A Parallel Genetic Algorithm Based on MPI for N-Queen

Kaiyu Wang, Zhenzhou Ji and Yihao Zhou

Department of Computer Science and Technology Harbin Institute Technology Weihai, 264209, China

Abstract—Genetic Algorithm which has certain ability to learn is an ideal tool to solve the complicated problems. Genetic Algorithm follow the Darwinian evolution, the model N-Queens is classic NP problems the problem scale is exponential growth with the growth of queens. So it costs a mass of time to solve it in numerous data. Therefore the genetic algorithm for parallel processing to solve large-scale NP has great significance. In the thesis we discuss the genetic algorithm and parallel technique, we have improved the technique specialized in the genetic algorithm and proved that the genetic algorithm can be speed up through lots of experiences and make optimization for the possible of early convergence and present a solution.

Keywords—genetic algorithm; MPI; parallel technique; numerous data

I. INTRODUCTION

With the continuous development of computer technology, people ushered the era of big data. Machine learning and artificial intelligence, and other computer technology also rapid development, so we put forward higher requirements for computing speed. Therefore parallel technology takes more attention. Genetic Algorithm is a computing model that simulate the evolutionary process of natural selection and genetic mechanism, this model is widely used in combination optimization, machine learning, signal processing, self-adaptation control and artificial life etc.[1].MPI is a standard based on message passing parallel programming, MPI is flexible and effective, has strong portability.

II. THEORY SUMMARY

A. Genetic Algorithm

Genetic Algorithm is a computing model that simulate the evolutionary process of natural selection and genetic mechanism, the model is a way through simulating the evolutionary process of nature selection find the optimal solution. The whole algorithm works as imitate gene coding, heredity. We use binary coding way to show the gene that can simplify the coding manner, and convenient processing by computers.[2].The Figure I shows the technological process of genetic algorithm.

B. N-Queen Problem

N-Queen problem is a classics NP problem, if we use backtracking method to solve the problem, the time complexity of it is $O(n!)$, when N is enormous enough then the backtracking method can’t be used. N-Queen problem is a feasibility decision problem of the position, it can be used as theoretical model of many realistic problem. For stance in urban planning we choose the position for such as traffic center, hospital, shopping mall etc., these building has a huge population flow. We can use N-Queen problem as an abstract model to study. We put the city into several area, to make sure the city function well, each area must have some prerequisite public facilities, for stance police department, fire brigade, post office and public transport, but every area has it particularity to choose the facilities. The commercial district of downtown and residential quarters have more requirements in public security and health, so those area need have hospital and police department etc. Again for stance railway station and airport these large public transport facilities need to refer to retrieve multiple conditions to choose position. Every facility location
is decided by multiple factors. The related data in the form of index increase with the increase of urban size. So we need the appropriate parallel method to calculate numerous data.

### III. ALGORITHM IMPLEMENTATION

#### A. Chromosome Coding

The ordinary coding method of N-Queen problem is double dimensional array, this way is easy to understand but occupy plenty of storage space is very inefficient. This thesis use natural coding way, establish a one-dimensional of length \( N \) to store N-queens, as \( A[i] \). In this way ion behalf of the queen’s row position, the value of \( A[i] \) is queen’s line position. In the coding way the searching space is \( N^N \) far way less than the searching space in double dimensional coding which can speed the algorithm convergence [3]. Figure II is the sketch map of chromosome coding.

![Figure II. Chromosome Coding Sketch Map](image)

#### B. Serial Genetic Algorithm Implementation

Serial genetic algorithm (SGA) is the foundation of study parallel genetic algorithm. First is initialize the queue parameters calculate the optimum fitness, then use the fitness estimate the quality of solution set. In the last do select, cross and mutate. Calculate the fitness is the important part of genetic algorithm, different fitness influence the every evolutionary iteration and the convergence speed. If the queen’s position is attack with other queens, we call it a conflict. The conflict is an important evaluation parameter to the queen’s position, Eq.1 is the design formulas of chromosome’s conflicts.

\[
X_{ij} = \begin{cases} 
0 & \text{if } |i-j| = |P_i - P_j| \\
1 & \text{if } |i-j| = |P_i - P_j| 
\end{cases}
\]

\[
f_x = \sum_{i=1}^{n} \sum_{j=1}^{n} X_{ij}
\]

The evolutionary operators of genetic algorithm is the main factor that suitable for processing large scarse problems. The evolutionary operators is constituted by select, crossover, mutate. The whole SGA is based on the population evolution, until the answer is match condition. After the experiment observation, the number of evolution is proportional to the problem size, and strictly monotone rose to the algorithm time. So the number of evolution and algorithm time can be used as objective criterion of SGA and parallel genetic algorithm. The Figure III is the tendency chart of evolution number and Figure IV is the tendency chart of algorithm time.

![Figure III. Tendency Chart of Evolution Number](image)

![Figure IV. Tendency Chart of Algorithm Time](image)

#### C. Parallel Genetic Algorithm

Parallel genetic algorithm (PGA) is on the basis of serial genetic algorithm, use MPI to transmit information. The thesis mainly do implementation and testing of coarse-grained.

The initial population of genetic algorithm is random generate, so each population has a different fitness, if a population has a higher fitness then it need a fewer number of evolution, can get a perfect answer sooner than other population. The coarse-grained model make each population evolution mutual independence, so we can take the minimum-cost answer as the locally optimal solution [4].

The coarse-cost model has widely use in practical application, mainly because it is easy to implement and structure simply. The Figure V is the flow chart of coarse-grained parallel genetic algorithm. Use MPI to rank and number every process then process run the algorithm independently [5]. We set a judge condition to check the answer precision, create a public communication domain, let processes put their answers into the domain, so we can summarize results and choose the perfect answer.
The Table I is the comparison of Coarse-grained PGA and SGA in 4 processor. The experiment is under the Windows 7 and Inter(R)Core(TM) i5-4460 3.20GHz, use MPI to realize.

<table>
<thead>
<tr>
<th>QUEENS</th>
<th>CPGA/S</th>
<th>SGA/S</th>
<th>ACCELERATE RATIO</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.001000</td>
<td>0.001000</td>
<td>1.000</td>
</tr>
<tr>
<td>16</td>
<td>0.001000</td>
<td>0.036000</td>
<td>3.600</td>
</tr>
<tr>
<td>32</td>
<td>0.017000</td>
<td>0.031000</td>
<td>1.832</td>
</tr>
<tr>
<td>64</td>
<td>0.111000</td>
<td>0.130000</td>
<td>1.182</td>
</tr>
<tr>
<td>128</td>
<td>0.569000</td>
<td>1.662000</td>
<td>2.921</td>
</tr>
<tr>
<td>256</td>
<td>4.054000</td>
<td>18.853000</td>
<td>4.650</td>
</tr>
<tr>
<td>512</td>
<td>39.494000</td>
<td>87.691000</td>
<td>2.220</td>
</tr>
</tbody>
</table>

In CPGA model with 4 processor we can get 2.485 average accelerate ratio, the accelerate ratio is rising tendency with the Queen number increase, so the better initial population is the advantage is more clearly. The accelerate rate is rising with processor number increase. Figure VI is the tendency chart of accelerate ratio. The accelerate ratio is the data of comprehensive test from 8 queen to 512 queen, on the basis of Amdahl law, if scale of data is invariant, processor increase so the parallel part of genetic algorithm is less in proportion, the efficiency will decline. In coarse-cost model algorithm structure was not damaged, if population number is numerous it can be equal division into every processor the accelerate ratio will increase with processor number rising.

FIGURE V. FLOW CHART OF COARSE-GRAINED PGA

FIGURE VI. TENDENCY CHART OF ACCELERATE RATIO

IV. OPTIMIZING OF CPGA

Genetic algorithm is simpleness, good commonality, robust and have implicit parallelism, suitable for processing large scale NP problems. But genetic algorithm has boundedness, because of evolution way monotonous, so it is easy to early convergence in large scale problem.

In the test of coarse-cost model when algorithm begin to work, number of conflict is drop in a very quick speed.8 queen, for example the initial number of conflict is between 6 and 7, after 20 times evolution, the conflicts will drop at about 3. In almost all test the model can improve over 60% conflicts in initial 15% evolution. In last 85% evolution algorithm can improve very little conflicts, the efficiency of conflicts improved is decrease as the evolution near complete. The early convergence is because of algorithm structure. Make an example of 8 queen: 6 4 7 1 3 5 2 8 is an answer of 8 queen, but the answer of N queen is not uniqueness, it is a solution set.6 4 7 1 3 5 2 8 is represent a relative order, so we can easy to know if a chromosome order is conform to that relative order it is an answer to N queen. As a result of this characteristic when conflicts is drop in a low state, the effect evolution factor have on individual is small and only mutate factor can work [6].

Because of the deficiency of SGA we can make improve in PGA. In coarse-cost model every parallel part don’t have active communicate, so each part can use a different evolution strategy. If a node make a threshold value judgment affirm this part is early convergence then each node make a communication. If this node have the best population then broadcast the population to other node and the other node give up the current population take the broadcast then continue evolution in their way. If the early convergence population is not the best population then give up the current population and wait for broadcast.

In this way optimize PGA use other different evolution way to avoid early convergence and algorithm fitness will not decrease make the performance reduction. Figure VII is Optimizing of the CPGA.

TABLE I COARSE-GRAINED PGA AND SGA

Advances in Intelligent Systems Research, volume 134
V. CONCLUSION

The thesis based on genetic algorithm to solve large-scale NP problem and make parallel optimization. The thesis use the N-Queen problem as the test set make study on the performance of SGA and PGA. Make an improved model of genetic algorithm for early convergence, the test show PGA in 4 processor can have a 2.485 accelerate ratio and it will increase with processor number increase. The PGA is can improve algorithm performance and get the better answer in schedule time.

ACKNOWLEGDMET

This paper is supported by The National Natural Science Foundation Project (61472100).

REFERENCE.


