# Graph Partitioning-based Query Acceleration of Power Graph Database

Chunhui Ren, Jianchao Lin, Xuan Wang

*Artificial Intelligence on Electric Power System State Grid Corporation Joint Laboratory (State Grid Smart Grid Research Institute Co., Ltd.)*

Chunhui Ren, Jianchao Lin, Xuan Wang

*Electric Power Research Institute of State Grid Jiangsu Electric Power Co., Ltd.*

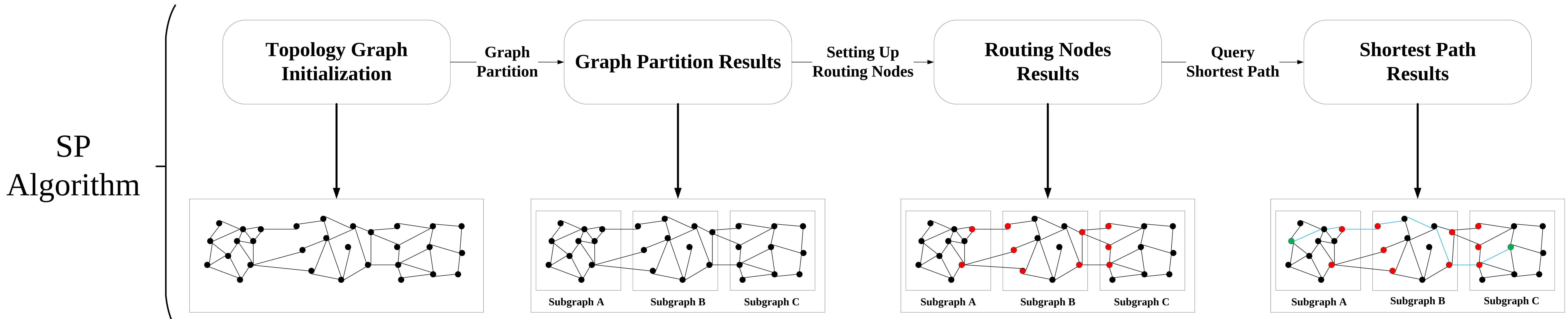


Fig.1. SP algorithm flow chart

## Introduction

With the continuous development of power systems, power facilities are increasing and the corresponding equipment data are increasing dramatically, in order to process equipment data efficiently, graph databases are used in power systems on a large scale and show good performance. For better topological analysis, GIS system is also applied in power systems, which combines database technology, computer technology and the scale, structure, and attributes of power grids as well as their real geographic location information to strengthen the topological analysis of power systems and favorably promote the development of smart grids .However, when facing municipal or provincial scale grid topologies, the shortest path query is under great pressure as the basis for the optimal power supply scheme and the best repair path of the distribution network in the topology analysis of the grid GIS graph database .Therefore, the performance of the grid GIS graph database for querying the shortest path has put forward higher requirements to enhance the emergency event handling capability.

## Methods

In this section, the SP algorithm based on graph partition and routing nodes is an important stage to realize the acceleration of data query for grid GIS graph database, which firstly divides the power grid topology graph into several subgraphs of disjoint and comparable size, secondly specifies the eligible routing nodes in the subgraphs, and the routing nodes can store the shortest path to other nodes in the subgraph they belong to, and finally the subgraphs communicate with each other through the routing nodes. The flow chart of SP algorithm is shown in Fig. 1.

## Experimental configuration

In this paper, simulation experiments were conducted using real data provided by the power system graph database of a provincial national grid, which has 1 million nodes and 6 million edges. The operating system and hardware configuration used for the simulation calculation in this paper are shown in Table I.

**Table** Ⅰ TEST ENVIRONMENT

| | |
|---|---|
| CPU↵ | Intel(R) Core (TM) i7-11700 CPU @ 2.50GHz↵ |
| Memory↵ | 16GB↵ |
| Operating System↵ | CentOS 7↵ |

## Result and discussion

**Table** Ⅱ TIME COMPARISON OF ITERATIONS TO 120 LAYERS,TIME(s)

| layers | SP Algorithm/ 10 nodes | SP Algorithm/ 20 nodes | PBFS Algorithm |
|---|---|---|---|
| 1 | 0.007 | 0.008 | 0.0065 |
| 10 | 0.00765 | 0.00795 | 0.009 |
| 20 | 0.0117 | 0.0125 | 0.013 |
| 30 | 0.0133 | 0.01422 | 0.015 |
| 40 | 0.0224 | 0.03 | 0.032 |
| 50 | 0.0465 | 0.064 | 8.4 |
| 60 | 0.067 | 0.0974 | 25.91 |
| 70 | 0.092 | 0.153 | 41.21 |
| 80 | 0.108 | 0.261 | 40 |
| 90 | 0.128 | 0.312 | 25.52 |
| 100 | 0.153 | 0.35 | 13.41 |
| 110 | 0.184 | 0.49 | 2.774 |
| 120 | 0.202 | 0.637 | 0.877 |

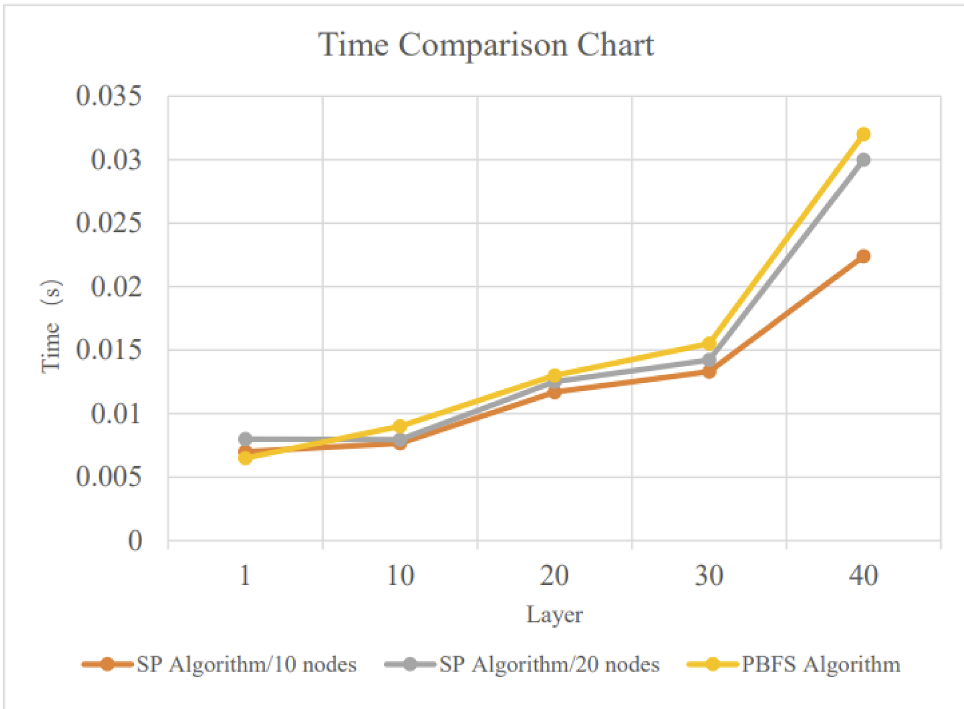


Fig.2. Time comparison chart from layer 1 recursion to layer 40

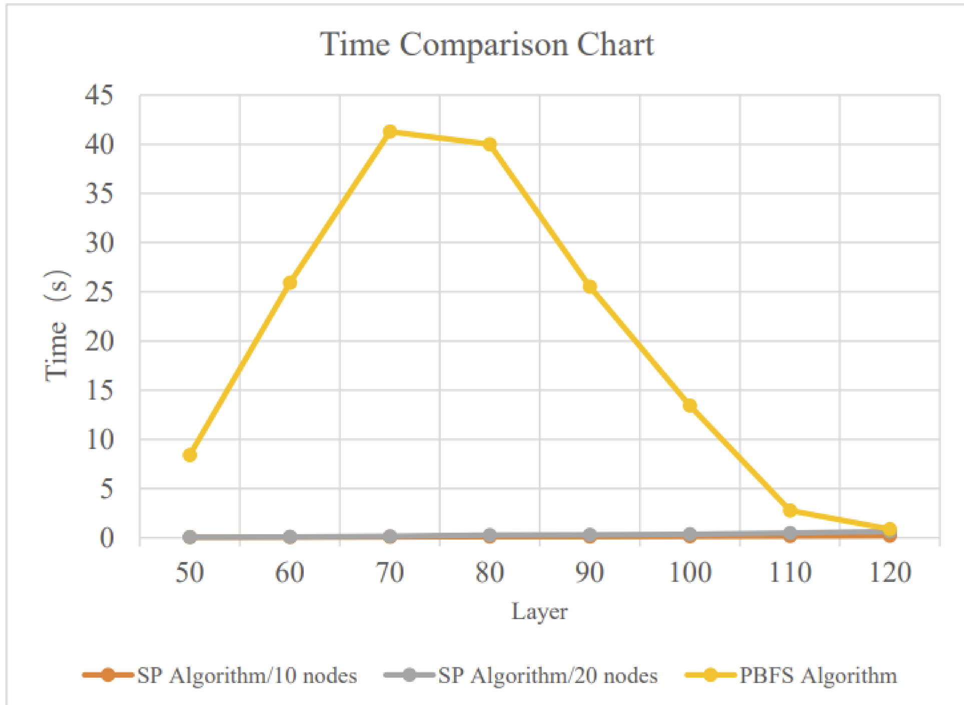


Fig.3. Time comparison chart from layer 50 recursion to layer 120

From Table II, Fig. 2 and Fig. 3, due to the application of graph partition and routing nodes, the query time stays within 1s in the case of subgraphs with the number of nodes of 10 and 20, respectively, where the query time for the subgraph with the number of nodes of 20 has been slightly larger than that for the subgraph with the number of nodes of 10. This is due to the fact that the number of subgraph nodes rises from 10 to 20 and the shortest path data stored in the routing nodes increases, querying the stored data in the routing nodes consumes excess time. In contrast to the PBFS algorithm, which consumes little difference from the SP algorithm in the recursive cases from level 1 to 40 and at level 120, the algorithm spends more time querying the shortest path in the recursive cases from level 40 to 110, where it reaches a maximum of 41.291s and a minimum of 2.774s. The above analysis shows that the SP algorithm, which divides the power grid topology graph into several subgraphs of comparable size and queries the shortest path by routing nodes, can effectively achieve the query acceleration effect and shows excellent performance.

## Conclusion

In the context of rapid expansion of power system scale, the query speed of grid GIS graph database in the shortest path needs to be improved so as to respond to emergencies in a timely manner. In order to achieve query acceleration in shortest path, this paper proposes a shortest path query algorithm based on graph partition and routing nodes, which achieves query acceleration by dividing the power grid topology graph into several subgraphs of comparable size and specifying routing nodes in the subgraphs, where the routing nodes store the shortest path to reach each node within the subgraph to which they belong, and filtering out nodes on non-shortest paths by querying the routing nodes . The proposed algorithm is experimentally verified to be effective in accelerating the query of shortest paths in the grid GIS graph database.