50

A WEB INTERFACE FOR ACCESSING SCHEDULING METHODS IN A DISTRIBUTED KNOWLEDGE BASE

Maria Leonilde R. Varela[1], Joaquim Nunes Aparício[2], Sílvio Carmo Silva[3]

[1]*University of Minho, School of Engineering, Dept. of Production and Systems, PORTUGAL*
Email: leonilde@dps.uminho.pt

[2]*New University of Lisbon, Faculty of Science and Technology, Dept. of Computer Science, PORTUGAL*
Email: jna@di.fct.unl.pt

[3]*University of Minho, School of Engineering, Dept. of Production and Systems*
Email: scarmo@dps.uminho.pt
PORTUGAL

With the increase of the Internet and virtual communities, interfaces for web applications and automated services are becoming an emergent necessity.
In this paper we propose a web interface for enabling access to different manufacturing scheduling methods, which can be remotely available and accessible from a distributed knowledge base. Through this interface users can register and join together as a community and share their scheduling knowledge. Scheduling problems can occur in several different manufacturing environments and the web system searches in the distributed knowledge base for appropriate solving methods. Therefore, through this interface once suitable available methods, for a given problem, are identified, it enables running one or more of them, made accessible by the system. Moreover, new methods can be continuously incorporated in the system's distributed repository, in a user-friendly way, feeding the knowledge base.

1. INTRODUCTION

In today's knowledge-based economy, the competitiveness of enterprises and the quality of work life are directly tied to the ability of effective creation and share of knowledge both, within and across organizations.

Manufacturing scheduling is a complex task that involves a wide range of knowledge. Slight differences in the manufacturing environment originate distinct problems, which even though being closely related, require different solving methods.

The effective and efficient resolution of those problems begins with the identification of suitable scheduling methods for solving them. When there are alternative methods to solve a problem we can obtain alternative solutions, which should be evaluated against specified criteria or objectives to be reached. Thus, users are able to properly solve a problem, through the execution of one or more scheduling methods, local or remotely accessible through the Internet, and, subsequently, select de most suited solution obtained.

This work attempts to offer new possibilities to scheduling, giving a contribution in terms of manufacturing scheduling problems solving through a web-based decision support system. The system follows a peer-to-peer computing model, which permits sharing scheduling knowledge by means of a distributed knowledge base. This distributed scheduling repository enables accessing knowledge arising from an extended range of contributors and, therefore, a potentially more useful search, through a widened search space. This infrastructure is based on the principles of virtual organizations (VO) [1, 2].

The system permits the characterization of each problem to be solved and, then, the access to corresponding solving methods. For problems identification, a problem classification model that includes a set of parameters is used. This classification model enables specifying problem classes to which real problem instances belong [8, 9]. The data representation model for scheduling problems and related concepts is based on XML (extensible markup language), used as a specification language for scheduling data representation and processing on the Internet [8, 9].

This paper is organized as follows. The next section briefly describes the nature of scheduling problems and the underlying classification model. Section 3 presents the main web system functionalities for supporting the scheduling decision making process by any end user who wants to solve a certain problem. The interface for enabling an easy and user-friendly way of feeding the knowledge base is also shown. Moreover, an example is given to illustrate the use of the system interface to help users to solve a specific problem. The main interface characteristics are briefly described. Finally, in section 4, some conclusions are reached.

2. MANUFACTURING SCHEDULING

Manufacturing scheduling consists on a temporal assignment of tasks to production resources, where one or more goals and constraints have to be considered. It is one of the significant tasks to be performed to achieve competitive production, which usually means to deliver products in time or to use resources efficiently. “Good” orderings to perform a series of given tasks have to be found, whereby specific objectives shall be optimized.

In order to perform the scheduling process it becomes necessary to clearly specify the problem to be solved. Manufacturing scheduling problems have a set of characteristics that must be clearly and unequivocally defined.

Due to the existence of a great variety of scheduling problems there is a need for a formal and systematic manner of problem representation that can serve as a basis for their classification. A framework for achieving this was developed by Varela, et al. [8, 9], based on existing notations available in the literature. This framework allows identifying the characteristics of each problem to be solved, and it is used as a basis for an XML-based problem specification model developed [8, 9].

The referred framework for problem representation includes three classes of notation parameters for each corresponding class of problem characteristics, in the form of $\alpha|\beta|\gamma$. The first class of characteristics, the α class, is related to the environment where the production is carried out. It specifies the production system type (α_1) and, eventually, the number of machines that exist in the system (α_2). The second class allows specifying the interrelated characteristics and constraints of jobs and production resources, which are expressed by the β $(\beta_1 \ldots \beta_{12})$ parameters, and also the performance criterion, which is the third class (γ). Some important processing constraints are imposed by the need for auxiliary resources, like robots and transportation devices and/or the existence of buffers, among others factors. The evaluation criterion, the third class of parameters, may include any kind of performance measure, namely multi-criteria measures [8, 9].

An example of use of this notation is "F3|n|Cmax" which reads as: "Scheduling of non-preemptable and independent tasks of arbitrary processing time lengths, arriving to the system at time 0, on a pure flow shop, with 3 machines, to minimize the maximum completion time or makespan (Cmax).

Good schedules strongly contribute to the company success. This may mean meeting deadlines for the accepted orders, low flow times, few ongoing jobs in the system, low inventory levels, high resource utilization and, certainly, low production costs. All these objectives can be better satisfied through the execution of the most suitable scheduling methods available for solving each particular problem.

3. WEB APPLICATION

3.1 Main functionalities

As referred before, the main purpose of this work consists on providing new possibilities for manufacturing scheduling. Therefore, a contribution is given in terms of the scheduling problems solving process through a web-based decision support system. This system is based on XML modeling and related technologies.
The association of scheduling problems to solving methods is done using the information available in a distributed knowledge base (DKB). The system is able to quickly assign methods to problems that occur in real world manufacturing environments and solve them through the execution of one or more appropriate methods' implementations that are local or remotely accessible through the Internet.

This web system encompasses several main functionalities, which include knowledge insertion, about scheduling problems and resolution methods, and correspondent information searching. Users can make requests for visualizing scheduling problem classes and methods' information or even browse information about other concepts presented by the system. The data can be shown in different views, using existing XSL (extensible stylesheet language) documents, adequate for each specific visualization request. Another important functionality is the execution of scheduling methods, given the manufacturing scheduling problem definition. The selection of one or more specific implemented scheduling methods is made through a searching process on the DKB. The system also enables different ways of problem results presentation and storage.

3.2 Peer-to-peer distributed knowledge base

Emerging peer-to-peer technology and appropriate networks suite well to the increasingly decentralized nature of modern companies and their industrial and business processes, whether it is a single enterprise or a group of companies [6].

The P2P framework provides the capabilities that allow users, or peers, to directly interact with each other [6, 7]. The web application put forward within this work follows this P2P computing model. Therefore, it has the capability of allowing a direct-interaction between the peers, which turns the computing environment decentralized, namely in terms of storage, computations, messaging, security and distribution. One of the greatest benefits of this P2P network, in the context of this work, is to easily support the concept of community. Consequently, it is possible for users to organize themselves into groups that can collaborate with each other in order to achieve a certain goal. In this context, this goal is to improve the resolution of manufacturing scheduling problems, giving a better decision-making support by enabling access to several different scheduling approaches. This is achieved, by providing a mechanism that allows them to share their scheduling knowledge and solving methods. The developed system is based on the principles of virtual organizations [1,2].

Within this organization, each peer can then be seen as a partner interested on solving scheduling problems and, whereby, looking for feasible solutions in the network, which is composed by the local knowledge base (KB) of each participating member. Whenever a member stores knowledge in his local KB, he is automatically contributing to the enrichment of the whole distributed knowledge repository that is available to all members of the organization.

Some peers of the organization can also act as ultra-peers. These special peers have the additional functionality of owning the list of the peers that belong to the VO. Such list contains information about the VO members and a flag that indicates the current state of each peer, which can be active or non-active. An ultra-peer also serves the purpose of configuring the P2P network as an open system allowing any external user to join the organization, or as a closed system, in the sense that only the nodes belonging to a certain company or domain can join the organization.

Each active peer is continuously waiting for requests from other peers or from browser-like users. When a request reaches a peer, it firstly asks to one of the ultra-peers for the list of other active peers. Next, it propagates the request to all the peers of that list. Once the replies have been returned, a compilation of the obtained results is presented to the user (Figure 1).

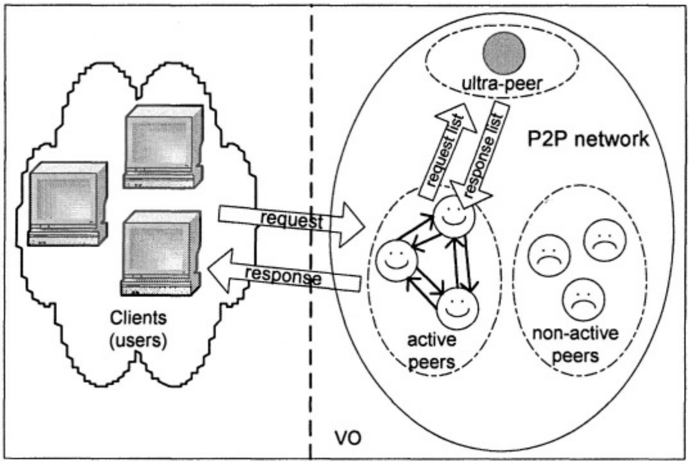


Figure 1 – Peer-to-peer framework

In order to fulfill those requests, each peer contains his own local knowledge base component, all together composing the DKB. Moreover, each of them has a main system component, which is an interface for introduction, validation, and transformation of manufacturing scheduling data. This interface is mainly controlled by DTD (Document Type Definition) and XSL documents stored in the knowledge base and all this information is stored in XML documents, which are validated according to associated DTDs, before being put in the corresponding knowledge base [8, 9].

At any time, external users can join the VO and configure themselves as active peers. This can be easily done, by just installing a set of common components that compose the interface for accessing the network and its DKB.

3.3 Feeding the knowledge base

At each peer the knowledge base can be continuously improved with new problem descriptions and available solving methods.

In the Internet many implementations may exist for a given method. From the point of view of the web system two implementations of the same method may differ if, for example, they differ on its outputs. Unfortunately not all implementations work in the same way. Therefore, for the system to be able to use such implementations in a programmatic way, they must also be described within the system. This description must include, among other things, the address to the running method or program and its signature, which, in turn, includes the definition of the parameters that are necessary for its execution (inputs) and its output format. Figure 2 illustrates the system interface for defining the method's signature for an implementation of the branch and bound method proposed by Ignall and Schrage for a flow shop problem [3, 4].

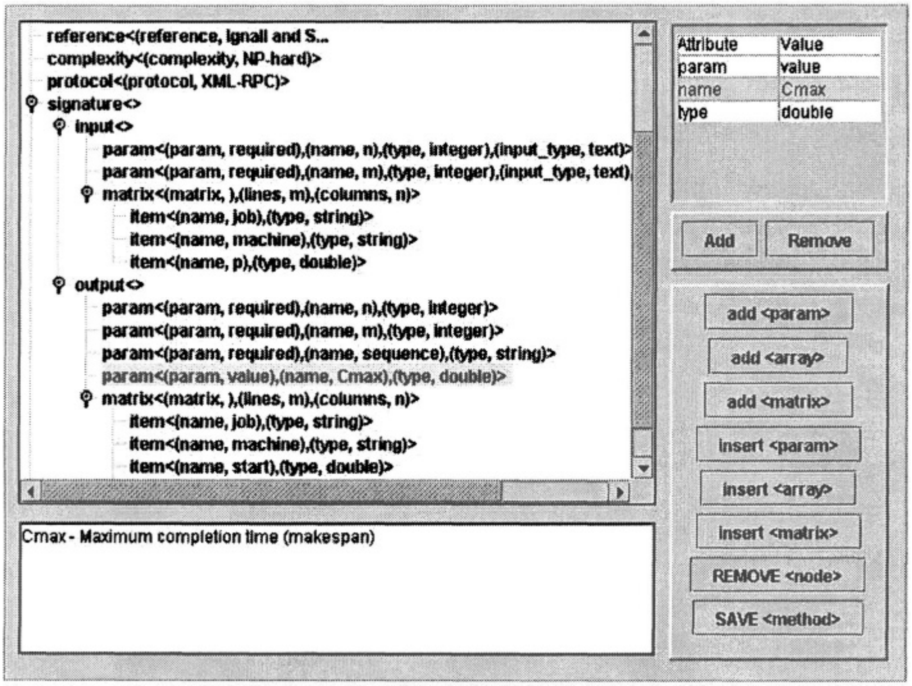


Figure 2 – Interface for new methods specification

The inputs include the definition of a parameter n for number of jobs to be processed, which is of integer type, a parameter m for number of machines in the production system and also a set of three items in a matrix structure, which represent the job name, the machine name and an additional parameter p that corresponds to job processing time. There is also the definition for the method's output following the same lines. This information is subsequently inserted in an XML document in order to enable further method's executions and information retrieval as well as automatic generation of interfaces for the implemented methods' inputs and outputs.

3.4 An example

For a better system's functionalities illustration let us consider a problem instance of the previously described F3|n|Cmax problem class, more precisely a problem with 4 jobs, which have to be processed in a flow shop with 3 machines. The objective, already known, consists on minimizing the maximum completion time (Cmax). Table 1 shows the time required for processing each job j on each machine i.

Table 1: Scheduling problem data

i / j	J1	J2	J3	J4
M1	3	11	7	10
M2	4	1	9	12
M3	10	5	13	2

Once the problem is correctly specified, accordingly to the referred classification nomenclature [8, 9], a request is sent to the active peers asking for equivalent or similar scheduling problems. The results are shown in Figure 3. The user can choose one or more related problem classes, specially if he or she is not totally sure about the problem to solve.

<table>
<tr><td colspan="5">Problem types for the general problem characteristics: [system_type=F, jobs=n, measure=Cmax]</td></tr>
<tr><th>Class</th><th>Complexity</th><th>Context</th><th>Problem characteristics</th><th>Select</th></tr>
<tr><td>F2|n|Cmax</td><td>Maximal Polinomially Solvable</td><td>≥</td><td>[(system_type, F), (machines, 2), (jobs, n), (measure, Cmax)]</td><td>☑</td></tr>
<tr><td>F2|rj,n|Cmax</td><td>Minimal NP-hard</td><td>≥</td><td>[(system_type, F), (jobs, n), (arrivals, rj), (machines, 2), (measure, Cmax)]</td><td>☐</td></tr>
<tr><td>F2|rj,n,no-wai</td><td>Maximal Polinomially Solvable</td><td>≥</td><td>[(system_type, F), (machines, 2), (jobs, n), (arrivals, rj), (buffers, no-wait), (measure, Cmax)]</td><td>☐</td></tr>
<tr><td>F3|rj,n|Cmax</td><td>Minimal NP-hard</td><td>≥</td><td>[(system_type, F), (machines, 3), (jobs, n), (arrivals, rj), (measure, Cmax)]</td><td>☐</td></tr>
<tr><td>F3|n|Cmax</td><td>Minimal NP-hard</td><td>≥</td><td>[(system_type, F), (machines, 3), (jobs, n), (measure, Cmax)]</td><td>☑</td></tr>
<tr><td>Fm|n|Cmax</td><td>Minimal NP-hard</td><td>≥</td><td>[(system_type, F), (machines, m), (jobs, n), (measure, Cmax)]</td><td>☑</td></tr>
<tr><td>Fm|prec,pji=1,</td><td>Minimal NP-hard</td><td>≥</td><td>[(system_type, F), (machines, m), (jobs, n), (precedences, prec), (times, pji=1), (measure, Cmax)]</td><td>☐</td></tr>
</table>

Figure 3 - Problem classes selection

Afterwards, using the selected problem classes, a second request is sent to the correspondent peers asking for the available solving methods. Figure 4 presents this information, which includes the link(s) for implemented methods' execution and other general information, for example, details about each problem class, the method's author and other related information. Behind this information the system also provides more details about the method and its implementation(s), so that an easier selection of adequate methods can be achieved.

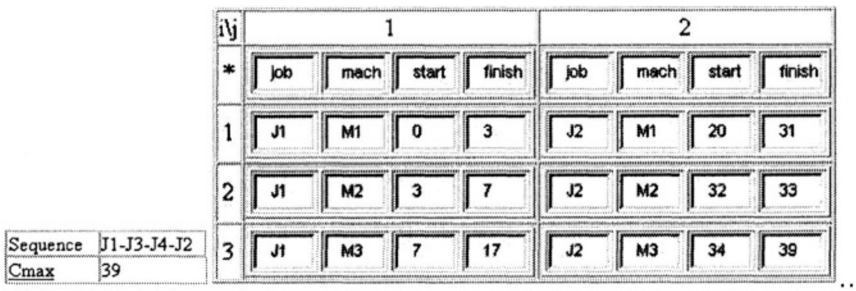
<table>
<tr><td colspan="8">Available methods for solving the problem classes: [class=F2|n|Cmax, class=F3|n|Cmax, class=Fm|n|Cmax]</td></tr>
<tr><th>ID</th><th>Prob. Class</th><th>Method</th><th>Problem characteristics</th><th>Reference</th><th>Complexity</th><th>Protocol</th><th>Select</th></tr>
<tr><td>1001</td><td>Fm|n|Cmax</td><td>BranchBoundIS</td><td>[(system_type, F), (machines, m), (jobs, n), (measure, Cmax)]</td><td>Ignall and Schrage, 1965</td><td>NP-hard</td><td>XML-RPC</td><td>◉</td></tr>
<tr><td>32</td><td>F2|n|Cmax</td><td>Johnson</td><td>[(system_type, F), (machines, 2), (jobs, n), (measure, Cmax)]</td><td>Johnson, 1954</td><td>Polynomial</td><td>XML-RPC</td><td>○</td></tr>
</table>

Figure 4 - Methods' implementations for solving problem

The methods are usually available in the knowledge base of the peers belonging to the VO but they can also be found in other sites not belonging to the community. An important aspect to be noticed is that the matching process, between problems and available methods, is performed by a built-in prolog search engine. This searching engine was developed using the SWI-Prolog V.5.2.1. free software tool available at http://www.swi-prolog.org/.

The Ignall and Schrage method [3, 4], previously selected in Figure 4, is a Branch-and-Bound (B&B) method available for solving F3|n|Cmax problems, which is an exponential time complexity method that belongs to the class of exact mathematical programming methods using the B&B technique. This method runs under the XML-RPC protocol, which is a Remote Procedure Call protocol that uses only XML messages [5, 10].

After running the selected method, the system presents an automatically generated interface for results presentation. Figure 5 illustrates the system interface for a direct presentation of problem results in the client browser for the problem under consideration.



Sequence	J1-J3-J4-J2
Cmax	39

i\j	1				2			
*	job	mach	start	finish	job	mach	start	finish
1	J1	M1	0	3	J2	M1	20	31
2	J1	M2	3	7	J2	M2	32	33
3	J1	M3	7	17	J2	M3	34	39

...

Figure 5 – Interface screenshots about problem results presentation

This kind of interface, for problem results presentation (as well as for problem data input) is automatically generated by the system according to the corresponding implemented method's signature. The implemented method's signature is specified when it is put forward in the distributed knowledge base. This new knowledge insertion process has been previously illustrated in section 3.3, where Figure 2 has shown the information that was added to the corresponding XML document about this B&B method.

The result from running an implemented method on a given problem instance can be delivered to the client as an XML file and/or can be transformed into some more expressive output, like a Gantt chart.

4. CONCLUSIONS

In this paper we presented a web system, based on a peer-to-peer (P2P) infrastructure, associated to a distributed knowledge base (DKB). This DKB is spread through a set of members (peers) forming a virtual organization. These members can store information about methods for solving different kinds of manufacturing scheduling problems, as well as its implementations. Each peer, in a decentralized computing environment, is able to directly interact with each other as well as with any other external user. This decentralized and direct collaboration is enabled through a dynamic web interface, which permits executing diverse scheduling functions, including the ability to represent different scheduling problems, search for appropriate solving methods and running methods that are available and accessible in the DKB, through this P2P network. As running different methods enables obtaining different solutions for a problem this system contributes to a better decision-making process, enhanced by means of the collaboration of the peers organized as a virtual organization.

5. REFERENCES

1. Camarinha-Matos, L. M.; Afsarmanesh, H., The Virtual Enterprise Concept, In: *Infrastructures for Virtual Enterprises, Networking Industrial Enterprises.* Kluwer Academic Publishers, 1999, pp. 3-14.
2. Camarinha-Matos, L. M.; Afsarmanesh, H., Design of a Virtual Community Infrastructure for Elderly Care, In: Collaborative Business Ecosystems and Virtual Enterprises, PRO-VE'02, Sesimbra, Portugal. Kluwer Academic Publishers, 2002, pp. 439-450.
3. Conway, R. W.; Maxwell, W. L.; Miller, L. W., Theory of Scheduling. England: Addison-Wesley Publishing Company, Inc., 1967.
4. Ignall, E.; Schrage, L.; Application of the Branch-and-Bound Technique to Some Flow-Shop Problems. Operations Research, 1965, 13:3.
5. Laurent, S., et al., Programming Web Services with XML-RPC. O'Reilly & Associates, Inc., 2001.
6. Papazoglou, M.P.; Krämer, B.J.; Yang, J.; Leveraging Web-Services and Peer-to-Peer Networks, In: Proceedings of Advanced Information Systems Engineering, 15th International Conference, CAiSE, Klagenfurt, Austria. June 16-18, 2003, pp. 485-501.
7. Terziyan, V.; Zharko, A.; Semantic Web and Peer-to-Peer: Integration and Interoperability in Industry, Industrial Ontologies Group, MIT Department, University of Jyvaskyla, Finland (http://www.cs.jyu.fi/ai/vagan/papers.html).
8. Varela, L. R.; Aparício, J. N.; Silva, C. S., An XML knowledge base system for scheduling problems, In: Proceedings of the Innovative Internet Computing System Conference, I2CS'02, Kuhlungsborn, Germany. Springer-Verlag in the Lecture Notes in Computer Science series, 2002; 61-70.
9. Varela, M. L. R.; Aparício, J. N.; Silva, S. C., Developing a Web Scheduling System Based on XML Modeling, In: Knowledge and Technology Integration in Product and Services – Balancing Knowledge and Technology in Product and Service Life Cycle, BASYS'02, Cancun, Mexico. Kluwer Academic Publishers, 2002, pp. 61-70.

10. Varela, M. L. R.; Aparício, J. N.; Silva, S. C., A Scheduling Web Service, In: Proceedings of the 1st Multidisciplinary International Conference on Scheduling: Theory and Applications, MISTA'03, Nottingham, UK. ASAP, The University of Nottingham. 2003, pg. 540-551.