Aqun ZHAO, Qi ZHAO

School of Computer and Information Technology, Beijing Jiaotong University

A New Routing Algorithm for Multi-path Transmission

***Abstract**. A new multi-path routing algorithm called MA* algorithm is proposed which combines the path similarity objective with the heuristic method. A new evaluation function construction method is devised for the MA* algorithm, which is proved feasible through theoretical analysis. The simulation experiments are carried out to compare the MA* algorithm with other multi-path routing algorithms applying the network topology generation tool. The results show that the MA* algorithm can find multiple paths with reasonable path cost and path similarity at quite low search times.*

***Streszczenie.** W artykule zaproponowano algorytm MA*, do routowania wielościeżkowego, uwzględniający założenie podobieństwa ścieżek i metodykę heurystyczną. Opracowano nową metodę oceny budowy dla algorytmu MA*, której wykonalność została udowodniona teoretycznie. Badania dowodzą, że algorytm MA* w krótkim czasie potrafi wyszukać ścieżki mnogie o uzasadnionych kosztach ścieżek i podobieństwie. (**Nowy algorytm routowania do transmisji wielościeżkowej**).*

Keywords: multi-path routing; path similarity; A* algorithm; evaluation function
Słowa kluczowe: routowanie wielościeżkowe, podobieństwo ścieżek, algorytm MA*, funkcja oceny

Introduction

With the rapid development of network and communication technology, the bandwidth of network and the quality of transmission links are improved. However, due to the emergence and development of new applications such as P2P (Peer to Peer) and multimedia applications, the traffic grows much faster. The sharp increasing network traffic cannot be transmitted efficiently in the traditional single-path transmission mode [1]. Whereas the multi-path transmission can balance load, reduce delay, improve efficiency of bandwidth and enhance reliability and fault-tolerance capability, so it becomes an effectual way to solve the traffic increasing problem in the network [2].

The multi-path routing algorithms are the foundation to realize the multi-path transmission, which are responsible for finding paths for multi-path transmission. The recent proposed multi-path routing algorithms can be divided into two kinds. The first kind aims at the path cost, such as the pre-k shortest paths algorithm (KSP) [3] and the discount shortest paths algorithm. The second kind aims at the path similarity, such as the node-disjoint routing algorithm and the edge split algorithm (ESA) [4]. However, it may cause problems if just considering one aspect of path cost and path similarity when finding paths. In [5], an efficient path selection algorithm (EPS) was proposed, which calculates the path cost according to the path weight and the relationship with other paths. But this algorithm generates some redundant paths and uses an exhaustive way to get the shortest path, which leads the algorithm inefficient, even unrealizable when the state space is huge.

In this paper, a new multi-path routing algorithm based on the A* algorithm called MA* algorithm is proposed which combines the path similarity objective with the heuristic method. The main idea of this algorithm is that the evaluation function is calculated according to the cost between the source node and the current node, the similarity of the current path with the shortest path, and the heuristic value between the current node and the destination node. Then the node with the minimum value of the evaluation function is chosen for further search and the links repeatedly used in the shortest path are punished. In this way, the search for some unnecessary paths can be avoided and the paths with reasonable path cost and path similarity can be obtained.

MA* algorithm

The MA* algorithm is based on the A* algorithm, which is widely used in the area of urban road planning, vehicle navigation and game path searching to solve the shortest path finding problems [6,7]. The A* algorithm is an admissible and best-first algorithm, which uses the evaluation function to evaluate an arbitrary node to the destination node referring to some constraint conditions. The evaluation function is defined as follows.

$$f(N) = g(N) + h(N) \tag{1}$$

Where $f(N)$ is the evaluation function, $g(N)$ is the cost of the path from the source node to the node N, and $h(N)$ is the heuristic value from node N to the destination node.

The MA* algorithm is based on the A* algorithm and executes the A* algorithm more times during the path search process. For multi-path routing, the path similarity is an important evaluation standard to determine whether the path is reasonable. In the MA* algorithm, the evaluation function is defined according to the path cost and the path similarity to ensure that the path has both a lower path cost and a lower path similarity.

The procedure of the MA* algorithm is shown as follows.

1) Set k=0 and PATH= NULL.

2) Put the source node S into the OPEN table, and set the CLOSE table as NULL.

3) If the OPEN table is not NULL, choose the node N with the smallest value of the f(*). The link constructed by node N and its father node will be punished if it is also in the shortest path. Choose the node with the smallest value of the f(*) again. Then put the node into the CLOSE table. If the OPEN is NULL, the MA* algorithm fails.

4) If node N is the destination node, the path is found. Set k=k+1 and put the path into the PATH table. Then go to step 6). Otherwise go to step 5).

5) Calculate the evaluation function value of each subsequent nodes of N, and put them into the OPEN table if they are not in the CLOSE table. Then go back to step 3).

6) If k is equal to the number of the desired path, the algorithm terminates. Otherwise go back to step 3).

The path similarity is a metric used to measure the similarity between two paths [8]. The needed shortest path can be found according to the path similarity and some other constraints. During the MA* algorithm search process, the links which are constructed by the current node and its subsequent nodes are the candidate links. The search path is determined according to the path similarity and the cost between the current node and the destination node. The path similarity here means the overall overlapping degree between the current path and the optimal path.

Let n be the number of nodes in the network, m_i be the number of the subsequent nodes of the current node i. The path similarity S_{ij} between the current path passing through node i and the optimal path can be defined as follows.

(2) $$S_{ij} = \sum_{k=1}^{n} v_{ij}^{(k)} \Big/ d, \quad i = 1,2,...,n;\ j = 1,2,...,m_i$$

(3) $$v_{ij}^{(k)} = d(x_A^{(k)}, x_{ij}^{(k)}), \quad k = 1,2,...,n, i = 1,2,...,n, j = 1,2,...m_i$$

Where the parameter *d* is the length from the source node to the destination node in the optimal path, and $v_{ij}^{(k)}$ denotes whether the *j*-th link directly connecting with node *i* comparing to the link passing through the *k*-th node in the optimal path is overlap. If the link is overlap, $d(x_A^{(k)}, x_{ij}^{(k)})=1$. Otherwise $d(x_A^{(k)}, x_{ij}^{(k)})=0$.

The similarity matrix of node *i* can be defined as follows.

(4) $$Sim_i = \begin{pmatrix} v_{i1}^{(1)} & v_{i1}^{(2)} & \cdots & v_{i1}^{(n)} \\ v_{i2}^{(1)} & v_{i2}^{(2)} & \cdots & v_{i2}^{(n)} \\ \vdots & \cdots & \cdots & \vdots \\ v_{im_i}^{(1)} & v_{im_i}^{(2)} & \cdots & v_{im_i}^{(n)} \end{pmatrix}, \quad i = 1,2,...,n$$

The path similarity can be calculated according to the similarity matrix.

The evaluation function denotes the evaluation value from the source node to the destination node, on which the MA* algorithm is based to choose the search direction. So the evaluation function determines the efficiency of the algorithm.

The function value after each iteration indicates the next search direction, so in the design procedure of the evaluation function not only the path cost but also the path similarity should be considered. The estimation of the global information is based on the cost between the subsequent node and the destination node. Each overlap link will be punished after each iteration to make search direction closer to the path which has a smaller path similarity.

Equation (1) shows that the evaluation function consists of two parts: the path cost from the source node to node *i*, called *g(i)* and the heuristic value from node *i* to the destination node, called *h(i)*. The value of *g(i)* can be defined by the cost between the source node and node *i* and the path similarity. Meanwhile the value of *h(i)* can be defined by the cost between node *i* and the destination node. The evaluation function in the MA* algorithm is designed as follows.

(5) $$f(i) = W_{si} + \alpha_i S_i + W_{di} \quad i = 1,2,...,n, \alpha_i >= 0$$

Where W_{si} denotes the cost from the source node S to node *i*; S_i is the path similarity from S to *i*; W_{di} is the cost of the shortest path from the destination node D to *i*; α_l denotes the coefficient of path similarity which is designed to make the magnitude of the path similarity and the path cost be consistent.

In the MA* algorithm the heuristic evaluation function should satisfy two conditions: the admissibility and the monotonicity. The proposed heuristic evaluation function above will be proved to satisfy the two conditions as follows.

Admissibility: For any node *i*, there is $h(i) \leqslant h^*(i)$, where $h^*(i)$ is the actual path cost and $h(i)$ is the heuristic value of *i*.

Proof: The *h*(*i*) in the evaluation function is defined as W_{di}, which means the cost of the shortest path calculated by the Dijkstra algorithm or other algorithm.

For the first path the heuristic value is the cost of the shortest path, so $h(i)=h^*(i)$. Then $h(i) \leqslant h^*(i)$ is satisfied. For other paths, *h*(*i*) is smaller than $h^*(i)$ due to the punishment to the overlap links. Therefore, the proposed heuristic evaluation function satisfies the admissibility condition.

Monotonicity: For any node pairs (n_i,n_j), where n_j is the subsequent node of n_i, there is $h(n_i)-h(n_j) \leqslant c(n_i,n_j)$, where $c(n_i,n_j)$ is the cost from n_i to n_j.

Proof: There are two kinds of relationships between nodes n_i and n_j. Case 1: n_i and n_j are in the same shortest path and they are connected directly. Case 2: n_i and n_j are not in the same shortest path.

Case 1: Because nodes n_i and n_j directly connect, so there is $h(n_i) = h(n_j)+c(n_j,n_i)$; otherwise, for the undirected graph, there is $c(n_i,n_j)= c(n_j,n_i)$. So $h(n_i)-h(n_j) \leqslant c(n_i,n_j)$ is right. Because nodes n_i and n_j are connected directly, there is $h(n_i) = h(n_j)+c(n_j,n_i)$. Meanwhile, for the undirected graph, there is $c(n_i,n_j)= c(n_j,n_i)$. Therefore $h(n_i)-h(n_j) \leqslant c(n_i,n_j)$ is satisfied.

Case 2: Prove it by contradiction. Suppose $h(n_i)-h(n_j)>c(n_i,n_j)$, so $h(n_i)>c(n_i,n_j)+h(n_j)$. For the undirected graph, there is $c(n_i,n_j)= c(n_j,n_i)$, so $h(n_i) > h(n_j)+c(n_j,n_i)$, which denotes the cost of the path passing through node n_j is smaller than the cost calculating according to the shortest path. So n_i and n_j are in the same shortest path, which is contrary with the condition of Case 2. So there is $h(n_i)-h(n_j) \leqslant c(n_i,n_j)$.

Therefore, the heuristic evaluation function satisfies the monotonicity condition, so the MA* algorithm can find the needed paths in the condition of the proposed evaluation function.

In the MA* algorithm, the overlap links are punished according to the path similarity after each iteration. That is, the cost is added to the paths with overlap links. For a link *l*, the added cost is designed as follows.

(6) $$gc_l^{t+1} = gc_l^t (1 + \phi / m_i)^\eta$$

Where gc_l^t is the cost value of link *l* in the *t*-th iteration; ϕ is the overlap punishment parameter; m_i is the number of links passing through *i*; η denotes whether the link overlap with the optimal path. If they overlap, the value is 1. Otherwise the value is 0.

Simulations

In order to investigate the performances of the MA* algorithm, simulation experiments are carried out. In the experiments, BRITE (short for Boston University representative Internet topology generator) [9] is used to generate the topology of the networks with 100, 300, 500, 700 and 1000 nodes respectively. The source and destination nodes are chosen randomly in these topologies. The multi-path routing algorithms are executed to find two paths between the source and destination nodes. The KSP, ESA and EPS algorithms are selected to compare with the MA* algorithm. For the MA* algorithm, set $\phi' = \sum_{j=1}^{d_i} gc_{l_j}^k \Big/ gc_l^k$, $\phi = \frac{\phi'}{3}$, $\alpha_i = \min\{gc_{l_i}^k\}$ in (5) and (6).

The search times needed to obtain the two paths, the path cost and the path similarity of the two obtained paths are chosen as the criterions to evaluate the performance of the algorithms. The search times reflects the overhead to implement the algorithms, while the path cost and the path similarity indicate the performance of the algorithms. For the above four algorithms, the first path is same and the difference is on the second path. In the simulations, the search times, the path cost of the second path, and the path similarity between two paths are compared. Each experiment is repeated 100 times.

Fig.1 shows the path cost of different algorithms and Table 1 shows the path similarity of them. As we can see, the path cost and the path similarity are contrary with each other. The KSP algorithm aims at obtaining the shortest path and doesn't consider the overlap of the links, so the path cost is the smallest while the path similarity is the highest. The ESA algorithm requires the links splited and deletes all the links in the first path when finding the second path, so the path cost is the highest and the path similarity is 0. The EPS algorithm is a compromise between the KSP algorithm and the ESA algorithm, which considers both the path cost and the path similarity, so the paths with more reasonable path cost and path similarity can be obtained. The MA* algorithm proposed in this paper also considers both the path cost and the path similarity synthetically. In the process to find the second path, the overlap links are punished. The experiment results show that the performance of the MA* algorithm is close to the EPS algorithm. The path cost is 6.5% larger than the EPS algorithm while the path similarity is 18.5% lower than the EPS algorithm.

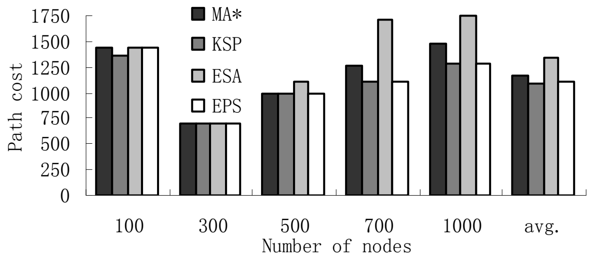

Fig.1. The path cost of different algorithms

Table 1. The parameters

Number of nodes	100	300	500	700	1000	avg.
MA*	0	0	0.25	0.25	0.6	0.22
KSP	1	0	0.25	0.25	0.6	0.42
ESA	0	0	0	0	0	0
EPS	0	0	0.5	0.25	0.6	0.27

Fig.2 shows the needed search times of different algorithms to obtain the two paths. For the EPS algorithm, the search times is more than others because double of the transmission paths should be chosen as candidate paths in the obtaining process. Meanwhile the needed search times is decreased in the MA* algorithm by evaluating the heuristic value from the current node to the destination node. In addition, the number of candidate paths is less than double of the transmission paths. Therefore, the needed search times of the MA* algorithm are much smaller than the EPS algorithm, which is reduced by 38.1% on average. Fig.3 also shows that the needed search times of the MA* algorithm has improved comparing to other algorithms such as ESA. Synthetically the following conclusion can be deduced that the MA* algorithm can find multiple paths with reasonable path cost and path similarity at quite low search times.

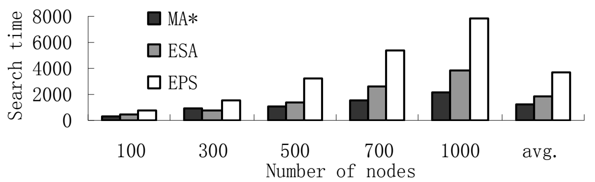

Fig.2. The search times of different algorithms

Furthermore we observe that the path cost and the path similarity change with the punishment parameter ϕ in the MA* algorithm. Thereby the simulation experiments are carried out for the MA* algorithm with different values of ϕ which are equal to $0.05\phi'$, $0.3\phi'$, $0.7\phi'$ and $2\phi'$ respectively. The experiment results are shown in Fig.3 which indicate that the path cost of the obtained paths by the MA* algorithm increases with the value of the parameter ϕ. And when ϕ is equal to $2\phi'$, the path cost of the obtained path is similar to the path found by the ESA algorithm. That is to say, if the punishment to the links is too large in the search process, the MA* algorithm will obtain the similar path as the ESA algorithm. So the following conclusion can be deduced that the MA* algorithm is a more generalized muti-path routing algorithm. The balance of path cost and path similarity can be achieved through reasonable adjustment of the punishment parameter.

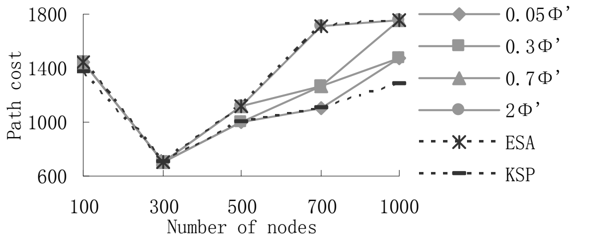

Fig.3. The path cost for different ϕ

Conclusions

In this paper, a new multi-path routing algorithm based on the A* algorithm called MA* algorithm is proposed which combines the path similarity objective with the heuristic method. A new evaluation function construction method is devised for the MA* algorithm, which is proved feasible through theoretical analysis. The simulation experiments are carried out to compare the MA* algorithm with other multi-path routing algorithms applying network topology generation tool. The results show that the MA* algorithm can acquire multiple paths with reasonable path cost and path similarity at quite low search times. In addition, the balance of path cost and path similarity can be achieved through reasonable adjustment of the punishment parameter. The innovation of MA* algorithm is that it can be applied in more wide multi-path transmission environments as a more generalized multi-path routing algorithm.

Acknowledgments: *Project supported by the Fundamental Research Funds for the Central Universities (No. 2012JBM025).*

REFERENCES

[1] Wischik D., Multipath: a new control architecture for the internet: technical perspective, *Commun. ACM*, 54 (1): 108-108, 2011

[2] Ramesh M.V., Reddy T.R., Latha T.S., An effective implementation of multi-path routing protocol in wireless mesh networks, *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, 2(3): 364-368, 2012

[3] Lian J., Zhang Y., Li C.J., An efficient k-shortest paths based routing algorithm, *Adv. Mater. Res.*, 532-533, 1775-1779, 2012

[4] Yang J.L., Liu M., Zhang X.Y., Node-disjoint multipath rougting based on correlation factor in ad hoc networks, *Mini-Micro System*, 27, 1669-1672, Sep. 2006

[5] Zhang J.X., Liang M.G., Wang Z.W., Efficient algorithm for finding multi-paths in telecommunication networks, *Key Engineering Materials*, Vols. 474-476, 2274-2278, 2011.

[6] Bai W.F., Han L., An improved A* algorithm in path planning, *2010 International Conference on CMCE*, 24-26, 2010

[7] Keyur Rana and Mukesh Zaveri, A-star algorithm for energy efficient routing in wireless sensor network, *Communications in Computer and Information Science*, 197(2): 232-241, 2011

[8] Liu H., Bao Y.L., A* alogrithm for the shortest path on vector maps, *Computer simulation*, 25(4), 253-257, 2008

[9] Medina A., Lakhina A., Matta I., Byers J., BRITE：an approach to universal topology generation, *Proc of the MASCOTS 2001*, Washington, 346-353, 2001

Authors*: prof. dr. Aqun ZHAO, School of Computer and Information Technology, Beijing Jiaotong University, No.3 Shang Yuan Cun, 100044 Beijing, China, E-mail: aqzhao@bjtu.edu.cn.*