**ABSTRACT**

In the iLoc tool, we propose a location-mapping/restacking service which provides a single window view. An efficient location mapping system is designed and implemented for locating the place of any employee from any location at any time. It allows the senior managers of the organization to identify, plan, review the existing resource and strategize the move globally. And it also allows the administration team to allocate or reallocate teams. This can be also strategized among the group/organization unit level to track the current location. Hence, the tool can be used to optimize and make precise usage of available space. During emergency relocation of teams for any particular group/teams, this will be a great help to solve the issue. In order to show the feasibility and effectiveness of the system, experiments were conducted on the large-scale iNautix campus to demonstrate that the iLoc – Location mapping/ restacking system provides better precision and reliability than manually done.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO** | **TITLE** | **PAGE NO** |
|  | **LIST OF FIGURES** | v |
|  | **LIST OF TABLES**  **ABSTRACT** | vi  i |
| **1** | **INTRODUCTION** | 1 |
|  | 1.1 INTRODUCTION | 2 |
|  | 1.2 SCHEDULING | 3 |
|  | 1.3 GENETIC ALGORITHM | 5 |
|  | 1.3.1 SELECTION OPERATOR | 6 |
|  | 1.3.2 CROSSOVER OPERATOR | 6 |
|  | 1.4 CROSSOVER OPERATOR | 7 |
|  | 1.4.1 SINGLE POINT CROSSOVER | 7 |
|  | 1.4.2 TOW POINT CROSSOVER | 8 |
|  | 1.4.3 THREE PARENT CROSSOVER | 8 |
|  | 1.4.4 UNIFORM CROSSOVER | 8 |
|  | 1.4.5 CROSSOVER FOR ORDERED CHROMOSOME | 10 |
|  | 1.5 MUTATION OPERATOR | 11 |
|  | 1.5.1 BIT STRING MUTATION | 12 |
|  | 1.5.2 FLIP BIT | 12 |
|  | 1.5.3 BOUNDARY | 12 |
|  | 1.5.4 NON-UNIFORM | 13 |
|  | 1.5.5 UNIFORM | 13 |
|  | 1.5.6 GUASSIAN | 13 |
|  | 1.5.7 SHRINK | 13 |
|  | 1.6 GAIN SCHEDULING | 14 |
|  | 1.7 LIST OF PROBLEMS FACED | 15 |
|  | 1.8 MOTIVATION TO DO THIS PROJECT | 16 |
| **2** | **LITERATURE REVIEW** | 17 |
| **3** | **PROBLEM STATEMENT** | 29 |
|  | 3.1 PROBLEM STATEMENT | 30 |
|  | 3.2 PROBLEM DESCRIPTION | 31 |
| **4** | **SYSTEM CONFIGURATION** | 34 |
|  | 4.1 HARDWARE ENVIRONMENT | 35 |
|  | 4.2 SOFTWARE CONFIGURATION | 35 |
|  | 4.2.1 LANGUAGE | 35 |
|  | 4.2.2 WORKING ENVIRONMENT | 36 |
|  | 4.2.3 VISUALIZATION TECHNIQUES | 36 |
|  | 4.2.4 MATHEMATICAL FUNCTION LIBRARY | 36 |
|  | 4.2.5 APPLICATION PROGRAM INTERFACE | 36 |
| **5** | **METHODOLOGY** | 37 |
|  | 5.1 MODULES | 38 |
|  | 5.2 MODULE DESCRIPTION | 38 |
|  | 5.2.1 CHECKING THE CONSTRAINTS | 38 |
|  | 5.2.2 CALCULATION OF MAKESPAN AND TOTAL COST | 39 |
|  | 5.2.3 CALCULATION OF FITNESS VALUE | 39 |
|  | 5.2.4 OPTIMIZATION OFSCHEDULING USING GA OPERATOR | 40 |
|  | 5.3 PROPOSED ALGORITHM | 41 |
|  | 5.3.1 ALGORITHM1-CALC\_FITNESS | 41 |
|  | 5.3.2 ALGORITHM2-MFGA | 42 |
| 6 | **TESTING** | 43 |
|  | 6.1 TIME COMPLEXITY | 44 |
| **7** | **RESULTS AND DISCUSSIONS** | 45 |
|  | 7.1 RESULTS INTERPRETED | 46 |
|  | 7.2 CONCLUSION AND FUTURE WORK | 51 |
| **8** | **REFERENCES** | 53 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE NO** | **DESCRIPTION** | **PAGE NO** |
| 1.1 | ARCHITECTURAL REPRESENTATION OF CLOUD COMPUTING | 3 |
| 1.2 | DIFFERENT TYPES OF ALGORITHMS IN SCHEDULING | 5 |
| 1.3 | ILLUSTRATION OF CROSSOVER OPERATOR | 7 |
| 1.4 | ILLUSTRATION OF SINGLE POINT CROSSOVER | 7 |
| 1.5 | ILLUSTRATION OF TWO POINT CROSSOVER | 8 |
| 1.6 | ILLUSTRATION OF UNIFORM CROSSOVER | 9 |
| 1.7 | ILLUSTRATION OF MUTATION OPERATOR | 11 |
| 1.8 | ILLUSTRATION OF BIT STRING MUTATION | 12 |
| 1.9 | WORKFLOW OF GENETIC ALGORITHM | 15 |
| 7.1 | COMPARISION OF THE ALGORITHMS | 47 |
| 7.2 | ILLUSTRATION OF FITNESS VALUE | 48 |
| 7.3 | RESULTS OF SIMULATION | 48 |
| 7.4 | ILLUSTRATION OF FITNESS WITH MAKESPAN | 49 |
| 7.5 | ILLUSTRATION OF FITNESS WITH COST | 50 |
| 7.6 | ILLUSRATION OF CASE(1) | 54 |
| 7.7 | ILLUSRATION OF CASE(2) | 55 |
| 7.8 | ILLUSRATION OF CASE(3) | 56 |
| 7.9 | ILLUSRATION OF CASE(4) | 57 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **TABLE NO** | **DESCRIPTION** | **PAGE NO** |
| 2.1 | COMPARISION OF THE LITERATURES RELATED TO GA | 25 |
| 6.1 | TIME COMPLEXITY OF THE MODULES | 44 |
| 7.1 | SAMPLE DATASET OF CONSUMERS | 51 |
| 7.2 | SAMPLE DATASET OF PROVIDERS | 53 |

**ABBREVIATION**

|  |  |
| --- | --- |
| VM | Virtual Machine |
| GA | Genetic Algorithm |
| NP-Hard | Non-deterministic Polynomial |
| SLA | Service Level Agreement |
| QoS | Quality of Service |
| ACO | Ant Colony Optimization |
| PSO | Particle Swarm Optimization |
| LCA | League Championship Algorithm |
| HUX | Half Uniform Crossover |
| PMX | Partially Matched Crossover |
| CX | Cycle Crossover |
| OX1 | Order Crossover Operator |
| OX2 | Order-based Crossover Operator |
| POS | Position-based Crossover Operator |
| VR | Voting Recombination Crossover Operator |