A Game Theoretic Approach for Resource Allocation based on Ant Colony Optimization in Emergency Management

Zhiyong Wang
School of Electronics and information engineering
Tongji University
Shanghai, China
wzy19840102@163.com

Weisheng Xu
School of Electronics and information engineering
Tongji University
Shanghai, China

Jijun Yang
School of Economics and Management
Tongji University
Shanghai, China

Jiazhen Peng
School of Electronics and information engineering
Tongji University
Shanghai, China

Abstract—**In the context of multiple emergencies occurring simultaneously, the optimal allocation of relief resources to multiple emergency locations is a challenging issue in emergency management. This work presents a noncooperative complete information game model for resource allocation and an algorithm for calculating Nash equilibrium (NE). In this model, the players represent the multiple emergency locations, strategies correspond to possible resource allocations, and the payoff is modeled as function of the cost of allocation and the amount of resources requested. Thus, the optimal results are determined by the Nash equilibrium of this game. Then we design an improved ant colony optimization (ACO) algorithm to obtain the Nash equilibrium by introducing dynamic random search technique. Experimental results show that the proposed model is effective in optimizing resource allocation during multiple emergencies for emergency management decision support.**

Keywords-game therory; ant colony algorithm; resource allocation; Nash equilibrium;

I. INTRODUCTION

Emergency management is a suggested management strategy for approaching emergencies. The federal emergency management agency defines it as the process of preparing for, mitigating, responding to and recovering from an emergency". The resource allocation and scheduling problem has received significant attention in emergency management and several resource optimization models have been proposed in recent years [1]-[5]. However, many of these literatures only focus on global objective [1] [3] and ignore the competition among emergency locations nor social fairness when resource supply is limited. Since resources available to some emergencies may be limited, it is important to allocate resources from different resource centers to multiple emergencies in a fair manner.

Based on the features of the resources allocation during multiple emergencies, and using game theory as a tool of analysis, this paper proposes a n person noncooperative game model, where the players (emergency locations), each having a set of strategies as to the possible allocations, compete for the allocation of resources from a limited supply, with an objective of optimizing their payoff. A Nash equilibrium (NE) algorithm is employed as the optimization algorithm for allocation. The Nash equilibrium is considered to be a socially optimal solution that ensures the rational behavior of the players and generates a good quality optimal solution. In order to obtain Nash equilibrium, we design an improved ant colony optimization (ACO) algorithm using dynamic random search technique. Thus the optimal resources allocation results are determined by the Nash equilibrium. Finally simulation results are given to demonstrate the feasibility and availability of the model and the algorithm presented in this paper.

II. MODELING OF A NONCOOPERATIVE GAME

Certain assumptions have been taken into consideration before establishing the model of a noncooperative game in a multi-emergency management scenario. The assumptions are as follows:

- All emergencies occur in a time-overlapped manner. The requests of all Emergency locations are received simultaneously by the emergency management system. Thus all game players can use a static step.
- The proposed game theoretic framework assumes complete information, i.e., every player knows the information of all other players such as strategy spaces, resource requirements, critical level and so on. And the resource availability in the region is constant.
- We assume each emergency location to be a rational player and to try to receive the resources at the minimum cost.

A. Identification of game region

Before realizing a game, the emergency management system should identify resource requirements of the emergency

This work is supported by National Natural Science Foundation of China Project No. 70871091.

978-1-4244-4994-1/09/$25.00

locations and check if the requested resources are available in the game region. If not, the game region needs to be expanded such that the nearby resource centers are also incorporated in the game region. That is

$$\sum_{i=1}^{n} D_i \le \sum_{j=1}^{m} R_j$$

Where D_i represents the total number of resource units required by emergency location i. R_j denotes the amount of resources units available at resources center j.

B. Generation and pruning of strategies

- *Generation of strategies.* A recursive algorithm is used to generate the set of all feasible strategies of player i, S_i, $i = 1, 2 \cdots n$. The following constraint is applied during the process of generation of strategies. Considering *j*th strategy belonging to player i, i.e., $s_{ij} = \{r_{i1}^j, r_{i2}^j, \cdots, r_{im}^j\}$, it must satisfy the following constraint:

$$\sum_{q=1}^{m} r_{iq}^j \le D_i, i = 1, 2, \cdots, n$$

  Where r_{iq}^j is the number of resources contributed by resource center q when player i chooses its *j*th strategy. It is imperative that an emergency location should not be allocated resources more than it requires. Excessive allocation not only causes the unfair share of resources but also leads to wastage of resources.
- *Transportation speed.*

$$v_{ij}(t) = \mu(t) \bullet \bar{v}_{ij}(t). \tag{1}$$

  Where $\bar{v}_{ij}(t)$ denote the average speed between the player i (emergency location i) and the resource center j. $\mu(t)$ is the road quality parameter of the road between i and j.

- *Cost function of allocation.* The cost of allocation of unit resource to the player i is defined as:

$$c_{ij} = \begin{cases} L_i \bullet \dfrac{d_{ij}}{v_{ij}(t)} \\ \infty \end{cases}. \tag{2}$$

  In (2), d_{ij} denotes the distance of player i from the resource center j. L_i is the critical level of player i. The cost is infinite if the resource does not reside in game region.

- *Pruning of strategies.* With the increment of the amount of resources, the number of strategies will explode, which increases the computation complexity. Hence it is necessary to control the explosion of the strategy space by eliminating dominated strategies. Here we propose an approach that evaluates strategies in descending order of cost. In this paper, the cost of s_{ij} is defined as follows :

$$\text{cost}(s_{ij}) = \max\{r_{i1}^j / c_{i1}, r_{i2}^j / c_{i2} \cdots, r_{im}^j / c_{im}\}. \tag{3}$$

  The basic idea behind the sorting is ordering strategies in order to provide the greatest amount of resources at the minimal cost.

C. Formulation of payoff matrices

Payoff matrix of Player i is formulated as:

$$\begin{bmatrix} u_{i1}^1 & u_{i1}^2 & \cdots & u_{i1}^{h(i)} \\ u_{i2}^1 & u_{i2}^2 & \cdots & u_{i2}^{h(i)} \\ \vdots & \vdots & \ddots & \vdots \\ u_{ig(i)}^1 & u_{ig(i)}^1 & \cdots & u_{ig(i)}^{h(i)} \end{bmatrix}$$

Where u_{ij}^k is the payoff to player i for choosing a particular strategy s_{ij} when other players select a combination of strategies s_{-ik}. s_{-ik} denotes the *k*th combination of strategies of players other than player i, $s_{-ik} \in \{S_1, \cdots, S_{i-1}, S_{i+1}, \cdots, S_n\}$. $g(i)$ represents the total number of strategies of player i. $h(i)$ is the total amount of strategies combinations of other players for player i, $h(i) = \prod_{j=1, j \ne i}^{n} g(j)$.

The payoff to player i for choosing *j*th strategy when other players make their selections can be modeled as:

$$u_{ij}^k = \sum_{q=1}^{m} (r_{iq}^j / (c_{iq} + \Delta c_{iq})). \tag{4}$$

In (4), Δc_{iq} denotes an increment in cost of allocation from next resource centers when the sum of resources required by all players exceeds the number of resources available at resource center q, i.e., $\sum_{t=1, t \ne i}^{n} r_{tq}^k + r_{iq}^j > R_q$. Where r_{tq}^k denotes the number of resources contributed by resource center q to the player t when players other than i select *k*th combination of strategies s_{-ik}.

III. ALGORITHM DESIGN.

Ant colony optimization is a novel meta-heuristic approach inspired by the behavior of ants in nature that communicate with pheromones trails. It is proposed for solving hard combinatorial optimization problems and was first used to solve TSP problem [6], and has been successfully applied to other problems such as quadratic assignment problem, vehicle routing problem, path selection problem and so on. Recently, many researchers have focused attention on the extension of ACO to continuous optimization [7]-[9].

The improved ACO algorithm to obtain the NE of the noncooperative game is as follows:

Step1: Initialization. Set the value of fundamental parameters of the ACO algorithm including the maximum generation number $GenMax$, the number of iteration for local search $gMax$, and the number of ants M. Set the initial pheromone value $\tau(i)=0$. Set $k=0$ and r_0. The ant colony is randomly created within the feasible search space.

Step2: Global search. According to [10], ACO's fitness function is constructed as follows:

$$f(X)=\sum_{i=1}^{n}\max\{E_i(X\|s_{ij})-E_i(X),0\,|\,j=1,2\cdots,m_i\}. \quad (5)$$

Where $E_i(X)$ is the expected payoff to player i if all players select mixed strategies $(x_{j_1}^1,x_{j_2}^2,\cdots,x_{j_n}^n)$. $E_i(X\|s_{ij})$ is the expected payoff to player i if all players except player i select mixed strategies $(x_{j_1}^1,x_{j_2}^2,\cdots,x_{j_n}^n)$ and player i uses pure strategy s_{ij}. Following the definition and property of NE, the fitness function $f(X)$ attains its minimum value 0, if and only if the solution X is the NE X^*.

Transition probability is defined as:

$$p(i,j)=\frac{\tau(j)\Delta f_{ij}}{\sum_{j=1}^{m}\tau(j)\Delta f_{ij}},\ j\neq i. \quad (6)$$

Where $f(X_i)$ represents the fitness value of ant i, $\Delta f_{ij}=f(X_i)-f(X_j)$. $\tau(j)$ is the pheromone value of ant j.

Ant i moves towards ant j with the probability $p(i,j)$, using the following equation:

$$X_i=\alpha X_i+(1-\alpha)X_j. \quad (7)$$

Where α is a random number uniformly distributed in the interval $(0,1)$. Equation (7) allows the ants not only to search in the mixed strategy profile space but also to move towards the high pheromone with minimal fitness value.

Step3: Save X_{best}, f_{best}.

Step4: Reset the iteration counter $g=0$. Load X_{best}, f_{best}; $X_{current}=X_{best}$.

Step5: Local search by dynamic random search technique [11].

$$X_m^i(g+1)=X_m^i(g)+R^i. \quad (8)$$

Where $X_m^i(g)$ represents the mixed strategy belonging to player i of ant m at the gth iteration, $m=1,2\cdots,M$. R^i denotes a random vector generated from $[-r_k,r_k]$, R^i must satisfy the constraint $\sum_{j=1}^{m_i}R_j^i=0$, $i=1,2\cdots,n$. If $X_k^i(g+1)$ move outside the feasible search space $\Omega=\{X\,|\,x_j^i\geq 0, i=1,2\cdots,n; j=1,2\cdots,m_i\}$, we recalculate $X_k^i(g+1)$ by using $X_m^i(g+1)=X_m^i(g)+w_m^i\bullet R^i$ instead of (8), where $w_m^i\in[0,1]$ is the maximum step length that is used to limit $X_k^i(g+1)$ in the feasible space, hence the ant colony move within the space of all feasible strategies in a game throughout the iteration.

$f_{new}=f(X_{current}+R)$
If $f_{new}<f_{best}$, then $f_{best}=f_{new}$, and $X_{best}=X_{current}+R$, $g\leftarrow g+1$. Go to Step 6.
If $f_{new}<f_{current}$, then $f_{current}=f_{new}$, and $X_{current}=X_{current}+R$ $g\leftarrow g+1$. Go to Step 6.
If $f_{new}=f(X_{current}-R)$
If $f_{new}<f_{best}$, then $f_{best}=f_{new}$, and $X_{best}=X_{current}-R$. $g\leftarrow g+1$. Go to Step 6.
If $f_{new}<f_{current}$, then $f_{current}=f_{new}$, and $X_{current}=X_{current}-R$. $g\leftarrow g+1$. Go to Step 6.

Step6: If $g<gMax$, then go to Step 4.

Step7: Pheromone updating. After local search, the value of each ant is updated according to the rule:

$$\tau(i)=(1-\rho)\times\tau(i)+\frac{1}{\lambda+f(X_i)}. \quad (9)$$

Where ρ is a volatility factor between 0 and 1. λ is a very small number which avoids a divide by zero error.

Step8: $k\leftarrow k+1$, $r_k\leftarrow 0.9\times r_{k-1}$.

Step9: If $k\leq GenMax$ or $f(X_{best})>\eta$ (η is a very small positive number), then go to step 2; else the algorithm is terminated.

IV. COMPUTATIONAL RESULTS.

Matlab 7.6 is used to compile the program of the above algorithm in the paper. The parameters are as follows: $M=30$, $GenMax=1000$, $gMax=10$, $\eta=10^{-4}$, $r_0=0.9$.

Considering a multi-emergency scenario, $m = 4$, $n = 3$, here we focus the optimization of the allocation of one type of resource. The allocation of other types of resources can be implemented in the same way. To simplify the problem, we assume that the average speed of each road is identical, i.e., 0.5 kilometers per minute. The distances between resource centers and emergency locations have been listed in Table I. The resource requirements of the emergency locations and the resource availability of resource centers have been presented in Table II. Other parameter values are randomly generated by the computer.

TABLE I. DISTANCES BETWEEN RESOURCE CENTERS AND EMERGENCY LOCATIONS.

distance(kilometers)	Emergency location 1	Emergency location 2	Emergency location 3
Resource center 1	5	2	2
Resource center 2	1.5	3	3.5
Resource center 3	3.5	1	4
Resource center 4	3	5.5	1

TABLE II. EMERGENCY REQUESTS AND RESOURCES AVAILABILITY

	D	*R*	*Critical level*
Emergency location 1	9		1
Emergency location 2	6		2
Emergency location 3	5		3
Resource center 1		6	
Resource center 2		8	
Resource center 3		3	
Resource center 4		4	

The modified ACO algorithm is run 5 times to solve this game and obtain a unique pure-strategy NE after 39 generations on the average. Table III shows computational results.

TABLE III. COMPUTATIONAL RESULTS

NO	*Generations*	*Optimal fitness value*
1	35	9.3494e-005
2	40	8.3312e-005
3	42	8.0186e-005
4	37	9.7882e-005
5	44	7.5230e-005

The NE $s^* = (s_1^*, s_2^*, s_3^*)$ of this game is

$$s_1^* = (1,8,0,0)$$

$$s_2^* = (3,0,3,0)$$

$$s_3^* = (1,0,0,4) .$$

V. COCLUSITON AND FURURE RESEARCH.

In this paper, a game theoretic approach for resource allocation during multiple emergencies is proposed. Our approach is an effective way of modeling the competition of requirements among all emergency locations and achieving a fair allocation of resources that benefits all the emergency locations in the game. An improved ACO algorithm is proposed to attain the Nash equilibrium. Simulation results show the effectiveness and feasibility of the model and algorithm.

For Future research, we would like to build models taking more actual factors into account, such as weather conditions, traffic state, road capacity and so on. Furthermore, we plan to investigate the construction of the payoff function in more realistic scenarios. In addition, we seek to enhance the performance and efficiency of the algorithm as the dimension of the problem increases. Also incorporating the algorithm with other learning techniques to find multiple Nash equilibriums will be included in our future work.

REFERENCES

[1] R. Minciardi, R. Sacile and E. Trasforini, “Resource allocation in integrated preoperational and operational management of natural hazards,” Risk Analysis, vol. 29, pp. 62–75, Jan 2009.

[2] U. Gupta and N. Ranganathan, “Multievent crisis management using noncooperative multistep games,” IEEE Transactions on Computers, vol. 56, pp. 577–589, May 2007.

[3] J. B. Sheu, “An emergency logistics distribution approach for quick response to urgent relief demand in disasters,” Transportation Research Part E-Logistics and Transportation Review, vol. 43, pp. 687–709, Nov 2007.

[4] J. Zhang, S. Shen and R. Yang, “Preference-order-based game modeling of multiple emergency resource allocation,” Qinghua Daxue Xuebao/Journal of Tsinghua University, vol. 47, pp. 2172–2175, Dec 2007. (in Chinese）

[5] N. Ranganathan, U. Gupta, R. Shetty and A. Murugavel, “An automated decision support system based on game theoretic optimization for emergenc,” Journal of Homeland Security and Emergency Management, vol. 4, pp. 1–25, Jul 2007.

[6] M. Dorigo, L. M. Gambardella, “Ant colony system: a cooperative learning approach to the traveling salesman problem,” IEEE Transactions on evolutionary computation , vol. 1, pp. 53–66, April 1997.

[7] K. Socha, M. Dorigo, “Ant colony optimization for continuous domains,” European Journal of Operational Research, vol. 183, pp. 1155–1173, Mar 2006.

[8] Z. Wang, X. Hang, W. Xu, J. yang, “Solving nash equilibrium based on improved ant colony algorithm ,” unpublished. (in Chinese）

[9] O. Baskan, S. Haldenbilen and H. Ceylan, “A new solution algorithm for improving performance of ant colony optimization,” Applied Mathematics and Computation, vol. 211, pp. 75–84, May 2009.

[10] Qian Yu, Xian~jia Wang, “Evolution algorithm for solving nash equilibrium based on particle swarm optimization,” Journal of Northeastern University (Natural Science Edition), vol. 52, pp. 25–29, Feb 2006. (in Chinese）

[11] C. Hamzacebi, “Improving genetic algorithms' performance by local search for continuous function optimizatio,” Applied Mathematics and Computation, vol. 196, pp. 309–317, Feb 2008.