and expected price, many other attributes in B_i's requirements are divided to two categories including attributes with hard constraints and attributes with soft constraints. Thus, B_i's requirements related to attributes with hard constraints are presented by $REQH_i$ and B_i's requirements related to attributes with soft constraints are presented by $REQS_i$. In particular, $REQH_i$ is defined by the following format.

$$REQH_i = \begin{pmatrix} A_1 & A_2 & \ldots & A_h \\ C_{i1} & C_{i2} & \ldots & C_{ih} \end{pmatrix}, \tag{1}$$

where h is a number of attributes with hard constraints in B_i's requirements; A_h indicates the h^{th} attribute name and C_{ih} is the constraint value of A_h. Hard constraints must be satisfied in the trade. Thus, hard constraints do not consider their weight.

Similarly, $REQS_i$ is defined by the following format.

$$REQS_i = \begin{pmatrix} A_1 & A_2 & \ldots & A_k \\ C_{i1} & C_{i2} & \ldots & C_{ik} \\ W_{i1} & W_{i2} & \ldots & W_{ik} \end{pmatrix}, \tag{2}$$

where A_k indicates the k^{th} attribute name; C_{ik} is the constraint value of A_k; and W_{ik} is the weight of the soft constraint C_{ik}. K'_B is a number of attribute with soft constraints in $REQS_i$. Thus $\sum_{k \in K'_B} W_{ik} = 1, W_{ik} \geq 0$.

A seller $S_j(j = 1,\ldots,m)$ is considered as a company or an organization which has resources to provide to e-marketplaces.

Definition 2. S_j's offers are related to trade volume, namely Q_{jv}; sale price, namely Q_{jp}; and many other attributes. Excepting the attributes of trade volume and sale

price, many other attributes in S_j's offers, namely OFF_j, is defined by the following format.

$$OFF_j = \begin{pmatrix} A_1 & A_2 & \dots & A_{t'} \\ Q_{j1} & Q_{j2} & \dots & Q_{jt'} \end{pmatrix}, \tag{3}$$

where t' is a number of attribute in S_j's offers; $A_{t'}$ indicates the t'^{th} attribute name and $Q_{jt'}$ is the constraint value of A'_t.

Definition 3. A broker *BR* is defined as a 3-tuple $BR =< \mathbf{B}, \mathbf{S}, \mathbf{r} >$, where **B** is a set of buyer, **S** is a set of seller and **r** is a set of rewards that *BR* can get from sellers (see definition 4).

Definition 4. A set of rewards **r** is defined as follows.

$$\mathbf{r} = \{r_1^{S_1}, r_2^{S_2}, \cdots, r_m^{S_m}\}, \tag{4}$$

where $r_m^{S_m}$ is a reward which seller S_m offers to a broker if S_m's commodities are bought by a certain buyer through a broker.

Based on trading information of buyers' requirements and sellers' offers, a broker's main responsibility is to allocate buyers' requirements to sellers' offers under the consideration of sellers' price discount offers as per trade volume and rewards sellers offer to a broker so that a buyer's satisfaction degree, a broker's turnovers, and a broker's profits are maximized. The broker-based matching approach using a multi-objective function is proposed and presented in Section 3.

3 A broker-based matching approach

3.1 Framework of the proposed approach

The framework of the proposed approach to help a broker to solve the trade allocation problem under a multi-attribute trading is presented in Fig. 1 as follows.

In the framework, trading information related to multi-attribute commodities in buyers' requirements and sellers' offers is submitted to a broker. Furthermore, a broker interacts with a seller to model a seller's price discount offers as per trade volume. From trading information of buyers and sellers, a broker carries out the calculation of buyers' satisfaction degree to determine a constraint satisfaction layer including a group of buyers and sellers to work in a broker's trade allocation processes. A group of buyers includes any buyer to satisfy at least a seller's offers. Similarly, a group of sellers includes any seller to satisfy at least a buyer's requirements. After that, a multi-objective function is generated based on calculating the satisfaction degree of all buyers, a broker's turnovers, and a broker's profits. Finally, the multi-objective function is solved by the multi-objective genetic algorithm to

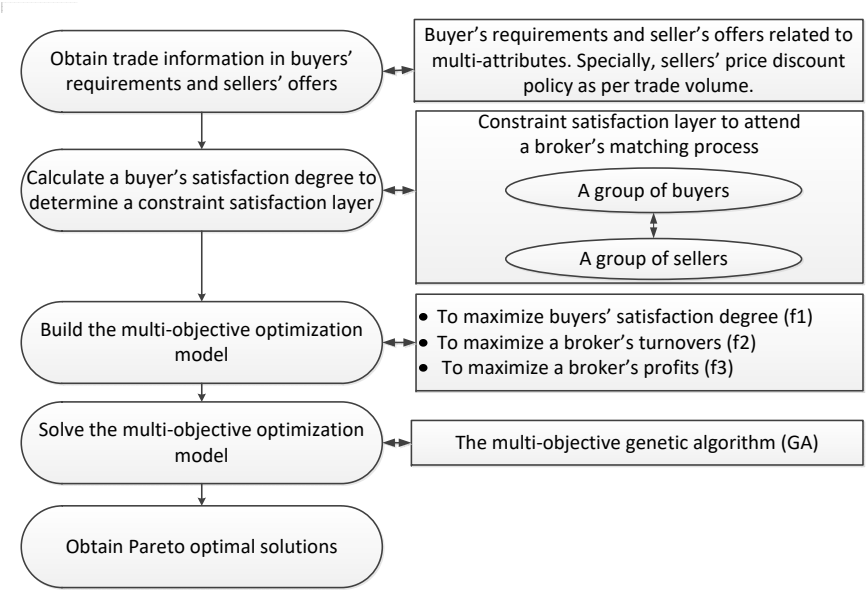


Fig. 1 The framework of a broker-based matching approach

find Pareto-optimal allocation solutions. In the following subsections, the main issues of the proposed approach, i.e., modelling sellers' price function as per trade volume, calculating buyers' satisfaction degree, building the multi-objective function and solving the multi-objective function are presented in detail.

3.2 Modelling a seller's price function as per trade volume

In reality, a seller usually offers price discount policy for a buyer as per trade volume so that a seller would like to encourage a buyer to buy a large number of commodities. It means that a price per unit for a buyer will be decreased if a buyer's trade volume increases. In this paper, the functional relationship between B_i's trade volume, namely C_{iv}, and S_j's price unit, namely Q_{jp}, can be presented in Equation 5 as follows.

$$Q_{jp} = \begin{cases} R_1 \; Q_{jv}^1 \leq C_{iv} < Q_{jv}^2, \\ R_2 \; Q_{jv}^2 \leq C_{iv} < Q_{jv}^3, \\ \cdots \qquad \cdots \qquad , \\ R_z \; Q_{jv}^{z-1} \leq C_{iv} < Q_{jv}^z \end{cases} \tag{5}$$

where Q^1_{jv} is a seller's minimal trade volume, which is offered to a buyer and Q^z_{jv} is a seller's maximal trade volume, which is offered to a buyer. Due to a seller's price discount offers as per trade volume, Equation (5) indicates that $R_1 > R_2 > \ldots R_{z-1} > R_z$ and $0 \leq Q^1_{jv} < Q^2_{jv}, \ldots Q^{z-1}_{jv} < Q^z_{jv}$. It means that the larger the trade volume, the lower price as per one unit of a commodity.

3.3 Building the calculation of a buyer's satisfaction degree

A buyer's satisfaction degree plays an important role in multi-attributes trading between buyers and sellers through a broker. It helps a broker to maximize a buyer's satisfaction degree through match processes between buyers and sellers. In particular, a buyer's satisfaction degree is calculated to attributes with hard constraints called β_{ijh} and soft constraints called β_{ijk} as follows:

(i) for attribute type with hard constraints

$$\beta_{ijh} = \begin{cases} -1 & \text{if } C_{ih} \neq Q_{jh} \\ 1 & \text{if } C_{ih} = Q_{jh} \end{cases} \tag{6}$$

$\beta_{ijh} = -1$ means that a seller S_j does not match with a buyer B_i for attribute h and $\beta_{ijh} = 1$ means that a seller S_j matches with a buyer B_i for attribute h.

(ii) for attribute type with benefit soft constraints: if $C_{ik} > Q_{jk}$ then $\beta_{ijk} = -1$. It means that a seller j does not satisfied a buyer i. If $C_{ik} \leq Q_{jk}$, then β_{ijk} is calculated as follows:

$$\beta_{ijk} = (\frac{Q_{jk} - Q_{min-k} + \phi}{Q_{max-k} - Q_{min-k} + \phi})^t \tag{7}$$

where $t = \frac{C_{ik}}{Q_{min-k}}$, Q_{min-k} is the minimal value of seller in the set of values for the attribute a_k and Q_{max-k} is the maximal value of seller in the set of values for the attribute a_k. A value $t \in (0,1]$ helps a broker to carry out comparing a buyer's satisfaction degree when t is used to calculate β_{ijk}. $\phi = \frac{Q_{min-k}}{2}$, ϕ helps a broker to solve some special cases such as only one seller in e-marketplaces or $Q_{max-k} = Q_{min-k}$. β_{ijk} increases when Q_{jk} increases or C_{ik} decreases.

β_{ijk} means that a seller S_j matches with a buyer B_i for attribute k with a buyer's satisfaction degree $(\frac{Q_{jk} - Q_{min-k} + \phi}{Q_{max-k} - Q_{min-k} + \phi})^t$. β_{ijk} is in-between 0 and 1. If β_{ijk} is near 1, it means that B_i is highly satisfied by S_j for attribute k.

(iii) for attribute type with cost soft constraints: if $C_{ik} < Q_{jk}$ then $\beta_{ijk} = -1$. It means that a seller j does not satisfied a buyer i. If $C_{ik} \geq Q_{jk}$ then β_{ijk} is calculated as follows:

$$\beta_{ijk} = (\frac{Q_{max-k} - Q_{jk} + \phi}{Q_{max-k} - Q_{min-k} + \phi})^{\frac{1}{t}} \tag{8}$$

β_{ijk} means that a seller S_j matches with a buyer B_i for attribute k with a buyer's satisfaction degree $(\frac{Q_{max-k}-Q_{jk}+\phi}{Q_{max-k}-Q_{min-k}+\phi})^{\frac{1}{t}}$. β_{ijk} is in-between 0 and 1. If β_{ijk} is near 1, it means that B_i is highly satisfied by S_j for attribute k.

β_{ijk} in this case increases when Q_{jk} decreases or C_{ik} increases.

In summary, a broker calculates B_i's satisfaction degree as per S_j's offers for attribute with hard constraints and soft constraints. The attributes with hard constraints are necessary conditions to be satisfied so they do not need to consider their weight. If attributes with hard constraints are not satisfied then B_i cannot match with S_j. On the other hand, attributes with soft constraints are necessary for using the weight because they can be relaxed with the given scope of values. In particular, B_i' satisfaction degree as per S_j for attributes with soft constraints is as follows.

$$\sum_{g=1}^{k} W_{ig}\beta_{ijg}, \tag{9}$$

where W_{ig} is a weight value of attribute A_g for B_i's requirements and $\sum_{g=1}^{k} W_{ig} = 1$.

3.4 Building a multi-objective function

A broker's decision making for a trade allocation process is driven by multi-objectives. Based on the definitions of a buyer, a seller and a broker (refer to Section 2), a proposed multi-objective function is presented by the three objectives (f_1, f_2, and f_3) and a set of constraints as follows.

$$f_1 = \sum_{i=1}^{n}\sum_{j=1}^{m}(\sum_{g=1}^{k} W_{ig}\beta_{ijg}x_{ij}) \tag{10}$$

$$f_2 = \sum_{i=1}^{n}\sum_{j=1}^{m} C_{iv}Q_{jp}x_{ij} \tag{11}$$

$$f_3 = \sum_{i=1}^{n}\sum_{j=1}^{m} Q_{jp}r_j^{S_j}C_{iv}x_{ij} \tag{12}$$

$$s.t. \sum_{i=1}^{n} x_{ij} \leq 1, \forall j \in m \tag{13}$$

$$\sum_{j=1}^{m} x_{ij} \leq 1, \forall i \in n \tag{14}$$

$$x_{ij} = 1, 0, \forall i \in n, \forall j \in m \tag{15}$$

$$\sum_{g=1}^{k} W_{ig} = 1, \ \forall i \in n \tag{16}$$

$$\sum_{i=1}^{n} C_{iv} x_{ij} \leq Q_{jv},\ \forall j \in m \tag{17}$$

$$x_{ij} = 0\ if\ \beta_{ijg} = -1,\ \beta_{ijg'} = -1,\ Q_{jv} < C_{iv},\ Q_{jp} > C_{ip}\ \forall g \in k,\ \forall g' \in h, \tag{18}$$

where C_{iv} is trade volume in B_i's requirements; C_{ip} is a expected price in B_i's requirements; Q_{jv} is trade volume in S_j's offers; Q_{jp} is a offer price in S_j's offers and x_{ij} is decision variables. The objective function in Equation (10) is established to maximize a buyer's satisfaction degree; the objective function in Equation (11) is established with the maximization of a broker's turnovers; and the objective function in Equation (12) is established with the maximization of a broker's profits. Constraints in Equation (13) are that each seller only matches with each buyer maximally; constraints in Equation (14) are that each buyer only matches with each seller maximally; constraints in Equation (15) are constraints of decision variable, if B_i matches with S_j, then $x_{ij} = 1$; otherwise, $x_{ij} = 0$. Constraints in Equation (16) denote the weight information of attributes with soft constraints in buyers' requirements; constraints in Equation (17) denote that S_j's trade volume is more than or equal to B_i's trade volume and constraints in Equation (18) determine a constraint satisfaction layer. Furthermore, the proposed multi-objective function can be solved by a multi-objective genetic algorithm [3].

4 Experiments

In this section, experimental results are illustrated to maximize a buyer's satisfaction degree, a broker's turnovers and a broker's profits using multi-objective optimization through allocating buyers' requirements to sellers' offers. In particular, the experimental setting is presented in Subsection 4.1; the experimental results are evaluated in Subsection 4.2; and discussions are presented in Subsection 4.3.

4.1 Experimental setting

In this experiment, we generate an artificial dataset of 5 buyers related to desktop's demand with model-Optiplex 960 and 10 sellers related to desktop's supply with model-Optiplex 960 in e-marketplaces. The detail contents of 5 buyers' requirements are presented in Table 1. Each desktop in a buyer's requirements contains 5 attributes, i.e., *price*, *trade volume*, *payment method*, *delivery time* and *warranty time*. Excepting *price* attribute and *trade volume* attribute, *payment method* is an attribute with hard constraints because their constraints are satisfied while attributes of *delivery time* and *warranty time* are attributes with soft constraints.

Table 1 Trading information of product in buyers' requirements

Buyer	Price (AUD)	Trade volume	Payment Method	Delivery time	Weight	Warranty time	Weight
B_1	200	20	PayPal	4	0.3	30	0.7
B_2	180	5	PayPal	5	0.4	35	0.6
B_3	190	15	PayPal	4	0.7	32	0.3
B_4	185	15	PayPal	5	0.45	26	0.55
B_5	210	11	PayPal	4	0.6	25	0.4

Similarly, each desktop in a seller's offers contains 6 attributes, i.e., *price*, *trade volume*, *payment method*, *delivery time*, *warranty time* and *rate of discount*. Each seller offers price discount as per a buyer's trade volume. Furthermore, all sellers agree that if their product is sold through a broker, the broker will receive a discount rate from sellers. The detail contents of 10 sellers' offers are presented in Table 2.

Table 2 Trading information of product in sellers' offers

Seller	Price (AUD)	Trade volume	Payment Method	Delivery time	Warranty time	Discount rate (r)
S_1	(180,170,150)	(0,18,22,40)	PayPal	3	37	10%
S_2	(170,150,140)	(0,12,18,30)	PayPal	4	38	1%
S_3	(160,150,140)	(0,8,15,25)	PayPal	4	39	3%
S_4	(165,160,155)	(0,9,13,25)	PayPal	4	40	7%
S_5	(175,170,160)	(0,15,20,25)	PayPal	4	41	3.5%
S_6	(168,160,155)	(0,12,16,22)	PayPal	3	40	8%
S_7	(175,170,160)	(0,20,30,35)	PayPal	3	42	11%
S_8	(180,175,160)	(0,15,20,28)	PayPal	3	40	12%
S_9	(160,160,155)	(0,16,18,25)	PayPal	3	35	2.5%
S_{10}	(180,180,175)	(0,8,15,25)	PayPal	3	38	9%

In this experiment, the proposed approach is evaluated under considering that a number of sellers is more than a number of buyers in e-marketplaces. More specifically, a broker's matching approach is tested to maximize a buyer's satisfaction degree, a broker's turnovers and a broker's profits based on a buyer's requirements in Table 1 and a seller's offers in Table 2.

4.2 Experimental results

A broker uses the proposed approach to maximize a buyer's satisfaction degree, a broker's turnovers and a broker's profits through allocating 5 buyers to 10 sellers in e-marketplaces. In order to evaluate the proposed approach, a multi-objective genetic algorithm developed by K.Deb [3] is used to solve a proposed multi-objective model related to a buyer's satisfaction degree, a broker's turnovers and a broker's profits. In particular, Fig. 2 shows all the feasible solutions in 23 generation computing with 1500 population size while f_1 is an objective function of a buyer's satisfac-

tion degree; f_2 is an objective function of a broker's turnovers and f_3 is an objective function of a broker's profits.

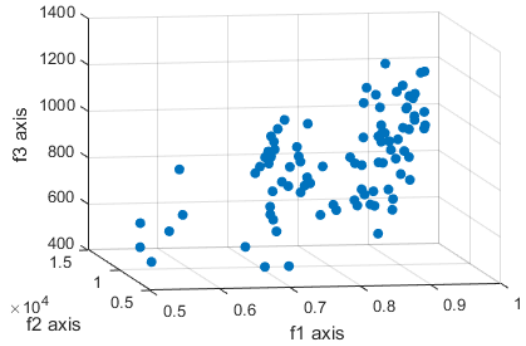


Fig. 2 Feasible solutions in 23 generation computing with 1500 population size

As shown in Fig. 3, the experimental results are converged in 60 generation computing so Pareto-optimal solutions are found out from all the feasible solutions.

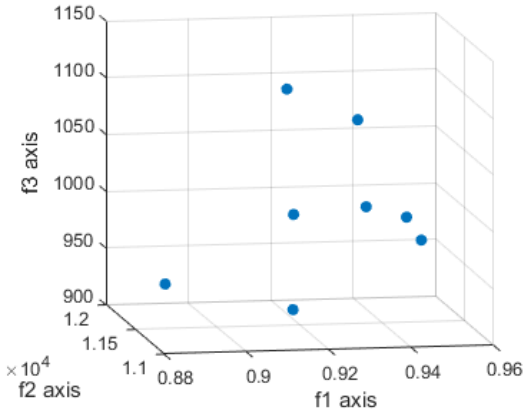


Fig. 3 Pareto-optimal solutions

The specific results of objective function values in Pareto-optimal solutions are presented in Table 3.

In summary, the proposed approach is perfectly performed under different situations in business environments. Based on specific situations in business environments, a broker should consider selecting which solutions in Pareto-optimal solutions to achieve a broker's goals. For instance, if a broker wants to choose maximizing a buyer's satisfaction degree in Pareto-optimal solutions, solution 6 in Table 3 is

Table 3 Objective function values in Pareto-optimal set

Pareto-optimal set	A buyer's satisfaction degree (f_1)	A broker's turnovers (f_2)	A broker's profits (f_3)
Solution 1	0.9150	AUD11,430	AUD1,112
Solution 2	0.9280	AUD11,120	AUD1,097
Solution 3	0.9180	AUD11,500	AUD997.8
Solution 4	0.9300	AUD11,130	AUD1,020
Solution 5	0.9440	AUD11,400	AUD997.3
Solution 6	0.9494	AUD11,500	AUD972.4
Solution 7	0.8874	AUD11,510	AUD939.1
Solution 8	0.9184	AUD11,510	AUD913.6

an optimal match solution. In particular, the optimal matching pairs between buyers and sellers in solution 6 are $B_1 \Leftrightarrow S_{10}$; $B_2 \Leftrightarrow S_6$; $B_3 \Leftrightarrow S_9$; $B_4 \Leftrightarrow S_7$; and $B_5 \Leftrightarrow S_8$.

4.3 Discussions

There has been a lot of previous work on regarding the indirect interaction between buyers and sellers through a broker in e-marketplaces. Jiang et al.[5] proposed an optimal allocation approach for a multi-attribute trading through a broker under simultaneously considering fuzzy information and indivisible demand. They firstly use fuzzy set theory to represent attributes in buyers' requirements and sellers' offers. Specifically, buyers and sellers' price offers can be presented under fuzzy information. Secondly, they proposed a method to calculate the matching degree based on the improved fuzzy information axiom. Finally, based on calculation results of the matching degree, they generate a multi-objective model under a multi-attribute trading with indivisible demand and develop a new algorithm to solve their model. However, their approach does not consider sellers' discount policy for a broker if a seller's products are sold to buyers through a broker. Li et al. [7] proposed a new method to match buyers and sellers through a third party, namely a matchmaker, in market environments by using a multi-objective optimization model. In particular, their multi-objective optimization model could help a matchmaker to maximize total satisfaction of buyers and sellers. They also proposed a new genetic algorithm to solve the multi-objective optimization model to find optimal matching pairs. However, their approach does not consider a broker's utility and sellers' discount policy to a broker.

To compare with the above approaches, the proposed approach in this paper addresses (i) considering sellers' discount policy for a broker if their products are sold to buyers through a broker; and (ii) building a multi-objective function to maximize a buyer's satisfaction degree; a broker's turnovers and a broker's profits based on buyers' requirements and sellers'offers in multi-attribute trading.

5 Conclusion and future work

This paper proposes a broker-based matching approach between a buyer's requirements and a seller's offers in multi-attribute trading using multi-objective optimization. The proposed approach is novel because (1) a framework is proposed to allocate buyers' requirements to sellers' offers based on sellers' price discount as per buyers' trade volume; (2) the proposed formula system is proposed to measure a buyer's satisfaction degree as per a seller's offers in multi-attribute trading; and (3) a multi-objective optimization model and a set of constraints are proposed to maximize a buyer's satisfaction degree, a broker's turnovers and a broker's profits. The proposed multi-objective function is solved by a multi-objective genetic algorithm to find out Pareto-optimal solutions. The experimental results demonstrate the good performance for the proposed approach in aspects of maximizing a buyer's satisfaction degree; a broker's turnovers and a broker's profits.

Future research includes extending the proposed matching approach to solve competition business environments between brokers and dynamic environments.

References

1. Alpár, F.Z.: Matchmaking framework for b2b e-marketplaces. Informatica Economica, **14**(4):164-170 (2010)
2. Badidi, E.: A broker-based framework for integrated sla-aware saas provisioning. International Journal on Cloud Computing: Services and Architecture (IJCCSA), **6**(2):1-19 (2016)
3. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE Transactions on Evolutionary Computation, **6**(2):182-197 (2002)
4. Giaglis, G.K., Klein, S., O'Keefe, R.M.: The role of intermediaries in electronic marketplaces: developing a contingency model. Information Systems Journal, **12**(3):231-246 (2002)
5. Jiang, Z.Z., Pan, Z.P., Ip, W., Chen, X.: Fuzzy multiobjective modeling and optimization for one-shot multiattribute exchanges with indivisible demand. IEEE Transactions on Fuzzy Systems, **24**(3):708-723 (2016)
6. Jiang, Z.Z., Tan, C., Chen, X., Sheng, Y.: A multi-objective matching approach for one-shot multi-attribute exchanges under a fuzzy environment. International Journal of Fuzzy Systems, **17**(1):53-66 (2015)
7. Li, X., Murata, T.: Priority based matchmaking method of buyers and suppliers in b2b e-marketplace using multi-objective optimization. In: Proceedings of the International Multi-Conference of Engineers and Computer Scientists, vol. 1 (2009)
8. Srivastava, N.K., Singh, P., Singh S.: Optimal adaptive csp scheduling on basis of priority of specific service using cloud broker. In: Proceedings of Industrial and Information Systems (ICIIS), pp. 1-6 (2014)
9. Standing, S., Standing, C., Love, P. E.: A review of research on e-marketplaces. Decision Support Systems, **49**(1):41-51 (2010)
10. Tiwari, A., Nagaraju A., Mahrishi, M.: An optimized scheduling algorithm for cloud broker using adaptive cost model. In: Proceedings of Advance Computing Conference (IACC), pp. 28-33 (2013)
11. Jrad, F., Tao, J., Streit, A., Knapper, R., Flath, C.: A utility–based approach for customised cloud service selection. International Journal of Computational Science and Engineering, **10**(1-2):32-44 (2015)

Uncovering the structure of a social network

Bryan Wilder, Nicole Immorlica, Eric Rice, and Milind Tambe

Abstract Models of social networks can guide policy decisions, but using them requires data about the social structure of a given population. In many applications, practitioners intervene in a population whose social network is initially unknown and must gather this data via a laborious process. We investigate how, given a model of an unknown social network, we can gather the least amount of data needed to make good decisions using that model. As a concrete setting for this issue, we introduce an extension of the influence maximization problem called *exploratory influence maximization*, in which an algorithm queries individual network nodes to learn their links. The goal is to locate a seed set nearly as influential as the global optimum using very few queries. Real world networks typically have community structure, in which nodes are arranged in densely connected subgroups. We present the ARISEN algorithm, which leverages community structure to find an influential seed set with a small number of queries. Experiments on real world networks of homeless youth, village populations in India, and others validate ARISEN's performance.

An expanded version of this paper is under review at another conference.

1 Introduction

Models and simulations of social systems can inform policy decisions: how best to intervene in a system to bring about a desired outcome. This work focuses in particular on social networks, and how network structure can be used to shape the behavior of a population. There are a variety of models for networks and the spread of social influence. However, in order to use these models in a particular application, practi-

Bryan Wilder, Eric Rice, Milind Tambe
University of Southern California, e-mail: {bwilder, ericr, tambe}@usc.edu

Nicole Immorlica
Microsoft Research, New England, e-mail: nicimm@gmail.com

tioners must elicit data. For instance, they must survey members of the population to obtain at least some of the links of the network. Once some data is available, simulations can help guide the design of effective interventions. Since data acquisition is a time-consuming and costly process, we study the following question: *using a model of a social network, can we reduce the amount of data that must be gathered in order to make good decisions?*

We answer this question in the specific context of the *influence maximization* problem [9, 24, 13]. In contexts ranging from health, to international development, to education, practitioners have used the social network of their target population to rapidly spread information and to change behavior in socially desirable ways. The challenge is to identify the influential members of the population, a task referred to as influence maximization. While previous work has delivered many computationally efficient algorithms for this problem, this work assumes that the social network is given explicitly as input. However, in many real-world domains, the network is not initially known and must be gathered via laborious field observations. For example, collecting network data from vulnerable populations such as homeless youth, while crucial for health interventions, requires significant time spent gathering field observations [22]. Social media data is often unavailable when access to technology is limited, for instance in developing countries or with vulnerable populations. Even when such data is available, it often includes many weak links which are not effective at spreading influence [4]. For instance, a person may have hundreds of Facebook friends who they barely know. In principle, the entire network could be reconstructed via surveys, and then existing influence maximization algorithms applied. However, exhaustively reconstructing the network is very labor-intensive and considered impractical in many situations [25]. For influence maximization to be relevant to many real-world problems, it must contend with limited *information* about the network, not just limited *computation*.

The major informational restriction is the number of nodes which may be surveyed to explore the network. Thus, a key question is: *how can we find influential nodes with a small number of queries?* Existing field work uses heuristics, such as sampling some percentage of the nodes and asking them to nominate influencers [25]. We formalize this problem as *exploratory influence maximization* and seek a principled algorithmic solution, i.e., an algorithm which makes a small number of queries and returns a set of seed nodes which are approximately as influential as the the globally optimal seed set. To the best of our knowledge, no previous work directly addresses this question from an algorithmic perspective (we survey the closest work in Section 3).

We show that for general graphs, any algorithm for exploratory influence maximization may perform arbitrarily badly unless it examines almost the entire network. However, real world networks have useful structure. In particular, social networks often have strong *community* structure, where nodes are arranged into groups which are connected tightly internally, but only weakly to the rest of the network [11, 18]. Consequently, influence mostly propagates in a local fashion. Community structure is the focal point for many models of social networks [2, 19, 14]. We focus on the simplest and most widespread of these, the *stochastic block model*. Using this model

as a guide, we leverage community structure to obtain a highly information-efficient algorithm. We make three main contributions. *First*, we introduce exploratory influence maximization and show that it is intractable for general graphs. *Second*, we present the ARISEN algorithm, which exploits community structure to find an influential seed set. *Third*, we show experimental results on a variety of networks (both synthetic and real) that verify ARISEN's performance. Our focus here is on introducing the algorithm and showing experimental results; theoretical analysis of ARISEN's performance will be presented in future work.

2 Exploratory influence maximization

As a motivating example, consider a homeless youth shelter which wishes to spread HIV prevention information [22]. It would try to harness the youths' social network and select the most influential peer leaders to spread information, but this network is not initially known. Constructing the network requires a laborious survey [22]. Our motivation is to mitigate this effort by querying only a few youth. Such queries require much less time than the day-long training peer leaders receive. We now formalize this problem and introduce the models that we consider for influence spread and network structure.

Influence maximization: The influence maximization problem [15], starts with a graph $G = (V, E)$, where $|V| = n$ and $|E| = m$. We assume throughout that G is undirected; social links are typically reciprocal [23]. An influencer selects K seed nodes with the aim of maximizing the expected size of the resulting influence cascade. We assume that influence propagates according to the independent cascade model (ICM), which is the most prevalent model in the literature. Initially, all nodes are inactive except for the seed set. When a node becomes active, it makes one attempt to activate each neighbor. Each attempt succeeds independently with probability q, where q is typically assumed to be the same for all edges [9, 15, 27]. Let $f(S)$ denote the expected number of activated nodes with seed set $S \subseteq V$. The objective is to compute $\arg\max_{|S| \leq K} f(S)$. Wherever necessary, we evaluated f by averaging over repeated simulations of the model [15, 24].

Local information: The edge set E is not initially known. Instead, the algorithm explores portions of the graph using local operations. We use the "Jump-Crawl" model [7], where the algorithm may either jump to a uniformly random node, or crawl along an edge from an already surveyed node to one of its neighbors. When visited, a node reveals all of its edges. We say that the *query cost* of an algorithm is the total number of nodes visited using either operation. Our goal is to find influential nodes with a query cost that is much less than n, the total number of nodes.

Stochastic Block Model: We use the SBM to model the underlying social network. The SBM originated in sociology [10] and lately has been intensively studied in computer science and statistics (see e.g. [1, 17, 21]). In the SBM, the network is partitioned into disjoint communities $C_1 \ldots C_L$. Each within-community edge is present independently with probability p_w and each between-community edge is

present independently with probability p_b. Notice that each community is an Erdős-Rényi random graph with additional random edges to other communities. We assume that $p_w \geq \frac{\log|C_i|}{|C_i|}$ for all C_i, since this is necessary for C_i to be internally connected [12]. While the SBM is a simplified model, our experimental results show that ARISEN also performs well on real-world graphs. ARISEN takes as input the parameters n, p_w, and p_b, but is not given any prior information about the realized draw of the network. It is reasonable to assume that the model parameters are known since they can be estimated using existing network data from a similar population (in our experiments, we show that this approach works well).

Objective: We compare to the globally optimal solution, i.e, the best performance if the entire network structure were known in advance. Let $f_E(S)$ give the expected number of nodes influenced by seed set S when the set of realized edges are E. Let $\mathscr{A}(E)$ be the (possibly random) seed set containing our algorithm's selections given edge set E. Let OPT be the expected value of the globally optimal solution which seeds K nodes. We measure the algorithm's performance by the ratio $OPT/\mathbb{E}[f_E(\mathscr{A}(E))]$, where the expectation is over both the randomness in the graph and the algorithm's choices.

3 Related work

First, Yadav et al. [27] and Wilder et al. [26], studied dynamic influence maximization over a series of rounds. Some edges are "uncertain" and are only present with some probability; the algorithm can gain information about these edges in each round. However, the majority of potential edges are known in advance. By contrast, our work does not require *any* known edges. Mihara et al. [20] also consider influence maximization over a series of rounds, but in their work the network is initially unknown. In each round, the algorithm makes some queries, selects some seed nodes, and observes all of the nodes which are activated by its chosen seeds. The ability to observe activated nodes makes our problem incomparable with theirs because these activations can reveal a great deal about the network and give the algorithm information that even the global optimizer does not have (in their work, the benchmark does not use the activations). Thus, we emphasize that our algorithm uses strictly less information. Further, in many applications, activations correspond to private behaviors (e.g. getting tested for HIV) and cannot be observed for practical or legal reasons.

Another line of work concerns local graph algorithms, where a local algorithm only uses the neighborhoods around individual nodes. Borgs et al. [5] study local algorithms for finding the root node in a preferential attachment graph and for constructing a minimum dominating set. Other work, including Bressen et al. [8] and Borgs et al. [6], aims to find nodes with high PageRank using local queries. These algorithms are not suitable for our problem since a great deal of previous work has observed that picking high PageRank nodes as seeds can prove highly suboptimal for influence maximization [16, 9, 13]. Essentially, PageRank identifies a set

of nodes that are *individually* central, while influence maximization aims to find a set of nodes which are *collectively* best at diffusing information. We also emphasize that our technical approach is entirely distinct from work on PageRank.

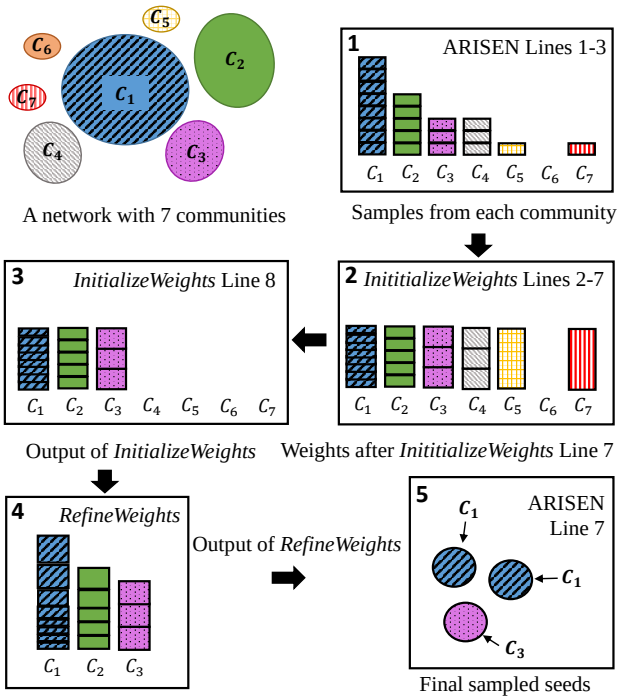


Fig. 1 Example run of ARISEN with $K = 3$. Each block is one sample, with current weight proportional to its height (e.g., in Frame 2, C_5 has one sample with very high weight).

4 Hardness result

We seek algorithms which examine a very small portion of the network, i.e., their query cost grows very slowly with n. The following shows that no algorithm with strictly sublinear query cost obtains a constant factor approximation for general graphs. The notation $o(1)$ refers to a term which goes to 0 as $n \to \infty$ (so $o(1)$ is smaller than every constant).

Theorem 1. *There exists a family of graphs on which any algorithm with query cost $O(n^{1-\varepsilon})$ for some $\varepsilon > 0$ has approximation ratio no better than $o(1)$*

Proof. Consider a family of graphs which consist of a star graph on $\log n$ nodes along with $n - \log n$ isolated nodes. Let $q = 1$ and $K = 1$. The algorithm gets utility

$\log n$ if it selects a node in the star, and utility 1 otherwise. The probability that finds any node in the star is $1-(1-\frac{\log n}{n})^{O(n^{1-\varepsilon})} \leq 1-e^{-\frac{\log n}{O(n^{\varepsilon})}}(1-\frac{\log^2 n}{n})^{O(n^{\varepsilon})} = o(1)$. Hence, its expected influence is $o(1)\log n+1$, while the optimal value is $\log n$, which gives an approximation ratio $\frac{o(1)\log n+1}{\log n} = o(1)$.

This motivates our focus on using properties of real-world networks, like community structure, to improve performance.

5 The ARISEN algorithm

We now introduce our main contribution, the ARISEN algorithm (*Approximating with Random walks to Influence a Socially Explored Network*); see Figure 1. The idea behind ARISEN (Algorithm 1) is to sample a set of T random nodes $\{v_1 ... v_T\}$ from G and explore a small subgraph H_i around each v_i by taking R steps of a random walk (Lines 1-3). We discard the first B steps of each walk as burn-in. R, T and B are inputs set by the user according to the size of the network and the number of seeds they wish to select. Intuitively, T should be greater than K so we can be sure of sampling each of the largest K communities. The subgraphs H_i are used to construct a weight vector $\mathbf{w}$ where w_i gives the weight associated with v_i (Lines 5-6). The algorithm then independently samples each seed from $\{v_1 ... v_T\}$ with probability proportional to $\mathbf{w}$ (Line 7).

The challenge is to construct weights $\mathbf{w}$ which balance two contradictory goals. First, we would oftentimes like to disperse the seed nodes throughout the network. For instance, if each community has equal size, we would like to seed K different communities. Second, we would other times like to place more seeds in large communities. For instance, if one community has 10,000 nodes and each other has only 100 nodes, we should focus on the large community. ARISEN navigates this tradeoff with the following ingredients (Algorithm 2). First, INITIALIZEWEIGHTS uses the random walk around each v_i to estimate the size of the community that v_i lies in. From these estimates, it constructs a $\mathbf{w}$ that, in expectation, seeds the largest K communities once each. Second, REFINEWEIGHTS tests if using a $\mathbf{w}'$ that puts more weight on large communities would increase the expected influence. The main novelty required is to carry out these steps using purely local information, since we will not generally be able to tell which of the v_i are in the same community.

We first formally describe the objective that ARISEN optimizes, which is a lower bound on its true influence. Let $f(X, C_i)$ denote the influence of seed set X on the subgraph C_i. ARISEN aims to optimize $\sum_{i=1}^{L} \mathbb{E}[f(X, C_i)]$, i.e., the influence spread within each community without considering between-community edges. We call this quantity the influence lower bound, denoted $ILB(X)$. When p_b is low and little influence spreads between communities (which is the case that we study), this is a good proxy for the true influence. We now explain ARISEN in detail, and how it optimizes the ILB.

5.1 Initial weights

Via the definition of the SBM, each community C_i has expected average degree $d_i = |C_i|p_w + (n - |C_i|)p_b$. By solving for C_i, this lets us estimate the size of the community from its average degree (Algorithm 2, Line 5). INITIALIZEWEIGHTS uses the nodes sampled in the random walk (after a burn-in period B) to estimate d_i. Since a random walk is biased towards high degree nodes, we use rejection sampling (Line 3) to obtain an unbiased estimate. A natural idea would be to choose the K samples with the largest estimated size as seeds. However, this idea fails because large communities will be sampled more often so we will likely seed the largest few communities many times, which is redundant. E.g., in the example in Figure 1, placing all of the seeds in C_1 would be suboptimal compared to also seeding C_2.

Algorithm 1 ARISEN

Require: R, T, B, K, n, p_w, p_b
1: **for** $i = 1...T$ **do**
2: Sample v_i uniformly at random from G
3: $H_i = R$ nodes on a random walk from v_i.
4: **end for**
5: **w =** INITIALIZEWEIGHTS($K, n, p_w, p_b, R, T, \mathbf{H}$)
6: $\mathbf{w}' =$ REFINEWEIGHTS(**w**, **H**)
7: Independently sample K nodes from **v** with probability proportion to $\mathbf{w}'$
8: **return** the sampled nodes

Complicating matters, the estimated sizes are only approximate, which rules out many simple ways of solving the problem. One solution is to weight each sample *inversely* to its size (Line 7), and then sample seeds with probability proportional to the weights. This inverse weighting evens out the sampling bias towards large communities. Using a weighted sampling scheme gives us a principled way to prioritize samples and facilitates later steps where we tune the weights to improve performance. In the example in Figure 1, all communities have the same total weight after this inverse weighting (Frame 2).

Next, the weights are truncated so that only the top K communities receive nonzero weight (Line 8). As a result, total weight approximately 1 is assigned to the samples from each of top K communities, and 0 weight assigned to smaller communities. In Figure 1, Frame 3 shows that only C_1, C_2 and C_3 have nonzero weight. The probability that each of the top K communities is seeded is thus nearly $1 - \left(1 - \frac{1}{K}\right)^K \geq 1 - 1/e$. Accounting for estimation error gives:

Proposition 1. *Let X be a seed set sampled according to the* **w** *output by* INITIALIZEWEIGHTS *and C_i be one of the top K communities. With probability at least*

$$1 - e^{-(1-\varepsilon)} - \varepsilon - \frac{1}{K} - o(1)$$

X hits either C_i or a community C_j with $|C_j| \geq (1-\varepsilon)|C_i|$.

This proposition states that with probability nearly $1-1/e$, the weights produced INITIALIZEWEIGHTS will hit either the top K communities, or communities nearly as large. Due to space restrictions, we omit proofs.

Algorithm 2 Weight computation

function INITIALIZEWEIGHTS($K, n, p_w, p_b, R, T, B, \mathbf{H}$)
  for $i = 1...T$ **do**
    Form H_i' by discarding the first B nodes of H_i and keeping each remaining node v_j w.p. $\frac{1}{d(v_j)}$
    $\hat{d} = \frac{1}{R}\sum_{u \in H_i'} d(u)$
    $\hat{S}_i = \frac{\hat{d} - p_b n}{p_w - p_b}$
  end for
  $w_j = \frac{n}{\hat{S}_j T}$.
  $\tau = \max\{\hat{S}_j | \sum_{\{i|\hat{S}_i \geq \hat{S}_j\}} w_i \geq K\}$
  For any j with $\hat{S}_j < \tau$, set $w_j = 0$.
  return $\mathbf{w}$, $\mathbf{H}$
end function

function REFINEWEIGHTS($\mathbf{w}$, $\mathbf{H}$)
  for $i = 1...T$ **do**
    $v_i = \arg\max_{v \in H_i} f(v, H_i)$
    $w_i' = \frac{n\beta(\hat{S}_i)^2}{T f(s_i, H_i)}$
  end for
  sort $\mathbf{w}'$ in increasing order
  for $i = 1...T$ **do**
    while ESTVAL($2w_i', w_{-i}$) > ESTVAL($\mathbf{w}'$) **do**
      $w_i' = 2w_i'$
    end while
    $w_i' = \text{BinarySearch}(w_i', 2w_i')$
  end for
  return $\mathbf{w}'$
end function

5.2 Refining the weights

The initial weights suffice to obtain the approximation guarantee proved below and are the best possible for some networks. However, they are overly pessimistic in other cases, such as when some communities are much larger than others. In such cases, it would be better to focus more seed nodes on large communities. We now outline REFINEWEIGHTS, which tunes the weights produced by INITIALIZEWEIGHTS to account for such scenarios. In essence, REFINEWEIGHTS tries to exploit easier cases where some communities are much larger than others.

REFINEWEIGHTS has two components. First, it sets v_i to be the most influential node in the sampled subgraph H_i (instead of the random starting node). This step also sets w_i' according to v_i's influence instead of the estimated size of its community (Line 15). $\beta(x)$ is defined to be the solution of the equation $\beta + e^{-\beta p_w q x} - 1 = 0$; this quantity arises as the expected fraction of a community that can be influenced by a given seed node [12]. Asymptotically, the two weighting schemes are identical, but using influence spread instead of size is more accurate for small networks. Second, it successively modifies each element of $\mathbf{w}$. Starting with the weights corresponding to the largest communities, it asks whether the *ILB* would be increased by doubling the w_i under consideration (Line 19). If yes, we set $w_i = 2w_i$ and ask if it can be doubled again. If no, REFINEWEIGHTS performs a binary search between w_i and $2w_i$ to find the best setting (Line 22). Then, it moves on to the weight corresponding to the next smallest community. In the example in Figure 1, Frame 4 shows that the weights of samples from C_1 and C_2 have been increased. Each change is made only if it improves the *ILB*, so we have:

Proposition 2. *Let* $\mathbf{w}$ *the output of* INITIALIZEWEIGHTS *and* $\mathbf{w}'$ *be the output of* REFINEWEIGHTS. *Then,* $\mathbb{E}_{X\sim\mathbf{w}'}[ILB(X)] \geq \mathbb{E}_{X\sim\mathbf{w}}[ILB(X)]$.

The key difficulty is determining if each modification increases the *ILB*. In the ESTVAL procedure, we provide a way to estimate the *ILB* using only local knowledge:

Proposition 3. $\text{ESTVAL}(\mathbf{w}) = \mathbb{E}_{X\sim\mathbf{w}}[ILB(X)]$

We give the main idea here; details are omitted due to space restrictions. Take any seed set X. Note that the influence within each C_i depends only on nodes in $X \cap C_i$, which we write as X_{C_i}. So, ILB can be rewritten as $\sum_{i=1}^{L} \mathbb{E}[f(X_{C_i}, C_i)]$. If we knew X_{C_i}, then we could calculate $\mathbb{E}[f(X_{C_i}, C_i)]$ by simulating draws from the SBM for the unobserved portions of C_i. Concretely, let H_i be the subgraph observed in community C_i, with estimated size $\hat{S}_i$. We simulate the rest of C_i by adding $\hat{S}_i - |H_i|$ new nodes, with edges between them and H_i randomly generated from the SBM. This is sufficient to choose the best seed node within H_i, as in Line 14. For Line 19, we need to estimate the *ILB*. The obstacle is that we do not know which of the $v_1 \ldots v_T$ lie in the same community (since a node will contribute less influence if there is another seed from the same community). However, we do know (approximately) how many other times each community is sampled, and the (approximate) weight that those samples will receive, so a careful set of simulations can be used to calculate the *ILB*.

6 Experiments

We now present experiments on an array of datasets:

- homeless: Two networks (a and b) gathered from the social network of homeless youth in Los Angeles and used to study HIV prevention. 150-200 nodes each.

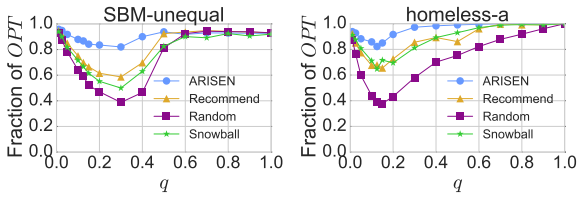


Fig. 2 Influence compared to OPT as q varies.

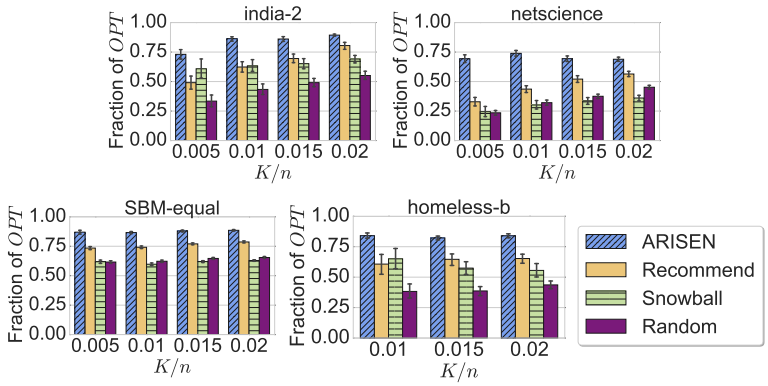


Fig. 3 Influence spread compared to OPT as K varies with $q = 0.15$.

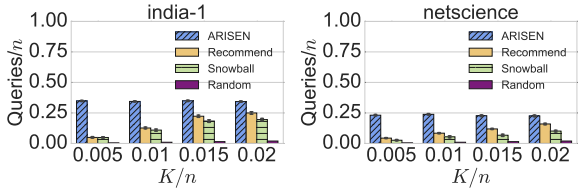


Fig. 4 Query complexity as K varies.

- india: Three networks of the household-level social contacts of villages in rural India. Gathered by Banerjee et al. [3] to study diffusion of information about microfinance programs. 250-350 nodes each.
- netscience[1]: a collaboration network of network science researchers; edges represents coauthorship. 1461 nodes.
- SBM: synthetic SBM graphs. SBM-equal has 10 communities of equal size ($p_w = \frac{4}{n}$, $p_b = \frac{0.2}{n}$) and SBM-unequal has 10 communities with size ranging from $\frac{1}{3}n$ to $\frac{1}{30}n$ ($p_w = \frac{6}{n}$, $p_b = \frac{0.2}{n}$). 1000 nodes each.

We focus on networks with about 100-1000 nodes because this is the size of real-world social groups of interest to us. We present results for ARISEN and three

[1] http://www-personal.umich.edu/ mejn/netdata/

benchmarks. First, *random*, which simply selects K nodes uniformly at random. Second, *recommend*, which for each of the K nodes, first queries a random node and then seeds their highest degree friend. Third, *snowball*, which starts from a random node and seeds that node's highest degree neighbor. It then seeds the highest degree neighbor of the first seed, and so on. We compare to *recommend* and *snowball* because these are the most common methods used in the field [25]. For each real network, p_w and p_b are estimated from a different network in the same category (for netscience, we use another collaboration network, astro-ph[2]). For the SBM datasets, we use another network from the same distribution. We present a cross-section of results across the datasets but the general trends are the same for all networks.

Our first set of results measures the influence spread of each algorithm against the optimal value. We approximate the optimal value using TIM [24], a state of the art influence maximization algorithm, run on the full network. As in previous work [9], we focus on when K is a small fraction of n. Figure 2 shows results as q is varied with $K = 0.01 \cdot n$. Each point averages over 50 runs for each algorithm. We see that ARISEN substantially outperforms all baselines, particularly when q is low. All differences between algorithms are statistically significant (t-test, $p < 10^{-7}$). Previous work [9] has also observed that when q is very high, influence maximization is sufficiently easy that nearly any algorithm performs well. Thus, Figure 3 presents results where K is varied with $q = 0.15$ fixed (since low q is when the problem is hard). We see that ARISEN uniformly outperforms the baselines, particularly when K is small. As K becomes larger, the baselines improve (again because the problem becomes easier). However, they are still outperformed by ARISEN.

Figure 4 examines each algorithm's query cost. ARISEN uses more queries than any of the baselines. However, its query cost is uniformly in the range $0.20 \cdot n - 0.35 \cdot n$, a relatively small portion of the network in absolute terms. Moreover, its query cost grows very slowly with respect to K. The baselines' query cost increases faster and for some networks nearly catches up with ARISEN by $K = 0.02 \cdot n$.

7 Conclusion

We introduced exploratory influence maximization to study influence maximization when the network is initially unknown. We presented the ARISEN algorithm, which exploits the community structure present in many real world social networks. Experimental results on a variety of networks show that ARISEN is competitive with the global optimum using a small number of queries.

References

1. Abbe, E., Sandon, C.: Community detection in general stochastic block models: Fundamental limits and efficient algorithms for recovery. In: FOCS, pp. 670–688. IEEE (2015)

2. Airoldi, E.M., Blei, D.M., Fienberg, S.E., Xing, E.P.: Mixed membership stochastic blockmodels. Journal of Machine Learning Research **9**(Sep), 1981–2014 (2008)
3. Banerjee, A., Chandrasekhar, A.G., Duflo, E., Jackson, M.O.: Gossip: Identifying central individuals in a social network. Tech. rep., National Bureau of Economic Research (2014)
4. Bond, R.M., Fariss, C.J., Jones, J.J., Kramer, A.D., Marlow, C., Settle, J.E., Fowler, J.H.: A 61-million-person experiment in social influence and political mobilization. Nature **489**(7415), 295–298 (2012)
5. Borgs, C., Brautbar, M., Chayes, J., Khanna, S., Lucier, B.: The power of local information in social networks. In: WINE, pp. 406–419. Springer (2012)
6. Borgs, C., Brautbar, M., Chayes, J., Teng, S.H.: Multiscale matrix sampling and sublinear-time pagerank computation. Internet Mathematics **10**(1-2), 20–48 (2014)
7. Brautbar, M., Kearns, M.J.: Local algorithms for finding interesting individuals in large networks. In: Innovations in Theoretical Computer Science, pp. 188–199 (2010)
8. Bressan, M., Peserico, E., Pretto, L.: The power of local information in pagerank. In: WWW, pp. 179–180. ACM (2013)
9. Chen, W., Wang, C., Wang, Y.: Scalable influence maximization for prevalent viral marketing in large-scale social networks. In: KDD, pp. 1029–1038. ACM (2010)
10. Fienberg, S.E., Wasserman, S.S.: Categorical data analysis of single sociometric relations. Sociological methodology **12**, 156–192 (1981)
11. Girvan, M., Newman, M.E.: Community structure in social and biological networks. PNAS **99**(12), 7821–7826 (2002)
12. Janson, S., Luczak, T., Rucinski, A.: Random graphs, vol. 45. John Wiley & Sons (2011)
13. Jung, K., Heo, W., Chen, W.: Irie: Scalable and robust influence maximization in social networks. In: ICDM, pp. 918–923. IEEE (2012)
14. Karrer, B., Newman, M.E.: Stochastic blockmodels and community structure in networks. Physical Review E **83**(1), 016,107 (2011)
15. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: KDD, pp. 137–146. ACM (2003)
16. Kimura, M., Saito, K., Nakano, R., Motoda, H.: Finding influential nodes in a social network from information diffusion data. In: Social Computing and Behavioral Modeling, pp. 1–8. Springer (2009)
17. Krzakala, F., Moore, C., Mossel, E., Neeman, J., Sly, A., Zdeborová, L., Zhang, P.: Spectral redemption in clustering sparse networks. PNAS **110**(52), 20,935–20,940 (2013)
18. Leskovec, J., Lang, K.J., Dasgupta, A., Mahoney, M.W.: Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. Internet Mathematics **6**(1), 29–123 (2009)
19. Li, C., Maini, P.K.: An evolving network model with community structure. Journal of Physics A: Mathematical and General **38**(45), 9741 (2005)
20. Mihara, S., Tsugawa, S., Ohsaki, H.: Influence maximization problem for unknown social networks. In: ASONAM, pp. 1539–1546. ACM (2015)
21. Mossel, E., Neeman, J., Sly, A.: Reconstruction and estimation in the planted partition model. Probability Theory and Related Fields **162**(3-4), 431–461 (2015)
22. Rice, E., Tulbert, E., Cederbaum, J., Adhikari, A.B., Milburn, N.G.: Mobilizing homeless youth for HIV prevention. Health education research **27**(2), 226–236 (2012)
23. Squartini, T., Picciolo, F., Ruzzenenti, F., Garlaschelli, D.: Reciprocity of weighted networks. Scientific Reports (2012)
24. Tang, Y., Xiao, X., Shi, Y.: Influence maximization: Near-optimal time complexity meets practical efficiency. In: KDD. ACM (2014)
25. Valente, T.W., Pumpuang, P.: Identifying opinion leaders to promote behavior change. Health Education & Behavior (2007)
26. Wilder, B., Yadav, A., Immorlica, N., Rice, E., Tambe, M.: Uncharted but not uninfluenced: Influence maximization with an uncertain network. In: AAMAS, pp. 740–748 (2017)
27. Yadav, A., Chan, H., Xin Jiang, A., Xu, H., Rice, E., Tambe, M.: Using social networks to aid homeless shelters: Dynamic influence maximization under uncertainty. In: AAMAS, pp. 740–748 (2016)