Computer Engineering and Applications
Computer Engineering and Applications Vol. 4, No. 2, June 2015

Parallel Continuous Double Auction for Service Allocation in Cloud Computing

Hossein Ebrahimpour-Komleh[1], Nima Farajian[2]

[1]*Department of Computer Engineering, Faculty of Computer engineering, Kashan University*
[2]*Department of Computer Engineering, Faculty of engineering,*
Eyvanekey Institue of Higher Education
ebrahimpour@gmail.com, nimaff2000@yahoo.com

ABSTRACT

Cloud Computing is a service oriented architecture in which every computing resources is delivered to users as a service. Nowadays market-oriented approach has attracted a lot of researchers because of its great ability to manage Cloud services efficiently and dynamically. Each service consists of various resources which all should be allocated to utilize the service. In this paper a parallel continuous double auction method for efficient service allocation in cloud computing is presented in which by using a novel parallel sorting algorithm at auctioneer, enables consumers to order various resources as workflow for utilizing requested services efficiently. Also in the presented method consumers and providers make bid and offer prices based on time factor. Experimental results show that proposed method is efficient in success rate, resource utilization and average connection time and also overall performance of system is improved by parallel approach.

Keywords: Three To Six Keywords, Times New Roman Font, Point Size 12, Fully Justified, Heading In Bold.

1. INTRODUCTION

Cloud computing offers a new computing model where resources such as computing power, storage and online applications can be shared as 'services' over the internet [1] and is growing increasingly popular and appears well-suited to meet the demand of resource sharing. Cloud computing consists of several interconnected and virtualized computer which can be served as a one or more unified computing resource based on service-level agreements SLAs(between providers and consumers [2]). In cloud computing users access the required services on demand and pay cost of them through pay-for-use model. Nowadays much of the research on cloud computing area is about virtualization and architecture infrastructure but resource management is one of the most important problems in Cloud Computing. Resource management in Cloud consists of finding required resource for each service and allocates them through interaction and SLA agreement between cloud participants.

It was noted [2] that a market-oriented approach for managing Cloud resources is necessary for regulating their supply and demand through flexible and dynamic pricing. Each consumer is looking for acquiring required resource with minimum price and each provider wants to sell his resource at maximum price. So a framework is needed in which all consumers and providers can use and rent their resources through interaction, massage passing and agreement with each other. But this can't be done simply because of (i) access to different multiple resources can be done simultaneously which cause competition between consumers (ii) the environment is highly dynamic and resources could be added or removed frequently (iii) Resource allocation decisions must be made quickly because of consumer deadline[4] .

In recent years, usage of market oriented methods for grid and cloud resource management is one of the solutions which has received much attention. It enables the regulation of supply and demand for resources; provides an incentive for resource owners to participate in Cloud; and motivates the users to trade-off between deadline, budget, and the required level of quality of service [3]. Two categories of market based models that are used for cloud resource management are commodities market models and auction models. In the commodity market model, providers specify their resource price and charge users according to the amount of resource they consume. In the auction model, each provider and consumer acts independently and they agree privately on the selling price.Auctions are used for products that have no standard values and the prices are affected by supply and demand at a specific time [3]. Generally there are 4 kind of auction: English auction, Dutch auction, First and Second Price Auction and Double auction. The double auction model has a high potential for cloud computing. In a double auction model, consumers submit bids and providers submit requests at any time during the trading period. If at any time there are bids and requests that match or are compatible with a price, then a trade is executed immediately [13].

Three most popular double auctions are: Preston-McAfee Double Auction Protocol (PMDA) [5], Threshold Price Double Auction Protocol (TPDA) [6], and Continuous Double Auction Protocol (CDA) Kant and Grosu [7] showed that the CDA protocol is better from both the resource's and the user's perspective providing high resource utilization in grid environments.

In this paper, an auction model for service allocation in cloud computing is presented which consider time, competition, opportunity and eagerness factor. The rest of the paper is organized as fallow. Section 2 describes some relevant works. The design details of this model and its mechanism are presented in Section 3. The evaluation and experimental results are shown in Section 4 and conclusion is provided in Section 5.

2. RELATED WORKS

Most of the studies and research on management of resources and services in cloud computing is derived from method and technique which has been used in grid computing. The reason for this is high similarity between cloud and grid computing and also earlier emergence of grid computing.

Buyya et al. [3] used economic based concepts including commodity market, posted price modeling, contract net models, bargaining modeling, etc. for grid resource allocation . Schnizler et al. [8] introduced the notion of using a double-sided combinational auction to allocate grid resources. Tan et al. [9] proposed a stable continues double auction, based on the more conventional CDA. It alleviates the unnecessarily volatile behavior of the CDA, while maintaining other beneficial features. Amar et al. [10] illustrated a comprehensive grid market model including a futures market and a centralized/decentralized spot market.

In [11] authors presented the auction allocation model and three auction-based protocols: First-Price Auction, Vickrey Auction, and Double Auction. They studied them in terms of economic efficiency and system performance. The results showed

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

that when they consider a mix of risk-averse and risk-neutral users First-Price Auction favors resources while Vickrey Auction favors users. On the other hand, the Double Auction favors both users and resources. Izakian et al. [12, 13] introduced a continuous double auction method in which resources are considered as provider agents and users as consumer agents. In each time step, each provider agent determines its requested value based on its workload and each consumer agent determines its bid value based on two constraints: the remaining time for bidding, and the remaining resources for bidding. In [14] authors present a reverse VCG auction for cloud computing in which consumers hold auctions for required resources and provider make bid and winner determination is based on lowest bid. Fujiwara et al. [15] Presented combinational auction model for cloud computing and design spot and future auction market for provider to put their resource in it. Importantly, this method use combinational auction which means auctioneer put multiple different resource type in auction and each consumer can bid for combination of them and winners determination is based on maximum profit. Adabi et al. [16] presented a new negotiation model for designing Market and Behavior-driven Negotiation Agents that addressed computational grid resource allocation problem. This work introduced rational negotiation protocol and negotiation policy that modeled the effective factors used by negotiators of real-life trading market for making concession amount in negotiation process. Teymouri [17] proposed a new UCDA method based on CDA method in which bids are updated by auctioneer itself. Also a method is presented for the providers to determine the resource price based on their workload and for users to determine their bids based on jobs deadlines.

Sim [18] presents a market-driven negotiation mechanism for Grid resource management which includes (i) a market-driven strategy and (ii) a relaxed-criteria negotiation protocol. Using a market-driven strategy, agents make adjustable amounts of concession by considering opportunity and time factors. Sim [19] presented a resource management model for cloud computing based on market-driven agent which introduced in [18]. In this model there are provider, consumer and broker agents that act as intermediaries between providers and consumers .Through a 4-stage resource discovery process (selection, evaluation, filtering, and recommendation), a set of broker agents match consumers' requests to advertisements from providers. Following the matching of requests to resources, consumer and provider agents negotiate for mutually acceptable resource time slots.

3. MARKET MECHANISM AND PROBLEM DEFINITION

3.1.CLOUD COMPUTING MARKET MODEL

We assume the cloud computing environment includes resource providers, consumers and an auction market. Consumers refer to cloud to use and consume its required services. Each service is composed of different resources which are provided by providers who sell them to make profit. Consumer should get all required resources for each service to utilize it.

For each entity in cloud market there is one agent. Consumer agent works on behalf of the consumer and purchase required resources of the requested service by interaction with auctioneers, provider agent which works on behalf of the provider and sells its resource to consumers through related auctioneer and auctioneer agent which receives bids and requests from all consumers and providers and executes trade for all matching offers. In this model for each resource type there is one auctioneer. In this paper we proposed a continuous double auction method for service allocation in which enables consumers to acquire all required resources of a service as a workflow to utilize requested service and also trading price is fairly relative to supply and demand condition of market. Fig 1 shows a cloud computing environment with the proposed mechanism.

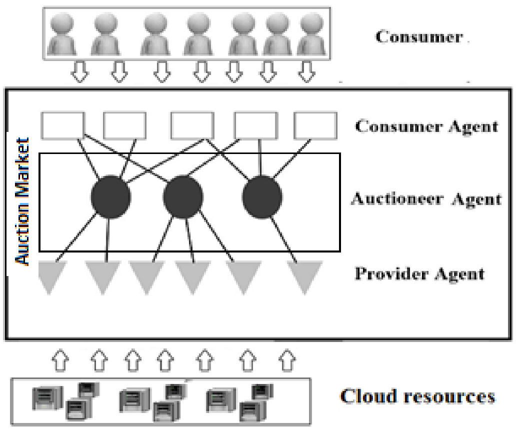

FIGURE 1. Resource allocation schema in proposed method

3.2 PROBLEM FORMULATION

Each resource R is characterized by five-tuple, Ri = (typ, spp, twp, rpp, mpp), in which typ is resource type(for example CPU, Hard Disk, Ram ...), spp is the resource specification (for CPU process speed , For RAM and Hard Disk capacity and ...), twp is current workload of resource and means that resource is busy until twp, rpp refer to lowest price for resource Ri(also called reserve price) and is expressed in form of Cloud units per second (C$ /S) . Also mpp is maximum price for resource Ri and is expressed in form of Cloud units per second (C$ /S). In this paper we assume that each resource is allocated by one consumer in each time unit.

Each service request Ci is composed of some resource requests which each of them is characterized by five-tuple, Ci,j=(typ,spc,ts,td,mps), typ and spc determine type and specification of requested resource ,ts is amount of time that consumer need to utilize the resource and td is deadline and consumer must utilize its requested resource before td. mps represents the budget allocated to Ci,j.

3.3 CONTINUOUS DOUBLE AUCTION METHOD

Each consumer is looking for utilizing its requested service with minimum price before its deadline and each provider wants to sell its resource at maximum price and also reduces idle time of its resource. To utilize a service all required resources

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

should be allocated before deadline and otherwise service failed to utilize and consumer must pay penalty to providers for all other resources which is allocated to it. In the continuous double auction method at each time unit, consumers and providers submit bids and requests to the related auctioneer. An auctioneer maintains a list of the current bids and requests and matches the two offers when the highest bid is higher than or equal to the lowest request the trade occurs at the average of matching request and bid prices.

Determining the bids and requests value of consumers and providers can be done autonomously by each agent and based on their objectives and market condition. Different parameters can be considered to calculate price value which in this paper we consider time, opportunity, competition and eagerness factor. We call this method parallel market-driven continuous double auction method (PCDA). This method is inspired by the work presented in [13].

3.3.1 DETERMINING BID VALUES FOR CONSUMER AGENTS

Consumer agent in each time unit determines its bid price based on deadlinetime, availability, competition and eagerness factor for each requested resource.

Time factor: Since consumers are generally sensitive to deadline in acquiring requested service, it is intuitive to consider time when formulating the bid price. Consumer agent time dependent bid price formula is determined in (1).

$$BidT(i,t) = \left[\frac{R_{min}}{mps} + \left(1 - \frac{R_{min}}{mps}\right) \left(1 - \frac{tre}{td}\right)^{\rho}\right] \quad (1)$$

Rmin is minimum reserve price between all providers and tre is time interval between t and td. $0<\rho<\infty$ is the concession making strategy. Three classes of strategies are specified as follows: Conservative ($\rho >1$) which the consumer maintains a low bid value until current time gets close to td, Linear ($\rho =1$), and Conciliatory ($0< \rho <1$) which the consumer starts with a bid value close to mps.

3.3.2 DETERMINING THE REQUEST VALUE FOR PROVIDER AGENT

Each provider has two objectives: (i) make maximum profit for selling its resource (ii) reduce idle time of its resource. Provider agent in each time unit determines its request value based on time and opportunity factor.

Time factor: Provider agent is sensitive to idle time of its resource and considers workload time for calculate price as in (9).

$$RequestT(i,t) = \left[\frac{\mathrm{rpp}}{mpp} + \left(1 - \frac{\mathrm{rpp}}{mpp}\right) \left(\frac{tst}{twp}\right)^{\tau}\right] \quad (2)$$

tst is time interval between t and twp. Based on (2) each provider offers its reserve price when its resource is idle and after allocation makes its price to mpp and with elapsing time decreases its price. Also τ is the concession making strategy like ρ and σ.

Opportunity factor: This factor expressed number of buyer for provider's resource. If number of buyer decrease (increase) it means that market situation is unfavorable (favorable) for provider and it should reduce (increase) its price. According to number of buyer, provider calculates price formula in (3).

$$RequestC(i,t) = \left[\frac{\mathrm{rpp}}{mpp} + \left(1 - \frac{\mathrm{rpp}}{mpp}\right)\left(1 - \frac{C_i^t - 1}{C_{avg}^t}\right)^{\pi}\right] \quad (3)$$

π is the concession making strategy similar to ρ,σ ,τ . C_i^t is available buyer in time t and C_avg^t is average number of buyer in m time unit ago and is determined using (4).

$$C_{avg} = \frac{\sum_{i=1}^{m} C_i^t}{m} \quad (4)$$

Provider agent finally makes its request price using (5).

$$Request\ (i,t) = \omega\ \times RequestT(i,t) + (1 -\ \omega) \times RequestR(i,t) \quad (5)$$

where $0\ \leq\ \omega\ \leq 1$

3.3.3 AUCTIONEER AGENT

In each time unit, consumer and provider agents calculate their bid and request values and submit them to the auctioneer. The auctioneer sorts the bid values in increasing order and request values in decreasing order. Trade occurs when 1) highest bid is more than or equal to the lowest request, 2) resource could be utilized by consumer as described in (4). When the above condition is satisfied, the trade occurs at the following price:

$$price = \frac{Highest\ Bid + Lowest\ Request}{2} \quad (6)$$

3.3.3.1 PARALLEL AUCTIONEER

In enterprise application where the number of cloud participants is really massive and also bids and offers rating is very huge, the system efficiency decreases and the auctioneer is unable to perform its tasks in a real time manner.

The main heavy task in the auctioneer system is sorting bids and offers. So in the purposed method we used a novel parallel method for sorting (PCM) which is optimal and has better performance than other usual parallel sorting algorithms. The PCM is based on a divide-and-conquer strategy and is composed of two stages. In a first stage, the data sequence to be sorted is partitioned into several pieces that are sorted concurrently (fully parallel). In a second stage, all sorted pieces are merged using a recursive procedure to finally obtain the full sorted data sequence. In each iteration of this procedure pairs of previously sorted pieces are selected, merged and

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

sorted in a concurrent way. This merging and sorting task of two sequence pieces is made by a new algorithm which is called PREZ [20].

So if the auctioneer system has multiprocessors, the overall performance of auctioneer is improved by using above technique.

4. EXPERIMENTAL RESULT

In order to study the efficiency of the method presented in this paper, we developed a computational grid simulator using the java agent development framework (JADE) [21]. JADE is a middleware aimed at developing multi agent systems and applications conforming to FIPA standards for intelligent agents. In our simulated system, consumers and providers are modeled as two kinds of agent and also for each resource type there is one auctioneer. Resource type and specification are shown in Table 1.

TABLE 1.

Resource Specification
CPU Speed (GHz) : 1.5 ~ 3.5
Memory(Mb):{512,1024,2048}
Disk Capacity(Mb) : 50 ~ 1000

In our experiments we use the system load concept i.e. the ratio of aggregated length and capacity of resource requests submitted to the cloud to the aggregated capacity that the cloud is capable of providing in the simulation period. The system load can be obtained using Eq. (7).

$$\rho_{Resource1} = \frac{\sum_{j=1}^{n}\left(spc_j \times ts_j\right)}{T_{total} \times \sum_{i=1}^{m} Spr_i} \tag{7}$$

In the simulated system there are 100 consumer agents and each consumer agent request 1 to 3 resources for its required service. Number of provider agent depends on system load and each provider agent provides one resource. For a fair comparison each provider set reserved price to 2$ and maximum price to 9$. The budget of each consumer is random integer between 3$ and 8$. Consumer agent utilization time of a resource (ts) is random integer number within range [2000, 5000] and deadline is from 2 to 5 times of ts and penalty fee for all resource is 5$. Also we set ω=0.5, φ =0.4 , μ=γ=0.3 and π=ρ=σ=τ=1.

To Evaluate efficiency of this method we consider success rate, average connection time, average resource utilization and average trading price (which is summarized in TABLE II and compared it with random continuous double auction (RCDA), continuous double auction method in [13] (CDA) and continuous double auction method in [17] (UCDA). In all simulation we call the presented method (PCDA).

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

TABLE 2.
Evaluation parameters

Average connection time	$T_{avg} = \frac{1}{n}\sum_{j=1}^{n}(T_j^m - T_j^r)$
T_j^m	The time at which a request is matched to a resource
T_j^r	The time at which a consumer submits a request
n	The total no. of matches of requests to resources
Success rate	$Succ = \frac{\sum R_{matched}}{R_{total}}$
$R_{matched}$	Number of requests that are matched to resource(s)
R_{total}	Total number of requests from consumers
Resource Utilization	$RU = \frac{T_{allocate}}{T_{total}}$
$T_{allocate}$	Total number of time slots allocated
T_{total}	Time window
Average Trading Price	$P_{avg} = \frac{1}{m}\sum_{j=1}^{m} P_j^{alloc}$
P_{trade}	Allocation price
m	The total no. of allocation
Average Trading Rate	$TRavg = \frac{TR_{Total}}{T_{total}}$
$T_{allocate}$	Total number of Trades
T_{total}	Time window

In the RCDA asks and bids are determined randomly between Rmin and mps for the users and the rpp and mpp for providers. Figure 2 (a-e) shows the result of simulation and comparison.

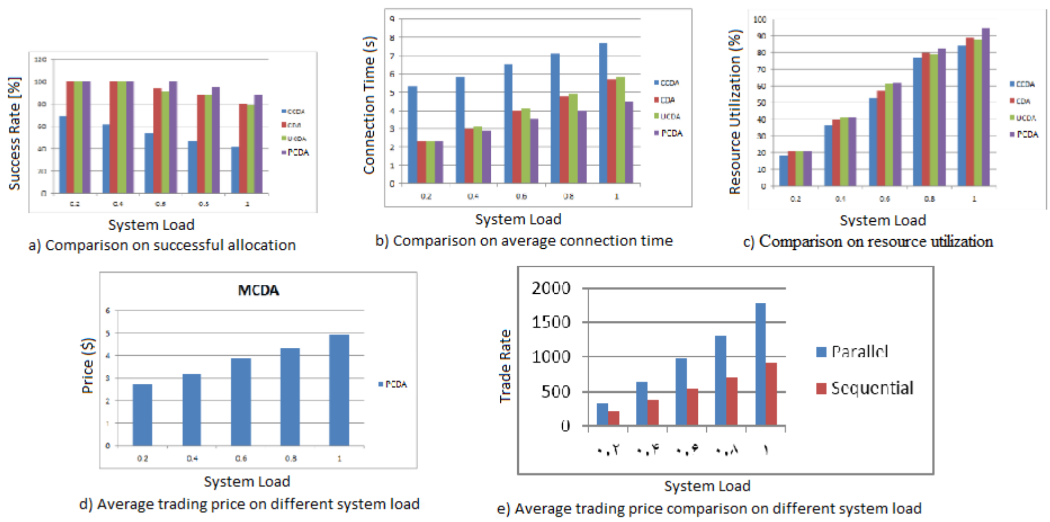

FIGURE 2. Comparison on successful allocation

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

Figure 2-(a),(b),(c) show successful rate, average connection time and resource utilization in different system load and as shown in them presented method is more efficient than other method especially in higher system load which is due to eagerness factor.

Figure 2-(d) shows average trading price and as shown in it with increasing (decreasing) in system load, market demand increase (decrease) so environment is more favourable (unfavourable) for providers and sell their resource with higher (lower) price.

In figure 2-(e) the effectiveness of parallel auctioneer is considered. In this experiment the Average Trading Rate is compared between the two system: system witch its auctioneer has one CPU core and system witch auctioneer has 8 CPU cores. The result indicates that the overall performance of auctioneer is improved by using multiprocessors and PCM algorithm.

5. CONCLUSION

In Cloud Computing resource allocation is based on interaction and agreement between cloud consumers and providers. According to highly dynamic environment and also different policies of each provider and consumer, we need a method in which each participant can make decision autonomously based on market condition and its interest. In this paper a parallel continuous double auction model is presented in which consumer and provider make bid and offer with time consideration. Experimental results illustrate that proposed method is efficient and intensive for both consumer and provider and also has acceptable result in multiprocessors environment.

REFERENCES

[1] Luis M. Vaquero, Luis Rodero-Merino, Juan Caceres, and Maik Lindner, "A Break in the Clouds: Towards a Cloud Definition", ACM SIGCOMM Computer Communication Review 50 Volume 39, Number 1, January 2009.

[2] R. Buyya et al. "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," Future Generation Computer Systems, vol. 25, no. 6, pp.599–616, June 2009.

[3] R. Buyya, D. Abramson, J. Giddy, H. Stockinger, Economic models for resource management and scheduling in Grid computing, The Journal of Concurrency and Computation 14 (13_15) (2002) 1507_1542.

[4] K. M. Sim. A Survey of Bargaining Models for Grid Resource Allocation. ACM SIGECOM: E-commerce Exchanges, Vol. 5, No. 5, January, 2006, pp. 22-32.

[5] R.P. McAfee, A dominant strategy double auction, Journal of Economic Theory 56 (1992) 434450.

[6] M. Yokoo, Y. Sakurai, S. Matsubara, Robust double auction protocol against false-name bids, in: Proc. of the 21st IEEE International Conference on Distributed Computing Systems, April 2001, pp. 137145.
[7] U. Kant, D. Grosu, Auction-based resource allocation protocols in Grids, in: 16th International Conference on Parallel and Distributed Computing and Systems, Nov. 2004, pp. 2027.
[8] B. Schnizler, D. Neumann, D. Veit, and D. Weinhardt, "Trading grid services - a multi-attribute combinatorial approach," EuropeanJournal of Operational Research, vol. 187, no. 3, 2008, pp. 943-961.
[9] Z. Tan and J. R. Gurd, "Market-based grid resource allocation using astable continuous double auction," Proc. 8th IEEE/ACM Int. Conf. on Grid Computing (Grid 2007) , 2007, pp. 283-290.
[10] L. Amar, J. Stosser, and E. Levy, "Harnessing migrations in a marketbased grid OS," Proc. 9th IEEE/ACM Int. Conf. on Grid Computing (Grid 2008) , 2008, pp. 85-94.
[11] Grosu, D., Das, A., "Auction-based Resource Allocation Protocols in Grids", 16th IASTED International Conference on Parallel and Distributed Computing and Systems, 2004.
[12] Izakian, H., Ladani, B.T., Zamanifar, K., Abraham, A., Snasel, V., "A Continuous Double Auction Method for Resource Allocation in Computational Grids", IEEE Symposium on Computational Intelligence in Scheduling, 2009, pp. 29-35, 2009.
[13] H. Izakian, A.Abraham, B. Tork Ladani . An auction method for resource allocation in computational grids . Future Generation Computer Systems vol 26, 2011 ,pp 228–235
[14] Xiaohong Wu at la.. Cloud computing resource allocation mechanism research based on reverse auction . . Energy Procedia vol 13, 2011 ,pp 736–741
[15] I. Fujiwara . Applying Double-sided Combinational Auctions to Resource Allocation in Cloud Computing . IEEE 10th Annual International Symposium on Applications and the Internet. 2010 .
[16] Adabi, S., Movaghar, A., Rahmani, A., M., Beigy, H., "Market-based Grid Resource Allocation Using New Negotiation Model", Journal of Network and Computer Applications, 2012.
[17] S. Teymouri, A. M. Rahmani. "A Continues Double Auction Method for Resource Allocation in Economic Grids", International Journal of Computer Application vol 43, 2012, pp 7-12.
[18] K. M. Sim. From Market-driven e-Negotiation Agents to Market-driven G-Negotiation Agents. Proc. of the IEEE Int. Conf. on e-Technology, e-Commerce and e-Services, pp. 408-413, Hong Kong, 2005.
[19] K. M. Sim. Agent-based Cloud Commerce. Proc. IEEE Int. Conf. on Industrial Eng. and Eng. Management, Hong Kong, 2009, pp. 717-721.
[20] E.Herruzo, G.Ruiz, A New Parallel Sorting Algorithm based on Odd-Even Mergesort, 15th EUROMICRO International Conference on Parallel, Distributed and Network-Based Processing, 2007.
[21] JADE, Java Agent Development Framework, 2004. http://jade.cselt.it.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)