ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

Volume 3, Special Issue 3, March 2014

2014 International Conference on Innovations in Engineering and Technology (ICIET'14)

On 21st & 22nd March Organized by

K.L.N. College of Engineering, Madurai, Tamil Nadu, India

Truthful Dynamic Scheduling Algorithm Based on Multi objective for Multi cloud Environments

M.Geethanjali, J.Angela Jennifa Sujana T.Revathi

P.G Scholar, Dept. of Information Technology, Mepco Schlenk Engineering College, Sivakasi, TamilNadu, India.

Assistant Professor (Sr. Grade), Dept. of Information Technology, Mepco Schlenk Engineering College, Sivakasi, TamilNadu, India.

Senior Professor & Head, Dept. of Information Technology, Mepco Schlenk Engineering College, Sivakasi, TamilNadu, India.

Abstract— Job scheduling is a complex problem for large environments like Clouds.Profit maximization for the Cloud Service Provider (CSP) is a key objective in the large-scale, heterogeneous, and multi cloud environments. In these types of environments, the service providers want to increase their profit but the customers (end-users) want to minimize the costs. Therefore the goal of selfish providers contradicts with the user's objective. So we design a new model based on multi cloud environment. In the proposed system, we try to negotiate with the different cloud service provider's services. In the negotiation process, we consider two main criteria: the time taken for executing the job and the cost the user has to pay for the job. Considering these two main objectives for our model, we incorporate a truthful algorithm for scheduling the job with respect to task completion time and monetary cost. The truthful algorithm uses game theory for deciding the winner. The experiments conducted using randomly generated workflows and real world workflows. We discovered that the generated solutions of the proposed mechanism are effective with multi-objective optimization.

Keywords— Scheduling, multicloud environment, game theory, reverse auction, truthful mechanism.

I. INTRODUCTION

Cloud computing is the delivery of computing services over the Internet. Cloud services allow individuals and businesses to use software and hardware that are managed by third parties at remote locations. Task scheduling is one of the major issues in cloudcomputing.Scheduling is usually based on trusting the private information about the status of resources which is provided by the Cloud Service Providers (CSP) those are present in cloud environments.

Usually the Cloud Service Providers submit this private information to a Cloud Information directory Service (CIS). CIS is used to store information about the resources belonging to the existing cloud providers.According to this information only the job is scheduled. Therefore this information must be true and is accessible to schedulers.

The users consider this information as complete and accurate. They can able to compare the offerings from different providers and move from one provider to another in order to achieve their objectives i.e., minimum time for the job execution and monetary cost. To use the services provided by the service providers, the users (consumers) need to pay providers only when they access computing services.

The main aim of CSP is to earn more amount of money but the customer wants their job execution with minimum payment. Hence, discrepancy exists between Cloud Service Provider and users. Game theory is used to solve this kind of problem. In cloud environments, users, Cloud Service Providers (CSPs), and brokers form the set of players [1]. Users submit jobs and require a variety of services for processing them. Brokers are user agents who are responsible for resource discovering, submitting jobs to the CSP, collecting the results, etc., and Service Level Agreement (SLA) managers who establish agreements between users and providers. Players periodically broadcast their current states throughout the system [16]: CSPs advertise their idle resources while users update their job descriptions and their beliefs of the resource demand by studying the current status of the system.

II.RELATED STUDY

A. Multicloud scheduling

Li et al.[9] tried to summarize a new optimization approach in clouds. In clouds, QoS guaranteeing is a significant work. Li et al. built a performance model to invest the cloud.

A novel approach of heuristic-based request scheduling [2] at each server, in each of thegeographicallydistributeddata centers, to globally minimize the penalty charged to the cloud computing system. They evaluate two variants of our heuristic-based approach, one based on the simulated annealing method of neighborhood searches and another based on gi–FIFO scheduling, which has been analytically proven to be the best schedule for percentile goals in a single machine, multi-class problem.

B. *Auction-Based Scheduling*

Challenges and requirements of the marketbasedgrid systems were thoroughly explored, andseveral auction-based resource management techniqueshave been put into practice in recent years. Grosu and Das rated three auctionprotocols for resource allocation in grid systems [10], andshowed that suppliers are better off selling their resourcesaccording to first-price (FP) auctions rather than secondpricemechanisms. Garg et al. [6] presented atruthful scheduling mechanism and studied the antisocialbehavior of service providers in distributed computingenvironments. Regev and Nisan [7] proposed the application of Vickrey auctions for allocating homogeneous computationalresources in distributed systems. Wellman et al. [13]presented an auction-based protocol to schedule resourceswith regard to different time constraint considerations.

C. MultiObjective Scheduling

Multi-objective genetic algorithm (MO-GA) [3] is designed and the research is focused on encoding rules, crossover operators, selection operators and the method of sorting Pareto solutions. Compared to existing scheduling algorithms, the results show that the proposed algorithm can obtain a better solution, and it provides a balance for the performance of multiple objectives.

D. Summary

All the above mentioned techniques are focused with static approaches that do not consider the dynamic load of resources in a real environment. Moreover, all the techniques believe that the resource information published by the providers is correct, which is doubtful in commercial clouds. Since the problem in this project involves multiple clouds, there is a need for the system that can cope with the environment requirements in presence of selfish providers [15].

III.PROPOSED SYSTEM

A. *MultiCloud Environment*

MultiCloud, which envisions a marketplace that enables brokers and providers to improve performance, reliability, and scalability of elastic applications by leveraging resources from multiple Clouds in order to seamlessly meet the user's demand by scaling them across various data centers. MultiCloud envisions market-oriented policies for provisioning of virtual machines across multiple data centers that can be adopted solely or together with ad hoc policies enforced by providers. Fig. 1 shows the multi cloud environment.

The Cloud Coordinator is the element that has to be present on each data center that wants to interact with MultiCloud parties. It is responsible for accepting all orders and verifying accuracy and completeness, providing customer support as needed and for communicating with other service providers for required information in order to provide efficient resource allocation. The Cloud Coordinator benefits both users and brokers in acquiring resources via MutiCloud and do not own resources to negotiate in the market.

In this case, the Cloud Coordinator can be seen as a data center that has no available local resources, and thus it always buys resources in the MultiCloud market place in response to changes in user's demand. i.e., The Cloud Coordinator allows providers to trade resources in response to changes in user requests.

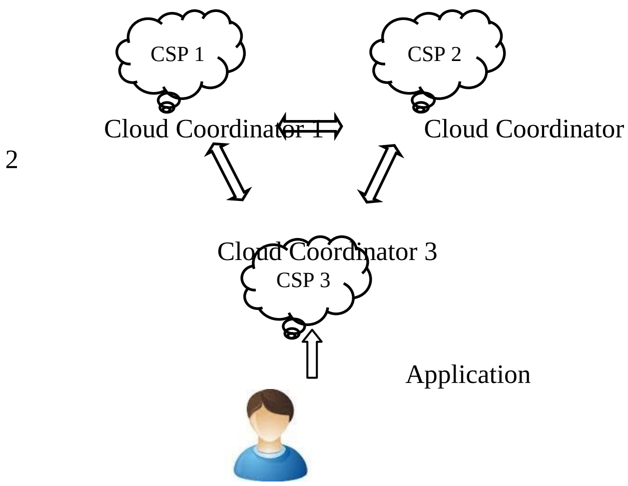

Fig. 1.MultiCloud Environment

The Cloud Coordinator also offers services so that other parties can negotiate resources/requests and also access services offered by the other CloudCoordinators. It also interfaces with the rest of the data center components so resourcesare bought and sold according to the data center demand. In this sense, the Cloud Coordinatoracts as a trading agent [8], even though its capabilities are not restricted to those ofa trading agent. In fact, the Cloud

Coordinator contains a trading agent, but it also carriesout other tasks : its backend contains a virtual machine management integration layer that allows utilization of available virtual machine management technologies, and its frontend is composed of a web service that communicates with other Cloud coordinators and a market engine that evaluates offers and resources(requests processing).

Because Cloud Coordinator has information about the local infrastructure and the corresponding utilization rate, as well as access to other Coordinators, this is the component where policies for trading of resources must be implemented.Also the Cloud Coordinators of the participating providers periodically update their details within this information registry [5]. Therefore a Cloud Coordinator is used for exporting Cloud services and their management driven by market-based trading and negotiation protocols for optimal QoS delivery at minimal cost and energy in multicloud environments.

B. *Need for MultiCloud Environment*

Multi-cloud strategy is the concomitant use of two or morecloud services to minimize the risk of widespread data loss or downtime due to a localized component failure in a cloud computing environment. A multicloud approach can steer traffic from different customer bases or partners through the fastest possible parts of the network. Some clouds are better suited than others for a particular task. For example, a certain cloud might handle large numbers of requests per unit time requiring small data transfers on the average, but a different cloud might perform better for smaller numbers of requests per unit time involving large data transfers on the average. Some organizations use a public cloud to make resources available to consumers over the Internet and a private cloud to provide hosted services to a limited number of people behind a firewall. A third type of cloud, called a hybrid cloud, may also be used to manage miscellaneous internal and external services.

C. Architecture

The multicloud environment comprises x selfish cloud providers, is illustrated in Fig. 2. A set of selfish cloud providers is present in the bottom layer. Wedesign a cloud information directory service which isto store the private information about the resources is established in the top of this layer. This service is directly used in the brokerage layerimplementing our proposed pricing model and schedulingmechanism by selecting the most adequate resources interms of execution time and monetary cost for users. Thetop layer is the user application connected to the brokeragelayer for scheduling purposes and to the bottom layer forsubmitting tasks.We assume without loss of generality that each cloudprovider has only one resource. The goal of each provider is to maximize its profit by executing as many tasks as possible.

D. Workflow Model

The workflow is modeled as Directed Acyclic Graph. The DAG consists of nodes and its corresponding edges that is represented as $DAG(V, E)$, where V is the set of nodes representing m dependent tasks and E indicates edges which represents the control and data flow dependencies between tasks. Each task m is characterized by its workload $weight(m)$and $E = (m, n, Trans(m, n))$ where $Trans(m, n)$ is the output of tasks m to n. Each workflow has an entry task and an exit task. Entry task is the first job enters into the system which has no predecessors. Exit task is the last job which has no successors.

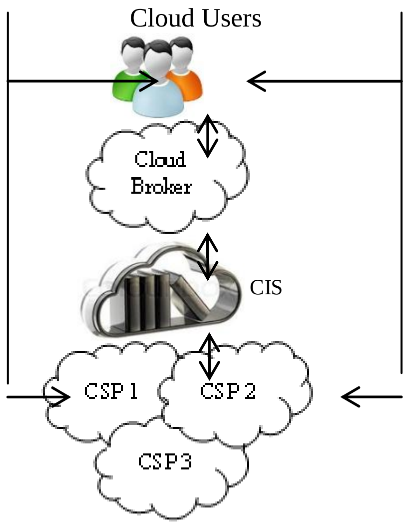

Fig.2.Multiple Cloud Service Providers Services

E. Scheduling

The workflow scheduling problem is defined with respect to two simultaneous minimization objectives: makespan and monetary cost of the execution. Each individual schedule of a task m is denoted as a pair (m, n) which means that the task m is assigned to resource$n$. The schedule of the entire workflow is denoted as $schedule = \{(m, n) | m \in V\}$. We assume the resource n is able to calculate the real completion time $time_{m,n}^{real}$ and the real computing cost $cost_{m,n}^{real}$ for executing every task$m$. For each resource, the real completion time of a task is sum of two components: the time to transfer the input data and the effective execution time.

To calculate the real completion time of a task, the resource must consider a number of internal details such as the virtual machine startup overhead, latency delay, current load, computing power, availability, ready time, communication bandwidth, task workload, and so on. Suppose that these calculations are performed by each provider internally and, therefore, ignore them in our model. The real cost of executing a task on a resource is again internally calculated by the provider based on its business model and by considering internal maintenance, operational, or energy consumption costs.

Before the execution of task m, $time_{m,n}^{real}$ is private information for the resource n and not accessible to any user or other resource. After the completion of the task, the user will be aware of $time_{m,n}^{real}$, but will still not know $cost_{m,n}^{real}$. Instead, the user will only know the price calculated by the broker.

Based on the real completion time of a single task, we define the workflow makespan as the time required for executing the whole workflow without violating the control and data flow dependencies between its tasks:

$$Makespan[DAG(V, E)] = \max_{m \in V \wedge (m,n) \in schedule} time_{m,n}^{real} .$$

Similarly, we define the cost as the sum of the costs of executing all workflow tasks:

$$Cost[DAG(V,E)] = \sum_{m \in V \wedge (m,n) \in schedule} cost_{m,n}^{real}.$$

IV.BI-OBJECTIVE SCHEDULING

The current cloud providers usually charge users based on a pay-as-you-go pricing model. With respect to our multi provider cloud model and the two considered objectives (makespan and monetary cost), a new pricing model and truthful scheduling mechanism to find the best resource for executing a task called BiObjective Scheduling approach (BOS) is implemented in the brokerage layer in Fig. 2. The mechanism is based on a reverse auction which is a common tool in a market with lots of sellers. Each auction is based on some rules that is related with game theory which define the identity of the winner. The most famous result in this area is the Vickrey-Clarke-Groves (VCG) which applies to utilitarian (sum of the valuations of the agents) objectives only.

A. Game Theory Concepts

There are several important concepts of game theory with focus on the mechanism design branch [11] [12]:

- Strategy is the action selected by each agent in each round of the game;
- Strategy profile is the set of strategies selected by all agents in each round of the game;
- Rational agent is the agent that selects its strategy based on maximizing its own profit regardless the effect of the selected strategy on the payoff of other agents;
- Nash equilibrium is a strategy profile of the game in which no agents can deviate from its strategy unilaterally and gain more;
- Truthful mechanism is a designed mechanism for which truth-telling is the only equilibrium. In other words, if the strategy profile of a mechanism is Nash equilibrium, the mechanism is truthful.

Therefore the selection of CSP having resources is based on game theory concepts in order to achieve two objectives.

B. BOS Auction

An auctioneer initiates an auction to select a proper resource for each task execution in BOS mechanism. For each task execution, the task's workload, the dependencies with other tasks, and the required input and output are announced to the resources. According to the requirements, the appropriate resource is allocated to the particular task.

C. Strategy Profile

The strategy profile incorporates the strategies of all agents those are participated in the auction that is initiated by task

Strategy profile $sp_{m,\ .} = \cup_{n \epsilon R}(time_{m,n}^{real}, cost_{m,n}^{real})$ is Nash equilibrium.

Strategy is a combination of its proposed time and proposed cost. Each resource n is demanded by the strategy. More than one strategy is allowed to bid the same resource in every auction.

D. Winner Selection

According to the minimum product of cost and time which are proposed by all agents, the winner w is selected in every auction.

$$time_{m,w}.cost_{m,w} = \min_{n \in R}\{time_{m,n}, cost_{m,n}\}.$$

Both objectives and the truthfulness of the mechanism have to be considered. Therefore multiplication aggregation is used for the selection of winner.

E. Payment Calculation

After selecting the winner, the task is submitted to the winning resource. After its completion, the resource will be paid based on the following payment function inspired from the Vickrey's second price auction:

$$Payment(m,n) = \begin{cases} \dfrac{time_{m,z}.cost_{m,z}}{time_{m,n}}, if\ n = w \wedge time_{m,n}^{real} \le time_{m,n}; \\ g(sp_{m,\ .}), \quad if\ n = w \wedge time_{m,n}^{real} > time_{m,n}; \\ 0, \quad if\ n = w. \end{cases}$$

where $g(sp_{m,\ .})$ a function of the strategy profile and z is the resource that proposes the second smallest bid or $time_{m,w}.cost_{m,w} = \min_{n \in R, n \neq w}\{time_{m,n}, cost_{m,n}\}$.The payment function causes the truthfulness of the mechanism is proved in the section A.

V.DYNAMIC WORKFLOW SCHEDULING

Dynamic workflow scheduling is an extension to the BOS mechanism. The proposed mechanism is a scheduling algorithm i.e., Dynamic Workflow BOS Algorithm that first orders the tasks that aresubmitted to the cloud providers according to the descending order of rank algorithm.

We calculate the rank of a task m is calculated according to the following recursive function:

$$\text{Rank(m)} = \begin{cases} \text{weight(m)} + \max_{\text{n} \in \text{desc (m)}}\{\text{Trans(m, n)} + \text{rank(n)}\}, \text{if m} \neq \text{exit;} \\ \text{weight(m)}, \quad \text{if m} = \text{exit.} \end{cases}$$

where $weight(m)$ represents the workload of task m, exit denotes exitTask and $Trans(m,n)$ is the output of tasks m to n.

Function $desc(m)$ returns the immediate successors of task m. After the ordering of tasks, the appropriate resources are allocated to the corresponding tasks that are in established order using BOS auction.

Finally, the payment for the task execution is calculated by the broker according to the payment function which is explained in the section E.

```
Algorithm: Dynamic Workflow BOS Algorithm
Input: The workflow application DAG(V, E)
Output: Workflow Scheduling
begin
   tasksList ← V;
   ranksList ← null;
   tempTask ← exitTask;
   while tasksList ≠ null do
      Append(ranksList, CalcRank(tempTask));
      tempTask ← Tail(tasksList);
      Remove(tasksList, tempTask);
      tempTask ← Tail(tasksList);
   end
   tasksList ← V;
   /
* Sort the tasksList according to the descending
   orderof ranksList */
   Sort(tasksList, ranksList);
   while tasksList ≠ null do
      auctioneer ← Head(tasksList);
      w ← BOS(auctioneer);
      Submit(auctioneer, w);
      Remove(tasksList, auctioneer);
End
Payment();
End
```

VI. EXPERIMENTAL SETUP AND DISCUSSIONS

We performed experiments with two types of workflows. They are randomly generated workflow using CloudSim and Real-world workflow. Real world workflow has been taken from Shiwa workflow repository (http://www.shiwa-workflow.eu/). The cloud environment is created with three service providers. We estimated the costs of resources using Amazon, Microsoft and Rackspace Web Services prices as précised in the Table I.

The workflows have been simulated using CloudSim simulator. The simulation performs,

1. The task scheduling according to the descending ranks order.
2. Strategy profile creation for each task along with each data center.
3. Selection of winner with respect to the minimum product of time and cost associated with each service provider.
4. Finally the cost payment to the winner for each task execution.

Table I
SUMMARY OF RESOURCE COST S

Resource Type	Cost($)		
	Amazon	Microsoft	Rackspace
Data Transfer	0.120	0.120	0.155
Storage	0.085	0.07	0.10

The solution of BOS is compared with two multi-objective evolutionary algorithms: SPEA2 [16] and NSGA-II [4] genetic algorithms. SPEA2 and NSGA-II are categorized as a-posteriori multi-objective optimization techniques [14], because they do not assume any a-priori knowledge aboutthe Pareto-set solutions.

In most cases, we can observe that the BOS algorithm generates non dominated solutions compared to the SPEA2and NSGA-II and in some cases, the solution of BOS even dominates them. Although theoretically no solution is able to dominate the Pareto set, this is possible in our case since the Pareto sets generated by the two evolutionary algorithms are only estimations.

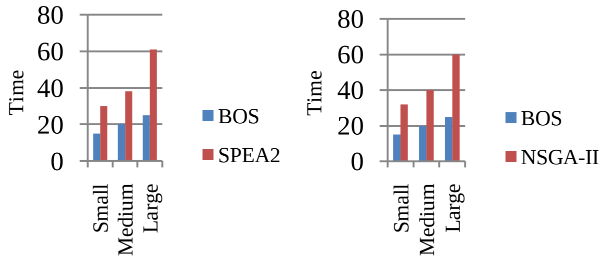

Fig. 3.Average Execution Time for Tasks Execution

TheBOS algorithm is a much faster than SPEA2 and NSGA-II, as showed in Table II.

Table II
THE AVERAGE EXECUTION TIME FOR TASKS

	Small	**Medium**	**Large**
BOS	14.32s	19.82s	22.51s
SPEA2	25.01s	39.71s	61.96s
NSGA-II	24.21	36.41s	59.82s

The results show the considerable differences in execution times. The reason for this high difference is the second degree polynomial time complexity of BOS, while the MultiObjective Evolutionary Algorithms (MOEAs) are complex evolutionary algorithms.

VII.CONCLUSION

In this paper, the problem of multi-objective workflow scheduling is visualized. We implemented truthful algorithm for dynamic scheduling of a single task in commercial multi cloud environment. The current approach is assessed by simulation runs on a set of synthetic randomly generated workflows and real-world workflow applications. The workflow scheduling which aims to minimize the cost and total time execution of user tasks has been carried out using the dynamic workflow scheduling algorithm with respect to the optimization of two main objectives, makespan and monetary cost. This algorithm negotiates with the different cloud service provider's objective. Therefore the truthfulness of the service providers in multi cloud environment has been proved using game theory concepts.

Simulation results showed that this new multi-objective algorithm significantly improves the performance of related approaches and discovered that the generated solutions of our proposed mechanism are approximately Pareto optimal solution set based on multi-objective paradigms that satisfy the user's objectives. Finally, we evaluate the proposed mechanism by comparing it with two classical multi-objective evolutionary algorithms namely SPEA2 and NSGA-II. The experimental results proved that our proposed mechanism yields much smaller execution times compared to the investigated MOEAs.

REFERENCES

[1] Broberg.J, Venugopal.S, and Buyya.R, "Market-Oriented Grids and Utility Computing: The State-of-the-Art and Future Directions", *J. Grid Computing*, vol. 6, no. 3, pp. 255–276, 2008.

[2] K. Boloor, R. Chirkova, T. Salo, and Y. Viniotis, "Heuristic-based request scheduling subject to a percentile response time SLA in a distributed cloud," *in Proc. IEEE GLOBECOM*, 2010.

[3] Jing Liu, Xing-GuoLuo, Xing-Ming Zhang, Fan Zhang and Bai-Nan Li, "Job Scheduling Model for Cloud Computing Based on Multi-Objective Genetic Algorithm", *IJCSI International Journal of Computer Science Issues*, Vol. 10, Issue 1, No 3, January 2013.

[4] Deb.K, Pratap.A, Agarwal.S, and Meyarivan.T, "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II", *IEEE Trans. Evolutionary Computation*, vol. 6, no. 2, pp. 182-197, Apr. 2002.

[5] Buyya R, Ranjan R, andCalheiros RN. "InterCloud: Utility-Oriented Federation of Cloud Computing Environments for Scaling of Application Services", *in Proc.Of the 10th international conference on Algorithms and Architectures for Parallel Processing*, Springer, pp.13–31, 2010.

[6] N. Garg, D. Grosu, and V. Chaudhary, "Antisocial Behavior of Agents in Scheduling Mechanisms," *IEEE Trans. Systems, Man, Cybernetics*, Part A: Systems and Humans, vol. 37, no. 6, pp. 946–954, Nov.2007.

[7] O. Regev and N. Nisan, "The POPCORN Market. Online Markets for Computational Resources," *in Decision Support Systems*, vol. 28, nos. 1/2, pp. 177–189, 2000.

[8] K. M. Sim, Agent-based cloud commerce, in Proc. of the IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), *IEEE, Hong Kong*, 2009, pp. 717–721.

[9] Li.J, Chinneck.J, Woodside.M, Litoiu.M, and Iszlai.G, "Performance model driven QoS guarantees and optimization in clouds", presented at Software Engineering Challenges of Cloud Computing,*ICSE Workshop on, Vancouver, BC*, Canada, 2009

[10] D. Grosu and A. Das, "Auctioning Resources in Grids: Model and Protocols," *Concurrency and Computation: Practice and Experience*, vol. 18, no. 15, pp. 1909–1927, 2006.

[11] Nisan.N and Ronen.A, "Algorithmic Mechanism Design", *STOC*, pp. 129-140, 1999.

[12] Nisan.N, Roughgarden.T, Tardos.E, and Vazirani.V, "Algorithmic Game Theory",*Cambridge Univ.* Press, 2007.

[13] M.P. Wellman, W.E. Walsh, P.R. Wurman, and J.K. MacKie-Mason, "Auction Protocols for Decentralized Scheduling,"*Games and Economic Behavior*, vol. 35, nos. 1/2, pp. 271–303, 2001.

[14] Marler.R.T and Arora.J.S, "Survey of Multi-Objective Optimization Methods in Engineering", *Structural and Multidisciplinary Optimization*, vol. 26, no. 6, pp. 369-395, Apr. 2004.

[15] Hamid MohammadiFard, RaduProdan, and Thomas Fahringer, "A Truthful Dynamic Workflow Scheduling Mechanism for Commercial Multicloud Environments", *IEEE transactions on parallel and distributed systems*, vol. 24, no. 6, june 2013.

[16] Zitzler.E, Laumanns.M, and Thiele.L, "SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization", *in Proc. Of Int'l Conf. Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems (EUROGEN '01)*, pp. 95-100, 2002.