

# Intelligent Prediction Method for Transport Resource Allocation

Yan Kong\* and Shuzhen Pan\*\*

Jiangsu Engineering Center of Network Monitoring,  
Nanjing University of Information, Science and Technology, Nanjing 210044, China

(Received February 19, 2019; accepted April 18, 2019)

**Keywords:** cooperative resource allocation, recurrent neural network, linear programming

In transport environments, the resources are owned by independent organizations that always need to work cooperatively to finish some tasks to achieve social welfare. However, social welfare may not be achieved owing to the absence of required resources. In addition, traffic is not regular enough to deal with all of the unpredicted and dynamically occurring tasks, and the unpredictability and dynamism of tasks are big challenging issues in resource allocation. Focusing on these challenges, in this paper, we propose a prediction model based on backpropagation neural network (BPNN) learning-based resource requirement prediction and linear programming, which address the resource requirement and the cooperation of resources, respectively. Evaluation results proved that the proposed prediction-based model was efficient for applications in resource utility.

## 1. Introduction

The allocation of traffic management resources is a type of resource allocation in dynamic and cooperative environments. In such a type of resource allocation, resources change unpredictably and dynamically owing to the continuous changes in some factors. In addition, resources always belong to more than one independent organization/person that have to work cooperatively to achieve some social welfare. For example, when a traffic accident occurs, traffic policemen, ambulance workers, and even fire fighters, who are managed by different organizations, may need to cooperatively work to save the victims of an accident. Resource allocation in dynamic and cooperative environments has also been studied in the contexts of both applications and research. For example, in research fields, unmanned vehicles<sup>(1)</sup> and RoboCop rescue operations<sup>(2–4)</sup> have been widely studied in recent years, whereas supply chain formation<sup>(5–9)</sup> has been paid much attention in application fields.

Nowadays, a conventional resource allocation method in a transport rescue system involves the allocation of resources after the occurrence of traffic accidents, which is very inefficient for certain emergency situations. The time consumed by these allocation schemes is huge, which is not suitable for real-world situations. For example, when personnel injuries occur in one traffic accident that is far away from the hospital, the resources from a hospital for rescue cannot meet the existing needs.<sup>(10)</sup> After these problems have been considered by scholars in

---

\*Corresponding author: e-mail: kongyan4282@163.com

\*\*Corresponding author: e-mail: shuzhen\_pan@126.com

<https://doi.org/10.18494/SAM.2019.2339>

the design of intelligent transportation systems, many algorithm models that can be used to predict traffic accidents have been proposed. These models are based on conventional machine learning algorithms and so on, but they still have unpredictable room for improvement in terms of prediction accuracy and response speed.

There are several challenging issues in the traffic management resource allocation. The following are two important issues addressed in this paper: how to achieve an efficient utility of the limited traffic management resources under dynamic and unpredictable traffic conditions<sup>(11,12)</sup> and how to coordinate independent organizations managing resources to make them cooperatively achieve the maximum social welfare, especially when the resources are not enough.

The following are the objectives of this study: to predict traffic accidents and accordingly preallocate the traffic management resources in highly dynamic traffic environments by considering the weather conditions, working days/holidays, rush hours, and so forth.

The remainder of this paper is organized as follows. The problem formalization of the backpropagation neural network (BPNN) learning-based accident prediction and the linear programming-based resource coordination are given in Sect. 2. The experimental evaluation is presented in Sect. 3, and a related work is given in Sect. 4. This paper concludes in Sect. 5.

## 2. Problem Formalization and Solving

In current traffic systems, the management resources are always preallocated and, only when traffic accidents occur, some required resources are rapidly allocated to minimize the loss caused by such accidents. Actually, traffic situations are affected by several factors, the analysis of which comprehensively provide traffic situations to some degree. Even if data are not available for all these factors, some are readily available, such as weather conditions, holidays/workdays, and rush hours. All of these factors are known beforehand, and thus can be used to help predict traffic situations and accordingly preallocate the limited traffic management resources in a timely manner. In this way, the loss caused by traffic accidents could be decreased by preassigning the resources. In addition, the preallocation of resources can improve the utility efficiency of the limited resources through an efficient coordination model that will be introduced in the next section.

In this study, we adopt a supervised learning method to predict the possibility of an accident by constructing and supervising an artificial neural network (ANN) model, which is shown in Fig. 1. In particular, we adopt BP with the SoftMax function as the output function of the output layer to predict both the possibility of an accident occurring in a location given the following five factors: weekend/workday, fog, rain, visibility, and wind speed. An ANN is a collection of connected nodes or units called artificial neurons. Each connection between any two artificial neurons can transmit a signal from one to the other. The artificial neuron that receives the signal can process and then transmit the signal to other artificial neurons connected to it. Backpropagation is a method used in artificial neural networks to calculate a gradient that is required in the calculation of the weights to be used in the network.<sup>(13)</sup> It is commonly used to train deep neural networks,<sup>(14)</sup> which are networks with more than one hidden layer.

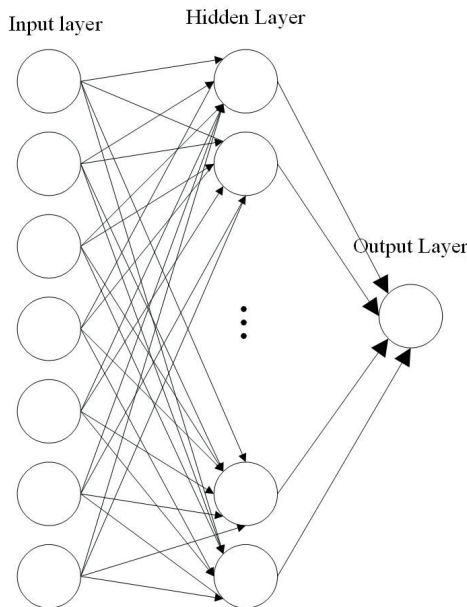


Fig. 1. ANN model.

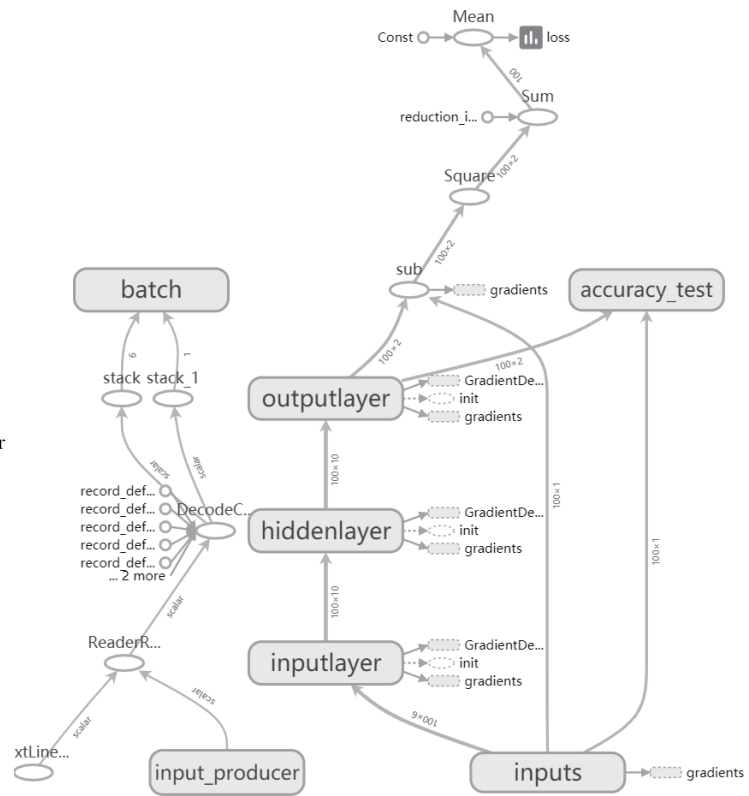


Fig. 2. Structure of the ANN model.

“In the input layer of the ANN model, there are seven nodes that accept seven characters. On the basis of the method used for traffic accident prediction in the above section, we can predict the probable reward of traffic management resource allocation. In particular, the preallocation of resources to the location where the possibility of an accident is high will obtain a high reward (welfare).

### 3. Experimental Evaluation

In this section, the experimental evaluation of the proposed model is introduced. The main part of the proposed model is based on the traffic prediction model called the BPNN, the structure of which is shown in Fig. 2.

The activation functions used in the three layers are sigmoid, relu, and SoftMax (Fig. 2). The SoftMax function in the output layer is used to predict the probability of an accident. In addition, the shape of batch data fed into the input layer is  $[100 \times 6]$  and the shape of data transferred to the hidden layer is  $[100 \times 10]$ . In the training phase, we use the batch gradient descent (BGD) to obtain the minimum value of the least squares loss function. We divide the training data into 10 batches for a rapid training phase. After developing the prediction model, we test the model by taking some data with the same distribution as the training data into the model and computing the test accuracy in 10 batches.

Then, we elaborate the environment of our experiment. The proposed model is simulated by using the Python programming language with the TensorFlow library on a Windows OS with Intel Core i5 processors running at 2.60 GHz and 8 G memory.

The main goal of the deep learning model is to find the most suitable weight to minimize the loss function. Then, we simulate the chart of changes in weight in the three layers in our model. The histogram charts of simulation results are shown in Fig. 3.

From these three subfigures, we observe that the changes in weight in the three layers are not the same. The horizontal axis shows the weight and the vertical axis shows the steps. The darker the color, the earlier the finish time, and the lighter the color, the later the finish time. In other words, the lower chart is closer to the end of the training phase.

Finally, we simulate the chart of changes in loss function. We use the least-squares loss function as the loss function in our model. Here, we give the accuracy and loss chart with different smoothing parameters such as 0.6, 0.95, and 0.99. The final results are given in Figs. 4–6.

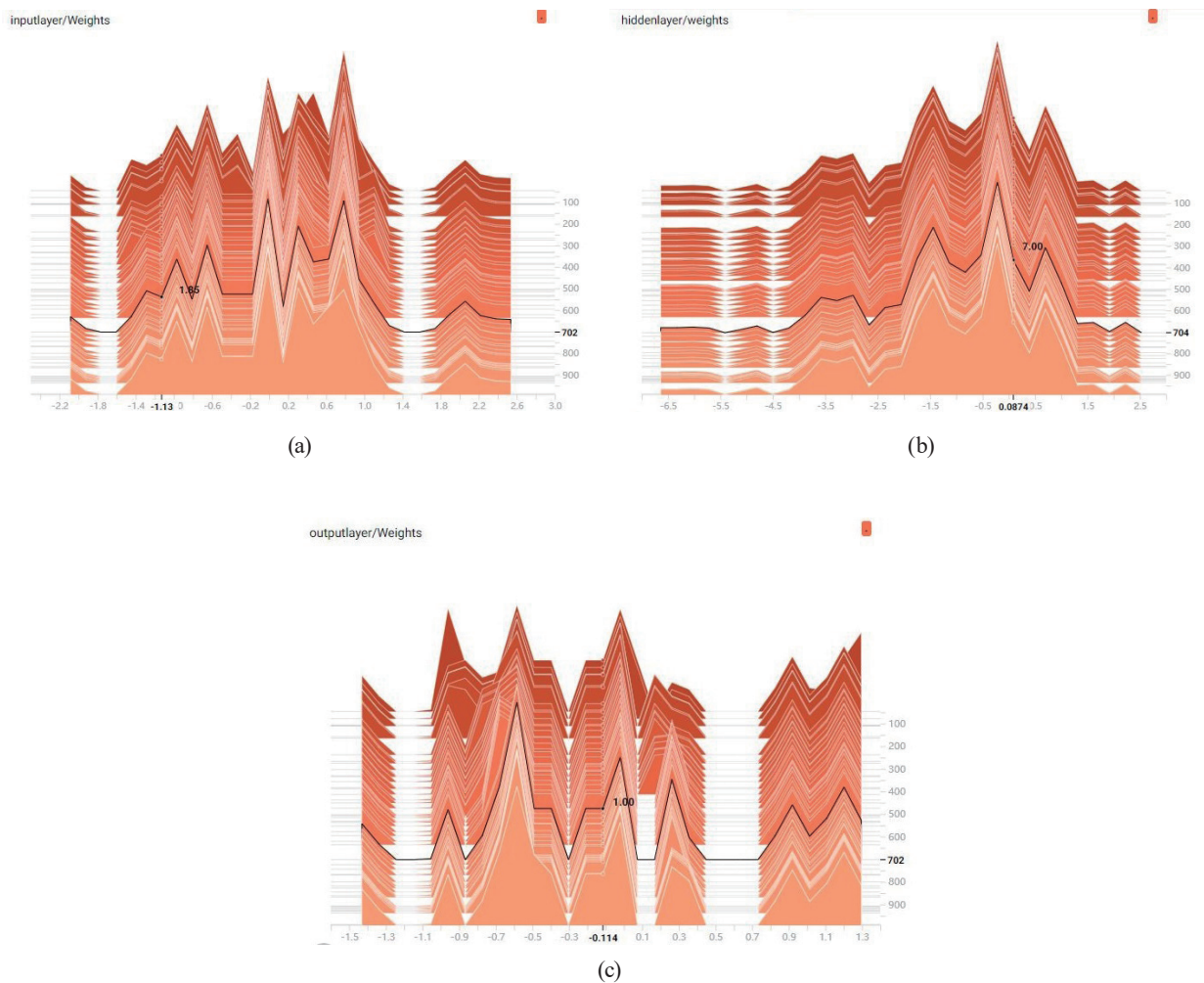
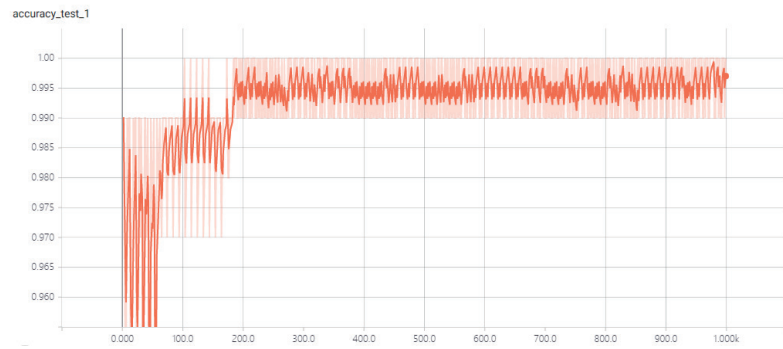
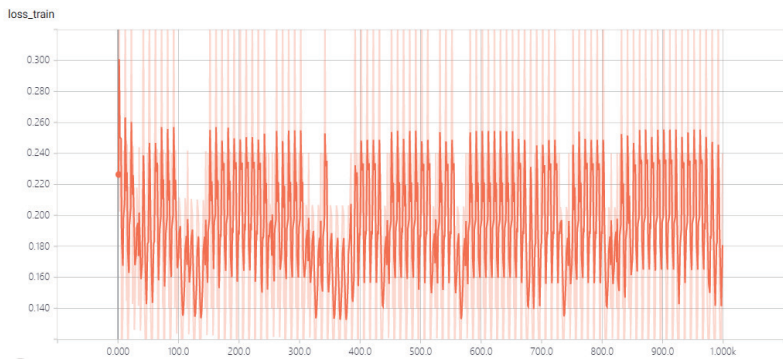


Fig. 3. (Color online) Weights of (a) input, (b) hidden, and (c) output layers during the learning process.

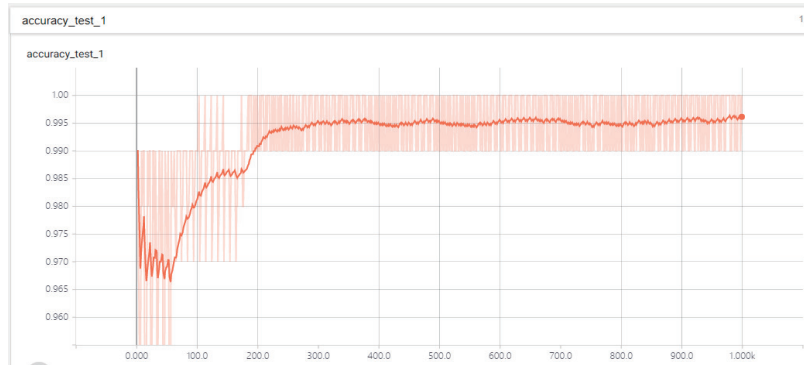


(a)

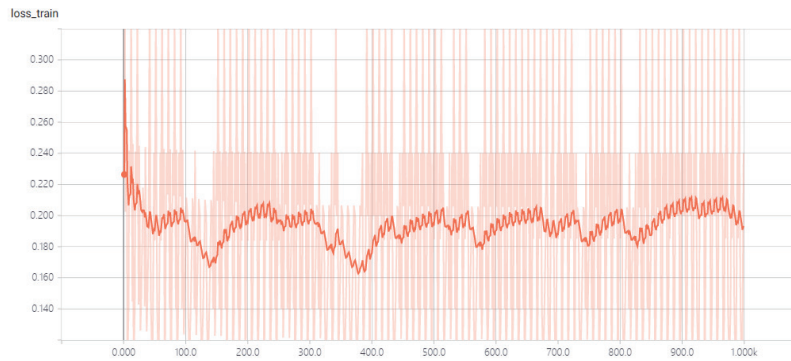


(b)

Fig. 4. (Color online) Smoothing is 0.6. (a) Accuracy and (b) loss.



(a)



(b)

Fig. 5. (Color online) Smoothing is 0.95. (a) Accuracy and (b) loss.

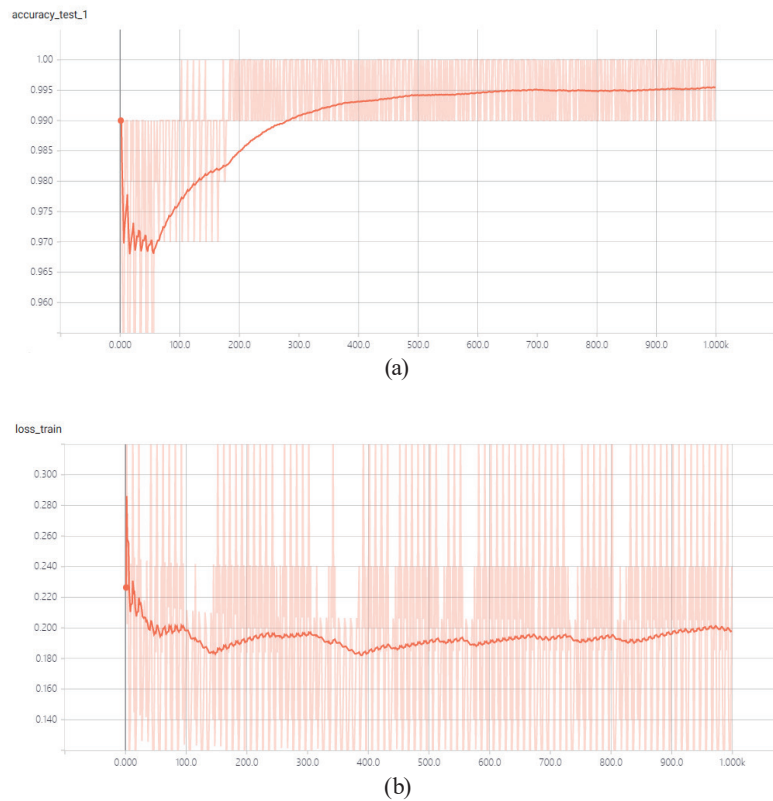


Fig. 6. (Color online) Smoothing is 0.99. (a) Accuracy and (b) loss.

From Figs. 4(a), 5(a), and 6(a), we determine that the average accuracy of our model is 99.5% when the number of steps is sufficiently large. In addition, the average loss of our model is lower than 0.19. This means that our model has a higher accuracy with a lower loss. The high accuracy of the accident prediction model means that the model gives the same result as the label during accidents. This is the most important target for testing our prediction model. From these figures, our accuracy is determined to be 99.5%, which meets the real requirement of the society.

## 4. Related Work

Recently, the main task allocation methods in the competitive and thus competitive cloud environments has included auction, negotiation, and belief propagation-based methods.

### 4.1 Auction-based methods

In recent years, many auction-based task allocation methods, such as strategic combinatorial auctions where providers quote strategically<sup>(6)</sup> and combinatorial auction-based methods that adopt integer programming to solve the NP-hard problem of group formation, have been proposed.<sup>(15)</sup> The combinatorial auction-based method for supply chain formation proposed

by Walsh *et al.*,<sup>(6)</sup> which concentrates on assembling resources needed by a task, allows for strategic quotes of resource providers. The problem with their method is that the auctioneer of the combinatorial auction is a central controller. In real applications, however, it is difficult to find such an auctioneer trusted by self-interested agents.

#### 4.2 Negotiation-based methods

There are two main types of negotiation-based task allocation methods where the task needs multiple resources: sequential negotiation<sup>(16)</sup> and simultaneous negotiation.<sup>(17–19)</sup> In the sequential negotiation method, the consumer sequentially negotiates with providers for the needed resources. One of the problems with the sequential negotiation method is that when some subsequent negotiations fail, the already obtained resources become invalid. The other problem is that the consumer needs to make a negotiation agenda (i.e., the negotiation order) in advance.<sup>(16)</sup> In the simultaneous negotiation method, the consumer negotiates for all the needed resources in parallel. Simultaneous negotiations can provide the consumer chances to adjust the negotiation strategy for any thread according to the status of other threads, thus enhancing the chance of the consumer to obtain all the needed resources.

There are many dimensions of negotiation strategy research, and we introduce some well-known negotiation strategies along the dimension of elements taken into account for offer (counter-) proposals. In the negotiation model devised by Faratin *et al.*,<sup>(20)</sup> the offer (counter-) proposal decision making takes time and the behaviors of negotiation partners are taken into account. The negotiation strategy proposed by Sim and Choi<sup>(21)</sup> determines the offer (counter-) proposals based on, for example, resource competition and trading alternatives. In recent years, the offer (counter-) proposals in the negotiation strategy proposed by Gatti *et al.*<sup>(22)</sup> has considered the task deadline. In 2009, An *et al.* developed a negotiation strategy that takes both the task deadline and the reserve price of consumers into account.<sup>(23)</sup> They extended their previous work by taking one more element, the environment dynamism, into account.<sup>(17)</sup> However, because of competition and secrecy among consumers and providers, negotiation partners may not be willing to reveal their real information, such as negotiation deadlines, reserve prices, and the real costs of providers to finish tasks.

#### 4.3 Belief propagation-based methods

Belief propagation has been widely used and has seen great success in the area of task allocation for a long time, such as RoboCop rescue,<sup>(2)</sup> sensor networks,<sup>(24–26)</sup> and low-power embedded devices.<sup>(27)</sup> Agents in the above applications are cooperative and aim at maximizing the global profit or minimizing the global cost. In 2010, inspired by the task dependency network (TDN) model devised by Walsh and Wellman,<sup>(28)</sup> Winsper and Chli encoded task allocation in competitive environments into a TDN model and adopted max-sum belief propagation to allocate tasks.<sup>(29)</sup> The evaluation results in Ref. 29 demonstrated that the max-sum belief propagation outperformed the combinatorial auction in Ref. 28. In 2012, the work in Ref. 29 was improved by Penya-Alba *et al.* in terms of both memory and communication

requirements of belief propagation.<sup>(8)</sup> However, neither the work in Ref. 29 nor that in Ref. 8 considers the elements of competitive environments, such as resource competition and agents' strategic offer (counter-) proposals.

## 5. Conclusions

In this paper, we propose an intelligent agent-based task allocation method in competitive and thus insecure cloud environments. The most distinguishing contributions of such a method include the designed intelligent agents, the strategic quotes of agents during belief propagation, and the robustness of agents against the insecurity of environments. The evaluation results demonstrate that the proposed method achieves good performance in terms of the success rate of task allocation, efficiency, and the number of messages passed during task allocation.

## Acknowledgments

This work is supported by the National Natural Science Foundation of China (Fund No. 61602254), the Natural Science Foundation of Jiangsu Province, China (Fund No. BK20160968), the Startup Foundation for Introducing Talent of NUIST (No. 2015r050) from Nanjing University of Information, Science and Technology, China, and the Priority Academic Program Development of Jiangsu Higher Education Institution (PAPD).

## References

- 1 H. Q. Wang, J. X. Peng, and L. I. Ling-Ling: *Pattern Recogn. Artif. Intell.* **19** (2006) 764. <https://doi.org/10.1360/crad20061223>
- 2 S. Ramchurn, A. Farinelli, K. Macarthur, and N. Jennings: *Comput. J.* **53** (2010) 1447. <https://doi.org/10.1093/comjnl/bxq022>
- 3 P. R. Ferreira, Jr., F. Dos Santos, A. L. Bazzan, D. Epstein, and S. J. Waskow: *Auton. Agents Multi-Agent Syst.* **20** (2010) 421. <https://doi.org/10.1007/s10458-009-9087-8>
- 4 R. Nair, T. Ito, M. Tambe, and S. Marsella: A short note, in: *RoboCup 2001: Robot Soccer World Cup V*, Springer (2002) 751. [https://doi.org/10.1007/3-540-45603-1\\_129](https://doi.org/10.1007/3-540-45603-1_129)
- 5 H. S. Kim and J. H. Cho: *Decis. Support Syst.* **49** (2010) 77. <https://doi.org/10.1016/j.dss.2010.01.004>
- 6 W. E. Walsh, M. P. Wellman, and F. Ygge: *Proc. 2nd ACM Conf. Electronic Commerce (2000)* 260. doi:10.1145/352871.352900. <http://doi.acm.org/10.1145/352871.352900>
- 7 T. Peña-Alba: *Proc. AAMAS, Int. Foundation for Autonomous Agents and Multiagent Systems (2013)* 1447. <http://dl.acm.org/citation.cfm?id=2484920.2485269>
- 8 T. Peña-Alba, M. Vinyals, J. Cerquides, and J. A. Rodríguez-Aguilar: *AAAI* (2012).
- 9 Y. Kong, M. Zhang, and D. Ye: *Knowledge-Based Syst.* **115** (2016) 123. <https://doi.org/10.1016/j.knosys.2016.10.016>
- 10 C.-H. Hsia and Y.-K. Lin: *J. Chung Cheng Inst. Technol.* **46** (2017) 1.
- 11 C.-H. Hsia: *IEEE Sensors J.* **18** (2018) 790. <https://doi.org/10.1109/jssen.2017.2772799>
- 12 J.-M. Guo, C.-H. Hsia, K. Wong, J.-Y. Wu, Y.-T. Wu, and N.-J. Wang: *IEEE Trans. Vehicular Tech.* **65** (2016) 4023. <https://doi.org/10.1109/jssen.2017.2772799>
- 13 R. J. Erb: *Pharm. Res.* **10** (1993) 165. <https://doi.org/10.1023/A:1018966222807>
- 14 J. McCaffrey: *MSDN Mag.*
- 15 A. Giovannucci, M. Vinyals, J. A. Rodríguez-Aguilar, and J. Cerquides: *Proc. 7th Int. Joint Conf. Autonomous Agents and Multiagent Systems (2008)* 1071. <https://doi.org/10.1145/1402298.1402370>
- 16 S. S. Fatima, M. Wooldridge, and N. R. Jennings: *Artif. Intell.* **152** (2004) 1. [https://doi.org/10.1016/s0004-3702\(03\)00115-2](https://doi.org/10.1016/s0004-3702(03)00115-2)



- 17 B. An, V. Lesser and K. M. Sim: *Auton. Agents Multi-Agent Syst.* **23** (2011) 114. <https://doi.org/10.1007/s10458-010-9137-2>
- 18 K. M. Sim and B. Shi: *IEEE Trans. Cybern.* **40** (2010) 753. <https://doi.org/10.1109/TSMCB.2009.2028870>
- 19 K. M. Sim and B. Shi: *Advances in Agent-Based Complex Automated Negotiations* (2009) 177. [https://doi.org/10.1007/978-3-642-03190-8\\_9](https://doi.org/10.1007/978-3-642-03190-8_9)
- 20 P. Faratin, C. Sierra, and N. Jennings: *Rob. Auton. Syst.* **24** (1998) 159. [https://doi.org/10.1016/s0921-8890\(98\)00029-3](https://doi.org/10.1016/s0921-8890(98)00029-3)
- 21 K. M. Sim and C. Y. Choi: *IEEE Trans. Cybern.* **33** (2003) 188. <https://doi.org/10.1109/TSMCB.2002.805694>
- 22 N. Gatti, D. Giunta, and S. Marino: *Artif. Intell.* **172** (2008) 1119. <https://doi.org/10.1016/j.artint.2007.11.007>
- 23 B. An, N. Gatti, and V. Lesser: *Proc. Int. Joint Conf. Web Intelligence and Intelligent Agent Technology* (2009) 403. <https://doi.org/10.1109/WI-IAT.2009.185>
- 24 A. Rogers, D. D. Corkill, and N. R. Jennings: *IEEE Intell. Syst.* **24** (2009) 13. <https://doi.org/10.1109/mis.2009.22>
- 25 R. Stranders, A. Farinelli, A. Rogers, and N. Jennings: *Proc. 21st Int. Joint Conf. AI (IJCAI)* (2009) 299. <https://doi.org/10.1145/1402298.1402313>
- 26 C. Crick and A. Pfeffer: *Proc. 19th Conf. Uncertainty in Artificial Intelligence* (2002) 159. <https://doi.org/10.1371/journal.pone.0041827>
- 27 A. Farinelli, A. Rogers, A. Petcu, and N. Jennings: *Int. Joint Conf. Auton. Agents Multiagent Syst.* (2008) 12.
- 28 W. E. Walsh and M. P. Wellman: *J. Artif. Intell. Res.* **19** (2003) 513. <https://doi.org/10.1613/jair.1213>
- 29 M. Winsper and M. Chli: *Agent-Mediated Electronic Commerce* (2010) 1. <https://doi.org/10.3233/978-1-60750-606-5-1125>

## About the Authors



**Yan Kong** received her BSc degree in Computer Science and Technology from Henan Normal University, China in 2008, and her MSc and Ph.D. degrees from Zhengzhou University, China, and the University of Wollongong, Australia, respectively. Her research interests include machine learning, artificial intelligence, and cloud computing. (kongyan4282@163.com)



**Shuzhen Pan** received her BSc degree from Nanjing University of Information Science and Technology, China in 2016. Her research interests include resource allocation, deep learning, and cloud computing. Her recent research work focuses on the efficient allocation of resources in traffic forecast. (shuzhen\_pan@126.com)