SIMULATION-BASED DESIGN BY HIERARCHICAL OPTIMIZATION

Ruchi Choudhary[1], P.Y. Papalambros[2], and Ali M. Malkawi[3]
[1]College of Architecture, Georgia Institute of Technology, Atlanta, Georgia-30332, USA
[2]Department of Mechanical Engineering, University of Michigan, Ann Arbor, USA
[3]School of Design, University of Pennsylvania, Philadelphia, USA

ABSTRACT

This paper critically examines the use of Analytic Target Cascading as a multi-level, hierarchical design optimization model for formulating simulation-based design tasks in architecture. A case study is used to illustrate the main steps involved in posing and solving an ATC problem. With an emphasis on problem formulation, this study is used as the basis of highlighting issues confronted while posing design analysis problems in a model-based systems framework.

INTRODUCTION

Simulation-based problems in architecture are often multi-criteria in the sense that they constitute simultaneous assessment of multiple performance requirements. When cast as decision-making problems, the expected outcome of a simulation-based problem is an aggregration of possible values for a set of design attributes that will yield desired performances. Such problems are ideally solved in one model so that trade-offs between different performance requirements may be explicitly understood. However, a multi-criteria problem whose function evaluations rely on different simulation models can quickly become too complex to be implemented within one model. In such cases, some form of problem decomposition becomes highly desirable. The main difficulty in decomposing a problem becomes that of aggregating design attributes that affect multiple criteria into a single compatible set. Without a rigorous framework, the aggregation can become non-intuitive – especially if the problem is large or complex.

A recent work extended Analytic Target Cascading (ATC), a model-based hierarchical optimization methodology, as means for achieving compatible solutions for large simulation-based problems found in the field of building performance (Choudhary et al., 2005). The ATC framework casts a problem into a set of decision-making subproblem organized hierarchically. Each subproblem constitutes an optimization problem and interdependencies among subproblems are included in the objective function of each subproblem. These interdepencies could include common design attributes among two or more subproblems or input-output dependencies among subproblems. The main assumptions underlying this work are (a) large simulation-based problems can be decomposed into a smaller set of interrelated subproblems, (b) each decomposed subproblem constitutes some design attributes that only affect its own behavioural response, (c) behavioural responses of a subproblem can be derived from a simulation as a function of its design attributes.

At a general level, this work falls into the area of computer-aided design. The broad community of computer-aided design in architecture has had a tradition of borrowing or extending design and decision-making paradigms used in engineering towards problem-solving in architecture. Examples of models appropriated towards facilitating and structuring the design analysis process would include artificial intelligence, collaborative design models, and systems design framework among other product development methodologies. Such efforts have no doubt contributed significant rigor and revealed new possibilities to a field where problem-solving by tradition relies on ad-hoc trial and error approaches. However, they also limit problems by often having a strong bearing on its formulation. By examining nuances, difficulties, and some impossibilties of implementing simulation-based building performance problems by ATC, this paper initiates the question of how much or whether at all problem-solving in architecture benefits from 'borrowed' models. The next two sections of this paper outline the ATC process and illustrate the formulation of one design scenario as an ATC problem. This work demostrates how ATC functions in the context of setting design attributes and determining average temperature setpoints for meeting specifications for HVAC sizing and thermal comfort at pre-design stage. For more details on ATC the reader is referred to Papalambros et al. 2002. A thorough description of the case study, may be found in Choudhary et.al 2005. The third section examines the ATC framework in the context of simulation-based problems found in architecture, focusing on those parts of problem formulation where differences between the nature of the problem and the method need to be reconciled.

ANALYTIC TARGET CASCADING

An ATC problem is set-up by decomposing the problem hierarchically into systems, subsystems, components, and so on. Each decomposed element is formulated as a separate design optimization problem including its local design variables, parameters, objectives, and constraints. Appropriate analysis models (these may be simulation tools) are associated with each decision model in the hierarchy. An analysis model evaluates design decisions by taking variables and parameters as input and returning its performance or behavioral response as output.

In the ATC process, top-level performance specifications are propagated down to lower level elements, which are then optimized to match the targets as closely as possible. The resulting responses are rebalanced at higher levels by iteratively adjusting targets and designs throughout the hierarchy until specified termination criteria are met. Figure 1 shows the ATC formulation in the standard index notation given by Michelena et.al. 2003.

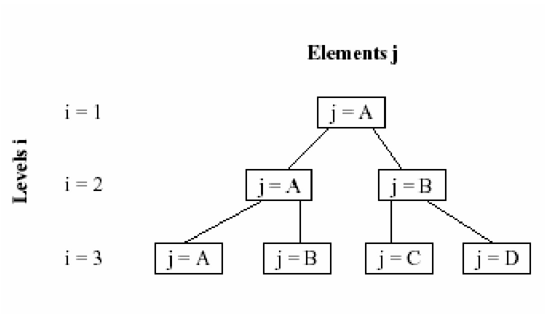

Figure 1: Standard index notation for a hierarchically partitioned problem

For target matching a problem S_{ij} for the j-th design model at the i-th level, the general formalization of the optimization model is stated as:

$$\underset{\tilde{\mathbf{x}}_{ij},\mathbf{y}_{(i+1)j},\varepsilon_{ij}^{R},\varepsilon_{ij}^{y}}{\text{minimize}} \left\|\mathbf{R}_{ij}-\mathbf{R}_{ij}^{U}\right\|_2^2+\left\|\mathbf{y}_{ij}-\mathbf{y}_{ij}^{U}\right\|_2^2+\varepsilon_{ij}^{R}+\varepsilon_{ij}^{y}$$

subject to:

$$\sum_{k\in C_{ij}}\left\|\mathbf{R}_{(i+1)k}-\mathbf{R}_{(i+1)k}^{L}\right\|_2^2\le\varepsilon_{ij}^{R},$$

$$\sum_{k\in C_{ij}}\left\|\mathbf{y}_{(i+1)k}-\mathbf{y}_{(i+1)k}^{L}\right\|_2^2\le\varepsilon_{ij}^{y},$$

$$\mathbf{g}_{ij}\left(\overline{\mathbf{x}}_{ij}\right)\le\mathbf{0},$$

$$\mathbf{h}_{ij}\left(\overline{\mathbf{x}}_{ij}\right)=\mathbf{0},$$

where:

- $\overline{\mathbf{x}}_{ij}=[\tilde{\mathbf{x}}_{ij},\mathbf{y}_{ij},\mathbf{R}_{(i+1)k_1},...,\mathbf{R}_{(i+1)k_{c_{ij}}}]^T$ is the vector of all decision variables of element j at level i,
- $\mathbf{R}_{ij}=\mathbf{r}_{ij}\left(\overline{\mathbf{x}}_{ij}\right)$, where $\mathbf{r}_{ij}$ is the vector function that represents the analysis model. It calculates the responses for element j at level i by taking in all its decision variables as input.
- $C_{ij}=\{k_1,...,k_{c_{ij}}\}$, and c_{ij} is the number of child elements,
- $\tilde{\mathbf{x}}_{ij}\in\Re^{n_{ij}}$ is the vector of local decision variables for element j at level i,
- $\mathbf{y}_{ij}\in\Re^{l_{ij}}$ is the vector of linking variables for element j at level i,
- ε_{ij}^{R} is the tolerance variable for consistency of targets set at element j level i and the responses of j's children,
- ε_{ij}^{y} is the tolerance variable for consistency of linking variables coordinated at element j level i for child elements at the $(i+1)th$ level,
- $\mathbf{R}_{ij}^{U}\in\Re^{d_{ij}}$ is the vector of response values cascaded to element j at level i as targets from its parent at level $(i-1)$,
- $\mathbf{y}_{ij}^{U}\in\Re^{l_{ij}}$ is the vector of coordinating linking variables for the linking variables in the children of element j at level i. This vector includes one copy of each linking variable from all of element j's children.
- $\mathbf{R}_{(i+1)k}^{L}\in\Re^{d_{(i+1)k}}$ is the vector of response variable values cascaded to the element j at level i from its $k-$th child at level $(i-1)$,
- $\mathbf{y}_{(i+1)k}^{L}\in\Re^{l_{(i+1)k}}$ is the vector of linking variable values cascaded to the element j at level i from its $k-$th child at level $(i-1)$,
- $\mathbf{g}_{ij}:\Re^{d_{ij}+n_{ij}+l_{ij}}\rightarrow\Re^{v_{ij}}$ and $\mathbf{h}_{ij}:\Re^{d_{ij}+n_{ij}+l_{ij}}\rightarrow\Re^{s_{ij}}$ are vector functions representing inequality and equality design constraints,
- $\|\quad\cdot\quad\|_2^2$ represents the square of the l_2 norm.

ILLUSTRATIVE STUDY

This particular study constitutes thermal performance and HVAC sizing (determination of thermal loads) for one floor in a health care facility. This problem is based on a survey of an existing health care unit and constitutes zones that have varying functional requirements and thermal conditions. The subproblems formulated for this study incorporate performance requirements for AHU capacity, energy consumption, thermal comfort, and ventilation. Hence, it also shows how dependencies between HVAC sizing, energy and thermal comfort, and CFD analysis tools can be coordinated within the ATC process.

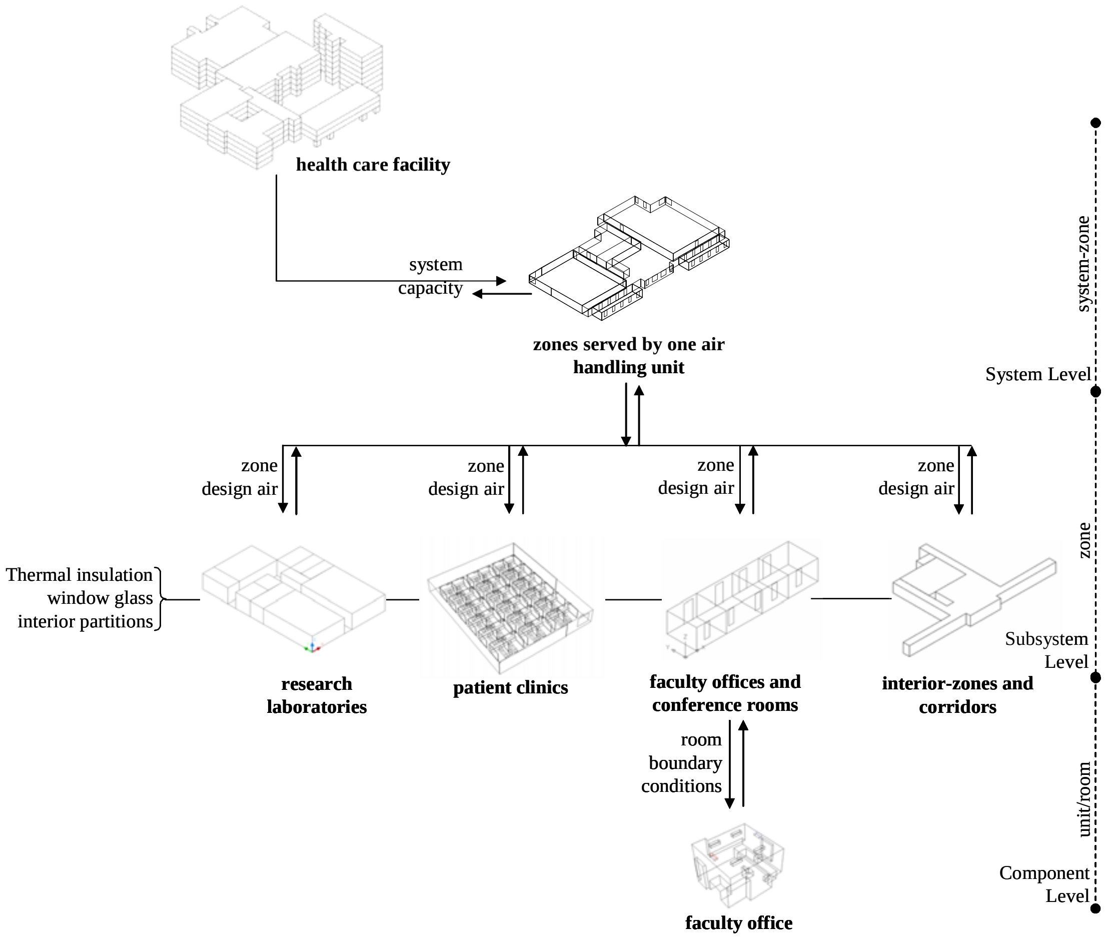

Figure 2: Hierarchical decomposition of the thermal and HVAC design problem

The design problem was posed as a three-level ATC problem (system, zone, and unit), where each level requires different scales of analyses (Figure 2). Target values for overall HVAC capacity were specified at the system level. The optmization model at this level is solved to match target values with the capacity of the HVAC system computed with respect to the design airflow rates for each zone. Optimal values of the zone airflow rates were then passed down to each zone at the lower level as target airflow rate. The subproblems at the zone level were solved for meeting the target airflow rate as closely as possible. The zone level problems are also constrained to satisfy thermal comfort requirements at all occupied hours of the design days. The variables at this level include properties of construction elements (such as insulation thickness and glass conductance), thermostatic setpoint temperatures, and zone supply temperatures for all rooms in zone. Common design variables (linking variables) shared at the system level include wall insulation, glass type (its conductance), and thermal properties of interior partitions. All design attributes related to the physical configuration were assumed to be part of the problem definition and held as fixed parameters. At the third level in the ATC hierarchy, one unit in a zone was selected for further analysis for examining ventilation efficiency at a specified timestep. The unit level problem configures the air inlet and outlet diffusers for meeting specifications of boundary conditions while satisfying required temperature and velocity distributions in the space.

All models have been implemented in MATLAB. The top and the subsystem level problems are solved using sequential quadratic programming (SQP) from the MATLAB optimization toolbox (www.mathworks.com), and the component level problem is solved using superEGO (Sasena, 2002). The top level analysis model derives the total system capacity by using the system and zone sizing objects in EnergyPlus. The subsystem level analysis models also use EnergyPlus, but to simulate a given zone for computing design heating and cooling airflow rates, thermal comfort values, and mean air temperatures. Values for heating and cooling energy, surface temperatures, and heat gains from internal loads can also be derived from EnergyPlus at this level. The subsystem analysis model aggregates this information in required forms, such as maximum and minimum values of mean air temperature, peak airflow rates, etc. The component level analysis model uses FLUENT (www.fluent.com), a CFD simulation tool, for computing local temperature and velocity profiles for a given air-distribution design. Data output by the CFD simulation is also processed by the analysis model into indices such as average draught temperatures, maximum velocity, and air diffusion performance index. All decision variables, targets, responses, and constraints included in the decision models are scaled to the same order of magnitude (between 0-1).

The target cascading process for this thermal and HVAC design problem can be summarized as follows: The top level problem is solved and design airflow rates of each zone are cascaded down as subsystem level targets. Then the four subsystem level problems are solved sequentially, one after another, for meeting the cascaded target design airflow rates. Once all subsystem level problems are solved, their responses are passed back to the top level problem, and this bi-level problem is solved iteratively until the deviation terms become smaller than a specified tolerance. Once the bi-level problem terminates with a feasible solution, thermal boundaries of the office are cascaded down to the third level as targets. The third level problem is solved for meeting the targets it receives from the subsystem level model. The component level solution is passed back to the top level as lower level responses. With the feedback from the component level, the top and the subsystem problems are now solved again to match the thermal boundaries passed back from the lower level. This process is continued until the variables and responses at each level stop changing over subsequent iterations.

Table 1: ATC results II – system-level targets and responses

SYSTEM LEVEL TARGETS (AHU SIZING DESIGN)	TARGETS	RESPONSES
Total objective function value at the top level (scaled)	0	0.46
Heating Capacity (W)	0	1819.2
Cooling Capacity (W)	0	101422
Deviation between Subsystem Targets and Responses ε_s^R (scaled)	0	**0.0038**
Deviation among Subsystem Linking Variables ε_s^y (scaled)	0	0.0

Table 2: ATC results II - values of targets and responses at the subsystem-level

SUBSYSTEM LEVEL SPECIFICATIONS		SUBSYSTEM TARGETS	SUBSYSTEM RESPONSES			
			Research Labs.	**Patient Clinics**	**Offices**	**Interior Zones**
RESPONSES	Heating design airflow rate (cfm):					
	Research Laboratories	4.2	0			
	Patient Clinics	2.1		0		
	Offices	21.2			42.4	
	Corridors	0				0
	Cooling design airflow rate (cfm):					
	Research Laboratories	11548	11548			
	Patient Clinics	8475		8617		
	Offices	1271			1271	
	Sterilization	2542				2638
	Corridors	4238				4445
	Utilities	1673				1663
LINKING VARIABLES	U-value - 4” partition (W-m/K)	0.58	0.58		0.58	
	U-value-interior wall (W-m/K)	1.69	1.7	1.7	1.7	
	Insulation layer-north wall (m)	0.14	0.14			
	Insulation layer-south wall (m)	0.13		0.138	0.13	0.13
	Glass thickness (m)	0.006	0.006	0.006	0.006	0.006
	Glass (solar) transmission	0.4	0.4	0.4	0.4	0.4
Deviation between Subsystem Targets and Responses					0.04	

Table 3: ATC results II - values of targets and responses at the component-level

COMPONENT LEVEL SPECIFICATIONS	COMPONENT LEVEL TARGETS	COMPONENT LEVEL RESPONSES
Cooling supply temperature (^oC)	14.8	14.69
Inside surface temperature of south wall (^oC)	26.4	26.5
Inside surface temperature of window wall (^oC)	29.63	29.63
Inside surface temperature of north wall (^oC)	26.54	26.6
Inside surface temperature of east wall (^oC)	27.2	27.2
Inside surface temperature of west wall (^oC)	27.0	27.2
Inside surface temperature of floor (^oC)	27.0	27.0

As the first case, unattainable target values and were specified for minimizing the HVAC size and all targets at each level were equally weighted. However, the results showed a large deviation between targets cascaded down to the zone level and responses passed back up – meaning that the solution obtained is not consistent. So in the second case higher weights are assigned to the lower level models so that "consistency between subproblems" is prioritized over minimizing overall AHU capacity. In addition, all decision models were given the solutions derived from the previous ATC run as the starting point of the optimization. The top and the subsystem level problem converged in eight iterations. The top level problem required 85-100 function evaluations to converge. The number of function evaluations for the subsystem level zone models ranged from 150-300. The component level problem converged in approximately 150 function evaluations and took the longest since it used a CFD model for analysis (the updated weights and good starting points increased speed of convergence). Table 1-3 show the solutions at two levels in ATC hierarchy. As shown, all targets are met within acceptable accuracy. This problem if solved "all-at-once" would constitute 47 decision variables and require coordination among three analysis models in one optimization model. The ATC process allowed evaluation of design decisions by specific focus and analysis, and structured the problem such that it can be easily isolated for further design. The main outcomes of this process are consistent values of performance specifications and compatible values of design variables for all models included in the hierarchy.

SIMULATION-BASED DESIGN BY ATC

The application illustrated in the previous section is used as the basis for examining the limits posed by the ATC framework on problem formulation. Topics covered in this discussion include: hierarchical organization of decision-making tasks in a simulation-based design scenario, setting up the decision and analysis models for implementing the ATC process, target setting, and interpretation of ATC results.

'Hierarchical' Decomposition

Breaking down an analysis task into a set of subproblems can be a fairly straightforward step in the building design context. As shown in the implementation of thermal and HVAC design, buildings can be decomposed by physical boundaries among functionally distinguishable zones and/or by environmental differences such as shading and orientation of the enclosing surfaces. When the problem is multi-disciplinary, i.e., when more than one performance criteria are considered , a physically decomposed zone may be further segregated by a performance aspects or by the analyses involved. The main constraint in such cases is that each decomposed subproblem must constitute some design variables that are not shared with any other subproblem. It is also desirable to limit shared or linking variables among subproblems to a reasonable number. Hence when two or more performance aspects are functions of common variables, then a good strategy is to include them in the same optimization model: In the ATC framework, a subproblem may be associated with two or more analysis models if required. Sometimes design decisions are made sequentially in a workflow routine, and could be also a basis for decomposing the problem into an ordered set of subproblems. However, sequential decomposition assumes uni-directionality of design information, which is hard to generalize as a process logic for architectural design programs.

These decompositions of complex or large design scenarios are commonly practiced in design consulting firms by using intuition and practical experience. Alternatively, a problem may also be decomposed by rigorous evaluations of dependencies among decision variables and functions included in the problem formulation (see Michelena and Papalambros, 1997). This is an engineering design approach requiring an enumeration of all decisions, targets, and constraints included in the problem, and is extremely useful for large problems where functional dependencies can be hard to track by insight alone. Choosing the appropriate decomposition strategy is problem dependent and derived from the type of decision variables and performance specifications included in the problem. For example, if physical configuration and geomtery of spaces was included in the thermal design scenario, it would require a revision of decomposition strategy based on functional dependencies of size and location of walls.

$$\begin{array}{c}
\boxed{\underset{\tilde{\mathbf{x}}_{s},\mathbf{R}_{ss}}{\text{minimize}} \left\| \mathbf{R}_{s}^{U} - \mathbf{R}_{s} \right\| + \left\| \mathbf{R}_{ss}^{L} - \mathbf{R}_{ss} \right\|} \\
\mathbf{R}_{ss}^{U} \Big\downarrow \qquad \Big\uparrow \mathbf{R}_{ss}^{L} \\
\boxed{\underset{\tilde{\mathbf{x}}_{ss}}{\text{minimize}} \left\| \mathbf{R}_{ss}^{U} - \mathbf{R}_{ss} \right\|}
\end{array}$$

Figure 3: A two-level hierarchical organization in the ATC framework

The real challenge of decomposing the thermal and HVAC study for ATC implementation was in the hierarchical organization of the decision-making tasks and in identifying key links among them. The target cascading framework is based on the premise that design decisions made at a level are responses of a lower level design problem. This condition is the main determinant in the hierarchical organization of the decomposed subproblems: Figure 3 shows a bi-level hierarchical organization in the ATC framework. In this decomposition $\mathbf{R}_{ss}$ is a vector of decision variables at system level s. The vector $\mathbf{R}_{ss}$ is passed down as target value $\mathbf{R}_{ss}^U$ that the subsystem model ss must match. Given this requirement, some general guidelines can be derived for hierarchical organization of decision models in a simulation-based design scenario.

A decision-making task that requires a simulation for evaluating design decisions can be hierarchically ordered by examining input-output relationships of the simulations: A simulation 'n' whose output is required as input for another simulation 'm' will be its child element. As an example, Figure 4 shows the decisions input and responses output from four different simulations. In this example, response outputs A1, A3, and A4 from simulations II, III, and IV are input requirements of simulation I. In addition, simulations II and III share common input variables B1 and B2. Simulations III and IV also share common input variables C1 and C2. Based on these observations, this problem is hierarchically organized in a bi-level model shown in Figure 5. Decisions variables input to simulation I constitute the top level problem at level i=0. This level has two child elements at level i=1. The first element at level i=1 determines the values of variables B1, B2, B3 and B4 using simulation II, and the second element finds optimal values of B1, B2, C1, and C2 based on responses computed by simulation III and IV. These two elements share linking variables B1 and B2, which are coordinated by their parent element at level i=0.

In addition to input-output links among simulations, other characteristics that influenced the model shown in Figure 5 include: (a) identifying common input variables between two or more simulations as linking variables, (b) identifying a common parent element for coordinating those linking variables, and (c) mapping simulation tools to appropriate decision models in the hierarchy.

While in principle such a study should be used to guide hierarchical organization, it is not always possible to follow it strictly. Subproblems may be ordered based on notions of physical hierarchy in buildings (building, zone, sub-zone, room), but still a lower order element requires some information of its higher order components. For instance, a building as a lower order element requires some information about its zones (this is also the basis for organizing the subproblems for the case described in this paper). Without a seamless input-output relationship, this sharing can violate the basic structure of ATC. In such cases, a good understanding of the limits of the methodology becomes critical for getting any meaningful results. Subproblems may also be ordered by the type or resolution of analysis tasks involved. For example, the illustrative study distinguishes the vertical order of decision models by analysis type. In the original presentation of the ATC literature simpler models are generally preferred at higher levels to provide quick evaluations of the proposed designs while more complex models are mapped to the lower levels to provide detailed evaluations (Kim, 2001).

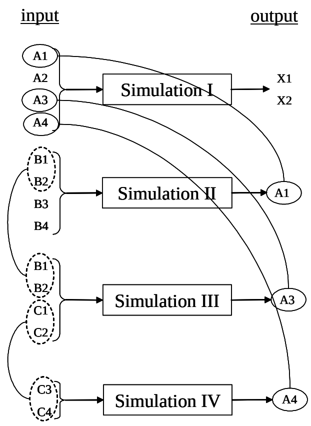

Figure 4: Example of input-output relationships as basis of hierarchical-organization

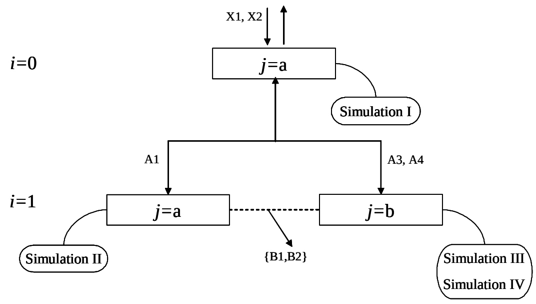

Figure 5: Hierarchical organization of the analysis problem shown in figure 4

Therefore, while the ATC framework imposes some general rules of formulation, the hierarchical organization of decomposed subproblems requires strategic design following a good insight into the problem and an examination of the analysis tasks involved. Still, there is room for work on assisting the user towards a robust model.

Implementation Setup

Another important step in laying out a problem for target cascading is 'model building'. Once a problem is decomposed, organized in a hierarchy, and key links among subproblems have been identified, decision and analysis models have to be formulated. For each subproblem, building a decision model constitutes formulating the optimization problem, choosing appropriate optimizers to solve it, and finally, posing the subproblem in the format required by the optimizer. The analysis model serves as an interface between the optimization model and the simulation tool required for evaluating subproblem decisions. So, preparing an analysis model means: (a) identifying simulation tools that will evaluate decision variables and return data which can be processed into responses required by the decision model, (b) processing the decision variables and design parameters into an input file format required by the simulation tool, and finally, (c) post-processing the data returned by the simulation tool into responses required by the decision model. Setting up these models can be extremely time-consuming, arduous, and almost impossible for a modeler in design practice. Therefore, a general computational framework that can assist in setting up the ATC problem is necessary before its benefits can be affected into practical problem-solving scenarios.

The 'implementation setup' prepared for the case studies presented in this paper uses three different optimizers, and three different analysis models in an ATC model prepared in the MATLAB environment (Figure 6). This setup is only generalizable for solving different design scenarios that include similar analysis tasks, but may also be suggested as a prototype for future, more general implementation of an ATC framework as a toolbox.

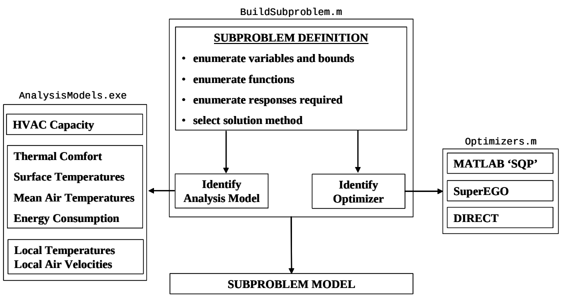

Figure 6: ATC framework implemented for the thermal and HVAC case study

Computational Run-Time

The ATC framework is designed for large and complex design scenarios that cannot be solved altogether in a rational framework and are difficult to implement by trial and error coordination of multiple analysis tasks involved. Therefore, even though computational run-time required by the ATC process may be long (1 day for the illustrative study), it is justified for finding solutions to problems that are not solvable otherwise, or at least not with comparable rigor. Still, it is useful to keep in mind model characteristics that will reduce the total solution time and derive quick solutions.

In the context of this case study, the time taken by analysis models for each function evaluation had the most significant influence on the total run-time. Therefore inexpensive models are generally preferred in the target cascading process. In building simulation, simpler analysis tools are often inaccurate or less sensitive to changes in input parameters. When evaluating a candidate simulation tool to be used in the ATC process, it is equally important to check function evaluation time as well as sensitivity of the tool to changes in decision variables and accuracy of the results. In cases where inexpensive simulations are not available, simpler models can be derived by using surrogate modeling techniques. What is ideally required is a repository of quick and responsive analysis tools that can be used for making relatively "quick" decisions and setting targets before detailed and individual design.

Other factors that influence the total run-time include dimensionality of individual subproblems and solution methods used for solving them. Since ATC is really a pre-design step in which all common decisions and inter-related performance targets must be determined before further design, it is usually good to include only those decisions and design relations that are essential in the inter-related design scenario. Subproblems with lower dimensions will have fewer degrees of freedom and therefore converge to a solution faster. Also, some solution methods have a faster convergence rate than others depending on their search processes, and therefore must be selected with due consideration of model properties as well as speed of convergence. For example, in the HVAC design case we used a surrogate-based optimization method (superEGO; Sasena, 2002) for reducing the number of function calls to the simulations.

Target Setting and Interpreting ATC Results

The building simulation community understands optimization in the setting of maximizing or minimizing a specified set of building performance goals. By proposing ATC as a hierarchical optimization framework, this paper suggests using optimization methods for minimizing deviations from given problem specifications and for deriving values of all other performance goals included in the model. Implicit in adopting the ATC model is the assumption that in practice a simulation-based design scenario is overseen by a specialist who, by experience and professional skill, is able to translate some of desired performance goals into numerical values as targets. Building performance targets may also be dictated by government regulations or client/corporate decisions. In cases where it is difficult to gauge reasonable values of performance goals, unattainable targets can be posed (as in the formulation of thermal and HVAC design problem).

If feasible targets are specified, the ATC process will derive consistent solutions at all levels in the ATC hierarchy. In complex design cases it is often hard to know if a consistent solution is possible with specified values of targets. Also, as shown in the thermal and HVAC design study and proven by Michalek and Papalambros (2005), a strictly consistent solution will not be derived when targets are unattainable. If inconsistencies in the ATC results are above acceptable limits the problem has to be modified after examining the results: Targets that are specified by intuition may be infeasible, or the design space may not allow targets to be met, or there may be numerical difficulties with the values of weighting coefficients specified at each level. In such cases the choice between changing the design space and/or changing the target values depends on the problem and the basis on which both of them were set. In some cases a trial ATC is useful as a basis for setting target values and then re-running the problem. Michalek and Papalambros (2003) also suggest a weighting update method that can derive solutions within user-specified tolerances by finding and updated values of weighting coefficients during the ATC process.

CONCLUDING REMARKS

This paper investigated Analytic Target Cascading as a hierarchical design optimization model for facilitating large-scale and simulation-based design tasks in architecture. It builds on the premise that benefits of numerical optimization techniques can be used as explicit and rigorous means of coordinating solutions from subproblems into a single compatible set. Its main accomplishment is in presenting numerical optimization in a broadened context of collaborative decision-making in building simulation for supporting interrelated building analysis scenarios. Specific benefits of this work to the building simulation community are:

- Organization of multi-criteria design analysis tasks in a linked hierarchical structure by key interrelationships.
- Formulation of a rigorous goal-driven approach to integrating and invoking simulation tools by specified performance targets.
- A formal model enabling explicit means of achieving consistent and concurrent design solutions in scenarios that require coordination among multiple performance specifications.

The case study presented in this paper also revealed several areas where it becomes difficult to apply the rules inherent in formal engineering design models. The paper highlighted them as areas that need special focus for using ATC in the building design context. A few other issues that merit mention here include: (a) investigating the effect of including local performance targets in subproblems, (b) investigating the quality of results for mixed-discrete problems (c) linking the current prototype to ATC models of other design aspects in model-based building evaluations, and (c) building analysis models that are identified by the performance aspect and decision variables.

From an implementation perspective the main requirement is a model that can assist the user in problem formulation and in hierarchical organization of decision-making tasks by rigorous criteria. In addition, building a general framework that contains a repository of decision and analysis models, thereby assisting the modeler in representation of the problem are also desirable. Some of these issues are practical and can be generalized as problems encountered in any model-based design domain. However issues related to hierarchical organization and robust model-building are domain dependent and require special attention and further research when extending this methodology to the building design context.

REFERENCES

Choudhary, R., Malkawi, A., and P.Y.Papalambros. 2005. Analytic target cascading in simulation-based building design, *Automation in Construction*, in press.

Choudhary, R., Papalambros, P. Y., and A. Malkawi. 2005. A Design Optimization Approach for Matching Thermal and HVAC Performance Targets with Simulation-based Decisions, *Architectural Engineering and Design Management*, Vol 1 (1).

Kim, H. M. 2001. Target Cascading in Optimal System Design. Ph. D. dissertation, University of Michigan, Ann Arbor, USA.

Michelena N., H. Park, and P.Y. Papalambros. 2003. Convergence Properties of Analytic Target Cascading. *AIAA Journal.* 41, no.5, 897-901.

Michalek, J. J. and P.Y. Papalambros. 2005. Technical Brief: Weights, Norms, and Notation in Analytical Target Cascading, to appear in the ASME Journal of Mechanical Design.

Michelena, N., and P.Y. Papalambros. 1997. A Hypergraph Framework for Optimal Model-Based Decomposition of Design Problems. *Journal of Computational Optimization and Applications*. 8, no. 2: 173-196.

Papalambros, P.Y. 2001. Analytical Target Cascading in Product Development. *Proceedings of the 3rd ASMO UK / ISSMO Conference on Engineering Design Optimization*. Harrogate, North Yorkshire, England, July 9-10.

Sasena, M. J. 2002. Flexibility and Efficiency Enhancements for Constrained Global Design Optimization with Kriging Approximations. Ph.D. dissertation, University of Michigan, 2002.