

Contents

Certificate	iii
Declaration by the Candidate	v
Copyright Transfer Certificate	vii
Preface	ix
Acknowledgements	xiii
Contents	xv
List of Figures	xix
List of Tables	xxiii
Abbreviations	xxviii
Symbols	xxxii
1 Introduction	1
1.1 On-demand Computing	1
1.2 Transaction Processing in On-demand Computing	2
1.3 Dependability	3
1.4 The Threats to Dependability: Failures, Errors, Faults	4
1.5 The means to Attain Dependability	5
1.6 Load Balanced Scheduling	6
1.7 Balanced Task Allocation	7
1.8 Load Balanced Transaction Processing Model	8

1.9	Motivation and Challenges	10
1.10	Dessertation Overview	12
1.11	Contribution and Impact	13
1.12	Organization	15
2	State of the Art	17
2.1	On-demand Computing	17
2.2	Load Balanced Transaction Scheduling	18
2.3	Load Balanced Task Allocation	23
3	Load Balanced Transaction Scheduling	27
3.1	Problem Formulation	28
3.2	The Proposed Ant Colony Optimization for Balanced Transaction Scheduling	31
3.2.1	The proposed algorithm: LBTS_ACO	31
3.2.2	Applying the LBTS_ACO Algorithm	37
3.2.3	Prevention of Premature Convergence of the Algorithm	38
3.2.4	Stagnation Avoidance	39
3.2.5	Convergence Test	39
3.3	Simulation and Results Analysis	40
3.3.1	Load on the System	44
3.3.2	Node Utilization	47
3.3.3	Transaction Throughput	49
3.3.4	Makespan Analysis	51
3.3.5	Miss Ratio	53
3.3.6	Load Balancing Speedup	55
3.3.7	Convergence Behaviour	57
3.4	Summary	57
4	Transaction Scheduling considering Availability	59
4.1	Problem Formulation	59
4.1.1	Makespan	62
4.1.2	Availability as Fitness Function	64
4.2	The Proposed Model	64
4.2.1	ACO approach	64
4.2.2	Proposed Algorithm: MATS_ACO	65
4.2.2.1	Prevention of Premature Convergence of the Algorithm	69
4.2.2.2	Stagnation Avoidance	69
4.2.2.3	Convergence Test	69
4.2.3	Applying the MATS_ACO Algorithm: Case Study (using NFSNet)	71
4.2.4	Time Complexity of MATS_ACO	73
4.3	Experimental Evaluations	74

4.3.1	Experiment with Varying Mean Time To Failure	76
4.3.2	Experiment with Varying Mean Time To Repair	78
4.3.3	Experiment with Varying Task Size	79
4.3.4	Experiment with Varying Number of Tasks	81
4.3.5	Experiment with Varying Number of Nodes	83
4.3.6	Experiment with Varying Processing Speed of Nodes	84
4.3.7	Experiment with Varying Load on Nodes	85
4.3.8	Makespan Analysis	87
4.3.9	Comparative Study of Proposed Algorithm	89
4.4	Summary	92
5	Load Balanced Scheduling considering Performability	95
5.1	Load Balanced Transaction Scheduling Model	95
5.2	Problem Formulation	97
5.3	Proposed Algorithm	102
5.4	Applying LBTS_HBO Algorithm	104
5.5	Simulation and Result Analysis	106
5.5.1	Comparison of Time Complexity of LBTS_HBO Algorithm with Other Algorithms	107
5.5.2	Availability Analysis of Resources in the System	108
5.5.3	Performability Analysis of the System	110
5.5.4	Miss Ratio of Transactions in the System	111
5.6	Observations	113
5.7	Summary	114
6	Load Balanced Transaction Allocation considering Reliability	115
6.1	Load Balanced Transaction Allocation Model	116
6.1.1	Assumptions	116
6.2	Problem Formulation	117
6.2.1	Reliability Model	117
6.3	Proposed Algorithm	122
6.3.1	Social Spider Optimization	122
6.3.2	Constraints Description	123
6.4	Applying the Algorithm	125
6.5	Simulation and Result Analysis	126
6.5.1	Result Evaluation	126
6.5.1.1	Resource availability	128
6.5.1.2	Reliability	133
6.5.1.3	Miss Ratio	139
6.6	Observations	145
6.7	Summary	145

7 Conclusion and Future Directions	147
7.1 Concluding Discussion	147
7.2 Future Directions	149
A List of Publications	165

List of Figures

1.1	Taxonomy showing relationship between Dependability and Attributes, Threats and Means (after Laprie et al.)	6
1.2	Transaction Processing Model with Load Balanced Scheduling	9
1.3	Overview of the dissertation's contributions	12
3.1	Working example of the LBTS_ACO when N=8	42
3.2	Load on the system with Transaction Processing	46
3.3	Load on the system without Transaction Processing	46
3.4	Load on the system with and without Transaction Processing	47
3.5	Load Deviation results	47
3.6	Node Utilization with Transaction Processing	48
3.7	Node Utilization without Transaction Processing	48
3.8	Node Utilization with and without Transaction Processing	49
3.9	Throughput when TM is used is used	50
3.10	Throughput when no TM is used	50
3.11	Comparison of the LBTS_ACO Throughput	51
3.12	Makespan when TM is used	53
3.13	Makespan when no TM is used	53
3.14	Comparison of the LBTS_ACO makespan	53
3.15	Miss Ratio with Transaction Processing	54
3.16	Miss Ratio without Transaction Processing	54
3.17	Comparison of the LBTS_ACO Miss Ratio	55
3.18	Load balancing speedup with Transaction Processing	56
3.19	Load balancing speedup without Transaction Processing	56
3.20	Comparison of the LBTS_ACO load balancing speedup	56
3.21	Load on the system for 40 simulations	57
3.22	Makespan results for 40 simulations	57
4.1	NFSNet	71
4.2	Availability observation with mean time to failure (MTTF) using GA	79
4.3	Availability observation with mean time to failure (MTTF) using EO	79
4.4	Availability observation with mean time to failure (MTTF) using MATS_ACO	79

4.5	Availability observation with mean time to repair (MTTR) using GA	80
4.6	Availability observation with mean time to repair (MTTR) using EO	80
4.7	Availability observation with mean time to repair (MTTR) using MATS_ACO	81
4.8	Availability observation with different task size using GA	81
4.9	Availability observation with different task size using EO	81
4.10	Availability observation with different task size using MATS_ACO	83
4.11	Availability observation with a different number of tasks using GA	83
4.12	Availability observation with a different number of tasks using EO	83
4.13	Availability observation with a different number of tasks using MATS_ACO	84
4.14	Availability observation with a different number of nodes using GA	85
4.15	Availability observation with a different number of nodes using EO	85
4.16	Availability observation with a different number of nodes using MATS_ACO	85
4.17	Availability observation with varying processing speed of nodes using GA	86
4.18	Availability observation with varying processing speed of nodes using EO	86
4.19	Availability observation with varying processing speed of nodes using MATS_ACO	86
4.20	Availability observation with different loads in nodes using GA	87
4.21	Availability observation with different loads in nodes using EO	87
4.22	Availability observation with different loads in nodes using MATS_ACO .	88
4.23	Makespan observation with different tasks with 50 tasks	89
4.24	Makespan observation with different tasks with 100 tasks	89
4.25	Makespan observation with different tasks: with 150 tasks	89
4.26	Makespan observation with different tasks: with 200 tasks	89
4.27	Availability Analysis	91
4.28	Makespan Analysis	91
4.29	Analysis of the Pareto Front of the bi-objective optimization problem . . .	92
5.1	Performability-Aware Load Balanced Transaction Scheduling Model	96
5.2	Working example of the LBTS_HBO when N=8	105
5.3	Resource Availability in LBTS_HBO with and without transaction management (TM)	109
5.4	Resource Availability when transaction management (TM) is used	110
5.5	Resource Availability when no transaction management (TM) is used . . .	110
5.6	Performability in LBTS_HBO with transaction management (TM) and without transaction management	112
5.7	Performability when transaction management is used	112
5.8	Performability when no transaction management is used	112
5.9	Miss Ratio in LBTS_HBO with and without transaction management (TM)	113
5.10	Miss Ratio when transaction management is used	114
5.11	Miss Ratio when no transaction management is used	114

6.1	Working example of the LBTA_SSO when N=8	127
6.2	Resource Availability in LBTA_SSO with and without TM in grid computing based environment	129
6.3	Resource Availability in LBTA_SSO with and without TM in cloud computing based environment	129
6.4	Resource Availability when TM is used in grid computing based environment	133
6.5	Resource Availability when TM is used in cloud computing based environment	133
6.6	Resource Availability when no TM is used in grid computing based environment	133
6.7	Resource Availability when no TM is used in cloud computing based environment	133
6.8	Resource availability comparison between grid and cloud computing system when transaction processing is used	134
6.9	Resource availability comparison between grid and cloud computing system when no transaction processing is used	134
6.10	Reliability in LBTA_SSO with TM and without TM in grid computing based environment	135
6.11	Reliability in LBTA_SSO with TM and without TM in cloud computing based environment	135
6.12	Reliability when TM is used in grid computing based environment	138
6.13	Reliability when TM is used in cloud computing based environment	138
6.14	Reliability when no TM is used in grid computing based environment	139
6.15	Reliability when no TM is used in cloud computing based environment	139
6.16	Reliability comparison between grid and cloud computing system when transaction processing is used	139
6.17	Reliability comparison between grid and cloud computing system when no transaction processing is used	139
6.18	Miss Ratio in LBTA_SSO with and without TM in grid computing based environment	141
6.19	Miss Ratio in LBTA_SSO with and without TM in cloud computing based environment	141
6.20	Miss Ratio when TM is used in grid computing based environment	141
6.21	Miss Ratio when TM is used in cloud computing based environment	141
6.22	Miss Ratio when no TM is used in grid computing based environment	144
6.23	Miss Ratio when no TM is used in cloud computing based environment	144
6.24	Miss Ratio comparison between grid and cloud computing when transaction processing is used	145
6.25	Miss Ratio comparison between grid and cloud computing when no transaction processing is used	145

List of Tables

2.1	Characteristics of the algorithms in the literature	22
3.1	Parameters of LBTS_ACO	32
3.2	Normality Shapiro-Wilk tests for the best results of load	44
3.3	Normality Shapiro-Wilk tests for the best results of makespan	44
3.4	Wilcoxon statistical tests for the best results (load) found for LBTS_ACO, EO, HLBA, GA, DLB, and Randomized algorithms. Assume null hypothesis $\mu_0 = 0$ and null hypothesis: two-sided, $\hat{\mu} < \mu$	44
3.5	Load on the system with 40 simulations	45
3.6	Load on the system	45
3.7	Node Utilization	47
3.8	Throughput (Number of the transactions per unit time) after executing for a particular time unit	49
3.9	Makespan with 40 simulations	52
3.10	Makespan	52
3.11	Miss Ratio (%)	54
3.12	load balancing speedup	55
4.1	The values of $MTTF_j$, $MTTR_j$, and A_j denote the values of mean time to failure, mean time to repair, and availability of the j^{th} node in NFSNet shown in FIGURE 4.1 in case study II where $N = 14$	72
4.2	Task scheduling.	73
4.3	Input parameters for ACO	74
4.4	Normality Shapiro-Wilk tests for the best results of availability	75
4.5	Wilcoxon statistical tests for the best results (availability) found for ACO, EO and GA algorithms. Assume null hypothesis $\mu_0 = 0$ and null hypothesis: two-sided, $\hat{\mu} < \mu$	75
4.6	Normality Shapiro-Wilk tests for the best results of makespan	75
4.7	Wilcoxon statistical tests for the best results (makespan) found for ACO, EO and GA algorithms. Assume null hypothesis $\mu_0 = 0$ and null hypothesis: two-sided, $\hat{\mu} < \mu$	76
4.8	Input parameters for ACO	76
4.9	Input parameters for GA [1]	77
4.10	Input parameters for EO [2]	77

4.11	The mean and median value of resource availability. FIGURE 4.2 shows the results for GA method while FIGURE 4.3 and FIGURE 4.4 shows the results for EO and MATS_ACO methods respectively.	78
4.12	The mean and median value of resource availability. FIGURE 4.5 shows the results for GA method while FIGURE 4.6 and FIGURE 4.7 shows the results for EO and MATS_ACO methods respectively.	80
4.13	The mean and median value of resource availability. FIGURE 4.8 shows the results for GA method while FIGURE 4.9 and FIGURE 4.10 shows the results for EO and MATS_ACO methods respectively.	82
4.14	The mean and median value of resource availability. FIGURE 4.11 shows the results for GA method while FIGURE 4.12 and FIGURE 4.13 shows the results for EO and MATS_ACO methods respectively.	82
4.15	The mean and median value of resource availability. FIGURE 4.14 shows the results for GA method while FIGURE 4.15 and FIGURE 4.16 shows the results for EO and MATS_ACO methods respectively.	84
4.16	The mean and median value of resource availability. FIGURE 4.17 shows the results for GA method while FIGURE 4.18 and FIGURE 4.19 shows the results for EO and MATS_ACO methods respectively.	87
4.17	The mean and median value of resource availability. FIGURE 4.20 shows the results for GA method while FIGURE 4.21 and FIGURE 4.22 shows the results for EO and MATS_ACO methods respectively.	88
4.18	The mean and median value of makespan calculated from results in FIGURE 4.23, 4.24, 4.25 and 4.26	90
4.19	Availability with 40 simulations	90
4.20	Availability comparison of our proposed algorithm with EO and GA with respect to time	90
4.21	Makespan comparison of our proposed algorithm with EO and GA with respect to number of tasks	91
4.22	Makespan with 40 simulations	91
5.1	Comparison of time complexity of our algorithm with other algorithms	107
5.2	Comparison of the characteristics of the existing algorithms with our proposed algorithm LBTS_HBO	108
5.3	Resource Availability	109
5.4	Performability	111
5.5	Miss Ratio (%)	113
6.1	Definitions	116
6.2	Normality Shapiro-Wilk tests and Wilcoxon statistical tests for best results found for LBTA_SSO, HBO, ACO, HLBA, DLB, Randomized algorithms in grid computing and cloud computing scenarios	128
6.3	Resource Availability in case of grid computing system for 40 simulations	130
6.4	Resource Availability in case of cloud computing system for 40 simulations	131

6.5	Resource availability in case of grid computing system	132
6.6	Resource availability in case of cloud computing system	132
6.7	Reliability when grid computing is used for 40 simulations	136
6.8	Reliability when cloud computing is used for 40 simulations	137
6.9	Reliability in grid computing	138
6.10	Reliability in cloud computing	138
6.11	Miss Ratio in grid computing for 40 simulations	142
6.12	Miss Ratio in cloud computing system for 40 simulations	143
6.13	Miss Ratio (%) in grid computing	144
6.14	Miss Ratio (%) in cloud computing	144

List of Algorithms

1	LBTS_ACO	35
2	MATS_ACO	68
3	LBTS_HBO	103
4	LBTA_SSO	123

Abbreviations

ACO	Ant Colony Optimization
DLB	Dynamic and Decentralized Load Balancing Algorithm
EO	Extremal Optimization
GA	Genetic Algorithm
HBO	Honey Bee Optimization
HLBA	Hierarchical Load Balanced Algorithm
LBTA_SSO	Load Balanced Transaction Allocation using SSO
LBTS_ACO	Load Balanced Transaction Scheduling using ACO
LBTS_HBO	Load Balanced Transaction Scheduling using HBO
MATS_ACO	Maximization of Availability for Transaction Scheduling using ACO
<i>MTTB_j</i>	Mean Time Between Failures at the j^{th} node
<i>MTTF_j</i>	Mean Time To Failure at the j^{th} node
<i>MTTR_j</i>	Mean Time To Repair at the j^{th} node
Randomized	Randomized Algorithm
SSO	Social Spider Optimization
TM	Transaction Management
WTM	Without Transaction Management

Symbols

A_j	Availability at the j^{th} node
$A_{c,\lambda}$	The conditional steady-state availability of servers c with load λ
A_λ	Availability of the resources under load λ
A_p	Availability of the grid system when some of nodes are arranged in parallel
A_s	Availability of entire grid system
C_{N_k}	The available processing capacity of the k^{th} node
c	number of servers available at time t
$cost_{ig}$	The inter communication cost between T_i and T_g measured in “words” (some unit of data quantity)
$cost(X)$	The cost of task allocation X
DM_i	Deadline-miss of i^{th} transaction
$D(T_i)$	Deadline of i^{th} transaction
E	The set of edges between nodes
e_{ik}	The expected execution time of transaction i running on node k
G	Complete undirected graph
G_i	The set of all nodes still to be visited by ant m
K	Maximum number of iterations
k	Counter for number of iterations
L_{ik}	The load required by all the transaction at k^{th} node $\forall i = 1, \dots, m$
L_j	Load on the j^{th} node

l_{kb}	The communication link from k^{th} to b^{th} node
M_k	The available memory size at k^{th} node
m	The number of ants
N	Set of nodes in the on-demand computing system
N_k	k^{th} node $\forall k = 1, \dots, n$
\mathcal{N}^i	List of nodes traveled by ant i
n	node
NIT_a	The number of instructions in the a^{th} task, $\forall a = 1, \dots, m$
p	Randomly generated quantity
p_0	Parameter used to attain quick convergence of the algorithm
$popsize$	Total number of solutions generated in the population
q	Random number
Q_c	The probability that there are exactly c servers are available
$R_{DM_i}(X)$	The reliability of system when there is no deadline-miss DM_i
$R_{k, kb, DM_i}(X)$	The reliability of system with no deadline-miss DM_i in addition to node k and link l_{kb} are operational
$R_k(X)$	The reliability of system when node k is operational
$R_{kb}(X)$	The reliability of system when link l_{kb} is operational
$R_{k, kb}(X)$	The reliability of system when both node k and link l_{kb} are operational
$R_{k, kb, DM_i, A_\lambda}(X)$	The reliability with no deadline-miss DM_i in addition to node k and link l_{kb} are operational considering the conditional steady-state availability A_λ of resources
S_i	The set of nodes allowed at the next step by ant m
T	Set of transactions
T_i	i^{th} transaction $\forall i = 1, \dots, m$
T_j	Total time for execution of allocated tasks on the j^{th} node
T_g	The maximum time taken by any node when $g \in j$
T_{miss}	Number of transactions that have missed their deadline
T_{total}	Total number of transactions

t	time
Vib_{best}	Strongest vibration among all nodes
Vib_{N_k}	Vibration of k^{th} node
$Vib_{N_{thres}}$	Threshold vibration
Vib_{tar}	Vibration of target node
$W_j(t)$	Waiting time at the j^{th} node
w_{kb}	The transmission rate of link l_{kb}
X	An m by n binary matrix corresponding to a task allocation
y_{ik}	The memory required by all the transactions at k^{th} node $\forall i = 1, \dots, m$
α	Factor that controls the importance of the trail
β	Factor that controls the importance of the visibility
$\eta_{j\bar{j}}$	Visibility from node j to node \bar{j}
λ_j	Task arrival rate at the j^{th} node
μ_j	Service rate of the j^{th} node
τ_j	Pheromone trail deposited on the j^{th} node
$\Delta\tau_j$	Quantity per unit time of pheromone trail laid on the node j by the i^{th} ant between time t and $t +$ number of iterations till this instant
ρ	Evaporation of the trail
$\{\phi\}$	The null set
γ	Failure rate of node
σ_{kb}	The failure rate of communication link l_{kb}
ψ_i	The rate of the deadline-miss failure in transaction T_i
η	Repair rate of node
λ	Arrival rate of transaction
μ	Processing rate of transaction
Π_i	the steady-state probability for an $M/M/c$ model

*This thesis is dedicated to my parents
For their endless love, support and encouragement*