FINEL 1103
pp: 1--24 (col.fig.: Nil)
PROD. TYPE: COM
ED: Smitha
PAGN: Vidya – SCAN: Mangala

Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Finite Elements in Analysis and Design ▮▮▮ (▮▮▮▮) ▮▮▮–▮▮▮
FINITE ELEMENTS IN ANALYSIS AND DESIGN
www.elsevier.com/locate/finel

Toward simulation-based design

Mark S. Shephard[a,*], Mark W. Beall[b], Robert M. O'Bara[b], Bruce E. Webster[b]

[a] *Scientific Computation Research Center, Rensselaer Polytechnic Institute, Troy, NY 12810-3590, USA*
[b] *Simmetrix, Inc., 10 Halfmoon Executive Park Drive, Clifton Park, NY 12065, USA*

Received 4 October 2003; accepted 30 November 2003

Abstract

This paper discusses the technologies needed to support the application of simulation-based design. Emphasis is placed on the technical components that must be added to existing CAD and CAE tools to enable the application of simulation-based design. These components include a simulation model manager, simulation data manager, adaptive control tools and simulation model generators. The application of these technologies to automotive climate control system design is demonstrated.

Keywords: Simulation-based design; CAE

1. Introduction

The introduction of computer-aided design (CAD) and computer-aided engineering (CAE) has had a major impact on the ability to produce better products. However, they have not come close to reaching their full potential with respect to supporting the engineering design process. This is because a few key components of the process have not been properly defined and developed. This paper provides a description of the missing components and discusses an effort to develop those capabilities and to integrate them with existing CAD/CAE components to produce the capabilities needed to support the application of simulation-based design.

Simulation-based design is a process in which simulation is the *primary* means of design evaluation and verification. When coupled with appropriate validation processes executed during the development of a simulation-based design system, the resulting capabilities can provide companies the ability to design superior products in less time and at lower costs. The application of simulation-based design in today's engineering design processes tends to be limited to two classes of situations. The first

* Corresponding author. Tel.: 518 276 6795; fax: 518 276 4886.
E-mail address: shephard@scorec.rpi.edu (M.S. Shephard).

0168-874X/$ - see front matter © 2003 Published by Elsevier B.V.
doi:10.1016/j.finel.2003.11.004

consist of situations where the analysis tools required to determine the requested design performance parameter are well qualified and operate directly off the information in the computer-aided design (CAD) system. An example of this type is the geometric interference checking such as applied by Boeing in the design of the 777. The second consist of situations where the cost associated with the application of the classic methods of prototype construction and test is prohibitively time consuming and expensive. An example in this class is automotive crash worthiness analysis, which all the major automotive manufacturers have invested in and now rely on heavily. The goal of the efforts described in this paper are to provide the technologies and associated software tools needed to support the application of simulation-based design for a wide variety of design situations requiring performance parameter evaluations as complex as those in crash worthiness analysis with the ease at which geometric interference checks can be performed with current CAD tools.

When one considers the efforts that have gone into the development and application of product data management (PDM), computer-aided design (CAD) and computer-aided engineering (CAE) technologies it is reasonable to question why these capabilities do not properly meet the needs of simulation-based design. From a basic level these technologies can address all the functionality needed in the process. They are capable of controlling the information within the design process, define the detailed geometric models and associated data, and execute the engineering analyses to obtain useful estimates of performance parameters. A key missing component is due to the differences in the model descriptions for the design model in the CAD system and the analysis model in the CAE system. Historically, this mismatch has meant the independent construction of each CAE analysis model independent of the CAD model. This process is not only time consuming and expensive, it is error prone and does not effectively support the correlation of design and analysis information in the design management system. In some areas, like finite element analysis, the introduction of automatic mesh generation technologies properly integrated with the CAD definition goes a long way to addressing the construction of the analysis models from the design information. This, however, is only a first step in addressing the needs of simulation-based design.

The effective integration of engineering analysis into the design process requires the ability to define simulation models of various levels of fidelity from early in the design process and to have the information determined by those analyses influence the design process. These functions are not well supported by the boundary representation used by the solid modelers in the CAD system. Although the feature modeling capabilities that are available with several CAD systems have some of the functionality needed for this process, they fall short of properly addressing the needs. The design-component model described in Section 3 is being used to address these needs within simulation-based design process.

Since many current CAE tools require substantial expertise to obtain reliable results, and design engineers should not be expected to be CAE experts, additional capabilities are required to reliably automate the application of the engineering analyses. The methods that are under development require the use of solution information from previous analyses to determine how to adaptively improve the analysis model to provide the requested level of accuracy. Since a full set of adaptive procedures are not available for all classes of analyses, it is also important to encode the knowledge of CAE experts for those cases.

There are a wide range of engineering analyses associated with simulation-based design. They range from analyses requiring only a definition of the geometry of the design and any motions parts will undergo, to solving mathematical models defined in terms of differential equations over the geometric domain defined where appropriate material parameters, loads, boundary conditions and initial conditions must be considered. The emphasis of the methods described in this paper is on simulation-based design processes requiring the solution to various sets of partial differential equations solved using techniques ranging for idealized 1-D “engineering models” to generalized numerical analysis methods like finite element procedures.

Section 2 overviews the key components of a simulation-based design system with emphasis on four components of the simulation model manager, simulation data manager, simulation model generators and adaptive control tools. Sections 3–6 discuss each of those four components in more detail. Section 7 then discusses how these technologies are being used in the design of automotive climate control systems.

2. Simulation-based design system components

Fig. 1 provides an overall view of the functional components needed to support SBD. These components include the product data managers, computer-aided design tools and computer-aided engineering tools heavily used in industry. It also includes four new components (those within the dashed line in Fig. 1) which are collectively referred to as the Simulation Environment for Engineering Design (SEED). The interactions indicated by the arrows in Fig. 1 represent only those interactions activated when an engineering simulation is executed.

The function of the existing components and their interaction with SEED are summarized as:

- The Product Data Manager initiates requests to have engineering performance parameters evaluated for a given design. SEED is responsible for determining those parameters and returning the information in a form meaningful to the design process.
- The Computer-Aided Design (CAD) Tools house the design model and support interactions with that model, with the key component of interest here being the definition of the geometric domain. Today’s commercial CAD systems include a solid model defined in terms of a boundary representation. They will often also include some form of feature model that will help support the design process.

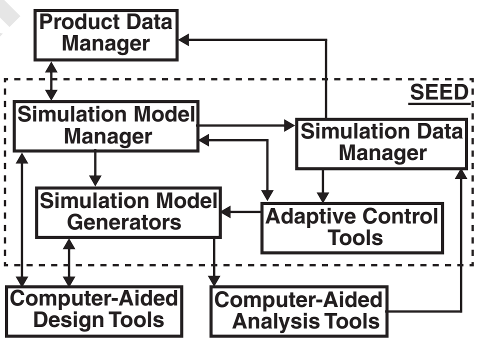

Fig. 1. Functional components needed for simulation-based design with the four new components in the dashed box defining the Simulation Environment for Engineering Design (SEED).

- The Computer-Aided Engineering (CAE) Tools include the engineering analysis programs in use today (e.g., 1-D engineering performance models, finite element analysis, etc.). The key characteristic of the CAE tools is that they operate on a model of the design specific to the analysis method being used (e.g., basic graph with parameters, a mesh with parameters, etc.) which is not necessarily maintained as a part of the design representation. For purposes of this discussion, it is assumed that when used in the appropriate manner and provided the appropriate analysis model, the CAE tools can compute the raw information needed to answer the given performance request.

The functional components of SEED are:

- The Simulation Model Manager that is responsible for controlling the overall simulation process and providing the base simulation model definition information needed by the other components.
- The Simulation Data Manager that is responsible for housing the simulation result information as needed during the simulation process.
- The Adaptive Control Tools that are responsible for selecting and controlling the simulation models so that the estimates of the performance parameters are to the level of accuracy requested.
- The Simulation Model Generators that are responsible for constructing the models used by the CAE tools accounting for the current design information and adaptively defined analysis model construction information.

Within a simulation-based design process, the product data manager communicates its request for performance parameter evaluation to the simulation model manager. The simulation data manager communicates information about the processes performed back to the product data manager. The simulation model manager executes two way communications with the computer-aided design tools. It provides the simulation model generators with information needed in constructing the input to the CAE tools, and provides the simulation data manager information needed to evaluate the requested performance parameters based on the CAE simulation results. The simulation model manager interacts with the CAD tools to ensure that a detailed geometric model representation consistent with the component-model definition appropriate for the simulation models needed is available.

The two way communications between the simulation model manager and adaptive control tools are used to communicate information on the performance information requested and the required model information and information on additional model change information that is required in the process of constructing the simulation models. The adaptive control tools communicate with the simulation model managers to indicate the simulation models that need to be generated. The adaptive nature of the simulation control process requires communication with the simulation data manager to obtain the results information needed to determine modifications needed in the simulation model for the next analysis iteration.

The simulation model generators obtain the information on which simulation models are to be constructed from the adaptive control procedures. To construct the simulation models, information is needed from the simulation model manager. In addition, since in many cases (e.g., finite element analysis) the model generation procedures need detailed geometric information, this information is obtained from an interaction with the CAD tool. This interaction is indicated as two-way since the existence of an appropriate boundary representation and operator libraries for the detailed geometric domain models exist such that the simulation model generation tools can obtain all needed domain information through simple interrogations of the geometric model. The simulation model manager is responsible for providing the information needed by the CAE tools in the form they require.

The CAE tools provide the simulation results information to the simulation data manager which is capable of extracting the results information needed to control the adaptive control processes or to evaluate the requested performance parameters in a form needed by the product data management system.

3. Simulation model manager

Given a request to determine one or more performance parameters to a given level of accuracy, the simulation model manager must determine if the state of the design has evolved to the point that those evaluations can be done, and, if it has, to provide the simulation procedures with the design definition information needed to perform the simulations.

To support engineering analysis, that are applied on specific combinations of components of a design using various levels of model idealization, a high-level functional view of the design is needed. This abstract model includes the functional components of the design, decomposed to the level of detail needed for the operations to be executed, and the relationships to other functional components. The individual components need to be attributed with the additional information to support the operations to be executed during the required simulations. Information important to simulation-based design includes geometry definition, material properties, loadings, and boundary conditions associated with the component and its interfaces to other components. An examination of commercial CAD systems would imply the needed functionalities and structures are available since these systems typically provide:

- An assembly graph of components used and an assembly model defining the instances of the individual components or sub-assemblies that define the current design [1].
- A boundary representation [2] for the current design consistent with the assembly graph.
- A structure to support the association of general attribute information to the entities in the assembly graph.

Although it is possible, with appropriate care, to obtain the information needed to support SBE processes from the information provided by the CAD system, the direct use of these structures does not effectively support the execution of adaptive simulation processes. In addition, there is limited uniformity between the CAD systems with respect to how one interacts with the structures. On the other hand, it is possible to add the structure needed for the simulation model manager to effectively integrate with the CAD system tools at the assembly tree level. In addition, the most technically complex aspect of interacting with the CAD model, the interrogation and modification of the detailed geometric model, is effectively supported by the CAD systems in a fairly uniform manner.

To support the high-level definition of the design for simulation processes, a component model structure has been defined. From the simplest point of view, the component model consists of a set of components and interfaces between the components where the interfaces are functionally meaningful. To meet the needs of SBD, each component can be defined in terms of sub-components providing a hierarchal description of the model. Components can have a rich set of attributes associated with them

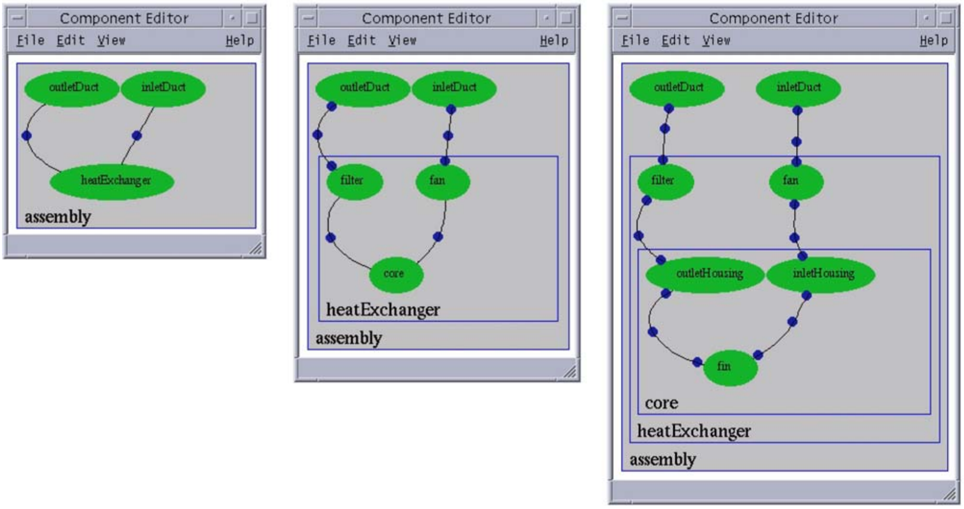

Fig. 2. Heat exchanger component model.

and the propagation of the attributes associated with components within the hierarchy is governed by a set of rules. In addition, it must be possible to have a complete geometric representation of the component model generated using the functionality of the CAD tools.

As a simple example of how the component model is defined, consider a heat exchanger design. The component model for this is shown in Fig. 2. The figure on the left shows the coarsest component representation where heat exchanger unit connects to the inlet and outlet duct. In the middle picture, it can be seen that the heat exchanger unit is comprised of a core, fan and filter. Note that the relation between the heat exchanger and the inlet is actually between the fan and the inlet duct, the ability to have these hierarchical relationships is part of the power of the component-model representation. The picture on the right shows that the core has a further decomposition.

From the conceptual level, the instance of the component model will be constructed from the master model version of the assembly (also called feature) model [3,4] taking into account the functional components to be included for the simulation process. Methods for extracting the component models accounting for multiple viewpoints are an active area of research [4–6] which has yet to be formalized to the point that the CAD systems fully support these concepts. However, experience indicates that with a bit of effort, the assembly models within the current CAD systems can provide the needed information. The example component models discussed in Section 7 have been successfully constructed using the assembly modeling capabilities of Pro/Engineer from PTC.

Within a SBD process, attributes are any needed information which is associated with a component that is not part of the geometric model definition. The types of attributes important to an SBD process are:

- Physical attributes including loads, material properties and boundary conditions.
- Modeling attributes that indicate the appropriate mathematical models and specific model idealization including geometric simplifications.

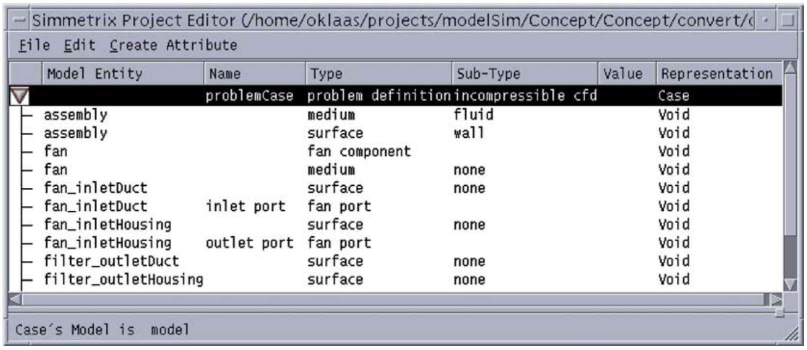

Fig. 3. Attributes for CFD analysis on component model of heat exchanger.

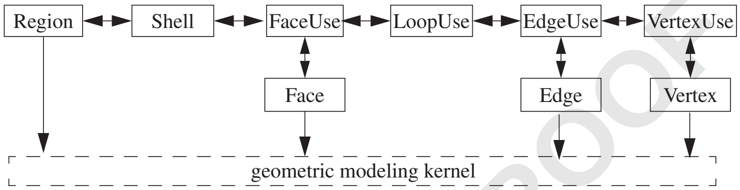

Fig. 4. Model topological adjacency information and relation to model geometry.

- Numerical analysis attributes used by the procedures defining the input needed to the CAE analysis procedures.
- Engineering performance parameters, the evaluation of which is the goal of the simulation processes.

Although the component model does deal with the association of attributes with the model components and interfaces, a richer capability is needed to define the physical attributes which are tensors that have general dependencies and variations. A specific set of structures and capabilities are included in the simulation model manager to support the definition, manipulation, grouping and evaluation of the physical attribute information [7] Fig. 3 shows a set of attributes appropriate for a CFD analysis of the heat exchanger component model.

The last key structure within the model manager is the topology associated with the geometric model. Since the geometric models needed for simulation processes are often general combinations of solids, surfaces, curves and points, a complete non-manifold representation in the form of the radial-edge data structure [2] (Fig. 4) is used. The primary reason the simulation model manager needs to maintain the topology of the geometric model is that CAD systems are not uniform in the way they treat topology, including some not automatically determining and representing specific non-manifold situations. However, since the geometric modelers, when properly used, can evaluate the non-manifold situations and support the geometric interrogations needed in the simulation processes, the geometric modeling functionality required can be supported by the combination of a non-manifold topology with the CAD system solid modeler. Within the simulation model manager, all interrogations of the geometric domain are made with operators that are driven by entities in the non-manifold topology. These operators are constructed for the various CAD system geometric modelers using their operator libraries [8–10].

For many classes of simulation, the analysis model is constructed from a spatial discretization of the geometric model. In these cases the relationship of the component model entities to the geometric model entities and the process of transforming attributes defined on the component model to the geometric models entities are critical. The CAD system assembly modelers maintain the relationship between the entities in the assembly model and the entities in the boundary representation of the geometric model it maintains. Therefore, these relationships can be constructed from viewpoint specific component models from which the correct non-manifold topological model is created by the simulation model manager. Proper care must be taken to construct the correct representations of any physical interfaces that may be defined by the interaction of component model entities.

The proper transformation of the attribute information from the component model entities to the non-manifold topological model entities must be governed by a set of rules. The effective application of the simulation model manager requires as many of these rules as possible to be generic with as few as possible being application domain specific. Maximizing the number of generic rules has been greatly facilitated by the inclusion of a component class abstraction so that each instance of a component can inherit the attributes and the rules associated with them. This functionality is particularly important for having the interfaces between components inherit the correct attributes.

Fig. 5 gives a simple example of the power of being able to control SBD processes through a component model that links to the detailed geometry needed for a simulation process. The three components of the heat exchange system are the same in both cases. The only difference is in the exchange unit sub-components in the two cases. For the case on the right there are three instances of

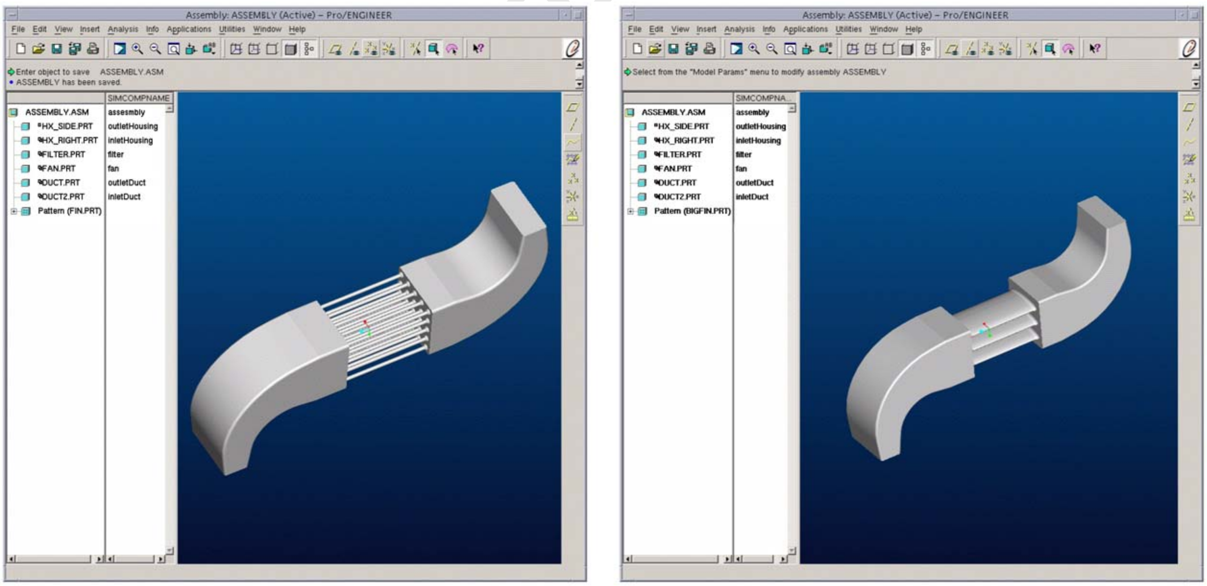

Fig. 5. Two heat exchanger designs.

a sub-component with an elongated cross section correctly positioned while in the figure on the left there are a number of instances of a circular tube cross section correctly positioned. The attributes as needed for a flow simulation are the same for both cases and the combination of the simulation model manager and CAD system geometric modeler can provide the appropriate information to the automate mesh generators and CAE flow solver even though the actual geometry is quite different in the two cases.

4. Simulation data manager

To effectively utilize simulation information within the design processes requires the ability to automatically evaluate the requested performance parameters and present that information to the user (be it a designer or optimization program) in a manner that can be utilized for decision making or further processing. These needs are not met by the output for the CAE analysis tools that consists of a set of discrete values associated with entities used by the numerical analysis procedures. However, an examination of the information used by the CAE analysis process indicates that in fact the information needed to evaluate the requested performance parameters is available, it is just not properly maintained. The function of the simulation data manager is to maintain the needed information and to provide the functionality needed to perform the requested performance parameter evaluations.

The majority of the analysis procedures within a SBD system employ a two step discretization process (e.g., finite elements, finite volumes, finite difference, partition of unity). In the first step the space/time domain of the simulation is decomposed into a set of entities, a mesh, the union[1] of which properly covers the space/time domain of interest. In the second step the mathematical equations to be solved are discretized over each of the entities defined in the first step in such a manner that the discrete system for the entire problem can be assembled and solved. For purposes of this discussion, consideration will focus on terminology and specific examples appropriate for the finite element method. However, the overall structures and approach work in much the same manner for the other methods.

To gain an appreciation for the information and operations the simulation data manager is responsible for, consider the simple performance parameter request of determining the downward force generated by the rear spoiler on a sports car. The evaluation of this performance parameter requires the execution of an external fluid flow over the entire vehicle under the appropriate drive conditions. Knowledge of the fluid flow field will allow the evaluation of the pressure over the car's exterior surfaces. The downward force on the spoiler is then obtained by the integration of the pressure over the surfaces associated with the spoiler component. Assuming that the analysis was performed using a finite element method that produces nodal pressures, the information that must be tracked to evaluate the downward force performance parameter include:

- Determining the geometric model faces defining the exterior surface of the spoiler component.
- Determining the finite elements that have faces that lie on those geometric model faces.
- Determining the nodes for each of the finite element faces and the discrete values of the pressures at those nodes.

[1] The operational definition of the union operation is a function of the equation discretization method used.

- Determining the interpolant associated with the pressure variation over each of those finite element faces so that, in combination with the nodal values, the pressure field can be integrated over that finite element face and summed with the integrals over the other finite element faces.

The structures used by the simulation data manager to support these types of operations are:

- a mesh topology,
- the association of the mesh entities with the geometric model entities, and the association of the geometric model entities with the component model entities,
- field structures that maintain the discrete values and interpolants defining the distribution of the analysis solution fields over the mesh.

The mesh input structures, the result of the first step in the discretization process, used by the analysis codes are quite varied as are the operations that need to be supported to operate on them. Therefore, the mesh structure used in SEED is a mesh topology which provides an effective means for meeting these needs [11,12]. Under the assumption that each topological mesh entity of dimension d, M_i^d, is bounded by a set of topological mesh entities of dimension $d-1$, $\{M_i^d\{M^{d-1}\}\}$, the full set of mesh topological entities are:

$$T_M = \{\{M\{M^0\}\},\ \{M\{M^1\}\},\ \{M\{M^2\}\},\ \{M\{M^3\}\}\}, \tag{1}$$

where $\{M\{M^d\}\}$, $d = 0, 1, 2, 3$ are, respectively, the set of vertices, edges, faces and regions which define the primary topological elements of the mesh domain. A key component of supporting mesh-based procedures in SBD is the association of the mesh with respect to the geometric model [9–12]. This association is referred to as classification in which the mesh topological entities are classified with respect to the geometric model topological entities upon which they lie.

Definition (Classification). The unique association of mesh topological entities of dimension d_i, $M_i^{d_i}$ to the topological entity of the geometric model of dimension d_j, $G_j^{d_j}$ where $d_i \leqslant d_j$, on which it lies is termed classification and is denoted $M_i^{d_i} \sqsubset G_j^{d_j}$ where the classification symbol, $\sqsubset$, indicates that the left hand entity, or set, is classified on the right hand entity.

Definition (Reverse classification). For each model entity, G_j^d, the set of equal order mesh entities classified on that model entity define the reverse classification information for that model entity. Reverse classification is denoted as

$$FC(G_j^d) = \{M_i^d | M_i^d \sqsubset G_j^d\}. \tag{2}$$

Functions to provide the reverse classification, that is the equal order mesh entities classified on a model entity, are useful in various steps of simulation processes such as constructing system contributors.

Mesh shape information can be effectively associated with the topological entities defining the mesh. In many cases this is limited to the coordinates of the mesh vertices and, if they exist, higher-order nodes associated with mesh edges, faces or regions. In addition, it is possible to associate other forms of geometric information with the mesh entities. For example, the association of Bezier curves and surface definitions with mesh edges and faces for use in p-version finite elements [13]. The mesh classification can be used to obtain other needed geometric information such as the coordinates of a new mesh vertex caused by splitting a mesh edge classified on a model face or to support the calculation of the geometric Jacobian information when doing an element stiffness integration.

In the second step of the discretization process the dependent variables of the mathematical equations being solved are approximated in terms of a finite number of basis functions and degrees of freedom (dof) over each element, e, where elements will be associated with appropriate mesh entities. Symbolically this process can be written as

$$\underset{\sim}{u}^{e} \sum_{i=1}^{ndof^{e}} \underset{\sim}{d}_{i} N_{i}, \tag{3}$$

where $\underset{\sim}{u}^{e}$ is the typical vector dependent variable written over the element e. N_i the ith $\underset{\sim}{u}$ basis functions on contributor e. $\underset{\sim}{d}_i$ the vector of dof associated with N_i in the construction of $\underset{\sim}{u}^{e}$.

It is common to think of the dof as multipliers and the basis functions as the distribution of the dependent variables. The basis functions and dof are defined such that an appropriate set of operations can be performed so that the dof of neighboring elements are properly coupled such that the resulting assembled system is a proper approximation of the mathematical equations to be solved.

The combination of the mesh structure (with classification), basis functions and dof can be used to define the dependent variables over the domain of the problem. This combination is referred to in SEED as a field. Fig. 6 graphically depicts a C^0 continuous field over a patch of entities. The fields associated with the dependent variables of the mathematical model are referred to as the primary fields. In many cases it is desirable to construct addition fields which are typically referred to as secondary fields. Often specific projection operations and constraints are associated with the construction of secondary fields. An example of a secondary field is a C^0 stress field constructed from a C^0 displacement-based finite element analysis. In this cases the stresses are related to appropriate combinations of derivatives of the displacements times appropriate material parameters, which are C^{-1} between elements when directly evaluated. In this case the C^0 stress field is constructed using a projection of the C^{-1} field using a set of C^0 stress field basis functions and dof.

Since the definition of fields is in terms of coordinate systems, the structures must understand the coordinate system the field is defined in, and support the transformation of the field into other coordinate systems. In addition, additional structures may be introduced to support various operations on them. For example, if one wants to support the general interrogation of a field for its value

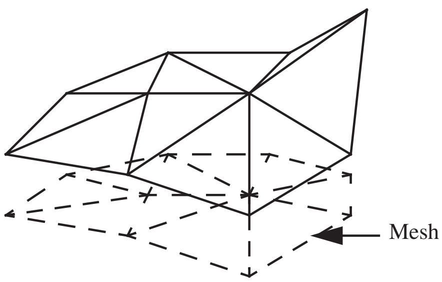

Fig. 6. Representation of a field defined over a mesh.

anywhere over the domain given just the coordinates of the point, search structures can be introduced to efficiently identify a small list of candidate mesh entities to check for the containment of the point of interest. The SEED simulation model manager supports these needs using both adjacency-based and spatially based search structures.

The linkage from the mesh entities to the geometric model entities, and from the geometric model to the component model supports the ability to perform field queries with respect to mesh, geometric and component models. Applications using SEED do make use of queries based on all three models. As already indicated, the queries that are performed on fields include simple point evaluations, differentiations, integrations, projections and various combinations. In some cases, these operations are supported by low-level interfaces. However, to support the full set of needs for applications interacting with simulation data manager, a high-level means of interacting with the field structures is needed.

The simulation data manager uses the analogy to the structure and queries used in relational databases to address the access to field information. In a standard relational database, data is organized into tables. Each column of a table holds a certain type of data (e.g. integers, real numbers, strings) and has a name that identifies it. Each row is one instance of the dataset that the table contains. A database is comprised of multiple tables. The power of relational databases comes from the fact that the data in each table can be related and complex queries built through those relations. The typical way that data in tables is related is through certain columns that represent the same data in each table.

The simulation data manager has an analogue to a table (called a result set), where each column represents a different field. The analogue to a row in a table is the variation of the field over a given mesh entity. Thus all of the fields in a result set (table) must be on the same mesh. Note that the data is not actually stored in this manner, it is only a convenient way to describe how this relates to a standard relational database.

Since different result sets may be on different meshes, relationships between results sets are indirect and controlled through either the geometric or component model that the two meshes are from, or simply a spatial relation. These relationships are a much more functional relation than typically seen in database management but are absolutely key to the effective operation of the simulation data manager.

Even given that substantial difference, the analogy to a relational database is a good way to think about the data, and the simulation model manager employs a query language similar to SQL to query the data. The "workhorse" command in SQL is the SELECT command. This is how virtually all information is extracted from a database. The simulation model manager extends the basic SELECT command to work with simulation fields. An example of a simple query would be (query language keywords are in all caps):

```
SELECT temperature FROM result1 WHERE pressure > 10
```

This would return a part of the field named "temperature" from the result set (table) named "result1" on the portion of the domain where the field "pressure" has a value greater than 10.

Since the field data is related to the mesh, geometric and component models, queries employing those relations are needed. The ON keyword has been introduced for this purpose. In addition, with the "secondary field" concept discussed above, a more complex query with the same data could be written as:

SELECT heat flux FROM result1 WHERE pressure > 10 ON outlet

In this case the query is only being done over the portion of the component model named "outlet" and the heat flux will be automatically computed from the temperature field. In general most of the arguments to the keywords in the SELECT statement can be arbitrary calculations which results in a system that can easily specify very complex queries.

5. Simulation model generators

Current CAD/CAE integration tools fall short of meeting the needs of simulation-based design in terms of the ability to automatically generate the adaptively controlled analysis models. They force users to work with low-level information specific to the CAE tools used. As designers are not CAE experts it is clear that they cannot be expected to use these tools effectively. In addition the CAD/CAE integration available today is typically limited to a one way transfer from CAD to CAE in a manner which is often error prone [8]. There is also a lack of communication of the CAE results back to the design as needed to support the coupling of the analysis procedures within engineering design processes.

The function of the simulation model generators is to automatically construct the input to the analysis programs in the CAE tools given the current state of the model definition and information provided by the adaptive control tools. Within SEED, the current state of the model definition is contained in the component model, the geometric model and the physical attributes. The adaptive control tool information is conveyed as model selection, model idealization and discretization control information. Since the adaptive control procedures are defining these attributes as part of an iterative solution process, these attributes can be associated with entities in the component model, geometric model and mesh models.

Some examples of adaptive control attributes that may be applied to entities in the various SEED models include:

- Indication that one set of components will be represented by 1-D engineering models for fluid flow that will be interfaced, with the correct interface boundary conditions, to full 3-D adaptive finite element discretizations of other components of the domain.
- Indicating the mesh size fields for an initial mesh over entities in the component or geometric model based on a priori expert rules.
- The indication the specific component or portion of a geometric model which are thin in one direction that can be replaced by a dimensionally reduced idealization in terms of a shell where the small dimension is replaced by a thickness parameter.
- Discretization error control information that indicates a new mesh size field in terms of parameters defined over an existing mesh.
- Multiscale models that require the application of atomic level model representation to determination the evolution of defects over specific elements of a finite element mesh [14].

Current emphasis in the development of SEED is to provide effective mechanisms to support specific commonly applied adaptive control attributes. These include:

- Support for applying different levels of models to components in the component model and controlling the interactions between components based on how each component is idealized.
- Support of generalized mesh size fields to control automatic generation [9] or modification [15] of meshes.
- Support for specific classes of geometric idealizations as specified at the component or geometric model level.

The component model represents the ideal level to specify the mathematical modeling and analysis idealization since it is easy to alter the level of geometric and physical representation for the components and their interfaces to other components allowing the direct construction of the geometric model and associated analysis attributes. This also allows the application of simplified analysis models even early in the design process before geometric detail of the entities and their interfaces are defined, or to support mixed levels of idealization for different components.

To briefly indicate how the component model supports such operations, some examples associated with the climate control SBD system presented in Section 7 are outlined. Some major components in the climate control system are the radiator, duct work, heat exchanger, registers, cabin geometry, underhood geometry and exterior geometry. Early in the design process, there may be little known about the vehicle other that what components interface to which and what some of the expected thermal loads may be like. At this point, simple 1-D engineering analysis models may be appropriate for initial component sizing since the existence of the components and their interfaces are known. As design proceeds, there will be situations where it will be desirable to consider different model levels for different components. For example, to analyze the cabin temperature history for a drive cycle, it is necessary to have a fully 3-D model on the interior of the passenger cabin. On the other hand, the simulation of the refrigerant systems that determines the heat extraction and generation of the climatized air can be determined by 1-D models coupled to 3-D idealized models for the evaporator and heater core components. In this case the interface between the duct connecting to the register needs to convert the 1-D thermal information into appropriate thermal boundary condition. As a final example, consider the radiator where even in the case of a 3-D flow analysis a simplified model of the core as a box with porous medium properties that adds heat to the fluid flowing through the box is the only viable model, since fully modeling the core would require modeling many very small complex fins which would force the overall problem size to increase by several orders of magnitude. The procedures used to determine the porosity parameters for the heat exchanger could come from experimental measurements or the application of homogenization on a unit cell that does include the full geometric detail.

SEED includes component-based automatic mesh generation tools [16] designed to be easily controlled from a high-level problem definition. The procedures have been specifically designed to generate valid meshes for general non-manifold geometric models through a direct interface with the solid modeling system [9,10] using the abstraction of topological entities and their adjacencies to control the mesh generation and other processes [9–11], and the solid modeling system's libraries [17–19] to construct the geometric interrogations [10] needed by the automatic mesh generator.

Given the topological structures of the geometry model and mesh, and the ability to perform the needed geometric interrogations, the mesh generation and modification procedures are constructed in terms of operator sets that:

- Determine topological adjacencies and mesh classification
- Interrogate the geometric model for shape parameters
- Create/delete/modify mesh entities and assign/modify shape information
- Evaluate/assign mesh entity attributes

The geometric model and mesh data structures used can provide any required adjacency directly or through traversing the stored adjacencies and local sorts [11]. These structures can be used to effectively support the specification of variable p-version meshes [20].

The automatic mesh generation procedures have been integrated with the geometric modeling kernels of SDRC's I-DEAS, PTC's Pro/Engineer and Dassault's CATIA CAD systems, as well as Spatial's ACIS, PTC's Granite and Unigraphics' Parasolid modeling kernels which are the basis for the majority of CAD systems on the market.

The process of supporting the geometric idealizations requires the coordination of the component model, the attribute information, the non-manifold topology controlled by the simulation model manager and the CAD tools. Although the specific of the operations required to a specific geometric idealization vary, the basic requirements of the process are the same and include:

- The ability to take the idealization information associated with the component and have the appropriate topological entities to represent the geometry of that component in the non-manifold topology created.
- The ability of the attributes associated with the component to be transferred to attributes associated with the appropriate non-manifold model entities.
- The ability of the CAD geometric modeler to perform any required geometric model operations.
- The ability of the CAD geometric modeler to support the geometric interrogations operation used by the simulation model generators.

A key point to be made here is that since the simulation model manager controls the non-manifold topology accessed by the simulation model generators, there are a number of geometric idealizations that can be performed without the need to force the CAD modeler to modify the existing geometric model. One example of this type is ignoring "geometric features". In many classes of analysis some of the small geometric features do not need to have all the topological entities defining them explicitly represented in the mesh. In many such cases the faces in the CAD model can be conceptually concatenated in the non-manifold model by have a single face that simply maintains a link to all the entities in the CAD model associated with it so that the needed geometric interrogations are supported.

A second example that will require requesting a limited number of specific geometric operations without the need to alter the detailed model in the CAD system are dimensional reductions. One such dimensional reduction is associated with models that have components that are thin in one dimension compared to the other two. In these cases, the volume associated with that thin portion of the model is to be replaced in the non-manifold model by a face in the analysis model. The "loops" at the "boundary" of the thin section are replaced by the edges. In addition, an attribute of "thickness" must be defined, "evaluated" and associated with the new face. To make this example more concrete, consider the heat exchanger shown in the left image of Fig. 7 that consists of two ducts and three

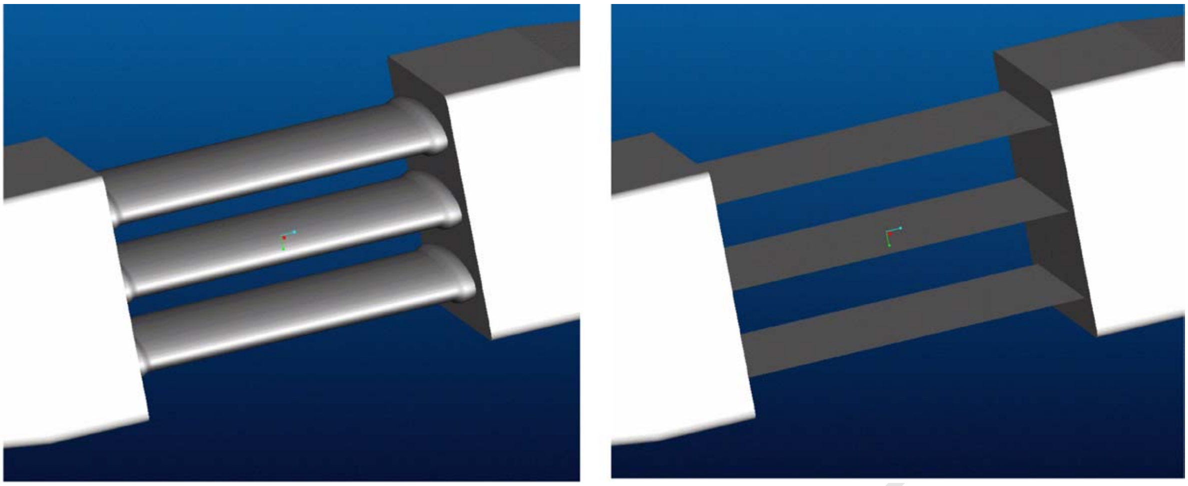

Fig. 7. Dimensional reduction for heat exchanger.

tube heat exchange sections. Assume for the purposes of an overall structural analysis it is acceptable to replace the tube sections with simple flat plates as indicated in the image on the right. This can be accomplished without specifically altering the original CAD solid model in the following way.

- The CAD system model is asked to:
  - Determine the mid-plane associated with each of the three tubes. In this case each of the mid-planes are rectangular faces. The rectangle defines a face bounded by four edges and four vertices.
  - Determine the faces associated with the loop defined at the ends of each tube.

  The CAD modeler can maintain the closure of each of the faces as separate entities with the original model remaining intact.
- The non-manifold model to support the structural analysis in the simulation model manager will:
  - Replace the face or faces defining each tube section with the planer face which maintains knowledge of the original model entities, as well as the independent face, defined in the CAD geometric modeler.
  - Replace the loops at the end of the tube section with an edge that corresponds to the correct edge of the independent rectangular face.
  - Modify the duct faces at the ends of the two ducts to each be a rectangular face with three edges that replace the three loops that used to represent the end loops of the airfoils.
- The non-manifold model will also set the link to the CAD model entities such that the geometric interrogations associated with the tube are directed to the correct planar faces and those associated with the duct end faces considering the original face geometry, the faces associated with the tube end loops and the edges of the tubes faces.
- A thickness attribute is added to each tube face in the geometry and the correct links and operations are made to the original geometric model to determine the sum of the perpendicular distance from the single tube face in the non-manifold model to the faces defining that tube in the unaltered original CAD model.

6. Adaptive control tools

These tools are responsible for determining the appropriate mathematical models, selecting discretization technologies, evaluating the accuracy of the predictions obtained, and determining the improvements of the models and discretizations needed to obtain the desired accuracy.

The first step is to determine which techniques can be used to perform the simulations needed. At a minimum, this includes ascertaining which mathematical models and discretization techniques can be applied to calculate the requested parameters. The product SBD definition must then be examined to determine if it is complete enough to apply one or more of the possible sets of mathematical modeling and discretization methods.

In most simulations there is no a priori means to construct the initial model to ensure the desired solution accuracy. Adaptive feedback loops [21,22], using the results of a given simulation to estimate the errors and determine the best means of improving the approximations, are the most robust approach to achieve the desired accuracy. Since mathematically based error estimation exists for only a limited number of simulation error contributions, it is necessary to support all means that are, and may become, available for controlling simulation accuracy. These range from rules [23], to experimental validation, to new mathematical modeling techniques.

Initial model selection is based on a priori knowledge. When model assumptions are related to the range of solution parameters expected, the results can be checked against the assumptions. In many other cases there are no robust procedures for deciding if the model is adequate. However, there is the ability to encode expert knowledge to aid in the selection and evaluation of models. Experimental validation must be considered part of this process [24]. When multiple interacting models are employed, these interactions must be properly controlled to ensure the accuracy of the overall simulation.

Early in the design, the detailed design geometry may not be complete, or at later points some geometric details may be ignored to reduce simulation costs. Thus, procedures to deal with idealized geometries are an important part of adaptive simulation control. When the influence of a feature is local there are reasonable methods to obtain a conservative estimate of its influence on the solution field [22]. Also, although physical objects are three-dimensional, there are many cases where the dominant behavior can be captured by a two- or one-dimensional simplification of the model. Procedures are under development to estimate the influence of reduced dimension representations and adaptively control their selection [25].

Infinite-dimensional mathematical models are discretized into finite dimensional problems for numerical solution. In many cases, robust methods to estimate this discretization error [21] and adapt the discretization are known.

7. Application of SEED to the design of automotive climate control systems

As SEED is being developed there are efforts underway to develop specific SBD systems for applications in the automotive, heavy equipment, consumer product and defense industries. The example outlined in this section is the most mature SBD system which has been developed as a collaborative effort with a tier one automotive supplier. The specific system is focused on the design of automotive climate control systems and takes advantage of previous efforts to define a parametric

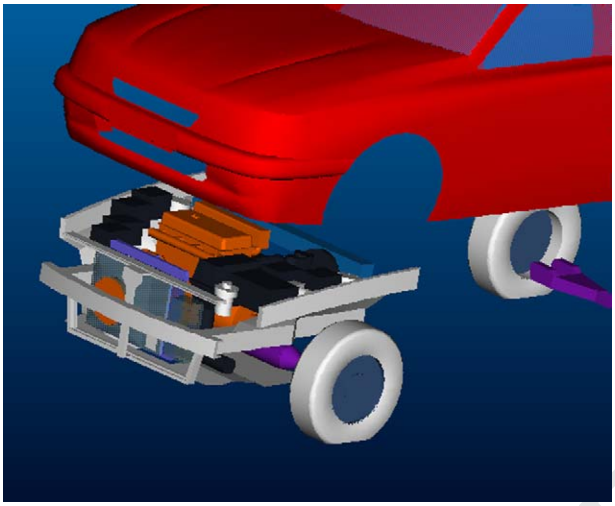

Fig. 8. Exploded view of major functional components.

vehicle model in the Pro/E CAD system. This parametric model includes a thermal/fluids engineering viewpoint structured to support accurate and validated simulation processes as needed to support the design of automotive climate control systems.

The users of the SBD system are design engineers who are charged with the selection and sizing of the climate control components associated with a vehicle platform to most effectively satisfy a set of design goals and constraints. The design goals and constraints range from the geometric parameters of climate control, to overall passenger comfort under a variety of driving conditions.

The parametric model has been decomposed into a set of high-level functional components such as overall exterior type, engine, suspension assembly, cabin interior assembly, etc. so that it can be synchronized with the component model in the simulation model manager. Fig. 8 shows a generic view of some of the major functional components for a particular vehicle platform. The structure of these components have been defined as appropriate for various vehicle platforms of interest. As part of the process the designer is able to select from and define the configurations of sub-components defining the major components. Fig. 9 shows an example of this for the underhood cooling system component where sub-components include fans, radiator cores, overflow tanks, etc.

As the design process continues the designer can select from discrete sub-component components for items as fan blades, pumps, etc. In addition, many of the components have geometric parameters that can be varied to satisfy various constraints and/or improve performance.

The design engineers must work in an environment structured to support the operations they will perform. From the perspective of those operations involving direct interaction with the design components and associated parameters, the CAD system user interface is the best suited of the available tools. In the current implementation the assembly modeling tools of the Pro/Engineer (Pro/E) CAD system are used. Pro/E provides the designer a good environment to load in basic predefined assembly tree templates and to work with them selecting and configuring the sub-components to be used and setting the appropriate design parameters. The process of making design decisions requires

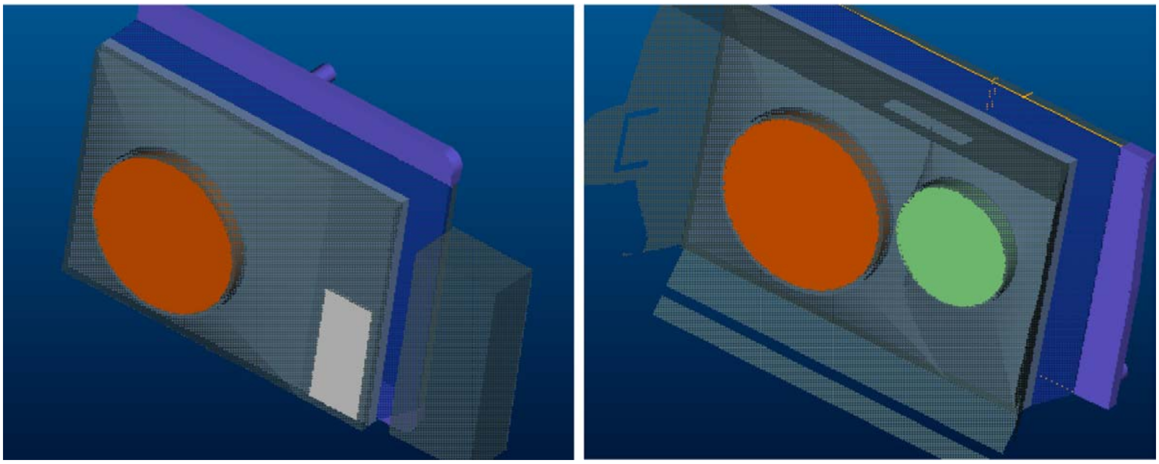

Fig. 9. Different sub-component selection for underhood cooling system component.

that ability to obtain useful estimates of a number of performance parameters. The evaluation of these parameters requires the execution of various engineering analyses which must be executed automatically. These processes are supported by SEED which begins with the current state of the design loaded into its component model and physical attribute structures. In addition, SEED will contain the information necessary to perform the required performance evaluations.

The designer works with high-level performance parameter specifications which in the case of interior comfort control design are carefully qualified by SBD specialists with respect to the comfort level response of occupants placed in the vehicle. Other performance parameters are related to a simple physical parameter, but often require the execution of extensive coupled simulations. An example of this type is radiator top water temperature that must be predicted to within a few degrees [centigrade]. The prediction of this engineering parameter requires full exterior/underhood flow analysis, coupled by calibrated engineering models for the heating/cooling system and solar radiation. This analysis needs to be performed for complete transient drive cycles.

The determination of specified performance parameters begins from a simulation template that is constructed from a priori knowledge codified into the analysis control attributes about what methods are appropriate at this point in the design process. The modeling and numerical analysis attributes associated with the components are used to provide the needed information which must include the levels of model idealization possible for each component as well as how various components can interact. A simple example of this is depicted in Fig. 10 for the heating core and connected duct in a heat exchanger where three different levels of idealization of the heating core are associated with two different levels of geometric representation of the duct.

In those cases where generalized numerical analysis procedures are used, a spatial discretization of the domain in terms of a mesh is needed. To support the mesh generation process the non-manifold topology and appropriate links to the geometric information in the CAD system needs to be constructed. This process requires specific interactions between SEED component, attribute and non-manifold model topology and the Pro/E assembly tree and geometric modeling engine. In cases where analysis idealizations reduce the dimensionality or simplify portions of the geometric

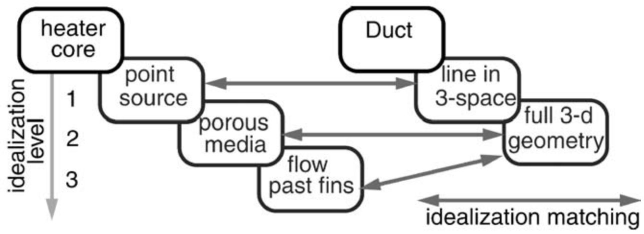

Fig. 10. Model idealizations associated with the two heat exchanger components.

domain, the system contains sufficient information to execute those processes. The initial numerical analysis attributes contain information that indicates the initial mesh size field over the domain and parameters to appropriately control the execution of the CAE program.

Although the application of adaptive model and discretization selection is the most appropriate means to control the accuracy of the solution process, reliable means for the a posteriori prediction of the solution error and adaptive control methods for all error contributions are not available. In addition, in many design situations, the similarity of the designs and analyses performed on them is such that capturing validated expert analysis models and procedures is a highly effective means of ensuring simulation reliability. The SBD application does contain adaptive methods for controlling mesh discretization for 3-D fluid flow analyses over complex geometries. For other analyses, and for the creation of effective initial meshes, a substantial effort was expended on capturing the knowledge CAE experts gained through the execution of many analyses over a range of design configurations for which validation was performed.

The SBD application has codified the analyses processes for a variety virtual tests that are performed using well controlled thermal/flow simulations. A subset of these are: cool-down and warm-up performance, defrost performance, hot climate soak, and city drive cycle. Many of these simulations require the transient solution of the Navier Strokes equations over general domains where thousands of time steps must be solved over meshes with million's of mesh points. These simulations are carried out using a stabilized finite element solver (ACUSIM [26]) that includes the required thermal fluid modeling components.

Fig. 11 shows the overall results of a design iteration for an exterior flow in the wind tunnel. For engine cooling, the important engineering performance information is the radiator top tank coolant temperature, the flow distribution through the heat exchangers, fan performance, and engine compartment air flow and temperature. The flow results in the underhood region critical to the determination of this information is shown in Fig. 12. The results obtained using these procedures have been compared to wind tunnel experimental tests for integrated air flow prediction and are found to be within the levels of accuracy needed in the design process.

Fig. 13 demonstrates the role of the adaptive mesh discretization error control. The top image shows the initial mesh generated based on the geometry-based mesh size specification information. Although these a priori specification procedures do a reasonable job of providing a useful initial mesh, they do not account for the detailed flow field interactions dictated by the geometric details that are varied from model to model. The deficiencies in the initial mesh are determined by the error estimation procedures and the mesh is automatically refined to yield the final mesh shown

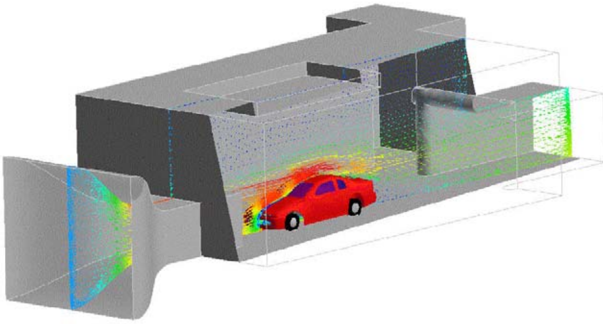

Fig. 11. Overall wind tunnel flow results.

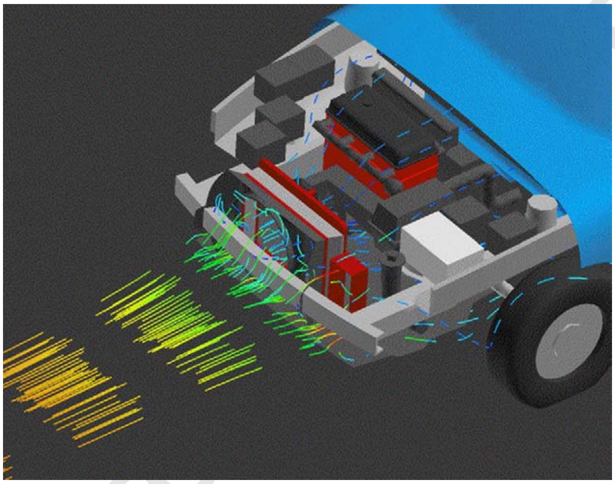

Fig. 12. Flow results in the critical under the hood region.

in the lower portion of Fig. 13. The application of adaptive mesh control, which will capture the interactions caused by interacting geometric details cannot be predicted in advance, in combination with encoded expert knowledge of the appropriate equations to be solved and mesh control to ensure that there is sufficient resolution for the adaptive procedures to effectively determine any missed details, provides designers with a robust ability to evaluate the performance parameters of interest to them.

The complex transient thermal/flow simulations required to evaluate the performance of a design can be run overnight on a cost effective parallel computer cluster. Although, the designer would like much faster turn around, a one-day cycle is found acceptable, especially considering that just a few years ago one of these simulations would have been considered a “national grand-challenge simulation” that would take days to run on the then worlds biggest parallel computer.

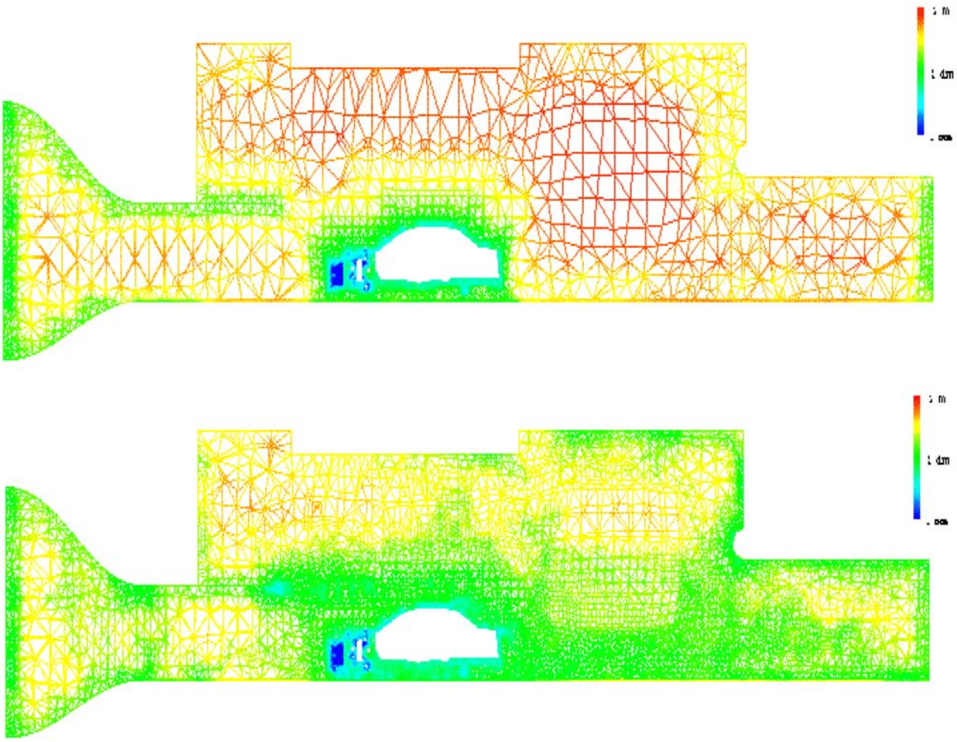

Fig. 13. Initial and adaptively refined mesh.

8. Closing remarks

Consideration of the capabilities needed to support simulation as a truly central component of engineering design indicates that although many components are already available, there are key missing components. These components, described here as the simulation model manager, simulation data manager, simulation model generators and adaptive control tools, play the key role in enabling simulation-based design. Although they can be viewed as the middleware that supports the proper integration of CAD and CAE, the historic mistake made by both CAD and CAE developers has been to take the simplest view of this integration as a direct data transfer process. Each of the SEED components contain substantial functionality needed to carry out specific steps in the processes of executing the analyses needed to properly evaluate design performance parameters.

In addition to the definition of the SEED functional components, this paper has provided a brief overview of the application of these technologies in the automotive industry. The climate control SBD application outlined is in production use today and has provided a clear demonstration that SBD technologies provide substantial cost and time savings. Of course the introduction of these technologies does have a substantial up-front cost. The design models have to be defined and the expert and adaptive control techniques must be encoded. The development of the SEED components will make it possible to carry out future SBD system developments in a cost effective manner.

The final ingredient to the introduction of SBD technologies into industry is the change in organizational practices and structures that are required for its effective use. Although beyond the scope of a paper, this is a critical area that companies must address as they move forward.

Acknowledgements

The authors gratefully acknowledge the NIST ATP program, award number 70NANB1H3035, for supporting the development of SEED.

References

[1] A. Noort, G.F.M. Hoek, W.F. Bronsvoort, Integrated part and assembly modeling, Comput. Aided Des. 34 (2002) 899–912.
[2] K. Weiler, The radial edge structure: a topological representation for non-manifold geometric modeling, in: M.J. Wozny, H. McLaughlin, J. Encarnacao (Eds.), Geometric Modeling for CAD Applications, IFIP WG5.2 Working Conference Rensselaerville, N.Y. (May 12–14, 1986), 1988, pp. 3–36.
[3] R. Bidarra, W.F. Bronsvoort, Semantic feature modeling, Comput. Aided Des. 32 (2000) 201–225.
[4] C.M. Hoffmann, R. Joan-Arinyo, CAD and the product master model, Comput. Aided Des. 30 (11) (1998) 905–918.
[5] W.F. Bronsvoort, F.W. Jansen, Feature modeling and conversion—key concepts to concurrent engineering, Comput. Ind. 21 (1) (1993) 61–86.
[6] C.M. Hoffmann, R. Joan-Arinyo, Distributed maintenance of multiple project views, Comput. Aided Des. 32 (2000) 421–431.
[7] R.M. O'Bara, M.W. Beall, M.S. Shephard, Attribute management system for engineering analysis, Eng. Comput. 18 (4) (2002) 339–351.
[8] M.W. Beall, J. Walsh, M.S. Shephard, Accessing CAD geometry for mesh generation, 12th International Meshing Roundtable, Sandia National Laboratories, 2003.
[9] M.S. Shephard, Meshing environment for geometry-based analysis, Int. J. Numer. Methods Eng. 47 (1–3) (2000).
[10] M.S. Shephard, M.K. Georges, Reliability of automatic 3-D mesh generation, Comput. Meth. Appl. Mech. Eng. 101 (1992) 443–462.
[11] M.W. Beall, M.S. Shephard, A general topology-based mesh data structure, Int. J. Numer. Meth. Eng. 40 (1997) 1573–1596.
[12] J.-F. Remacle, M.S. Shephard, An algorithm oriented mesh database, Int. J. Numer. Methods Eng. 58 (2003) 349–374.
[13] S. Dey, R.M. O'Bara, M.S. Shephard, Curvilinear mesh generation in 3D, Comput. Aided Des. 33 (2001) 199–209.
[14] D. Datta, R.C. Picu, M.S. Shephard, Multigrid Atomistic Continuum Method: An adaptive approach to bridge continuum with atomistic analysis, Scientific Computation Research Center, RPI, Troy, NY, 2003, submitted for publication.
[15] X. Li, M.S. Shephard, M.W. Beall, 3-D Anisotropic Mesh Adaptation by Mesh Modifications, Comput. Meth. Appl. Mech. Eng. (2003), submitted for publication.
[16] Simmetrix, http://www.simmetrix.com/, Simmetrix Inc., 10 Halfmoon Executive Park Drive, Clifton Park, NY 12065.
[17] EDS Corp., Parasolid V6 Functional Description, Electronic Data Systems Corporation, 13736 Riverport Drive, Maryland Heights, MO 63043, 1994.
[18] PTC, Pro/TOOLKIT Reference Manual, Parametric Technologies Corp., 128 Technology Drive, Waltham, MA, 02154, 1997.
[19] Spatial Technologies Inc., ACIS Geometric Modeler Interface Guide, Spatial Technology, Inc., 2425 55th Street, Building A, Boulder, CO, 80301-5704, 1990.
[20] M.S. Shephard, S. Dey, J.E. Flaherty, A straightforward structure to construct shape functions for variable p-order meshes, Comput. Methods Appl. Mech. Eng. 147 (1997) 209–233.

[21] M. Ainsworth, J.T. Oden, A Posteriori Error Estimation in Finite Element Analysis, Wiley-Interscience, Wiley, NY, 2000.
[22] M.S. Shephard, P.L. Baehmann, M.K. Georges, E.V. Korngold, Framework for the reliable generation and control of analysis idealizations, Comput. Meth. Appl. Mech. Eng. 82 (1990) 257–280.
[23] D. Holzhauer, I. Grosse, Finite element analysis using component decomposition and knowledge-based control, Eng. Comput. 15 (4) (1999) 315–325.
[24] D.P. Aeschkiman, W.L. Oberkampf, Experimental methodology for computational fluid dynamics code validation, AIAA J. 36 (5) (1998) 733–741.
[25] R.L. Actis, B.A. Szabo, C. Schwab, Hierarchic models for laminated plates and shells, Comput. Meth. Appl. Mech. Eng. 172 (1) (1999) 79–172.
[26] AcuSolve Command Reference Manual, Version 1.3, ACUSIM Software, 14395 Saratoga Ave., Suite 110, Saratoga, CA 95070, www.acusim.com, 2000.