Springer

Journal of
Mechanical
Science and
Technology

Journal of Mechanical Science and Technology 23 (2009) 3209~3220

www.springerlink.com/content/1738-494x
DOI 10.1007/s12206-009-1005-0

A Framework for automated finite element analysis with an ontology-based approach †

Wei Sun, Qinyi Ma* and Shuang Chen

School of Mechanical Engineering, Dalian University of Technology, No. 2, Ling Gong road, Gan Jing Zi District, Dalian City, Liao Ning Province, 116024, China

(Manuscript Received April 1, 2009; Revised August 20, 2009; Accepted September 11, 2009)

Abstract

To obtain accurate and reliable results of finite element analysis (FEA) requires a high level of expertise and the full-scale physical context information, which are bottlenecks restricting the application of achievements of FEA in industry. This paper proposes an ontology-based framework including a hierarchy transfer approach and a three-stage automated finite element analysis method to solve these problems. The hierarchy transfer approach is proposed to create different transfer formats according to the data, information, and knowledge, to carry out the integration at different levels. The knowledge found in design and FEA theories is presented by ontology in order to uniformly describe the physical phenomenon with the same semantic meanings. This involves the development of a shared design and FEA ontology, as well as, specific application ontologies in the Ontology Web Language (OWL). The three-stage automated finite element analysis method is applied to mark up artifact in problem definition, to reuse domain knowledge in problem formulation, and to enable the automation of the FEM analysis process in the solution routine with the application of AI techniques. The feasibility and effectiveness of the framework and concepts are empirically validated by a case study.

Keywords: FEA; Ontology; Automated analysis; Information modeling; CAD

1. Introduction

The finite element method (FEM) is the most successful numerical method to analyze stresses and deformations in physical structures [1]. Accuracy and reliability of the FEM analysis results depend heavily on the quality of the decisions made during the analysis process and the full-scale physical context information abstained during the decision making process.

However, to exploit contemporary potentialities of FEM to solve a complex engineering problem requires a high level of expertise [2]. For example, in FEM, structures have to be represented as finite element meshes. Most authors agree that the most time consuming part of undertaking a finite element (FE) analysis is the creation of the analysis model, which is still based mostly on the user's experience [3]. However, to many finite element users, generating a quality initial mesh for a given problem, which leads to a solution within an acceptable error, is still a challenging task since they lack proper expertise and adequate experience. Furthermore, current FE preprocessors operate at a relatively low level, primarily the numerical level-at which analysts must interact with finite-element programs to specify the elements of a model. They do not let analysts simply describe a physical structure with high-level analysis objectives and obtain corresponding finite-element models appropriate for these objectives [2].

A knowledge-based framework was proposed for assisting users in setting up, interpreting, and hierarchically refining finite-element models in a structural

† This paper was recommended for publication in revised form by Associate Editor Chang-Wan Kim
*Corresponding author. Tel.: +86 411 81890042, Fax.: +86 411 84708812
E-mail address: onlypony@163.com

engineering domain in [2]. The prototype of an intelligent rule-based consultative system was developed by Marina and Bojan [4] to provide such advice when considering a description of the design structure's critical area. This system was encoded in Prolog. However, these systems have the problem of domain-specific knowledge tightly coupled with procedures and rules, making the systems difficult to maintain and extend.

One of the most promising ways to separate and represent domain knowledge from operational or problem solving knowledge is through the use of ontologies. Ian illustrated how formal ontologies of engineering analysis modeling knowledge might facilitate knowledge exchange and improve reuse, adaptability, and interoperability of analysis models [5]. Masaharu et al. [6] reported a physical ontology-based support system for knowledge-intensive engineering called Knowledge Intensive Engineering Framework (KIEF) to integrate multiple engineering models and to allow more flexible use of them. The use of description logic (DL) concepts to describe archived models and build expandable classification hierarchies to facilitate retrieval was proposed and illustrated in [7]. Wriggers [8] analyzed the knowledge and reasoning involved in solving preprocessing tasks and showed that an automated system can be implemented using case-based reasoning (CBR) technology. Case retrieval and adaptation algorithms for this model were described. However, all of these researches have not taken the conceptual design into account. There is often a drop in the knowledge accessibility level, even the information level, when transferring from CAD to FEA. As a result, true AFEA cannot be achieved since human interaction is necessary to rebuild information / knowledge lost in the transfer. The requirement of data / information / knowledge sharing among heterogeneous application system and cross-functional teams is increasing.

The various information modeling approaches in engineering design and analysis taken by researchers are outlined in three levels: product information modeling at the data level, the information level, and the knowledge level. Data is understood as discrete, atomistic, tiny packets that have no inherent structure or necessary relationship between them. In contrast to data, information is data that is structured and put into context, so that it is transferable, but the immediate value of information depends on the potential of the user to sort, interpret and integrate it with their own experience. Knowledge goes one step further and implies the combination of information with the user's own experiences to create a capacity for action [9].

Initial Graphics Exchange Specification (IGES) [10] is a typical example of this standard mainly for 2D drawing layouts, and supports only data level exchange [6]. So, IGES is not able to transfer information and knowledge, although it is very well suited to transfer geometrical data. STEP (Standard for the Exchange of Product model data) is a standard of a computer-interpretable representation and an exchange of product data, and is successful in the transfer of product shape in terms of its geometry and topology [11]. However, ISO 10303 focuses on the translation of terminologies from one CAD system to another. It does not attempt to translate the meaning associated with the design from one context to another. Moreover, information that is lost in one context may be needed in another [12]. As a result there is often a drop in the knowledge accessibility level even the information level when transferring from one process to another.

An effort of significant relevance is the development of the Process Specification Language (PSL) at the National Institute of Standards and Technology (NIST) [13]. The Core Product Model (CPM) [14] was developed at NIST as a high level abstraction for representing product related information, to support data exchange, in a distributive and a collaborative environment. The Product Lifecycle Management (PLM) concept that was proposed [15] to extend CPM holds the promise of seamlessly integrating all the information produced throughout all phases of a product's life cycle to everyone belonging to an organization at every managerial and technical level. However, the transfer formats are incapable of capturing knowledge, and making use of such information still depends on personal experiences.

The third level is the knowledge level. Efforts to add information and/or knowledge to the geometric data from the intelligent tool to enable the exchange of product model are discussed in [16]. It is shown that in a knowledge-based environment information/knowledge stored in the product model can be extracted and made accessible. In [12], an ontological approach is proposed to enable the exchange of features between applications. It models participating ontologies and creates a common intermediate ontology. Rules are manually specified to enable the mapping of concepts from one domain to another. In [5, 6], they propose a formal set of ontologies for classifying analysis modeling knowledge, based on the concept that engineering analysis models are knowledge-based abstractions of physical systems. However, the above research efforts mainly focus on certain aspects of product model exchanges, without involving the whole process of the AFEA.

For these reasons, an ontology-based framework for automated finite element analysis (AFEA) is proposed to assist users in defining the appropriate finite element model more easily, faster, and more experience independent, even with no human interaction. It provides a meaningful representation of the product model that will facilitate seamless interoperability between CAD and FEA, and will allow computer systems to 'understand' digital design models on the level of engineering meaning and not just geometry. It also supplies a set of intelligent tools that can generate high-level abstractions of the analysis objectives, identify problem type, and provide an appropriate solution routine. Such framework for finite element analysis has great potential for improving the overall efficiency and reliability of analysis.

The paper is organized as follows: In Section 2, we present an ontology-based framework to solve these problems through the formal representation of product information and the application of AI techniques. The knowledge found in design and FEA theories is described in Section 3 through an ontology approach, including such concepts as physical causal relationships and conceptual dependencies. Section 4 addresses a three-stage knowledge-based finite element analysis method to reuse domain knowledge, and to enable the automation of the FEM analysis process. We illustrate the framework through an implementation in a structural engineering domain in Section 5. Finally, this paper concludes with observations and scope for future work.

2. The framework for AFEA with an ontology-based approach

We first concentrate on the product development process supporting AFEA. Based on this, we can choose the data exchange format and transformation mode to complete the products knowledge transfer.

2.1 The process of AFEA

In traditional finite element modeling, the analysis attributes (non-geometry information such as loads, boundary conditions, and analysis type) are set after the finite element mesh generation. However, the effect of these analysis attributes must be considered in model simplification and mesh generation. Otherwise, the generated numerical model might be unreasonable and FEA would produce big inaccuracy, possibly giving incorrect results [17]. Ideally, in order to simplify the model and rationalize mesh generation, the non-geometry information (analysis attributes) should be extracted and the effect of these attributes should be taken into consideration. Some of this information (analysis attributes) is expressed explicitly in the design phase; others are implicitly included in the artifact structure, function and behavior; and others are available from logical deduction.

Thus, a knowledge-based approach should be used to acquire the analysis attributes during the design. According to [18], the process of FEA can be divided into three phases: problem definition, problem formulation and FEA solution routine, applying AI technologies to automate and/or support the tasks required in problem definition/formulation, and leaving other algorithmic tasks in solution routine to automate by the analysis subsystems of modern CAD systems and the finite element analysis systems. In this paper, we thus adapt three phases by expanding them with conceptual stages of design, and specify the roles of our three phases. The relationship between these three phases is shown in Fig. 1.

Problem definition requires that the designer processes product structure, marks them with engineering semantic, and raises unverified engineering issues from a design perspective. The knowledge-based system would mark the related knowledge through knowledge extraction and pass it on to the next phase. The problem formulation phase is where the knowledge-based system would change the engineering

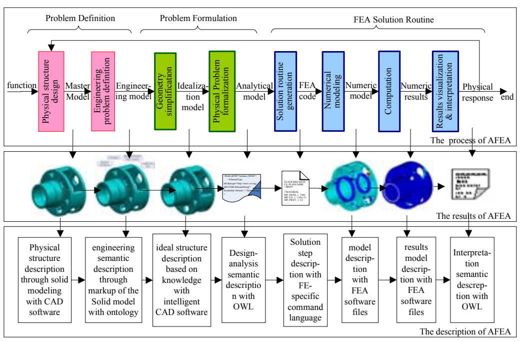


Fig. 1. The process of AFEA.

problems raised at the problem definition phase into formal FEA descriptions according to specific scenarios posted by engineers. The FEA solution routine is where the knowledge-based system creates the corresponding solving code by deduction from formulated problems and processes to final results.

2.2 The framework for AFEA

We can conclude from the analysis of Section 2.1 that the different models are associated by geometry data. So, if the geometry features are marked with engineering semantic, then by feature mapping between different models during the process, the semantic information integration of a whole design-analysis process can be implemented. Furthermore, if domain knowledge is represented by logical description statements which consist of semantic variables and instruction characters, then by semantic information mapping, the knowledge integration of an entire design-analysis process can be achieved. The framework of AFEA is listed below (Fig. 2, inspired by [16]). There are two issues to resolve, a perfect representation scheme for product modeling and a knowledge-based approach for AFEA.

2.2.1 The hierarchy transfer approach for product modeling

A hierarchy transfer approach is proposed to support AFEA at three levels. We use STEP or special data interface integrated by CAD/FEA software to transfer the data, OWL to semantic information and knowledge. The detailed transfer formats are listed

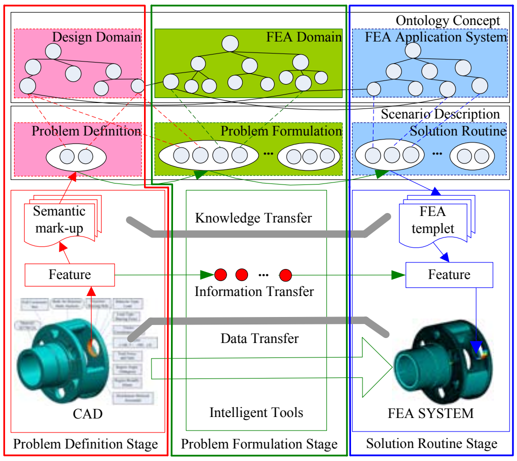


Fig. 2. The framework for AFEA.

below:

Data – At this level, many standards such as STEP, IGES, Parasolid, have been successful in the transfer of product shape in terms of its geometry and topology. Now many FEA software systems provide dedicated interface for data transformation. They are advantageous in geometry data transformation.

Information – Feature technology helps to incorporate the multi-viewpoint information into a single design representation in that this would allow each specialty a clearer view of the other specialties [19]. The large number of interacting CAD systems necessitates the development of a single neutral intermediate format. So we use the neutral semantic file OWL to transfer artifact function semantic information, FEA boundary and load information, etc., which is marked by the ontologies to the analysis phase by feature recognition and matching.

Knowledge – In this paper, we propose an ontology-based approach to build a large-scale engineering knowledge base to support the AFEA. Design ontology and FEA ontology are built in section 4. They are axiomatic ontologies that will be helpful for integrating with other disciplines (CAD, CAPP, PDM and optimization etc.) and other heterogeneous CAD/CAE systems.

2.2.2 The ontology-based three-stage AFEA method

Adapted with [18], an ontology based three-stage automated finite element analysis method and the application of AI techniques are described below:

Problem definition – Based on a user defined scenario, we can use the mentioned ontology to label artifacts semantically and to build the relation between semantics and the artifact parameters (geometry model). Reuse scenarios are an important part of the process by which modelers learn to apply the finite element method to problems within a domain. It involves reviewing and understanding archived models of similar problems and associated information including assumptions made in model development and model implementation techniques [7].

Problem formulation – The CBR system retrieves a similar scenario from the case repository to get the parameterized semantic FEA-template represented by Description Logic (DL) ontologies. It is a general FEA standard executing program for one product type by expressing specific artifacts related information (i.e., the action area and value of load) with semantic variables. We can re-use the FEA solution routine for different design objects with the same physical phenomena through instancing the values of semantic variables. The detailed process of using CBR and ontology matching technologies to change the engineering problem defined at the design phase into standard analysis problem and solution routine is illustrated in Section 4.

Solution routine – We can instantiate variables of the solution by integrating information. The above-mentioned semantic variable instantiation can be matched with the semantics in artifacts information template to find the artifact characters information. Furthermore, the information can be extracted and the semantic variables are instantiated to concrete values. To set-up an FE analysis, more product information is required than just geometry. Normally, this information is supplied to the FE-tool by assigning properties to the elements. In an automated environment, the element properties must be derived from the product information supplied by the intelligent tools. The main issue is to re-establish the relationship that was available in the product model between geometry and the attributes in the FE-tool. By integrating the data, we can use the geometry feature-related design and analysis semantic to implement the above-analyzed program on geometry entity. By using these features, which are unique for each geometric entity in both the design and FEA world, a 1-to-1 relation can be made. Every assignment of FEM properties, like element properties, loads and boundary conditions, based on the product attributes must be done indirectly through the relation with the geometric entity to where the element or node belongs [1, 17].

In the next section, the knowledge found in design and FEA theories is described through an ontology approach, including such concepts as physical causal relationships and conceptual dependencies.

3. Knowledge presentation for AFEA-the foundation of the proposed framework

Literature documents several methods that are proposed for building an ontology. Ontology can be simply defined as a formal, explicit specification of a shared conceptualization. A formal ontology refers to the complex semantics of concepts and the relations among concepts, their properties, attributes, values, constraints, and rules. This section describes the procedure of expressing AFEA knowledge into the intermediate ontology.

Our objective in this paper is to develop and implement an approach for data exchange between designers and analysts. To realize this, we decided to develop a design domain ontology and an FEA domain ontology. The design ontology is responsible for the definition of the physical structure and engineering problems. The FEA ontology abstracts physical problem solving by an FEA method.

3.1 Design domain ontology

This phase involves identifying key concepts and relationships in the domain of product design. The CPM is extended. Some key concepts concerned in this research can be briefly described in Fig. 3 (The figure only shows a hierarchical relationship between key concepts).

The Artifact represents a distinct entity in the design, whether that entity is the entire product or one of its subsystems, parts or components. It has a Specification and is an aggregation of Features, Function, Form, and Behavior. The Function specifies what the Artifact is supposed to do. The Form may be viewed as the proposed design solution to the problem specified by the function and consists of the artifact's Geometry (shape and structure may be synonymous to geometry in some contexts) and the Material it is composed of. The Feature represents any information in the Artifact that is an aggregation of Function and Form. Behavior represents how the artifact's form implements its Function; one or more causal models, such as Finite Element Analysis (FEA) or Computational Fluid Mechanics (CFM) models, may be used to evaluate it. More detail on the textual descriptions of the terms defining these concepts and relationships can be found in [15]. The MasterModel serves as the

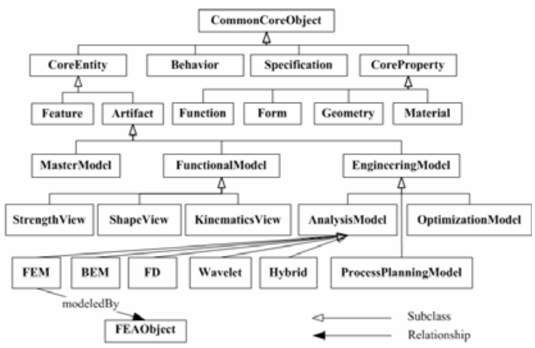


Fig. 3. Design domain ontology.

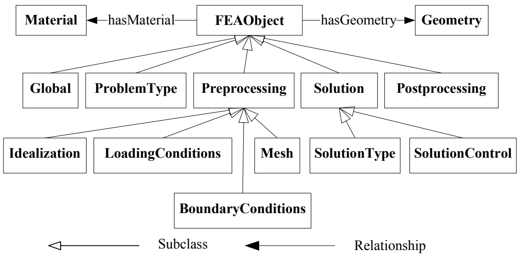


Fig. 4. FEA domain ontology.

```xml
<owl:Class rdf:about="#FEM">
<rdfs:subClassOf rdf:resource="#AnalysisModel"/>
<rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#modeledBy"/>
      <owl:allValuesFrom
rdf:resource="#FEAObject"/>
    </owl:Restriction>
    ……
  </rdfs:subClassOf>
</owl:Class>
```

(a) Encoding classes in OWL RDF/XML

```xml
<owl:ObjectProperty rdf:ID="hasForms">
  <rdfs:domain>
   <owl:Class>
    <owl:unionOf rdf:parseType="Collection">
     <owl:Class rdf:about="# Artifact"/>
     <owl:Class rdf:about="#Feature"/>
    </owl:unionOf>
   </owl:Class>
  </rdfs:domain>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#modeling">
  <rdfs:domain rdf:resource="#FEAObject" />
  <rdfs:range rdf:resource="#FEM"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#modeledBy">
  <owl:inverseOf rdf:resource="#modeling"/>
</owl:ObjectProperty>
……
<owl:DatatypeProperty rdf:ID="Author">
  <rdfs:domain rdf:resource="#Specification"/>
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >the author of artifact</rdfs:comment>
</owl:DatatypeProperty>
```

(b) Encoding properties in OWL RDF/XML

```xml
<Artifact rdf:ID="PlanetCarrier">
  <specified>
   <Specification rdf:ID="PlanetCarrier _2">
    <Project xml:lang="en">1.5MW_WT_2</Project>
    ……
    <Author xml:lang="en">Mike</Author>
   </Specification>
  </specified>
</Artifact>
```

(c) Encoding individuals in OWL RDF/XML

Fig. 5. The encoding ontology in OWL.

global repository of information on a product; in practice, it may be implemented as a centralized, distributed, federated or virtual database. The FunctionModel describes function characteristics and indicators. Each EngineeringModel represents an abstraction of the product of interest to a specific functional domain at a particular stage in the lifecycle of a product. We further subtype the EngineeringModel as analysis models, optimization models and process planning model etc. The class of analysis models has a number of subclasses according to specific numerical techniques, such as boundary element method, finite element method, finite difference method, wavelet methods, and hybrid methods involving one or more of the previous methods. As new methods emerge, additional subclasses can be included here [5, 15]. FEM is an idealized EngineeringModel of the MasterModel by the finite element method.

3.2 FEA domain ontology

The FEA domain ontology is intended to present a generic FEA activity. We present the formal ontology for the representation of FEA knowledge as Fig. 4, after extracting analysis modeling knowledge from engineers and incorporating this knowledge into a computational environment.

An FEA represents a distinct entity in FEA. It is an aggregation of Global, ProblemType, Preprocessing, Solution and Post processing. The Global describes the global information of FEA, including document specification, the unit system and the coordinate system. The ProblemType specifies the type of analysis problem, such as structural static analysis, kinetic analysis and thermodynamic analysis. The Preprocessing represents how to make geometry simplifications for building a solid model, to apply boundary conditions and load on idealization geometry and to appoint meshing type, finite elements type and size selection. The Solution describes the algorithm selection, an interactive control of algorithm parameters, and the Postprocessing represents results visualization, and results interpretation in the domain terms.

3.3 Encoding the ontology in OWL

OWL provides RDF/XML syntax to represent ontology-based domain knowledge. For example, the encoding of the class FEM in OWL RDF/XML is shown in Fig. 5(a). The subClassOf construct indicates that FEM is an AnalysisModel. The OWL: allVal-uesFrom restricted on the modeledBy object property means that a FE model is modeledBy FEAObject.

Moreover, both object properties and data properties are also encoded in OWL RDF/XML syntax. Fig. 5(b) shows the part of RDF/XML files that encodes the object properties hasForms, modeling, modeledBy, is_part_of as well as the data property Author. The OWL:objectProperty is used to indicate that the represented property is an object property. The rdfs: domain (rdfs:range) construct refers to the do-main (range) of a object property. Fig. 5(c) shows the encoding of individuals of classes in OWL RDF/ XML syntax. An individual of the class Artifact, namely PlanetCarrier is specified in this figure.

We use Protégé as ontology editing tools which supports the creation, maintenance and population of ontologies. It is a free, open source platform and offers direct communication with several rule engines, such as Jena, Racer and Jess. Furthermore, Protégé can be extended by way of a plug-in architecture and a Java-based application programming interface (API) for building knowledge-based tools and applications [20].

4. The intelligent support of AFEA--the realization of the proposed framework

Two FE analysis modes are described in [21]. Routine analysis entails identifying and updating parameter values that refer to device FEA attributes. Adaptive analysis involves adapting a model's implementtation strategy by adding new commands or removing existing commands, usually to integrate new information about the physical problem being studied. The new information is new product data or information about the physical context. So the former is available just by modifying the parameters of the FEA template, while the latter needs knowledge reasoning to adapt the template. According to these two modes, we propose an AFEA realization flow, shown in Fig. 6.

The realizations are somewhat different in the problem formulation stage, which is the second step of three steps in AFEA. (1) The designer or intelligent system uses the domain ontology introduced in Section 4 to mark the artifact with engineering semantic (function behavior, form etc.) in the design phase, and raises some engineering issues (only limited to FEA issues in this paper) which are applied to verify the current design. (2) Based on the defined issues, the intelligent system can get the general semantic descriptions of FEA by a formalized express and knowledge reasoning (adaptive reuse of parametric FEA models needs this operation). (3) The solution routine of the FEA application system is obtained through ontology concept mapping between the axiomatic ontology and the application ontology. Then the intelligent system matches FEA and engineering semantics information with artifact geometry information, and instantiates the logical described FEA template to the commands file of the application system. After that, the application system inputs the commands file and executes the preprocessing, solving and postprocessing actions. Thus, these three steps can be divided into three levels. The knowledge level is based on ontology and CBR. The information level includes information transformation and executable commands file generation. The data level is responsible for numerical model processing, solving and post-processing. In the next section we will introduce the main methods of the above-mentioned three procedures concisely, and the detailed implementation will be introduced in another paper. In order to

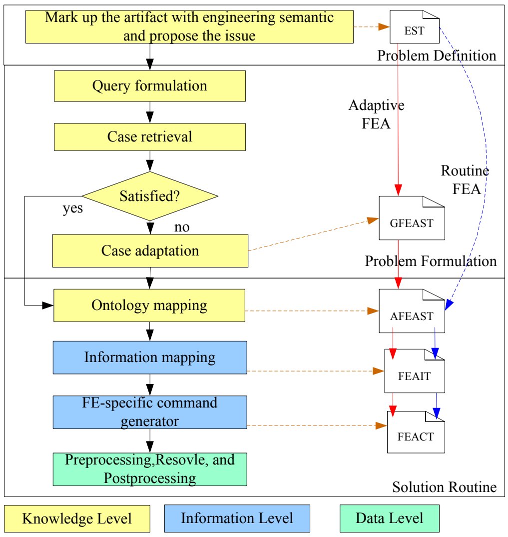


Fig. 6. The concise AFEA realization flow under the different analysis mode.

illustrate the process of knowledge transfer, the two software, viz., Unigraphics and ANSYS, and static analysis of the planet carrier are studied in detail.

4.1 Problem definition

Problem definition involves that a designer processes products structure is marked with engineering semantic and raises unverified engineering issues from a design perspective.

The semantic markup of artifact models is about encapsulating information from several engineering viewpoints within the same primary model. It allows computer systems to 'understand' digital design models on the level of engineering meaning and not just geometry [22]. We use design domain ontology as markup language, and employ a combination of feature techniques: design by features, interactive feature creation and automatic feature recognition. This will ensure the validity of the markup applied to a model, and will enable the interpretation and collation of the entire markup in order to produce a representation of the overall information state of the design. We apply and store the markup with the model representation and create an OWL file of the markup to allow it to be processed. The OWL file named Engineering Semantic Table(EST) stores problem type, engineering semantic such as function, behavior, thickness, material, and relevant geometric information such as, the key point coordinate required in the integration process of design and FEA.

4.2 Problem formulation

Once a feature or property of an object is marked in the design concept, FEA concept, and terminology, its significance from the viewpoint of either discipline is known by the intelligent system. Since many FEA tools have some form of programmatic interface, the intelligent system could therefore create the potential for carrying out FEA processes partially or fully automatically. This is done by identifying the artifact's engineering semantic, reasoning and creating the logic description of the FEA process. The main characteristic properties of the data and knowledge involved in the analysis are described by Peter Wriggers in [8]. Based on these conclusions, an ontologycal knowledge representation model, case-based reasoning and case adaptation reasoning are selected as the appropriate AI technologies for the intelligent support of engineering analysis.

Case Representation: A context-aware reasoning model is proposed in the field of digital documents retrieval [17]. This proposal for case representation of engineering problems is inspired by this similarity. The ontology-based domain knowledge representation model KRM is structured according to the used "physical problem context—formal FEA problem solution" reasoning schema (adapted with [18]) in the following way: KRM = {PC, SA, DP}, where PC is the set of elements of description of engineering problems context presented by design domain ontology, SA the set of logical description of FEA solution activity presented by FEA domain ontology, and DP the set of dependencies between properties of objects and engineering problems, and formal FEA problems. The model uses ontology described context as the cases' characteristics to transfer the raw current physical phenomenon into uniformly described FEA problem solution. After an engineering problem context is formally represented as the structured qualitative model, case retrieval and adaptation algorithms are carried out to compare the current scenario with the similar existing cases stored in the case library.

Case Retrieval: The similarity of the 2 cases is computed by the semantic meaning of their physical contexts and the semantic similarity of context is estimated by the concept distance in the concept hierarchy [7]. The semantic similarity between 2 cases is computed by cosine method. If the retrieval engine finds a similar case, it would present FEA solution description of the case as the result to the user; otherwise a new case would be generated by a hybrid expert system.

Case Adaptation: A hybrid expert system model was developed incorporating both key parts of the traditional rules-based expert system and the artificial neural network in [23]. It is suitable for this work. The knowledge base from experienced analysts and engineers is constructed, and an artificial neural network technique is explored to make the system learnable by training the rules base represented in SWRL, a rule language based on OWL. The FEA solution formal description retrieved case can be adapted through either the hybrid expert system or human-computer interaction between FEA experts and intelligent system. Furthermore, the feedback inference system is proposed, so that the decision-making unit is able to use an iterative method to make inferences.

Finally, the FEA solution logical description presented by an OWL file named General FEA Semantic Table (GFEAST) is obtained by the above reasoning process.

4.3 Solution routine

The input information of the supported solution routine process consists of a textual description of the problem, a graphical description of the technical object (draft, sketch) and implicit information obtained by the automated system in the interactive querying procedure. The reasoning process performed by the automated intelligent support system consists of the following fundamental steps.

Ontology mapping: Semantic mappings between equivalent terms, the application ontologies, and FEA ontologies are established and both the application ontology and the FEA ontology are encoded in OWL. Therefore, they have the same syntactic. The steps to translate the semantics of analysis information from FEA to an application A are as follows, referring [12]: First, the semantic equivalence matrices are determined; then the translation of analysis information is achieved by translating values of the properties of individuals to create equivalent individuals (as instances of equivalent concepts) in the other ontology. After mapping, an OWL file named Application FEA Semantic Table (AFEAST) is obtained.

Information mapping: This process is responsible for the re-association of design information and FEA information. The role of the intelligent tool is to import AFEAST and EST, read and analyze the content of the AFEAST, map the semantic variables from the tables and instantiate variables in AFEAST according to the DL syntax, semantics, axioms and facts [24]. In particular, each semantic variable in every logic statement in AFEAST is compared to the corresponding variable in EST and, as soon as a match is found, all the information stored in the EST is automatically mapped on the corresponding representation of the given AFEAST. As a matter of fact, every semantic variable in AFEAST has input value transferred from the design process, and an FEA Information Table (FEAIT) is created to store all the above information. After the information transfer from the EST to the FEAIT has been completed, a session proceeds with an FE-specific command generation.

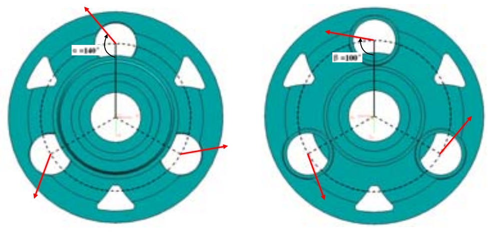
FE-specific command generator: This process is responsible for the translation between the OWL and the FE-specific command language, in this case APDL for ANSYS. We mark an APDL function semantic with terminology and concepts that the FEA ontology contains, and each parameter in function is parameterized. Every OWL template has a one-to-one correspondence with an APDL function. When the APDL generator inputs the FEAIT, a reasoning engine starts to parse it and maps the semantic with the OWL template. The resulting APDL fragment is instantiated with the parameter value from the FEAIT. After parsing the whole FEAIT file, an APDL file named FEA Command Table(FEACT) is created that will be used to generate the FE Model.

Simulation Realization: After the APDL commands have been carried out, the ANSYS proceeds to the FEACT and sets up the FE Model. Mesh is generated, materials properties are assigned, boundary conditions are positioned and attached to the given surface areas, then the analysis is performed and finally an automatic post-processing is carried out.

5. An illustrative example

In this section, the adaptation of FEA model to solve a new problem can be illustrated using static analysis models of a planet carrier.

The stored planet carrier case is one of components in the 1.5MW wind turbine gearbox. The physical model is established here. The three planets introduce bearing forces into the planet carrier. The six bearing forces were calculated analytically beforehand. In this



(a) Force at the front (b) Force at the rear

Fig. 7. The planet carrier model.

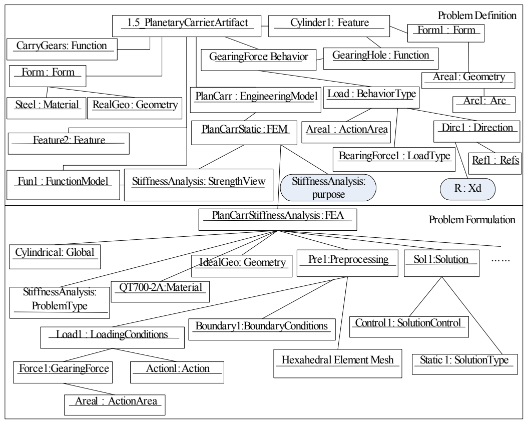


Fig. 8. Ontology fragment for the planet carrier.

load case, the forces take effect based on their amount and direction according to Fig. 7(a), 7(b). The planet carrier example considered in this section was presented in detail in Ref. [25], illustrating the representtation of both the load and the boundary conditions.

Fig. 8 shows a partial instance diagram of the planet carrier in ontology (the contents and values of the attributes of these instances are not shown in detail). The problem definition is described with instances of the Function, Form, Behavior, Material, Feature, FunctionModel, and EngineeringModel classes presented using design domain ontology. The figure also shows the problem formulation of this case with FEA ontology. For example, the 1.5_PlanetaryCarrier, an instance of Artifact, has the Material of QT700-2A and one Behavior of GearingForce is loading on the Area1. The case is encoded in OWL (represented in Fig. 5) and stored in the system.

Now, the current task is to design a planet carrier of 5MW wind turbine gearbox. In this work, the semantic markup has been applied to a geometric CAD model in NX3 API. The interfaces of problem definition are presented in Fig. 9(a).

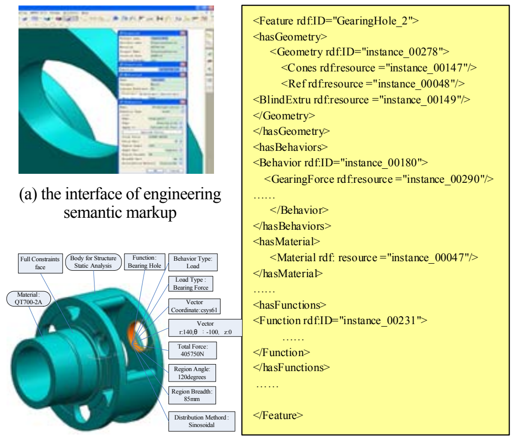
When the designer builds a product model of any of the domain world, e.g., by using a CAD tool, he/she can associate it with the corresponding class in the ontology, i.e., creating a semantic index. Though the product models are non-executable informal models, their association with formal ontology models makes it possible to reason about the product models automatically. An example of the semantic annotation of a CAD product model of planet carrier is shown in Fig. 9(b) and the result file named EST is shown in Fig. 9(c).

Based on the defined current case, the intelligent system finds the 1.5MW case, which has the similar context of this case. Then the system gets the general semantic descriptions of FEA by a formalized express and knowledge reasoning. Fig. 10(a) shows the representative snippets of GFEAST of the retrieved case. It has a semantic variable “hexahedral element” as the property value of “ElementType” in GFEAST. After mapping, an AFEAST file of current problem is obtained, as shown in Fig. 10(b), 10(c). The semantic variable “hexahedral element” is instanced with “Solid45” in FEAST_ANSYS, and “C3D8” in FEAST_ ABAQUS. In this work, we use ANSYS as the FEA software.

The AFEAST_ANSYS is read and analyzed by the intelligent system, mapping and instantiating the semantic variables in EST according to the DL syntax, semantics, axioms and facts. As a matter of fact, every semantic variable in AFEAST has input value transferring from the design process, and an FEA Information Table (FEAIT) is created to store all the above information, as shown in Fig. 11. Since the different models are associated by geometry data, so after the geometry features are marked with engineering semantic in EST, and domain knowledge is represented in AFEAST_ANSYS by logical description statements which consist of similar semantic variables and instruction characters in EST, the information mapping between them can achieve either the semantic information integration or the knowledge integration of a whole design-analysis process.

After the information transfer from the EST to the FEAIT has been completed, a session proceeds with an FE-specific command (in this case APDL for ANSYS) generation. As presented in Fig. 12, after parsing the whole FEAIT file, an APDL file named FEA Command Table(FEACT) is created that will be used to generate the FE Model.

In this case, the initial geometric model exported from UG with the format of Parasolid is imported into

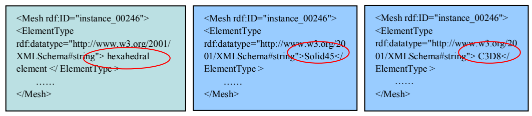


(a) the interface of engineering semantic markup

(b) a sketch of planet carrier after markup

(c) the EST output in OWL

Fig. 9. The engineering semantic markup of planet carrier.



(a) GFEAST (b) AFEAST_ANSYS (c) AFEAST_ABAQUS

Fig. 10. Semantic description of FEA Solution routine in different system.

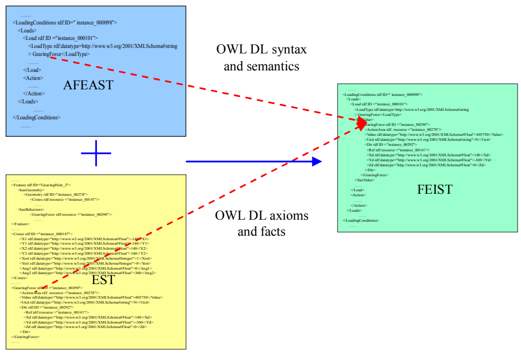
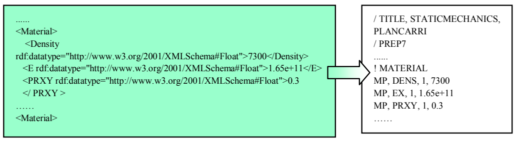


Fig. 11. Information mapping.

ANSYS by the APDL. The exchange of geometry data between CAD and FEA is achieved. ANSYS proceeds to the FEACT and sets up the FE Model. Mesh is generated, materials properties are assigned, boundary conditions are positioned and attached to the given surface areas, then the analysis is performed and finally an automatic post-processing is carried out. Fig. 13. is the automated FE model of current planet carrier in the 5MW project.



(a) FEIST (b) FEACT

Fig. 12. FE-specific command generation.

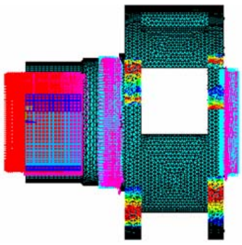


Fig. 13. The FE model of planet carrier.

6. Conclusions

The concept of AFEA presented assumes the use of an ontology-based approach. We propose such a scheme based on identifying and classifying structural configurations and fundamental analysis modeling knowledge into a set of formal ontologies described in OWL. This involves the development of a shared design and FEA ontology and specific application ontologies. Furthermore, a hierarchy transfer approach is proposed to achieve true inter-operability of different engineering object models. This makes the AFEA possible, since no human interaction is necessary to rebuild data/information/knowledge lost in the transfer. Moreover, a three-stage transfer method is applied to mark up the artifact in problem definition, to reuse this domain knowledge in problem formulation, and to enable automation of the FEM analysis process in solution routine. The approach can support the semantic presentation of a product both in the conceptual stages of design and in the post design-verify stages, and also, the automated finite element analysis.

This approach to AFEA is feasible, but it has not yet been implemented commercially, and developing ontologies to support engineering design/analysis modeling knowledge and implementing them is a work in progress. Currently, an automated knowledge-based system is being developed using the proposed models and algorithms. The system is being implemented as a web application on the J2EE platform. The developed models and algorithms are to be evaluated on the test knowledge base in the domain of structural analysis.

This method is not limited between CAD and FEA. It is readily expandable with other engineering domains, such as optimization, process planning, and manufacture. The authors' future work is to develop more diverse domain-specific engineering ontologies to get a more extensive multidisciplinary integration. The development of an intelligent post-processing system provides suitable advice when considering that a description of the design structure's critical area is also important work.

Acknowledgment

This work is supported by the Program for New Century Excellent Talents in the University, China (No.NCET-05-0285). The authors gratefully acknowledge the co-workers who aided in the preparation of this paper.

References

[1] D. Bojan, Finite element mesh design expert system, *Knowledge-Based Systems*, 15 (2002) 315-322.

[2] M. T. George and J. F. Steven, Knowledge-Based Assistance for Finite-Element Modeling, *IEEE Intelligent Systems*, 11 (3) (1996) 23-32.

[3] M. Pinfold and C.Chapman, The application of KBE techniques to the FE model creation of an automotive body structure, *Computers in Industry*, 44 (1) (2001) 1-10.

[4] N. Marina and D. Bojan, Intelligent FEA-based design improvement, *Engineering Applications of Artificial Intelligence archive*, 21 (2008) 1239-1254.

[5] I. R. Grosse, J. M. Miltion-Beniot and J. C. Wileden, Ontologies for supporting engineering analysis models, *AI EDAM: Artificial Intelligence for Engineering Design, Analysis, and Manufacturing*, 19 (2005) 1-18.

[6] Y. Masaharu, U. Yasushi, T. Hideaki, S. Yoshiki, N. Yutaka and T. Tetsuo, Physical concept ontology for the knowledge intensive engineering framework, *Advanced Engineering Informatics*, 18 (2) (2004) 95-113.

[7] N. Udoyen, and D. Rosen, Description Logic Representation of Finite Element Analysis Models for Automated Retrieval, *Journal of Computing and Information Science in Engineering*, 8 (2008) 1-10.

[8] P. Wriggers, M. Siplivaya, I. Joukova and R. Slivin, Intelligent support of the preprocessing stage of engineering analysis using case-based reasoning,

Engineering with Computers, 24 (2008) 383-404.
[9] V. Gokula and C.Amaresh, Understanding the Knowledge Needs of Designers during Design Process in Industry, *Journal of computing and information science in engineering*, 8 (1) (2008) 103-145.
[10] ANSI/US PRO/IPO 100, Initial graphics exchange specification IGES5.3, (1996).
[11] J. Owen, *STEP: An Introduction. Information Geometers*, Winchester, U.K., (1993).
[12] L. Patil, D. Dutta and R. Sriram, Ontology-Based Exchange of Product Data Semantics, *IEEE Transactions on Automation Science and Engineering*, 2 (2005) 213-225.
[13] NIST, The Process Specification Language, http://www.mel.nist.gov/psl/, (2005).
[14] S. J. Fenves, *A core product model for representing design information*, National Institute of Standards and Technology, USA, (2001).
[15] R. Sudarsan, R. J. Fenves, R. D. Sriram and F. Wang, A product information modeling framework for product lifecycle management, *Computer-Aided Design*, 37 (2005) 1399-1411.
[16] M. Nawijn, M. J. L. Van-Tooren, J. Berends and P. Arendsen, Automated Finite Element Analysis in a Knowledge Based Engineering Environment, *44th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, (2006).
[17] Y. F. Zhang, *Research on application of Feature-based modeling in finite element analysis modeling*, Doctoral Thesis, school of civil engineering, Wuhan University, China, (2004).
[18] P. Wriggers, M. Siplivaya, I. Joukova and R. Slivin, Intelligent support of engineering analysis using ontology and case-based reasoning, *Engineering Applications of Artificial Intelligence*, 20 (2008) 709-720.
[19] D. Davies and C. A. McMahon, Multiple view-point design modeling through semantic markup, *Proceedings of IDETC/CIE 2006 ASME 2006 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, Philadelphia, Pennsylvania, USA, (2006).
[20] W. Zhao and J. K. Liu, OWL/SWRL representation methodology for EXPRESS-driven product information model Part II: Practice, *Computers in Industry*, 59 (2008) 590-600.
[21] N. Udoyen, *Information Modeling for Intent-Based Retrieval of Parametric Finite Element Analysis Models*, Doctoral Thesis, Mechanical Engineering, Georgia Tech, Georgia, USA (2006).
[22] W. Sun, Q. Y. Ma, L. Guo and T. Y. Gao, Product knowledge documents retrieval based on hybrid semantic model, J*ournal of Chongqing University (Nat Sci Ed)*, 31 (2008) 1198-1203.
[23] S. X. Li and X. Qiao, Hybrid expert system and its application on finite element modeling, *Chinese journal o f computational mechanics*, 14 (1997) 400-406.
[24] Ivan Herman, Web Ontology Language (OWL), http://www.w3.org/2004/OWL/, (2007).
[25] Z. H. Zhang, Z. M. Liu, H. P. Zhang and P. Yang, Structural analysis and optimization on the planet frame of large scaled gearbox of wind turbine, *Journal of Machine Design*, 25 (2008) 54-56.

Wei Sun is currently a professor and a PhD candidate supervisor in theSchool of Mechanical Engineering, Dalian University of Technology, China. His main research interests include knowledge-based product digital design, design and optimization of complex mechanical equipment, mechanical transmission and structure CAD/CAE, management of product design resource and process. E-mail: sunwei@dlut.edu.cn

Qin-Yi Ma is currently a PhD candidate in the School of Mechanical Engineering, Dalian University of Technology, China. Her research interests include knowledge-based product digital design, interoperability and the application of ontologies in CAD/CAE. E-mail: onlypony@163.com

Shuang Chen is currently a Master's candidate in the School of Mechanical Engineering, Dalian University of Technology, China. Her research interests include knowledge-based product digital design, structure CAD/CAE. E-mail: chen355113@gmail.com