# Altair SimSolid™ Technology Overview

## Abstract

Computer Aided Design (CAD) and Computer Aided Engineering (CAE) are intrinsic parts of modern engineering. CAE simulation tools are extremely important because they allow for performance validation and optimization of a product design before the product is physically created. Today this is typically done using software based on Finite Element Analysis (FEA).

The weak link in traditional FEA is the need to create an accurate and efficient finite element mesh. The meshing process would typically fail if done on original geometry or produces excessive number of elements which would often make FEA impractical. In case of assemblies, incompatible meshes on adjacent parts make handling part connections extremely difficult. In practice, geometry model simplification is a must for a successful FEA. But the simplification is a non-trivial step which requires a highly skilled specialist in both FEA and CAD technologies.

Altair SimSolid™ is intended to drastically streamline simulation process and increase engineering productivity by replacing the underlying FEA technology, rather than through improvements to the user interface. New technology eliminates the most failure-prone and skills/labor consuming stages of analysis process.

This whitepaper was written to provide users of SimSolid simulation software with basic understanding of how the technology works. The theoretical foundation of SimSolid is discussed with respect to mathematical background, computer implementation, and positioning among other numerical methods.

## Product Summary

SimSolid is a simulation software application which performs statics, dynamics and thermal, analyses of structural systems. SimSolid always uses fully featured, not simplified, solid geometry models in the analysis and does not use a mesh.

The SimSolid computational engine is based on breakthrough extensions to the theory of external approximations. External approximations are a generalization of Finite Element Method (FEM) in terms that:

- Absolutely arbitrary geometrical shapes can be used as “finite elements”;
- Basis functions which approximate the field of interest in the “element” can be of arbitrary class and are independent of the “element” shape, alternatively to strictly structured polynomials used in conventional FEA.

SimSolid controls the solution accuracy using multi-pass adaptive analysis. Adaptivity can be defined globally or on a part. Adaptivity is always active. SimSolid provides smart connections for bolts and welds to make assembly modeling easier and more robust. The SimSolid methodology is fast and efficient. It provides superior performance metrics for computational time and memory footprint that allow very large and/or complex assemblies to be solved quickly on desktop class PC’s.

# SimSolid Theoretical Background

The following discusses the theoretical background of SimSolid and its software implementation workflow. Then it is compared to methods used in traditional FEA.

### Overview of Initial Research

The Ritz-Galerkin method invented at the beginning of 20th century for the approximate solution of boundary value problems assumes that functions that approximate the solution are analytical functions defined on the whole domain of interest. In practical applications these functions were either trigonometric or polynomials which were infinitely smooth, i.e. they had an infinite number of derivatives. There were two main problems with such functions. Firstly, it was difficult or impossible to construct such functions that a priori meet essential boundary conditions on boundaries of arbitrary domains (in structural analysis the conditions appear as displacement constraints). And secondly, the equation system built on such functions was ill-conditioned and numerically unstable which did not allow solving real life problems with enough accuracy.

The Finite Element Method (FEM) that appeared in 1950s was just a different implementation of the classical Ritz-Galerkin approach, but it succeeded in solving of both – constraints and numerical instability issues because it consistently used functions with local supports called finite elements. Though locally the basis functions of finite elements were infinitely differentiable standard polynomials, global basis functions assembled from local polynomials were not smooth at all – even their first derivatives were discontinuous. The FEM's success proved that requirements to the continuity of the approximation functions must be met only to a certain degree - just enough to provide finite energy when they are substituted into an energy functional of a boundary value problem. The spaces of such functions were introduced and investigated by Sobolev in the 1930s.

The next step in the relaxation of continuity requirements on approximation functions was the introduction of the concept of external approximations [1]. The name "external" was used in the following context. When approximation functions belong to a Sobolev space of functions with finite energy the approximation is called "internal" which means that while the approximation is refined, and the solution is converging to the exact solution, the approximation functions are always inside the Sobolev space. Alternatively, in external approximations, the approximation functions do not belong to a Sobolev space at every refinement step (they have infinite energy), but in the limit, when number of degrees of freedom (DOF) approaches infinity, the limit function must belong to the corresponding Sobolev space, i.e. it must recover the necessary smoothness properties. The abstract theory of external approximations was developed in reference [2].

The technological foundations of SimSolid have been published in reference [3]. It develops the abstract theory of external approximations. In reference [2] it was applied to the particular case of approximations by finite elements under the assumption that the elements are of absolutely arbitrary shape. In the result the necessary and sufficient condition of external approximations by finite elements has been established and convergence theorems have been proven. It was also shown that the theorems were constructive, i.e. they not only defined hallmarks of external approximations, but also provided a mechanism to build them.

### Theoretical Background

An abstract boundary value problem (BVP) is formulated as to find a function $U$ which fulfills the equations:

$$AU = h \text{ inside the domain } \Omega \tag{1}$$

$$LU = g \text{ at the domain boundary } \Gamma \tag{2}$$

where $A$ and $L$ are differential operators.

Some boundary value problems can be equally formulated in a variational form such as to find a function $U$ which provides a functional $F(U)$ at minimum value, where the functional $F(U)$ is usually an energy functional.

In 1908 W. Ritz proposed a method of finding an approximate solution of a boundary value problem by approximating it with a linear combination of some basis functions

$$U_h = \sum_{i=1}^{n} a_i p_i \tag{3}$$

where $a_i$ are unknown factors, and $p_i$ are basis approximation functions.

The factors $a_i$ are found by minimizing the energy functional

$$F(\sum_{i=1}^{n} a_i p_i) = min \tag{4}$$

If the boundary value problem is linear, then the minimization problem (4) can be reduced to a linear algebraic equation system with respect to the factors $a_i$

$$\boldsymbol{Kd} = \boldsymbol{f} \tag{5}$$

here $\mathbf{K}$ is a symmetric matrix, $\mathbf{d}$ is the vector of unknown factors $a_i$, and $\mathbf{f}$ is a right-hand side of the system.

In FEM the matrix $\mathbf{K}$ is called stiffness matrix, the vector $\mathbf{f}$ is called load vector, and the factors $a_i$ are called degrees of freedom.

In 1915 Galerkin proposed another approximate method of solving the boundary value problem (1)-(2). According to the Galerkin method the unknown solution $U$ is approximated as

$$U_h = U_0 + \sum_{i=1}^{n} a_i p_i \tag{6}$$

Where $U_0$ is some function which fulfills nonhomogeneous boundary conditions (2), $p_i$ are analytical approximation functions which fulfill homogeneous boundary conditions, $a_i$ are unknown factors.

Substitution of (6) into (1) results in the residual

$$R = AU_0 + \sum_{i=1}^{n} a_i A p_i - h \tag{7}$$

The unknown factors $a_i$ are found from the equation system

$$\int_{\Omega} R p_i \, d\Omega = 0 \tag{8}$$

If a boundary value problem is linear, then system (8) is a system of linear algebraic equations.

The Galerkin method does not use a variational formulation of the boundary value problem. Therefore, its applicability is much wider.

Ritz and Galerkin methods proved to be effective means of solving problems in engineering and science. At the same time mathematical justification of the methods faced significant difficulties which were solved with the introduction of functional analysis as a mathematical discipline.

Modern theory of the Ritz-Galerkin method is based on the concept of the weak formulation of the boundary value problem. The weak formulation of a boundary value problem consists of finding a function $u \in V$ from a corresponding Sobolev space which fulfills an abstract variational equation

$$a(u, v) = f(v) \text{ for any function } v \in V \tag{9}$$

here $V$ is some subspace of a Sobolev space, $a(u,v)$ is generally an unsymmetrical bilinear form which is continuous on the space product $\mathrm{V} \times \mathrm{V}$, $f(v)$ is some linear form in $\mathrm{V}$.

In structural analysis, the Sobolev space is a space of functions with finite strain energy.

In the Ritz-Galerkin method the space $V$ is approximated with some finite-dimensional space $X_h$, and the approximate solution is found in form (3) where the functions $p_i$ belong to the space $X_h$. Therefore, the discretized formulation of a boundary value problem is:

Find a function $U_h \in X_h$ which fulfills the equation

$$a(U_h, V_h) = f(V_h) \text{ for any function } V_h \in X_h \tag{10}$$

Substitution of (3) into (10) results in a linear algebraic equation system from which unknown factors $a_i$ are found.

In the classic Ritz-Galerkin method $X_h$ is a space of analytical functions defined on the whole domain $\Omega$; the factors $a_i$ have no physical meaning. In conventional Finite Element Method $X_h$ is a space of piecewise polynomials and the factors $a_i$ are values of the function $U_h$ in the nodes of finite elements. In structural analysis they are displacements of the nodes.

Many modifications of Ritz-Galerkin methods have been invented. They differ by the variational equations (9) and by the classes of basis functions (3) used to approximate the solution. The same boundary value problem can have several equivalent formulations (9) which differ by the spaces $V$.

**External Finite Element Approximations**

As already mentioned, internal finite element approximations are built on functions that belong to a corresponding Sobolev space. These functions must meet certain continuity conditions on inter-element boundaries. For instance, when 2D or 3D theory of elasticity problems are under consideration, the functions need to be continuous between finite elements. For plate bending problems not only the functions, but their first derivatives must be continuous as well.

The continuity conditions are quite restrictive. They can be met only for very simple shapes of finite elements using standard interpolation polynomials as basis functions. These polynomials are associated with the element nodes. To provide inter-element compatibility the same interpolation functions are used to represent the finite element shapes. In the case of curved boundaries, mapping onto a canonical element is used to provide compatibility. The geometry of finite elements and their approximation functions are tightly coupled.

To improve the approximation quality of finite elements, researchers invented incompatible or nonconforming finite elements. In incompatible elements, the interpolation basis functions of the elements of standard shape are enriched by some other polynomials. The additional functions create discontinuity across inter-element boundaries, but incompatible finite elements often provided much better accuracy than the compatible ones. This resulted in difficulties of mathematical proof of convergence and in inconsistencies of results.

A comprehensive theory of external finite element approximations was developed in reference [3]. In the theory the word “finite element” was used to designate an arbitrary shaped sub-domain of the domain $\Omega$, so the definition of finite elements was not restricted anymore to canonical shapes or other shapes obtained from a canonical shape by mapping. The entire domain $\Omega$ could be considered as one finite element, and therefore, for assemblies a part of an assembly could be one “finite element” in FEM terminology. Another assumption was that the approximation functions inside the finite element could be arbitrary - not necessarily polynomials. The only requirement was that the functions belong to the corresponding Sobolev space, so they need to be sufficiently smooth inside the element.

The task was to find conditions under which the non-conforming approximations would be external approximations, i.e. they would converge to the exact solution of a boundary value problem from “outside” of a Sobolev space. The necessary and sufficient condition for the external approximations was found. The condition happens to be constructive – its

formulation gives the recipe for building finite elements that meet the condition. Convergence theorems and error estimates have been proven.

It was shown that the necessary and sufficient condition for a finite element approximation to be external is:

$$\langle \delta, \gamma U \rangle = 0 \tag{11}$$

Here <,> is the duality pairing in certain functional spaces defined on inter-element boundaries, $\delta$ and $\gamma$ are some operators, and $U$ are approximation functions defined inside the element.

As one can see, condition (11) does not relate to a boundary value problem formulation, or to a solution method (Galerkin, Ritz, etc.). It imposes constraints on basis functions of finite elements which just guarantee that the limit approximation function will belong to the corresponding Sobolev space, so it will have the necessary smoothness properties.

Therefore, even before the solution method is chosen (Galerkin, Ritz, etc.), one may construct finite element spaces that have important properties. These properties can be just "good to have", as, for instance, when solving elasticity problems, it is not required to use functions that fulfill the equilibrium in volume, but it might be useful because the use of such functions increases accuracy and reduces the number of DOF. Other properties can be crucial, for instance, only divergence-free functions can provide unconditionally stable solutions for incompressible materials.

Then condition (11) can be extended by continuity from the duality pairing into an inner product in other function spaces:

$$(g, \gamma U) = 0 \tag{12}$$

Here $g$ are functions defined on inter-element boundaries, they are called boundary functions. Boundary functions are functions of surface parameters and they generate boundary DOF that are integrals of products of boundary functions onto finite element basis functions over the finite element boundary:

$$\int_{\Gamma} g_k \gamma U \, d\Gamma, k = 1,2, \dots, N \tag{13}$$

Here $\Gamma$ is the boundary of the finite element, $g_k$ are functions defined on the boundary of the finite element, and $U$ is a function to be approximated on the element (for example displacements in structural analysis).

For comparison, the degrees of freedom in the FEM are the value of the function $U$ in the node $i$ of a finite element:

$$U(x_i, y_i, z_i) \tag{14}$$

The functions $g_k$ in expression (13) are basis functions from finite dimensional space $G_h$ of functions defined on the element boundaries. They can be arbitrary, the only requirement is that the spaces $G_h$ must be dense in the space of boundary functions, i.e. they must be able to converge in the space of boundary functions. The latter is easily fulfilled in case $g_k$ are polynomials or piecewise polynomials defined on the element boundaries.

The functionals (13) are called boundary degrees of freedom. They do not have physical meaning and they represent approximation functions from space of finite elements compatible when the number of boundary DOF converges to infinity. The boundary DOF are responsible for meeting inter-element continuity conditions and essential boundary conditions. In adaptive solution the number of the boundary DOF is managed automatically to meet the convergence criteria.

The boundary DOF (13) are not the only DOF produced when external approximations are built. Other DOF are called internal DOF because they are associated with the element volume. Internal DOF are defined automatically when the approximation of the solution within a finite element is being built. Finally, the approximation of a function $U$ on the element looks as follows:

$$U_h = \sum_{i=1}^{n} a_i(U) p_i + \sum_{k=1}^{N} \left( \int_{\Gamma} g_k \gamma U \, d\Gamma \right) p_k \tag{15}$$

Here $a_i$ are the internal DOF of the element (some factors), $p_i$ are basis functions for the internal DOF, $\int_\Gamma g_k \gamma U \, d\Gamma$ are the boundary DOF, $p_k$ are basis functions for the boundary DOF.

The basis functions $p_i$ and $p_k$ constitute a finite-dimensional space $P$ of approximation functions of a finite element. It was proven that for convergence the space $P$ must be complete, for instance, in case of polynomial space it should contain all polynomials up to a certain degree assigned to an adaptive iteration.

Basis functions of a finite element are not pre-defined because the element has an arbitrary shape. They are built on-the-fly during a solution run. What is pre-defined at an adaptive pass is the whole space *P* of approximation functions of the element. The algorithm of building basis functions of an element at an adaptive pass works as follows:

- A set of boundary functions $g_k$ is defined;
- A complete space $P$ of approximation functions of the element is defined by choosing a complete set of generic basis functions. In the case of polynomial spaces, a complete space of polynomials of a certain degree is specified. For instance, generic second-degree polynomials for 3D problems are:

$$\{1, x, y, z, x^2, xy, y^2, xz, z^2, yz\}$$

- Generic basis functions are generated automatically on-the-fly for every sub-domain during the solution when the stiffness matrix of a sub-domain is evaluated;
- The basis functions $p_i$ and $p_k$ are found automatically by solving a certain system of linear algebraic equations.

After basis functions of an element have been found, the element stiffness matrix and load vector are evaluated the same way as it is done in conventional FEM by integrating energy over the element volume and loads over the element boundary.

**Geometry-Function Decoupling**

Geometry-function decoupling is the core feature of the SimSolid technology. As one can see from the above, the basis functions of an arbitrary element are built from generic basis functions on-the-fly during the solution. Neither element geometry representation is used in building the generic functions, nor the functions dictate the shape of the element. The only requirement to the space $P$ of approximation functions of an element is that $P$ must be a subspace of a corresponding Sobolev space associated with the formulation of boundary value problems. Therefore, any combination of generic basis functions is allowed provided they are linearly independent.

The geometry-function decoupling proved to be the key feature which provides better performance, better accuracy, robustness, less computer resources, less modeling errors.

The following substantial benefits can be realized when finding an accurate solution for a specific problem, or managing adaptive solutions:

1. It is possible to build special approximations that make approximate solutions of boundary value problems unconditionally stable. For instance, when parts made of incompressible materials are simulated, SimSolid uses divergence-free functions which exactly meet the incompressibility condition. Here is an example of some generic divergence-free 3D functions of 3rd degree, where $(u, v, w)$ are displacement components:

$$u = -xz^2, v = yz^2, w = 0$$

$$u = -3xz^2, v = 0, w = z^3$$

$$u = -2xyz, v = y^2z, w = 0$$

$$u = -xyz, v = 0, w = yz^2$$

$$u = -xy^2, v = 0, w = y^2z$$

2. Neighboring parts may have approximation functions of different classes. For instance, in case an assembly contains parts made of compressible and incompressible materials (rubber insertions, or cavities with a liquid) the approximation functions for the incompressible material are built as special divergence-free functions. On the neighboring parts with compressible material, regular functions like standard polynomials are used.

3. It is always possible to use basis functions that a-priori fulfill the governing equations of boundary value problems which provide better accuracy and reduce the number of DOF. For instance, thermo-elastic problems are solved using a complete polynomial solution of the corresponding governing equations:

$$\begin{aligned}(\lambda+\mu)\frac{\partial\varepsilon}{\partial x}+\mu\Delta u &= \frac{\alpha E}{1-2\nu}\frac{\partial T}{\partial x}\\ (\lambda+\mu)\frac{\partial\varepsilon}{\partial y}+\mu\Delta v &= \frac{\alpha E}{1-2\nu}\frac{\partial T}{\partial y}\\ (\lambda+\mu)\frac{\partial\varepsilon}{\partial z}+\mu\Delta w &= \frac{\alpha E}{1-2\nu}\frac{\partial T}{\partial z}\end{aligned} \tag{16}$$

here $(u, v, w)$ are displacement components,

$$\varepsilon = \frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}$$

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}, \mu = \frac{E}{2(1+\nu)}$$

$$\Delta = \frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}+\frac{\partial^2}{\partial z^2}$$

Further, $\alpha$ is the thermal expansion coefficient; $E$ is Young's modulus, $\nu$ is Poisson's ratio, and $T$ is the temperature field. The equation system (16) is non-homogeneous. For instance, when:

$$E = 1, \nu = 0.25, \alpha = 1$$

And the temperature field is described by a monomial:

$$T = ax^m y^n z^p$$

the solution of the non-homogeneous problem for a=1, m=0, n=2, p=3 is:

$$u = 0, v = 0.1667yz^5, w = 0.4167y^2z^4 - 0.02778z^6 \tag{17}$$

Here is an example of a polynomial solution of the homogeneous equations (16):

$$u = 20x^4z - 20x^2z^3, v = 20x^3yz - 20xyz^3, w = 8x^5 - 60x^3z^2 \tag{18}$$

When solving a thermo-elastic problem, polynomial approximations of temperature $T$ are imported from thermal analysis, functions of type (17) are generated for every element, and generic functions of type (18) are used to build basis functions of the elements.

For heat transfer problems harmonic polynomials are used as basis functions which precisely fulfill the corresponding equation of heat transfer.

Here are some third order generic harmonic functions:

$$f_1 = x^3 - 3xz^2$$

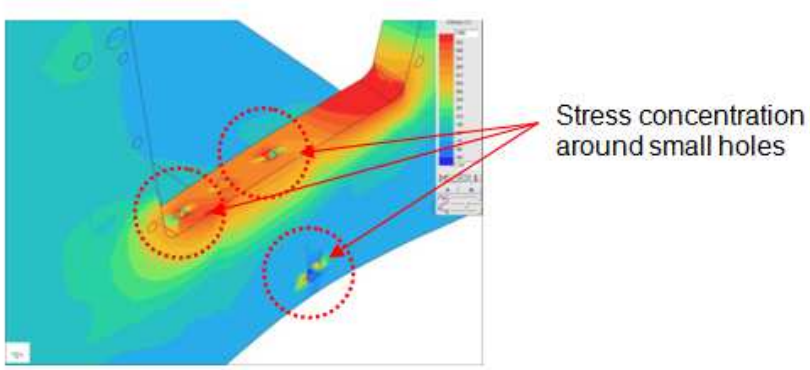
$$f_2 = x^2y - yz^2$$

$$f_3 = xy^2 - xz^2$$

$$f_4 = y^3 - 3yz^2$$

$$f_5 = 3xz^2 - z^3$$

4. The approximations are always built in the physical coordinate space without mapping onto a canonic shape. Therefore, the properties of generic basis functions are preserved throughout the solution which eliminates a substantial source of approximation errors.
5. A complete set of basis functions is always used to approximate solutions on a sub-domain. Completeness means that no functions are missing from a space of a certain degree. For instance, if the solution is approximated with harmonic polynomials of degree 5, then all harmonic generic polynomials of degree 5 are included into the approximation space of a sub-domain. This provides high accuracy, ease of building p-adaptive solutions globally and locally, and ease of implementation of new types of problem-specific basis functions.
6. Geometry-functions decoupling allows effectively handle assemblies of parts with incomparable geometries in terms of size and shape (multi-scale assemblies).
7. Local effects like concentrated forces, cracks, stress concentration, etc., can be easily simulated by enriching the approximation space of sub-domains with special functions that possess corresponding characteristics associated with the feature:




## SimSolid Software Implementation

The SimSolid software implementation workflow is summarized in the following:

**Step 1.** Process geometry. Assembly geometry is imported using either file based or CAD embedded readers. Direct document access is provided to all mainstream CAD systems including Catia, NX, Creo, SOLIDWORKS, Inventor, SolidEdge, and Onshape as well as common neutral file formats such as JT, STEP, Parasolid, STL and ACIS. To improve performance, CAD faces are stored internally in an efficient faceted form along with assembly, subassembly, part and face tree structure.

During model import, SimSolid will automatically identify special part geometry such as bolts, nuts and washers and part features such as thin walls, through holes and fillets. This information will be used for initial adaption conditions as well as automation of sliding/bonded connection properties.

**Step 2.** Create part connections. Bonded and sliding contact interfaces between parts of assembly are found automatically. More general separating and closing non-linear contact may be specified manually.

**Step 3.** Analysis parameter specification. An analysis type (linear or nonlinear static, modal, thermal, transient dynamic) is selected and boundary conditions are applied to the model. At this point, the model is ready to analyze. No mesh creation steps are required by the user.

**Step 4.** Adaptive solution is performed. SimSolid employs a proprietary adaptive technology to automatically refine the solution in the areas where it is necessary to achieve the highest accuracy. The maximum number of adaptive iterations is set on either a global or local (part) basis. Adaptivity is always active in the solver methods.

**Step 5.** Results display mesh is built over the geometry. A display mesh is used to show results on the model geometry. Result quantities are evaluated in the vertices of the display mesh. The quantities evaluation is done "on-the-fly" when the post-processor is launched. The vertex values are not saved, instead the analytical approximants of the field of interest are stored which provides significant savings in memory in particular for non-linear and dynamic response analyses. It also allows for effective coupled analyses when results of one analysis are directly used in analytical form in other analyses.

**Step 6.** Re-analysis is fast. SimSolid remembers the output display mesh. Additional analyses or load cases can be done quickly. Unique to SimSolid is the ability to incrementally increase the solution detail on a part or assembly basis. Simply change the solution settings and rerun the model. No laborious remeshing is required.

**Step 7.** Design Studies. SimSolid provides unique design study capabilities. Design studies are used to collect and quantify design geometry variation. Each SimSolid project can contain multiple design studies and each study can contain multiple analyses. SimSolid design studies are associative to CAD geometry updates. As the geometry evolves, the study can be updated without having to recreate the existing analysis configuration settings.

## Comparing Traditional FEA and SimSolid Modeling Methods

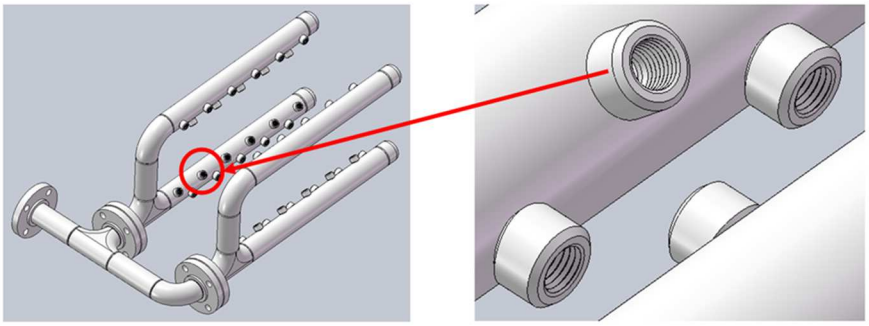
Implementation of analysis in the design process means that analysis results are used to make design decisions. It is, therefore, important that analysis tools provide results with predictable accuracy. Analysis results validation is a complex problem because all numerical methods are approximate and there can be many sources of errors including the major ones: modeling errors and approximation errors.

SimSolid eliminates major sources of errors associated with conventional FEA. It also introduces a new approach to adaptive solution refinement based on exact measures of error on surface of a structure.

### Sources of Modeling Errors in Traditional FEA

Modeling errors occur when the CAD geometry model is being modified to make it suitable for traditional FEA meshing. The modification can include many steps such as assembly simplification, part de-featuring, surface idealization and geometry face clean up.

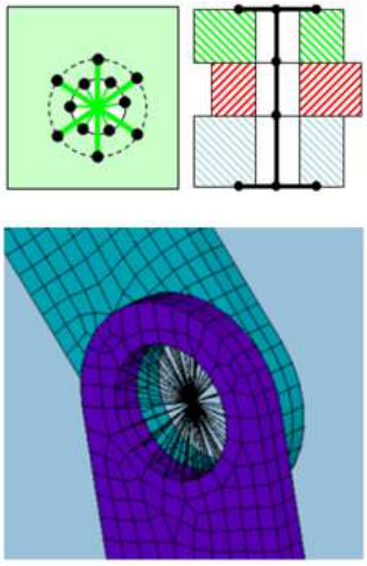
Successful meshing is pre-requisite to obtaining any results in FEA. Even if only global displacements are of interest, the geometry still has to be meshed to the smallest detail. Furthermore, meshing must use the correct element type, show correct element shape (no degeneration or bad aspect ratio) and have enough elements to model the expected stress pattern. These quality requirements are quite difficult to satisfy for complex parts. Adaptive re-meshing to satisfy numerical convergence is possible but not practical in many situations and is not commonly performed in design analysis of assemblies.



For assemblies the situation is getting even worse because meshes in contact areas of parts must be either compatible or good enough to provide meaningful results. The latter gets practically impossible in case of multi-scale assemblies when large parts are connected through small parts like bolts, nuts, rivets, pins, etc.

Assembly idealization is also dependent on the solution method. Pictured above is an example of small parts that can be simply removed in a static analysis but need to be replaced by mass points or artificial bodies with six inertia moments in a dynamic analysis in order not to change the mass distribution in the structure.

Other sources of error in traditional FEA include special element consideration for connections. Bolts and welds are problematic in that both special elements and special mesh patterns are required to model them adequately. Below are examples of connection idealizations of a bolt replaced by beam and spider rods.



The final stumbling block is solving. Even if model has been successfully meshed, solution is still not assured. Having meshed complex geometry, the model is often found to be too large to be solved within a reasonable time or contains poorly shaped elements that cause instabilities in the numeric of traditional FEA solver methods.

Using the traditional FEA workflow to manage these potential error sources is complex. Training, and retraining, can be expensive and time consuming. Occasional (infrequent) use of simulation is especially problematic. Errors introduced by mis-application of a user interface workflow are far too common.

**SimSolid Approach**

All the traditional complexities of geometry simplification and meshing are not present in SimSolid. The true geometry is processed directly. Assemblies can have parts with different scales (big/small or thick/thin). Connections between parts are more straightforward to apply. The time required to setup the simulation model is dramatically shorter and the reduction in the required process steps means there is much less possibility of user error.

Adaptive refinement is always activated during the solution process. Simple controls are available to increase the relative number of equations or to adjust the number of adaptive solution passes performed. This can be done on both a global (all parts) or part local basis. The degree of solution completeness can be easily managed by the user at a high level without the need to create and closely manage complex meshing patterns.

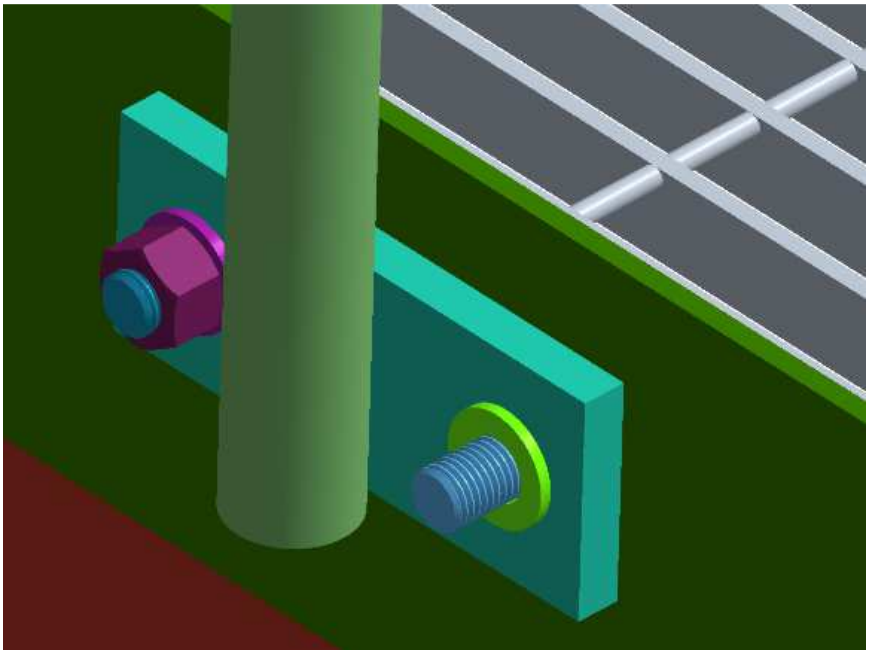
## Application Examples

Included below are several representative SimSolid solution examples.

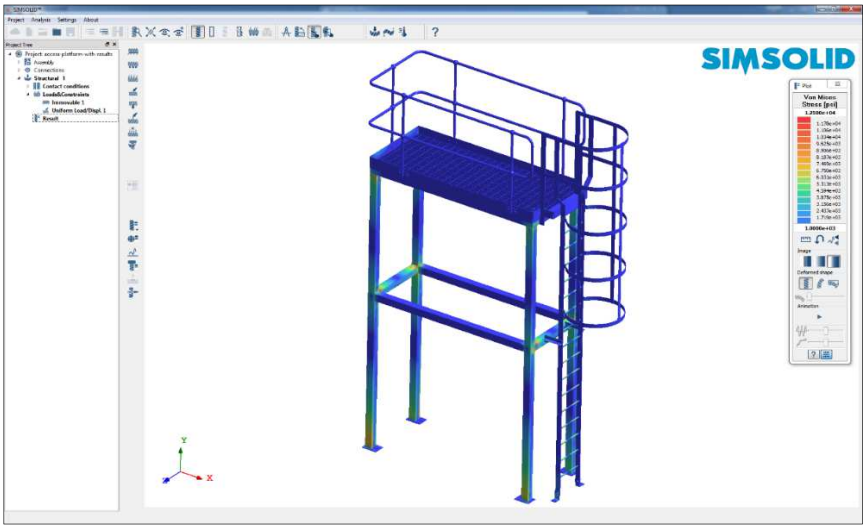
**Access Platform**

An assembly of 153 parts consisting of 10 small bolts, washers and nuts as well as 123 other parts including hollow tubular frames, solid ladder rungs and a single large complex floor grate was analyzed. 412 connections were automatically generated. This model illustrates the ability of SimSolid to efficiently analysis a large assembly with parts of varying size scales (small to large) and varying geometric complexity.

*Close-up detail of a bolt, nut, and washer geometry.*

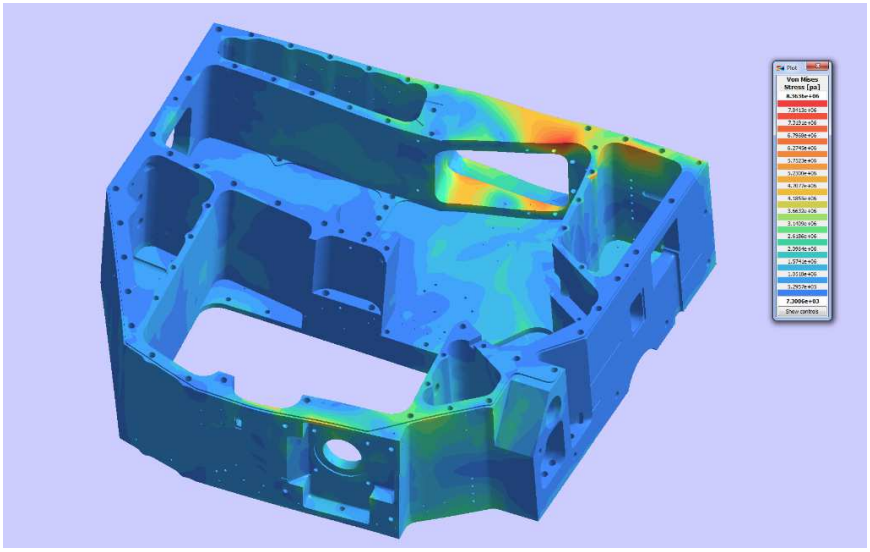


*SimSolid simulation solution.*



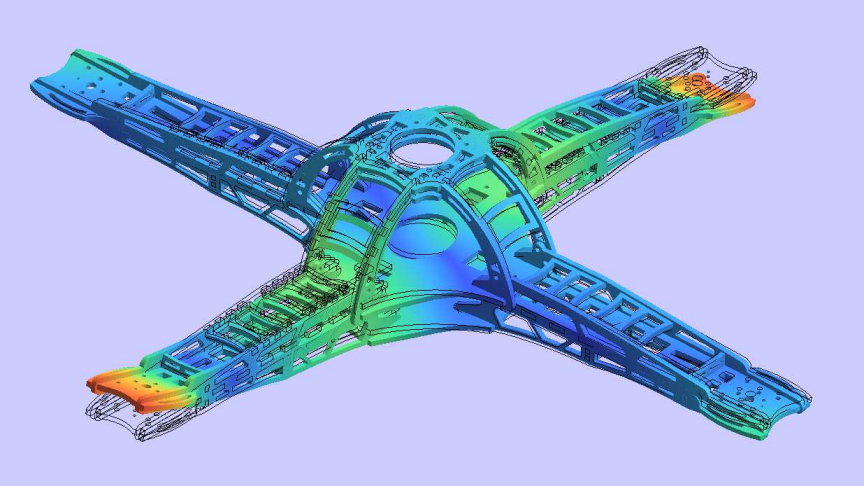

**Complex Machined Plate**

Static analysis of a complex machine plate consisting of 1,200+ faces and 150+ small holes. Typical solution time on an 8-core Intel i7 is under one minute.



**Modal Analysis of a Quad Copter Frame**

Quad copter frame consisting of 48 parts and 986 connections. Unconstrained modal analysis to calculate the first 10 flexible modes. Typical solution time under one minute.



## Conclusion

For the simulation to truly drive the design process it needs to work lock step with each geometry concept and concept modification. The complexity of traditional FEA eliminates its use in all but the most trivial of design conditions. Simulation working directly on design geometry provides a path to quick meaningful answers that guide designers and engineers to more optimal design scenarios.

Only SimSolid can provide this by not only eliminating time consuming and expertise extensive geometry simplification techniques such as defeaturing and mid-planning but by also eliminating the mesh all together. The result is a simulation tool that is both:

- Fast enough with respect to both model and solve time to be used every day;
- Simple enough to be used occasionally without the need for extensive training and monitoring.

Try SimSolid for yourself. We think you will agree that it is how design simulation should be done. For more information and to conduct a trial simulation of our product, please visit our website at www.altair.com/simsolid.

## References


[1] Cea, Jean. Approximation Variationelle des Problems aux Limites. Ann. Inst. Fourier, 1964, 14 (2), pp. 345-444.

[2] Aubin, Jean-Pierre. Approximation of Elliptic Boundary Value Problems. Wiley Interscience, 1972.

[3] Apanovitch, Victor. The Method of External Finite Element Approximations. Minsk, 1991.