CAN ALGORITHMS SUPPORT THE SPECIFICATION OF CONSTRUCTION SCHEDULES?

SUBMITTED: January 2006
REVISED: April 2006
PUBLISHED: July 2006 at http://www.itcon.org/2006/39/
EDITOR: P. Katranuschkov

W. Huhnt, Professor Dr.-Ing.
Institute of Civil Engineering, Berlin University of Technology, Germany
email: huhnt@tu-berlin.de

F. Enge, Research Assistant
Institute of Civil Engineering, Berlin University of Technology, Germany
email: felix.enge@tu-berlin.de

SUMMARY: *Construction processes require preparation phases where rules, agreements and arrangements for the execution of the project are discussed and determined. Detailed information is worked out where among others the construction activities are named and follow up charts are determined. In practice, follow up charts are developed that describe the project in different levels of detail. The quality of these follow up charts depends on the experiences of construction managers. The user has to specify the construction activities and their interdependencies. Some of these interdependencies are of technological nature. The technological interdependencies must be considered whereas other interdependencies, for example restrictions in the availability of resources, might be considered. Scheduling tools support construction managers with a wide range of functionalities, e.g. the critical path method (CPM) or load balancing algorithms. However, these tools do not support the user specifying interdependencies in such a way that completeness and correctness of interdependencies can be guaranteed. They document user input, and completeness and logical correctness of follow up charts can only be checked by inspection. This paper presents a modeling technique where the technological interdependencies between construction activities are treated as results of algorithms. Each construction activity is modeled individually and independently of other activities by describing its preconditions and its results. The interdependencies between activities are calculated from the activity-oriented specification based on relational algebra. A proposal for the order of activities is computed. However, rescheduling is still necessary to consider additional restrictions. The advantage in using the presented modeling technique is that logical correctness and completeness of the proposal can be guaranteed so that the quality of scheduling improves. A practical example is presented to illustrate how the modeling technique works and what results can be calculated to support work that is at present time specification work of construction managers. This paper describes the modeling technique in a context where specifically the use of this technique for construction scheduling is explained. Some more theoretical aspects have already been published (Huhnt, 2005). The approach presented in this paper is a new approach in the area of semi-automate approaches to generate construction schedules. Differences to existing approaches and advantages in contrast to existing approaches are described.*

KEYWORDS: *Construction processes, process modeling, construction scheduling.*

1. INTRODUCTION

Construction processes in civil engineering are characterized by specific peculiarities. The list of these peculiarities is long and covers the individuality of each project, different clients, different requirements, different project participants, different time frames, different budgets, and so on. An extensive number of construction methods have been developed and lots of building materials can be selected, and clients often prefer individual designs, individual combinations of building materials, individual construction methods, etc. This results in unforeseeable combinations of construction activities so that it is impossible to specify an overall construction model that is valid for and can be adapted to each project. Exceptions can be found in the area of prefabricated buildings where providers restrict choices so that the number of possible combinations is reduced and known.

As a consequence of their properties, construction projects need to be prepared individually. In civil engineering, work plans or project manuals document agreements and rules for the execution of projects. These work plans and manuals cover descriptions of activities, responsibilities, deadlines, budgets, as well as interdependencies between activities specifying target values for the execution of construction processes (DBV, 1998). Especially follow up charts need to be worked out individually. They are part of a work plan or a project manual. A follow up chart can be defined by a set of construction activities and a set of relations between activities. These relations describe that a specific activity has to be executed before another activity. Four different types of relations are differentiated in network planning technique: finish-start, finish-finish, start-start, and start-finish. Activities and relations can be weighted by their duration. Of course, a lot of additional information is necessary for scheduling like information about human resources that can be assigned to activities, information about construction equipment, logistic information e.g. information about the flow of material, or costs. However, the core of a time schedule or a follow up chart consists of a set of activities and a set of relations between activities. These activities and their relations describe how and in which order the activities shall be executed.

This definition of a follow up chart is close to definitions of process models. A process model used in the context of business process modeling can be regarded as a description of what a specific process will or shall look like. The elements of a process are, at the simplest level, activities and relations between them. A specific modeling technique provides types of process elements like events or functions and types of relations like an event that occurs after the execution of a specific function. The resulting structure can be mapped onto graphs so that a theoretical background is available. This is also valid for follow up charts so that for both, process modeling and scheduling, the same theoretical background can be used.

The theoretical background, namely graph theory, is already widely used in civil engineering projects. For instance, efficient algorithms have been developed as part of graph theory applying the critical-path method (CPM) to calculate early and late activity start and finish points in time, and floats. Optimization methods are available and used to balance the utilization of construction equipment. However, the execution of all these algorithms requires a specified process model or a specified follow up chart. Specifically the technological interdependencies between the activities are required. Once the technological interdependencies are given, algorithms can assist to reschedule where further restrictions like the availability of specific equipment are considered.

This paper discusses an approach where the theoretical background is used in the area of construction scheduling one step earlier than today’s practice. The technological interdependencies between construction activities are regarded as a result of algorithms and not as a result of human thinking only. A specific modeling technique is presented for the specification of construction activities where each activity can be described independently of other activities. Based on this specification, the technological interdependencies between the activities are computed. This reduces the effort of specification. A project manager or a project team can focus on the specification of each construction activity independently of other activities. Correctness and completeness of the calculated interdependencies can be guaranteed with respect to user input. A proposal for a follow up chart is generated. This proposal is the starting point for further tasks in scheduling where all well known and practice proven methods can be used to develop an optimal process for the specific project.

The modeling technique presented in this paper is illustrated by a small example. Results are presented that have been calculated by a pilot implementation. The pilot implementation exports its results to Microsoft® Project so that a proposal for a follow up chart is available in an existing scheduling tool. However, further converters can be implemented so that the presented modeling technique including the available pilot implementation can be regarded as a pre-processor technique in construction scheduling independently of specific tools that are already used for scheduling in civil engineering.

In practice, the specification of the technological interdependencies is up to project managers or project teams. In general, this specification is extensive. Consistency and correctness cannot be checked automatically, but the correctness of these interdependencies is a prerequisite in scheduling so that follow up charts need to be validated by experts. This requires an extensive effort. In addition, subsequent modifications are necessary. Specifically in construction projects where clients are allowed to modify requirements during the project execution phases, a lot of effort is put in the specification and the validation of follow up charts.

As the manual creation of a schedule is an error-prone and time-consuming process research approaches exist to automate this process. Knowledge-Based Expert Systems are some of these approaches. These systems are computer programs that incorporate human expertise represented as knowledge source entities to participate in the problem solving process. Three essential parts are typical for a Knowledge-Based Expert System: (1) The context; (2) the interference mechanism; and (3) the knowledge source. The context contains information specific to a project and generated information. The knowledge source contains human expertise. It is a set of project independent well distinguishable entities of condition-action pairs. If the condition of a knowledge source is satisfied its action is enqueued to be performed at a specific point in time. The mechanism managing the actions to be performed is called interference mechanism. Each activity operates on the context and puts its results into the context.

Hendrickson et al. (1987) introduced CONSTRUCTION PLANEX which is a Knowledge-Based Expert System that automatically generates a schedule taking into account construction methods, resource information as well as further factors. It generates activities, precedence relationships between the activities, estimates durations and costs.

Navinchandra et al. (1988) developed GHOST, a generator of hierarchical networks for construction projects. In contrast to other systems, the knowledge base is not used to build the network but to criticize it. Starting from a network with all activities in parallel each knowledge source is used to criticize the optimistic, but probably infeasible network, to create precedence relationships that finally lead to a feasible network.

Winstanley (et al. 1993, 1995) describes OARPLAN, a project planning system which uses declarative knowledge of object-action pairs to identify activities and object relationships for activity sequencing. The general concept of the system is to determine and sequence in a set of activities derived from a CAD-model.

Some research focused on reusing experience from past projects to build a schedule for a new project. Dzeng and Tommelein (1993, 1997) introduced CasePlan, a case-based reasoning system. The knowledge-base for a case-based reasoning system comprises a set of cases and a mechanism for retrieving cases and adopting their solutions to suit the new project. A case-based reasoning system thus applies three phases: (1) Identifying and retrieving useful cases; (2) adapting the retrieved case to the new project; and (3) classifying and archiving new project approaches as cases. CasePlan matches the product model of archived cases to the product model of a considered project and generates an appropriate construction schedule.

Fischer et al. (1996) and Aalami et al. (1998) focused on formalizing the semantics of relations between activities in schedules. The aim was to develop a computer-interpretable Construction Method Model Template (CMMT) i.e. abstracted skeletal plans to represent planning knowledge, and resource models to formalize the assumptions of planners, so that planners can easily develop schedules and schedule alternatives from a CAD drawing.

During the planning process, detailed construction activities are generated either for a component defined in a project description, or for an activity already existing in a project plan. Depending on the planning principle these approaches can be referred to as a component-based planning approach or an activity-based planning approach. CONSTRUCTION PLANEX and GHOST are component-based planning approaches while OARPLAN represents an activity-based planning approach. CasePlan comprises both approaches.

The approaches published in literature achieved partly excellent results. However, they have not influenced the construction industry in such a way that the use of algorithms to “specify” construction schedules is state of the art. Component-based approaches have the disadvantage that the level of detail concerning the activities is restricted to a single activity for each component. This level of detail is not suitable for each construction project. Case-based approaches and knowledge-based expert systems where existing knowledge is used for the generation of construction activities and their interdependencies work pretty well if no additional knowledge is necessary for a new project. Of course, case-based systems and knowledge-based systems can be expanded and additional knowledge can be stored in theses systems. However, this is only shifting the complexity from the project to the administration of the knowledge base.

The approach presented in this paper is not focused on the aim to generate a complete project plan automatically. The authors of this paper think that such an approach would not work for construction processes. Construction processes are individual, and as a consequence of this individuality, a knowledge base covering the complete knowledge that is necessary to generate construction schedules cannot be set up. New building technologies or new building materials always require the representation of additional knowledge. Therefore, the authors of this paper accept that the project manager is necessary for specific input. The approach presented tries to reduce the complexity of the required input in such a way that the specification of this input does not require an overall view on all construction activities or even knowledge on any other activity. Each construction activity is specified independently of other activities by a construction manager. The interdependencies between the activities are computed. These interdependencies are a result. Therefore, the overall context of the construction process is computed. This enables an efficient "specification" of construction activities because algorithms are used to compute the interdependencies between the activities.

2. COMPONENTS AND STATES

The approach presented in this paper requires a description of the building that has to be constructed, rebuild or torn down. The building has to be decomposed into components. Components need to be named. Types of components need to be introduced. These types of components describe the manufacturing process of the components that are from this type. For this purpose, status variables are used. Thus, the states a component passes during its manufacturing process are described by its type. The type does not address activities; it consists of status variables in a specific order. The order describes the manufacturing process for components of this type in such a way that only the component itself is considered. So, types of components can be modeled independently of others. Specific descriptions of types of components have been published that already make use of the concept of describing manufacturing processes of components by status variables. These descriptions are used for calculation purpose. This section describes how a building can be decomposed, how types of components can be developed, and how existing descriptions of types of components can be used.

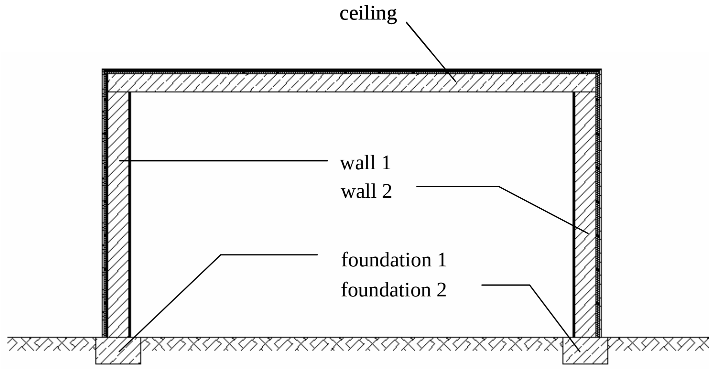


FIG. 1: Components of a building.

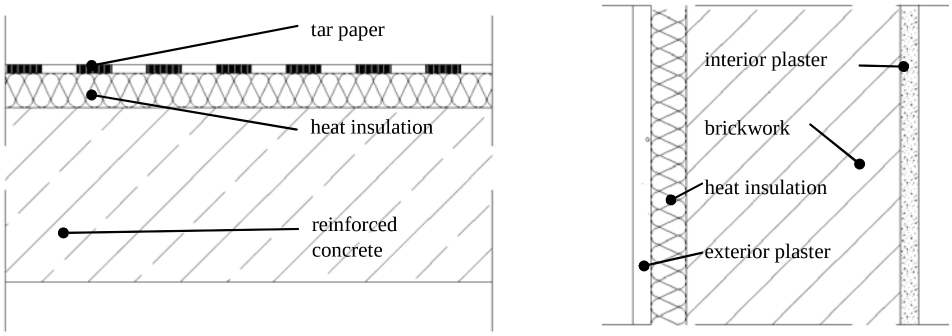


FIG. 2: Detailed constructions of ceiling and walls.

2.1 Components

A component can be defined as a part of a building. The level of detail is not prescribed, e.g., a specific wall can be a component, and also all walls that form a side of a floor can be a component. It is also possible to sum up several parts of an assembly to components. As an example, a wall with a door can form a component whereas also the wall itself and the door that has to build in that wall can be components. The approach presented does not prescribe how components are modeled. However, modeling components is closely connected to modeling types of components as shown in section 2.2.

The building that is used as an example in this paper is shown in Fig. 1. It is modeled in such a way that it consists of five components, two foundations, two walls, and one ceiling. These components are named as shown in Fig. 1.

2.2 Types of component

A component type summarizes those components whose manufacturing process is identical. This is described by exactly the same set of status variables and the same ordering relation in the set of status variables. Assigning a component type to a component allows mapping the manufacturing process modeled for the type to the component. Notice, that a component type does not represent a skeletal plan consisting of activities, instead the manufacturing process is captured as consequence of status values and their ordering relation.

Ordering relations are necessary to structure a set of status variables, determining the order in the manufacturing process of a component, when a specific status is reached. The simplest ordering relation is a sequence. For example, producing a foundation is a sequential process, where (1) formwork has to be placed first; (2) the second step is placing reinforcement; (3) the third step is placing concrete; (4) in a fourth step concrete has to set; (5) and in a fifth step formwork has to be removed. Five status variables can be specified to describe this sequential process: formwork removed, reinforcement placed, concrete placed, concrete set, formwork removed. The ordering relation of these five status variables is shown in Fig. 3. The ordering relation is strict. A suitable data structure to store the sequential status variables is a list.

In general, the manufacturing process of a component is not sequential. A component has a body consisting of wood, bricks, or concrete and different layers might be necessary on each surface. Examples are shown in Fig. 2, where the detailed constructions of the walls and the ceiling are shown. The manufacturing processes for such components splits after the body has been produced. Fig. 4 and Fig. 5 show the status variables and the ordering relations. A suitable data structure to store such status variables is a tree. A predecessor-successor-relation can be derived for status variables if they lie on the same path. If they lie on different paths, no statement can be made concerning their order. The underlying ordering relation is a partial strict ordering relation.

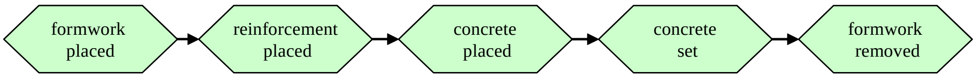


FIG. 3: Status variables for foundations.

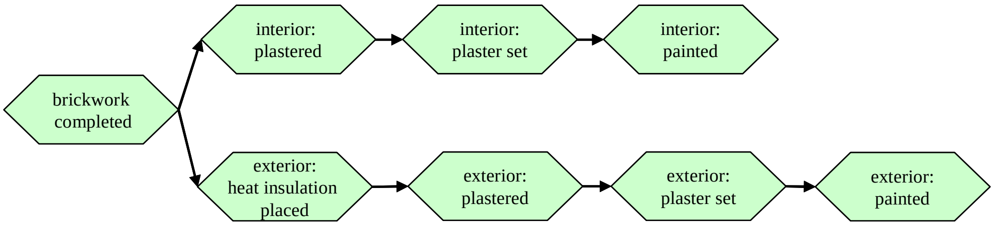


FIG. 4: Status variables for walls.

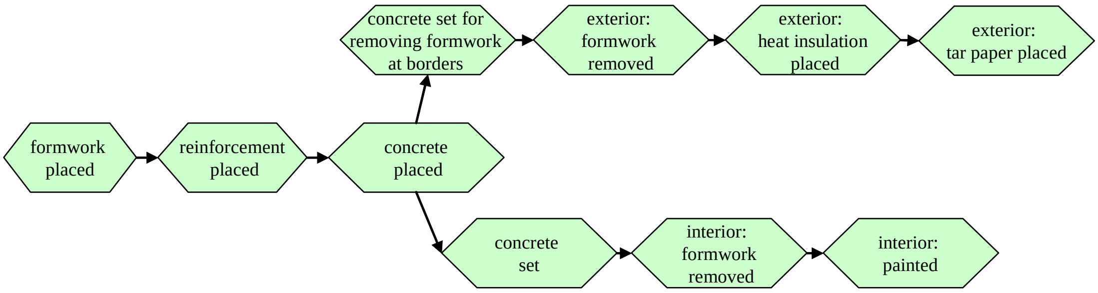


FIG. 5: Status variables for ceilings.

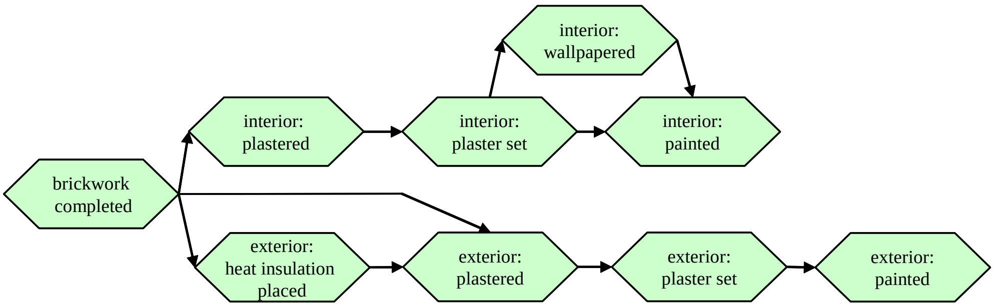


FIG. 6: Status variables for walls with variants

If the aim is to avoid the specification of different component types for slightly differing components, an acyclic graph is a suitable data structure to store the set of status variables. Fig. 6 shows a component type that is slightly different compared to the type shown in Fig. 4. The component type shown in Fig. 6 includes wallpapers on the interior surface as an additional possibility so that the interior surface can be painted, wallpapered, or wallpapered and painted afterwards. In addition, a relation between “brickwork completed” and “interior: plastered” is specified so that placing heat insulation can be regarded as optional. The property of the ordering relation is partial strict. In a graph, different paths can be joined. Thus, the use of graphs is not restricted to slightly differing component types with optional status values. Graphs are necessary to model production processes where more than one precondition is required before a subsequent state can be reached. If, for instance, a framework has to be painted, a graph would be a suitable data structure to describe that the framework itself and for instance the brickwork for closing the space between columns and bars need to be produced and placed before the complete framework can be painted.

The introduction of types of components is a suitable approach in civil engineering to encapsulate the knowledge of construction methods. It is possible to predefine project independent generalized templates that can be reused to describe the manufacturing process of each component occurrence, representing an instance of a specific component type. Assigning a component type to a project specific component, as shown in Fig.7, imposes a set of status variables that needs to be passed through by project specific activities.

Component	**Type of component**
foundation 1	foundation type
foundation 2	foundation type
wall 1	wall type 1
wall 2	wall type 1
ceiling	ceiling type

FIG. 7: Assigning types to components.

2.3 Existing descriptions of types of components

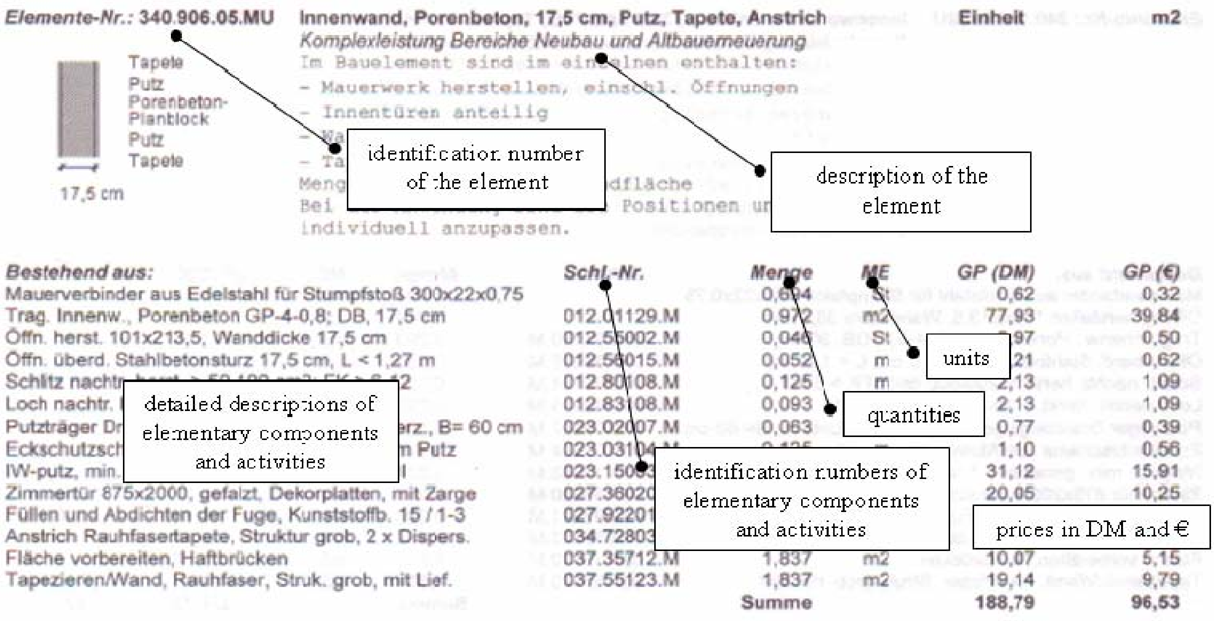
Some research focused on reusing knowledge from previous projects in future projects. Dzeng and Tommelein (1993, 1997) introduced CasePlan, which is a tool to select schedules for reuse. The tool allows a user to interact with, in favor of determining schedule fragments of case schedules and to adapt them to the needs of the future project. As a constraint, this tool addresses more-or-less standardized designs.

Fischer et al. (1996) and Aalami et al. (1998) use computer-interpretable construction method models, i.e. abstracted skeletal plans to represent planning knowledge, and resource models to formalize the assumptions of planners.

In Germany, it is state of the art to use descriptions of types of components for calculation purpose. The German standard (DIN 276, 1990) describes a classification system for costs in building construction. Cost evaluation methods have been developed. They make use of the standardized cost classification system and structure the costs by elements respectively components. Accordingly these methods are called element- or component-methods. As this method assigns costs directly to a component, the structuring depth of DIN 276, consisting of three hierarchy levels, is not always differentiating sufficiently. Approaches have been developed introducing further levels of detail. Schäfer (2002) gives an overview on the different element-methods. To allow coupling of the cost classification, as described in DIN 276, with an execution-phase-oriented classification, as used for the descriptions of bills of quantities, Sommer (1994) and Schmitz et al. (1995) have added two further classification levels.

Mittag (2000) developed an extensive database that takes the enlarged classification hierarchy into account, offering element descriptions including a cost based weighting, to provide data for the element-method. Each dataset is a description of a single element or a combination of single elements grouped to a complex element in an execution-phase-oriented manner. The complex elements described by Mittag represent components in the understanding of the presented approach in this paper. Only little effort is required to derive status values and status graphs from the descriptions of the complex elements. First a description needs to be analyzed, mapping characteristic status information onto state variables, and than an ordering relation in the set of status variables needs to be set up. It is thus possible to take advantage of a vast expert knowledge base, determining project independent component types with a common praxis level of detail that is comparable for all types derived from the same classification system.

Fig. 8 shows an example of a type description by Mittag. The different layers of the wall are named. Each layer is described in detail. Quantities are given, and a price for each detail is given as well. The prices given by Mittag are average prices from the construction market in Germany. These prices can only be regarded as indications because they might vary from region to region and project to project. However, construction companies in Germany maintain such data so that an excellent basis is available for calculation purpose that can more generally be used for the description of the manufacturing process as described above.



Elemente-Nr.: 340.906.05.MU **Innenwand, Porenbeton, 17,5 cm, Putz, Tapete, Anstrich** **Einheit** **m2**

Komplexleistung Bereiche Neubau und Altbauerneuerung

Im Bauelement sind im einzelnen enthalten:
- Mauerwerk herstellen, einschl. Öffnungen
- Innentüren anteilig
- Wa
- Ta
Meng dfläche
Bei Positionen un
individuell anzupassen.

Bestehend aus:	*Schl.-Nr.*	*Menge*	*ME*	*GP (DM)*	*GP (€)*
Mauerverbinder aus Edelstahl für Stumpfstoß 300x22x0,75		0,694	St	0,62	0,32
Trag. Innenw., Porenbeton GP-4-0,8; DB, 17,5 cm	012.01129.M	0,972	m2	77,93	39,84
Öffn. herst. 101x213,5, Wanddicke 17,5 cm	012.55002.M	0,046	St	[illegible]	0,50
Öffn. überd. Stahlbetonsturz 17,5 cm, L < 1,27 m	012.56015.M	0,052	m	[illegible]	0,62
Schlitz nachtr. [illegible]	012.80108.M	0,125	m	2,13	1,09
Loch nachtr. [illegible]	012.83108.M	0,093	[illegible]	2,13	1,09
Putzträger Dr [illegible] erz., B= 60 cm	023.02007.M	0,063	[illegible]	0,77	0,39
Eckschutzsch [illegible] n Putz	023.03104.M	[illegible]	[illegible]	1,10	0,56
IW-putz, min. [illegible]	023.150[illegible]	[illegible]	[illegible]	31,12	15,91
Zimmertür 875x2000, gefalzt, Dekorplatten, mit Zarge	027.3602[illegible]	[illegible]	[illegible]	20,05	10,25
Füllen und Abdichten der Fuge, Kunststoffb. 15 / 1-3	027.92201[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
Anstrich Rauhfasertapete, Struktur grob, 2 x Dispers.	034.72803[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
Fläche vorbereiten, Haftbrücken	037.35712.M	1,837	m2	10,07	5,15
Tapezieren/Wand, Rauhfaser, Struk. grob, mit Lief.	037.55123.M	1,837	m2	19,14	9,79
		Summe		**188,79**	**96,53**

FIG. 8: Existing description of a component type (Mittag, 2000).

2.4 Weighting components and types of components

Each component of a building can be weighted by the price that is necessary for its production. In general, this price is estimated in calculations. As described in section 2.3, so-called element-methods can be used so that the price of each component of a building can be determined. Percentage values can be specified for each status variable in the manufacturing process of a component. These values describe the portion of the complete price that is necessary to reach the specific status. Fig. 9 shows the prices of the components of the example shown in Fig. 1. Fig. 10 shows the percentage values that are specified for the status variables of the foundations, the walls, and the ceiling.

Component	Price
foundation 1	680,- €
foundation 2	680,- €
wall 1	3.650,- €
wall 2	3.650,- €
ceiling	7.300,- €

FIG. 9: Prices of components.

Status values of foundation type	Weight
formwork placed	0,15
reinforcement placed	0,55
concrete placed	0,26
concrete set	0,00
formwork removed	0,04

Status values of wall type 1	Weight
brickwork completed	0,31
interior: plastered	0,05
interior: plaster set	0,00
interior: painted	0,04
exterior: heat insulation placed	0,22
exterior: plastered	0,33
exterior: plaster set	0,00
exterior: painted	0,05

Status values of ceiling type	Weight
formwork placed	0,15
reinforcement placed	0,41
concrete placed	0,13
concrete set for removing formwork at borders	0,00
exterior: formwork removed	0,01
exterior: heat insulation placed	0,11
exterior: tar paper placed	0,10
concrete set	0,00
interior: formwork removed	0,04
interior: painted	0,05

FIG. 10: Percentage values of status variables.

3. CONSTRUCTION ACTIVITIES

The complete construction work of a project has to be subdivided into construction activities. This has to be done by a construction manager or a project team. The level of detail is not prescribed by the approach presented in this paper. However, an equivalent level of detail should be chosen with respect to the status values that have been chosen to model the manufacturing process of the components. Fig. 11 shows the construction activities to produce the structure shown in Fig. 1. The level of detail that has been chosen is coordinated with the level of detail for the manufacturing processes of the components as shown in Fig. 3, Fig. 4, and Fig. 5.

The approach presented in this paper is based on the assumption that construction activities can be described by their results and their prerequisites. Further information like craftsman, construction equipment or others is not considered as part of the description of construction activities in this paper.

The result of a construction activity is one or more components in specific states so that the activity itself consists of the work required to achieve the specified status values for those components. Fig. 12 shows an example of a construction activity. The activity “foundation 1: concrete placing” has foundation 1 in status “concrete placed” as its result. As described above, the level of detail chosen is up to the user. It is also possible to specify a single activity for both foundations for concrete placing. However, once the level of detail has been chosen it can only be detailed with a lot of effort. For example, the construction activities that are modeled as described appear in the same level of detail in the follow up chart. Subsequent detailing in the follow up chart requires a lot of additional effort whereas summing up in the follow up chart can be done easily.

Prerequisites of a construction activity can be components in specific states. Prerequisites need to be specified if the execution of a construction activity requires components in specific states as a precondition for the work that has to be executed. Prerequisites of components that are results of the same construction activity need not be specified. For example, the construction activity “foundation 1: concrete placing” as shown in Fig. 12 requires foundation 1 in state “formwork placed” only. This need not be specified. Section 4 explains that this information can be calculated. In Fig. 13, the construction activity “ceiling: formwork placing” is shown. This activity requires the brickwork of both walls being completed. These preconditions are specified by the names of the components and their required states.

Construction activities
ceiling: concrete placing
ceiling: concrete setting for formwork at borders removing
ceiling: concrete setting for formwork removing
ceiling: formwork placing
ceiling: inside formwork removing
ceiling: inside painting
ceiling: outside formwork removing
ceiling: outside heat insulation placing
ceiling: outside tar paper placing
ceiling: reinforcement placing
foundation 1: concrete placing
foundation 1: concrete setting
foundation 1: formwork placing
foundation 1: formwork removing
foundation 1: reinforcement placing
foundation 2: concrete placing
…
wall 1: brickwork
wall 1: inside painting
wall 1: inside plaster setting
wall 1: inside plastering
wall 1: outside heat insulation placing
wall 1: outside painting
wall 1: outside plaster setting
wall 1: outside plastering
wall 2: brickwork
…

FIG. 11: Construction activities for the production of the structure shown in Fig. 1.

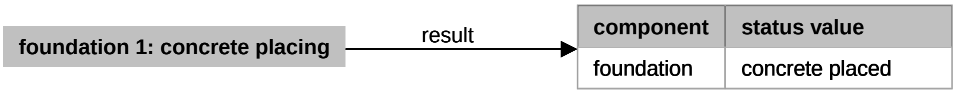


FIG. 12: Construction activity "foundation 1: concrete placing".

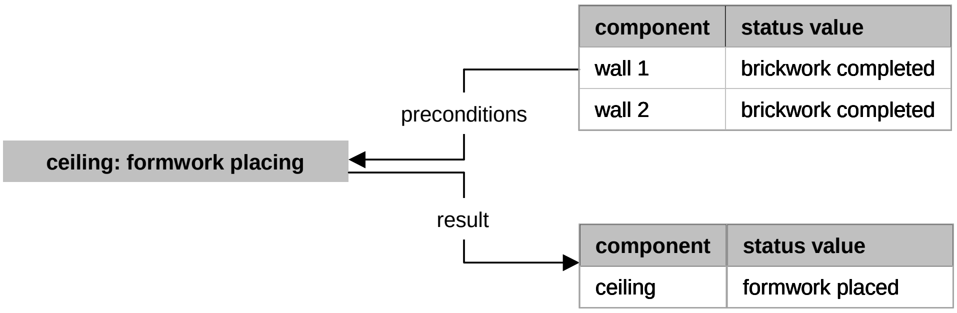


FIG. 13: Construction activity "ceiling: formwork placing".

The presented modeling technique allows specifying the construction process in a activity-oriented way. Each activity needs to be named and two types of relations need to be specified as described. This allows focusing on each activity independently of other activities. The overall process does not need to be known when each construction activity is worked out. Only components-status tuples need to be specified that represent either a precondition for the execution of an activity or a result of the execution of an activity. This is an advantage. The user can focus on each activity independently. The overall context will be computed as describe in sections 4 and 5.

4. COMPUTING RELATIONS

In a first step, relations between activities are calculated based on the specified relations between activities and components in specific states. Two rules are evaluated for this purpose:

1. Consider an activity a that has a component c_p in state s_p as a prerequisite: All activities that have the component c_p in state s_i as a result where s_i is less or equal s_p have to be executed before activity a.
2. Consider an activity a that has a component c_r in state s_r as a result: All activities that have the component c_r in state s_i as a result where s_i is less than s_r have to be executed before activity a.

Based on these two rules, relations between activities are calculated. Each relation represents a dependency between two activities, saying that the one has to be executed before the other one. Rule one allows conjoining the manufacturing process of different components. Rule two ensures that the manufacturing process of a considered component has reached a specific state. All activities, modifying the component's state without reaching the specified status value, have to be executed before. Due to this rule, no prerequisites concerning a component that is a result need to be modeled. They are calculated based on the specified component types.

The result of the computation is a specified graph, consisting of activities representing the set of nodes and the calculated relations representing the set of edges.

The advantage of calculating interdependencies between activities is that completeness can be guaranteed. Once a component in a specific state has been specified as a precondition, all activities that are necessary to achieve this state are considered to be executed before the actual activity can start. This is also true to the results of an activity. Especially if subsequent modifications are necessary, only those activities need to be reviewed that are directly influenced by the modification. The overall context is calculated, and correctness and completeness can be guaranteed if user input has been specified in a correct way.

For the calculation of relations between activities it is functional to identify activities according to their results. A result is identified by a component-state tuple. A map is a suitable data structure to store the relation of a result to the activity that produces that result. To set up the map, all activities need to be traversed and for each result of an activity an entry in the map needs to be added where the key consists of the component-state tuple and the value is the activity.

The algorithm calculating the set of dependencies represents a loop over the set of activities, which comprises two blocks. In each cycle of the loop an activity is identified and denoted by pivot. The first block represents a nested loop over all preconditions of pivot; the second block represents a nested loop over all results of pivot. For each precondition of pivot represented as component-status tuple it is checked whether there is an entry in the above mentioned map. If this is the case, an activity is identified that needs to be performed before pivot. A precedence relation will be created between the identified activity and pivot. The procedure concerning a result of pivot differs slightly. Before looking up the component-status tuple in the map it is necessary to determine the predecessor component-status tuple to the actual result. If there is an entry in the map for the predecessor component-status tuple, the corresponding activity needs to be performed before pivot. Analogously a precedence relation will be created between the identified activity and pivot. The complexity of the algorithm is $O(|a|*(|a_p|+|a_r|))$, where $|a|$ is the number of activities, $|a_p|$ is the average number of preconditions of an activity, and $|a_r|$ is the average number of results of an activity. The complexity is dominated by the number of activities $|a|$, as it is significantly greater than $|a_p|+|a_r|$.

5. SORTING ACTIVITIES

The relation between the activities can be used to sort the set of activities topologically if they do not cover a cycle. Cycles can occur if an activity a_i requires a component c_m in state s_k and has a component c_n in state s_l as output, whereas another activity a_j requires c_n in state s_l and has c_m in state s_k as output. Such a situation is shown in Fig. 14. These conflicts can be solved if the affected activities are executed in parallel. The affected activities that have to be executed in parallel can be replaced by a major activity so that cycles can be avoided.

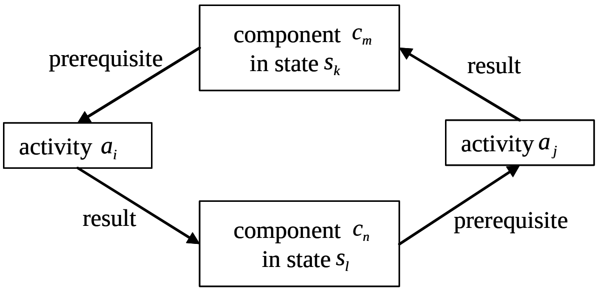
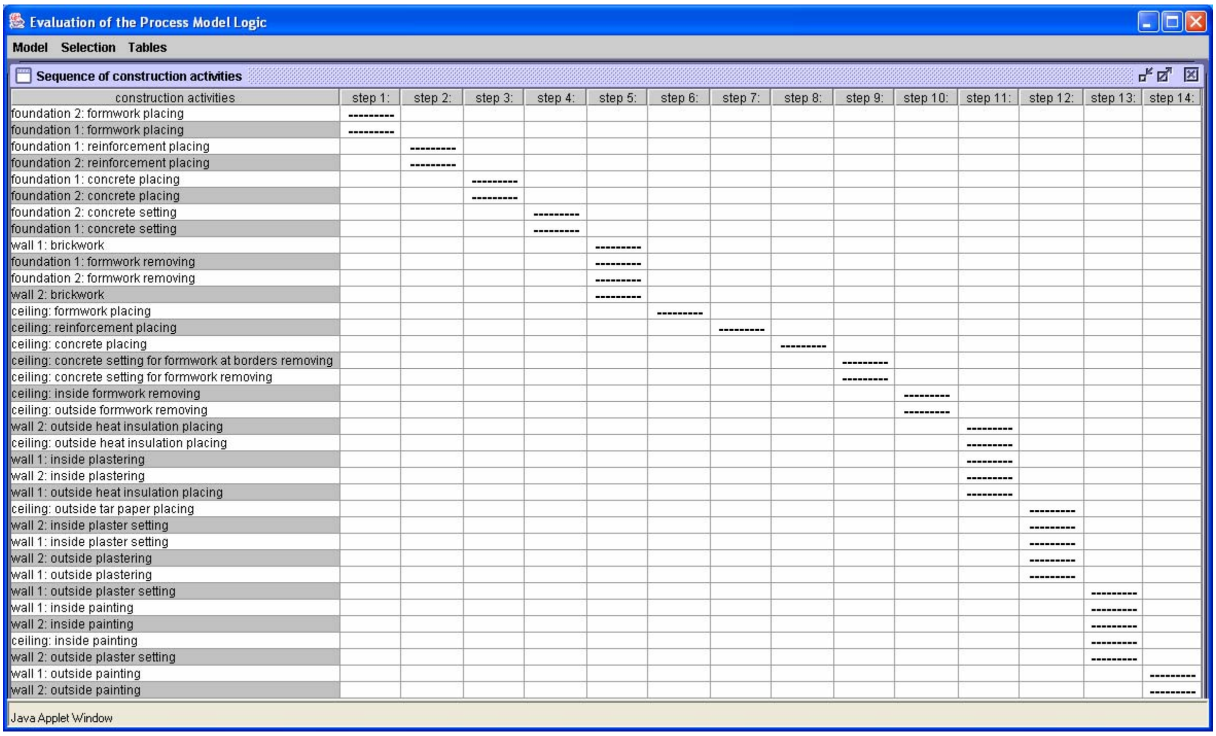


FIG. 14: Cycle.

In general, several solutions for an order of activities are valid where all relations between activities are considered. A topological sort algorithm based on the breadth-first-search is chosen. Initially the set of nodes without predecessors is determined. These nodes are assigned to the actual logical step one. All successors need to be performed at least one step later. For each node the before mentioned set of successors is determined and accordingly assigned to the following step. Once all nodes of the first step are treated, they are marked as such, and the next cycle of the algorithm begins. First the actual logical step is incremented. Then the set of nodes without predecessors is determined, whereas nodes marked as treated are not considered any more. These nodes are assigned to the actual step; their set of successors is determined and assigned to the subsequent step. This procedure is repeated until no further node is unmarked. If a node is assigned to a step repeatedly the later step is decisive. Since in the worst case breadth-first search has to consider all paths to all possible nodes, the complexity of breadth-first-search is $O(|V| + |E|)$, where $|V|$ is the number of nodes, and $|E|$ the number of edges in the graph.

In general, several orders of activities are valid for a process where specific relations between activities are given. All these orders form a solution set. The algorithm described above results in a solution for the order of activities where the solution has the lowest number of logical steps, and each activity is inserted at the earliest logical step when it can be executed (Pahl and Damrath, 2001, Turau, 1996). The computed solution needs to be evaluated by a construction manager. The construction manager must be able to select another solution from the solution set. The approach presented in this paper is focused on technological interdependencies between construction activities only. Additional circumstances, e.g. the availability of resources might influence the user to select another solution from the solution set.

Fig. 15 shows the solution for the construction sequence of the building introduced in Fig. 1. This solution has been calculated based on the breadth-first-search. It has not jet been edited and modified. Only technological interdependencies between activities are considered. Further interdependencies might exist, e.g. the availability of special equipment does not allow the execution of specific activities in parallel. Such interdependencies are not yet considered so that the calculated order of activities can only be regarded as a proposal. Scheduling is necessary. However, the calculated proposal can be used as input for scheduling as described in section 7.



construction activities	step 1:	step 2:	step 3:	step 4:	step 5:	step 6:	step 7:	step 8:	step 9:	step 10:	step 11:	step 12:	step 13:	step 14:
foundation 2: formwork placing	---------													
foundation 1: formwork placing	---------													
foundation 1: reinforcement placing		---------												
foundation 2: reinforcement placing		---------												
foundation 1: concrete placing			---------											
foundation 2: concrete placing			---------											
foundation 2: concrete setting				---------										
foundation 1: concrete setting				---------										
wall 1: brickwork					---------									
foundation 1: formwork removing					---------									
foundation 2: formwork removing					---------									
wall 2: brickwork					---------									
ceiling: formwork placing						---------								
ceiling: reinforcement placing							---------							
ceiling: concrete placing								---------						
ceiling: concrete setting for formwork at borders removing									---------					
ceiling: concrete setting for formwork removing									---------					
ceiling: inside formwork removing										---------				
ceiling: outside formwork removing										---------				
wall 2: outside heat insulation placing											---------			
ceiling: outside heat insulation placing											---------			
wall 1: inside plastering											---------			
wall 2: inside plastering											---------			
wall 1: outside heat insulation placing											---------			
ceiling: outside tar paper placing												---------		
wall 2: inside plaster setting												---------		
wall 1: inside plaster setting												---------		
wall 2: outside plastering												---------		
wall 1: outside plastering												---------		
wall 1: outside plaster setting													---------	
wall 1: inside painting													---------	
wall 2: inside painting													---------	
ceiling: inside painting													---------	
wall 2: outside plaster setting													---------	
wall 1: outside painting														---------
wall 2: outside painting														---------

FIG. 15: Order of activities.

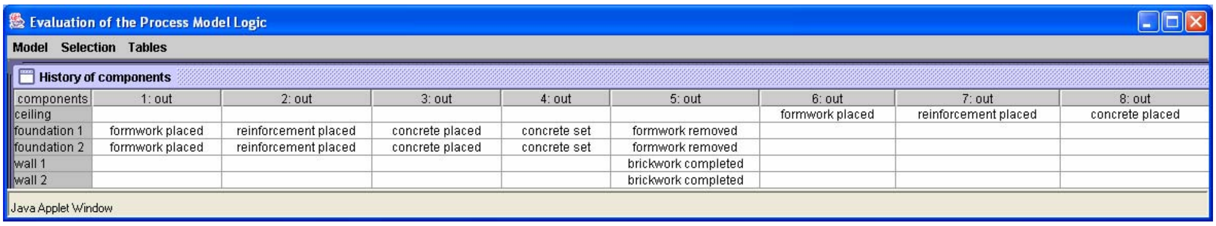
6. QUALITY, TIME, AND MONEY

The calculated order of activities can be evaluated, and it can be weighted. Evaluation of the order of activities gives information about the production of the components. Activities modify the components. The status variables can be regarded as a description of the quality of components. This history can be computed. Components are already weighted by costs so that the history of costs can also be calculated. In addition, the order of activities can be weighted by deadlines so that information about time can be computed as well. The presented modeling technique gives therefore information about the most relevant categories in management at a stage where real scheduling has not jet been started: quality, time, and money.

6.1 Quality

Fig. 16 shows an extract of the history of the components of the building shown in Fig. 1. The order of activities assigns activities to project steps. The specified relation between activities and their results is evaluated. The result is an assignment of the status variables of each component to a specific project step when that component reaches that status.

The calculated history of components can be checked. Each component has to pass through its complete manufacturing process during the execution of the project. It can happen that a specific component reaches two states in a specific project step. For example, a specific wall might be painted on both sides in the same project step. However, the modeling technique guarantees that this only happens if the status variables are not on the same path in the manufacturing process of that component. Missing status variables of components can be calculated so that the user gets information about the completeness of his input.

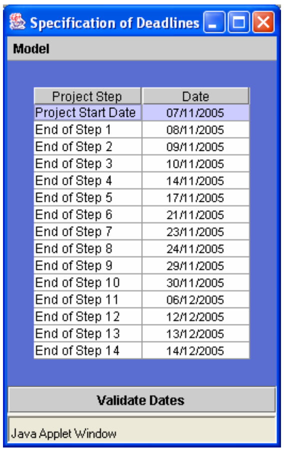


components	1: out	2: out	3: out	4: out	5: out	6: out	7: out	8: out
ceiling						formwork placed	reinforcement placed	concrete placed
foundation 1	formwork placed	reinforcement placed	concrete placed	concrete set	formwork removed			
foundation 2	formwork placed	reinforcement placed	concrete placed	concrete set	formwork removed			
wall 1					brickwork completed			
wall 2					brickwork completed			

FIG. 16: History of components (extract).

6.2 Time

The project steps can be weighted by deadlines. In addition, the project start has to be specified. These specifications allow mapping the order of activities and all other evaluations onto a time scale. Fig. 17 shows the project start and deadlines for the construction of the building shown in Fig. 1. Time frames can be derived for each activity in a project step if start and finish of the project step are transferred to the associated activity. However, the time frame describes not the duration of an activity. The determination of durations might be necessary. Methods have been published in literature. The presented approach does not replace these methods. They might be used as part of further scheduling activities as described in section 7.

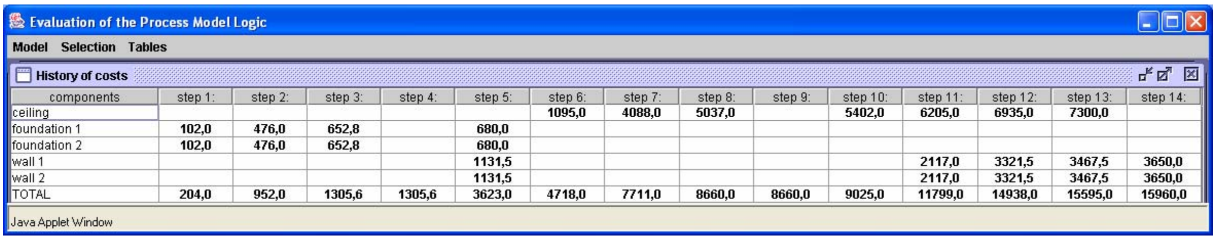


Project Step	Date
Project Start Date	07/11/2005
End of Step 1	08/11/2005
End of Step 2	09/11/2005
End of Step 3	10/11/2005
End of Step 4	14/11/2005
End of Step 5	17/11/2005
End of Step 6	21/11/2005
End of Step 7	23/11/2005
End of Step 8	24/11/2005
End of Step 9	29/11/2005
End of Step 10	30/11/2005
End of Step 11	06/12/2005
End of Step 12	12/12/2005
End of Step 13	13/12/2005
End of Step 14	14/12/2005

FIG. 17: Project start and deadlines.

6.3 Costs

Evaluating the status values of components and the costs that are assigned to each component provides an overview on the history of costs. For each status value, a percentage rate has been specified as part of the manufacturing process. This percentage rate can be multiplied by the costs of the component. All costs of a project step are added so that the history of costs can be calculated for the whole project. Fig. 18 shows the history of costs of the project shown in Fig. 1. From this information, S-curves can be derived that are of common use in project management to show progresses.



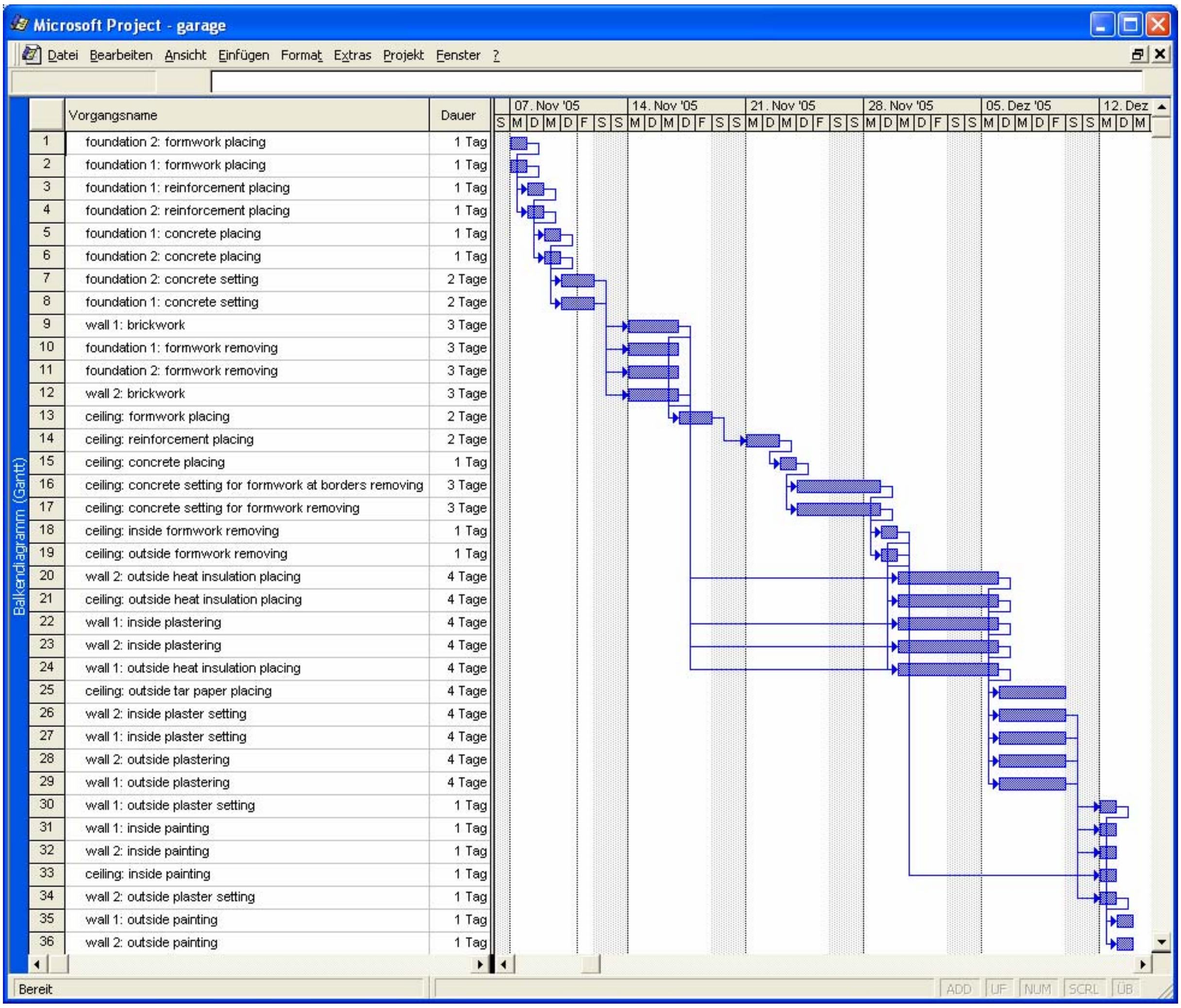
components	step 1:	step 2:	step 3:	step 4:	step 5:	step 6:	step 7:	step 8:	step 9:	step 10:	step 11:	step 12:	step 13:	step 14:
ceiling						1095,0	4088,0	5037,0		5402,0	6205,0	6935,0	7300,0	
foundation 1	102,0	476,0	652,8		680,0									
foundation 2	102,0	476,0	652,8		680,0									
wall 1					1131,5						2117,0	3321,5	3467,5	3650,0
wall 2					1131,5						2117,0	3321,5	3467,5	3650,0
TOTAL	204,0	952,0	1305,6	1305,6	3623,0	4718,0	7711,0	8660,0	8660,0	9025,0	11799,0	14938,0	15595,0	15960,0

FIG. 18: History of costs.

7. SCHEDULING

The output of the algorithms presented in this paper can be used as input for scheduling. Scheduling tools are available at the markets that can be used to develop follow up charts where, beside technological interdependencies, other influences like restrictions in the availability of resources, aspects of financing or the weather are considered. Fig. 19 shows the export of the order of activities as shown in Fig. 15 to Microsoft® Project, mapped onto the time scale by project start and deadlines for the project steps as shown in Fig. 17.

Scheduling tools offer lots of functionalities, and they are able to consider aspects like working days, working hours per working day, or waiting periods. The modeling technique as presented in this paper covers a simple model for the specification of deadlines. The model does not include the specification of durations of construction activities. Activities that have to be executed in the same project step are treated in the same manner. During the import to the scheduling tool, the deadlines of the project steps are used to compute a value for the duration of each activity of that project step, based on 5 working days per week and 8 working hours per day. This value can only be regarded as an approximation of the duration. It needs to be edited. In addition, waiting periods, which are usually modeled in scheduling tools as relations between activities with a specified duration, are treated as normal activities in the presented modeling technique. For example, activities like "foundation 1: concrete setting" would be modeled in tools like Microsoft® Project as an interdependency between activities with a specified duration that describes the waiting period before the succeeding activity can start. They appear now as separate activities so that rescheduling might be required.

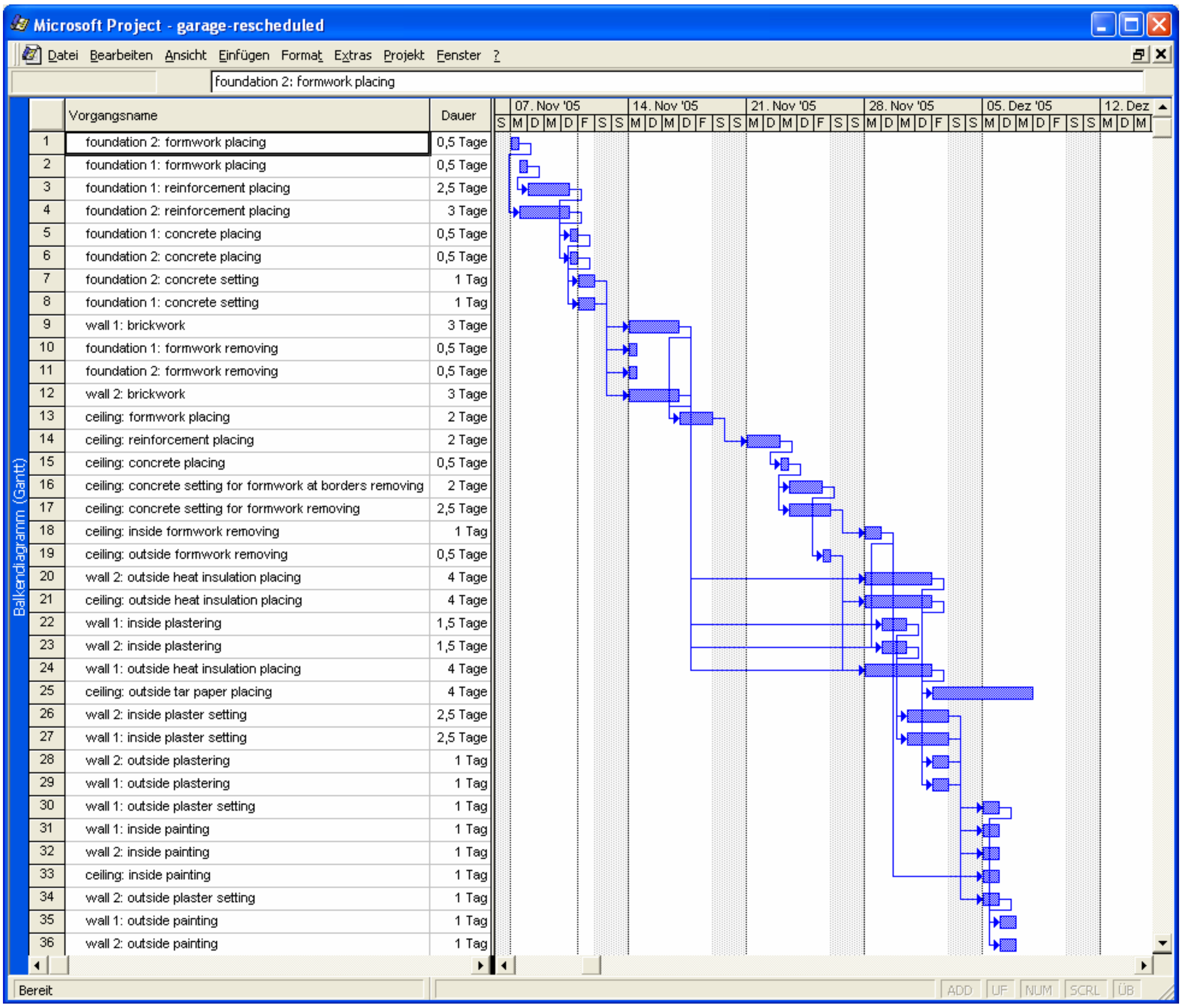


	Vorgangsname	Dauer
1	foundation 2: formwork placing	1 Tag
2	foundation 1: formwork placing	1 Tag
3	foundation 1: reinforcement placing	1 Tag
4	foundation 2: reinforcement placing	1 Tag
5	foundation 1: concrete placing	1 Tag
6	foundation 2: concrete placing	1 Tag
7	foundation 2: concrete setting	2 Tage
8	foundation 1: concrete setting	2 Tage
9	wall 1: brickwork	3 Tage
10	foundation 1: formwork removing	3 Tage
11	foundation 2: formwork removing	3 Tage
12	wall 2: brickwork	3 Tage
13	ceiling: formwork placing	2 Tage
14	ceiling: reinforcement placing	2 Tage
15	ceiling: concrete placing	1 Tag
16	ceiling: concrete setting for formwork at borders removing	3 Tage
17	ceiling: concrete setting for formwork removing	3 Tage
18	ceiling: inside formwork removing	1 Tag
19	ceiling: outside formwork removing	1 Tag
20	wall 2: outside heat insulation placing	4 Tage
21	ceiling: outside heat insulation placing	4 Tage
22	wall 1: inside plastering	4 Tage
23	wall 2: inside plastering	4 Tage
24	wall 1: outside heat insulation placing	4 Tage
25	ceiling: outside tar paper placing	4 Tage
26	wall 2: inside plaster setting	4 Tage
27	wall 1: inside plaster setting	4 Tage
28	wall 2: outside plastering	4 Tage
29	wall 1: outside plastering	4 Tage
30	wall 1: outside plaster setting	1 Tag
31	wall 1: inside painting	1 Tag
32	wall 2: inside painting	1 Tag
33	ceiling: inside painting	1 Tag
34	wall 2: outside plaster setting	1 Tag
35	wall 1: outside painting	1 Tag
36	wall 2: outside painting	1 Tag

FIG. 19: Import of the order of activities to a scheduling tool.

Fig. 20 shows a result of rescheduling where durations have been reworked and some activities have been postponed to avoid peaks in the resources required.

The import to scheduling tools shows the advantage of the modeling technique presented. Scheduling can now start on a basis where the technological interdependencies between activities are complete and correct. Completeness and correctness can be guaranteed. Rescheduling is required to consider additional influences, but rescheduling can now be done much more efficiently, and it is not so much prone to errors as in usual approaches.



foundation 2: formwork placing

	Vorgangsname	Dauer
1	foundation 2: formwork placing	0,5 Tage
2	foundation 1: formwork placing	0,5 Tage
3	foundation 1: reinforcement placing	2,5 Tage
4	foundation 2: reinforcement placing	3 Tage
5	foundation 1: concrete placing	0,5 Tage
6	foundation 2: concrete placing	0,5 Tage
7	foundation 2: concrete setting	1 Tag
8	foundation 1: concrete setting	1 Tag
9	wall 1: brickwork	3 Tage
10	foundation 1: formwork removing	0,5 Tage
11	foundation 2: formwork removing	0,5 Tage
12	wall 2: brickwork	3 Tage
13	ceiling: formwork placing	2 Tage
14	ceiling: reinforcement placing	2 Tage
15	ceiling: concrete placing	0,5 Tage
16	ceiling: concrete setting for formwork at borders removing	2 Tage
17	ceiling: concrete setting for formwork removing	2,5 Tage
18	ceiling: inside formwork removing	1 Tag
19	ceiling: outside formwork removing	0,5 Tage
20	wall 2: outside heat insulation placing	4 Tage
21	ceiling: outside heat insulation placing	4 Tage
22	wall 1: inside plastering	1,5 Tage
23	wall 2: inside plastering	1,5 Tage
24	wall 1: outside heat insulation placing	4 Tage
25	ceiling: outside tar paper placing	4 Tage
26	wall 2: inside plaster setting	2,5 Tage
27	wall 1: inside plaster setting	2,5 Tage
28	wall 2: outside plastering	1 Tag
29	wall 1: outside plastering	1 Tag
30	wall 1: outside plaster setting	1 Tag
31	wall 1: inside painting	1 Tag
32	wall 2: inside painting	1 Tag
33	ceiling: inside painting	1 Tag
34	wall 2: outside plaster setting	1 Tag
35	wall 1: outside painting	1 Tag
36	wall 2: outside painting	1 Tag

FIG. 20: Rescheduled order of activities.

8. ADDITIONAL CONCEPTS

Lots of assumptions have been made in the modeling technique presented, and it is possible to use different assumptions so that additional concepts can be integrated into the modeling technique. Two major additional concepts are discussed in this section to illustrate the potential of the presented modeling technique.

8.1 Basic types and component types

In section 2.2, types of components have been discussed that might be used to describe the manufacturing process of slightly differing components. Such component types can be defined as basic types to denominate that they can be used as a basis for different components. As main difference between the use of component types and basic types, the following aspect is pointed out: If a basic type has been assigned to a component, not each single status value is relevant to describe the manufacturing process of the underlying component, contradicting to a component type where this is a requirement.

This section revives the difference between the two types regarding the specification process. Depending on using component types or basic types, different policies are possible (Fig. 21).

Variant A: In a first step, only basic types are defined. These are assigned to the project specific components. Components are weighted by their costs. Afterwards, the specification of the construction activities including their relations to the components in states (precondition and result) takes place. The computation of the order of activities is performed using the basic types evaluating the rules as described in section 4 and ordering the computed relations between activities as described in section 5. At this stage, no percentage weighting has taken place. Before this can be done, it is necessary to determine a component type for each component. This is realized by back tracing the affected status variables for each component. Thus, the component types are assigned to the components in a separate step by computation. If prices are explicitly specified for reaching each status value in a basic type, percentage values can be derived for a component that does not use each status value. All status values that are reached by this component during the process form 100%. The proportion of its price to the sum determines the part of each status value. Now, deadlines can be assigned to project steps, and all evaluations can take place as described in section 6. The computed component types can be visualized. If different component types are computed than expected, incorrect specifications might have taken place.

Variant B: Contrary to variant A, variant B requires component types to be defined at the very beginning of the specification process. This can be done on the basis of basic types, which beyond that are of no further interest for this variant. Together with the specification of the component types the percentage weighting takes place. Instead of a basic type, the user assigns in variant B a component type directly to each project specific component. Specifying a type for a component thus results in a set of status values that need to be set by construction activities. It can be checked automatically whether this has been considered in the specification of construction activities.

Variant A	**Variant B**
1. Specification of basic types	1. Specification of types of components
2. Weighting basic type with prices	2. Weighting types of components
3. Assignment of basic types to components	3. Assignment of types to components
4. Specification of construction activities	4. Specification of construction activities
5. Computation of the order of activities	5. Computation of the order of activities
6. Counting back types of components	6. Evaluation of the model
7. Evaluation of the model	

FIG. 21: Different specification procedures.

The presented variants result in different specification effort at different point in times, and they offer different technique for checking user input. It is necessary to test the best variant in practice.

8.2 Durations of activities

It is possible to enlarge the modeling technique in such a way that the user specifies durations for each activity. Deadlines can be determined for each construction activity if the start of the project and the durations of all activities have been specified. Minor modifications of the topological sort algorithm are necessary to do that computation, but the effort of specification increases compared to the specification of deadlines for project steps. Thus, it is a question of weighting effort and return against each other whether the specification of durations of activities is a further beneficial concept. In general, the duration of construction activities depend on resources so that it might be more efficient to leave out all considerations concerning restrictions in the use of resources from this concept of "preprocessing" scheduling.

However, the presented modeling technique can be enlarged in such a way that specific activities can be weighted. For example, the duration concrete requires reaching a specific strength is of technological nature. Such "technological durations" can be specified in a component type or in a basic type as an attribute of a specific status value, in case of concrete as an attribute of the status value "concrete set". The value of a "technological duration" can be regarded as a weight that can be passed to the activity that sets the specific state. The duration can be exported to a scheduling tool describing a time frame that is independent of working days per week or working hours per day, and that has to be considered in the follow up chart as a technological restriction. Existing descriptions of component types only address a single aspect, e.g. costs. These descriptions need to be reviewed, and the technological restrictions in time need to be added. It has to be investigated whether the results justify the effort of additional specification.

9. CONCLUSIONS AND OUTLOOK

The presented modeling technique is a further step in enlarging the use of information technology in the area of scheduling construction processes. At present time, for a general approach valid for the wide range of different construction projects, practice and academic approaches are a long way away from each other. In practice, the use of information technology in scheduling is restricted to support planners disassembling a construction processes into activities and specifying interdependencies between activities relying on the experience of the person doing the job. Academic approaches are taking advantage of information technology by automating the development of schedules, generating a list of activities, as well as precedence relationships between these under physical, structural, resource and safety constraints. Many of these approaches have in common the component based view on the process which results in activities where each one of them describes the erection of an entire project component. This restricts the level of detail a generated schedule can reach.

The presented modeling technique addresses the gap between manually specified schedules and generated schedules. The aim is to keep flexibility of modeling the process at a user defined level of detail but to reduce the error-prone and time consuming effort of specifying the relations between activities. For this purpose, the disassembling of the construction processes into activities is left to the user, even though the level of detail is influenced by the set of applied component types, but algorithms are used to calculate the technological interdependencies between the activities. The presented modeling technique addresses the specification of technological interdependencies between activities and shows that algorithms can be used in a beneficial way to compute interdependencies completely and correctly. Technological interdependencies must be considered in a project. Because they are computed and therefore available, scheduling can start using a checked basis so that the quality of scheduling can be improved.

An equivalent modeling technique has been developed for engineering planning processes. This technique has been successfully tested in real engineering projects (Huhnt and Lawrence, 2004). Practical tests of the presented technique for construction processes are in progress. Specifically the use of existing descriptions of components require additional research so that existing knowledge can be used in a beneficial way to compute correct and consistent orders of construction activities.

The use of the presented modeling technique results in further questions and research. The distinction between the logic of a construction process and its weights requires efficient techniques to guarantee the completeness and consistency of weights. For instance, costs need to be assigned to all components of a building. Interdependencies between disciplines need to be considered. Cost information need to be checked, and consistency has to be reviewed. Complete weightings open the way to optimization problems. Several solutions are valid for an order of activities. Overall optimization techniques need to be investigated to select “the best”. These overall optimization techniques need to consider additional restrictions that result for instance from restrictions in the availability of resources. Expanding the presented technique to consider further restrictions and constraints is a challenge and has to be investigated in future.

10. REFERENCES

Aalami F., Levitt R.E. & Fischer M. (1998). *A Customizable Representation for Construction Method Models,* Working Paper Nr. 51, CIFE, Stanford University, Stanford, CA, USA.

DBV (1998). *Anleitung zur Erstellung von Projekthandbüchern*, Deutscher Beton- und Bautechnik-Verein, Berlin.

DIN 276 (1993). *Kosten im Hochbau,* Deutsches Institut für Normung e. V., Beuth-Verlag, Berlin

Dzeng R.J. & Tommelein I.D. (1993). Using Product Models to Plan Construction, *Proc. 5th Intl. Conf. on Comp. in Civil and Building Engrg.,* June 7-9, Anaheim, CA, ASCE, New York, USA, 1778-1785.

Dzeng R.J. & Tommelein I.D. (1997). Boiler Erection Scheduling Using Product Models and Case-Based Reasoning, *J. of Constr. Engrg. and Mgmt.,* ASCE, 123(3), 338-347.

Fischer M.A. & Aalami F. (1996). Scheduling with Computer-Interpretable Construction Method Models, *J. of Constr. Engrg. and Mgmt.*, ASCE, 122(4), 337-347.

Hendrickson C., Zozaya-Gorostiza C., Rehak D., Baracco-Miller E. & Lim. P. (1987). Expert System for Construction Planning, *J. of Comp. in Civil Engrg.*, ASCE, 1(4), 253-269.

Huhnt W. (2005). Generating Sequences of Construction Tasks, *Proceedings of 22 nd Conference on Information Technology in Construction*, Dresden, 17-22.

Huhnt W. & Lawrence J. (2004). Methodik zur Modellierung konsistenter Soll-Vorgaben für komplexe Planungsprozesse, *Bauingenieur*, No. 79, 50-56, Springer, Düsseldorf.

Mittag M. (2000). *Arbeits- und Kontrollhandbuch zur Bauplanung und –ausführung nach § 15 HOAI*, WEKA Baufachverlage GmbH, Kissing.

Navinchandra D., Sriram D. & Logcher, R.D. (1988). GHOST: Project Network Generator, *J. of Comp. in Civil Engrg.*, ASCE, 2(3), 239-254.

Pahl P. J. & Damrath R. (2001). *Mathematical Foundations of Computational Engineering,* Springer, Berlin, Germany.

Schäfer W. (2002). *Kostenplanung – Teil II*, Lecture Notes, Institute of Construction Management, Rostock, Germany.

Schmitz H., Gerlach R. & Meisel U. (1995): *Baukosten, Preiswerter Neubau von Ein- und Mehrfamilienhäusern,* Verlag für Wirtschaft und Verwaltung Hubert Wingen, Essen.

Sommer H. (1994). *Projektmanagement im Hochbau*, Drees & Sommer AG, 1. Auflage, Springer Verlag, Berlin.

Turau V. (1996). *Algorithmische Graphentheorie*, Addison-Wesley, Bonn.

Winstanley G., Chacon M. & Levitt R. E. (1993). An Integrated Project Planning Environment. *International Journal of Intelligent Systems Engineering,* Vol 2, No.2, pp 91-106, Summer 1993.

Winstanley G. (1995). A Planning System for Large-Scale Industrial Projects. *Proc. of the IEE Colloquium on "Knowledge-Based Approaches to Automation in Construction",* Digest No. 1995/129. ISSN 0963-3308.