

# Prediction of consequences for planning situation based decisions

A. Lähr & K.-U. Bletzinger

*Chair of Structural Analysis, Technical University Munich, Germany*

**ABSTRACT:** Consequences of a decision made by a planner (e.g. a project manager, or an engineer) within a collaborative environment can hardly be foreseen. For example, such a collaborative scenario is represented by a planning process in AEC. In particular, during certain planning stages alternatives have to be considered which significantly influence the overall result. Today's AEC planning procedures can be very much improved by predicting simulation methods to judge about the quality impact of certain design or planning modifications. Also, the proper interpretation of data is very crucial to give suitable insight into the characteristic consequences of individual planning decisions. This contribution presents an approach to achieve this goal by discussing needs, problems and implementation for the actual state of our research.

## 1 INTRODUCTION

Efficiency of a planning process is a prerequisite for the success of a product in AEC. Mistakes or sub-optimal decisions can lead to severe negative influences on the ongoing planning process and therefore on the whole product. As a consequence, prediction of planning decisions would contribute to achieve a better result.

The idea of simulating the planning process is well known in automotive industry. There are many techniques to support designers, which are already implemented. They are frequently used to apply multidisciplinary optimization (MDO). However, in AEC most problems arise on the different working environments of the planning participants. The most important difference is the lack of shared common product data models. Those models are being developed now (see below), but are not yet capable of keeping data consistently for a whole planning process in AEC. Another discrepancy is that in AEC there is hardly any discipline-spanning in-house-manufacturing. Different disciplines are distributed to various companies and therefore locations, which usually don't work together for any longer than the duration of one planning project. Thus, the effort to adjust their systems for co-operation for every new project is usually too high to work economically.

The idea of our research project is to realize the interweavement of the different disciplines participating in a planning process in AEC as depicted below in Figure 1.

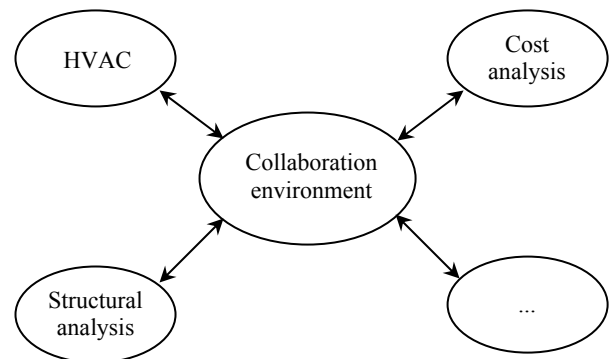


Figure 1. Collaboration schema.

An agent based communication environment allows flexible and scalable implementation of such a co-operative scenario. Furthermore, co-operation with existing agent communication systems is facilitated by using the FIPA standardized Agent Communication Language (FIPA-ACL) for message transfer.

For analysis of planning situations we apply methods from the field of sensitivity analysis, which are typically used in optimization disciplines (e.g. structural or topology optimization). They deal with the question how specific parameters of a system change if certain input parameters vary. Among others, Design of Experiment (DOE) and Response Surface Methods (RSM) are used. Applying these methods the level of impact (consequences) to particular planning parameters can be estimated, which arise by alteration of specified input parameters.



Predominantly, this affects recurring parts of the whole planning process (e.g. dimensioning of constructional elements) if they are subject to multi-disciplinary involvement.

Communication with respect to data exchange is based on a shared common product model. A promising approach is the Industry Foundation Classes (IFC) because of its standardized data exchange methods (ISO 10303: STEP, SDAI). The potential amount of planner's applications supporting this exchange formats steadily increases.

## 2 RELATED WORK

The International Alliance for Interoperability ([http://www.iai-international.org/iai\\_international/](http://www.iai-international.org/iai_international/)) released IFC2x2 as the actual version in May 2003. This release has significant changes to the prior IFC2x version (e.g. ST-4 extension for Structural Analysis Model and Steel Constructions, see [http://cib.bau.tu-dresden.de/icss/structural\\_papers](http://cib.bau.tu-dresden.de/icss/structural_papers)). However, only few planner applications implement the actual version of standard, at the moment.

To serve business needs the product model has to be managed by model servers, which are capable of keeping consistency, multi-user access handling, etc. Among others, the EPM Technology (see <http://www.epmtech.jotne.com/>) company is engaged in implementing these services. Their product, called EPM Data Manager, provides powerful functionality to handle EXPRESS based product models. Also, the Institute of Applied Computer Science in Civil Engineering at the University of Technology Dresden developed a model server along with the iCSS project (see <http://cib.bau.tu-dresden.de/icss/>) to fulfill the postulated needs.

Extended functionality, like communication between the participants, project management, knowledge based services is obtained by cooperation platforms. Recently, several research projects were involved in developing those constructs. ISTforCE (see <http://www.istforce.com/>) and iCSS research projects have been finished. ArKos (see <http://www.arkos.info/>) is a new and actual project of several companies and research institutes.

To find more acceptance in practice much more functionalities have to be implemented in such platforms. For example, the previous services (in particular IFC model servers) were not able to grant access to partial models. Only either individual objects or the whole model could be exported. The Institute of Applied Computer Science in Civil Engineering at TU Dresden developed a global model subset definition schema (Weise et al. (2003)) to define partial models. It is capable of representing a specific view of the model (for example the structural model, or the HVAC model). This is one important

step to support collaboration of planners from different disciplines.

Collaboration and co-operation in a planning process implies combining different planner's applications. Data transfer between them can be achieved by using and "understanding" the underlying common shared product model. Unfortunately, the number of planner applications implementing such product models, in particular IFC, is not very high. Most of them support read access from IFC only. Exceptions are, among few others, mainly CAD-applications like ArchiCad (see <http://www.graphisoft.de/>) or ADT (see <http://www.autodesk.com>) which produce IFC data in form of a STEP Physical File (ISO 10303-21 IS 1994). However, they usually support the geometric model only.

## 3 COLLABORATIVE ENVIRONMENT

### 3.1 General

As mentioned in section 2 several projects are dealing or dealt with environments for collaborative work in AEC. However, we develop our own model, because we need additional features like dynamic workflow management or analyzing support for certain parts of a planning process (see below).

A collaborative environment (CE) in our context needs to be capable to include workflow, involved planners and underlying product model data for a certain scenario. Such a scenario is set up by an analyst who defines the product model data and possible changes (variants) to that model. Afterwards, affected planners have to process this information and provide the CE with their results. Lastly, the analyst will pick up the data and do an analysis of that process resulting in an assessment of consequences and pointing out alternatives for the accomplished modifications.

Heterogeneity and different demands of planner applications make generic interfaces highly essential. Therefore, development of an integration mechanism for involved planners is necessary. Furthermore, workflow management should not be handled statically by the analyst himself. A more dynamic workflow as well as product data management model has to be introduced.

In the following an agent based CE is presented, which possesses all the postulated properties.

### 3.2 Agent technology

Agent technology is a methodology for handling network communication on a very high level of abstraction. A software agent is an independent computer program, which is able to autonomously inter-



act with its environment. Decisions to be made are based on its perceptions and contribute to reach its goal. For detailed information see Woolbridge & Jennings (1995) and Ferber (1999). Furthermore, agent mobility is an important feature in our context. It enables an agent to migrate to other environments. Actually, agents are created on an agent platform, which can be spanned over several computers connected via network. The agent environments on these computers are called containers. Two types of agent mobility can be specified. Intra-platform mobility, which means migrating from one container to another within the same platform and inter-platform mobility for cross platform migration. In foreign containers agents are able to communicate and interact with agents and services residing there. The instrument of conversation is the agent communication language (ACL), a standard released by the Foundation for Intelligent Physical Agents (see <http://www.fipa.org/>).

So called multi agent systems (MAS) conglomerate several agent platforms and allow the agents to move to other platforms (inter-platform mobility). Such agents are called mobile agents and strongly support collaboration of geographically distributed planners.

### 3.3 Integration mechanism

To integrate the different involved persons (planners and analyst) the class of proxy agents is introduced. A proxy agent (PA) represents the participating person or institution within the agent environment (multi agent system). Accordingly, a PA has to act as an interface to external, planner specific software, which mostly has been approved over a long time. This technique is also known as software wrapping. A software wrapper is a thin software program or module that converts program-specific input and output operations into generic sets of commands that can apply to a wide range of programs.

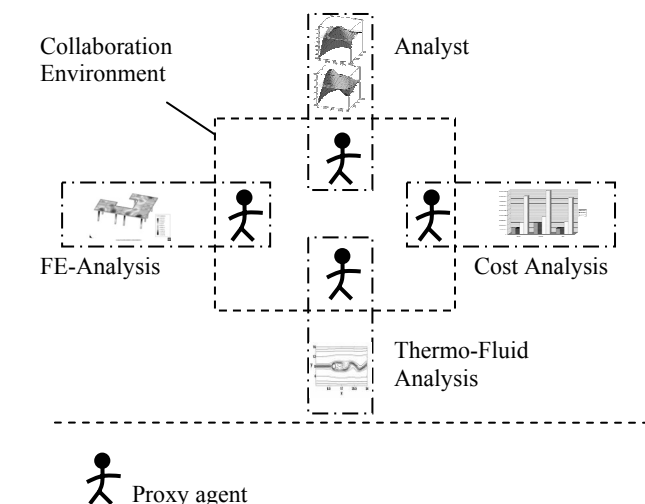


Figure 2. Integration by proxy agents

Furthermore, a proxy agent is immobile and resides in a local container on the same computer as the software to be integrated. The local container is connected to the (main) platform and the proxy agent is able to interact with this environment.

Figure 2 depicts the integration concept. The central rectangle (dotted line) represents the agent environment. Note that there is no distinction between agent container and platform when the term agent environment is used. The dot and dashed line describes a local container (e.g. a personal computer) and contains the proxy agent as well as the application to be integrated.

### 3.4 Workflow management

Typically, complexity of a planning process in AEC renders determination of the planners affected by another planner's decision highly delicate. Furthermore, because within this environment mainly partials of a planning process are to be defined, there is a designated internal workflow management.

There is one central workflow manager, who gets notified if an event happens that affects the workflow of a run. A run in this context is the process starting with initial definition of the product model data and ending when all results from the involved planners are provided to the CE's data management (see 3.5).

The workflow manager is an immobile software agent. Furthermore, there is no direct communication between a workflow manager and a proxy manager, because there is no static process chain steered by a central institution. A more flexible and dynamic way is letting the planners choose when to contribute to the run. To realize this, another class of agents are introduced, the monitor agents. Monitor agents are mobile agents, which have an owner. As mobile agents they can migrate to the workflow manager's environment and attach themselves to it. They possess a filter to screen the workflow events of the workflow manager and, in case of interest, inform their owner (proxy agent), who can react on such events.

Everytime the workflow manager gets an event notice, he delegates it to the attached monitor agents. Figure 3 depicts the processing of such an incoming event notice. After delegation to the monitor agents the second monitor agent realizes an event of interest and contacts his owner (proxy agent). An Event notice is sent by a proxy agent when any CE relevant activity finished. Also, the product model agent (see below) is able to initiate (fire) these events.



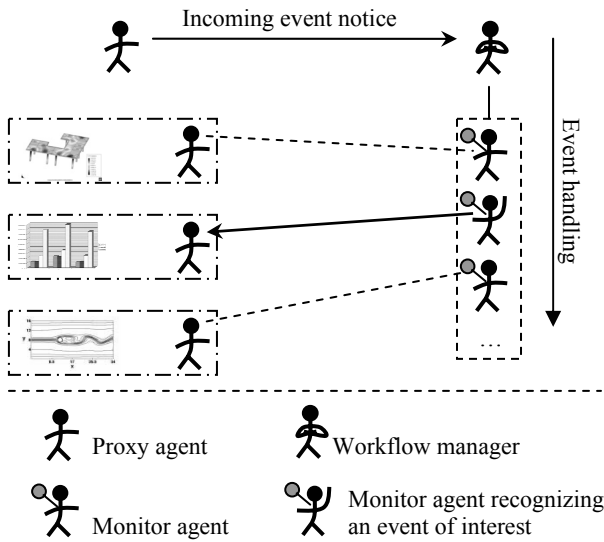


Figure 3. Workflow management

### 3.5 Product model management

The variety of data depends on the definition of the product model. We decided to implement the Industry Foundation Classes (IFC), maintained by the International Alliance for Interoperability because it is widely accepted in this field. Further developments, like the release of IFC2x2 in May 2003 seem to be a step forward to enable integration of the standard in current applications.

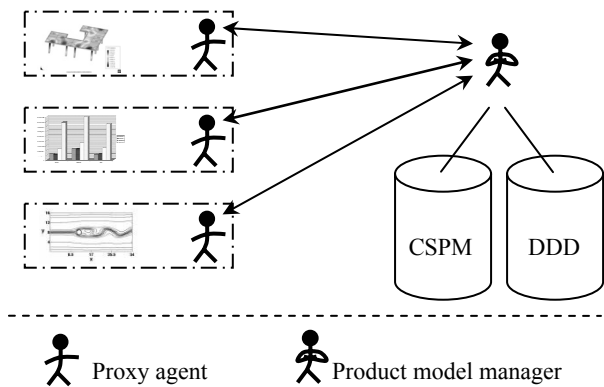


Figure 4. Product model management

Thus, planner application specific data may not be transferred along with the mentioned standardized product data model (IFC). Three reasons are responsible for this:

- 1 The product model definition is still under development and certain features are not available yet.
- 2 The planner specific data will not be part of the standardized model, because it is considered to be of no interest for other involved persons.
- 3 Used planner applications may be unable to write their data back into the product data model consistently.

In particular, for our purpose, planner specific data may be very interesting indeed. This fact leads to the strategy to split the product data into two parts: A common shared product model (CSPM) based on IFC and an extension called dedicated domain data (DDD). If DDD cannot be expressed directly in IFC, it has to be converted to IFC Property Sets. IFC Property Sets are generic data containers, which can be attached to common IFC objects by a link to the global unique ID (GUID).

The question is how to (i) transfer the data to the planner, (ii) modify it and (iii) merge it back to the product data model. Usually, a model server is responsible for accessing and keeping consistency of the global model. This task is fulfilled by a product model agent. The target applications, which will be used, must be able to read the data exchange format IFC as ISO 10303 Part 21 - Step Physical File (SPF) format to realize step (i). Clearly, step (ii) is left to the applications themselves and step (iii) again is to be managed by the product model agent.

A problem arises if DDD gets very large because of the variety or size of the result sets or the domain specific data. Transferring and parsing of such a file can easily lead to severe performance losses. The product model is able to filter corresponding DDD parts to keep exchanged data slim. This means that for every application/domain a certain subset of DDD data can be provided along with the CSPM.

## 4 ANALYSIS METHODS

### 4.1 Sensitivity analysis

Basically, sensitivity analysis is concerned with the question of how response  $y$  of a system varies, due to an alteration of the input parameters  $x$ .

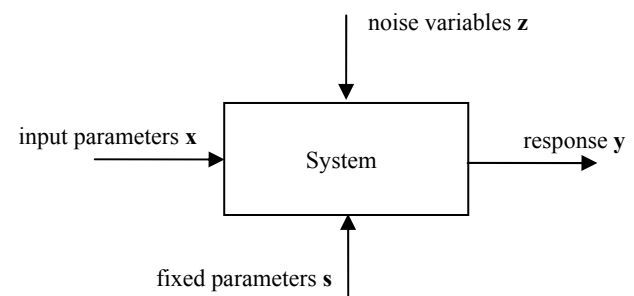


Figure 5. Sensitivity analysis

For completeness in Figure X additional parameters  $s$  and  $z$  are depicted. Noise variables  $z$  are non-influenceable, like uncertainties, etc. Fixed parameters  $s$  are parameters, which remain the same during the whole analysis.

Reasonability of a sensitivity analysis depends on the knowledge of the underlying system. Thus, it can be used to identify an unknown or predict the response of a sufficiently acquainted system.



Particularly, there is to be distinguished between screening methods, local and global methods. Screening methods are used to identify significant system parameters. One of the simplest applications of screening methods is the one-at-a-time (OAT) method. Their major limitation is the neglecting of parameter interaction (Saltelli et al. (2000)). Local sensitivities of a system are imagined at best as partial derivatives with respect to the input parameters. The system has to be known as a function to directly derive the sensitivities from. Otherwise, methods like finite-difference approximation support obtaining the slopes of the calculated system. Considering "real" systems, an analyst is faced with the influence of noise variables. They are uncertainties, like measurement errors, rounding errors, etc., and can't be controlled directly. Therefore, global methods of the sensitivity analysis are applied to cover the total scope of input parameters. The goal remains the same, namely to analyze the system alteration by variation of input parameters.

#### 4.2 Surrogate models

In AEC sometimes a planner's task means high time and effort costs, especially when testing various scenarios with different input parameters. To reduce this effort surrogate models can be created, which processing costs are significantly cheaper than the original ones. Generally, the model cannot represent the real planner's contribution exactly. This is to be faced with the crucial reduction of complexity and effort. The derivation of a surrogate model is realized by applying statistical approximation methods like Kriging to the response data of the tested planner's task. Kriging is a procedure with a statistical, interpolative character and was originally developed by Sacks et al. and presented in Sacks et al. (1989). Further approximation methods are being examined at the moment.

Starting from certain system information and educating to a corresponding function behavior of significant input parameters is a task for Response Surface Methods (RSM). For example, supported by regression analysis, mostly linear or quadratic approximation is used. Furthermore, by application of Kriging these approximations are supplemented by a random process, which allows exact function values at the input sampling points. Aforementioned system information may be of experimental or numerical nature.

#### 4.3 Design of experiments

DOE is a methodology to choose the location of the sampling points, used to accomplish statistical experiments. To make an expedient statement of a statistically derived model a certain number of experiments is to be processed. This number can be

controlled by using different designs like e.g. full factorial, fractional factorial designs as well as Latin hypercube sampling (see Box and Draper (1987)).

## 5 APPLICATION

In this section an example of an analysis will be presented. The participating disciplines for this scenario are structural analysis and thermo-fluid analysis. Former is represented by an application from Institute of Computer Science in Civil Engineering at the Technical University Munich (Germany) and the latter is a computer code of the Institute of Computer-applications in Civil Engineering at the Technical University of Braunschweig (Germany). The example arises from a co-operation of our research projects within the DFG priority program 1103.

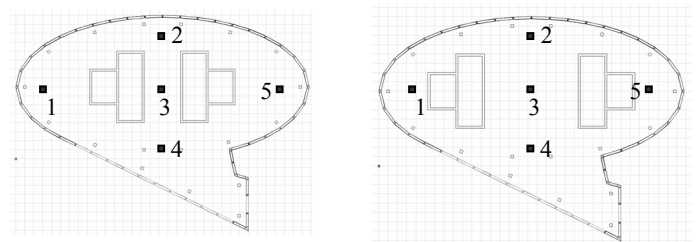


Figure 6. Floor plan. Two variants with different distances between the stairways and locations for the measure points  $p_1..p_5$ .

Depicted in Figure 6 is a floor plan of an office storey. It is to be analyzed how the distance between the stairways is correlated with (a) the room climate at several measuring points and (b) the maximum deflection of the ground floor. The result of this analysis will be an idea of a beneficial distance between the stairways with respect to a well suited combination of (a) and (b). Furthermore, locations with good climatic conditions shall be estimated.

At first, several ( $n_i$ ) variants of product models are created by the analyst and provided to a common product model management by using the collaborative environment, described in section 3. The participating planners calculate each variant and send their results back to the CE. Afterwards, the analyst has access to all results of all variants and can begin with the analysis by defining input and output parameters. Input parameters  $\mathbf{X}_i$  for this case are the distance of the stairways ( $x_{i1}$ ) and the position of the measure points ( $x_{i2}$ ), where

$$\mathbf{X}_i = [x_{i1}, x_{i2}, \dots, x_{ik}, \dots, x_{in}], \quad i = 1..n_i, k = 1..n_k \quad (1)$$

and  $n_k$  is the number of input parameters of a scenario.

The max. deflection of a base plate ( $y_{i1}$ ), air temperatures ( $y_{i2}$ ) and velocities ( $y_{i3}$ ) at the measure points  $mp_j$  ( $j=1..5$ ) are treated as the output parameters of variant  $i$ .



Output parameters for each run are combined to one single system response  $Y_i$  for each variant  $i$ , where

$$Y_i(\mathbf{X}_i) = \sum_{k=1}^n f(y_{ik}(\mathbf{X}_i)) \quad (2)$$

Therefore, these output parameters have to be scaled and weighted. Scaling is done by preference functions  $p$ , which penalize their target for deviation from the considered individual optimum values or exceeding of critical borders by yielding high values.

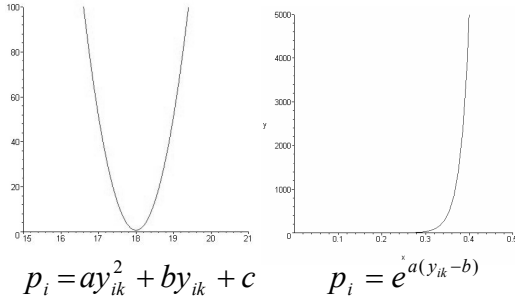


Figure 7. Preference functions.

Depicted in Figure 7 are two preference functions. On the left hand side a parabolic one, which typically filters optimal values and on the right hand side an exponential one, which filters out values beyond a critical border.

Typically, an analyst needs to assign the importance of single parameters. This is realized by weighting functions like the following.

$$Y_i(\mathbf{X}_i) = \sum_{k=1}^n g_i p_i(y_{ik}(\mathbf{X}_i)) \quad (3)$$

After application of preference  $p_i$  and weighting functions  $g_i$  we obtain dimensionless values  $Y_i$ , which are treated as a quality measure. Clearly, this quality depends on the choice of preference and weighting functions and is therefore an individual assessment of the analyst. Due to the definition of the preference function (see above) low quality values denote good quality circumstances and vice versa.

Next, the analyst defines his design space, spanned by input parameters  $X_i$  and the response  $Y_i$ . Overall, we have  $n_i * n_{mp} = 20$  grid points with discrete values. They allow the approximation of a response surface. Depicted below is the resulting surface obtained by application of the Kriging method (see above).

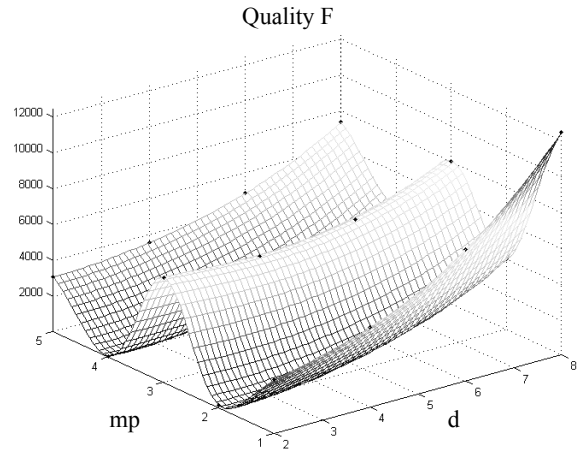


Figure 8. Quality over d and mp.

Here we can see parameter combinations which are beneficial (low values) with respect to our quality definition, as well as unfavourables. Some observations can be done directly out of this diagram:

- A global trend, where quality gets worse the larger the distance of the stairways gets.
- At measure point 3 there constantly is a disadvantageous quality

Sensitivities of the input parameters  $X_i$  give even more information. They are partial derivatives of the quality with respect to the input parameters. Figure 9 displays the sensitivity of the stairway distance d:

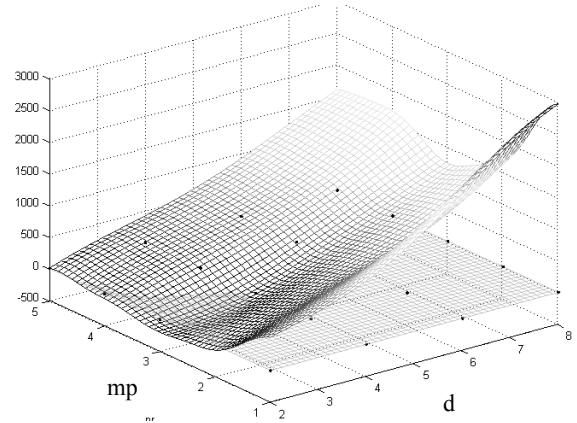


Figure 9. Sensitivity of d.

The most obvious observation here is that in the area around  $nr=1$  and  $d=8m$  the sensitivity of d is very high. This means that a modification of d at this input parameter combination will lead to significant changes of the quality. Thus, an analyst may identify areas where a great amount of improvement can be easily achieved.

For completeness Figure 10 depicts the sensitivity of the measure point location, which actually represents the climate at a specified location.



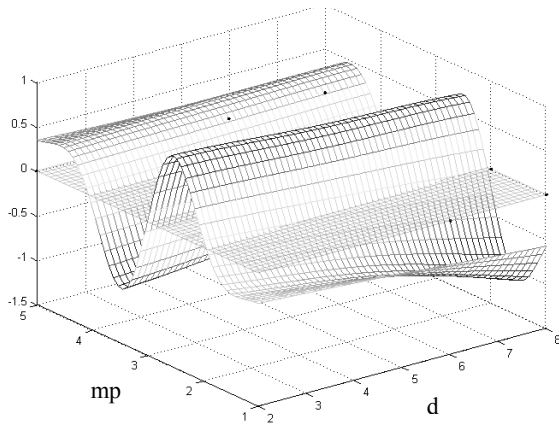


Figure 10. Sensitivity of mp.

An interesting observation can be made if the climate conditions are filtered out and we look on the quality of d only:

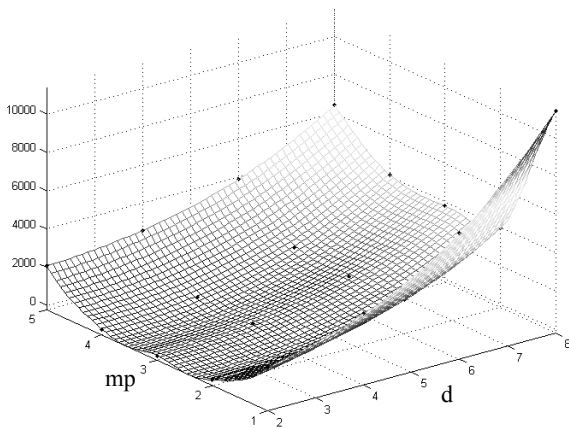


Figure 11. Quality over d and mp without influence of climate conditions.

Here, we can see that the disadvantageous circumstances must be caused by the climatic conditions. Displayed in Figure 12 is the climatic quality, split up in air velocity and air temperature:

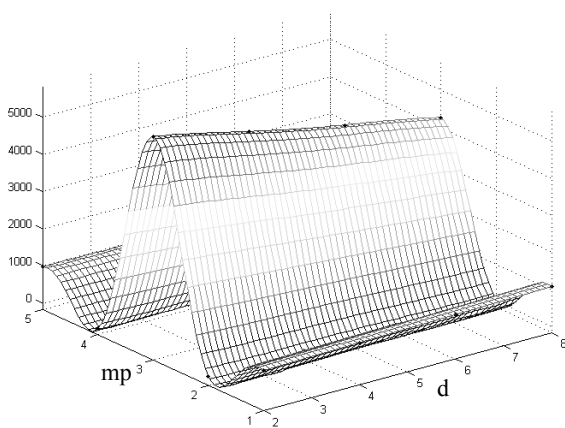


Figure 12. Quality over d and mp for air velocity only.

We find that the air velocity at  $p=3$  gets worse the closer d is, which is a contradiction to the global trend (see above).

It can be seen that the analyst has various possibilities of analyzing and assessing certain scenarios.

## 6 CONCLUSION

An analysis method was presented to support planning decisions on basis of sensitivity analysis in a heterogeneous planning environment in structural engineering.

Still, product model data management is a difficult problem. Since almost no applications support the check-in of their local data to a common shared product model, our above described DDD concept is used for this issue. For a manageable amount of different planner applications this is applicable, but beyond that the effort to keep track of the variable data format will become very high. Besides that, we seek cooperation with researchers of product model technology to develop a reliable solution for product model data transactions.

During the analysis a response surface is being approximated. Typically, many steps in a planning process in AEC cannot be made without human interaction. This limits the number of variants to be calculated. Thus, the amount of sampling points is relatively low and the approximation of a response surface may be relatively inaccurate. Otherwise, in many cases it would be sufficient to get a coarse idea how to enhance the quality of a product.

After finishing of implementation of the agent based collaboration environment our research concentrates on the analysis methods. Especially the human computer interaction shall be examined. Predominantly, this means to facilitate the usability for engineers or project managers, for example by linking the results to the visualization of a scenario.

Within the next year this method shall be tested and verified on some real case scenarios, especially within DFG funded priority program 1103 (see below).

## 7 ACKNOWLEDGEMENTS

The authors would like to thank the German Research Foundation (DFG) for funding the priority project "Network-based Co-operative Planning Processes In Structural Engineering" (SPP 1103) and in particular the included project "Sensitivity analysis of network-spanning product- and structural models with regard to planning processes in structural engineering" (BL 306/9-1), which are a great support for the presented research activities.



## REFERENCES

- Box, G. E. P. & Draper, N. R. 1987. Empirical Model-Building and Response Surfaces. New York : Wiley
- Ferber, J. 1999. Multi-Agent Systems . An Introduction to Distributed Artificial Intelligence. Harlow: Addison-Wesley
- Hartmann, D. et al. 2004. Integration of Product Model Databases into Multi-Agent Systems. Proceedings of the Xth International Conference on Computing in Civil and Building Engineering (ICCCBE-X), Weimar, Germany
- Romberg, R et al. 2003. Numerische Simulation auf Basis des IFC-Bauwerksmodells. 16. International colloquium about application of computer science and mathematics in architecture and civil engineering (IKM), Weimar, Germany
- Sacks, J. et al. 1989. Design and Analysis of Computer Experiments. Statistical Science 4(4): 409-435
- Saltelli, A. et al. 2000. Sensitivity Analysis. Chichester : Wiley
- Taguchi, G. 1986. Introduction to Quality Engineering: Designing Quality into Products and Processes. New York: Kraus International
- Weise M. Et al. 2003. Generalised Model Subset Definition Schema. Paper presented at the 20th CIB W78 Conference on Information Technology in Construction, Waiheke Island, April 2003, New Zealand.
- Woolbridge, M. & Jennings, N. R. 1995. Agent Theories, Architecture, and Languages: A Survey, In Intelligent Agents. Berlin: Springer-Verlag

