A Mathematical Predictive Model for an Autonomic System to Grid Environments

Alberto Sánchez and María S. Pérez

Computer Science School,
Universidad Politécnica de Madrid,
Madrid, Spain
{ascampos, mperez}@fi.upm.es

Abstract. One of the most important aims of the Grid technology is using geographically distributed resources. Nevertheless, Grid environments have a great problem: the system management is very complex because of the large number of resources. Thus, improving the system performance is a hard task and it would be advisable to build an autonomic system in charge of the system management. The autonomic system must take decisions based on the analysis of the monitored data of the whole Grid trying to improve the system performance. These decisions should not only take into account the current conditions of the Grid but the predictions of the further future behaviour of the system too. In this sense, we propose a mathematical model to decide the optimal policy based on predictions made thanks to the known past behaviour of the system. This paper shows our model on the basis of the decision theory.

Keywords: Grid computing, Autonomic computing, performance models, predictive models.

1 Introduction

A Grid environment [Fos02] could be understood as the result of the geographically distributed computing and storage resources. As system, we should try to get the best performance in any kind of access to any grid element or the whole environment.

Although the Grid community has changed its directions towards a services model, as is described in [FKNT02] and [CFF$^+$], the high number of resources, which constitute the grid, makes difficult the system management and therefore to obtain the maximum system performance.

In order to make easier the system management, administrators must not be in charge of solving the whole system complexity and they could be helped by an autonomic system. Autonomic computing [RAC] [Iac03] is used to describe the set of technologies that enable applications to become more self-managing. Self-management involves self-configuring, self-healing, self-optimising, and self-protecting capabilities. The word *autonomic* is borrowed from physiology; as a human body knows when it needs to breathe, software is being developed to enable a computer system to know when it needs to configure itself and in our case study optimise itself. Taking decisions based on the data analysis is one of the problems that must be solved if we want to

O. Gervasi et al. (Eds.): ICCSA 2005, LNCS 3482, pp. 109–117, 2005.

obtain and use the strategy that optimises the system and maximises its expected performance.

This decision problem, seen in the field of system analysis and evaluation, can be found in another different areas. For example, in Health Sciences different strategies have been studied to help to take decisions about the illness of a patient, analysing the observed facts and symptoms found in him, like in [Dei] and [Abi99].

Sciences, like Phisics and Chemistry, are characterised by establishing deterministic laws. Thus, if an experiment is repeated in same conditions, same results are obtained because of they are defined by natural laws. Nevertheless, in our field, the usual situations are random and do not comply a natural law. Thus, it is necessary to make a probabilistic data analysis based on the statistic regularity principle. This principle states that if the number of repetitions of a experiment is large, its relative frequency tends to be stabilized in the value represented by its probability.

This probabilistic data analysis should analyse the current state of the system. However, only this feature is not enough. It is required to predict the future system behaviour to select the best strategy that maximises the expected system performance. We must increase the system performance in a concrete point of time and in future actions. Nowadays, there is a lack of mathematical predictive models to take decisions suitable for Grid environments. This paper shows a predictive model based on different techniques, which are used in decision theory and could be used in an autonomic system to make easier this hard task.

The outline of this paper is as follows. Section 2 defines the background of known mathematical models used to build our approach. Section 3 describes our proposal, which is based on the mathematical models shown below. Finally, Section 4 explains the main conclusions and outlines the ongoing and future work.

2 Statistic Models

Getting the best strategy adapted to the necessity of taking decisions is a hard problem. With the aim of simplifying it, first of all, we will use different known techniques.

2.1 Decision Theory

Keeney and Raiffa in [KR93] define the decision analysis as a procedure that helps to people to decide when they have to face difficult situations.

The are several methods to select the best decision. Each one shows a different perspective of tackling the problem.

Maximin. This method uses the criterion of selecting the action that has the best of the worst possible consequences.

In this sense, it is possible that the selected action does not maximise the profit because the aim is obtaining a result that was not harmful. This method could be useful in an autonomic system that wants to improve the response time of the set of applications running over it without getting the minimum respond time of a particular application.

Expected Value. The idea of this method is assigning occurrence probabilities to each event and select the decision whose expected value was the highest one. For calculating the expected value of each action is necessary to add the result of multiplying the assigned values to each action, if the event occurs, and its occurrence probability.

The problem is how to obtain the occurrence probabilities of the events. These probabilities can be predefined or can be extracted from the past behaviour of the system. In this last case, it would be necessary to accept that the system behaves in the future in a similar way than in the past.

2.2 Decision Trees

In [RS61] the origins of decision trees can be found. Decision trees are a very useful resource to represent decision problems composed of several decisions, making easier the understanding and the resolution of complex problems.

In short, a decision tree is a probability tree that has three types of nodes:

1. Decision nodes. They have tree branches that represent possible decisions.
2. Chance nodes. They represent the possible states.
3. Value nodes. They are terminal nodes that indicate the utility associated to the taken decisions and states visited.

A decision tree can be seen as a set of policies that indicate action plans. The evaluation of decision trees aims identifying an optimum strategy, searching the strategy that maximises the expected utility.

The use of decision trees is useful to represent system policies and how they affect to the system. The disadvantage of solving the system using decision trees is that the change state probabilities must be defined "a priori", and this makes difficult the prediction of future behaviours.

2.3 Bayesian Approach

One of the most important strategies to take decisions is the Bayesian approach. This approach uses the famous Bayes theorem presented in 1763 in [Bay63].

The Bayes theorem is enunciated as the following: being A_1, A_2, ..., A_n a complete system of events with probability different to 0, and being B another event whose conditional probabilities are known $p(B|A_i)$, then the probability $p(A_i|B)$ is:

$$p(A_i|B) = \frac{p(A_i) \times p(B|A_i)}{\sum_{i=1}^{k} p(B|A_i) \times p(A_i)}$$

By using this theorem, we can represent an objective decision procedure taking into account the appearance probability of a certain event based on real facts. This allows us to calculate the probability an event happens while different additional information about the occurrence or not of related event is incorporated.

In short, the steps of the Bayesian approach are the following:

1. Calculate the "a priori" probability that A event occurs.
2. Calculate the probability that different events occur which depend of A, knowing that A has happened.
3. Calculate the "a posteriori" probability the A event occurs knowing that the other events have occurred by means of Bayes theorem.

The Bayesian approach is the most used selection method. In our case, it is not useful to make predictions about the system behaviour. In conclusion, the Bayesian approach allows us to know the probabilities of being in a certain state of the system based on monitored data. But this does not allow us to infer which is the probability that in the future the policy applied by this method maximises the system performance.

Although the Bayesian approach does not solve our full problem, it is an interesting start point to calculate the initial probabilities.

2.4 Markovian Approach

The Markovian approach is based on Markov chains. A Markov chain is a stochastic process that represents a system whose state can change. These changes are not predetermined, although the probability of getting the next state depending on the previous state, is known [Dav93]. In addition, this probability must be constant along time.

Markov chains include in the probabilistic analysis remunerations, numeric value associated to the transition between two states, without assuming nothing about the indicated value nature. This value could include every aspect needed for the model, as losses as profits. After a large number of steps, the transition probabilities between states reach a limit value called *stationary probability*. This one can be used to take decisions.

It is possible that the stationaries probabilities do not depend on the initial state. In this sense, all probabilities tend to a limit value since we increment the number of transitions among states. All states can be reached in the future and the behaviour does not change along time.

A Markovian process is said to have decision if for each transition a variable can be fixed and it is possible to choose between different sets of transition probabilities and different values from associated remuneration.

A Markov chain with remuneration is created from the definition of policies, rules that fixes for each state the value of the decision. For each policy it is necessary to define the probabilities of transition between the different states and the existing remunerations between them.

Due to the decision capacity of the system, the maximum remuneration in a large period of time,is obtained. This maximum remuneration policy is called *optimum policy*.

In order to establish the optimal policy, two methods can be used:

1. Iterating in the state space until the system converge to a certain policy that maximises the expected remuneration.
2. Iterating in the policy space. It allows the problem to be solved in an infinite horizon.

Thus, it will be necessary to define:

1. A matrix of transition probabilities between states.
2. A matrix of benefits or losses obtained when a transition between two states is made. At first the predefined values of remuneration will indicate the obtained benefits if this decision is taken.
3. The establishment of policies in the system. For each policy, probabilities and remunerations must be defined.

Although the Markovian approach allow us to infer the probability to reach a certain state in a further future (infinite horizon), it is necessary not to forget that the future behaviour is random and we are only inferring its behaviour in a probabilistic way.

3 Proposal

The concept of grouping is fundamental in every aspect of the life. Edwin P.Hubble, which is considered the founder of the observational cosmology, said in the thirties that the best place for searching for a galaxy is next to another one, describing the concept of galaxy grouping. Like in real life, computer science has a significant number of groupings, such as process group or user group, which are used for representing sets of objects from the computing field.

In a grid environment, the concept of grouping is very important. The use of different clusters that belongs to the grid generates the abstraction of the server concept, entrusting to the cluster instead of the server the storage of the information. This causes the need of knowing the parameters in a cluster level.

For knowing this parameters, a mathematical formalism should be defined. These parameters could be monitored to improve the decisions about its performance. Two types of parameters can be defined that must be managed by the system:

1. Basic parameters. They lead the operation of the system. By its influence on the performance we might emphasise:
   - Capacity (C). Occupation percentage of the hard disk of each server.
   - Load of the Network (N). Busy rate of the network.
   - Workload (W).
2. Advanced parameters. Parameters that have an influence in the performance of the autonomic system. It is very important:
   - Time window (T). Period of time in which the system monitors its performance. It should be calculated depending of the environment.

But it is necessary to consider and monitor every node belonged to the cluster to know the value of the cluster parameter. The parameter model based on the grouping concept is shown in Figure 1.

There are different methods to figure out the cluster parameters based on the nodes parameters. In the case seen in Figure 1, the parameters values for the cluster are the worst values of all nodes that compose it. In this sense, the system makes decisions based on the worst possible case. Other policies that the system can use are based on other variants.

The autonomic management of the system must face the decisions taken in a certain time trying to improve the future behaviour of the system. The decisions taken in the present should consider the future previsions. For predicting, it should be very useful to consider the past behaviour of the system.

In order to know the past behaviour of the system the events must be monitored each certain period of time corresponding to the time window (T). Then, we can calculate the occurrence probabilities of each event that are updated when new occurrences occur into the system.

Being $G_1, \ldots, G_n$ clusters

where $G_j = \{S_1, ..., S_m\}$ $\forall$ i $\in \{1, ..., n\}$

Being C_{ji} = Node capacity $S_i \in G_j$

Then C_j = min C_{ji} $\forall$ i $\in \{1, ..., m\}$

and C_j is the capacity in a parallel cluster G_j

Being N_{ji} = load of the network to access to the node $S_i \in G_j$

Then N_j = max N_{ji} $\forall$ i $\in \{1, ..., m\}$

and N_j is the load of the network to access to the cluster G_j

Being W_{ji} = workload of node $S_i \in G_j$

Then W_j = max W_{ji} $\forall$ i $\in \{1, ..., m\}$

and W_j is the workload of the cluster G_j

Fig. 1. Parameters model

The system evolves in a non deterministic way around a set of states. Due to this behaviour, the system operation can be formalised following the lines indicated by the statistical method of Markov chains. This statistical method provides the capability of taking decisions.

The initial probabilities for the Markovian approach can be calculated from the data acquired previously into the system. To obtain the values of probability considering the parameters that have an influence in the performance, the Bayesian approach with the formula of the inverse probabilities can be used, calculating the probability that the parameters of the system take certain values in a certain state. By means of Bayes theorem, the inverse probability can be obtained, that is, the probability of staying in a state when parameters have certain values.

This probability calculated for the state changes will not be determinate but it varies with the changes that occur into the system. The Markovian approach defines the probability of reaching a state as a constant, but our model must allows changes of probability along time.

Considering that the obtained probabilities of the system study will vary along time and the predictions can change, the system must be represented by different Markov chains. Each Markov chain will formalise the future operation of the system in a certain moment. A homogeneous operation of the system is assumed so that the probabilities obtained by the data analysis in the past are taken as reference in the future to predict the system behaviour. Therefore, we build a new Markov chain each T time.

On the other hand, everything that helps to make better predictions should be taken into account. Therefore, the autonomic system should make possible the use of hints of the future system performance. To do this, the autonomic system must modify the remuneration predetermined to the transitions between states based on the extracted knowledge of the use of hints. The hints could change the predefined values of benefit assigned to the Markov model, fitting better the behaviour predicted of the system.

The number of states depends on the number of parameters that define the system. In a first approach, for each parameter, it should be advisable to define a state that represent its normal values, another with optimal values and a last state for values that make worse the system performance. The number of states by parameter could be increased to model in a better way the system, but the complexity of the problem grows exponentially and requires huge calculation time than it could not be assumed if a real time analysis is desired.

After a certain number of transitions sufficiently high, the information of the initial state will have been lost, and the probability that the system will be in a state will depend on this one. In this sense, predictions of the behaviour of the system can be made because the possibility of being in a state is known and it is possible to take decisions based on the collected data.

For the inclusion of the decision policies in the system and mainly with the aim of formalising its representation, decision trees can be used. In this sense, policies are represented by means of decision trees, but the Markovian approach will be applied to solve the optimal policy, being able to take decisions based on the values of transition probability between predicted states.

The different defined policies that manage the system operation can be transformed into a Markov chain with remuneration. Its evolution will be defined by an evolutionary sequence that follows a Markovian decision process that affects to the transition to the following stage. This kind of systems is denominated E/D (Evolution/Decision) systems.

4 Conclusions and Future Work

In this paper we have deeply analysed the way of obtaining a better performance in a Grid environment by using an autonomic system. We propose to improve the taken decisions selecting the optimum policy. This optimum policy should not only take into account the current conditions of the Grid but the future behaviour of the system. We have analysed different mathematical approaches to get the benefits that can contribute to our model.

In summary, our goal is to make a prediction about the future behaviour of the system according to the analysis of the current and the past system states. Thus, it will be necessary to do the following steps:

1. Finding the occurrence probabilities of being in a certain state, the values of the system parameters are in a certain rank. Thus, system logs and different monitoring tools could be used.

2. Finding the probability that the system is in one or another state knowing that the parameters of the system have a certain value by means of the Bayes theorem.
3. Creation of the initial matrix of probabilities of transition between states, from the previously collected data.
4. In case of the use of hints, it will be necessary to modify the matrix of remunerations obtained previously based on these ones, which indicate the future operation of the system. Thus, it is necessary to increase the remuneration in the transition towards those states that have a greater probability of occurrence according to indicated in the hints.
5. Calculating the matrix of remunerations, trying to lead the system towards the most beneficial states to reach an improvement in the system performance.
6. Establishment of policies in the system formalising them by means of decision trees. The Markovian problem will be affected so that the previous matrices can have different probabilities and remunerations for each policy.
7. Resolution of the problem of searching the optimal policy that maximises the expected remuneration following the proposed Markovian approach. This will be the decision that will be taken in real time on the system to improve the future performance of such system.

As future work, we aim to adapt this mathematical model to a real environment. Furthermore, we want to study the increase of the number of states that define the problem to adapt it in a better way to environment changes, and how this affects to the computing time. Finally, we will analyse different monitoring tools to find one which can be adapted to our needs and, therefore, which allows us to measure new important parameters.

Acknowledgements

This research has been partially supported by Universidad Politécnica de Madrid under Project titled “MAPFS-Grid, a new Multiagent and Autonomic I/O Infrastructure for Grid Environments”.

References

[Abi99] Syed Sibte Raza Abidi. Ai amidst the healthcare revolution: Towards an ’intelligent’ tele-healthcare environment. In Hamid R. Arabnia, editor, *IC-AI*, pages 172–176. CSREA Press, 1999.

[Bay63] T. Bayes. An essay towards solving a problem in the doctrine of chances. *Philosphical Transactions of the Royal Society*, 53:370–418, 1763.

[CFF+] Karl Czajkowski, Donald F. Ferguson, Ian Foster, Jeffrey Frey, Steve Graham, Igor Sedukhin, David Snelling, Steve Tuecke, and William Vambenepe. The WS-Resource Framework version 1.0.

[Dav93] M.H.A. Davis. Markov models and optimization. *Monographs on Statistics and Applied Probability*, 49, 1993.

[Dei] Stefano Rizzi Deis. Medical decision support in clinical record management systems.

[FKNT02] I. Foster, C. Kesselman, J. Nick, and S. Tuecke. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration, 2002.

[Fos02] Ian Foster. What is the Grid? A Three Point Checklist. *Grid Today*, 1(6), July 2002.

[Iac03] IBM and autonomic computing, editors. *An architectural blueprint for autonomic computing*, April 2003.

[KR93] R. L. Keeney and H. Raiffa. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. Cambridge University Press, 1993.

[RAC] IBM Research Autonomic Computing. *http://www.research.ibm.com/autonomic/*.

[RS61] H Raiffa and R Schlaifer. *Applied Statistical Decision Theory*. Harvard University, 1961.