path 1-2, the cost is 1. Then the node 2 is relaxed, we get the path 1-2-3, the cost is 6. The process continues until the destination node 6 is relaxed. At that time, the path is 1-2-3-4-6, the cost is 11. Return to the source node, we get the path is 1-5, the cost is 2. Finally, the node 5 is relaxed, we get the path 1-5-6, the cost is 5. The shortest path of this graph is 1-5-6, the cost is 5. This process takes too much time to find the shortest path. However, it will get the shortest path is easy, if we use SPFA algorithm. First, the node 1 is relaxed, we get the path 1-2 and 1-5. Second, the node 2 is relaxed, we get the path 1-2-3, the cost is 6.Then the node 5 is relaxed, we can get the shortest path is 1-5-6, the cost is 5. Because the cost 6 is greater than the cost 5. In other words,the cost of path 1-5-6 lower than path 1-2-3. So the cost of path 1-5-6 certainly be lower than path 1-2-3-4-6. SPFA algorithm is not to be affected by the depth graphs.

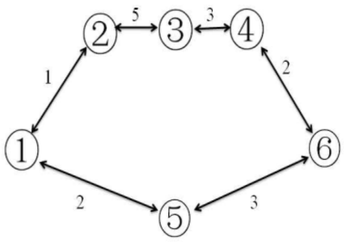

Fig. 4: Example of a random map

Grid maps take advantage of special characteristics of the improved algorithm. The iterative times of two algorithms are the same, when using grid maps. The proposed algorithm will take time to solve the depth problem. Likewise, SPFA algorithm will take time to solve the breadth problem. The depth and the breadth of grid maps are equivalent. On grid maps, the proposed algorithm is more efficient than the original SPFA algorithm. As shown in Table.2.

Table.2: Grid maps data

Data	SPFA	Proposed algorithm
NON-Negative weights V=490000	8973ms	2826ms
NON-Negative weights V=1000000	22075ms	6072ms

As is shown in the table, we can see clearly the relationship between the proposed and SPFA algorithms. In both cases, SPFA algorithm takes 31048 milliseconds, and the proposed algorithm only takes 8898 milliseconds that reduced two-third in time consumption.SPFA algorithm is better than the proposed algorithm in the case of random data. However, the proposed algorithm is more effective on grid maps.

4. 4 . The Application of Grid Map

Grid maps may be arbitrary, or can be based on specific distances. A grid locates a unique square region on the map. The communication may use a simple grid system. For example, this might be used for land navigating for walkers or cyclists etc. The growing availability and decreasing cost of handheld GPS receivers enable determination of accurate grid maps without needing a map. First of all, the city map is divided into a grid of several squares. Each building treat as the node. Thus, we can use the proposed algorithm to compute the shortest distance between the two buildings.

5. Conclusions

The first experimental result shows that SPFA algorithm and Dijkstra's algorithm have almost same efficiency on the random graphs. But, on grid maps the proposed algorithm is very efficient. As shown in Table.2, when CPU is Intel P7350 2.00GHz, windows 7(32bit), SPFA algorithm takes 31048 milliseconds by Matlab, the proposed algorithm only takes 8898 milliseconds that reduced two-third in time consumption. So the proposed algorithm is more effective on grid maps. I can verify that on grid maps the proposed algorithm is very efficient, with the same amount of data within the same grid map that we would save more time. The proposed algorithm than SPFA algorithm reduced two-third in time consumption. This may be broaden great development space for the proposed algorithm in the future.

References

[1] Duan, Fanding, " A Faster Algorithm For Shortest Path-SPFA",*Journal of Southwest Jiaotong university* ,Vol.29,No.6, pp. 207-212, 1994

[2] Tianrui Li, Luole Qi, Da Ruan, "An efficient algorithm for the single-source shortest path problem in graph theory", *International Conference on Intelligent System and Knowledge Engineering,* Vol.1, pp. 152-157, November 2008.

[3] Dijkstra, E.W. "A note on two problems in connexion with graphs". *Numerische Mathematik*, pp,269–271,1959

[4] Kroger, Martin . "Shortest multiple disconnected path for the analysis of entanglements in two- and three-dimensional polymeric systems". *Computer Physics Communications* ,168 (168): pp.209－232,2005.

[5] Boris Cherkassky, Andrew V. Goldberg, Tomasz Radzik, "Shortest paths algorithms: Theory and experimental evaluation", *Mathematical Programming*, 129-174, May 1995.