

**PARALLEL HYBRID GENETIC ALGORITHM FOR
SOLVING TRAVELING SALESMAN PROBLEM**

www.Technicalpapers.co.nr

Abstract:

This paper deals with the basics of genetic algorithm, a universal method of optimization and the need for parallel genetic algorithm. This also discusses briefly on the traveling salesman problem. A methodology is proposed to solve this problem using the different parallelism models.

Keywords: *Genetic algorithm, parallelism, traveling salesman problem.*

1. GENETIC ALGORITHM:

Genetic algorithm (GA) is an evolutionary optimization approach. Genetic algorithms are universal methods of optimization as they are easily applicable to many practical problems. GA follows the concept of solution evolution by stochastically developing generations of solution populations, using a given fitness. They are particularly applicable to problems, which are large, non-linear and possibly discrete in nature. The GA procedure is based on the Darwinian principle of survival of the fittest.

1.1 Four stages:

The GA consist four main stages: evaluation, selection, crossover and mutation.

- Generate a population
- While a solution has not been found
 - o Evaluate all members of the population.
 - o Create a new population using the Genetic Operators.
- Report the solution.

1.1.1 Create a Random Initial State:

An initial population is created from a random selection of solutions (which are analogous to chromosomes).

1.1.2 Evaluate Fitness:

A value for fitness is assigned to each solution (chromosome) depending on how close it actually is to solving the problem (thus arriving to the answer of the desired problem). (These "solutions" are not to be confused with "answers" to the problem, think of them as possible characteristics that the system would employ in order to reach the answer.)

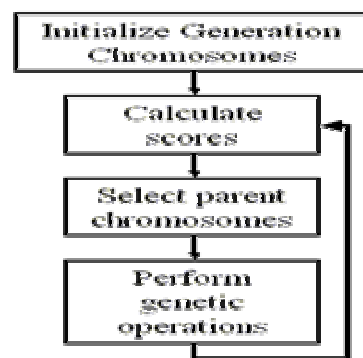
1.1.3 Reproduce (& Children Mutate):

Those chromosomes with a higher fitness value are more likely to reproduce offspring (which can mutate after reproduction). The offspring is a product of the father and mother, whose composition

consists of a combination of genes from them (this process is known as "crossing over").

1.1.4 Next Generation:

If the new generation contains a solution that produces an output that is close enough or equal to the desired answer then the problem has been solved. If this is not the case, then the new generation will go through the same process as their parents did. This will continue until a solution is reached.



2. NEED FOR PARALLEL GENETIC ALGORITHM:

- GAs are slow to solve even on the fastest machines today. This is mainly due to a vast number of possible solutions combined with objective evaluation. GAs attempt to find an optimum solution without having to explore all possible solutions.
- One of the main advantages in using GAs is the fact that they are inherently parallel algorithms. Little effort is needed to modify an existing GA to make use of a parallel computer.

3. TRAVELING SALESMAN PROBLEM:

3.1 Problem Statement:

In this hypothetical problem a salesman must visit a number of cities. The distances between the cities are known. The problem is to determine the order in which the salesman visit each city just once in turn, as to minimize the total distance traveled.

3.2 Complexity:

The most direct solution would be to try all the permutations and see which one is cheapest. And hence the number of permutations is $n!$, where n is the number of cities to be visited. This solution rapidly becomes impractical. Let us consider 10 cities. The total number of possible routes is 10 factorial or 3,628,800. In this case it is very difficult to find the shortest path to visit all the cities.

4. PROBLEM SOLVING: HYBRID GA

4.1 Initial Population :

The random routes are selected in the initial population.

4.2 Fitness function:

The fitness function is the sum of the cost incurred in traveling from city node 1 to N where N is the total number of cities to be visited.

4.3 Crossover :

Crossover algorithm is used to combine the two parent tours to make the child tours. A crossover exchanges parts of parent genes. Children “hopefully succeed” “good characteristics” of parents.

Two new concepts namely, RemoveSharp and LocalOpt are introduced.

- In RemoveSharp, the city causing sharp rise in distance in route is removed from its current position and inserted in a position where the distance rise caused by that city will be minimum.
- LocalOpt deals with optimizing localized parts of routes.

These two methods are done to the parents that are selected.

Those parents undergo these refining methods and then passed onto the crossover step. The crossover technique followed here is Partial Ordered Crossover also called as, Partially Matched Crossover (PMX) and Deterministic crossover.

50% OF POPULATION : PMX (PARTIALLY MATCHED CROSSOVER)

PMX may be viewed as crossover of permutations, that guarantees that all positions are found once in the route.

PMX proceeds as follows:

1. The two chromosomes are aligned.
2. Two crossing points are selected uniformly at random along the strings..
3. The center section is used to effect a cross through position-by-position exchange operation
4. Alleles are moved to their new positions in the offspring

```
Parent 1  3 7 1 9 | 6 4 5 | 2 8
Parent 2  4 7 8 5 | 3 9 2 | 1 6
( 3 <--> 4 ) ( 9 <--> 5 ) ( 2 <--> 6 )
Child 1   4 7 1 5 | 3 9 2 | 6 8
Child 2   3 7 8 9 | 6 4 5 | 1 2
```

ANOTHER 50% OF POPULATION: DETERMINISTIC

Child 1:

First city of child1 is the first city of the first parent. Now, the distance between first city is checked with the second city of both the parents. Whichever distance is smaller, that 2nd city from parent is selected as the 2nd city of child1. then, distance between the second city of child1 and the third city of both the parents are checked and the process goes on.

Problem and its solution:

- If the selected city already present in the route, select another city.
- If both the cities are found in the route, select a random city which is not placed in the route before.

Child 2:

First city of child2 is the first city of the second parent. The same process as stated above is followed here.

4.4 Mutation:

Eventually, this GA would make every solution look identical. This is not ideal. Once every tour in the population is identical, the GA will not be able to find a better solution.

- Mutation is a method where some cities in the child tours are randomly altered (swapped) to produce a new unique tour.

And thus, the child tours are sent to the fitness evaluation step, where these are considered as parents and the process continues.

5. PARALLELISM:

In a serial GA, the evaluation part / score calculation is the most time consuming. The parallel GA configuration is of the form of Master and slave. The simplest way to parallelize a GA is to use the synchronous master-slave model.

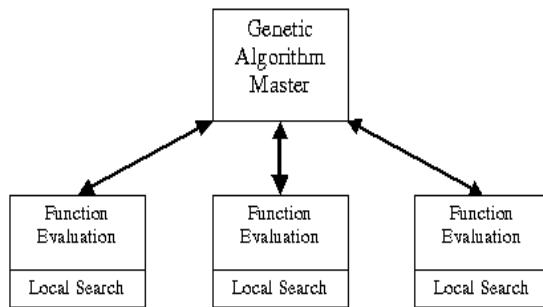
Models:

There are two main models of GA parallelism are the island model and the global population model.

- In the island model individuals are divided into several subpopulations assigned to concurrent processes which perform local genetic operations, and solutions are exchanged by migration.
- In the global population model a temporary global population is created once for every several generations by gathering all individuals in one process and redistributing them randomly.

5.1.1 Island model approach:

In this model we parallelize the GA by dividing the population into a number of smaller populations, placing each population on a separate processor. The standard GA algorithm is performed on each sub-population on each processor, but periodically we allow a number of good solutions from each sub-population to migrate to other populations(migration technique). The simple solution is to arrange the processors in a mesh format and migrate a small number of candidates to all neighbours. It has been found that this form of parallelization produces good speed up characteristics and can actually result in the production of better results for a variety of technical reasons.



Master-Slave parallel genetic algorithm model.

6. MPI:

MPI stands for Message Passing Interface.

The Message Passing Interface, MPI, is a standard API (Application Programming Interface) that can be used to create parallel applications. MPI is designed primarily to support the SPMD, single program multiple data model. Message passing is a method by which data from one processor's memory is copied to the memory of another processor. It is a de facto standard for communication among the nodes running a parallel program on a distributed memory system.

Most MPI implementations consist of a specific set of routines. Fortran, C or C++ languages are capable of interfacing with such routine libraries.

Advantages:

- Portability (because MPI has been implemented for almost every distributed memory architecture) There is no need to modify your source code when you port your application to a different platform that supports (and is compliant with) the MPI standard.
- Speed : Because each implementation is optimized for the hardware on which it runs.
- Standardization: MPI is the only message passing library which can be considered as a standard. It is supported on virtually all HPC platforms. Practically, it has replaced all previous message passing libraries.
- Scalability: Programs can usually achieve higher performance level .

6. FINDINGS AND CONCLUSIONS

There are so many methods to solve great challenge problems like traveling salesman problem and this is one of the techniques to solve and parallelizing it. This paper shows the unique and efficient way of genetic algorithm (hybrid GA). As the GAs are slow to solve even on the fastest machines today, there is a necessity to adopt a technique that enables to solve these problems faster. This parallel genetic algorithm is one of the techniques to solve GA at a faster rate. This method provides a better efficiency compared to sequential methods.

REFERENCE:

www.Technicalpapers.co.nr