C&R White Paper, Revision 2, May 15, 2000

Reliability Engineering and Robust Design: New Methods for Thermal/Fluid Engineering

Brent A. Cullimore
C&R Technologies, Inc.
Littleton Colorado, USA

ABSTRACT

Recent years have witnessed more improvement to the SINDA/FLUINT thermohydraulic analyzer than at any other time in its long history. These improvements have included not only expansions in analytic power, but also the additions of high-level modules that offer revolutions in thermal/ fluid engineering itself.

One such high-level module, “Reliability Engineering,” is described in this paper. Reliability Engineering means considering tolerances in design parameters, uncertainties in environments, uncertainties in application (e.g. usage scenarios), and variations in manufacturing as the stochastic phenomena that they are. Using this approach, the probability that a design will achieve its required performance (i.e., the reliability) is calculated, providing an assessment of risk or confidence in the design, and quantifying the amount of over- or under-design present.

The design to be evaluated for reliability will likely have been produced using traditional methods. Possibly, the design was generated using the Solver optimizer, another high-level module available in SINDA/FLUINT. Using design optimization, the user quantifies the goals that make one design better than another (mass, efficiency, etc.), and specifies the thresholds or requirements which render a given design viable or useless (exceeding a performance limit, etc.). SINDA/FLUINT then automatically searches for an optimal design.

Robust Design means factoring reliability into the development of the design itself: designing for a target reliability and thereby avoiding either costly over-design or dangerous under-design in the first place. Such an approach eliminates a deterministic stack-up of tolerances, worst-case scenarios, safety factors, and margins that have been the traditional approaches for treating uncertainties.

In any real system or product, heat transfer and fluid flow play a limited role: there are many other aspects to a successful design than the realm of thermal/fluids that is encompassed by SINDA/FLUINT. Therefore, this paper concludes with brief descriptions of methods for performing interdisciplinary design tasks.

INTRODUCTION: THE NEED FOR A NEW METHOD

Overdesign is common and expensive. In large scale projects, each discipline (thermal, structural, power, etc.) communicates worst-case requirements to other disciplines rather than attempting to couple the design analyses. This leads to designs that are heavier and more costly than they need to be, *and in some cases does not even result in a safer or more reliable design*.

For example, it is common for power specialists to require that nickel-hydrogen batteries never exceed 15°C. This creates a serious thermal control challenge, requiring additional structural mass, technology risk, and, ironically, heater power. In fact, nickel-hydrogen batteries do not fail at 15°C, they simply become less reliable and more likely to fail the longer they operate at elevated temperatures. Occasional exposure temperatures up to as high as 30°C are tolerable but undesirable, yet total avoidance of any temperature greater than 15°C during any mission phase becomes the task of the thermal control specialist. The thermal control specialist might even resort to fancier and therefore more risky thermal control options to achieve this requirement, resulting in a *less* reliable overall design than if temperature excursions had been better tolerated in the battery design requirements! Examples of such overdesign abound.

Even within one discipline, overdesign exists due to stack-up of margins and worst-case scenarios until the design case is unrealistic and will likely never occur. A worst-case (*unlikely*) spacecraft attitude is combined with end-of-life *expected* degradations of optical coatings, *estimations* of worst-case electronic dissipations, and *predictions* of worst-case conductive interface performance, etc. Additional margin is then added to cover *uncertainties* in thermal modeling, environment, and component sizing (11°C prediction margin plus either 10°C margin from qualification on passive designs or 25% control authority on active designs, per MIL-STD 1540c). Only when meeting an extreme stack-up of margins and uncertainties becomes impossible does a renegotiation of adequate margin begin, and such renegotiations are seldom based on any mathematical rigor or true knowledge of the underlying risk.

In the aerospace industry, which is heavily influenced by understandably cautious military standards, such overdesign compensates for unknowns and unforeseen problems. Success in such a design environment is a necessity, and cost is a secondary consideration.

In commercial satellites, on the other hand, cost is a primary consideration. An overall satellite reliability of 99% may be desired, but if significant savings result from a reduced reliability of 98%, the latter option will be seriously considered. For example, it is common to apply a 5°C uncertainty to thermal analysis predictions in a commercial environment versus an 11°C uncertainty dictated by MIL STD 1540c. “Safer” is also much more costly.

NASA’s “faster, better, cheaper” campaign in many ways represents a shift from a military perspective to a commercial one: additional risk may be *intelligently* traded against reduced mission cost.

Statistical variations and uncertainties are intrinsic to thermal/fluid designs. They occur in the form of:

1. Dimensional tolerances and property or performance uncertainties. Examples: interference fits, epoxy bond line thicknesses, as-built insulation performance, degradation of optical coatings, conductance across interfaces, convection coefficients, two-phase pressure drops.
2. Boundary conditions. Examples: weather, orbital environments, solar constants.
3. Requirements and design margin. Examples: battery dissipation levels, equipment failure (temperature control) limits, heat pipe excess capacity, heater margin.

Uncertainties abound in thermal design, and performance specifications (design requirements) are usually negotiable, meaning that they can be violated occasionally or under certain circumstances. As an alternative to stacking up worst-case margins, uncertainties, the engineer could combine these factors statistically to yield information about the degree of confidence (“reliability”) in a particular point design. In other words, the engineer could generate not just a single performance predictions but also a distribution of performance predictions with associated probabilities of occurrence, as shown graphically in Figure 1.

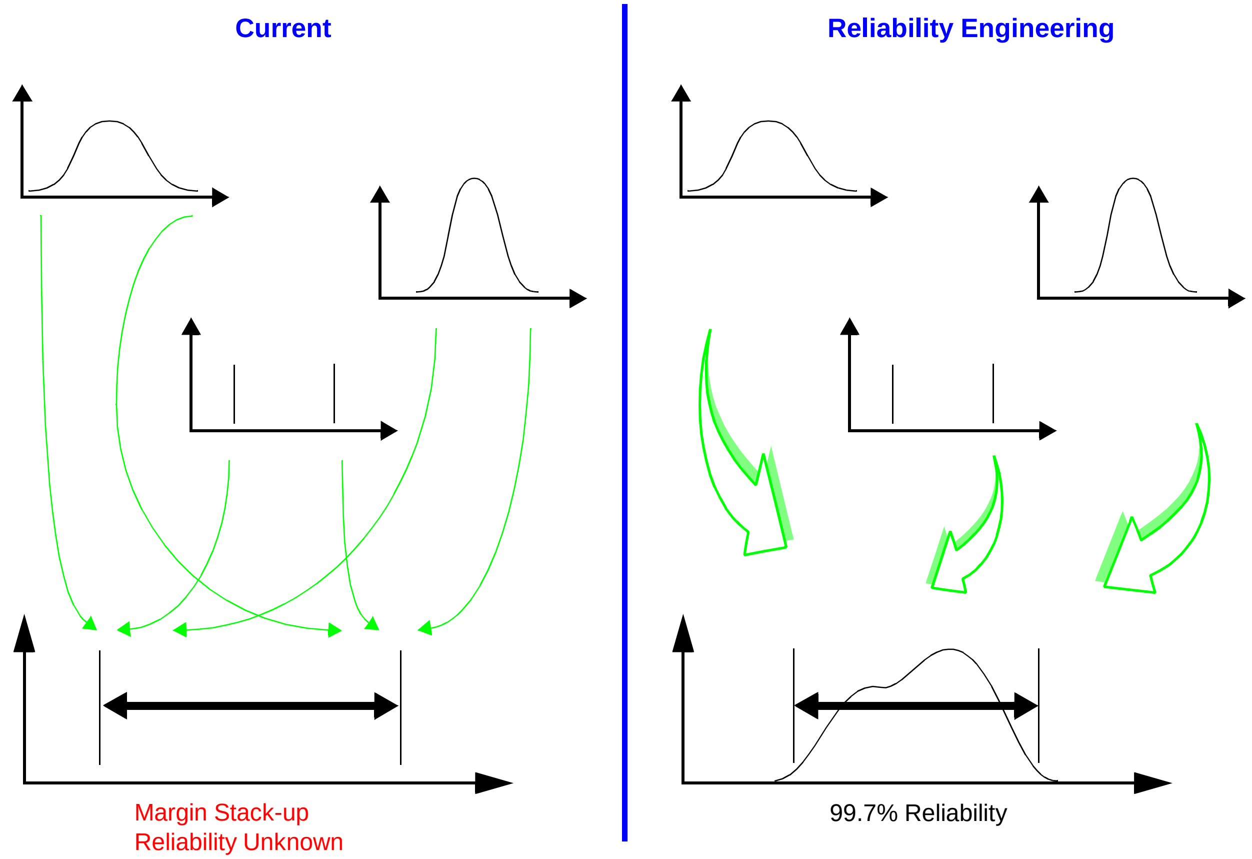


Figure 1: Avoiding Overdesign by Combining Uncertainties and Meeting Requirements Statistically

Consider an example. During the design of the space station single-phase ammonia coolant loop, the question arose of compliance with requirements given the uncertainty in the manufacture of flow control orifices. In other words, the baseline design included specific orifice sizes as needed to achieve a balance of flow rates between parallel legs such that no single payload would have less than the required flow rate (and hence be at risk of overheating). Even slight changes in the orifice dimension could result in uneven flow distributions, such that a worst-case stack-up of orifice sizes would definitely cause a lower or upper temperature control limit to be exceeded. Recognizing that such a problem should not be treated using a worst-case but rather a probability distribution, the confidence in the final design was determined quantitatively using statistical combinations of various orifice sizes. Unfortunately, since an older version of SINDA/FLUINT was employed which had no such statistical design features, considerable work was expended to perform the analysis.

Another space station example is the “design-to-freeze” radiator. Thawing ammonia ice can rupture fluid lines, and hence high strength materials and other design measures were used to overcome the problem. The number of expected fatigue cycles had to be treated statistically combining estimates of loads and environments over the life of the station. Also, the worst case design point for the thaw stress resulted from a stack-up of various uncertainties in radiator performance, environmental heating rates, etc. Because a worst-case stack-up resulted in an unrealistically harsh design case with no potential design solution, development and negotiation of a reasonable design case had to be performed to provide adequate confidence in the resulting design. The resulting design case was also used as the basis for the validation test program.

Although only two examples are provided above, opportunities for treating limits not as fixed “goal posts” but as probabilistic distributions abound in most engineering problems. Engineers are simply not accustomed to dealing with design problems in this manner in part because of training and in part because of lack of tools.

INTRODUCTION: SINDA/FLUINT

SINDA/FLUINT (Ref 1) is the NASA-standard heat transfer and fluid flow analyzer for thermal control systems. Because of its general formulation, it is also used in other aerospace specialties such as environmental control (ECLSS) and liquid propulsion, and in terrestrial industries such as electronics packaging, automotive, refrigeration, and power generation.

SINDA/FLUINT is used to design and simulate thermal/fluid systems that can be represented in networks corresponding to finite difference, finite element, and/or lumped parameter equations. In addition to conduction, convection, and radiation heat transfer, the program can model steady or unsteady single- and two-phase flow networks, including nonreacting mixtures and nonequilibrium phenomena.

SINDA

SINDA uses a thermal network approach, breaking a problem down into points at which energy is conserved (*nodes*), and into the paths (*conductors*) through which these points exchange energy via radiation and conduction. While often applied as a lumped-parameter modeling tool, the program can also be used to solve the finite difference (FDM) or finite element (FEM) equations for conduction in appropriately meshed shells or solids. One can employ finite difference, finite element, and arbitrary (lumped parameter) nodes all within the same model.

An important improvement over ancestral versions of SINDA is the inclusion of submodels, which enable analysts to subdivide a large network of nodes and conductors into collections of subnetworks consisting of nodes, conductors, or both. Submodels represent a convenient means of combining separately developed models, each with its own control variables, customization logic, solution method, and perhaps conflicting node and conductor numbering schemes. More often, they are simply used to improve the organization and legibility of the model, or to perform high-level simulation manipulations such as dynamically swapping sets of boundary conditions, evaluating alternate designs or components, or simulating variable configurations.

Solutions may be performed in single- or double-precision without any model or logic changes. Also, either iterative or simultaneous (optimally reordered sparse matrix) solutions may be used in steady-state or transient analyses. SINDA/FLUINT provides a powerful means for creating highly customized solution schemes by permitting the user to vary the underlying methods on a submodel-by-submodel basis.

FLUINT

To answer the need to model two-phase fluid systems and to replace the cumbersome and limited “one-way conductor” methods employed by ancestral versions of SINDA for fluid flow simulation, FLUINT development was initiated by NASA in the 1980’s as a major expansion of SINDA. All major development has been completed, providing unmatched thermohydraulic analysis capability. Thermal and fluid models may be used alone or together to solve conjugate heat transfer problems as typically found in thermal control, propulsion, and energy systems.

FLUINT introduced a new type of submodel composed of network elements*,* *lumps* and *paths*, which are analogous to traditional thermal nodes and conductors, but which are much more suited to fluid system modeling. Unlike thermal networks, fluid networks are able to simultaneously conserve mass and momentum as well as energy.

Lumps are subdivided into *tanks* (control volumes), *junctions* (volumeless conservation points, instantaneous control volumes), and *plena* (boundary states). Paths are subdivided into *tubes* (inertial ducts), or *connectors* (instantaneous flow passages including short ducts [*STUBE* connectors], valves, etc.).

In addition to lumps and paths, there are three additional fluid network elements: *ties*, *fties*, and *ifaces*. Ties represent heat transfer between the fluid and the wall (i.e., between FLUINT and SINDA). Fties or “fluid ties” represent heat transfer within the fluid itself. Ifaces or “interface elements” represent moving boundaries between adjacent control volumes.

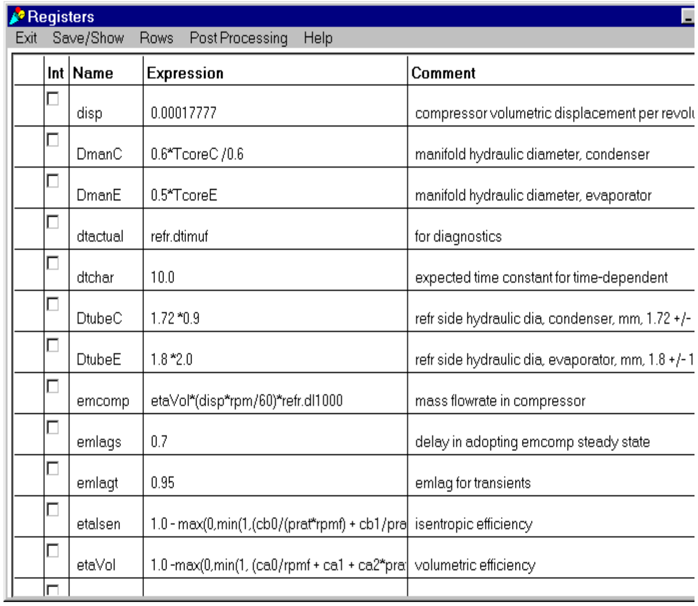
Paralleling SINDA while at the same time extending the SINDA design philosophy, FLUINT models can be constructed that employ fully transient thermohydraulic solutions (using tanks and tubes), or that perform pseudo-steady transient solutions (neglecting perhaps inertial effects and other mass and energy storage terms using junctions and STUBE connectors), or that employ both techniques at once. In other words, the engineer has the ability to approximate or idealize where possible, and to focus computational resources where necessary. Like SINDA, full access is provided in logic and in spreadsheet relationships not only to the basic modeling parameters (dimensions, properties, loss factors, etc.), but also to derived or abstract solution parameters (e.g., the exponent on flow rate of the friction coefficient), and to underlying correlations for heat transfer, pressure drop, etc.

Although the user can build models of custom parts and control systems, prepackaged tools are provided for modeling common components such as pipes, pumps, valves, filters, accumulators, etc. Table 1 presents the overall organization of SINDA/FLUINT modeling tools.

Single- or two-phase flow can be modeled either for pure components (e.g., steam and water), for nonvolatile/noncondensible mixtures (e.g., air and oil), and for condensible/volatile mixtures (e.g., air and oil and steam and water). Gases can dissolve into or evolve from the liquid phases according to saturation relationships and finite rate mass transfer. Up to 26 nonreacting substances can be mixed within each fluid submodel, and up to 25 fluid submodels can be used.

Two-phase flow is by default homogeneous (uniform velocity: equal liquid and gas velocities) and in phasic equilibrium (perfectly mixed: equal temperatures and pressures between phases). However, it is a simple matter to elect the prediction of flow regimes, to model slip flow (unequal liquid and gas velocities), to model phasic nonequilibrium in quasi-stagnant volumes and within duct flows, and to model nonequilibrium expansions in valves, orifices, and venturis.

Unique features such as time- and direction-varying body forces and capillary device models are important to the aerospace industry. Because they are unique, such tools have found uses in nonaerospace applications such as modeling rotating machinery.



Int	Name	Expression	Comment
□	disp	0.00017777	compressor volumetric displacement per revolu
□	DmanC	0.6*TcoreC /0.6	manifold hydraulic diameter, condenser
□	DmanE	0.5*TcoreE	manifold hydraulic diameter, evaporator
□	dtactual	refr.dtimuf	for diagnostics
□	dtchar	10.0	expected time constant for time-dependent
□	DtubeC	1.72 *0.9	refr side hydraulic dia, condenser, mm, 1.72 +/-
□	DtubeE	1.8 *2.0	refr side hydraulic dia, evaporator, mm, 1.8 +/- 1
□	emcomp	etaVol*(disp*rpm/60)*refr.dl1000	mass flowrate in compressor
□	emlags	0.7	delay in adopting emcomp steady state
□	emlagt	0.95	emlag for transients
□	etaIsen	1.0 - max(0,min(1,(cb0/(prat*rpmf) + cb1/pra	isentropic efficiency
□	etaVol	1.0 -max(0,min(1, (ca0/rpmf + ca1 + ca2*pra	volumetric efficiency
□			

Figure 2: Part of the Built-in Spreadsheet: User-defined Registers

BUILT-IN SPREADSHEET

A built-in spreadsheet enables the user to define custom (and perhaps interrelated) variables (Figure 2) call *registers*. The user can also define complex self-resolving interrelationships between inputs, and also between inputs and outputs. This spreadsheet allows rapid and consistent model changes, minimizes the need for user logic, and makes parametric and sensitivity studies trivially easy to perform.

The ability to create a SINDA/FLUINT model whose network parameters and logic are completely controlled by a few centralized registers enables high-level modules to be added. One of these high-level modules is the focus of this paper, but to fully explain it, another high-level module must first be introduced.

THE SOLVER

The Solver was the first top-level design module in SINDA/FLUINT. It was released in 1997 as part of Version 4.0.

The Solver is a fully featured nonlinear programming system that can be used for a variety of purposes:

1. Goal Seeking: the ability to solve for an input value given a desired response (output value). When used in this mode, the Solver eliminates the need to write iteration logic. For example, the user might wish to know what coolant pump flow rate results in an electronics temperature of 20°C. Or, the user may wish to find the conductivity of a plate or fin required to achieve a heat rejection efficiency of 95%.
2. Optimization (design synthesis): the ability to use SINDA/FLUINT to help size or select design parameters. The user defines which parameters are to be sized or

selected along with an objective (“ What makes one design better than another?”) and possibly some constraints (“What limits render a particular design viable or useless?”).

3. Test Correlation (calibration): the ability to adjust the model (not the design) until best-estimate values for uncertain parameters are generated. The user defines which parameters are uncertain, and provides test data to match against. Many correlation methods are available along with various data handling and comparison utilities: automated test data correlation is currently the primary use of the Solver module.

In all of the above cases, the user defines an *evaluation procedure*, or an arbitrarily complex series of SINDA/FLUINT solutions that tell the Solver how a particular design (for optimization) or model (for correlation) stacks up against the goals and requirements. Frequently, this procedure is no more complicated than a single steady state solution, but it can use any solutions or utilities available in SINDA/FLUINT to perform the task. In essence, using the Solver is like tasking a traditional SINDA/FLUINT model to run itself repeatedly until it achieves some user-defined objective (Figure 3).

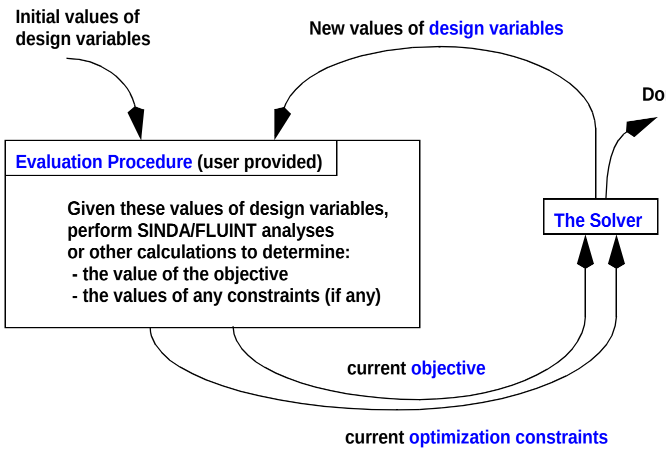

Initial values of design variables
New values of design variables
Done
Evaluation Procedure (user provided)
Given these values of design variables, perform SINDA/FLUINT analyses or other calculations to determine:
- the value of the objective
- the values of any constraints (if any)
The Solver
current objective
current optimization constraints

Figure 3: “The Solver:” Optimization and Test Data Correlation Module

Further description on the Solver is available in Reference 2. Knowledge of this module is a prerequisite for the subsequent discussion on Robust Design. However, a few key points need to be made before leaving this topic.

Without the high-level modules, SINDA/FLUINT is used in a traditional point-design fashion: given a specific and deterministic design and a fixed environment and usage scenario, steady-state and/or transient simulations are run to determine how the design performed. This method is *not* a natural way of performing common engineering tasks. Rather, it is readily available because it is what is “easily” achieved using numerical solutions. Because this type of software is all that has been available, a generation of engineers has been trained in these point-design evaluation methods, forgetting perhaps what the original intent of using them was: to produce good designs, and not just to evaluate point designs.

The Solver module offers a revolution in SINDA/FLUINT usage because it represents an automation of the design process itself, and not an automation of a subprocess: point-design evaluation. Reliability Engineering offers a similar revolution because it permits many point-designs to be evaluated at a higher level. Combining the Solver and Reliability Engineering yields Robust Design: factoring reliability into the automated process of design synthesis itself, and thereby producing a design quantitatively balances risk and cost.

ACCESSIBILITY

Concurrent developments have made advanced design features in SINDA/FLUINT more accessible. C&R’s Sinaps*Plus*® is a complete nongeometric (circuit sketchpad) pre- and postprocessor for SINDA/FLUINT. C&R’s Thermal Desktop® (with the optional RadCAD® radiation analyzer) is a geometric (CAD/FEM/FDM) interface that brings traditional thermal modeling practices into a concurrent engineering environment. A freely distributed plotting program is also available: EZ-XY™.

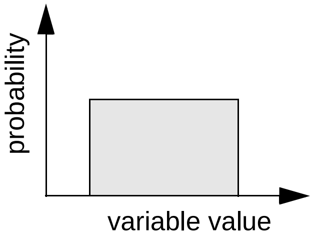
RANDOM VARIABLES AND THEIR DISTRIBUTIONS

To use the Reliability Engineering module in SINDA/FLUINT, the user starts by identifying which parameters (dimensions, properties, boundary conditions, etc.) are uncertain. These *random variables* will be allowed to vary over a prescribed range, and any one value of such a random variable has a given probability of occurrence, at least in comparison to other values. This variation is called a *probability distribution*.

Once a parametric model is built using registers, a subset of these variables are identified as random. The user must then describe the distribution function of each random variable using one of three methods described next.

UNIFORM DISTRIBUTIONS

The simplest type of distribution is a uniform one: the random variable may assume any value with equal probability between a lower limit and an upper limit, as shown at the right.



This is an important class of distributions because it represents an easy transition from the current margin-based approach of worst-case high and low values. The margin-based approach to handling uncertainty is excessively conservative, corresponding to two delta (spike) distribution functions at the upper and lower limits, whereas the uni-

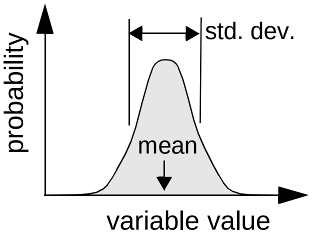
Table 1: SINDA/FLUINT Hierachy of Modeling Options

- Thermal/Fluid Models
    - Registers, Expressions, and Spreadsheet Relationships
    - Concurrently Executed User Logic
    - Thermal Submodels
        - Nodes
            - Diffusion (finite capacitance)
                - Temperature-varying
                - Time-varying
            - Arithmetic (massless: instantaneous)
            - Boundary (constant temp.)
            - Heater (constant temp., returns power)
        - Conductors
            - Linear (conduction, advection)
                - Temperature-varying
                - Time-varying
            - Radiation
                - Temperature-varying
                - Time-varying
        - Sources
            - Temperature-varying
            - Time-varying
    - Fluid Submodels
        - Lumps
            - Tanks (finite volume)
                - Twinned tanks (nonequilibrium modeling)
            - Junctions (zero volume: instantaneous)
            - Plena (constant temperature, pressure)
        - Paths
            - Tubes (finite inertia)
                - twinned tubes (slip flow)
            - Connectors (zero inertia: instantaneous)
                - short tubes (STUBEs)
                    - twinned STUBEs (slip flow)
                - valves
                    - check valves, control valves
                    - pressure regulating valves
                - K-factor losses, bidirectional or not
                - pumps, fixed or variable speed
                - constant mass or volumetric flow rate
                - capillary elements (CAPILs)
        - Ties (heat transfer)
            - user-input conductance
            - program-calculation (convection) conductance
        - Duct macros (subdivided pipelines)
        - Capillary evaporator-pumps (CAPPMP macros)
        - Ifaces (control volume interfaces), with or without inertia
            - flat (zero pressure difference)
            - offset (finite pressure difference)
            - spring (i.e., bellows, etc.)
            - spherical bubble
            - wick (liquid-vapor interface in porous structure)
        - Fties (fluid-to-fluid ties)
            - axial in a duct
            - user-input conductance
            - constant heat rate
        - Auxiliary Utilities
            - choked flow detection and modeling
            - waterhammer and acoustic wave modeling
            - compressors
- Solutions
    - Steady-state
    - Transient
    - Goal Seeking
    - Design Optimization
    - Test Data Correlation
    - Reliability Estimation
    - Robust Design

form distribution acknowledges that values in between are at least as likely to occur as the extremes. Unlike the margin-based approach to uncertainty, the Reliability Engineering approach makes no presumptions about which *combinations* of upper and lower limits yield problematic performance. Nonetheless, the uniform distribution is very simplistic: in most distributions values near the extremes are much less likely to occur than values near the middle.

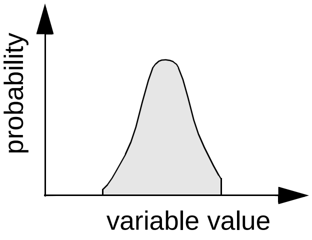
NORMAL DISTRIBUTIONS

The most common type of nontrivial distribution is the normal or Gaussian distribution. It is a symmetric distribution that can be completely described by a mean value and a standard deviation. Many times, an engineer will know the nominal value of a parameter along with an upper and/or lower limit. Frequently these upper and lower limits correspond to a known number of standard deviations (usually about three) off the mean.

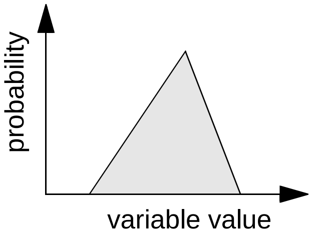


ARBITRARY DISTRIBUTIONS

Sometimes, a normal (Gaussian) distribution is appropriate, but a theoretical range between negative and positive infinity is nonphysical or would cause numerical problems: a truncated normal distribution is required (shown at right



Another possibility is a triangular (witch’s hat) distribution, useful when all that is known is a most likely value plus a lower and upper bound (shown at right).



In fact, there are many types of distributions available (e.g., log normal, Weibull, Chi-square, etc.), each suited for a different purpose. It is also possible that a distribution function is produced from test or manufacturing data or from a previous analysis.

To support any such distribution, SINDA/FLUINT accepts a user-supplied table (array) of value versus probability. Any number of points can be used to define the distribution function. SINDA/FLUINT itself can be used to generate the function for use in a future run using Fortran-style calculations.

RELIABILITY CONSTRAINTS (FAILURE LIMITS)

“Reliability” is the probability that a design will not exceed limits defining failure. For example, a design might be considered a failure if a critical component exceeded an upper or lower bound on a temperature, if a heater switched on and off excessively, if a pressure exceeded 25% of the burst pressure, etc. There may be many such failure limits.

A list of responses of interest to the designer (e.g., the temperature of the critical component) can be created as well as upper and/or lower limits on those responses (the failure limits). Collectively, these are referred to as *reliability constraints*. One such reliability constraint might appear as follows:

```
Tmin <= battery.T100 <= Tmax
```

meaning that a failure will be assumed to exist if the temperature of node 100 in submodel “battery” goes below Tmin or above Tmax.

While the program must know what responses are desired and what the limits are on those responses in order to calculate reliability, such foresight is helpful but strictly not required. A user might forget to define any responses, or may indicate a response of interest without applying any limits to it.

In other words, the user might decide after having made a run to impose a new limit, or to investigate a new response. Such hindsight is afforded by expansions to postprocessing tools such as EZ-XY.

RELIABILITY ESTIMATION METHODS

SINDA/FLUINT offers three very different statistical analysis routines. These routines all perturb random variables according to their specified distributions, execute the evaluation procedure provided by the user (perhaps just a single steady state solution), and monitor reliability constraints (if any) to produce statistics regarding those responses, including the probability of a successful design. Figure 4 indicates this top-level data flow for the Reliability Engineering module.

However, the methods used by each of the three routines are intentionally very different, providing the user with a wide range of options. These statistical analysis routines are described next. Table 3 is a summary of the options available.

MONTE CARLO SAMPLING

The simplest approach is that taken by the SAMPLE routine: a Monte Carlo method in which values of random variables are selected randomly according to their probability distribution functions. As an example, for a uniform distribution any value within the valid range is selected using a uniform random number generator. For normal distributions, random values are selected, but values near the center (the mean) will be generated more frequently than those at the extremes.

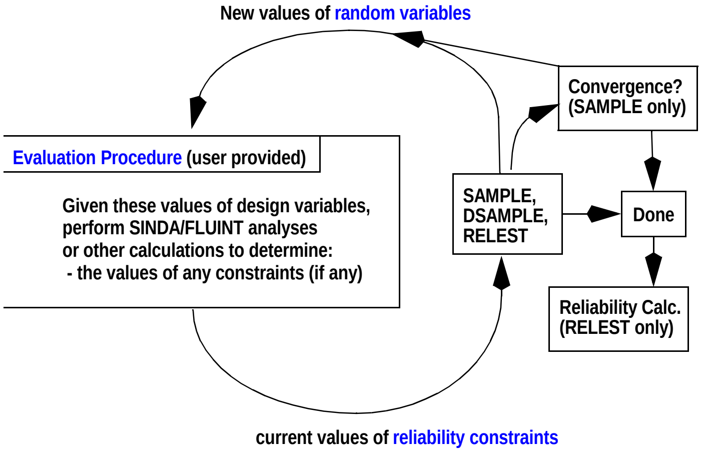

New values of random variables
Convergence? (SAMPLE only)
Evaluation Procedure (user provided)
Given these values of design variables, perform SINDA/FLUINT analyses or other calculations to determine:
- the values of any constraints (if any)
SAMPLE, DSAMPLE, RELEST
Done
Reliability Calc. (RELEST only)
current values of reliability constraints

Figure 4: Flow Chart for Reliability Prediction Methods

Table 3: Comparison of Reliability Estimation Routines

Routine	**SAMPLE**	**DSAMPLE**	**RELEST**
Method	Monte Carlo sampling	Descriptive sampling	Gradient method
Speed	Slow	Intermediate	Fast
Convergence Detected?	Yes	No	No
Fixed Execution Cost?	No	Yes	Yes
Overall Reliability?	Yes	Yes	No
Cumulative?	Yes	Somewhat	No
Applicability?	Unlimited	Unlimited	Limited. Assumes: - Gaussian variables - Linear responses - Continuous responses - Fixed failure limits

The Monte Carlo approach requires many samples (on the order of 1000: 100 to 10,000) and is therefore expensive. However, it yields the most information. Furthermore, the accuracy of the estimation can be controlled at least relatively if not absolutely: the SAMPLE routine detects convergence as defined by negligible change in the selected responses and their associated limits (i.e., the reliability constraints) between any two consecutive samples.

Monte Carlo Sampling provides two methods of predicting reliability. The first is a simple tally of the number of times a failure limit was not exceeded divided by the total number of samples. A similar method is used to predict overall reliability:* the percent of all sampled cases that did not exceed any limits. (In the limit of a single constraint with only an upper or only a lower limit, the overall reliability is the same as the reliability for that constraint.)

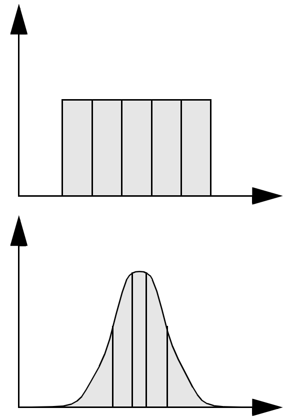
* This method only works if all the reliability constraints are independent (in series).

A second method is to accumulate statistics (mean and standard deviation) about every indicated response, and then to assume a normal (Gaussian) distribution for that response. The probability of exceeding any limit can then be calculated using the assumed profile.

DESCRIPTIVE SAMPLING

A faster alternative to Monte Carlo sampling is descriptive sampling, which is used in the DSAMPLE routine. This approach has a known cost: the user specifies the number of samples to be made (based on what they can afford). This number becomes the resolution with which the distributions in the random variables are subdivided.

For example, if 100 samples are to be used, each input profile will be divided into 100 regions of equal probability. For uniform distributions, one hundred equal regions will be used. For normal distributions, the region near the mean will be more finely subdivided than the extremes such that each region is equally probably and therefore contains the same area (integral of probability over the random variable values: the *cumulative distribution function*). This subdivision is illustrated at the right using five subdivisions.



Once the distributions of the random variables have been subdivided, only one value from each subdivision (the center of the corresponding region in the cumulative distribution function) is sampled, since each of these values is as probable as any of the others. There is still randomness involved for more than one random variable: each cell, while sampled only once, is selected at random. For example, the 5th cell of variable #1 might be combined with the 88th cell of variable #2 in one run, but the 5th cell of variable #1 might be combined with the 42nd cell of variable #2 in a second run.

For the same number of samples, descriptive sampling yields more accurate results than Monte Carlo sampling. Typically, descriptive sampling takes only 10 to 20% as many samples as does the Monte Carlo method does to achieve the same accuracy. However, Monte Carlo sampling retains certain advantages, the most important of which is a measure of confidence that enough samples have been taken for the given problem. In other words, there is no convergence test possible in descriptive sampling. Furthermore, Monte Carlo sampling is more readily cumulative (repeated runs can be combined for more accuracy than can repeated runs of descriptive sampling), and it can yield a more accurate prediction of the overall reliability than can descriptive sampling.

GRADIENT METHOD

A method for estimating reliability is available that is even faster than DSAMPLE, but has even more limitations: RELEST. This technique is not a sampling technique at all. Rather, it estimates reliability by measuring gradients in the responses with respect to the random variables, and by assuming (but not requiring) that all distributions (both input and response) are normal (Gaussian). It further assumes that the mean of the responses can be predicted using the mean values of the random variables, and that response variations from that point are linear with respect to changes in inputs.

RELEST requires only N+1 evaluations, where N is the number of random variables. This is often an order of magnitude smaller than what DSAMPLE requires, which is itself often an order of magnitude smaller than will SAMPLE requires: *RELEST is comparatively cheap.*

The first evaluation uses the mean values of random variables, and assumes that the resulting responses are the means of those functions. The next (and final) N evaluations perturb each random variable (in input order) such that the gradients of each response with respect to each input variable can be estimated using finite differences. RELEST then assumes a first order Taylor series of variance (the square of standard deviation) can be applied to estimate the variance (and therefore standard deviation) of each response given the variance of each random variable, whether those variables are normal or not. Now the code has enough information to predict reliabilities: it has an estimate for the mean and standard deviation of each response, and can therefore predict the likelihood that a response will assume any given value.

RELEST cannot predict overall reliability much less the tallied estimate of reliability that a sampling routine can, and should be used with caution in cases with nonlinear responses and non-normal random variables. It also cannot handle variable failure limits. Furthermore, unlike sampling techniques, the accuracy of RELEST is not cumulative: repeated calls do not affect the accuracy of the results. However, because it is so inexpensive, RELEST is often plays an important role in Robust Resign (described later).

DATABASE AND POSTPROCESSING

In important part of the Reliability Engineering module is the database that can be created to store the samples or gradient perturbations made in the previously described routines.

One purpose of creating such a database is to be able to accumulate results in subsequent runs. For example, it may be desired to add 1000 more Monte Carlo samples to the samples taken in a previous run, in order to add to the accuracy of the predictions.

A second purpose of creating the databases is to be able to visualize the resulting response distributions by plotting histograms, such as the two EZ-XY histograms displayed in Figure 5. The user can also produce scatter plots to see how any two parameters are related to each other.

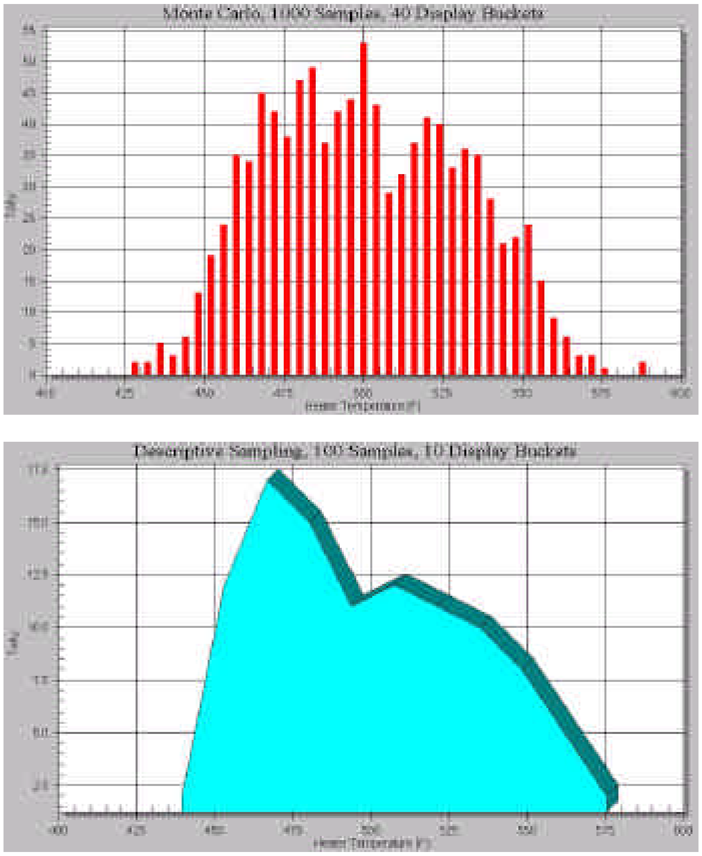


Figure 5: Sample Postprocessing: EX-ZY Histograms

However, the most important use of the database is to be able to apply hindsight while postprocessing: to be able to define new responses of interest, or new limits to previously defined responses. Generating the samples can be an expensive proposition when using sampling methods, and so storing a database is very important in case failure thresholds change or are redefined, or simply if the user forgot to define a reliability constraint in the first place.

A BRIEF EXAMPLE

Consider a metal bar that is heated on one end and which radiates to deep space on the other end, and is otherwise insulated. The length and thickness of the bar are known, as are the material properties. However, the width of the bar, the power applied, and the emissivity of the exposed (radiating) surface are less certain. The emissivity can assume any value from 0.08 to 0.12. The width of the bar is nominally 1 inch, and is expected to have a Gaussian distribution with a standard deviation of 0.01 inch. Similarly, the input power is nominally 10W but has a Gaussian distribution with a standard deviation of 0.5W.

What are the chances that the temperature of the heated side of the bar will not exceed 500°F under steady conditions?

A one-dimensional SINDA model of the bar is built using registers to define key dimensions and properties. Three of these registers are defined as random variables: WIDE, POWER, and EMIS corresponding to the above three uncertain terms. The definition of these registers, their identification as random variables, and the specification of their distributions is as follows:

```
HEADER REGISTER DATA
      ...
      EMIS              = 0.1
      WIDE              = 1.0
      POWER             = 10.0
HEADER RANDOM DATA
      EMIS, UNIFORM,      0.08, 0.12
      WIDE, NORMAL,       SD = 0.01
      POWER, NORMAL,      SD = 0.5
```

The heated side of the bar corresponds to node #1 in submodel “sub1,” and therefore the reliability constraint is simply defined as:

```
HEADER RELCONSTRAINT DATA
      SUB1.T1 <= 500.0
```

The evaluation procedure is simply a steady state solution:

```
HEADER RELPROCEDURE
      CALL STEADY
```

Now one of more of the reliability routines (SAMPLE, DSAMPLE, RELEST) can be called from the main solution block of SINDA/FLUINT (called “OPERATIONS”), along with calls for output and/or database write operations. The following calls for descriptive sampling (100 samples by default) plus tabulated output of the predicted reliability:

```
      CALL DSAMPLE
      CALL RCSTTAB
```

Details of the SINDA model are omitted for brevity, but the above sample illustrates how easily Reliability Engineering can be applied to an existing model that uses registers. Older models not originally built using registers and expressions can be easily retrofitted, adding multiplying factors that are initially equal to unity.

In the above case, due to the presence of a non-normal random variable and the highly non-linear behavior of this radiation dominated problem, the RELEST routine can only be used as a first approximation. Such a fast but approximate calculation is ideal if reliability is estimated as a part of the evaluation procedure for a design optimization, as described next.

ROBUST DESIGN

Assume that a thermal control system is being designed for a component whose temperature cannot exceed 40°C. Traditionally, the user would iteratively develop such a design, and then stack up worst case conditions to assure that the temperature would never exceed some lower threshold (perhaps 30°C) allowing for safety factors or margin, which hopefully have some basis in experience if not test data.

If the degree of uncertainty in the inputs can be quantified, then the probability of exceeding 30°C or 40°C could be determined using the Reliability Engineering module described above.

Perhaps the Solver optimization module could be used to find a deterministic (nonrandom) design that will just meet the 30°C threshold. Any variation in parameters will then result in a reliability of roughly 50% relative to 30°C, with a higher probability of not exceeding 40°C. If the chances of exceeding 40°C are too great, the design must be regenerated using a greater safety margin: applying perhaps a 25°C limit during the redesign process (whether manual or automated). In other words, even with automated design synthesis using the Solver optimization module, *the margin is itself unknown and must be estimated iteratively*.

Robust Design means being able to factor the ultimate reliability into the design process: using reliability as a basis for synthesizing the design in the first place, and avoiding high-level design iterations.

The Reliability Engineering module described in this paper enables a user to estimate the reliability of a point design based on uncertainties in the dimensions, properties, boundary conditions, etc. The Solver optimization module enables a user to size or select dimensions, properties, etc. such that mass is minimized, or such that performance is maximized, etc. This section lists ways in which these two modules can be combined to yield even more powerful design tools.

Listed below are a few possible combinations of these modules:

1. a design can be selected using the Solver, and then (in the same or later run) the reliability of that design can be estimated
2. the reliability of a design can be used as an objective (“maximize reliability” or “minimize the chances of failure”)
3. the reliability of a design can be used as an optimization constraint (“find the minimum mass design that achieves a reliability of at least 99%”)
4. the range or variance of a random variable can be used as a design variable (“what variation can be tolerated: how tight must tolerances be?”)

In the first case, the Solver and Reliability Engineering modules are not combined so much as executed in series. Often, the random variable is expressed as the uncertainty in a parameter rather than the parameter itself. For example, a pipe diameter might be defined as a mean value plus a random value (whose mean is zero):

```
    DH    = Dmean + Drandom
```

The mean diameter (*Dmean*) might be selected using the optimizer (with *Drandom* equal to zero), and then the reliability of the design might be evaluated about that mean using *Drandom* as a random variable.

However, the real power of Robust Design is reflected by the second, third, and fourth cases listed above: reliability-based optimization to replace a margin or safety factor approach.

EXAMPLE: TRADITIONAL APPROACH

Assume a computer chip fails when the semiconductor junction temperature exceeds 125°C: its *qualification* temperature. During acceptance testing of any particular unit, the junction temperature is stressed to 115°C, and it is therefore intended that this temperature (115°C) should never be exceeded during the life of the electronics: a 10°C margin exists as a minimum.

During the product design the junction temperature is not allowed to exceed 104°C, adding another 11°C of margin (using U.S. military standard MIL-STD-1540c for passive thermal designs as an example) to cover uncertainties in inputs (performance, environments) as well as uncertainties or inaccuracies in the model itself.

Worst case stack-ups are produced of hot and cold cases (environments, dissipations, etc.), beginning-of life (undegraded) properties versus end-of-life (degraded) properties, etc. and the designs are adjusted until the predictions show 21°C margin from the upper and lower bounds of qualification temperatures, and 11°C margin from the acceptance temperatures.*

The margins are shown graphically at the top of Figure 6 for the upper end of the temperature limits.

EXPANDING THE TRADITIONAL APPROACH

Optimization and Reliability Engineering can be used to enhance the current design process.

* This description oversimplifies for clarity. Generally, an even greater uncertainty margin (17°C) is recommended during preliminary design, and 11°C is applied to a model that has been calibrated (perhaps using the Solver module) to test data to within about 3°C.

Most designs are produced iteratively and manually. The Solver optimization module can be tasked to synthesize a design automatically or at least semi-automatically. In the above example, this would be performed by applying the limits as *optimization constraint* (similar to but independent from reliability constraints):

```
    Tjunc <= 125.0-10.0-11.0
```

Whether the design has been produced manually or the Solver has been used, the reliability of the design can still be estimated using the Reliability Engineering module. In this case, a *reliability constraint* of

```
    Tjunc <= 125.0-10.0
```

is applied as a failure limit. In other words, the reliability is defined as the chances of not exceeding the acceptance temperatures. In essence, *the validity of the 11°C margin (which was used to generate the design) is being tested*, as shown in Figure 6. The 11°C margin will either be too cautious, resulting in costly over-design, or will be inadequate, resulting in risky under-design.

The amount of over- or under-design can only be quantitatively measured using reliability estimation methods. Either way, a truly optimal design will achieve exactly the required reliability for the thermal subsystem and thus be neither over- nor under-designed. Any excesses in either direction are justification for revisiting the design itself.

REPLACING THE TRADITIONAL APPROACH

Revisiting a design is costly: it would have been far better to have achieved the target reliability in the first place using Robust Design methods.

To use Robust Design methods, the reliability constraint is still applied

```
    Tjunc <= 125.0-10.0
```

but the optimization constraint is replaced by:

0.997 <= RelAct

where “0.997” is the required thermal subsystem reliability, and “*RelAct*” is the actual reliability predicted for the current design using the Reliability Engineering module. In other words, *reliability estimation becomes part of the design evaluation process.*

As was noted above, meeting a reliability requirement is but one possible option. Other options include maximizing reliability (making *RelAct* the objective) while meeting some other mass or power budget. Also, presuming the engineer had some control on tolerancing (machining, subassembly acceptance criteria, etc.), Robust Design can also be used to calculate what range of uncertainties is acceptable.

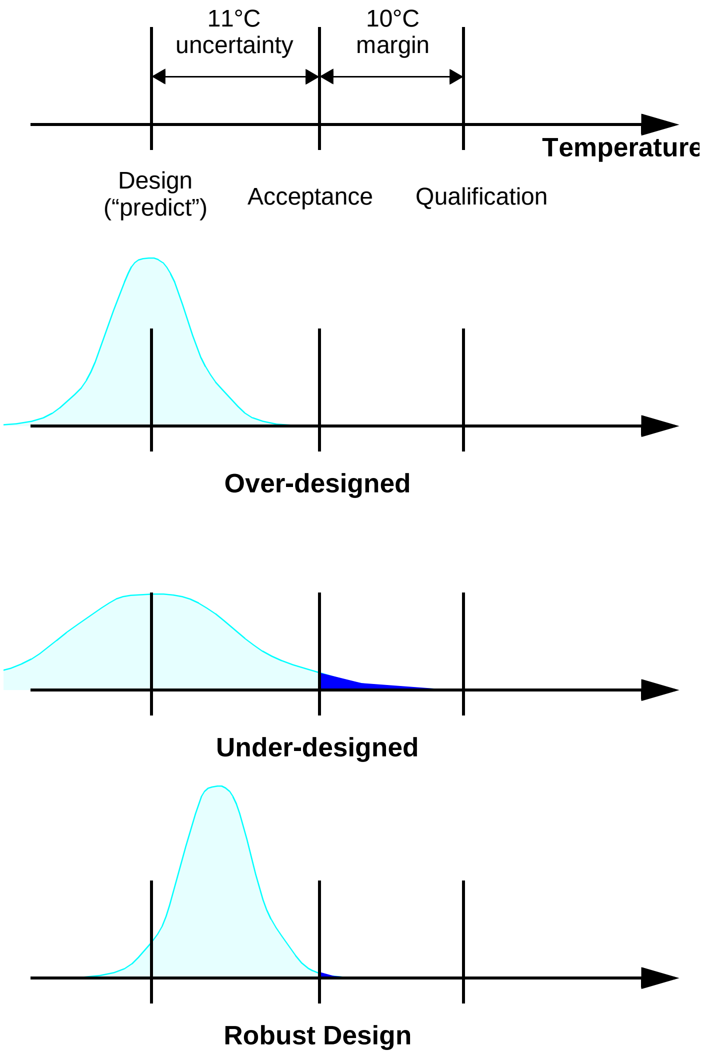


Figure 6: Traditional vs. Robust Design

MULTIDISCIPLINARY DESIGN GENERATION AND EVALUATION

Extending the previous example, note that even the 125°C limit levied upon the thermal designer is itself uncertain: it contains margins and/or a hidden reliability predictions. A truly optimal multidisciplinary design would factor in the reliability of the chip directly, rather than indirectly as an inflexible limit imposed upon the thermal designer. Even the final 10°C margin would be subject to replacement by statistical methods.

Commercial tools exist such as Engineous’ iSIGHT® (www.engineous.com) that can perform optimization, reliability estimation, and robust design generation at a higher level than what can be accomplished within a thermal/fluid analyzer such as SINDA/FLUINT. Codes such as iSIGHT enable the inclusion of almost any point-design simulation tool within any arbitrarily complex design evaluation process. SINDA/FLUINT is being expanded to provide direct links to iSIGHT to encourage such high-level integration.

RELEVANT THERMAL DESKTOP EXPANSIONS

C&R's Thermal Desktop® has been expanded to be parametric, allowing geometry, orbits, optical and material properties, etc. to be defined using expressions and symbols (analogous to SINDA/FLUINT registers).

More importantly, a direct link is being established between SINDA/FLUINT and Thermal Desktop: *Thermal Desktop calculations can be invoked dynamically from within SINDA/FLUINT during processor executions.* This provides the ability to include variations in radiation and geometric conductance/capacitance results while using the optimization, correlation, and reliability engineering modules. For this reason, interfaces to these modules are currently being added to Thermal Desktop. *The traditional separation of thermal math models (TMM) and geometric math models (GMM) is being eliminated.*

CONCLUSIONS

The ability to determine the amount of over- or under-design present in a thermal/fluid system has been added to SINDA/FLUINT, permitting uncertainties to be treated statistically in addition to traditional deterministic methods. More importantly, the potential to eliminate over-design due to stack-ups of margins, safety factors, and tolerances has been added, taking into account uncertainties early in the design process by designing for reliability.

Good software automates existing processes, reducing the effort required to create new products. Great software revolutionizes the processes, empowering the creation of better products. The addition of the Reliability Engineering module to SINDA/FLUINT, especially combined with previously existing modules such as optimization, attempts to assure SINDA/FLUINT's place in the latter category.

ACKNOWLEDGMENTS

SINDA/FLUINT would not exist were it not for the continuing support of the Crew and Thermal Systems Division of the NASA Johnson Space Center. Dr. Eugene Ungar was the technical advisor for the Reliability Engineering module, among other improvements.

CONTACT

C&R Technologies, Inc.
9 Red Fox Lane
Littleton CO 80127-5710
303 971 0292
303 742 1540 (FAX)
info@crtech.com
www.crtech.com

REFERENCES

User's manuals, tutorials, and training notes for all software discussed are freely available in PDF format at www.crtech.com

1. Cullimore, B. et al; "Thermohydraulic Solutions for Thermal Control, Propulsion, Fire Suppression, and Environmental Control Systems;" SAE 1999-01-2159.
2. Cullimore, B; "Optimization, Data Correlation, and Parametric Analysis Features in SINDA/FLUINT Version 4.0;" SAE-981574.