A FRAMEWORK FOR CONSTRUCTION REQUIREMENTS BASED PLANNING UTILIZING CONSTRAINTS LOGIC PROGRAMMING

David K.H. Chua[1] and K.W. Yeoh[2]

ABSTRACT

In the lean construction philosophy, the management of constraints is essential to reduce project delays. These constraints can be derived from construction requirements which define the characteristics of the construction project. This paper discusses the evolution and classification of requirements. Additionally, a framework to semantically map the construction requirements to schedule constraints called PDM++ is proposed, which models the schedule impact of such requirements. Finally, an analysis methodology is proposed to identify the criticality of constraints and construction requirements. This allows project managers to subsequently manage these critical requirements. An illustrative example is presented to demonstrate the usage of PDM++ and the proposed analysis methodology.

KEY WORDS

Construction requirements, constraints management, integrated planning and scheduling, computer-aided scheduling.

INTRODUCTION

In the lean construction philosophy, the management of constraints between activities is essential in pre-empting schedule variations and consequently reducing project delays. These constraints define the underlying "physics" of the construction project system, and the effective management of these constraints can lead to minimized variations at the project level (Howell 1999).

In general, these constraints are governed by the project's requirements. Kamara et al (2000) differentiated project requirements into three main types: Client requirements, design requirements and finally, construction requirements. Client requirements refer to the business needs of the stakeholders in the project, while design requirements include the design specifications and the governing regulatory codes of practice. Construction requirements are described as the concerns and constraints that should be fulfilled for conducting procurement, construction and logistic processes (Song and Chua 2006). As such, construction requirements represent the information flow between processes, key resource interdependencies, product component sequences/interconnections, intermediate function requirements

1 Associate Professor, Dept of Civil Engineering, National University of Singapore, 10 Kent Ridge Crescent, Singapore 119260, (65) 65162195, FAX (65) 67791635, cvedavid@nus.edu.sg

2 Research Scholar, Dept of Civil Engineering, National University of Singapore, 10 Kent Ridge Crescent, Singapore 119260, (65) 65164643, FAX (65) 67791635, g0600355@nus.edu.sg

like temporary works or supporting product components, and even contractual deadlines. Hence, it is necessary to identify such requirements so that feasible construction planning can occur. Despite this, little attention has been accorded to the impact of construction requirements on the project schedule through its associated schedule (temporal) constraints.

Schedule constraints may be derived from functional and non-functional construction requirements. Functional construction requirements are the construction intentions for supporting a construction process or for sustaining the in-progress structure. Non-functional requirements on the other hand, refer to the performance constraints like capacity and productivity (Song and Chua 2006). The fulfilment of construction requirements is necessary to ensure that scheduling conflicts do not arise at upstream activities, and increase the variability of downstream activities.

Traditional planning and scheduling models like Critical Path Method (CPM) and Linear Scheduling Method (LSM) cannot adequately capture many of these construction requirements, like work/resource continuity and process concurrency/overlap (Jaafari 1984). Additionally, CPM dictates work sequences when alternative work sequences exists which also fulfil the construction requirements. This limits the semantic translation of requirements to schedule constraints.

Ideally, constraints management should be carried out by both higher-level Project Managers and the lower-level Project Supervisors. However, the lack of detail and transparency in the translation of requirements to constraints disrupt the transfer of plans from managers to supervisors or from general contractors to their subcontractors. Misinterpretation of the constraints could amount to rework and contribute additional waste in the project lifecycle.

Current lean construction frameworks have realized the importance of including the resource and information availabilities to identify the project bottlenecks (Chua and Shen 2005). This implies a need to integrate the planning (precedence constraints) process with the scheduling process of key resources for in-depth analysis. Additionally, the Last Planner system advocates the importance of managing constraints to minimize work flow variability and uncertainty (Ballard and Howell 1998). As such, it can be seen that successful project management is achieved through the proper management of constraints, which is equivalently the fulfilment of the construction requirements.

This paper presents a framework which semantically maps the construction requirements to schedule constraints. This effectively minimizes the loss of detail through the mapping process, reducing ambiguity in the plan from higher to lower levels of project management. The framework essentially establishes an integrated planning and scheduling model, which is implemented through Constraint Logic Programming (CLP). The CLP methodology inherently allows for alternative schedules to be obtained through an in-built search and inference engine.

Finally, an analysis methodology identifying the criticality (and conversely, the flexibility) of constraints/requirements is also introduced. This constraint criticality coupled with increased transparency of the framework, allows all project participants to identify and focus on the driving requirements for project success. The proposed framework and methodology is demonstrated in an illustrative example.

MODELLING METHODOLOGY

FRAMEWORK FOR TRANSLATING REQUIREMENTS TO SCHEDULE CONSTRAINTS

The main project requirement types have been defined by Kamara et al (2000) as Client, Design and Construction requirements. The evolution process of project requirements from client requirements to design requirements and construction requirements follows as shown in Figure 1.

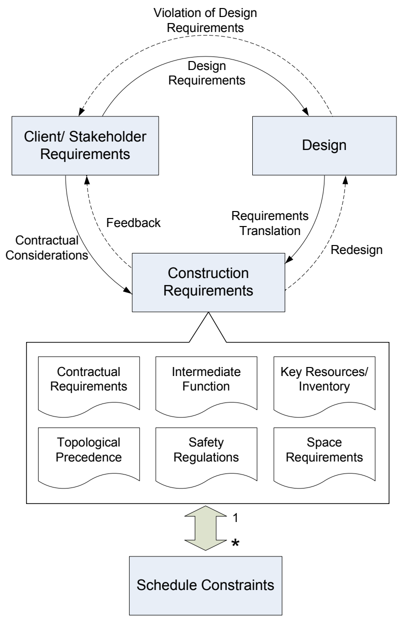


Figure 1: Evolution of Requirements

The elicitation of client requirements is a necessary step during the conceptual phase of the project, allowing us to define the project's value to the client, and the various stakeholders. The client requirements are then translated into the design requirements, which are also subjected to the design regulatory standards. Violation of design standards may be feedback to review the client's business needs. The design requirements are translated to the construction requirements, usually in the form of shop drawings. However, construction requirements will often require that the initial design requirements be reviewed to facilitate practical construction methods, often subjected to site/environmental conditions. Client requirements also directly impact the construction requirements by contractually specifying datelines and specific construction methods and/or materials. Inversely, the inability to satisfy construction requirements may also be feedback to the clients, possibly casing a redefinition of the client's business needs.

The gathered construction requirements may be modelled as functional or non-functional. Functional requirements refer to construction intentions for supporting a construction process, while the non-functional requirements are performance constraints such as capacity, productivity and inventory. The necessary individual construction activities and their corresponding temporal relationships may be inferred from the functional and non-functional construction requirements through the consideration of the relational logic from the following perspectives: Topological Precedence (from Product Models), Intermediate Function (e.g. Temporary Works), Space Requirements, Key Resources (Inventory), Safety Requirements and Contractual Requirements.

Examples of construction requirements may be given as: “Painting needs to be done after the scaffold erection and before the scaffold dismantling” or “The supervision of the formwork installation and the formwork fabrication is done by the same foreman”. These construction requirements are collated in a “Requirements List”, which also indicates the construction processes that are related to the requirement. The list provides justification for the planning process, and allows greater transparency between contractors and their specialty subcontractors. Furthermore, it allows for information to be shared between the different specialty subcontractors, leading to greater facilitation for the management of requirements.

Finally, the specified construction requirements are analyzed for relevant temporal constraints, which are translated syntactically using the proposed model in the following section. The link from requirements to constraints follows a one-to-many relationship, where one requirement may lead to multiple constraints, while a constraint may belong to only one requirement.

A REQUIREMENTS SCHEDULING LOGIC NETWORK MODEL

This paper develops a network model similar to Precedence Diagramming Method (PDM), but with an extended syntax such that common construction requirements derived from the above framework may be described temporally as logical constraints between activities. Much of these new logical constraints are inspired by Artificial Intelligence representation approaches developed by Allen (1984), and Song and Chua (2007) have used Allen’s temporal relations to model the class of intermediate function requirements.

The proposed model, PDM++ may be graphically represented as a constraint network $G = (V(D), E)$ where vertices V represent the construction activities each with an individual domain D being the activity starting times, while edges E represents the temporal logic constraints/relationships defined between activities. Due to its similarity to PDM, it may subsume present PDM models.

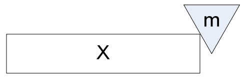
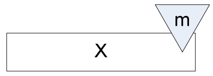
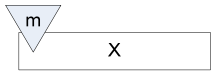
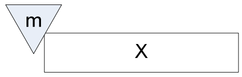
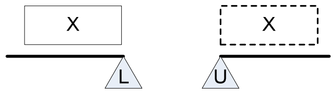
Assuming that the activities are non splittable and having fixed durations, allow the End point of an activity to be expressed as a simple linear function of the Start point only, which is expressed as

$$X^{-} + d_X = X^{+} \qquad \text{Equation 1}$$

where X^+ denotes the End point of the activity, X^- denotes the Start point and d_X is the duration of the activity.

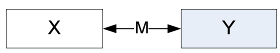
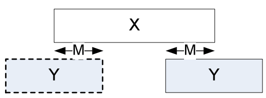
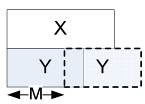
PDM++ generally comprises of two different types of constraints: Unary and Binary. Unary constraints are defined as constraints affecting a single activity (vertex). Binary constraints, on the other hand, define the relationship between two activities or vertexes. The following table depicts the unary constraints of PDM++.

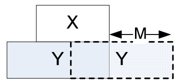
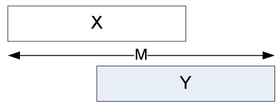
Table 1: Unary PDM++ Relationships

Relationship	**Mathematical Definition**	**Pictorial Representation**
X Due-before (*m*)	$X^- + d_X \leq m$	
X Due-after (*m*)	$X^- + d_X \geq m$	
X Start-before (*m*)	$X^- \leq m$	
X Start-after (*m*)	$X^- \geq m$	
X Cannot-Occur [*L*,*U*]	$(X^- + d_X \leq L) \vee (X^- \geq U)$	

The binary constraints in PDM++ may generally be distinguished into three different relationship types: Minimal-lag type, Maximal-lag type and Non-lag type. The Minimal-lag type defined in Table 2 is the usual lag definitions adopted in present PDM. This means that the constraint must *at least* satisfy the given lag time, *m* in the relationship. The Maximal-lag type, also defined in Table 2 is adopted from prior research by Hajdu (1997). The intended meaning of the relationship is such that the constraint must *at most* satisfy the given lag time, *m*. The different mathematical representations for Minimal-lag and Maximal-lag types are distinguished in Table 2 by adding a tilde to the Maximal-lag type relationships.

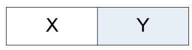
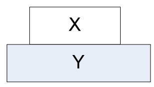
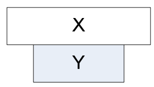
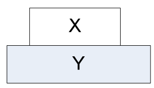
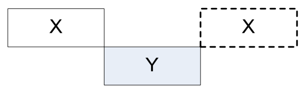
Table 2: Binary PDM++ Relationships

Relationship	**Mathematical Definition**	**Pictorial Representation**
X Before(*m*) *Y* *Y* After(*m*) *X*	$X^- + d_X + m \leq Y^-$	
X Before(~*m*) *Y* *Y* After(~*m*) *X*	$X^- + d_X + m \geq Y^-$	
X Overlaps(*m*) *Y* *Y* Overlapped-by(*m*) *Y*	$(X^- + d_X \geq Y^- + m) \wedge (Y^- + d_Y \geq X^- + m)$	
X Overlaps(~*m*) *Y* *Y* Overlapped-by(~*m*) *Y*	$(X^- + d_X \leq Y^- + m) \wedge (Y^- + d_Y \leq X^- + m)$	
X Starts(*m*) *Y* *Y* Started-by(*m*) *X*	$X^- + m \leq Y^-$	
X Starts(~*m*) *Y* *Y* Started-by(~*m*) *X*	$X^- + m \geq Y^-$	

X Finishes(m) Y Y Finished-by(m) X	$X^- + d_X + m \le Y^- + d_Y$	
X Finishes($\sim m$) Y Y Finished-by($\sim m$) X	$X^- + d_X + m \ge Y^- + d_Y$	
X Start-Finish(m) Y	$X^- + m \le Y^- + d_Y$	
X Start-Finish($\sim m$) Y	$X^- + m \ge Y^- + d_Y$	

The third type of relationships, the Non-lag type is proposed in PDM++, and depicted in Table 3. The Non-lag type, as the name suggests, is independent of any lag times, and provides greater descriptive capabilities to define the relationship between two activities, based on the interpretation of the construction requirement.

Table 3: Non-Lag Type Binary PDM++ Relationships

Relationship	**Mathematical Definition**	**Pictorial Representation**
X Meets Y Y Met-by X	$X^- + d_x = Y^-$	
X Within Y Y Without X	$(X^- \ge Y^-) \wedge (X^- + d_X \le Y^- + d_Y)$	
X Concurrent Y	**Case 1:** $d_X \ge d_Y$ $(X^- \le Y^-) \wedge (X^- + d_X \ge Y^- + d_Y)$	
	Case 2: $d_Y > d_X$ $(X^- \ge Y^-) \wedge (X^- + d_X \le Y^- + d_Y)$	
X Disjoint Y	$(X^- + d_X \le Y^-) \vee (Y^- + d_Y \le X^-)$	

The above relationships allow a semantic description of construction activities which closely follows the natural language for construction requirements. The above relationships differ from the normal PDM by describing relations between the activities rather than describing constraints between the endpoints of the activities (Start point and End point). Such an "interval-to-interval" representation allows for richer semantic context to describe requirements.

Additionally, the inclusion of logical operators expands the syntax for capturing and subsequently translating the requirements to temporal constraints. The mathematical relations above then translate the interval-to-interval descriptions to relationships relating the different start points and finish points.

PDM++ is implemented in *ECL^iPS^e*, a CLP language with the interval constraint library. The output generated is then either a set of feasible schedules which fulfil the mathematical representation of the constraints arising from the construction requirements while also optimizing the project makespan, or no feasible schedule exists. For each activity in a feasible schedule, a domain of values is returned indicating the range of possible start times for that particular activity.

CONSTRAINTS ANALYSIS

The above PDM++ model emphasizes the management of constraints rather than just solely managing activities on critical paths. This is because PDM++ may generate several alternative schedules, with differing critical paths. A constraint is identified as being critical if it is a binding constraint, i.e. the activities affected by the constraint has a singleton value in its domain, and the constraint is exactly satisfied by the singleton values. These critical constraints can then be traced back to its construction requirement, allowing appropriate managerial action to be taken.

Further, a set of constraints which is identified as critical in all the alternative schedules may warrant greater attention from managers. Delays or violations of any constraint in this "super-critical" set will invariably affect the project makespan. Another set of constraints are identified, which are critical in only some of the alternative schedules. This "sub-critical" set also requires attention from managers. Identifying this "sub-critical" set allows for plan flexibility when unforeseen circumstances occur which perturb the plan. Hence, when a "sub-critical" constraint is perturbed, a possible mitigation may be to proceed with an alternative schedule where the affected constraint is no longer critical.

The effective identification of "super-critical" and "sub-critical" constraints allow managers to identify the driving requirements of a project. Also, managers can then identify a "secondary" set of requirements which if perturbed, could force alternative schedules to be considered.

ILLUSTRATIVE EXAMPLE

A simplified example of installing a steel pipe rack in an oil refinery is used to demonstrate the application of PDM++ and its corresponding constraints analysis. The entire pipe rack is divided into three phases of construction, with Phase 1 and 3 spanning a length of 8m and a height of 2.5m, and Phase 2 spanning 5m by 4m. Figure 2. provides a 3D perspective.

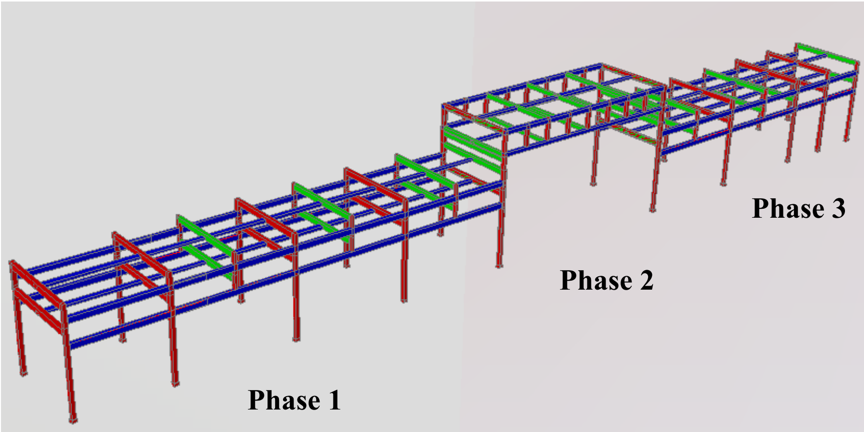


Figure 2: 3D Perspective of Pipe Rack Installation

Some of the pertinent project requirements are detailed as follows:

- Requirement 1: Both phases of shallow foundations are done concurrently.
- Requirement 2: Piperack columns are carried out by the same subcontractor.
- Requirement 3: For each phase, the scaffold erection can be done after one day of piperack column installation.
- Requirement 4: The first and second phase of scaffold erection must be done concurrently, before the start of the third phase.
- Requirement 5: The pipe laying must be carried out continuously.

The results of solving the network are shown in Table 3. Two alternative schedules are generated which eventually give the same project makespan of 42 days.

Figure 3. shows the resulting project constraint network describing the above requirements graphically. The temporal constraints are indicated on the directed arcs. Directed arcs without any indications are assumed to depict the "before" constraint, which is analogous to the normal precedent constraint in PDM. The "super-critical" constraints are highlighted in bold, while the "sub-critical" constraints are marked by dotted lines.

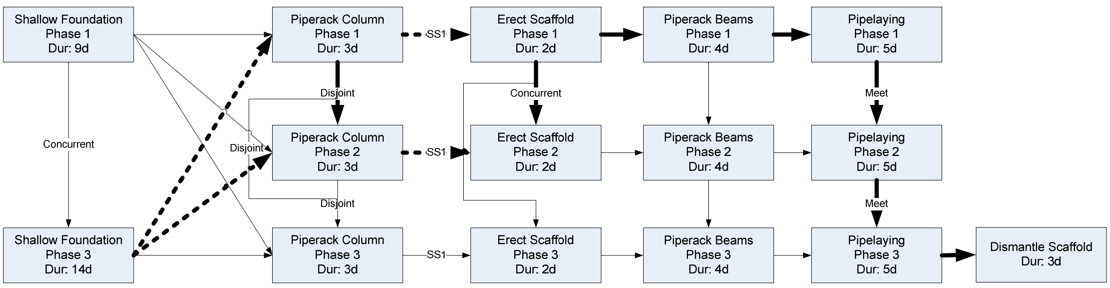


Figure 3: Project Constraint Network

Table 3: Activity Start Times

Activities	Schedule 1 Start Dates	Schedule 2 Start Dates
Shallow Foundation Phase 1	[0 .. 4]	[0 .. 4]
Shallow Foundation Phase 3	0	0
Piperack Column Phase 1	14	17
Piperack Column Phase 2	17	14
Piperack Column Phase 3	[20 .. 26]	[20 .. 26]
Erect Scaffold Phase 1	18	18
Erect Scaffold Phase 2	18	18
Erect Scaffold Phase 3	[21 .. 27]	[21 .. 27]
Piperack Beams Phase 1	20	20
Piperack Beams Phase 2	[24 .. 25]	[24 .. 25]
Piperack Beams Phase 3	[28 .. 29]	[28 .. 29]
Pipelaying Phase 1	24	24
Pipelaying Phase 2	29	29
Pipelaying Phase 3	34	34
Dismantle Scaffold	39	39

From the constraints analysis, we may draw some interesting conclusions. Firstly, management of the subcontractor for piperack column installation is vital especially for the first two phases. Secondly, the concurrency of having to erect scaffolds for phase 1 and 2 constrains the project makespan by imposing additional restrictions to the work sequence. Thirdly, the work continuity of the pipelaying activities also dictates the project makespan. Lastly, the “sub-critical” constraint set allows the project manager greater plan flexibility. For example, if the activity “Piperack Column Phase 1” is delayed, then the alternative schedule may be chosen, with “Piperack Column Phase 2” commencing first.

The above analysis allows the project manager to identify and analyze the critical constraints, as well as to identify the underlying construction requirement which leads to the critical constraint. The alternative schedules identified through solving the PDM++ model through CLP allows project managers to deal with uncertainties in the project schedule.

CONCLUSIONS

The above paper identifies the importance of construction requirements to lean construction. Early identification of construction requirements may lead to reduced rework and better identification of alternative work sequences to increase schedule flexibility. The paper also shows a model which traces the development of construction requirements from client and design considerations, and subsequently models the temporal implications of considering such construction requirements on the schedule.

The temporal implications may be captured by using a semantically more representative modelling model PDM++. In fulfilling the construction requirements, the model may generate multiple alternative schedules. An analysis methodology to identify the criticality of constraints and subsequently, that of requirements is proposed in this paper, which considers the criticality of constraints across multiple schedules. This allows project managers to subsequently manage the critical requirements, and propose alternative schedules when uncertain schedule perturbations occur. An illustrative example is provided in the form of a simplified case study.

Future directions of research will involve more in-depth study of the behaviour of flexible requirements in multiple feasible schedules, as well as the adoption of inference techniques to eliminate redundant requirements. Quantification measures of constraint criticality will also be studied. Additionally, ongoing research to extend the model to handle uncertainty in requirements and activity durations is being carried out.

REFERENCES

Allen, J. F. (1984). "Towards a general theory of action and time." *Artif. Intell.*, 23(2), 123-154.

Ballard, G., and Howell, G. (1998). "Shielding Production: Essential Step in Production Control." *Journal of Construction Engineering and Management*, 124(1), 11-17.

Chua, D. K. H., and Shen, L. J. (2005). "Key Constraints Analysis with Integrated Production Scheduler." *Journal of Construction Engineering and Management*, 131(7), 753-764.

Hajdu, M. (1997). *Network Scheduling Techniques for Construction Project Management*, Springer.

Howell, G. A. (1999). "What is Lean Construction." Seventh Annual Conference of the International Group for Lean Construction (IGLC-7), Berkeley, CA, 1-10.

J. M. Kamara, C. J. Anumba., N. F. O. Evbuomwan. (2000). "Establishing and processing client requirements - a key aspect of concurrent engineering in construction." *Engineering Construction & Architectural Management*, 7(1), 15-28.

Jaafari, A. (1984). "Criticism of CPM for Project Planning Analysis." *Journal of Construction Engineering and Management*, 110(2), 222-233.

Song, Y., and Chua, D. K. H. (2006). "Modeling of Functional Construction Requirements for Constructability Analysis." *Journal of Construction Engineering and Management*, 132(12), 1314-1326.

Song, Y., and Chua, D. K. H. (2007). "Temporal Logic Representation Schema for Intermediate Function." *Journal of Construction Engineering and Management*, 133(4), 277-286.