Resources, Time & Money: Why project schedules simply don't work

S.M. Holzer & F. Geyer
Institute of Mathematics and Construction Informatics, University of the Federal Armed Forces Munich, Germany

ABSTRACT: Construction of buildings is a very special kind of production because each product is generally produced only once, as an individual. Therefore, there is no means of an empirical assessment of cost as a function of resource deployment, time, and sequence of tasks, by way of *experimentation* as in the case of industrial mass production. On the other hand, buildings are very expensive, so that there is a great need for reliable cost estimation, estimate updates during construction, and cost analysis. Due to the lack of known, reliable functional relations between the key cost-relevant quantities, very coarse simplifications of the system need to be introduced. The contribution discusses ways to develop such simplified cost models, without rendering valueless the results obtained by such a model. We assume that the whole cost estimation process will be implemented into an overall schedule planning system, without introducing insurmountable overhead.

1 INTRODUCTION

Commercial software tools are available for the problem of temporal *scheduling* of tasks in a production process. Such tools are based on graph theory and determine the longest path through the graph of processes and the sequence rules between them. The longest path corresponds to the minimum production time required. Input data required for the construction of a time schedule include any two of the following three: *time* required for each individual task, *efficiency* of the resources allocated to the task, *size* of the task.

Typical post-processing options included in a scheduling tool are presentation of the computation in the form of a bar chart (Gantt chart), as well as resource-versus-time charts. Adjusting the schedule to possibly limited resource availability is usually not well supported by commercial scheduling tools.

However, the time schedule is not really what one wants first when considering a construction process. Rather, what one needs first is a good estimate of the final *cost*. Often, the total duration of the construction is given as a constraint rather than sought for, and it is therefore the prime concern how to allocate resources in such a way that one meets the deadline at minimum cost.

More important still, it may happen that, during the actual execution of the project, we encounter unexpected delays or problems. In such a case, we will want to know as soon as possible what financial risk is entailed by the delay or change in the schedule.

In other words, at the beginning of a new construction project, we do not really want to plan a project schedule, but what we need is a *financial plan and model*. One of the reasons why this is particularly difficult in the construction business is that buildings are individual, unique products. As opposed to stationary mass production, it is not a feasible way to perform a test and determine the relations between task size, resource allocation, time, and cost experimentally.

The link between the size of a task, the time required, and, ultimately, the cost estimation is provided by the resources employed (human and machinery) for accomplishing the task.

Resource assessment has a long tradition in stationary mass production. For example, the *Reichsausschuß für Arbeitszeitermittlung* (REFA, committee for the investigation of working time) was founded as early as 1924 in Berlin, Germany. It continues to this day to work in rationalization of production processes in mass production.

Unfortunately, these ideas cannot be directly transferred to construction industry because the working conditions play a key role in the net efficiency which will be attained by the resources; however, the working conditions are a priori unknown in construction industry. In construction industry, the working conditions depend strongly on the process itself, and also on incontrollable and even unforseeable outer influences (e.g., weather, just to begin with). This is the main reason why even the initally computed time schedules usually fail to coincide with the actual construction sequence. In fact, what one would need is a feed-back iterative planning of the construction schedule itself, starting from some assumed working conditions and then performing an update of the assumed conditions, based on an analysis of the computed schedule. Once we have computed a "trial" schedule, we can detect collisions and problems of any kind which will deteriorate working conditions.

There is no hope of really overcoming the problem of indetermined resource efficiencies in construction processes in a deterministic way. Therefore, there is also probably no way to foretell construction cost and financial risk precisely. However, there is some hope at least to *bracket* both temporal and financial efforts connected to a specific construction project by employing some fuzzified model.

2 RELATION BETWEEN RESOURCES AND COST

Construction cost can be associated with three types of cost generators:

- cost of material
- cost of human and machine resources
- cost of financing

The first kind of cost is rather straightforward to determine once one knows *what* to build. However, the other two depend mainly on *how and when* one builds.

In the following, we start from the (simplified) assumption that *what* we want to build is fixed.

Time and resource cost are strongly related. Resource cost is determined by resource allocation to individual tasks. Once we have allocated a certain number of a certain type of resource to a task in the construction process, we can compute the time required to complete the task. The time required, multiplied by the unit cost of the resource per unit time, gives the actual cost of the resource.

However, such a computation holds only under strongly idealized conditions. In reality, the *efficiency* (output of a unit resource per unit time) of a resource depends in some non-linear way on the number of resources assigned ("if two masons need four days to build a wall, four masons will *not* accomplish the same task in two days"). Therefore, the time schedule itself depends non-linearly on resource allocation.

Similarly, the unit *cost* per resource may depend non-linearly on the number of resources assigned ("buy one get one free").

Both these two non-linearities are, of course, also present in stationary mass production. However, there they can be assessed at least empirically by the way of tests on a reasonably big number of prototypes. Building unique constructions, this is not a feasible way of resource assessment.

It is therefore probably a hopeless endeavour to determine functional expressions relating task size, time required, and cost in the construction business. Even if they were available, it would probably be vain hope that someone would take the trouble to input all these data into a scheduling tool.

In addition, tasks cannot be considered independent of each other. One of the main sources of financial trouble in the construction business is related to mutual hindering occuring between different tasks. E.g., assume that a painter A paints the wall in a room. Simultaneously, a floor layer B creates the floor in some other room. So far no problem. However, the efficiency of both will drop radically if both work in the same room at the same time. Also, painter A will work less efficiently if floor layer B has already done his job in the room to get painted because he will have to protect the new floor from the paint. However, before we have computed the construction schedule, we cannot foretell whether any of these cases will happen or not. We can only guess and adjust the efficiencies which we can expect for A and B in a trial-and-error procedure, perhaps guided by some automatic optimization tool.

However, the project schedule usually does not contain the data which we would need to update the efficiency defects resulting from mutual hindering. Such constraints are typically of a *geometric* type (the painter need to pass through room 1 in order to reach his working place, while the floorlayer may just be working in room 1) and are completely outside the scope of the scheduling tool.

Efficiency and cost of resources may even, in some cases, depend directly on absolute date/time: It is more costly to excavate a foundation in winter than in summer.

Finally, resources may be limited from above and below: We may need a certain number of workers to be technically able to accomplish a certain task, and, on the other hand, it may be impossible to acquire or accommodate more than some fixed maximum number of workers. These constraints may get active under some conditions which again depend on the time schedule. In order to simplify the problem of cost modeling, however, we will assume in the following that resources have unlimited availability, so that it only becomes a matter of money to buy or rent whatever we need in order to complete the construction. We do not lose much by this simplifying assumption because limits on resources can still be handled in a penalty approach by making them very costly outside the permissible interval.

Cost of *financing* depends on both material cost and resource cost. We need the functional relation between the sum of material and resource cost, as well as the customer's payment plan, in order to be able to compute the resulting cost of financing.

3 RESOURCE MODELING

Anyone who plans a construction project has some rough idea about "reasonable" resources attributable to a certain task. Essentially, this is what he has to do even in case he just wants to compute the tentative construction time schedule.

Furthermore, the project planner has some idea of the output achieved, under "normal" conditions, per unit resource.

However, the project planner has also some rough idea about how the resource efficiency decays if resource deployment is less than optimal, or if working conditions deviate from "normal" conditions.

Let us, for the time being, assume that all tasks may be grouped qualitatively into three classes, A="quite insensitive to influences", B="very sensitive", C="not particularly sensitive". The quality "sensitivity" describes the effect which changes in resource allocation and in the general working condition have on the output. Qualitatively, the three sensitivity classes of resources might be represented in a graph similar to figure 1:

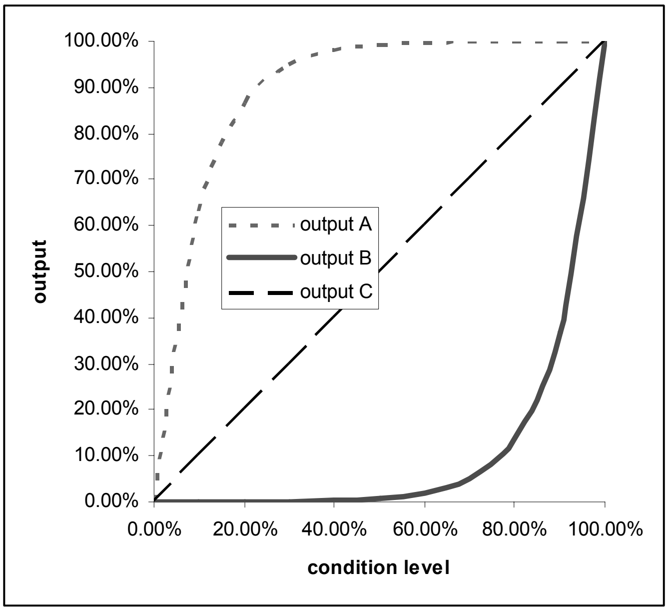

Figure 1. Three sensitivity classes of resources

In fact, the figure shows (quite arbitrarily) a mathematical formulation of the three sensitivity classes. We might as well have quantified the three classes by some number, running from class A corresponding to 0, to 1 corresponding to class B. One could use some parametrized function $f(x,a)$ with parameter a which fulfils the following conditions:

$$f(x,a)\begin{cases} \to 1 \text{ on } 0 < x \leq 1 \text{ for } a \to 0 \\ = x \text{ for } a = \frac{1}{2} \\ \to 0 \text{ on } 0 \leq x < 1 \text{ for } a \to 1 \end{cases}$$

and $0 \leq f(x,a) \leq 1$ for $0 \leq x \leq 1$.

Let us now agree to use the parameter a with $0 \leq a \leq 1$ as a measure of sensitivity.

"Sensitivity" makes sense only if we also define sensitivity with respect to what. Here, we have introduced another very simplifying assumption.

The quantity x on the abscissa reads "condition level", once again assuming values between 0 and 1. This rather imprecise quantity is assumed to describe, in some *overall* measure, *all* kinds of effects that we may encounter when actually carrying out the task: too few or too many workers assigned to the task, too many others around which hinder the process, effects of foul weather, and so on. x is a completely *abstract* quantity, so to speak dimensionless. How a specific task depends precisely on changes in the working conditions is described by parameter a, not by x. In other words, x=0.5 is only an abstract measure of "conflict" inflicted upon a specific task. That task, on the other hand, may in turn be quite insensitive to "conflict", which will be expressed by a close to zero, rather than using a different scale for x for each task individually. x is a scale-independent quantity. The reason for introducing this quantity is that, later on, we would wish to select x for each task automatically in a rather coarse way, without asking the user for any input on that. All experience and external knowledge of the user will be represented solely by specifying a.

Assume, for the moment, that someone is able to provide concrete numbers for x and a. Then, we have a computable functional relation between working conditions and resource output.

The presentation in the preceding paragraphs has been such as to suggest an application of *fuzzy algebra* in a next step. Using, e.g., simple triangle fuzzy numbers, we might be able to model the sensitivity a assigned to a specific resource by a simple bi-linear membership grade function on the set [0;1].

The figure 2 shows a model for a resource which is "not quite sensitive", the maximum of the membership grade function being at a=0.4, corresponding to a curve slightly above the diagonal in the preceding figure. Using this fuzzified value of a relieves us of fixing a definite number for it, but burdens us by using fuzzy algebra for the evaluation of the output function.

The next question is whether it makes any sense to handle the working conditions x as a fuzzy quality as well. This question is actually posed the wrong way. The working conditions are not a primal quantity, but rather a dependent variable. What we need to do is to fix some deterministic default number (typically x=1) for the working conditions (i.e., "normal conditions") initially. Subsequently, we enter into a fuzzified computation of the project schedule with this value.

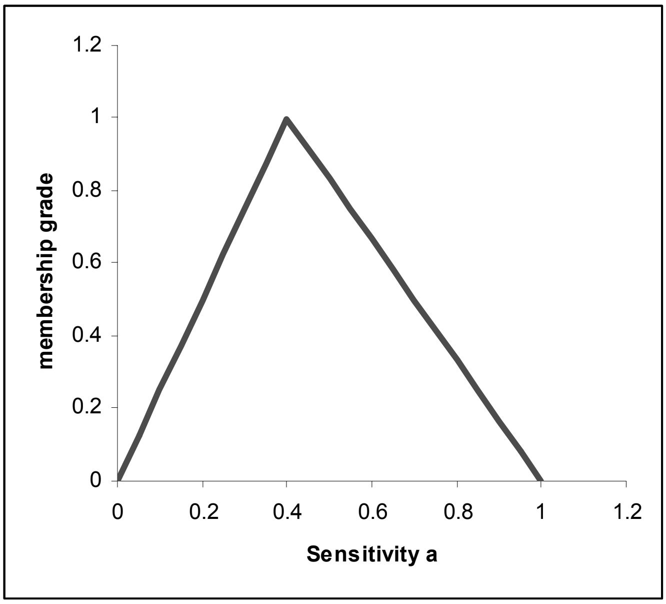

Figure 2. Bi-linear membership grade function of sensitivity

To this end, we can use the normal graph-theoretical longest path solver. We require nothing but the standard input, namely, some resource assignment for every task, and the "normal" output for the resources assigned.

We will arrive at a fuzzyfied schedule in a similar way as it has been done, e.g. in Freundt (2004). However, as opposed to Freundt 2004, we are not mainly interested in that specific schedule, but rather in its effects on the yet to be fixed parameter x, and, ultimately,on predicted construction cost.

To achieve this, we need to extend the data present in the process model. At least, we need some model on how typically some constellations influence working conditions of some sub-set of our tasks.

Many tasks in the schedule are linked to geometrical information. Each part of the building that needs to be erected has some location in space. This data is typically available in a CAD program. We require some link between the scheduling and the CAD code.

Next, we can introduce some measure of the influence of proximity in space and time on the quality x of the working conditions. This need not be a very detailed model, rather some rough estimate (e.g., we might assume that any two tasks having to be carried out within a distance of less than 5 m from each other during "roughly the same time" will deteriorate each other's working conditions to 0.6 or by a factor of 0.6). If we use this deterioration function to compute the quality measure x from our fuzzy output of the scheduling problem, we will essentially obtain a fuzzy parameter x. This holds even if the deterioration function itself does not introduce any new fuzzy parameters. In order to keep fuzziness limited, it is probably best to de-fuzzify x at this instant.

Of course, it is sensible to reset the condition number x to 1 in case there are no conflicts left for a specific task, due to shifts caused elsewhere in the project schedule.

We are next going to feed back the working conditions x into the schedule, and we obtain a new, hopefully more realistic, schedule.

Note that, so far, adjusting of the condition number can only have one effect, namely, extending the duration of a specific task.

4 STRUCTURE OF A BASIC COST MODEL

Until now, we have dealt only with resources, and, indirectly, with time, but not with money. However, it is straightforward to obtain monetary assessments of the project once we have a somewhat realistic temporal schedule. The same question as before arises when we want to associate money with resource employment: Should there be a deterministic function relating resource usage to money, or is it better to model that in a fuzzy way? We are following the guideline of keeping fuzziness to an absolute minimum in order to obtain meaningful results in the end. Therefore, we suggest that we use a *fixed* unit price for the resources, rather than any fuzzified relation. The unit price does not depend on the quantity of the resource used.

This amounts to saying that we put *all* fuzziness *exclusively* into the parameter a. In other words, the indeterminateness of the "output" of a resource comprises also the indeterminateness of cost. All effects related to quantity discounts and the like are considered as properties of the condition to output map described by x and a.

After this daring straightforward modeling decision, we have a working model which, after some iterations, will hopefully converge to a stationary solution for the scheduling problem. If we encounter conflicts, this will delay our project progress; delays may occasionally lead to resolving conflicts elsewhere, which we can also handle by resetting the condition number to 1, However, it is by no means sure that we may not find ourselves locked in an infinite repetition of useless iteration cycles. Convergence is not guaranteed at all, or rather, frankly speaking, quite unlikely. However, the effects of lack of convergence on our main goal, money, may be less dramatic. Therefore, we may altogether dispense with convergence!

Assuming (still quite naively) that there exists at least a locally optimal solution with respect to some objective function, we can, however, "force" such a problem to converge towards the solution if we employ some suitable optimization technique.

We are not at a loss to formulate a suitable objective function which we want to minimize. One such function would be, of course, total cost.

However, what are our design parameters? Of course, the amount of resources assigned to each task. It is important to note that the problem has changed a great deal now with respect to the classical scheduling problem: The scheduling is now turned into a mere side calculation required in order to determine the trial cost of the project. The governing, driving problem is the cost estimation problem. We do not really care for a "converged" solution to the scheduling problem. Some rough idea of the process schedule will be fine enough for our ultimate goal of cost estimation provided the prediction of the active deterioration functions is not too far from reality.

If a certain task is hampered by suboptimal conditions, it will be delayed by reduced output in our model. This may not be realistic in all cases because we may rather opt for paying more for that task, but get it completed in time. However, such an expedient is still available in our model, precisely by increasing the resources allotted to the task. This will increase cost while possibly keeping the duration constant. In other words, our model is complete in this respect. Remember that we are going to use fixed unit prices for everything, putting all fuzziness into a and all design parameters into resource allocation.

5 SCOPE OF THE MODEL

The model presented looks pretty simple. It remains to give an outline how such a model can be implemented and used. Let us first consider the problem of input. We require:

1 assigning the peak value of the task-specific sensitivity a on the scale 0 to 1 to each kind of a resource
2 input of at least some "deterioration functions".

The first requirement is probably easy to fulfill. The second is more complicated. At this point, we have to look for the other contributions of the present workshop. Referring to Huhnt, we find that tasks are derived from task classes. Of course, it suffices to define "deterioration functions" between *classes* of tasks rather than individual tasks. The evaluation of the deterioration function for each member of the set of one task class to all members of the other class involved can be carried out automatically. Rules such as "floor layers and painters are likely to hamper each other when working close to each other at the same time" can be implemented once and for all. The functional representation of such rules might even recur to *fuzzy logic* (rather than fuzzy algebra) because the input data are fuzzy anyhow. Details have to be fixed during the actual implementation phase of the research project outlined here.

Huhnt's contribution offers the direct access to the longest-path problem once we have performed resource assignment to tasks. Whereas "generating sequences of construction tasks" is based on mere topological sorting of the tasks, the longest path problem can be solved on an edge-weighted graph by Dijkstra's algorithm. Implementation of that is straightforward on a graph-centered code like Huhnt's.

6 CONLUSIONS AND OUTLOOK

In the present contribution, we have outlined a method to incorporate cost estimation into project scheduling under the specific constraints posed by the construction industry. We have resorted to a mildly fuzzified representation of the operating resources employed, and we have put all indeterminateness associated with the project exclusively into the resources. All other relations have been kept strictly deterministic, with the single exception of possibly modeling the mutual influence of spatial or temporal proximity of possibly conflicting tasks also in a fuzzy approach.

We have also demonstrated that such a model can be incorporated within the scope of a classical graph-based scheduling tool, or, alternatively, within Huhnt's graph-based sequencing tool.

The model developed is such that the input which needs to be provided by the user is in fact limited to earmarking each resource used in a class by some "sensitivity class", specifying the peak value of the parameter a in the range 0 to 1. Furthermore, some super-user has to implement at least a few very important "deterioration functions" for task classes, rather than individual tasks. In order to obtain geometrical information, the scheduling needs to be linked up to a geometric design tool (CAD).

Our approach differs from the one presented by Freundt (2004) essentially in the following points:

1 Our main goal is not a fuzzy representation of the schedule problem; rather, for us, the fuzzy scheduling serves only as an input for the *cost assessment*.
2 We require only *one single* input value for the fuzzification because the parameter a is supposed to occupy the interval [0;1], so that only the peak location of a is free. Freundt, by contrast, needs input of three values.
3 Our approach is oriented towards a more strictly causal and more fine-grain modeling than Freundt's: Rather than modeling execution times

as a primal fuzzy quantity, we identify loss of efficiency in the resources as a cause for delays (which means, by the way, that we do not suppose that tasks can be completed ahead of time at all). Therefore, we obtain execution times by a fuzzy algebra computation as fuzzy numbers, but we do not have to ask the user to estimate them directly.

4 Scheduling is achieved in an iterative fashion in our general approach.

5 We have opened an approach towards *cost minimization* by the causal approach because *resource allocation* remains a free, deterministic design parameter in our model.

The next steps required are incorporation of the model into existing code frameworks, as well as embedding the scheduling task into an outer optimization loop.

What remains to be done is to explore the precise nature of the cost minimization problem and select suitable optimization strategies. We hesitate to assume that simple strategies would suffice to solve this kind of problem.

Finally, the whole model needs to be embedded into a cost control tool which will be used to supervise the actual construction process. During construction, details on the actual performance of the resource become available, which will help us to adjust the model by replacing the $x \to a$ map by observed data. At the same time, the number of design variables is continually decreasing, so that it becomes increasingly easy to find out what has to be done or what can be done in order to prevent the worst financial disaster of a project running less than perfectly.

REFERENCES

Huhnt, W. 2005. Generating sequences of construction tasks. Contribution to this minisymposium.

Freundt, M. 2004. Ein flexibles Modell für die Bauablaufplanung auf Basis von Graphentheorie und Fuzzy-Zahlen. Ph. D. thesis, Bauhaus-Universität Weimar. Includes further references to scheduling problems and fuzzy numbers.

Hanss, M. 2005. Applied fuzzy arithmetic.