A Markovian Futures Market for Computing Power

Fernando Martínez Ortuño, Uli Harder and Peter G. Harrison
Department of Computing
Imperial College London
Huxley Building
180 Queens Gate
London SW7 2RH, UK

ABSTRACT

In this paper we describe aspects of a market model for Grid computing. In particular we concentrate on Grid computing provided by a peer-to-peer network architecture. In this network nodes can either buy or sell computing power in exchange for money. Building on previous publications we develop a mathematical market model using Markov chains. The behaviour of each agent in the market is described by a Markov chain of decisions on buying, selling or holding. Considering the contributions of all agents, we calculate the global Markov chain of the market state as a whole, by making use of a concept of *market pressure* that reduces the state space of the entire market model. We show that the Markov chain model describes the market behaviour seen in a simulation extremely well. In a similar way to other perishable commodity markets like fish and electricity, we also provide a model for trading future contracts on the purchase and sale of computing power in this market. Using Markov Decision Processes we derive an optimal trading strategy. This work introduces a pioneer mathematical model for future global peer-to-peer Grid computing architectures like MaGoG (Middleware for activating the Global open Grid), where we have derived a global transition probability matrix that determines the behaviour of the market by summing up the contributions of different kinds of market participants.

Categories and Subject Descriptors

C.4 [**Performance of Systems**]: Reliability, availability, and serviceability; Modeling techniques

General Terms

Performance

Keywords

Grid, Performance, Markov Decision Process, Market, Futures

WOSP/SIPEW'10, January 28–30, 2010, San Jose, California, USA.

1. INTRODUCTION

In the past there have been many papers on using economic ideas to provide fair access to computing resources or profit-making. In 1966 Greenberger [8] reasoned to add cost to queueing systems. Sutherland describes an auction system used at Harvard to provide fair access to a PDP-1 [14]. In 1975 Cotton [5] mentions international computing markets are discussed but not in the context of commodity markets. The advent of the Grid [7] brings the idea of providing computing service as a commodity like electricity. Due to the sheer size of the Grid the allocation and pricing is more likely to happen on the basis of a market like a commodities market. There have been various studies in the past discussing allocation strategies and market implementations [3, 13, 15, 17]. The potential market is likely to consist of a large number of providers and users, and end-users will not be able to make completely rational decisions. In this paper, we consider a global peer-to-peer market for Grid Computing, where consumers and providers freely trade computing power without the need for a central server. The main reason to use peer-to-peer (Catallaxy [2], MaGoG [4]) is that centralised approaches (Tycoon [11]) are less likely to scale as a very large system with a central node. In previous work [9, 10], a simulation approach to a Grid Computing market model was carried out. In this paper we present an integral analytic approach by using Markov chains and Markov decision processes. More generally speaking a market for CPU cycles is going to be a market of a perishable commodity as it is not possible to store unused CPU cycles. Here we introduce future contracts on Grid computing and also investigate an optimal trading strategy using Markov decision processes (MDPs) [12].

2. THE MARKET MODEL

We consider a peer-to-peer market for Grid computing, where agents that buy and sell computing power send messages to each other, trying to look for an agreement without the need for a central server. This scheme closely follows that of the MaGoG system [4]. In this paper we simplify the architecture by allowing each node to buy/sell only one resource at a time as service, as in [9, 10]. Nodes hold on to resources for a fixed length of time.

Previously, we presented an analytical approximation for the price evolution in a simulation of such a system [10]. This approximation assumed that the network was fully connected, and therefore that the system would behave as a single global market place. The results showed that this is actually a good approximation, and therefore we again make this assumption for the present work. Initially, we describe the behaviour of each agent by a Markov chain, approach supported by the fact that most mathematical models for financial markets consider the price follows a Markov process. Other non-Markovian models for this system were presented in [9, 10].

In this paper, for each individual, there are three possible trading actions available: buy(+), sell(−) or hold(0); which correspond to the three possible states of the Markov chain of each agent $\{-, 0, +\}$. The Markov chain of agent i is defined by its transition probability matrix T_i. This has elements $T_i(a, b)$ equal to the probability that agent i moves from state a to state b in the next time slot by taking the appropriate action, for $a, b \in \mathcal{S} = \{-, 0, +\}$. Each agent i takes decisions independently according to his particular T_i and the market is formed by the collection of these agents and hence characterised by the set of all the transition matrices.

The evolution of the market as a whole is determined by the actions of all agents. To model the market, we consider the variation in price, rather than the price itself, in order to avoid having an infinite number of states. For this purpose, we introduce the concept of *market pressure*, which determines the direction of the market price and is established by supply and demand from the agents. An excess of demand will push the price up, whereas an excess of supply will force it down. If supply and demand equalize, the price will remain the same.

We now define a new, discrete time Markov chain to model the evolution of the whole market, taking into account the contributions of the individual Markov chains of every agent. If we were to define a global Markov chain whose possible states were all possible combinations of the individual states of the agents, the global chain that models the market would have 3^N states, where N is the number of agents in the market, since every agent can be in any of his three possible states $\{-, 0, +\}$.

To avoid this state space explosion we make use of the concept of market pressure to reduce the number of possible states in the market model. We define the market state to be the sum of the states of the individual agents, so that a buy action of one agent is cancelled out by a sell action of another agent. With this consideration, the number of states for the global Markov chain is reduced to $2N+1$, namely $-N$ (where every agent is in sell-mode), $-N+1, \ldots, N$ (where all agents are in buy-mode).

2.1 The transition probability matrix of the market

The new, reduced, global Markov chain that models the market as described above has one-step transition probability matrix:

$$M = (m_{sd} \mid -N \le s, d \le N), \tag{1}$$

where each element m_{sd} is the probability of the market going from state s to state d, among the $2N+1$ available states.

In a heterogeneous market, let agent k's Markov chain be irreducible and have equilibrium probability state vector $\boldsymbol{\pi}_k = (\pi_{k-}, \pi_{k0}, \pi_{k+})$, with components corresponding to local states $\{-, 0, +\}$ respectively. Further, let the probability generating function (pgf) of the transition probabilities out of state $i \in \{-1, 0, +1\}$ at agent k have pgf $A_k(z; i) = T_k(i, -)z^{-1} + T_k(i, 0) + T_k(i, +)z$. Now define the N-component vector random variable $\vec{Y}_n$ to be the joint state of the agents just after the nth transition instant, with $\vec{Y}_0$ being the initial joint state. Similarly, let $X_n = |\vec{Y}_n|$ be the corresponding global state just after the nth transition, where $|\vec{v}\,| = \sum_{k=1}^N v_k$ is the sum of the elements of a vector $\vec{v}$. Then, $m_{sd} = \lim_{n\to\infty} \mathbb{P}(X_n = d \mid X_{n-1} = s)$ where the limit exists provided the agent-chains are irreducible. Hence m_{sd} is the coefficient of z^d in the generating function $G(z; s) = \lim_{n\to\infty} G_n(z; s)$ where the pgf $G_n(z; s) = \mathbb{E}\,[z^{X_n} \mid X_{n-1} = s]$, which is what we now determine.

Proposition 1.

$$G(z; s) = \frac{\sum_{\vec{\ell}:|\vec{\ell}|=s} \prod_{k=1}^N \pi_{k\ell_k} A_k(z; \ell_k)}{\sum_{\vec{\ell}:|\vec{\ell}|=s} \prod_{k=1}^N \pi_{k\ell_k}}$$

Proof.

$$\begin{aligned}
G_n(z; s) &= \mathbb{E}\,[\,\mathbb{E}\,[z^{|\vec{Y}_n|} \mid Y_{n-1}, X_{n-1} = s] \mid X_{n-1} = s] \\
&= \mathbb{E}\Big[\prod_{k=1}^N \mathbb{E}\,[z^{Y_{nk}} \mid Y_{n-1}, X_{n-1} = s] \mid X_{n-1} = s\Big] \\
&= \mathbb{E}\Big[\prod_{k=1}^N \mathbb{E}\,[z^{Y_{nk}} \mid Y_{n-1,k}] \mid X_{n-1} = s\Big] \\
&= \mathbb{E}\Big[\prod_{k=1}^N A_k(z; Y_{n-1,k}) \mid X_{n-1} = s\Big] \\
&= \sum_{\vec{\ell}:|\vec{\ell}|=s} \mathbb{P}(\vec{Y}_{n-1} = \vec{\ell} \mid |\vec{\ell}| = s) \prod_{k=1}^N A_k(z; \ell_k)
\end{aligned}$$

using the agents' independence and the Markov property. The result now follows as $n \to \infty$ $\diamondsuit$

When all the agents are identically specified, proposition 1 simplifies greatly as follows.

Corollary 1. *If all the agents are identical with local state $u = -, 0, +$ row-transition pgf $A(z; u)$ and equilibrium probability vector $\boldsymbol{\pi}$, for $s \ge 0$,*

$$G(z; s) = \frac{\sum_{n=0}^{\lfloor (N-s)/2 \rfloor} \frac{N!}{n!(N-s-2n)!(n+s)!} B_-(z)^n B_0(z)^{N-s-2n} B_+(z)^{n+s}}{\sum_{n=0}^{\lfloor (N-s)/2 \rfloor} \frac{N!}{n!(N-s-2n)!(n+s)!} \pi_-^n \pi_0^{N-s-2n} \pi_+^{n+s}}$$

where $B_u(z) = \pi_u A(z; u)$ for $u = -, 0, +$. For $s < 0$,

$$G(z; s) = \frac{\sum_{n=0}^{\lfloor (N+s)/2 \rfloor} \frac{N!}{n!(N+s-2n)!(n-s)!} B_+(z)^n B_0(z)^{N+s-2n} B_-(z)^{n-s}}{\sum_{n=0}^{\lfloor (N+s)/2 \rfloor} \frac{N!}{n!(N+s-2n)!(n-s)!} \pi_+^n \pi_0^{N+s-2n} \pi_-^{n-s}}$$

Proof.

Each term in the product in the numerator of the proposition is $\pi_{\ell_k} A(z; \ell_k)$, where ℓ_k takes one of the three values $-, 0, +$. Let there be n_-, n_0, n_+ occurrences respectively, where $n_- + n_0 + n_+ = N$. Moreover, to have global state s, we must have $n_+ - n_- = s$. The sum then simplifies to:

$$\sum_{\substack{(n_-, n_0, n_+):\\ n_+ - n_- = s\\ n_- + n_0 + n_+ = N}} \frac{N!}{n_-!n_0!n_+!} B_-(z)^{n_-} B_0(z)^{n_0} B_+(z)^{n_+}$$

For $s \geq 0, n_+$ must be at least s and $n_- = n_+ - s$ so that $N = n_0 + 2n_+ - s$. Since $n_0 \geq 0,\ 2n_+ \leq N + s$ and so the range of n_+ is $[s, \lfloor (N+s)/2 \rfloor$ and for each n_+, the values of n_0 and n_- are fixed at $n_{=}n_+ - s$ and $n_0 = N - 2n_+ + s$. The result now follows by changing the summation variable n_+ to $n = n_+ - s$. For $s < 0$, we must have $n_- \geq -s$ and the analogous result follows by interchanging the roles of n_+ and n_-. $\diamondsuit$ As already noted, the elements of M are defined by the coefficients of $G(z; s)$.

When all the agents are identical, the generating functions $G(z; s)$ are quick to compute by Corollary 1. At the other extreme, if the agents are all different (completely heterogeneous case), then Proposition 1 must be used and requires, for each of the $2N + 1$ values of its second argument, sums over state spaces of 3^N elements. This is completely impractical for even moderate N. However, typically we will have neither of these extremes but more likely a partition of a small number of sets of identical agents.

Let the equilibrium probability vector for the $2n_i + 1$ aggregate (sub)states of the ith agent-type be denoted $\boldsymbol{\phi}_i$, defined by $\phi_{iv} = \sum_{\vec{\ell}:|\vec{\ell}|=v} \prod_{j=1}^{n_i} \pi_{k_j \ell_{k_j}}$ for $-n_i \leq v \leq n_i$, where the sequence numbers of the agents of type i are here denoted $k_1, \ldots, k_{n_i}$. Again, this follows because the agents are independent. Now let the transition probabilities out of aggregate state v in type i have pgf $C_i(z, v)$, computed using Corollary 1 with $s = v$, applied to states numbered $k_1, \ldots, k_{n_i}$ instead of $1, \ldots, n_i$. Then we have the following result.

PROPOSITION 2. *For a collection of agents partitioned into r types, as defined above*

$$G(z; s) = \frac{\sum_{\vec{\ell}:|\vec{\ell}|=s} \prod_{k=1}^{r} \phi_{k\ell_k} C_k(z; \ell_k)}{\sum_{\vec{\ell}:|\vec{\ell}|=s} \prod_{k=1}^{r} \phi_{k\ell_k}}$$

where ℓ_k ranges over $[-n_k, n_k]$ for $1 \leq k \leq r$ (as opposed to [-1,1] for $1 \leq k \leq N$ in Proposition 1).

Proof.

The proof follows that of Proposition 1, but with the probabilities ϕ replacing π, the products being taken over agent types instead of individual agents, and the sums being over vectors of aggregate type-states instead of individual agent-states $\diamondsuit$

This is the proposition that we used in our experiments, described in the next section, where we considered a market with a small number of *types* of trader – just two in fact – but large numbers of each type. In the next subsection we give a simple illustrative example with two agents in all, of different types, so that propositions 1 and 2 become equivalent.

2.2 An example

As a small example of a market model such as the above, we consider two participating market agents with matrices T_1 and T_2

$$T_1 = \begin{pmatrix} 0.3 & 0.4 & 0.3 \\ 0.3 & 0.4 & 0.3 \\ 0.3 & 0.4 & 0.3 \end{pmatrix}, T_2 = \begin{pmatrix} 0.4 & 0.3 & 0.3 \\ 0.4 & 0.2 & 0.4 \\ 0.3 & 0.4 & 0.3 \end{pmatrix} \quad (2)$$

where the first row contains the probabilities of going from state $\{-\}$ to, respectively and in order from left to right, states $\{-, 0, +\}$. The second and third rows specify the corresponding transition probabilities from states $\{0\}$ and $\{+\}$ respectively. The first is considered 'neutral',and the second agent could be termed a 'fearful seller'.

Using these local agent transition matrices in the equations of the previous section, we calculate the transition probability matrix of the global market as:

$$M = \begin{pmatrix} 0.120000 & 0.250000 & 0.330000 & 0.210000 & 0.090000 \\ 0.120000 & 0.238533 & 0.326178 & 0.213822 & 0.101467 \\ 0.111000 & 0.236000 & 0.329333 & 0.222667 & 0.101000 \\ 0.102222 & 0.231852 & 0.331852 & 0.231852 & 0.102222 \\ 0.090000 & 0.240000 & 0.340000 & 0.240000 & 0.090000 \end{pmatrix} \quad (3)$$

3. SIMULATION MODEL

We have built a simulation program that models our peer-to-peer market for Grid Computing. In this program, we make use of a Barabási-Albert (BA) graph [1] to represent the peer-to-peer network, which is formed by a number of different nodes or market agents. BA graphs are an example of small-world scale-free networks often found in social networks [16]. The behaviour of each agent is determined by a particular Markov chain on the decisions of sell, hold or buy: $\{-, 0, +\}$. This is the only difference to our previous simulation work [9, 10].

The total average of the local market states of all pubs at every epoch gives the global market state of the network.

The objective of the simulation is to verify that the simplification of the peer-to-peer market network as a single central market place is appropriate. To this aim, we define a number of market participants with their respective Markov chains. On the one hand, we calculate analytically the equilibrium probabilities of the global market formed by these agents as described in previous sections. On the other hand, we implement these agents into the nodes of a peer-to-peer network, which we use in our simulation program, and find their results.

3.1 Simulation results

The mathematical analysis presented in section 2 derives the transition probability matrix of the market as a whole considering the individual transition matrices of all agents. Despite being this global transition probability matrix the essence of the market evolution, in this section we simplify the comparison between the analytic model and the simulation results to the global steady state probabilities of the market, i.e., the probability density function (pdf). We make the comparison for two different kinds of networks: a fully connected one (ideal simulation setup), and a random network (BA), which might not be fully connected (non-ideal simulation setup).

For a fully connected network, where the messages of all agents are able to get together once per epoch, i.e., in a simulation that follows precisely the analytic model, the simulation results prove to be exactly the same as in the analytic model. These results correspond to a market with 128 agents, all having a transition probability matrix $T_n = T_1$. We also have run simulations with the second, non-ideal setup, where the network might not be completely connected, or the agents might receive several copies of the same message. For all these non-ideal setups, we have used BA networks of 128, 512 and 1024 nodes, generated with the software package *igraph* [6]. In this section we explain the results that correspond to the random network of 128 nodes. The summary of the results for the other two graphs are specified by Table 1.

In the 128-node network, each node has a pub or buffer with a capacity for 128 messages, and the TTL of the messages is 7. We have run the simulations for 200,000 epochs, and we have analysed three different non-ideal setups by changing the agent types in the network.

The first non-ideal setup is formed by a network in which the 128 nodes have the same transition probability matrix: T_n. The pdfs of the global market state of the analytic and the simulated results are identical within the margins of errors of the simulation. Both are approximated well by a normal distribution. For the not fully connected graph the result has to be rescaled to fit the distribution of the analytic result to compensate for the overhead in the communication. The normal shape of the pdfs is explained by a Central Limit Theorem reasoning. This justification is stronger when the number of market participants increases. The average market state in both the simulation and in the analytic model is 0, as expected from the global neutral effect in a market formed by agents with matrices given by T_n.

The second non-ideal simulation setup is formed by a network of all 128 nodes having the transition probability matrix:

$$T_s = \begin{pmatrix} 0.6 & 0.2 & 0.2 \\ 0.4 & 0.2 & 0.4 \\ 0.2 & 0.6 & 0.2 \end{pmatrix}, \tag{4}$$

with a tendency to the sell action.

In this second setup, both the analytic and simulated results present again a normal shape, but with negative mean. However, their means are slightly different, being the analytic one less negative than the simulated. This result is explained by the fact that, when there is a non-zero mean for the global market state, the simulated market state in a non-fully connected network tends to increase in absolute value. This is because nodes in the network might receive replicated copies of the same message.

The third non-ideal simulation setup is formed by a network with 64 agents that have the transition probability matrix T_n and 64 agents whose transition probability matrix is the one in 4. This setup causes an average market state between 0 and the negative mean value in the second setup. In this final comparison, there is a small discrepancy between the mean in the simulations and the mean provided by the analytic model, being again due to the non-fully connected network. Since the average market state is less negative than in the second non-ideal simulation setup, this discrepancy is smaller. We observe again normal shapes in the distributions of both the analytic and simulated results.

Table 1 provides the summary of the results of the scale factors S for the three non-ideal simulation setups and the three different-sized random networks of size N and an average number of hops h. We also show the mean μ_s obtained in the simulation and the mean μ_a of the analytic model and he variance σ^2 of both the analytic model and the scaled simulation results. As expected, the scale factor increases with the average number of hops between nodes in the network, and the mean in the simulations gets larger in absolute value compared with the analytic mean when the average market state is non-zero. The variance of the results increases with the number of market participants, as predicted by the Central Limit Theorem. As a whole, the

N	h	S	μ_s	μ_s	σ^2
First non-ideal simulation setup					
128	4.4	1.49	0	0	76.8
512	6.6	2.65	0.04	0	307.2
1024	6.9	2.92	0.10	0	614.4
Second non-ideal simulation setup					
128	4.4	1.44	-29.56	-22.26	85.17
512	6.6	2.53	-110.49	-89.04	340.69
1024	6.9	2.75	-213.69	-178.08	681.38
Third non-ideal simulation setup					
128	4.4	1.48	-14.60	-11.12	80.99
512	6.6	2.60	-47.66	-44.52	323.94
1024	6.9	2.86	-96.69	-89.04	647.89

Table 1: Summary of simulation results.

ideal simulation setup has a perfect coincidence with the analytic model, whereas the non-ideal simulation setup has a very similar behaviour as well in so far as the distributions are both normal. Let us emphasize that in this section we have simplified the comparison between the analytic model and the simulation results to the probability density function of the market. This pdf is shown to be normal, as it can be deduced by a Central Limit Theorem reasoning. However, the analytic analysis presented in section 2 is still necessary in order to obtain the global transition probability matrix of the whole market, which is used in the next sections of this paper to analyse the market behaviour and find optimal trading strategies.

4. FUTURES TRADING OF COMPUTING POWER

The inability to store a CPU cycle makes its trading physically impossible, and brings immediate similarities with electricity markets, where derivatives are traded. We expect that a global futures market of computing power will emerge, and this is analysed in the current section. Specifically, we analyse the performance of a futures trader that operates in such a market, and adapts his decisions according to the optimization of a certain objective, with the expectancy of finding a pattern in the behaviour of the market. We specify this problem as a Markov Decision Process (MDP).

In our model, the set of decision epochs is discrete and infinite: $I \equiv \mathbb{N}\backslash\{0\}$. The state of the MDP is formed by the state of the market on the one hand, and by the state of the trader on the other hand.

We assume that the decisions of a single individual can not affect the evolution of prices. In particular, we consider that the price evolution of the market is given by a certain transition probability matrix, which has been generated as described in section 2. In addition to the price evolution, we include a second variable to model the state of the market: the trading volume, which gives information about the number of transactions.

Therefore the evolution of the market is given by a transition probability matrix like the one in expression 1. The state (an integer between $-N$ and N) indicates the variation in price with respect to the previous deal price. The trading volume is given by the absolute value of the price variation, i.e., the volume will always be a natural number between 0 and N. This volume can be understood as the number of units of future contracts available to be bought or sold at the given price.

These two variables, price variation (i) and trading volume ($|i|$), define the state of the market: $M_i = (i, |i|), \quad \text{for} \quad i \in \mathbb{Z}, -N \leq i \leq N$ which in turn determines the first two variables that define the state of the MDP.

The third and final variable that defines the state of the MDP is the position of the futures trader. The trader can buy, sell or hold at every time step. He can only buy or sell 1 futures contract at every time step, but over time he can accumulate up to N contracts. His actions therefore imply an integer number between $-N$ and N, which is his *open position*, i.e., the number of futures contracts he has (bought or sold) and are pending liquidation. The open position of the trader is $T_{pos} = pos, \quad \text{for} \quad pos \in \mathbb{Z} \cap [-N, N]$.

Consequently, taking into account both the market state and the trader's position, the state space of the MDP, S, is formed by $S_{i,pos} = (i, |i|, pos), \quad \text{for} \quad i, pos \in \mathbb{Z} \cap [-N, N]$ where i is the market price variation given by 1, $|i|$ is the trading volume and pos is the open position of the trader. Since the price variation can have $2N+1$ values, the trading volume is directly determined from the price variation and the trader can be in $2N+1$ different positions, the total number of states of the MDP is $(2N+1)^2$.

On the other hand, the actions the trader can take are: $Ac_s = \{-1, 0, 1\}, \quad for \quad s \in S$ with the condition of having an absolute open position that is limited to be between $-N$ and N. Therefore the available actions for the trader will depend on his current position, being limited to $\{0, 1\}$ when his open position is $-N$ and to $\{-1, 0\}$ when his open position is N.

To complete the definition of the MDP, a reward for the trader is established, which is conditioned by his actions. The reward that the trader is given is divided in two parts. The first part comes from the profit/loss in the operation he is immersed in due to the market price variation and his current open position. This reward is given as the multiplication of the trader's open position at the next decision epoch (which will depend on his current open position and his action taken at the current decision epoch) by the market price variation at the next decision epoch, which is uncertain and depends on the transition probabilities in expression 1. Consequently we will have to calculate its expected value in the current state of the system. Taking this into account, this first reward given to the trader when the system is in state s and he chooses to do action $a \in Ac_s$, is:

$$r_1(s,a) = \sum_{j \in S} r_1(s,a,j) p(j|s,a), \tag{5}$$

where $r_1(s,a,j)$ is the reward given to the trader when the system is in state s, the trader chooses to do action $a \in Ac_s$ and the system evolves to state j at the next decision epoch. Numerically, this reward is evaluated as $r_1(s,a,j) = i_j * pos_j$ being i_j and pos_j the market price variation and the position of the trader respectively when the system is in state j. With regard to $p(j|s,a)$, this is the probability of the system going from state s to state j when the trader chooses to do action $a \in Ac_s$. Since the open position of the trader at the next decision epoch is immediately calculated at the current decision epoch from his current open position plus his action $a \in Ac_s$, this conditional probability is directly given by expression 1, i.e., by the transition probability matrix of the market.

The second part of the reward is determined by the availability to buy or sell the remaining future contracts that the trader still has, which depends on the available trading volume. This second reward is actually a penalty, and therefore it will be zero in the best case and negative in the other cases. Since the available trading volume at the next decision epoch is unknown at the present decision epoch, this reward also depends on the next state of the system (and not only on the current state of the system and the trader's action), and therefore its expected value is given by:

$$r_2(s,a) = \sum_{j \in S} r_2(s,a,j) p(j|s,a), \tag{6}$$

with the same interpretation as expression 5, except for the fact that in this case $r_2(s,a,j) = -c * max(|pos_j| - |i|_j, 0)$ where $c \in \mathbb{R}^+$ is a penalty factor, and $|pos_j|$ and $|i|_j$ are, respectively, the absolute open position of the trader and the available trading volume when the system is in state j.

The total reward the trader is given when the system is in state s and he chooses to do action $a \in Ac_s$, is:

$$r(s,a) = r_1(s,a) + r_2(s,a) \tag{7}$$

4.1 An optimal trading policy

We now approach the problem of finding an optimal trading strategy for a futures trader that operates in a market as defined in section 4. In practice, this consists of finding an optimal policy for the MDP.

We specify our problem to be an infinite-horizon Markov decision process, and we apply the expected total discounted reward optimality criterion [12]. This means that the MDP continues in time until infinity, although we apply a discount factor λ, with $0 \leq \lambda < 1$, which makes future rewards less valuable.

With this setting, the expected total present value of the income stream obtained by using a policy π, when the system is in state s at the first decision epoch, is:

$$v_\lambda^\pi(s) = E_s^\pi \{ \sum_{t=1}^{\infty} \lambda^{t-1} r(X_t, Y_t) \}, \tag{8}$$

where $r(X_t, Y_t)$ is the reward received when using action Y_t in state X_t, and t is the decision epoch ($t \in I$). Using a discount factor, together with finite rewards, ensures the convergence of the series [12]. The objective is to find the policy π that maximizes expression 8.

We use linear programming to find an optimal policy. A detailed explanation on how to transform a discounted Markov decision problem into a linear programming problem can be found in [12]. Specifically, choosing $\alpha(j), j \in S$ (being S the state space of the MDP) to be positive scalars with $\sum_{j \in S} \alpha(j) = 1$, the primal linear program consists of minimizing: $\sum_{j \in S} \alpha(j) v(j)$ subject to: $v(s) - \sum_{j \in S} \lambda p(j|s,a) v(j) \geq$

$r(s,a)$ for $a \in Ac_s$ and $s \in S$, and $v(s)$ unconstrained for all $s \in S$. The dual linear program consists of maximizing $\sum_{s \in S} \sum_{a \in Ac_s} r(s,a)x(s,a)$ subject to $\sum_{a \in Ac_j} x(j,a) - \sum_{s \in S} \sum_{a \in Ac_s} \lambda p(j|s,a)x(s,a) = \alpha(j)$ and $x(s,a) \geq 0$ for $a \in Ac_s$ and $s \in S$. By solving the dual formulation of the problem, we find the values for the $x(s,a)$. We then obtain a decision rule for each state by choosing the action that gives the highest probability as given by $P\{d_x(s) = a\} = \frac{x(s,a)}{\sum_{a' \in Ac_s} x(s,a')}$ The set of the decision rules for each of the states of the MDP forms the policy.

4.1.1 Example

In this section we present an example of an MDP with $N = 2$ agents and a transition probability given by expression 3. The market can be in one of 5 states. The trader can be in one of his five positions and therefore the MDP has 25 states, which form the state space S, defined as the set of states: $S_{i,pos} = (i, |i|, pos)$, for $i, pos \in \mathbb{Z} \cap [-2, 2]$ being i the value of the price variation, $|i|$ the trading volume and pos the position of the trader. The possible actions of the trader are as before $\{-1, 0, 1\}$ except for the states in which the trader has a position of -2, where his possible actions will only be $\{0, 1\}$, and for the states in which the trader has a position of 2, where his possible actions will only be $\{-1, 0\}$.

We apply rewards as indicated in expressions 5, 6 and 7, with a penalty factor of $c = 0.1$, a discount factor $\lambda = 0.95$ and the same value for all the $\alpha(j)$. The dual is solved with the GNU Linear Programming Kit. The solutions to the 65 variables of the dual problem are obtained by software. Then, by applying the last expression from the previous section, we obtain a decision rule for each state, and therefore a policy, which in this case is optimal. The values are $S_{0,-1} = S_{0,0} = S_{0,1} = S_{0,2} = S_{1,-1} = S_{1,0} = S_{1,1} = S_{1,2} = S_{2,-1} = S_{2,0} = S_{2,1} = S_{2,2} = S_{-2,-1} = S_{-2,2} = S_{-1,-1} = S_{-1,1} = S_{-1,2} = -1$ and $S_{1,-2} = S_{2,-2} = S_{-2,1} = S_{-1,-2} = 0$ and finally $S_{0,-2} = S_{-2,-2} = S_{-2,0} = S_{-1,0} = 1$. The action -1 is predominant among the trader's decisions, which makes sense in a bear market as defined by expression 3.

5. CONCLUSION

This paper has approached the modelling of a future global peer-to-peer market for Grid Computing by the use of Markov chains. A first mathematical model has been introduced, in which each of the market agents is modelled by a Markov chain that reflects its behaviour. We have also presented the concept of *market pressure*, which has allowed us to obtain a global Markov chain that models the market as a whole by combining the individual contributions of all market participants, and avoiding the problem of state explosion. The simulation of such a system has proved to be accurate.

The impossibility of storing CPU cycles makes a futures market for computing power a natural development, which has been analysed, and an optimal trading policy has been derived using MDPs.

Further work will consist of the transformation of the agents' behaviours into utility functions, which will then be used to price options. On the other hand, an alternative modelling approach from the global perspective of the market will be presented with stochastic differential equations.

6. REFERENCES

[1] A.-L.Barabasi and R.Albert. Emergence of scaling in random networks. *Science*, 286:173, 1999.

[2] O. Ardaiz, P. Artigas, T. Eymann, F. Freitag, L. Navarro, and M. Reinicke. The catallaxy approach for decentralized economic-based allocation in grid resource and service markets. *Applied Intelligence*, 25(2):131–145, 2006.

[3] R. Buyya, D. Abramson, and S. Venugopal. The grid economy. *Proceedings of the IEEE*, 93(3):698–714, Mar. 2005.

[4] J. Cohen, C. Richardson, U. Harder, F. M. Ortuño, and J. Darlington. Node-level Architecture Design and Simulation of the MAGOG Grid Middleware. In *AusGrid 2009*, volume 99, pages 57–66, January 2009.

[5] I. W. Cotton. Microeconomics and the Market for Computer Services. *Computing Surveys*, 7(2):95–111, 1975.

[6] G. Csàrdi and T. Nepusz. The igraph software package for complex network research. *InterJournal Complex Systems*, page 1695, 2006.

[7] I. Foster and C. Kesselman. Computational Grids. *Chapter 2 of "The Grid: Blueprint for a New Computing Infrastructure", Morgan-Kaufman*, 1999.

[8] M. Greenberger. The priority problem and computer time sharing. *Management Science*, 12(11):888–906, 1966.

[9] U. Harder and F. M. Ortuño. Simulation of a peer to peer market for Grid Computing. In K. Al-Begain, A. Heindl, and M. Telek, editors, *ASMTA 2008*, volume 5055 of *LNCS*, pages 234–248. Springer, Berlin/Heidelberg, 2008.

[10] U. Harder and F. M. Ortuño. A more realistic Peer-to-Peer Grid Market Model. In J. T. Bradley, editor, *EPEW 2009*, volume 5652 of *LNCS*, pages 149–154. Springer, Berlin/Heidelberg, 2009.

[11] K. Lai, L. Rasmusson, E. Adar, L. Zhang, and B. A. Huberman. Tycoon: An implementation of a distributed, market-based resource allocation system. *Multiagent Grid Syst.*, 1(3):169–182, 2005.

[12] M. L. Puterman. Markov Decision Processes. Discrete Stochastic Dynamic Programming. In *Markov Decision Processes. Discrete Stochastic Dynamic Programming*. Wiley Inter Science, 1994.

[13] O. Regev and N. Nisan. The POPCORN Market an Online Market for Computational Resources. *Proceedings of the first international conference on Information and computation economies*, pages 148 – 157, 1998.

[14] I. E. Sutherland. A futures market in computer time. *Commun. ACM*, 11(6):449–451, 1968.

[15] C. A. Waldspurger, T. Hogg, B. A. Huberman, J. O. Kephart, and W. S. Stornetta. Spawn: A Distributed Computational Economy. *Software Engineering*, 18(2):103–117, 1992.

[16] F. Wang, Y. Moreno, and Y. Sun. Structure of peer-to-peer social networks. *Physical Review E*, 73:036123, 2006.

[17] R. Wolski, J. S. Plank, J. Brevik, and T. Bryan. Analyzing Market-based Resource Allocation Strategies for the Computational Grid. *The Int. J. of HPC Applications*, 15(3):258–281, 2001.