A MAPPING BASED APPROACH TO SCHEDULE INTEGRATION IN HETEROGENOUS ENVIRONMENTS

Mohsin K. Siddiqui[1], William J. O'Brien[2], Ting-Kwei Wang[3]

ABSTRACT

Construction schedules are one of the most important project controls for construction project management. Schedule coordination traditionally has been a distributed and iterative exercise in which the General Contractor maintains the master schedule and the other participants schedule their activities using their individual constraint information. This paper introduces the concept of schedule mappings as a way to establish links between multiple participant schedules representing the same construction activity. Coordinating activities in a disconnected heterogeneous environment, where participants use different scheduling methods and model activities at various levels of details, is a challenging exercise which is not amply supported and explored by existing research. Coordination is usually an ongoing repetitive effort taken on by the General Contractor. Schedule mappings provide a structured mechanism to represent and share schedules and related process information across firms allowing rapid evaluation of schedule alternatives in response to a schedule change. We can view the shared schedule and process information at multiple levels of detail.

KEY WORDS

Schedule Integration, Construction Process Modeling, Decision Support, Constraint Propagation, Distributed Computing in Engineering

INTRODUCTION

Current scheduling technologies allow us to generate scalable and flexible models and allow us to optimize individual schedules under a range of constraints and objectives. Typically, scheduling research has focused on optimization with a well defined and static view of the schedule as a constraint satisfaction problem (Dechter 2003). However, in practice, scheduling is a dynamic iterative process in an unpredictable and uncertain execution environment (Smith 2003).

The different parties involved in a construction project view their responsibilities at a level of detail that is most useful for their business objectives. A General Contractor (GC) or

[1] PhD. Candidate, Department of Civil, Architectural and Environmental Engineering, University of Texas at Austin, 1 University Station C1752, Austin, TX 78712-0273, USA, mohsink@mail.utexas.edu

[2] Assistant Professor, Department of Civil, Architectural and Environmental Engineering, University of Texas at Austin, 1 University Station C1752, Austin, TX 78712-0273, USA, wjob@mail.utexas.edu

[3] PhD. Candidate, Department of Civil, Architectural and Environmental Engineering, University of Texas at Austin, 1 University Station C1752, Austin, TX 78712-0273, USA, tingkwei@mail.utexas.edu

Construction Manager (CM) in most cases owns the master schedule and uses critical path networks (Antill and Woodhead 1990) often representing activities for a subcontractor at an aggregated level of detail. The GC is usually unaware of the specific resource and capacity constraints of the subcontractors, which can limit the ability of the GC to coordinate schedules (O'Brien and Fischer 2000). A subcontractor models its responsibility in the project at a finer level of detail for proper control and management relying on critical path networks for the overall project view. Site staff usually relies on bar charts and activity lists for detailed planning of specific site tasks (Mawdesley et al. 1997).

Research in construction schedule integration has focused on integrating schedules with design, planning and cost (Froese et al. 1996; Rasdorf and Abudayyeh 1991). Other research focuses on providing a centralized platform and visual tools for different participants to provide their planning input (Dawood et al. 2005). A schedule is essentially a distributed artifact with different stake holders owing parts of it; most research and applications seek to integrate these perspectives into a single view. Some research has focused on providing various project participants pre-formatted schedule components that allow for a modicum of customization to individual needs (Dzeng and Wang 2003).These pre-formatted modules can then be easily integrated to a larger, integrated schedule. Our approach focuses on a heterogeneous environment ensuring greater flexibility by allowing project participants to maintain their own choices about the method and level of detail they wish to employ. We provide a method to compare schedules for the same construction tasks and establishing links across these schedules. Such links would allow us to integrate participant views and allow us to rapidly evaluate alternatives in response to a schedule change. In this paper, we describe schedule mappings and a tree-based approach, Mapping Trees, to capture the information generated during the schedule mapping process. Our representation provides a skeleton which can represent the construction scheduling constraints of multiple participants.

The remainder of this paper starts by introducing a practical problem followed by stating our assumptions and key terminology. We elaborate on the idea of a schedule Mapping for a two party case using the introduced terminology. We outline the processes of initial discovery of a mapping, derivation of smaller mappings from the initial mapping, and validation of the smallest possible mappings. We then outline our criteria for selecting a storage option to represent the information gathered during the discovery process and introduce the mapping tree. Future research and extensions are discussed in the end.

CASE EXAMPLE: CENTEX VS MILLER

Our case example focuses on a subset of a construction project for two project participants – a construction manager and a subcontractor. Centex was the CM for the project with different subcontractors provided various services including Miller Electric responsible for the electrical work (case details are drawn from (Castro-Raventós 2002)). As the CM, Centex is responsible for the master schedule. The master schedule uses a hierarchy in which activities are grouped by floor and activities for the subcontractors are represented in this schedule. Miller is responsible for the electrical activities on the project and approaches scheduling in a very different manner by grouping their activities using internal cost codes. Figure 1 lists Centex and Miller activities.

Usually there is no direct 1:1 mapping between schedules in such a case and delay of any activity in either schedule cannot be readily reflected in the corresponding schedule of the other affected party. Centex and Miller both schedule their tasks considering only their respective constraints and Miller could be initially bound by Centex to schedule its activities in a given finite time period. Such restrictions imposed by Centex require significant initial coordination, and any subsequent change in either schedule would require a similar iterative effort for re-coordination (O'Brien et al. 1995). An approach that can capture the initial coordination results and expose the constraints of the participants involved would simplify the handling of subsequent changes.

Centex Schedule		Miller Electric Schedule	
Activity ID	Activity Description	Activity ID	Description
0211	Elec. Rough in/Equipment/Tie-in	32	0200-Fixtures
1020	Overhead Elec. Rough in	33	0300-Wiring Devices
1040	In-wall Elec. Rough in	34	0500-Cable Tray
1210	Install Light Fixtures	35	0602-Above Grade
1255	Install Electrical Finishes	36	0900-Single Conductor Wire
2020	Overhead Elec. Rough in	37	1300-Equipment
2045	In-wall Elec. Rough in	39	0200-Fixtures
2090	Install Light Fixtures	40	0300-Wiring Devices
2110	Install Electrical Finishes	41	0500-Cable Tray
3020	Install Electrical Finishes	42	0602-Above Grade
3045	In-wall Elec. Rough in	43	0900-Single Conductor Wire
3090	Install Light Fixtures	44	1300-Equipment
3110	Install Electrical Finishes	46	0200-Fixtures
		47	0300-Wiring Devices
		48	0500-Cable Tray
		49	0602-Above Grade
		50	0900-Single Conductor Wire
		51	1300-Equipment
		53	0300-Wiring Devices
		54	0602-Above Grade
		55	0900-Single Conductor Wire
		56	1300-Equipment

Figure 1: All activities from Miller and corresponding electrical work activities from Centex

KEY TERMS AND ASSUMPTIONS

As a first step, we are focusing on temporal constraints only. We are considering FS (Finish – Start) relationships among activities and extend our model to capture other constraints as a part of our future work. Our focus is on formalizing and integrating existing information to provide the groundwork for additional functionality. We focus on a two party case (one General Contractor / Construction Manager and one Subcontractor) for this paper and will describe in general terms about how our approach can be extend to include multiple parties. As most scheduling research in construction focuses on a single perspective, there is a need to introduce a few key terms and definitions that support a schedule mapping perspective. These are defined below:

ACTIVITY SET

A meaningful group of activities extracted from a participant's schedule that can be mapped to a meaningful group of activities from the other participant's schedule is called an activity set. An activity set might consist of one or many activities and potentially map to an activity set with one or many activities. The owner of the activity set is the participant who owns the schedule from which that activity set has been extracted.

TIME WINDOW

A time window is the representation of the planning time horizon of an activity set as viewed by its owner. A time window is characterized by a single start and a single finish date. The start date represents the earliest date from the start dates of activities in an activity set. The finish date is identified as the latest date from the finish dates of activities in an activity set.

INITIAL MAPPING

An Initial mapping consists of two activity sets where the first activity set includes all the scheduled activities of one participant. Corresponding activities are identified from the coordinator's schedule and are grouped into the second activity set. A time window is associated with each of these activity sets.

COORDINATOR

Each project has a master schedule that is owned and maintained by a single firm. The owner of the master schedule plays the role of the *coordinator* and orchestrates the activities of the other participants to achieve overall project objectives. A CM or GC usually plays this role in the construction domain. Information in the coordinator's schedule provides the necessary details about precedence constraints and is essential for building the Mapping Tree.

CONTROLLING PARTICIPANT

A mapping is a collection of activity sets identified from the schedules of two stakeholders. These project players normally have very different time and resource flexibility around their scheduled activities. Rescheduling activities of an overcommitted participant may result in very long delays and must be avoided. The stakeholder that imposes the least flexible constraints is identified as the *controlling participant* for that mapping. Constraints of such participant drive the process of validating the mappings and suggesting solutions.

INTRODUCTION TO SCHEDULE MAPPINGS

The idea behind schedule mappings is to discover activity sets from the schedules of stakeholders that represent the same project task (Figure 2). A schedule mapping provides means to represent information about such discovered activity sets. The information captured includes unique identifiers for the activities, the time windows and temporal constraint information. Figure 3 represents the high level overview of a schedule mapping characterized by two time windows with their respective start and finish dates shown on a time scale where:

S1	:	Start time as viewed by Participant 1 (coordinator)
S2	:	Start time as viewed by Participant 2
F1	:	Finish time as viewed by Participant 1 (coordinator)
F2	:	Finish time as viewed by Participant 2

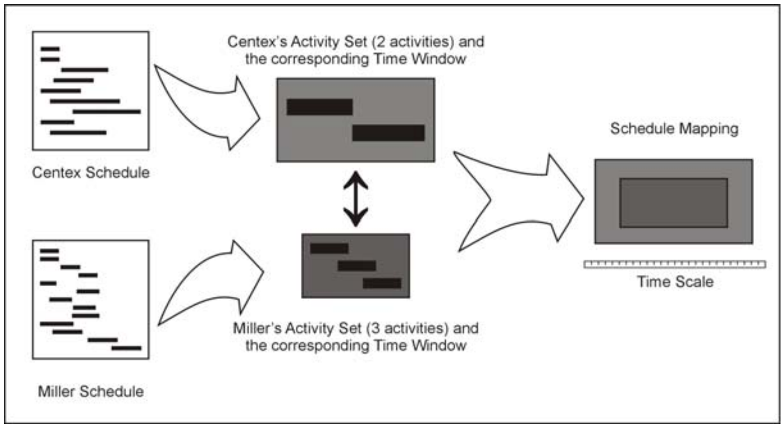

Figure 2: Mapping of activity sets resulting in a schedule mapping

The outer window would typically represent the time window of the coordinator's activity set. In addition to the time windows, each mapping represents the following information about the activity sets it covers.

- Number of activities in each activity set (1:1, 1:m, m:n etc)
- Lists of activity IDs for activities in both the activity sets
- Start and finish dates for each time window
- Precedence constraints information for the mapping (not activity)

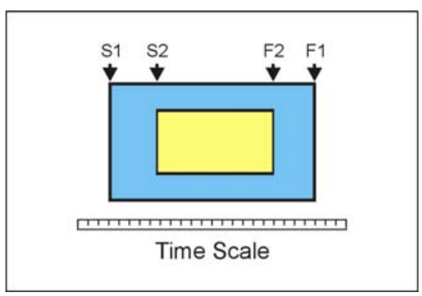

Figure 3: High level overview of an Initial schedule mapping

Figure 3 shows a *valid* mapping where a mapping is considered valid as long as both participants have a temporally consistent view and the equation below ensures that.

$$S1 <= S2 < = F2 <= F1.$$

Any violation of the above equation results in an inconsistency as show in Figure 4. An invalid mapping identifies inconsistency in the time horizons of the participants and all such conflicts must be resolved. Validation is discussed as a part of the discovery process, below.

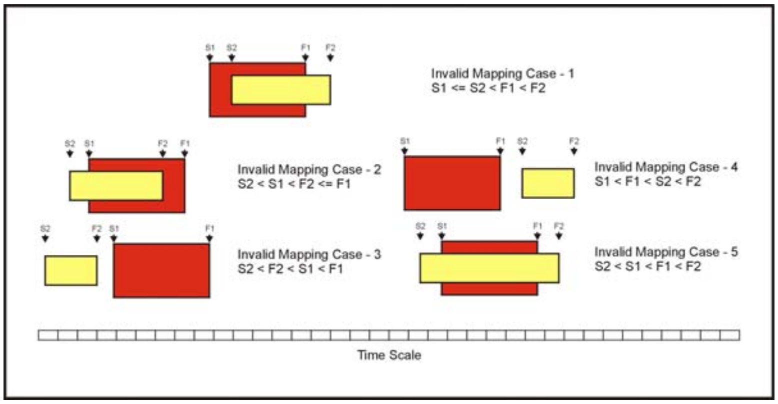

Figure 4: Possible Invalied Mapping Cases

DISCOVERY OF SCHEDULE MAPPINGS

STEP 1: INITIAL MAPPING

In this step we discover and record the Initial Mapping between the schedules of the participants. The diverse backgrounds of the planners along with the heterogeneity of our problem make it a difficult task to discover the Initial Mappings through an automated process. We acknowledge that human involvement in this process is indispensable. However, we can assist the mapping process by scanning the coordinator's schedule using clues based on nomenclature, work area, time overlaps and resource utilization information. Figure 1 can be classified as an initial mapping which is obtained by scanning the Centex schedule for activities that correspond to Miller's activities.

STEP 2: MAPPING DECOMPOSITION

In this step we break down the initial mapping into smaller sub mappings. The initial mappings provide a very high level relationship among the activities of the coordinator and the other party with huge time windows. The decomposition process consists of manually discovering meaningful relationships among the activities in the activity sets that make up the initial mapping. Figure 5 shows the result of the decomposition process which has been continued until we discover the smallest possible mappings.

Smallest Possible Mappings

We can recursively decompose any mapping into smaller mappings by trying to further discover relationships among activities listed in the activity sets of an existing mapping in an effort to shrink the time windows. The reduced size of time windows would potentially allow us to localize the effect of subsequent changes and minimize coordination needs (Smith 2003). The recursion would end when we are either left with only one activity in either activity set or no further relationships can be inferred from the available information. The resulting mappings are referred to as the *smallest possible mappings* or the *smallest possible decompositions*. These final, smallest possible decompositions can be 1:1, 1:n, or m:n mappings between individual activities. Note that figure 5 depicts several m:n mappings.

It must be noted that the hierarchy of decomposition has meaningful information embedded into it. The way we split these mappings into smaller mappings usually follows a mix of scheduling hierarchy used by the coordinator and the other participant.

Mapping No	Centex Schedule ID	Centex Schedule Activity Name	Miller Schedule ID	Miller Schedule Activity Name
1	1st Floor Activities		1st Floor	
2	1210	Install Light Fixtures	32	0200-Fixtures
3	1040 1255 1020	In-wall Elec. Rough in Install Electrical Finishes Overhead Elec. Rough in	33 34 35 36 37	0300-Wiring Devices 0500-Cable Tray 0602-Above Grade 0900-Single Conductor Wire 1300-Equipment
4	2nd Floor Activities		2nd Floor	
5	2090	Install Light Fixtures	39	0200-Fixtures
6	2045 2110 2020	In-wall Elec. Rough in Install Electrical Finishes Overhead Elec. Rough in	40 41 42 43 44	0300-Wiring Devices 0500-Cable Tray 0602-Above Grade 0900-Single Conductor Wire 1300-Equipment
7	3rd Floor Activities		3rd Floor	
8	3090	Install Light Fixtures	46	0200-Fixtures
9	3045 3110 3020	In-wall Elec. Rough in Install Electrical Finishes Install Electrical Finishes	47 48 49 50 51	0300-Wiring Devices 0500-Cable Tray 0602-Above Grade 0900-Single Conductor Wire 1300-Equipment
10	Structure/Exterior/Roof		Roof	
11	0211	Elec. Rough in/Equipment/Tie-in	53 54 55 56	0300-Wiring Devices 0602-Above Grade 0900-Single Conductor Wire 1300-Equipment

Figure 5: Smallest possible mappings extracted from initial mapping in Figure 1

STEP 3: ADDING PRECEDENCE INFORMATION TO THE SCHEDULE MAPPINGS

Once the mappings have been established, we add the constraint information for each smallest possible mapping. We can access precedence constraint information about each activity in the coordinator's activity set by probing the coordinator's schedule. If the related activity lies within the same activity set, that constraint information is ignored and not recorded as a part of the Mapping. If an activity's precedence information points outside its parent mapping, we locate the parent of the target and transform it into an equivalent precedence constraint among the respective parent mappings. It is accomplished by offsetting the dates and lags relative to the mapping start and finish dates instead of activity dates. We record predecessor and successor information in the respective mappings for each such instance. The information acquired includes the type of constraint, the target mapping, and the temporal offset that must be maintained between the mappings.

STEP 4: TIME CONSISTENCY VALIDATION

Once we have discovered the information from the schedules of the participants, we must ensure that no mapping is invalid as shown in Figure 4. We independently verify each mapping and resolve the conflicts causing minimal disturbance by modifying the schedule of only one participant. The presence of temporal constraints complicates the validation process where we have to ensure that any resolution does not violate these constraints. We accomplish that by using a “Right Shift” algorithm specific details of which are beyond the scope of this paper.

Human involvement in this process is indispensable and our current approach would provide a solution that’s valid but not necessarily optimal. A user will evaluate the results of our validation and will chose to accept or reject the proposed solution.

STORAGE OF MAPPING INFORMATION – MAPPING TREE

The choice of a storage alternative would influence the reliability and performance of our overall application. For an efficient, accurate and scalable solution, information gathered during the discovery and decomposition process must be stored in a format that would allow the following:

- Ability to uniquely identify a mapping at any level of decomposition
- Ability to maintain or reconstruct all information about the initial mapping and for all subsequent decompositions
- Avoid redundant storage of information to make the solution scalable. Information for initial mappings, in most cases, can be inferred from their decompositions. Most up to date information about each activity can be accessed from the original schedules
- Ability to represent precedence information for efficient propagation of delays
- Efficient means for retrieval of Mappings and Activities

Our recommended approach is to store detailed information about the smallest possible mappings and build a modified Tree index on top of that. Our proposed Tree indexing structure borrows characteristics from other classical database indexing structures. In this approach we only store the start and finish dates for the initial mapping and subsequent decompositions. Detailed information is stored only for the smallest possible mappings. Detailed information can be computed for higher level decompositions by simply summing up information from levels below.

The ability to reconstruct information about intermediate decompositions is essential for our proposed system. We need to store only the name and time window information for intermediate decompositions and the remaining information is simply an aggregation of data of all the children of an intermediate node. Figure 6 shows the representation of a generic tree for our case using information from Figure 5. The constraints are represented by arrows between the smallest possible decompositions.

We believe that using the tree based representation is beneficial as it allows us to reason about the schedule mappings at any granularity in the hierarchy. We can simply prune the tree and conduct the same validity analysis without any major changes. In addition, a participant might elect one of the intermediate nodes as more significant for their particular application in which case all the children of that node will be merged. For the sake of interoperability, we share this information using XML documents and a portion of the XML generated for the tree in Figure 6 is show in Figure 7.

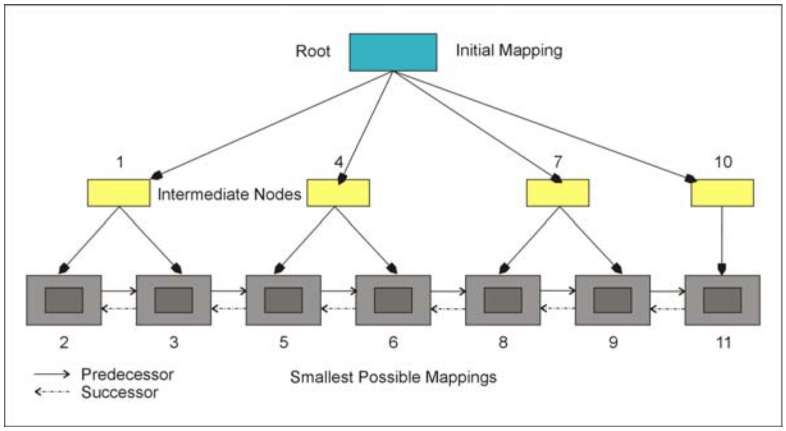

Figure 6: Tree based index for the Smallest Possible Mappings using data from Figure 5

```xml
.....
....
....
 <Mapping ID = "1">
        <StartDate>05/01/2002</StartDate>
        <FinishDate>08/31/2003</FinishDate>
        <ChildMapping>2</ChildMapping>
        <ChildMapping>3</ChildMapping>
</Mapping>
...
...
...
<Mapping ID = "3">
        <ActivitySetA>

        <StartDate>05/01/2002</StartDate>

        <FinishDate>05/15/2002</FinishDate>

        <NumActivities>3</NumActivities>
                <ActivityID>1040</ActivityID>
                <ActivityID>1255</ActivityID>
                <ActivityID>1020</ActivityID>
        </ActivitySetA>
<ActivitySetB>

        <StartDate>05/04/2002</StartDate>

        <FinishDate>05/13/2002</FinishDate>

        <NumActivities>5</NumActivities>
                <ActivityID>33</ActivityID>
                <ActivityID>34</ActivityID>
                <ActivityID>35</ActivityID
                <ActivityID>36</ActivityID>
                <ActivityID>37</ActivityID>
        </ActivitySetB>
        <Predecessor>
                <MappingID>2</MappingID>
                <Lag>0</Lag>
        </Predecessor>
        <Successor>
                <MappingID>5</MappingID>
                <Lag>0</Lag>
        </Successor>
</Mapping>
....
.....
```

Figure 7: XML representation of a Mapping Tree

CONCLUSIONS

In this paper we have introduced the idea of schedule mappings as a way to integrate schedules in a multiparty heterogeneous environment. Traditional scheduling approaches rely on scheduling activities from the viewpoint of a single participant. Our approach provides means to integrate schedules of different parties while allowing them to independently maintain their respective schedules. Our approach maintains the flexibility and provides a tool to assist in distributed coordination.

Our representation can be thought of as an *overlay of networks*. A two party case has been introduced which can be extended to a multiparty scenario by using the following approach. In a multiparty scenario, we can build similar mapping trees for each subcontractor vs. general contractor schedule and then link those trees at the leaf level using the precedence information from the general contractor's schedule. The result would be a *network of overlaid networks,* which would allow rapid assessment of time impact of a change and a reasonable exploration of alternatives. The resulting artifact, in essence, is an integrated master schedule comprising of static links among dynamic schedule mappings which can serve as a foundation for more intelligent decision support tools.

We have focused on a small category of temporal constraints for this paper and we are extending our approach to cover other complex constraint types. Once a strong temporal network has been built and validated, we can further extend our technique to expose resource constraints of the involved participants. Our research also focuses on automated response to a schedule change by using a "Right Shift" algorithm to propagate delays.

ACKNOWLEDGEMNTS

The authors wish to thank the National Science Foundation for its support of this research under grants CMS-0542206 and CMS-0531797. We also wish to thank Dr. Joachim Hammer (Associate Professor) and Jungmin Shin (PhD Student) Department of Computer and Information Science Engineering, University of Florida, for their insightful feedback.

REFERENCES

Antill, J. M., and Woodhead, R. W. (1990). *Critical path methods in construction practice*, Wiley, New York.

Castro-Raventós, R. (2002). "Comparative Case Studies of Subcontractor Information Control Systems," Unpublished M.S. Thesis, University of Florida.

Dawood, N., Scott, D., Sriprasert, E., and Mallasi, Z. "The Visual Construction Site (VIRCON) Tools: An Industrial Evaluation." *ITCon*, 43-54.

Dechter, R. (2003). *Constraint processing*, Morgan Kaufmann Publishers, San Francisco.

Dzeng, R.-J., and Wang, W.-C. (2003). "Automatic schedule integration for highway projects." *Automation in Construction*, 12(4), 447-461.

Froese, T., Yu, K., and Shahid, S. (1996). "Project modeling in construction applications." *Congress on Computing in Civil Engineering, Proceedings*, 572-578.

Mawdesley, M., O'Reilly, M. P., and Askew, W. (1997). *Planning and controlling construction projects : the best laid plans*, Longman, Essex, England.

O'Brien, W. J., and Fischer, M. A. (2000). "Importance of capacity constraints to construction cost and schedule." *Journal of Construction Engineering and Management*, 126(5), 366-373.

O'Brien, W. J., Fischer, M. A., and Jucker, J. V. (1995). "An economic view of project coordination." *Construction Management & Economics*, 13(5), 393-400.

Rasdorf, W. J., and Abudayyeh, O. Y. (1991). "Cost- and schedule-control integration. Issues and needs." *Journal of Construction Engineering and Management*, 117(3), 486-502.

Smith, S. F. "Is Scheduling a Solved Problem?" *Proceedings First Multi-Disciplinary International Conference on Scheduling: Theory and Applications (MISTA 03)*, Nottingham, UK.