ADAPTIVE SELECTION OF ENGINE TECHNOLOGY SOLUTION SETS FROM A LARGE COMBINATORIAL SPACE

Dr. Bryce Roth*, Mr. Brian German†, Prof. Dimitri Mavris††
Georgia Institute of Technology
Atlanta, GA 30332-0150
and
Mr. Noel Macsotai‡
General Electric Aircraft Engines
One Neumann Way MD G326, Cincinnati, OH 45215

Abstract

This paper describes a method to assist in selecting technology concepts from amongst a pool of candidates such that the resulting concepts yield the best compromise between conflicting objectives, such as design performance and technology risk. The heart of this method is a unique technology impact forecasting environment that is used in conjunction with a genetic algorithm as a tool to efficiently explore the technology combinatorial space. The technique is applied to a commercial turbofan engine technology selection problem of practical interest. A pool of forty technology concepts is proposed and evaluated, the objective being to determine which subset of technologies is the best candidate to go forward into development given conflicting objectives on performance, engine manufacturing cost, and design risk (i.e. cumulative technology readiness).

Introduction

The design of large, multi-element systems is one of the most challenging aspects of engineering because it inherently requires a balance and synthesis of many elements into a functional, coherent whole. A key task in this process is evaluating and selecting new technologies to be incorporated into the next generation product. Technologies must usually be selected during the early stages of design, and it frequently occurs that the designer must choose from a pool of technology options at various levels of readiness. It is critical to quickly and accurately pare down technology options to those that exhibit the best possible balance amongst increased performance, manufacturing cost, design risk, etc.

In general, the complexity of such systems increases geometrically with the number of design options available and the size of the system. Moreover, the earlier in the design process, the more design options are available and the greater the inherent complexity of the problem. It is almost axiomatic that each successive generation of systems will be more complex than its predecessors and will moreover be designed in less time than previously. Consequently, the task of selecting the best combination of new technologies at the early design stages is a formidable challenge and will be more so with each successive product generation.

Fortunately, there are a variety of exciting developments in the field of complex systems design that promise to overcome many of these difficulties. These are based on ideas from a wide variety of fields such as complexity theory, evolutionary computing, adaptive systems, biomimetics, and genetics methods. Collectively, these methods have the potential to revolutionize the way complex systems are designed. Specifically, they promise the ability to create "emergent solutions", that is, solutions that were not explicitly devised by the designer, but are rather allowed to emerge from the "primordial soup" of design possibilities.

This paper will examine one application that represents only a small step toward the ultimate potential of these new techniques – the use of genetic algorithms for the problem of selecting an optimal set of engine technologies from a pool of technology candidates. The engine technology selection problem is a good example for this type of technique because it is inherently multi-objective, multi-disciplinary, and highly cross-coupled.

Combinatorial Optimization of Technologies

Mathematically speaking, technology selection is an exercise in *multi-objective combinatorial optimization*. This is a relatively difficult class of problem to solve, for two reasons. First, the multi-objective nature of the problem implies that there is no single figure of merit to be optimized, but rather the best solution is that which yields an optimum compromise between objectives.

The second reason is that the number of possible combinations increases geometrically with each additional option. For example, if a problem has two options per technology and n technologies, then the number of possible combinations is 2^n. If another technology option is added, the total number of combinations increases to $2(2^n)=2^{n+1}$. If the options may take on more than two discrete levels, or if the parameters are continuous, the problem becomes even more intractable. This is sometimes referred to as the "curse of dimensionality" because each additional dimension brings about a geometric increase in complexity.

---

* Research Engineer, Aerospace Systems Design Laboratory (ASDL), School of Aerospace, Georgia Tech. Member, AIAA.
† Graduate Research Assistant, ASDL. Student Member, AIAA.
†† Boeing Professor, Department of Aerospace Engineering, and Director, ASDL. Associate Fellow, AIAA.
‡ Lead Engineer, Adv. Engineering Programs Dept, GEAE. Member, AIAA.

One way to deal with this problem in technology evaluation studies is through evolutionary/incremental design. It is said that success begets success, and in the engineering world this is largely because of the complexity and cost associated with the design of modern systems. Once a winning combination of technologies is found to work well for a particular application, it is evolved in successive generations and revisions. Each generation refines the design, but the basic premise remains the same. Typically, this process of evolution continues until two criteria are met: 1) there is a demand for a revolutionary change in performance and 2) current technology has reached a maturity plateau such that it is not capable of meeting that demand. The limitation of this approach is that it is slow, and it is seldom possible to investigate more than a handful of the many possible technology combinations.

This limitation has been ameliorated in part through modern analytical models that enable detailed performance simulation of any combination of technologies at greatly reduced expense relative to older "cut and try" methods. This is a vast improvement, but there is still considerable expense associated with constructing and testing such models, especially for preliminary-level estimates of technology benefit where model uncertainty/fidelity is a significant concern.

A second way of coping with the very large design space has been expert opinion. With years of experience, engineers gain an intuitive comprehension of which technology options will work well together and which are non-starters. This intuition, coupled with a well-honed creativity for synthesizing workable designs, has been and will undoubtedly continue to be a formula for success. However, there are limiting factors on the horizon. First, the complexity of modern systems is continually increasing. As time progresses it becomes more difficult for any individual to have sufficient depth of knowledge in every aspect of the design to completely understand the impact of each design decision. Second, it takes a great deal of time and effort to become an expert. Companies today are increasingly unwilling to make this type of investment in a single individual. Instead they are pouring resources into developing "expert" systems to mimic expert knowledge. These expert systems are not likely to be capable of creatively synthesizing new ideas or technology concepts. Therefore, there is a fundamental need for tools that can assist and augment designers' intuitive knowledge with a clear, unbiased, and structured approach.

Technology Identification, Evaluation, and Selection (TIES) Method

The technique used herein to evaluate the impact of various technologies is the Technology Identification, Evaluation, and Selection (TIES) method. TIES is a generic process for technology evaluation that provides a formal means to assist designers in selecting technologies for application to complex engineering systems. TIES has been under development for the past several years and is described in detail by Mavris and Kirby in Refs. 1-6. The method facilitates evaluation of the impact that technologies have on system figures of merit (FoMs) and leverages this information to allow informed decisions on what technologies should be incorporated.

Fundamental Premise of TIES

The fundamental premise of the TIES method is that the impact of all technologies can be quantified in terms of changes in a few key parameters, the technology metrics. In the broadest sense, *a technology metric is a generic measure of technological capability*. For example, lift to drag ratio (L/D) is a good metric for aerodynamic technology capability for a given class of aircraft and specific fuel consumption (SFC) is a good metric for propulsive technology capability for a given class of engine. Both of these metrics directly impact aircraft range, fuel consumption, and endurance according to the relation shown in Fig. 1.

The central idea of the TIES method is that the impact of a given technology at the system level manifests itself primarily in terms of changes in technology metrics. Therefore, if an analytical relationship between technology metrics and system figures of merit (FoMs) is available, then one need only quantify technology impact in terms of changes in metrics, which can then be used to calculate overall performance. In the example above, the Breguet range equation is one such analytical relationship linking technology metrics to system performance: it describes range as a function of L/D and SFC.

The main advantage of quantifying technology impact in terms of technology metrics is that *once the relationships between metrics and FoMs have been created, the impact of any technology can be quickly and easily evaluated without the need to create an explicit model of the specific technology*. Instead, the delta in technology metrics can be determined (usually with reasonably high accuracy) through expert opinion and/or analysis. These deltas in technology metrics are essentially a compact embodiment of the technology model. The result is an analytical tool that uses a blend of analysis and expert opinion to yield a highly cost effective, timely, and flexible means of evaluating multiple technology impact.

$$R = \left(\frac{L}{D}\right)\left(\frac{V}{SFC}\right)\ln\left(\frac{W_f}{W_i}\right)$$

Measure of Aero. Tech.
Measure of Propulsion Tech.
Function of Mission + Structural Technology

Fig. 1 Breguet Range Equation Maps Aerodynamic, Propulsion, and Structural Technology Metrics to Yield Aircraft Range.

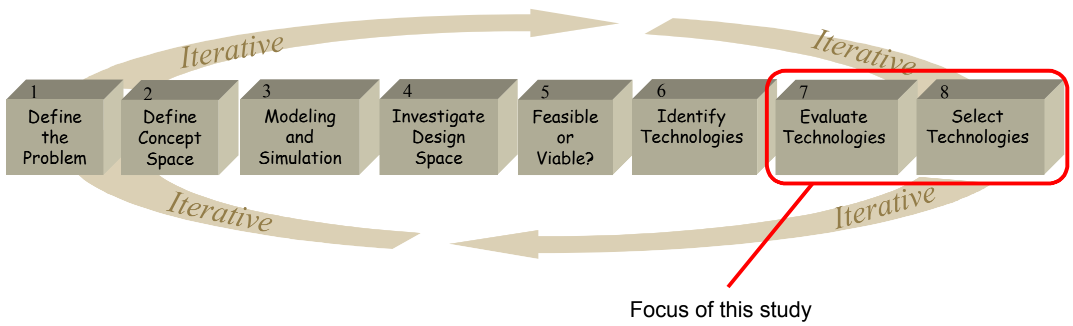

Fig. 2 Steps in the TIES Method (Adapted from Reference 3)

The TIES Framework

The general steps in the TIES method are shown in Fig. 2, and a detailed description of the entire TIES method is provided in Ref. 1. Although TIES includes formalized processes for identifying technology need, the focus of this study is the last two steps of the method: evaluating and selecting technology combinations that create a feasible or viable design space. The steps of the TIES method that are pertinent to this study are:

Steps 1-2: Determine the system responses (or FoMs) of interest and key technology metrics through which the impact is modeled

Step 3: Use analysis tools/models to create metamodel relationships linking technology metrics to system performance FoMs (the technology impact forecasting environment)

Step 6: Map technologies to deltas in technology metrics

Step 7: Evaluate the technology combinatorial space to identify the most promising combinations

Step 8: Use this information to select the appropriate technology combination

The system physics are modeled in step 3, creation of the Technology Impact Forecasting (TIF) environment. TIF is the technology evaluation engine used to evaluate the impact of technologies. It is nothing more than a set of analytical relations mapping technology metrics to system-level figures of merit. These relations can take the form of a computer program such as a mission analysis routine, a CFD code, or a linked system of programs. Therefore, TIF is the embodiment of several computational models of the complex system (a simple example of such a model is the Breguet Range Equation).

If the computation time/cost of evaluating complete analytical models is prohibitive, it is often desirable to create a metamodel (such as a response surface equation, RSE) which captures the essence of the complex analytical model while still being inexpensive to evaluate. The use of metamodels allows the TIF environment to be implemented in a very compact form using a standard statistical software package. Moreover, some of these packages provide interactive graphical depictions of the system responses to the technology metrics, which can be useful in answering "what-if" questions. Details of this method are given in Refs. 4 and 5.

Expressed in mathematical terms, one can think of the technology selection problem as consisting of an abstract space having *m* dimensions, each dimension representing a technology metric. If the impact of each technology manifests itself as a delta in these technology metrics, then we can think of a technology in an abstract sense as a *vector* of technology metric deltas that moves the state-of-the-art from some baseline datum out to a new level of capability. Some of these vectors may only take discrete lengths (i.e. a technology that is either "all" or "nothing"), while some technology vectors will have a continuum of admissible lengths. Some of these vectors are orthogonal to one another (i.e. independent), whilst others are not orthogonal (mutually exclusive options are such an example).

Mechanics of Multiple Technology Evaluation

The mechanics of evaluating performance of designs employing arbitrary mixes of technology is relatively straightforward. First, assume that the technology mix employed in any given design is represented by a binary vector of length *m*, where *m* is the number of technologies under consideration.[§]

$$\vec{T} = \{t_1 \quad t_2 \quad \cdots \quad t_m\}$$
$$\text{where} \begin{cases} t_i = 1 \Rightarrow (\text{Technology 'i' is used}) \\ t_i = 0 \Rightarrow (\text{Technology 'i' is not used}) \end{cases} \tag{1}$$

Next, assume that the impact of each technology concept is quantified (usually via expert opinion) in terms of deltas in a vector of key technology metrics, k_i. This information can be assembled into a matrix format known as a technology impact matrix, or TIM.[6] Each row of the TIM contains a complete description of a single technology in terms of deltas on the *n* elements of k_i. For instance, the first row of the TIM shown below would contain n elements that completely quantify the impact of technology 1 in terms of deltas in the 'k-factors'.

[§] This implicitly assumes that all technologies are either "on" or "off". This assumption can be relaxed to include a continuum of possible states if a suitable technology model is available.

$$\begin{array}{c} \\ t_1 \\ \vdots \\ t_m \end{array} \begin{array}{c} \begin{array}{ccc} k_1 & \cdots & k_n \end{array} \\ \begin{bmatrix} \Delta_{11} & \cdots & \Delta_{1n} \\ \vdots & \ddots & \vdots \\ \Delta_{m1} & \cdots & \Delta_{mn} \end{bmatrix} \end{array} = [TIM] \tag{2}$$

If the simultaneous impact of multiple technologies can be assumed to be additive, then the cumulative impact of all technologies present in a given configuration can be evaluated by simply multiplying the T vector by the TIM.

$$\vec{K}_{1\times n} = \vec{T}_{1\times m}[TIM]_{m\times n} \tag{3}$$

It should be noted that this approach implicitly assumes that interactions between technology impacts are small relative to the primary effect. This resultant K vector is thus a complete description of the impact of all technologies present in the T vector. If one thinks of the T vector as being analogous to the 'DNA' describing the design, then the K vector is analogous to the phenotype of the design; it is the physical manifestation of the technology impact. However, the ultimate aim is to estimate the system performance of the design, so the phenotype must be evaluated to determine its ultimate performance. The resultant response can be easily evaluated using a response surface equation relating the K-vector to the response value. When there are more than a few elements in the K-vector, it is usually most convenient to use a standard quadratic RSE.

$$R(\vec{K}) = I + \sum_{i=1}^{n} l_i k_i + \sum_{i=1}^{n}\sum_{j=i}^{n} q_{ji} k_i k_j \tag{4}$$

Equations written in this form can easily be evaluated using a simple matrix equation. Note that I is the equation intercept, [L] is a vector of linear regression coefficients, and [Q] is an upper triangular matrix of quadratic regression coefficients.

$$R = I + \vec{K}_{1\times n}[L]_{n\times 1} + \vec{K}_{1\times n}[Q]_{n\times n}\vec{K}^{T}_{n\times 1} \tag{5}$$

For technologies that are completely independent of one another, the above equations are sufficient to completely evaluate the technology impact. However, for those cases where the technologies are not independent of one another, it is necessary to augment the above analysis with further equations that define the relationships between technologies.

Technology Interrelationships

There are a variety of technology interrelationships that can exist, and the exact nature of these relationships will vary from problem to problem. Therefore, it is not practical to give a complete and comprehensive description of every possible relationship between technologies. However, there are a number of common interrelationships that arise frequently enough as to merit a detailed discussion and development.

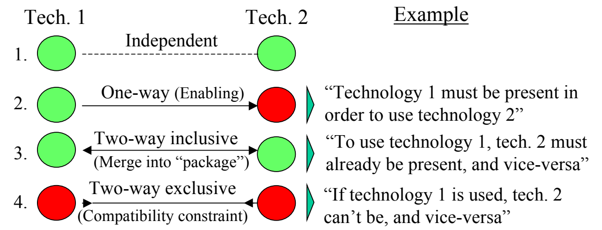

Fig. 3 Logical Relationships Between Technologies.

By far the most common technology inter-dependencies encountered are logical relationships. These consist of "if-then-else" type relations, and are summarized in Fig. 3. The simplest possible relationship between any two technologies is for them to be completely *independent* of one another. A second common relationship is the one-way relationship or *enabling technology*. An enabling technology is defined as a technology that must be employed as a necessary prerequisite to enable the use of another technology. A third type of relationship is *two-way inclusivity* wherein if one technology is used, so must another and vice-versa. The fourth logical relationship is *two-way exclusivity*, wherein if one technology is used, a second cannot be included and vice-versa. Additionally there can be multi-way interactions amongst three or more technologies as well as non-logical relationships wherein the addition of one technology changes the impact of another. The latter two situations are not addressed here.

Each of these technology relations can be captured in the technology impact evaluation with suitable modifications. First, enabling relationships are captured through an enabling technologies matrix. This is nothing more than an $m\times m$ matrix with each row representing an enabled technology and each column representing an enabling technology. The matrix elements are either zero (indicating no enabling relationship) or one (indicating that the technology in the current column enables the technology in current row).

$$\begin{array}{c} \\ t_1 \\ t_2 \\ \vdots \\ t_m \end{array} \begin{array}{c} \begin{array}{cccc} t_1 & t_2 & \cdots & t_m \end{array} \\ \begin{bmatrix} 1 & e_{12} & \cdots & e_{1m} \\ e_{21} & 1 & \ddots & \vdots \\ \vdots & \ddots & 1 & e_{m-1,m} \\ e_{m1} & \cdots & e_{m,m-1} & 1 \end{bmatrix} \end{array} = [E] \tag{6}$$

$$\text{where} \begin{cases} e_{ij} = 1 \Rightarrow (\text{Tech. 'j' enables 'i'}) \\ e_{ij} = 0 \Rightarrow (\text{Techs. 'i' and 'j' independent}) \end{cases}$$

Enabling relationships can be introduced into the technology analysis process in two ways. First, they can be treated as an external constraint during the combinatorial optimization process. This involves minimizing the number of unsatisfied enabling relationships by treating the sum of violated enabling constraints as an additional objective function that is minimized during the optimization process. The number of violated enabling constraints is easily computed using a simple equation.

$$EC = \left\| \rho\left(\vec{T}[E]\right) \right\|_{\inf} \tag{7}$$

The ρ function** is defined here as a function that maps all counting numbers into 1 and all other numbers into zero. The number of enabling constraint violations (EC) can therefore be treated as an objective function to be minimized during combinatorial optimization (i.e. an external constraint).

Alternatively, the enabling relationships can be rigidly enforced by applying a transform to the T-vector that *adds in* the needed enablers based on the relationships defined in the enabling matrix. These can be calculated using a simple matrix equation.

$$\vec{T}' = \rho\left(\vec{T}[E]\right) \tag{8}$$

In this case the T-prime vector now becomes the input to the technology evaluation process, replacing the T-vector.

The second technology interrelationship commonly of interest is mutual inclusivity. The best way to treat this type of constraint is to merge the individual technologies into a single package and treat that package as a technology suite that is either on or off. It should be pointed out that this case differs from the previous in that mutual inclusivity implies absolute positive correlation between technologies, whereas enabling implies that one technology is dependent whilst the second is independent.

The third interdependency is mutual exclusivity. This implies perfect negative correlation between technologies, meaning that two technologies are incompatible. These constraints are incorporated into the analysis via a compatibility matrix .[6]

$$\begin{array}{c} \\ t_1 \\ t_2 \\ \vdots \\ t_m \end{array} \begin{array}{c} \begin{array}{cccc} t_1 & t_2 & \cdots & t_m \end{array} \\ \begin{bmatrix} 0 & c_{12} & \cdots & c_{1m} \\ & 0 & \ddots & \vdots \\ & & 0 & c_{m-1,m} \\ 0 & & & 0 \end{bmatrix} \end{array} = [C] \tag{9}$$

$$\text{where} \begin{cases} c_{ij} = 1 \Rightarrow (\text{Tech. 'j' incompatible w/ 'i'}) \\ c_{ij} = 0 \Rightarrow (\text{Techs. are independent}) \end{cases}$$

Compatibility can be treated as an external constraint in the same way as the enabling constraints by using the number of technology incompatibilities that occur as an objective function to be minimized. The number of technology incompatibilities can be calculated in similar manner to the enabling technologies.

$$CC = \left\| \rho\left(\vec{T}[C]\right) \right\|_{\inf} \tag{10}$$

Alternatively, one can rigidly enforce compatibility by transforming the T-vector to *turn off* technologies such that compatibility is satisfied. The vector of incompatible technologies can be calculated using an equation similar to that for enabling constraints.

$$\vec{T}' = \vec{T} - \bigcup\left\{\vec{T}, \rho\left(\vec{T}[C]\right)\right\} \tag{11}$$

This approach has an implicit bias toward retaining the first technology in the T-vector and removing the later incompatible technology within the ordinal pair.

Both approaches to handling compatibility and enabling constraints work reasonably well. The external constraint approach has the advantage of being simple to implement, fast to evaluate, and easy to modify. Its principle disadvantage is that there is no guarantee that the constraints will be satisfied, only that they are driven in that direction. The latter approach guarantees satisfaction of constraints, but is slightly more complicated to implement, is more expensive to evaluate, and inherently introduces bias into the solution. In particular, rigid enforcement of compatibility constraints as formulated above will tend to bias the solution away from technologies that have compatibility restrictions. Furthermore, the order in which the enabling and compatibility constraints are rigidly enforced will impact the solution. Finally, if enabling and compatibility are rigidly enforced, it is possible to encounter multi-way interactions amongst technologies (tech 1 is enabled by 2 that is enabled by 3 which is not compatible with 1), with unpredictable results. Consequently, the external constraint approach is used for the technology study described herein due to its simplicity.

It should be noted that the formulation presented here is but one of many. For example, it may be possible to combine the compatibility and enabling matrices into a single matrix if a suitably elegant formulation is devised. In addition, it is possible to create formulations to account for interactions amongst technology impacts. However, this is beyond the intended scope of this discussion, and the reader is referred to Ref. 3 for further discussion.

__Technology Selection via Genetic Algorithms__

After a TIF environment has been developed, the technologies can be evaluated to determine their impact on the system responses. Within the TIES method, the technologies are first evaluated individually. This evaluation is accomplished by simply specifying the vector of metrics for each technology as inputs to the modeling and simulation environment. Although these individual technology impacts provide valuable information, the true strength of the TIES method is that it allows rapid evaluation of multiple technologies applied concurrently.

A limiting factor in exploring this space is that the size of the combinatorial problem becomes immense as the number of technologies within the candidate pool increases. For instance, with 40 technologies, the number of possible combinations is 2^{40}, or approximately 1.1

** Algorithmically, the rho function is easily implemented in code via 'if' statements and 'do' loops.

trillion combinations. Even with the greatly simplified computational environment afforded by TIF, exhaustive search is intractable for all but very simple problems. If there are constraints on technology compatibility, enabling relationships, etc., then the number of admissible combinations can be reduced considerably, but usually not enough to permit exhaustive search.

This problem can be solved through the introduction of a genetic algorithm (GA) as a combinatorial search engine to efficiently explore the design space and find the optimum technology combinations. A genetic algorithm is an optimization technique that mimics biological reproduction and evolution.[7] In this case, it uses a "chromosome string" of binary digits to represent whether a certain technology is "on" (1) or "off" (0). Thus, for 40 technologies, the chromosome string is a 40 digit binary number.

A GA typically consists of several operations including a reproduction or replication algorithm, a crossover function, and a mutation operator. The purpose of the reproduction function is to replicate those members of the GA population with the highest fitness and delete those members with the lowest fitness. This process is similar to the concept of the "survival of the fittest." The crossover function interchanges information between chromosomes of the fittest individuals in the population by randomly swapping sections of the chromosome strings between "parents." In mutation, individual bits are randomly selected from within chromosome strings and their values are swapped. This process ensures that combinations with bit values not contained within the initial population have a possibility of being created.

The advantage of genetic algorithms for solving the technology combinatorial problem is that they can "home in" on promising regions of the combinatorial space very rapidly. With 40 bit combinations, GAs cannot exhaustively investigate the entire design space or even explore more than a tiny fraction of it; however, with repeated runs, the GA can identify promising patterns and find the global optimum in a very repeatable manner.

GAs are implemented within the TIES framework by "wrapping" them around the TIF metamodels. The TIF metamodels collectively play the role of fitness functions that are used to evaluate the overall "goodness" of any particular technology suite. Typically, the GA is initialized with a fixed population of technology combinations, and runs for several generations. As the GA is applying mutation and crossover operations to the population, it calls the TIF metamodels to evaluate fitness. The most promising chromosome strings (technology combinations) are those with high fitness. This information is used by the GA to find optimal combinations of chromosome strings. Multiple objectives are accommodated by having the GA use each of the objective functions with a predetermined frequency (weighting).

Technology Selection for Turbofan Engines

This section illustrates the application of genetic algorithms for technology selection as a component of the TIES methodology via a technology selection study for a new turbofan engine. The focus of the study was to illustrate the procedure by selecting the optimum technology suite from a combinatorial space that would be intractable to explore using an exhaustive search.

Baseline Engine and Aircraft

The baseline engine considered in this study is a current-technology high-bypass commercial turbofan engine. The engine is representative of today's state of the art and therefore makes a good point of departure for technology studies. The baseline aircraft is a notional twin-engine commercial widebody. This aircraft was selected because the long mission range makes the vehicle performance and economics highly sensitive to fuel consumption and engine weight and is therefore a good candidate for application of advanced engine technologies. The aircraft is assumed to have a fixed airframe zero fuel weight (less engine and pylon weight), although empty weight is allowed to vary as engine and pylon weight fluctuate. Standard mission rules and assumptions are used, assuming standard day conditions, and horsepower extraction/engine bleed flow were scheduled based on typical requirements. Three missions are considered in this study: 3,000 nmi, 6,000 nmi, and design range. These were chosen to give a relatively broad spectrum of likely scenarios that the actual vehicle will encounter in operational use.

Analytical Model

The first step in implementing the technology evaluation portion of the TIES method is to define technology metrics and ranges that capture the impact of the technologies under consideration. The engine technology metrics (K-factors) and ranges for this technology study are listed in Table 1. These parameters

Table 1: Engine Technology Metrics and Ranges Relative to Baseline.

Technology metric	*Min*	*Max*
Δ Engine weight	-7.5%	+2.5%
Δ T41	-6%	+6%
Δ OPR	0%	+40%
Δ CPR	-21%	+21%
Δ Compressor efficiency	-0.5 pt	+1.5 pt
Δ Booster efficiency	-0.5 pt	+1.5 pt
Δ Fan efficiency	-0.5 pt	+1.5 pt
Δ HPT efficiency	-0.5 pt	+1.5 pt
Δ LPT efficiency	-0.5 pt	+1.5 pt
Δ HPT Ch. Cool	-5%	+10%
Δ HPT NC Cool	-4%	+8%

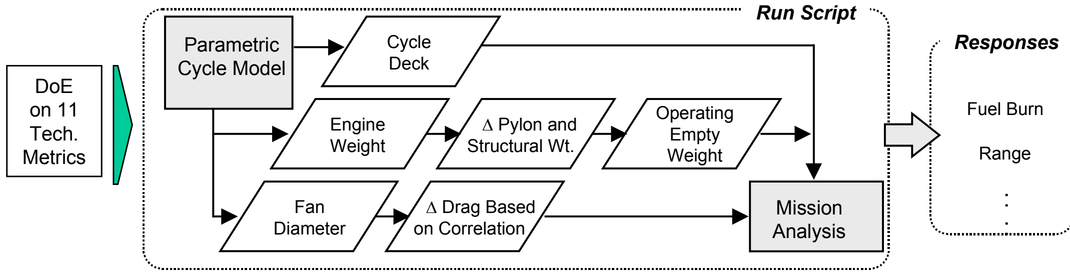

Fig. 4 Analysis Process Flow.

fall into three categories: cycle, aerothermo, and weight parameters. These were selected based on their importance as technology metrics for commercial turbofan engines. The cycle parameters of interest are core pressure ratio (CPR), overall pressure ratio (OPR), and maximum allowable turbine inlet temperature (T41). The aerothermo technology metrics considered in this study consist of five turbomachinery efficiencies and two high pressure turbine (HPT) cooling flow rates. These 11 factors are sufficient to capture the impact of all technologies considered for the current problem.

The basic analysis setup for this study is illustrated in Fig. 4. First, a parametric engine deck is used to generate basic cycle performance and flowpath data based on an input vector consisting of the 11 technology metrics mentioned previously. Next, engine performance data is fed to a mission analysis code in the form of an engine deck. Calculated engine weight is then added to the airframe weight (with a correction on pylon weight to account for changes in engine weight) to arrive at aircraft empty weight. In addition, a correction on aircraft drag is applied based on fan diameter to account for changes in nacelle drag. Finally, the corrected aircraft model is analyzed using a standard mission analysis code to determine aircraft fuel burn for 3K and 6K nmi missions and range for the design range mission.

The system-level responses of primary interest for this study are shown in Table 2. The primary objective is in finding technologies to reduce total fuel consumption. Therefore, mission fuel required for the 6,000 nmi mission is of primary import for this study. In addition, design range is also of interest, and, to a lesser extent, mission fuel for the 3,000 nmi mission.

Creation of Technology Space RSEs

Once the technology metric ranges are selected and the analytical model is set up, the next step is to create response surface equations relating the technology metrics to the system responses. These equations are of the general form shown below.

(Des. Rng) = f(ΔOPR, ΔT41, …, Eng Wt, ΔFan η, ...)

(6K Fuel) = f(ΔOPR, ΔT41, …, Eng Wt, ΔFan η, ...)

etc.

In this case, a design of experiments approach was used to construct a 151 case central composite matrix varying 11 variables at 3 levels. These 151 cases were then analyzed using the analysis framework shown in Fig. 4 to obtain vehicle and engine performance. This data set was then analyzed using standard response surface analysis techniques to obtain RSEs for each response as a function of the technology metrics.

Typical results for these equations are illustrated in Fig. 5, which shows prediction profiles for design range and 6K fuel burn as a function of the technology metrics. Each subplot in the figure can be thought of as a sensitivity in that the steeper the line, the more sensitive the response is to the corresponding factor. Moreover, the RSE also captures curvature in the response, which is information above and beyond typical sensitivities. Note that all the sensitivities depicted on this plot show roughly the same order of magnitude, meaning that all the technology metrics have roughly equal influence on system level performance. The equations obtained from this process have been validated and generally predict performance within 1-2% of the value obtained by the full analysis.

Technology Impact Matrix

The next step in the analysis process was to select a set of technologies for consideration and quantify their impact in terms of deltas in the technology metrics of Table 1. A set of 40 technologies applicable to high bypass turbofan engines was selected for analysis in this study. These 40 technologies consisted of 10 compressor technologies, 9 HPT technologies, 10 LP spool technologies, 7 frame/sump/bearing technologies, and 4 combustor technologies. This specific set of technologies was selected by a group of experts as being those that are currently of greatest interest for the commercial turbofan

Table 2: System Responses of Interest.

Propulsion System
- Cruise SFC (M0.8, 35K)
- Exhaust Gas Temperature (R)
- Fan Diameter (in)
- Engine Weight (lb)

Installed Performance
- Design Range (nmi)
- Mission Fuel Burn (lb, 3K & 6K nmi)

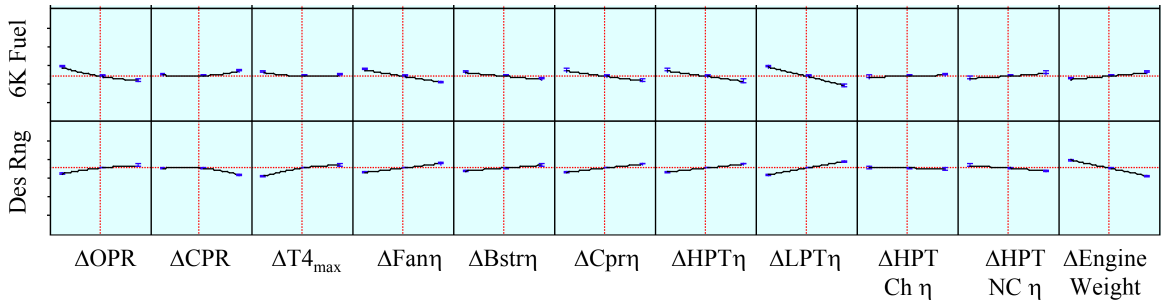

Fig. 5 Prediction Profiles for Design Range and Mission Fuel Burn as a Function of Engine Technology Metrics.

application. These same experts also evaluated the impact of each technology in terms of the technology metrics specified previously. Table 3 shows a subset of the TIM focusing on the 10 compressor technologies, numbered 1 through 10. Note that the last columns are technology readiness level and relative manufacturing cost, respectively. These are subjective ratings based on the intuition of experts as to the technology readiness (using the NASA TRL scale of 1-9, with 9 being production-ready) and the production cost. Note that a production cost score of 0 means the technology is roughly the same cost as the baseline, while positive scores indicate higher cost.

In addition, this same set of experts evaluated the interactions between technologies to determine which were incompatible with one another, and where enabling technology relationships existed. This information was captured in the form of a compatibility matrix and an enabling technologies matrix. These matrices in conjunction with the RSEs described previously constitute a complete analytical model for the evaluation of technologies. In effect, the matrices and RSEs can conceptually be thought of as a "black box" that maps any combination of technologies into system level responses.

Engine Technology Selection Via Genetic Algorithm

As an example of the power of the genetic algorithm approach, consider a scenario where the objective is to find the set of technologies that represents the optimal balance between technology risk, production cost, and fuel efficiency for the 6K nmi mission. The approach typically used for this type of problem is to use a "one off" analysis where all technologies are applied at once and individual technologies are taken off one at a time to evaluate the system benefit of each technology. The delta between the "all on" case (modified by removing any incompatibilities) and the various "one off" results represents the benefit provided by that technology. The technologies can then be ranked according to their performance benefit. It would then be up to the experts to decide based on intuition and experience which technologies do not provide sufficient performance benefit relative to their cost and risk to merit inclusion in the final solution. This selection process is relatively simple to implement, but has the drawback that it is subjective in that two different experts may arrive at different technology solution sets. Moreover, since the technologies are either "on" or "off", standard gradient-based optimization techniques are of little use.

However, a genetic algorithm is not subject to these shortcomings. Both continuous and discrete variables can be optimized, and it can very efficiently select an optimal combination from amongst many possibilities. Therefore, if genetic algorithms are used in conjunction with the technology evaluation environment described previously, it is possible to *analytically* determine which technology set represents the optimal compromise

Table 3: Technology Impact Matrix for 10 Compressor Technologies.

Technology Number	Δ Engine Weight (%)	Δ CPR (%)	Δ OPR (%)	Δ Max TIT (%)	Δ Fan Efficiency (pt)	Δ Booster Efficiency (pt)	Δ Compressor Efficiency (pt)	Δ HPT Efficiency (pt)	Δ LPT Efficiency (pt)	Δ HPT Charg. Cooling (%)	Δ HPT Non-Ch. Cool (%)	Technology Readiness Level	Relative Cost (Current Tech.)
1	-0.02	1.9	2.1	0.3	0.0	0.0	0.2	0.0	0.0	0.0	0.0	5	1
2	0.00	1.9	2.1	0.3	0.0	0.0	0.2	0.0	0.0	0.0	0.0	2	3
3	-0.16	0.5	0.5	0.2	0.0	0.0	0.1	0.0	0.0	0.0	0.0	9	1
4	0.00	2.9	3.1	0.9	0.0	0.0	0.5	0.0	0.0	0.0	0.0	6	1
5	-0.01	6.0	12.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4	-1
6	0.00	1.0	1.0	0.9	0.0	0.0	0.5	0.0	0.0	0.0	0.0	2	1
7	0.00	2.9	3.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	6	1
8	-0.03	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	9	0
9	-0.02	0.0	0.0	0.3	0.0	0.0	0.2	0.0	0.0	0.0	0.0	6	3
10	-0.90	0.0	0.0	0.4	0.0	0.1	0.1	0.0	0.0	0.0	0.0	4	3
etc…													

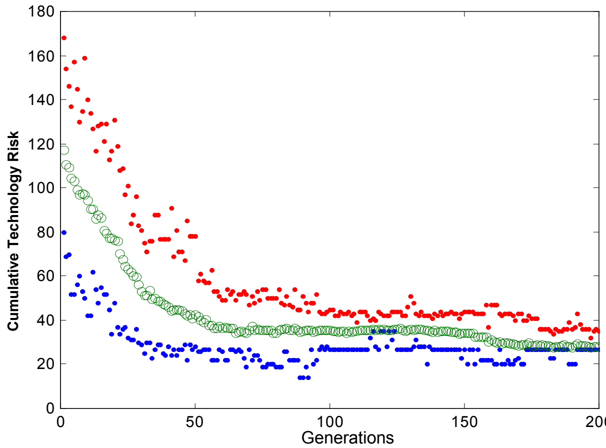

Fig. 6 Evolution of Cumulative Technology Risk in the Engine Design Population.

between the three objective functions: technology risk (given by the compliment of TRL), relative shop cost, and 6K fuel burn. The only subjectivity arises in the importance weightings that the user assigns to the objective functions and the technology metric deltas assigned to each technology.

For this problem, a simple tournament-style genetic algorithm code was implemented in MATLAB. The input parameters are population size, mutation rate, and number of generations (or tournament rounds). The initial population is selected at random, with a 50% probability that any technology in any given population member will be turned on. The genome consists of a 40 bit string with one bit per technology. Crossover is performed by splicing the genomes of two parents together at a randomly selected point. Mutation consists of a random bit flip occurring with a user-specified frequency. Population size is held constant by deleting the weaker of a randomly selected pair of population members competing in each tournament. The weaker member is determined by lower fitness, as evaluated by randomly selecting one of the three objectives. In this example, the three objectives are assumed to have equal probability of selection as the fitness function (i.e. equal weighting).

In this study, a population of 200 designs was allowed to evolve over 200 generations with a crossover rate of 70% and a mutation rate of 2%. This procedure required approximately 2 minutes of run time on a PC. Typical results from this scenario are shown in Fig. 6, which shows the evolution of cumulative technology risk over 200 generations. The top set of points is the maximum technology risk design in the population, while the bottom represents the minimum. The circles in the middle represent the average technology risk in the aggregate population. Note that the average technology risk is relatively high initially and decreases precipitously as high risk, low payoff technologies are removed from the population. A similar trend for cumulative shop cost

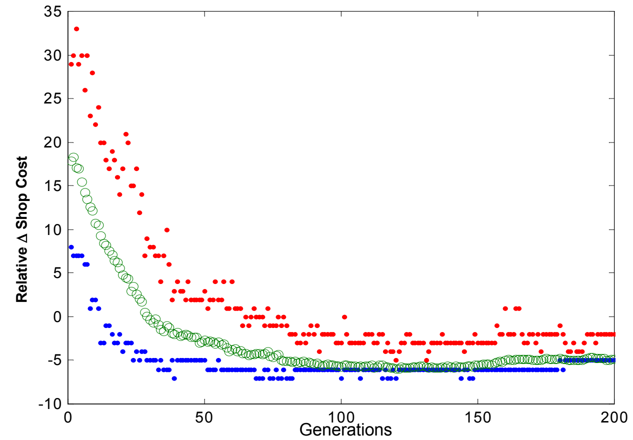

Fig. 7 Evolution of Cumulative Shop Cost in the Engine Design Population.

is shown in Fig. 7, with the final average shop cost actually being slightly lower than the baseline.

Fig. 8 shows a similar plot for percent change in 6,000 nmi mission fuel burn relative to an arbitrary reference point. Note that the trend is reversed in this case from the previous objectives, with the final solution having a fuel burn 3% higher than the initial population. This is because each technology has a 50% probability of being active in any member of the initial population. However, as the population evolves, the high risk, high cost solutions are removed from the population, and the average fuel burn increases as a consequence. Note that the last generation for all three objective functions is fairly well converged, as evidenced by the relatively narrow separation between min and max in the population. This indicates that the population has become highly uniform, all members having converged to essentially the same set of technologies. Note that unlike classical optimization, the population member with the minimum fuel burn in the last generation is not necessarily the individual that represents the best solution set. This is because that same member may have high risk or high cost. Rather, the *most frequently occurring* technologies in the final population represent the converged optimal solution for the prescribed objective criteria.

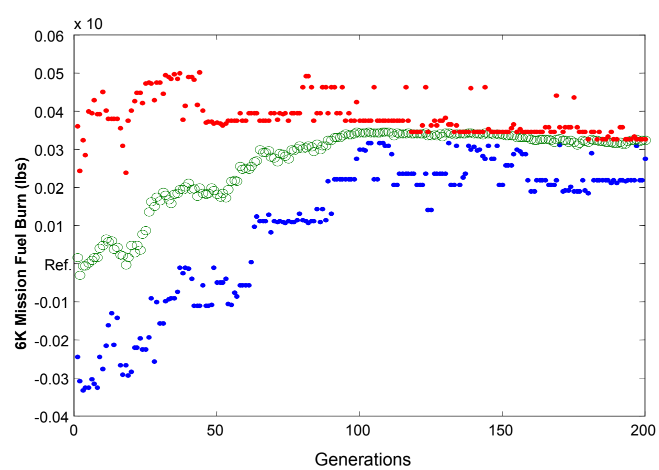

Fig. 8 Evolution of 6K Fuel Burn.

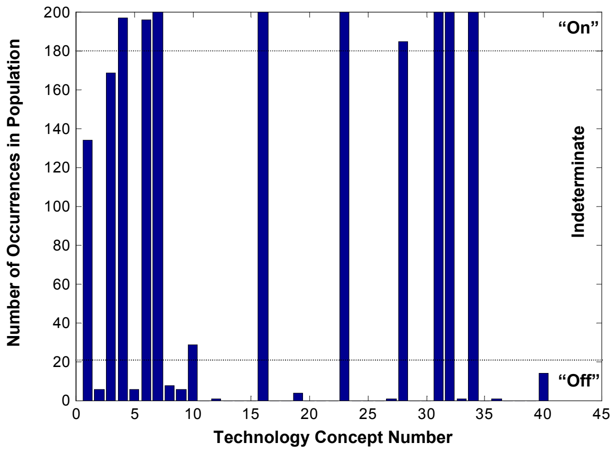

Fig. 9 Optimal Technology Solution Set.

The resulting solution set from this genetic optimization process is shown in Fig. 9. This figure shows the number of occurrences of each technology (numbered technology 1-40) in the last generation. Thus, if a technology is uniformly present in the last generation (i.e. it was ‘selected’), then it appears 200 times in the final generation. This occurs for technologies 7, 16, 23, 31, 32, and 34. However, since random mutations occur throughout the optimization process, there will always be some members of the population that have been perturbed from the optimal point through mutation but have not had sufficient opportunity to converge back to the optimum. Therefore, it is reasonable to define a cut-off point above which one can say that the GA selects the technology. In this case, a technology is taken as being selected if it is present in 90% or more of the population. Likewise, a technology is not selected if it occurs in less than 10% of the population. Technologies which occur more frequently than 10% but less than 90% are indeterminate. No conclusive decision can be drawn regarding these technologies because their impact on the objective functions is so balanced as to have no net impact on population fitness. Thus, the technologies representing the optimal balance between cost, risk and performance are 4, 6, 7, 16, 23, 28, 31, 32, and 34. The indeterminate technologies are 1, 3, and 10. All other technologies do not provide sufficient performance benefit to warrant the added risk and cost.

Conclusions

One of the most difficult problems in selecting technologies for application to a complex system design is in dealing with the geometric increase in the number of possible concepts as more technologies are added to the candidate pool. This paper has presented a method for addressing this problem using genetic algorithms within the framework of the Technology Identification, Evaluation, and Selection method applied to a notional engine technology selection problem. The results of this research show that genetic algorithms implemented in the TIES environment offer substantial advantages in efficiently exploring technology combinatorial spaces to select the most promising technology combinations. Moreover, the end result is an *analytical* solution obtained through the synthesis of expert knowledge and analytical models. Finally, the TIES environment offers considerable flexibility in that the basic environment can be used for a variety of purposes, and the analytical technology impact model can be re-used with ease if so desired for future studies.

A few points bare mention regarding implementation of the analysis method. First, one of the assumptions of the technology evaluation method is that the technology impacts are additive. For those cases where there is significant interaction between technologies, it may be necessary to modify the analysis method to account for the interactions. Second, the accuracy of the analysis depends heavily on accurately populating the technology impact matrix. Therefore, considerable care should be taken to ensure that all participants employ the same assumptions in preparing the TIM. Additionally, the TIM should be thoroughly validated before use. Finally, the stochastic nature of the genetic algorithm prevents one from conclusively stating that the final population represents a true optimum. For this reason, it is advisable to conduct repeated runs of the algorithm to give confidence in the final selection. The results of repeated tests for the example problem in this study show good repeatability in the most prevalent technologies.

Acknowledgements

The authors would like to thank the Office of Naval Research for supporting portions of this research. Thanks also to Mr. Larry Dunbar, Dr. Dave Halstead, and Mr. Greg Steinmetz of GEAE for their contributions. In addition, we would like to thank Mr. Mathew Graham and Mr. Tom Ender for their contributions.

References

[1] Mavris, D.N., Kirby, M.R., "Technology Identification, Evaluation, and Selection for Commercial Transport Aircraft", 58th Annual Conference of the Society of Allied Weight Engineers, San Jose, California 24-26 May, 1999.

[2] Kirby, M.R., Mavris, D.N., "A Method for Technology Selection Based on Benefit, Available Schedule and Budget Resources", SAE 2000-01-5563, presented at the 2000 World Aviation Congress, San Diego, CA, October 10-12, 2000.

[3] Kirby, M.R., *A Method for Technology Identification, Evaluation, and Selection in Conceptual and Preliminary Aircraft Design*, Ph.D. Thesis, Georgia Institute of Technology, March 2001.

[4] Kirby, M.R., Mavris, D.N., "Forecasting the Impact of Technology Infusion on Subsonic Transport Affordability", World Aviation Congress and Exposition, Anaheim, CA, September 28-30, 1998. SAE-985576.

[5] Mavris, D.N., Kirby, M.R., Qiu, S., "Technology Impact Forecasting for a High Speed Civil Transport", World Aviation Congress and Exposition, Anaheim, CA,September 28-30, 1998. SAE-985547.

[6] Kirby, M.R., Mavris, D.N., “Forecasting Technology Uncertainty in Preliminary Aircraft Design”, SAE Paper 1999-01-5631, 1999 World Aviation Congress, October 1999.

[7] Goldberg, D., *Genetic Algorithms in Search Optimization, & Machine Learning*, Addison-Wesley, Reading, MA, 1989.