39th AIAA Thermophysics Conference
25 - 28 June 2007, Miami, FL

AIAA 2007-4054

An Advanced Exergy and Energy Simulation Tool for Large-Scale Design/Optimization of Aerospace Systems

Kehinde Alabi[*], Foluso Ladeinde.[†]
TTC Technologies, Inc., P. O. Box 1527, Stony Brook, NY, 11790-0609

Michael vonSpakovsky[‡]
Department of Mechanical Engineering, VPI, Blacksburg, VA, 24061-0238

and

David Moorhouse[§], Jose Camberos[**]
AFRL/VASD, Wright-Patterson Air Force Base, Dayton, OH 45433-7542

This paper discussed procedures developed for the design/optimization of large-scale aerospace systems; using the iterative local global optimization (ILGO) – procedure. The ILGO procedure obviates the need for nested optimization loops in the design of a complex system decomposed into several subsystems. A new object-based scripting tool is developed and used to analyze an advanced tactical fighter with an exergy-based objective. The results are compared with those obtained using a weight-based objective.

I. INTRODUCTION

Aircraft design is a complicated undertaking involving numerous variables and requiring the convergence of experts from different disciplines. This paper focuses on the development of a practical exergy-based software tool for the design and optimization of complex aerospace systems. The emphasis is on the entire system as a unit, incorporating the complex interactions and requirements of the various subsystems or components comprising the system. Traditionally, engineers have tackled this task using trade-off analysis, handbooks and specifications, and rule of thumb. With increasing computational power, attention has focused on the solution of design/optimization problems in an integrated manner. Initial solutions have been based on reduced models in which many of the subsystem or component details are simplified, leading to inaccurate solutions.

Formulating and solving the large aerospace systems as a single-level problem is difficult for three reasons. The first reason is the large size of such problems and the sheer number of variables. Secondly, the tools for analyzing the different components typically consist of computer programs on different platforms, making an integrated analysis difficult. A third reason is the fact that most systems are designed by several different engineering units, sometimes from different organizations or different geographic locations. Consequently, procedures that utilize decomposition methods are needed for realistic large-scale design/optimization efforts.

Several decomposition methods have been used, including physical, disciplinary, conceptual, and time-based decomposition.[1-4] Decomposing into subsystems allows the modeling of each subsystem by different groups of engineers, making projects more manageable.

---

[*] Research Engineer, Ph.D, AIAA Member.
[†] Director of Research, AIAA Life Member & Associate Fellow.

[‡] Professor & Director for Center for Energy Systems Research, AIAA Member.

[§] Director, Multidisciplinary Technologies Center, AIAA Associate Fellow.
[**] Aerospace Engineer, AIAA Associate Fellow.

One consideration in the design of aircraft systems is whether or not to design for a single point or over an entire mission. Designs optimized for a single point are easier to formulate but do not guarantee an optimal vehicle over a mission range and/or over the life of the vehicle, nor does it consider variations in atmospheric conditions associated with different missions.

Munoz and von Spakovsky[5] and Rancruel and von Spakovsky[6,7] developed and successfully applied various decomposition procedures to a high performance aircraft system. The subsystems include the airframe – aerodynamics, propulsion, fuel loop, environmental control, thermal management, electrical, hydraulic, oil loop, controls, and expendable payload. The procedure developed in their work is employed in the current study.

The level of detail simulating the components that should be included in the system requires some consideration. Multi-level approaches permit varying levels of detail or degrees of fidelity to be used in an integrated fashion. A problem with the multi-level optimization approach is the nested optimization loops implied in the procedure. The iterative local-global optimization (ILGO) procedure[1,5,8-11] removes the need to nest the optimization by using “shadow functions” or a gradient-type formulation based on the coupling functions.

In the current study, the objective function is based on exergy destruction. This allows the analysis of systems consisting of several energy objectives which are difficult to combine to be analyzed with a single exergy-based objective. This is important since multi-objective analysis becomes more difficult, computationally expensive, and complicated in analyzing, as the number of objectives increase. In addition, exergy-based analysis allows the identification of regions with potential for design improvement in the course of the process.

This paper reports on the tools developed to permit the modeling, design, and optimization of large aerospace systems. The procedures are packed into a new modeling platform called iSCRIPT, which enables the modeling of engineering systems using paradigm that focuses on developing components composed of engineering variables, and systems composed of components, in a building-block framework. iSCRIPT can also be used as a straightforward programming tool and implements a programming language that was designed to accept models developed in FORTRAN or MATLAB. The ILGO procedure for system optimization is also implemented in iSCRIPT. Finally, iSCRIPT performs optimization in parallel, without requiring the user to explicitly parallelize the program.

The next section introduces current tools used in the optimization of large scale systems, persistent problems with these methods, and how the current simulation tool addresses these problems. The tools developed for the modeling, design, and optimization of large scale aerospace systems are described in Section III. Section IV describes the formulation of the single objective exergy function as well as the weight-based objective functions. Section V presents the results of the analysis and concluding remarks are contained in Section VI.

II. Design/Optimization Analysis Procedures

The techniques for handling a multi-level design/optimization problem can be divided into three tasks shown in figure 1.

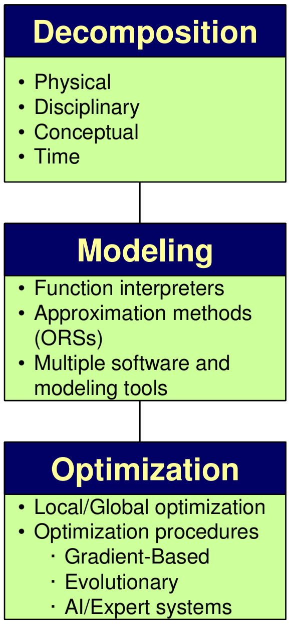


Figure 1. Parts of an applied MDO analysis

A. Decomposition

In physical decomposition, the system is divided up into physically-interacting subsystems, each possessing a certain degree of autonomy but depending on other subsystems via a number of coupling variables (Alexandrov & Kodiyalam, 1998)[2]. Disciplinary decomposition divides the system along the lines of different disciplines such as thermodynamic, economic, aerodynamic, etc. Conceptual decomposition breaks down the system according to the type of variables. For instance, operational variables which vary in time and those that do not vary in time. Time decomposition decomposes a dynamic problem into a series of quasi-stationary ones or a series of stationary time segments.[10,11]

B. Modeling

Modeling of the various subsystems typically involves software products from different vendors. A great challenge in this step is the integration of the different software products. Several levels can be identified in the multi-level modeling and optimization process:

Low Level Function Interpreters and Symbolic Language Programs

These are tools that allow an engineer to specify the equations and models comprising a component in mathematical form, aggregate these low level models into higher level models through additional mathematical expressions and functions. In principle, a complete system can be built using low level functions. However, the procedure is difficult and prone to error. Sample tools in this category include spreadsheets such as Excel, programming languages, and scripting languages such as MATLAB, Mathematica or Maple.

Aggregated Component Tools

These are prepackaged tools for specific models. Examples include engine simulator for computing the thrust and weight of the propulsion system (e.g. Weight Analysis of Turbine Engines, WATE) or ADVISOR – a public domain drive-train analysis tool, or a heat exchanger program for the various heat exchangers in the sub-systems of the aircraft. Component tools are typically treated as a “black box” in the integration of models into the complete system.

Approximation Tools

Response surfaces may be generated from measurements on a component as a function of selected decision variables and used as the model in the multi-level optimization phase.

C. Optimization

The use of detailed representation of a component is a critical factor in terms of computational resources. The problems of interest for aerospace systems are typically multi-objective, with objectives such as:

- drag Minimization (or maximization of lift/drag ratio in a mission segment)
- gross take-off weight minimization
- fuel consumption minimization
- minimization of acoustic noise during take-off and landing
- cost minimization (capital, operating, and environmental)

Multi-objective optimization can be officially done only by a few methods, such as genetic-based algorithms. Many cannot be solved at all even by the best algorithms. In addition, the analysis of the results is not trivial once more than three objectives are active. Multi-objective pareto optimal fronts (POFs), which usually form the basis of multi-objective analysis, are difficult to interpret or visualize for dimensions greater than three. Exergy-based methods provide a means of reducing the number of dimensions by combining several different energy-based objectives into one single objective function. The considerable requirements of the optimization task are further alleviated by the use of the iterative local/global optimization (ILGO) method and the choice of optimization procedure as discussed below.

Gradient-based optimization methods work well for subsystems with continuous variables but are prone to local optima. Procedures based on evolutionary algorithms and expert systems are more computationally intensive but they are not as prone to local optima and can be used for mixed integer problems. Aerospace problems typically include both integer, Boolean, and continuous variables. Evolutionary algorithms are utilized to isolate the optima while the gradient based method can be combined with the evolutionary algorithms to speed up the "climb to the peak", once all integer variables are set. In other words, a combination of several optimization procedures is typically used for a complex problem.

III. Tools Developed for Large-Scale System Design and Optimization

The techniques used in the current project have been developed into a program called iSCRIPT. iSCRIPT is developed from a conceptual design based on object representation of components, sub-systems, and systems in a building block approach.

The design allows the inclusion of components, which may form subsystems, and subsystems to make up systems. The variables of a subsystem can be included on a component-by-component basis in the definition of the subsystems. Models from other environments including MATLAB or FORTRAN can also be included in an iSCRIPT project.

The elements (objects) and their governing rules are illustrated in Table 1.

Table 1. Model Elements of the iSCRIPT Implementation

Major Elements		
Element	**Parent**	**Rules**
Components	Sub-Systems	Instances of components may belong to sub-systems or systems
Sub-Systems	Systems	Instances of sub-systems may belong to systems
Systems		

Sub-Elements		
Element	**Parent**	**Rules**
Variables	Component	▪ Must belong to a component ▪ Can be assigned values in a model ▪ Must be unique within components
Models	Component	▪ Specified as a script and referencing major elements and variables ▪ May be an external program called from iSCRIPT
Coupling Models	Sub-system/ System	▪ Specified as a script and referencing major elements and variables
Objective	Sub-system/ System	▪ Specified as a script and referencing major elements and variables

In addition, to the above, each element consists of properties and methods which allow them to be processed easily as objects in an iSCRIPT implementation. Most prominent of the properties are those for the Variable. This object has properties that include name, type, value, units etc. as illustrated in Table 2.

Table 2. Properties of the Variable Object

Property		**Example**
Name	*Usual set of variable properties in traditional programming environments*	T
Type		Real
Dimensions (scalars are dim 0)		0
Value		Afterburner.T = 2500
Parent Component	*Additional set of variable properties in iSCRIPT*	Afterburner
Default value		3200 °R
Upper and Lower Bounds		1000°R - 3600°R
Engineering unit		°R

As a result of the properties assigned to variables in iSCRIPT, many of the system constraints are automatically implemented in the variable declaration in iSCRIPT. In addition, iSCRIPT can potentially support unit conversion and unit integrity check in a component or system model.

A. The iSCRIPT Modeling Tool

Systems may be modeled in iSCRIPT by specifying equations that describe the system's components. In this sense, components could be developed in several script files which may include several variables, a main script segment, and several subprograms. iSCRIPT has all the features of a programming language environment including decision structures, loop elements, array variables, and subprogram units.

B. Optimization Procedure

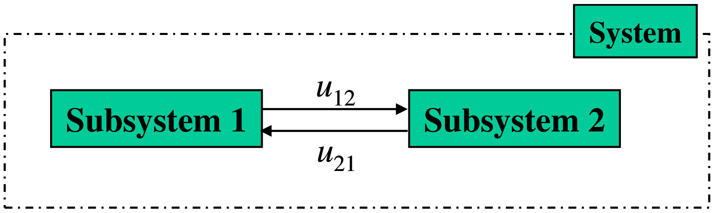
Optimization is accomplished using a genetic algorithm (GA) program based on the procedure developed at LENI, France.[12,13] The algorithm is multi-objective and is based on the clustering pareto evolutionary algorithm (CPEA). Aspects of this algorithm distinct from traditional GA procedures include the direct handling of continuous variables (rather than converting them into binary form) and the unbounding of the population size. The use of continuous variables required a blending of variables during the evolutionary cross-over of two individual realizations is also a feature of the procedure allowing robust handling of real variables as well as integer or Boolean quantities. The unbounded size of the population allows the population to grow as desired to capture the problem space, within some computational limit. A clustering procedure is used to ensure that all local optima are captured.

In the current implementation of GA, the focus is on robustness since this paper is based on a single objective. Therefore, the various parameters that influence the performance of the algorithm were reduced to five variables including the number of initial sample evaluations, maximum population size, maximum number of generations, mutation frequency, and a convergence parameter. Default values that work well for most problems were also set for these variables.

C. The Iterative Local/Global Optimization (ILGO) Procedure

The iterative local/global optimization procedure[1,5,12,13,17,18] avoids the need for nested optimization by utilizing the gradient or response of each subsystem to the variation of the coupling functions.

Consider a system decomposed into two subsystems, with variables, $\bar{X}_1, \bar{X}_2$, and the model equations:



Subsystem 1	**Subsystem 2**	**Coupling function**
Models: $H_1(\vec{X}_1)=0$	Models: $H_2(\vec{X}_2)=0$	$\xi=\xi(u_{12})$
Constraints: $G_1(\vec{X}_1)\le 0$	Constraints: $G_2(\vec{X}_2)\le 0$	$\psi=\psi(u_{21})$
$\vec{X}_1=[X_1,X_2,\ldots,X_n,\xi,\psi]$	$\vec{X}_2=[X_{n+1},\ldots,X_m,\xi,\psi]$	$\vec{X}$ may include mixed integers and binaries

Figure 2. ILGO Procedure in iSCRIPT illustrated using two subsystems

and objective function

$$f=f(Z,\xi,\psi)$$

where $Z=Z_1\cup Z_2$, $\vec{Z}_1=\{\vec{X}_a,\vec{X}_a\in\vec{X}_1\}$, $\vec{Z}_2=\{\vec{X}_b,\vec{X}_b\in\vec{X}_2\}$ and $Z_1\cap Z_2=0\;\forall\, i\neq j$

Assuming the coupling functions ξ, ψ between the subsystems, the overall objective becomes:

Minimize $f=f_1(\vec{Z}_1,\xi,\psi)+f_2(\vec{Z}_2,\xi,\psi)$

w.r.t. $\bar{X}_1$

$$\vec{H}_1(\vec{X}_1)=0$$
$$\vec{G}_1(\vec{X}_1)\le 0$$

and $\bar{X}_2$

$$\vec{H}_2(\vec{X}_2)=0$$
$$\vec{G}_2(\vec{X}_2)\le 0$$

The overall optimization can then be completed in two steps, one at the subsystem level and the other at the overall system level. The subsystem level optimization consists of two optimization tasks resulting in two local optimum values f_1^* and f_2^*, where:

$$f_1^*=f_1\left(\vec{Z}_1^*,\xi_1^*,\psi_1^*\right)\quad\text{and}\quad f_2^*=f_2\left(\vec{Z}_2^*,\xi_2^*,\psi_2^*\right)$$

Note that initially, ξ_1^*, ψ_1^*, may not coincide with ξ_2^*, ψ_2^* at the realizations corresponding to f_1^* and f_2^* for both systems.

The system level optimization consists of one task:

Optimize $f = f_1^*(\xi,\psi) + f_2^*(\xi,\psi)$ subject to ξ_1, ψ_1.

Shadow functions of the objective function at the subsystem level are used at the system level to accomplish an overall optimum $f^{**} = f^*(\xi^*,\psi^*)$ as follows:

$$[\xi,\psi]^{l+1} = [\xi,\psi]^l + \begin{bmatrix} \frac{\partial f_1^*}{\partial \xi}(\xi,\psi) & \frac{\partial f_1^*}{\partial \psi}(\xi,\psi) \\ \frac{\partial f_2^*}{\partial \xi}(\xi,\psi) & \frac{\partial f_2^*}{\partial \psi}(\xi,\psi) \end{bmatrix}^l \begin{bmatrix} \Delta\xi \\ \Delta\psi \end{bmatrix}^l .$$

D. Automatic Parallelization

iSCRIPT optimization tasks are automatically parallelized based on a fine-grained, self-scheduling algorithm. In the formulation, the optimization is considered as a set of evaluations of several viable realizations (individuals) based on an objective function or multiple objective functions. This consideration is valid for virtually all optimization methods including gradient-based (in which the individuals are calculated and used to compute a gradient), or genetic algorithm (in which the individuals are members of a population of realizations which are selected to assure better individuals as the calculations proceed).

Due to its hierarchical, object-oriented formulation, iSCRIPT groups an optimization task according to sub-components of the component of the system being optimized. Each evaluation is simply an instance of a component evaluation task (called Component.Execute in iSCRIPT). The evaluations are scheduled using the self-scheduling algorithm depicted below.

Consider the optimization of an objective $f(\bar{\mathbf{u}})$
based on a model $\mathbf{H}(\bar{\mathbf{U}}) = 0$, $\mathbf{G}(\bar{\mathbf{U}}) \le 0$, $\bar{\mathbf{u}} \in \bar{\mathbf{U}}$
and a set of optimization variables, $\bar{\mathbf{u}}$

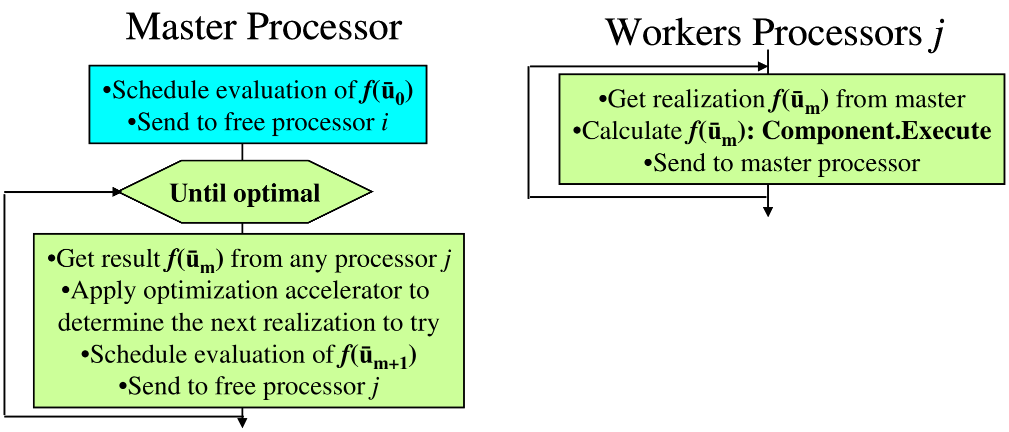


Figure 3. Algorithm of the Automatic Parallelization Procedure in iSCRIPT

The self-scheduling algorithm illustrated in Figure 3 consists of one master processor whose task is simply to schedule the execution of the model (representing the system or component) for any other available processor. Each free processor completes the model evaluation based on $\mathbf{H}(\bar{\mathbf{U}}) = 0$, $\mathbf{G}(\bar{\mathbf{U}}) \le 0$ and sends the result of the evaluation (in terms of the objective function) to the master processor. The master processor tags this processor as free, computes a gradient or determines based on genetic algorithm operations which new set of variables may perform better and assigns this next task to a free processor. The process continues until the optimization is complete. This implementation is completely transparent to the user. In other words, the user is relieved of the task of parallelizing their own model, as would be necessary in traditional programming environments.

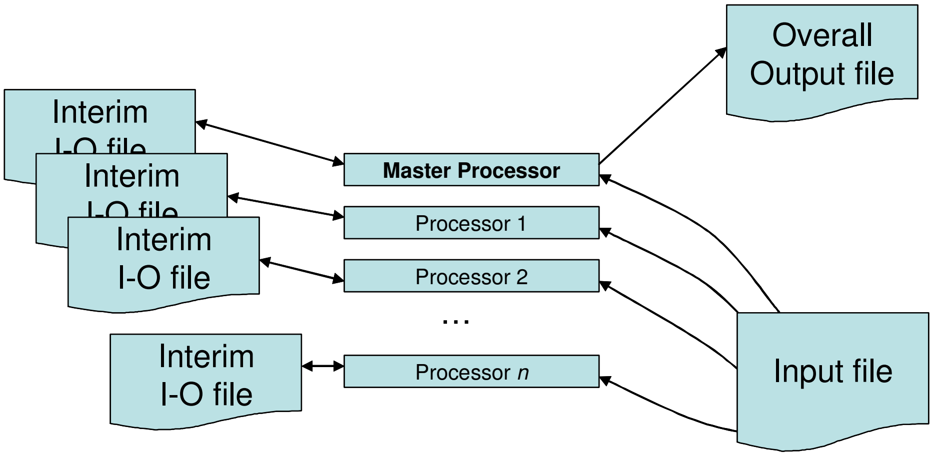


Figure 4. Parallel Input-Output Procedure for an Optimization Task in iSCRIPT

In order for the above procedure to be compatible with models containing input-output (I/O) commands, parallel I/O procedure had to be implemented in iSCRIPT. The procedure for parallel I/O is based on a conceptual design that considers optimization as one task (been completed in parallel). As a result, any overall output is completed only once (for the optimized realization) by the master processor as illustrated in Figure 4. However, if the task requires that input be read from a file, all processors evaluating a realization accesses this input file in parallel. In figure 4, interim I/O files are input-output files generated in the process of evaluating the specific realization by a processor. This includes files used to exchange parameter sets and results with external executables when third-party programs are used in evaluating the model. These files are unique to each processor in the modeling of each realization.

IV. Formulation

The objective is to optimize aircraft using exergy and compare the results with that using gross take-off weight as an objective.

A. Decomposition

As mentioned earlier in this paper, we have adopted the decomposition method used in Munoz and von-Spakovsky et. al.[5] The subsystems include environmental control (ECS), a fuel loop (FLS), a vapor compression/PAO loop (VC/PAOS), a propulsion (PS), and an airframe - aerodynamics (AFS-A). To these we have added an oil loop subsystem (OLS), central hydraulic subsystem (CHS), flight control subsystem (FCS) and an electrical subsystem (ES), which were not present in the model of Munoz and von-Spakovsky et. al.[5] The subsystem *coupling functions* used for the iterative local-global optimization (ILGO) are those developed by von Spakovsky, Muñoz, and Rancruel;[1,5,8-11] to which we have added coupling functions for the OLS, CHS, FCS, and ES; to account for system-level interactions between components and/or subsystems.

The variables, model and constraints for each subsystem may be found in various references. [1,5,12,13,17,18] However, we illustrate the modeling requirement using the environmental control subsystem (ECS) which is illustrated in Figure 5.

Table 3. Supersonic vehicle system-level coupling function variables and fixed parameters

Subsystem	Synthesis/Design: Coupling Function Variable	
Propulsion (PS)	W_{PS}	Weight of PS
Fuel loop (FLS)	W_{FLS}	Weight of FLS
	D_{FLS}	FLS momentum drag
	$\dot{E}_{FLS}$	FLS power requirement

Vapor compression/ PAO loops (VC/PAOS)	$W_{VC/PAOS}$	Weight of VC/PAOS
	$D_{VC/PAOS}$	VC/PAOS momentum drag
	$\dot{E}_{VC\,/\,PAOS}$	VC/PAOS power requirement
	$\dot{Q}_{VC\,/\,PAOS}$	VC/PAOS heat rejection
Environmental control (ECS)	W_{ECS}	Weight of ECS
	D_{ECS}	ECS momentum drag
	$\dot{m}_{ECS}$	ECS bleed air flow rate requirement
Airframe (AFS-A)	W_{AFS}	Weight of AFS
	D_{AFS}	AFS drag
Oil loop subsystem (OLS)	W_{OLS}	Weight of OLS
	$\dot{Q}_{OLS\,/\,FLS}$	OLS/FLS heat rejection
	$\dot{E}_{OLS}$	Oil loop power requirement
Subsystem	**Fixed Parameter Coupling Function**	
Permanent payload (PPAYS)	W_{PPAYS}	Weight of PPAYS
Expendable payload (EPAYS)	W_{EPAYS}	Weight of EPAYS
Flight Control (FCS)	W_{FCS}	Weight of FCS
Central Hydraulic (CHS)	W_{CHS}	Weight of CHS
Electrical (ES)	W_{EGS}	Weight of ES

Figure 5 shows the schematic of a typical air bootstrap environmental control subsystem used for cooling the cabin and avionics of high performance supersonic aircraft. It is for this type of subsystem that the list of synthesis/design and operational variables given in Table 4 has been developed.

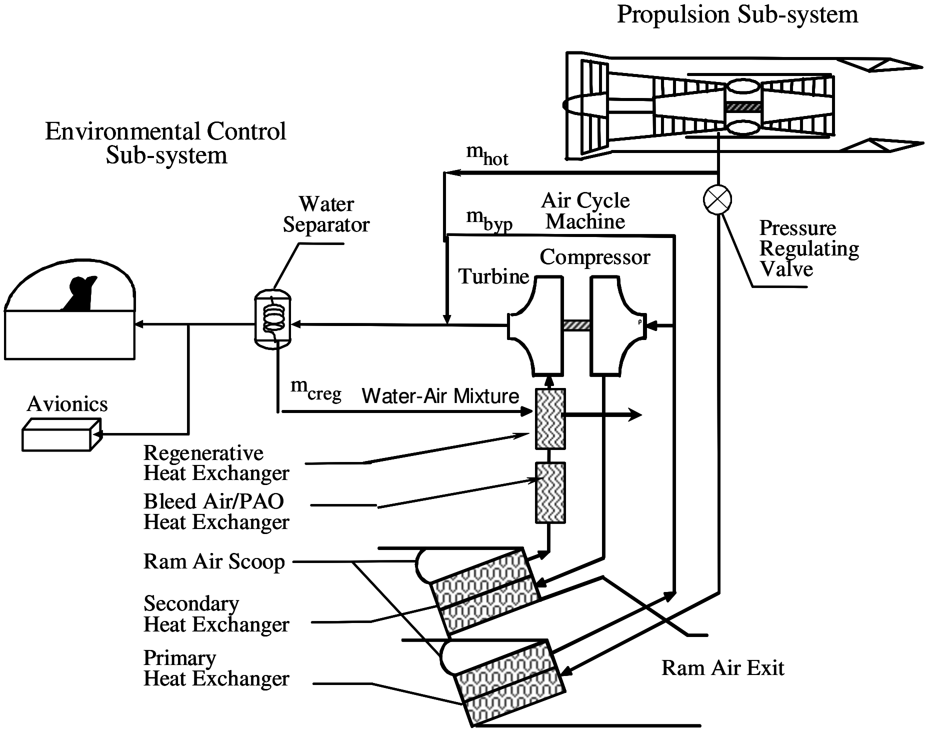


Figure 5. Supersonic vehicle ECS schematic.

Furthermore, in addition to design decision variables, Table 4 lists a number of synthesis variables, which take on only integer or binary values (e.g., the variables Reg_1 and Reg_2 and Fin_{hot} and Fin_{cold}) and, thus, represent configurational changes at the component level (e.g., the type of fin geometry to use) or at the subsystem level (e.g., the placement of the secondary regenerative heat exchanger within the ECS configuration).

Also, note that Table 4 lists a set of dependent variables for ECS. This list is not exhaustive but simply provides an example of some of the many dependent variables for which values can be determined once the values for the decision (independent) variables have been calculated.

Details of the subsystem model, other subsystem variables, constraints and models can be found in the References. [1,5,8-11,14]

Table 4. Supersonic vehicle ECS component-level synthesis/design and operational decision variables, dependent variables, and constraint limits.

Component	**Synthesis/Design Decision Variable**			**Constraint**
Primary and secondary heat exchangers	L_c	Cold side length (m)		$0.06<L_c<0.9$
	L_h	Hot side length (m)		$0.5<L_h<0.9$
	L_n	Non flow length (m)		$0.5<L_n<0.9$
Air cycle machine	PR_{cp}	Compressor design pressure ratio		$1.8<PR_{cp}<3.0$
	PR_{tb}	Turbine design pressure ratio		$PR_{tb}<12$
First and second regenerative heat exchangers	L_c	Cold side length (m)		$0.15<L_c<0.3$
	L_h	Hot side length (m)		$0.3<L_h<0.5$
	L_n	Non flow length (m)		$0.3<L_n<0.5$
	Reg_1 Reg_2	Existence-nonexistence of regenerative heat exchanger in configuration		$Reg_1, Reg_2=0,1$ $Reg_1+Reg_2=1$
Ram air inlet, outlet	A_1, A_2	Areas of inlet, outlet (cm^2)		$120< A_1, A_2<220$
Primary and secondary heat exchanger fin type: hot and cold sides[††, ‡‡]	Fin_{hot} Fin_{cold}	Fin No.	Surface designation	Re_{max}
		1	¼(s)-11.1	8000
		2	1/8-15.2	6000
		3	1/8-13.95	6000
		4	1/8-15.61	6000
		5	1/8-19.86	5000
		6	1/9-22.68	5000
		7	1/9-25.01	4000
		8	1/9-24.12	4000
		9	1/10-27.03	4000
		10	1/10-19.35	4000
Component	**Operational Decision Variable[§§]**			**Constraint**
Pressure regulating valve	PR_{vv}	Pressure setting		$PR_{vv}<6.0$
Low pressure bleed port	BP_{low}	Low pressure bleed port[***]		$BP_{low}=0,1$
High pressure bleed port	BP_{high}	High pressure bleed port[5]		$BP_{high}=0,1$
Splitter	m_{byp}	Bypass air flow rate		$m_{byp}<0.2$ kg/s
Bleed port	m_{hot}	Hot air flow rate		$m_{hot}<0.2$ kg/s
Regenerative heat exchanger	m_{creg}	Cold air flow rate		$m_{creg}<0.2$ kg/s

---

[††] Discrete variable.

[‡‡] The plate thickness is 0.254 mm.

[§§] Each operational decision variable is unique to a particular off-design mission segment.

[***] Binary variable: 0 means no bleed air is taken from the bleed port.

Component	Dependent Variable		Constraint
Cold and hot sides of heat exchangers	Re_c[†††]	Reynolds number, cold air side	$R_{ec/} R_{emax} <1$
	Re_h[6]	Reynolds number, hot air side	$R_{eh/} R_{emax} <1$
Cabin and avionics	T_{cold}[6]	Cooling air temperature	$\lvert T_{cold}-T_{sched}\rvert < 3$
	P_{cold}[6]	Cooling air pressure	$P_{cold} = P_{sched}$
	m_{cold}[6]	Cooling air flow rate	$m_{cold} = m_{sched}$
ACM	W_{cp}, W_{tb}[6]	Compressor and turbine work	$W_{cp}=W_{tb}$

IV. Optimization of the Advanced Tactical Fighter

The exergy-based objective function for the system modeled above may be expressed as:

Minimize

$$Ex_{PS}(\vec{x}_{PS}) + Ex_{ECS}(\vec{x}_{ECS}) + Ex_{FLS}(\vec{x}_{FLS}) + Ex_{AFS-A}(\vec{x}_{AFS-A}) + Ex_{VCPAOS}(\vec{x}_{VCPAOS}) + \\ + Ex_{OLS}(\vec{x}_{OLS}) + Ex_{FCS}(\vec{x}_{FCS}) + Ex_{CHS}(\vec{x}_{CHS}) + Ex_{ES}(\vec{x}_{ES}) \tag{21}$$

w.r.t. a set of design and operational decision variables, $\vec{x}_{subsystem\_i}$ (21a)

subject to a set of equality constraints (PS, ECS, FLS, AFS-A, VCPAOS, OLS, FCS, CHS, and ES models; mission) and inequality constraints. (21b)

The weight-based objective may be expressed as:

Minimize

$$W_{PS}(\vec{x}_{PS}) + W_{ECS}(\vec{x}_{ECS}) + W_{FLS}(\vec{x}_{FLS}) + W_{AFS-A}(\vec{x}_{AFS-A}) + W_{VCPAOS}(\vec{x}_{VCPAOS}) + \\ + W_{OLS}(\vec{x}_{OLS}) + W_{FCS}(\vec{x}_{FCS}) + W_{CHS}(\vec{x}_{CHS}) + W_{ES}(\vec{x}_{ES}) \tag{22}$$

w.r.t. a set of design and operational decision variables, $\vec{x}_{subsystem\_i}$ (22a)

subject to a set of equality constraints (PS, ECS, FLS, AFS-A, VCPAOS, OLS, FCS, CHS, and ES models; mission) and inequality constraints. (22b)

Only one mission segment is considered in the current work: the combat air patrol (Table 5).

Table 5. Description of mission segments selected for demonstrating the iSCRIPT software tool.

Mission Segment	Description
Combat Air Patrol	Perform combat air patrol at 9,150m and Mach Number 1.6 for 20 min

A. Results of Exergy-Based Analysis

The exergy based analysis resulted in an optimum vehicle with improved take-off weight, drag, and fuel consumption were compared with the initial design as well as the optimized aircraft based on the gross take-off weight as an objective. The analysis was completed in three stages:

1. Development of each subsystem independently. During this stage, the values of the coupling functions were set to a "reasonable" fixed value based on previous work.[5]
2. Optimization of each subsystem independently. However, the values of the coupling functions were set from results of the performance analysis of other coupling function when the values are

---

[†††] This variable takes different values at different mission segments.

determined by those subsystems. For instance the computed weight of the other subsystems and a coupling variable is available in the AFS-A model once the other subsystems have been evaluated. Two objectives were considered:

   a. Gross take-off weight (*W*)
   b. Total exergy destruction (*Ex*)

3. Optimization of the entire system using ILGO with exergy as an objective.

The results for stage 1 provide initial values that are contrasted with the results in stage 2 for the gross take-off weight and exergy objectives. These results are presented in Table 6 through 12 for each subsystem.

Table 6. Optimum Values of the AFS-A Decision Variables.

Component	**Decision Variables**		**Initial Values**	**Optimized at Subsystem Level**		**Optimized at System Level**
				***W* objective**	***Ex* objective**	***Ex* objective**
Wing	AR	Aspect ratio	3.05	3.86	3.852	3.852
	t/c	Thickness ratio	0.07711	0.0642	0.0637	0.0640
	λ	Taper ratio	0.254	0.98	0.982	0.984
	Λ	Sweep angle	31	35.26	34.263	35.04
	$Sref$	Reference area (ft^2)	381	379.52	379.14	379.52
Tail	T_AR	Tail aspect ratio	5	5.12	5.116	5.107
	T_t/c	Tail thickness ratio	0.1206	0.0622	0.0624	0.0619
	T_λ	Tail taper ratio	0.46	0.498	0.496	0.495
Subsystem	**Figure of Merit**					
AFS	W_{empty}	**Aircraft empty weight (kg)**	**14,681**	**13,948.4**	**13,942.7**	**10247.12**
	Ex_{AFS}	**Exergy loss in AFS (kJ/s)**	**4972.72**	**4921.4**	**4834.5**	**4523.4**

Table 7. Optimum Values of the ECS Decision Variables.

Component	**Decision Variables**		**Initial Values**	**Optimized at Subsystem Level**		**Optimized at System Level**
				***W* objective**	***Ex* objective**	***Ex* objective**
Primary heat exchanger	L_c	Cold side length (m)	0.06	0.0656	0.0657	0.0642
	L_h	Hot side length (m)	0.8	0.542	0.543	0.538
	L_n	Non flow length (m)	0.6	0.71100	0.712	0.712
	Fin_{cold}	Fin type cold stream	11	11	11	11
	Fin_{hot}	Fin type hot stream	11	14	14	14
Secondary heat exchanger	L_c	Cold side length (m)	0.06	0.352	0.344	0.343
	L_h	Hot side length (m)	0.9	0.451	0.471	0.449
	L_n	Non flow length (m)	0.6	0.871	0.868	0.867
	Fin_{cold}	Fin type cold stream	11	12	12	12
	Fin_{hot}	Fin type hot stream	11	14	14	14
Regenerative	Reg_1	Regenerative HX	1	0	0	0

HX		Flow on/off				
Air cycle machine	PR_{cp}	Compressor design pressure ratio	2.798	2.243	2.244	2.214
Bleed air / hot PAOS heat exchanger	L_c	Cold side length (m)	0.40	0.1931	0.193	0.194
	L_h	Hot side length (m)	0.50	0.4872	0.487	0.487
	L_n	Non flow length (m)	0.30	0.3121	0.312	0.304
	Reg_2	Bleed/PAO HX Flow on/off	0	1	1	1
	Fin_{cold}	Fin type cold stream	11	12	12	12
	Fin_{hot}	Fin type hot stream	11	15	15	15
Component	**Operational Decision Variables**					
Pressure regulating valve	PR_{vv}	Pressure setting	3.0978	2.6288	2.631	2.624
Splitter	m_{byp}	Bypass mass flow rate	0.0385	0.0290	0.0291	0.0274
Bleed port	m_{hot}	Hot mass flow rate	0.0107	0.0012	0.0011	0.0002
Regenerative heat exchanger	m_{creg}	Cold air flow rate	0.0563	0.0483	0.0482	0.0473
Subsystem	**Figure of Merit**					
ECS	W_{ECS}	**Weight of entire subsystem (kg)**	**365.83**	**165.82758**	**155.82758**	**132.276**
	Ex_{ECS}	**Exergy loss in AFS (kJ/s)**	**801.73**	**423.38**	**425.54**	**382.2841**

Table 8. Optimum Values of the PS Decision Variables.

Component	**Decision Variables**		**Initial Values**	**Optimized at Subsystem Level**		**Optimized at System Level**
				***W* objective**	***Ex* objective**	***Ex* objective**
Fan	α	Fan bypass ratio	0.1	0.362	0.364	0.361
	PR_{fan}	Fan design pressure ratio	3.8	3.73	3.701	3.72
Compressor	PR_{comp}	Compressor design pressure ratio	17.0	14.2	13.82	14.1
Turbine	T_B	Turbine inlet temperature (^{o}R)	3200	2725.6	2711.4	2711.4
Afterburner	T_{aft}	Afterburner Temperature (^{o}R)	3600	2267.7	2235.2	2259.6
Component	**Figure of Merit**					
Engine	Ex_{PS}	**Exergy loss in PS (kJ/s)**	**80,904.90**	**74456.25**	**64456.25**	**72409.03**
	W_{PS}	**Engine weight (kg)**	**3976.225**	**3213.87**	**2914.399**	**3205.26**

Table 9. Optimum Values of the FLS Decision Variables.

Component	**Decision Variables**		**Initial Values**	**Optimized at Subsystem Level**		**Optimized at System Level**
				***W* objective**	***Ex* objective**	***Ex* objective**
Fuel / Ram air heat exchanger	L_c	Cold side length (m)	0.0698	0.050	0.047	0.0496
	L_h	Hot side length (m)	0.5176	0.2176	0.2158	0.2167
	L_n	Non flow length (m)	0.506	0.2061	0.1674	0.2011
	Fin_{cold}	Fin type cold stream	20	11	11	11
	Fin_{hot}	Fin type hot stream	14	12	12	12
Ram Air Inlet	A_i	Area of inlet, outlet (m^2)	0.012	0.0118	0.0113	0.0112
Component	**Figure of Merit**					
Fuel Loop Subsystem	Ex_{Fcs}	**Exergy loss in FLS (kJ/s)**	**66.489**	**14.098**	**2.67**	**3.00682**
	W_{FLS}	**FLS weight (kg)**	**379.02**	**248.188**	**389.262**	**392.85**

Table 10. Optimum Values of the VCPAOS Decision Variables.

Component	**Variables**		**Initial**	**Optimized at Subsystem Level**		**Optimized at System Level**
				W objective	**Ex objective**	**Ex objective**
Bleed PAO heat transfer	L_c	Cold side length (m)	0.083	0.099	0.285	0.172
	L_h	Hot side length (m)	0.756	0.159	0.220	0.222
	L_n	Non flow length (m)	0.420	0.048	0.299	0.189
	Fin_{cold}	Fin type cold stream	15	11	11	11
	Fin_{hot}	Fin type hot stream	20	12	11	11
Ram PAO heat transfer	L_c	Cold side length (m)	0.249	0.035	0.216	0.203
	L_h	Hot side length (m)	0.312	0.183	0.437	0.425
	L_n	Non flow length (m)	0.814	0.192	0.509	0.482
	Fin_{cold}	Fin type cold stream	16	11	12	12
	Fin_{hot}	Fin type hot stream	18	12	11	11
Condenser	L_c	Cold side length (m)	0.143	0.209	0.216	0.201
	L_h	Hot side length (m)	0.755	0.140	0.437	0.418
	L_n	Non flow length (m)	0.757	0.079	0.507	0.479
	Fin_{cold}	Fin type cold stream	12	12	11	11
	Fin_{hot}	Fin type hot stream	12	11	12	12
Evaporator	L_c	Cold side length (m)	0.734	0.055	0.246	0.225
	L_h	Hot side length (m)	0.165	0.205	0.159	0.152
	L_n	Non flow length (m)	0.730	0.114	0.288	0.278
	Fin_{cold}	Fin type cold stream	12	11	11	11
	Fin_{hot}	Fin type hot stream	12	11	11	11

Component	**Figure of Merit**					
Environmental Control Subsystem	W_{VCPAOS}	**Weight of entire subsystem (kg)**	**466.74**	**121.18**	**207.94**	**163.22**
	Ex_{VCPAOS}	**Exergy loss in VCPAOS (KJ/s)**	**57.14**	**44.35**	**16.44**	**19.42**

Table 11. Optimum Values of the OLS Decision Variables.

Component	**Decision Variables**		**Initial Values**	**Optimized at Subsystem Level**		**Optimized at System Level**
				***W* objective**	***Ex* objective**	***Ex* objective**
Oil loop	$\dot{m}$	Flow rate (kg/s)	0.9	0.9	0.02952	0.0423
Component	**Figure of Merit**					
OLS	Ex_{PS}	**Exergy loss in PS (kJ/s)**	**1.843**	**1.843**	**0.0605**	**0.0824**
	W_{OLS}	**OLS weight (kg)**	**34.86**	**34.86**	**34.86**	**34.86**

The overall gross take-off weight are respectively 13,948.4 and 13,942.7 kgs when weight and exergy are the objectives. However, even though exergy and weight objectives both produce smaller aircrafts compared to the initial design (14,681 kg), the weight is distributed in different ways. For the exergy objective model, for instance, the fuel loop subsystem is heavier and greater cooling capacity is derived from the FLS. In addition, the engine is smaller for the exergy-based objective because of the lower drag. As a result, exergy-based design produces a better plane which is about as heavy but consumes less fuel. The results for weight, drag, and thrust specific fuel consumption, TSFC are presented in Figure 6.

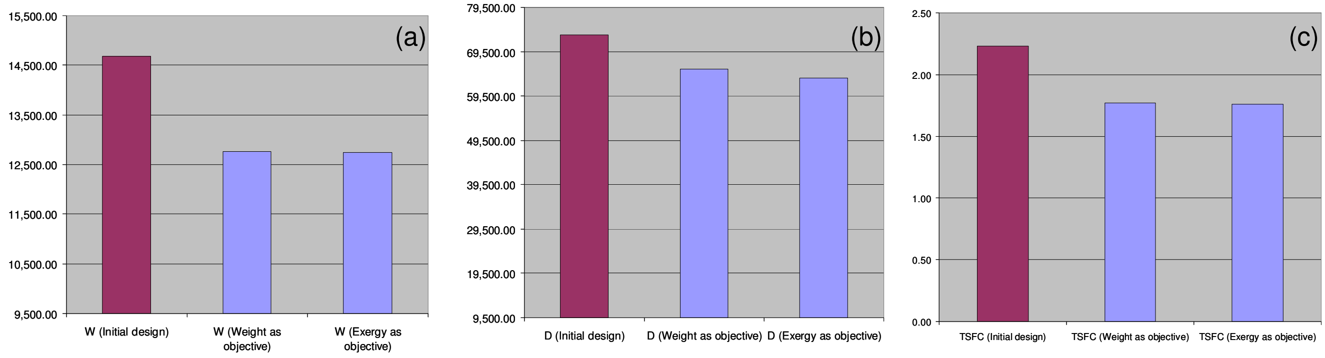


Figure 6. Comparison of results for weight and exergy as objective functions (a) gross take-off weight, (b) total drag, (c) thrust specific fuel consumption TSFC.

In addition to the performance of the exergy-based objective compared to weight as an objective, analysis of the details of each component in terms of the exergy destroyed may additionally highlight components where improvements are needed.

For an example, refer to the plot in Figure 7 of exergy destruction for the PS subsystem. Using information such as those illustrated by the plot showing the relationship between several decision variables including fan bypass ratio and compressor ratio, the designer is able to understand why the optimization drove the vehicle design to the result produced. In addition, the designer may use this information to determine what additional variables may be provided degrees of freedom at precisely those sites where the largest inefficiencies are pinpointed by the exergy based optimization. Such information is simply not available from a conventional 1st Law approach.

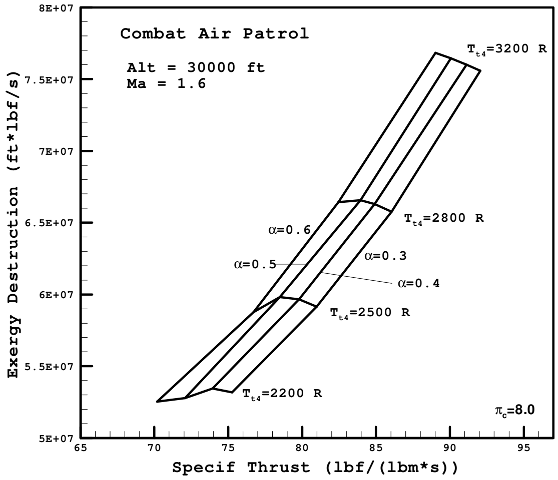


Figure 7. Total exergy destruction for the aircraft during the combat air patrol mission segment versus specific thrust, turbine inlet temperature (T_4), fan by-pass ratio (α), and a compressor ratio (π_c) of 8.

V. Conclusions

In this paper, procedures based on exergy-analysis for designing and optimizing aircraft are described. The procedures are packaged into the software tool iSCRIPT. iSCRIPT contains full features of a programming language as well as powerful, robust genetic algorithm optimization programs, procedures for complete system-level optimization based on the integrated local-global optimization (ILGO) procedure, and can automatically deploy an optimization task in parallel.

An advanced tactical aircraft system has been used to illustrate the concept. Optimizations using weight and exergy as the objective functions are compared. The superiority of the exergy-based procedure lies in the lower computational requirement and the convenience of analysis of a single objective function. In addition, the exergy-based procedure provides a loss map for various components in the system. This is invaluable as it assists the designer in the design/optimization process.

References

[1]Munoz, J. R.,, 2002, "A Decomposition Strategy Based on Thermoeconomic Isolation Applied to the Optimal Synthesis/Design and Operation of an Advanced Fighter Aircraft System," M..S. thesis, Advisor: M. R. von Spakovsky, Mechanical Engineering Dept., Virginia Polytechnic University, Blacksburg, VA February 2002.

[2]Alexandrov, N. M., Kodiyalam, S., 1998, "Initial Results of an MDO Method Evaluation Study," AIAA Paper: AIAA-98-4884.

[3]Le, A., Gray, K., Baker, M. L., 2003, "Building the Aerodynamics Module for the Integrated Hypersonic Aeromechanics Tool," AIAA Paper.

[4]Alabi, K., Ladeinde, F., von Spakovsky, M.R., Moorhouse, D., Camberos, J., "Assessing CFD Modeling of Entropy Generation for the Air Frame Subsystem in an Integrated Aircraft Design/Synthesis Procedure," AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, Jan 2006.

[5]Munoz, J. R., von Spakovsky, M. R., 2001, "The Application of Decomposition to the Large Scale Synthesis/Design Optimization of Aircraft Energy Systems," International Journal of Applied Thermodynamics, June, Vol. 4, No. 2.

[6] Brewer, K. M., 2005, “Exergy Methods for Mission-Level Analysis and Optimizatio of Generic Hypersonic Vehicle Concepts,” M.S. thesis, Advisor: M. R. von Spakovsky, Mechanical Engineering Dept., Virginia Polytechnic University, Blacksburg, VA September 2005.

[7] Butt, J., 2005, “Exergy Analysis and Optimization Applied to an Aircraft System with Wing Morphing,” M.S. thesis, Advisor: M. R. con Spakovsky, Mechanical Engineering Dept., Virginia Polytechnic University, Blacksburg, VA September 2005.

[8] Munoz, J. R., von Spakovsky, M. R., 2001, “The Application of Decomposition to the Large Scale Synthesis/Design Optimization of Aircraft Energy Systems,” International Journal of Applied Thermodynamics, June, Vol. 4, No. 2.

[9] Munoz, J. R., von Spakovsky, M. R., 2001, “A Decomposition Approach for the Large Scale Synthesis/Design Optimization of Highly Coupled, Highly Dynamic Energy Systems,” International Journal of Applied Thermodynamics, June, Vol. 4, No. 2.

[10] Rancruel, D. F., von Spakovsky, M. R., 2004, “Use of a Unique Decomposition Strategy for the Optimal Synthesis/Design and Operation of an Advanced Fighter Aircraft System,” 10th AIAA/ISSMO Multi-disciplinary Analysis and Optimization Conference, Aug 30 – Sept 1, Albany New York.

[11] Rancruel, D. F., von Spakovsky, M. R., 2004, “Decomposition with Thermodynamic Isolation Applied to the Optimal Synthesis/Design of an Advanced Fighter Aircraft System,” 10th AIAA/ISSMO Multi-disciplinary Analysis and Optimization Conference, Aug 30 – Sept 1, Albany New York.

[12] Molyneaux, A., “A Practical Evolutionary Method for the Multi-Objective Optimization of Complex Energy Systems, including Drivetrains. M.Sc. Thesis, Ecole Polytechnique Federale De Lausanne, EPFL, 2002.

[13] Leyland G. B., “Multi-Objective Optimization Applued to Industrial Energy Problems”, Ph.D Thesis. Ecole Polytechnique Federale De Lausanne, EPFL, 2002.

[14] Alabi, K., Ladeinde, F., vonSpakovsky M., Moorhouse, D., Camberos, J. 2006. Irreversible Thermodynamic Modeling of the Air Frame Subsystem as part of an Integrated Aircraft Design/Synthesis Procedure. Int. Journal of Thermodynamics, Dec 2006.