CONTROL INTEGRATION PROCESS IN AIRCRAFT SYSTEMS DEVELOPMENT

Hugh H.T. Liu, Anton de Ruiter
Institute for Aerospace Studies, University of Toronto
4925 Dufferin Street, Toronto,Ontario, Canada
Contact Author: Professor H.H.T. Liu: liu@utias.utoronto.ca

Keywords: *aircraft systems, control, integration, process*

Abstract

The tasks of controlling aircraft systems are quite challenging. First of all, the increasing level of aircraft systems complexity drives high demand for better "control" to achieve and improve performance, under a more sophisticated configuration. Secondly, there are multiple performance requirements to be achieved by a finite set of controllers. Some of those requirements may even conflict with each other, often leaving the designer a difficult decision to make. Thirdly, the aircraft systems are made up of many subsystems that may interact, such as the flight control systems and the propulsion control systems. Therefore, control for multi-objectives and integration becomes a critical stage in the aircraft systems development process. In this paper, we present a survey of research work in the fields of integrated control and multi-objective control. Further, we propose a uniform presentation to treat these two topics simultaneously, which becomes the starting point of our future work on integrated aircraft systems control for multi-objectives.

1 Introduction

On a modern aircraft, control is almost involved in every system, subsystem, or component. For example, flight control systems enhances the stability and maneuvability [22, 9]; engine control improves propulsion performance [20]; environmental control system provides proper pressure and temperature to the crew, cabin, and avionics instruments. The purpose of the control engineering is to meet the performance specifications through control systems design. Engineering intensive application and the complexity of aircraft systems make it necessary for a systematic design process, where, in simplification, involves requirements definition, design, and integration and testing.

A typical systems engineering process normally starts with requirement definition, to identify the customer requirements, and to translate into technical specifications. One should also define the overall system architecture, and to allocate specifications to appropriate subsystems or components. As well, attention must be paid in requirement verification and validation, to ensure the proposed specifications are complete and accurate enough to reflect the customer requirements, and to ensure compliance with safety or other quality assurance standards, as well as other processes. After the requirements are defined, the design is performed by different teams in their specific disciplinary areas at the subsystem or component level. Assume that extensive testing is also conducted at this level. Then the system engineers will be responsible for integration and testing, to assemble subsystems together, and to verify overall performance. For an aircraft development application, the integration and testing process contains test-rig testing or bench testing, aircraft ground testing, and eventually flight testing. Typically this stage is expensive and time consuming.

In this paper, we address the topic of control and integration process in aircraft systems development. Based on our survey of research and development work in related fields, Two challenges are identified in the areas of integrated control and multi-objective control respectively. Further, we propose a uniform presentation to treat these two topics simultaneously, which becomes the starting point of our future work on integrated aircraft systems control for multi-objectives.

The rest of this paper is organized as follows. In Section 2, the systems development process is briefly introduced and a highlight is given where control and integration task stands in the process. Right after Section 2, detailed discussions of control and integrations are presented in Section 3, followed by the R&D work survey in the areas of integrated control (Section 4) and multi-objective control (Section 5). Further, A uniform presentation to treat these two problems simultaneously is given in Section 6. Finally, our concluding remarks are given in Section 7.

2 Aircraft Systems Design and Development Process

The increasing level of complexity and sophistication of aircraft systems require a systematic development approach, which has become one study topic of a specialized engineering discipline, the systems engineering. The systems engineering process is guided by a number of agencies. In aerospace sector, perhaps the most popular documents include the Society of Automobile Engineers (SAE) ARP 4754: "System Development Processes", ARP 4761: "Safety Assessment Process Guidelines and Methods", the Federal Aviation Authority (FAA) AC 25.1309-1A: "System Design and Analysis", Joint Airworthiness Authority (JAA) AMJ 25.1309: "System Design and Analysis", among others. Recently, the IEEE also published a standard document, IEEE-STD-1220-1998: "IEEE Standard for application and management of the systems engineering process".

No matter what guideline or standard is adopted, generally speaking, the purpose of the systems engineering is to provide a systematic approach for product development and management across its life cycle, from concept to operation and to the retirement. A typical set of phases can be identified according to the product life cycle [1, 19, 20]. A concept phase is to understand the customer's emerging needs and to establish what the system shall be capable of accomplishing. At this phase, typical considerations include the primary role and functions of the required system, the expectations, constraints, operational scenarios, interfaces and boundaries. The result is a documented requirement baseline. At the definition phase, the initial concepts will be developed into a firm definition, to examine the feasibility, to evaluate the risks, and to provide candidates of competing feasible solutions. At the design phase, the competing candidates are assessed and the winning system is selected. Also, the functional analysis is performed to translate the validated requirement baseline into a functional architecture, and further, synthesis into design tasks that can be manufactured. Later, testing and integration phase is to ensure the completeness of the design (verification) and compliance with the original customer's requirements (validation).

The control systems development are not only guided by the systems development process, but also involved in each phase during the development. For example, at the concept or mission phase, requirements must be considered for those quantities required to define flight control activities [19]. As a matter of fact, four main iteration loops can be seen in flight control law development [24]: off-line design (at the concept and design phases), pilot-in-the-loop simulation, iron bird testing, and flight testing (at the testing and integration phase).

During the system development process, the increasing level of system sophistication and increase interrelation of systems is also making the development process more difficult [20], especially in the testing and integration phase. The ability to capture all of the system requirements and interdependencies between systems has to be established at an early stage. Further, the control law complexity is directly related to the complexity of the control task. In fact, today's development of a new civil aircraft program lasts about three years, and the integration and testing phase, including flight tests, will account for more than one year [24]. In the next section, we will highlight some control and integration challenges in the aircraft systems development process, before we focus on two specific challenges.

3 Control and Integration Challenges

The purpose of the control engineering is to meet the performance specifications through control systems design. The success of the control systems will be judged by their compliance to overall performance, after they are implemented and integrated. From the integration point of view, there are several challenges that one may have to face during the control system development.

Requirement Capture. Requirement capture is a key activity in identifying and quantifying all the necessary strands of information which contribute to a complete and coherent system design. Two main methods are commonly used: 1) top-down approach, and 2) bottom-up approach. The top-down approach decomposes the system requirements into smaller functional modules. It is convenient for control system design since some control channels or loops are implemented at a sub-system level. The challenge is how to partition and allocate the top-level requirements into lower level functional specifications without loss of overall performance satisfaction. On the other hand, the bottom-up approach is best applied to systems where the lower level control subsystems may be well understood or be separately manufactured. However, the process of integrating these modules into a higher subset presents difficulties as the interaction between the individual subsystems is not fully understood or documented before they are assembled and integrated for testing.

Simulation. Modern aircraft systems development more and more relies on high-fidelity simulation with hardware-in-the-loop-capability to replace the expensive prototype. This cost-effective "virtual aircraft" solution seems to be trend-setting and is well accepted among airframe manufactures. On the other hand, such high demand and dependence on simulation requires that simulation not only be used for individual design evaluation and assessment, but also be used for integration testing.

Interface Definition. The control systems, such as flight control systems, utility control systems, along with other aircraft systems, may be provided by different vendors and suppliers. Once contractual arrangements are established with all of the other vendors, a full definition of interfaces may be commenced. This task is critical to ensure all individually developed or prototyped components and subsystems can be put together to function as a whole system.

Optimization Process. The control systems development is also a process of selecting among competing feasible solutions, such as different control structures and configurations. In principle, a complete evolution is the result of optimization process.

In summary, the control and integration is important and critical in aircraft systems development process. The challenges remain for the choice of approaches in dealing with performance and configuration, and the usage of the simulation technology. In this paper, we will focus on two specific topics, one is the integrated control, and the other one is the multi-objective control, both are tightly related to the aforementioned process and challenges.

4 Integrated Control

The majority of the occurences in the literature for integrated control systems are motivated by the application of integrated flight and propulsion control because the coupling between the propulsion system dynamics and the airframe dynamics is significant enough that the traditional separate flight control system design and propulsion control system design may no longer be adequate [12]. The integrated flight/propulsion system is viewed as two subsystems, the airframe and the engine respectively, being treated together.

The aircraft input-output at an operating point is given by the LTI system

$$\begin{bmatrix} y_a \\ y_e \end{bmatrix} = \begin{bmatrix} G_a & G_{ae} \\ G_{ea} & G_e \end{bmatrix} \begin{bmatrix} u_a \\ u_e \end{bmatrix} \tag{1}$$

where y is the output, u is the control input and the subscripts a and e represent airframe and engine variables respectively.

There are generally two design approaches from the control integration point of view. The so-called *centralized approach* [12] takes advantage of its explicit consideration of all the possible interactions between subsystems, based on integrated models of the overall system. The centralized control law is defined as

$$\begin{bmatrix} u_a \\ u_e \end{bmatrix} = \begin{bmatrix} K_a & K_{ae} \\ K_{ea} & K_e \end{bmatrix} \begin{bmatrix} y_{ac} - y_a \\ y_{ec} - y_e \end{bmatrix} \tag{2}$$

where y_{ac} and y_{ec} are the airframe and engine commands respectively. The drawback is, however, in practice the implementation may not be possible, or even advisable for a number of reasons. First of all, different subsystems may be designed, constructed and tested by different manufacturers. Design accountability and other commercial issues will often dictate that each party retains control over and takes responsibility for all aspects of the subsystem for which it is responsible. In addition, since the types and levels of control schemes appropriate for different subsystems may vary dramatically, the performance achieved by dedicated sub-controllers may be difficult to match with general centralized design [2].

On the other hand, the so-called *decentralized approach* follows the given hierarchical architecture. The system is broken down into various (possibly interacting) subsystems, and their required dynamic characteristics are defined or derived based on overall system performance requirements. The decentralized control law is defined as

$$\begin{bmatrix} u_a \\ u_e \end{bmatrix} = \begin{bmatrix} K_a & 0 \\ 0 & K_e \end{bmatrix} \begin{bmatrix} y_{ac} - y_a \\ y_{ec} - y_e \end{bmatrix} \tag{3}$$

In other words, we require any overall system to be partitioned, in the sense that separate sub-controllers will be required to address different control specifications in different parts of the system, and integrated in the sense that overall system performance objectives are achieved in the face of interactions between the various subsystems. The advantages of the decentralized approach are obvious [21]: 1) a simple control design is easier to achieve for a specific subsystem or control channel; and 2) the subsystems can be manufactured and tested independently and extensively before being assembled (integrated), which is a preferred choice of manufacturers. However, as we mentioned before, this approach will leave the integration and testing process in an ad hoc manner. Especially when the complexity of a control system grows fast, and the interacting effects (especially negative effects) of subsystems can no longer be ignored since significant deterioration in overall performance may arise from interactions between subsystems.

Another related issue is that a decentralized stabilizing controller may not even exist. The concept of *decentralized fixed modes* is needed for the following discussion. Wang and Davison [33] gave the necessary and sufficient conditions for the existence of a decentralized control law in the following statement: "A stabilizing decentralized dynamic compensator exists if and only if all decentralized fixed modes of the system lie in the open left-half plane."

Schierman and Schmidt [26] show that with a completely decentralized controller (when it exists) for integrated flight and propulsion control, the system may not be able to achieve the performance that would have been achieved if a centralized controller had been used. Obviously, the best possible performance would come from a centralized controller, but given the impracticality of implementing such a controller, a compromise between centralized and decentralized control is sought. This is the subject of the majority of papers describing integrated control. Practical investigations of the implementation of integrated flight and propulsion control have shown improved performance overall [5, 7, 29, 32].

The methods developed in the literature for designing integrated control systems can be divided into three groups: "Centralized Design Partitioning for Decentralized Implementation", "Direct Decentralized Design" and "Decentralized Design via Performance Specifications". All are done in the context of integrated flight and propulsion control.

The main idea of the *Centralized Design Partitioning for Decentralized Implementation* approach is to design a centralized controller with the desired closed-loop properties. The design is then partitioned into possibly lower-order decentralized controllers such that the closed-loop performance and robustness of the decentralized implementation closely match that of the centralized design [11, 13, 23, 27]. The problem can be stated as follows: Given is a centralized controller $K(s)$ such that $u(s) = K(s)e(s)$ where $u(s) = \begin{bmatrix} u_a \\ u_e \end{bmatrix}$ and $e(s) = \begin{bmatrix} e_a \\ e_e \end{bmatrix}$ and $K(s) = \begin{bmatrix} K_a & K_{ae} \\ K_{ea} & K_e \end{bmatrix}$ and a choice of intermediate variables z_{ea} (propulsion quantities affecting airframe, e.g. propulsive forces and moments), find subcontrollers $K^a(s)$ and $K^e(s)$ with $\begin{bmatrix} u_a \\ z_{eac} \end{bmatrix} = K^a(s)e_a$ and $u_e = K^e(s)\begin{bmatrix} e_e \\ e_{ea} \end{bmatrix}$ where $e_{ea} = z_{eac} - z_{ea}$, so that the closed-loop performance and robustness with $K^a(s)$ and $K^e(s)$ closely match that of the closed-loop performance and robustness with the centralized controller $K(s)$. When the subcontrollers are assembled as an equivalent centralized controller $\tilde{K}(s)$, they have the form

$$\tilde{K}(s) = \begin{bmatrix} \tilde{K}_{aa} & 0 \\ \tilde{K}_{ea} & \tilde{K}_{ee} \end{bmatrix}. \tag{4}$$

The subcontrollers are then designed so that $\tilde{K}(s)$ closely matches $K(s)$. A suggested way of doing this is by solving

$$\min_p ||(K(s) - \tilde{K}(p,s)).W(s)||_\infty$$

where p are the parameters of the decentralized controllers. These parameters depend on the synthesis method chosen for designing the decentralized controllers. Garg defines p to be the entries of the matrices in the state-space realization of the controllers. This problem may be solved using a general unconstrained nonlinear optimization algorithm. A good choice for $W(s)$ is $W(s) = G(s)$. Obviously, for good matching, $K_{ae}(s)$ must be small in the centralized controller since $\tilde{K}_{ae}(s) = 0$. It must be noted here that one-way coupling between subsystems has been assumed. The technique can be modified to account for two-way coupling.

Chen and Voulgaris [6] impose an upper triangular structure for the *Direct Decentralized Design*:

$$K(s) = \begin{bmatrix} K_{aa} & K_{ae} \\ 0 & K_{ee} \end{bmatrix} \tag{5}$$

The design is then performed at the centralized level using this controller. They use an H_∞ technique to guarantee that $||T(s)||_\infty \leq \gamma$, where $T(s)$ is the closed-loop transfer function.

Given the lower triangular structure of Garg's assembled decentralized controller, Chen and Voulgaris have assumed coupling in the other direction. However, the idea is clear. The advantage of this approach is that the engine loop is separated (it wouldn't be if coupling were assumed in the other direction). The resulting airframe controller is not decentralized since it depends on engine outputs. This method could be combined with Garg's method, since as mentioned we wish one of the off-diagonal block terms to be small for one-way coupling.

Neighbors and Rock consider a general hierarchical partitioned system for *Decentralized Design via Performance Specifications* [21, 25]. In this approach, G is considered as the main plant and T_j are the subsystems (actuators). All plant-subsystems and subsystem-subsystem coupling signals are contained in the vectors u_c and u. The subsystems are described by $T_j = Q_j + \Delta Q_j$, where Q_j is the nominal subsystem and ΔQ_j is the deviation from the nominal. Defining the matrix

$$\Delta \triangleq \operatorname{diag} \Delta Q_j,$$

The performance specifications on the overall system are assumed to be of the form

$$||M_{\Delta e_i d_i}(j\omega)||_2 < b_{pi}(\omega). \qquad (6)$$

where $M_\Delta = M_{11} + M_{12}\Delta(I - M_{22}\Delta)^{-1}M_{21}$ and $M_{\Delta e_i d_i}$ is the transfer function from d_i to e_i in that matrix. The method then gives specifications on ΔQ_i, that if met, will guarantee that (6) are met. These specifications are of the form

$$||\Delta Q_i(j\omega)||_2 < b_s(\omega).$$

This method has the advantage that the subsystem specifications can be given to the subsystem designers and then the subsystems can be designed independently. The method has the drawback that the specifications must be given in the form (6). Also, since the subsystem specifications are sufficient conditions for overall performance, but not necessary conditions, the design methodology is conservative in nature which could ultimately lead to the unnecessary relaxation of original design specifications.

5 Multi-Objective Control

There appear to be a number of different notions of multiple objective control design in the literature. In one, a controller is sought to satisfy a set of n objectives which may be given as a set of constraints on the closed-loop system, H

$$\phi_i(H) \leq d_i \quad i = 1, ..., n, \qquad (7)$$

where, loosely speaking, ϕ_i is a positive valued function such that the smaller it is, the better it is. A controller that satisfies (7) is said to be feasible. In another formulation, a controller is sought to minimize a multiple number of objective functions, ie. to solve

$$\min\{\phi_i(H)\} \quad i = 1, ..., n. \qquad (8)$$

Or, a controller is sought to minimize a multiple number of objective functions subject to some constraints

$$\begin{array}{lll} \min & \{\phi_i(H)\}, & i \in \mathcal{S} \\ \text{s.t.} & \phi_i(H) \leq d_i, & i \in \mathcal{H} \end{array} \qquad (9)$$

where the constraints corresponding to $i \in \mathcal{H}$ are requirements with numerical bounds that must be met. The requirements corresponding to $i \in \mathcal{S}$ ae quantities that should be as small as possible, but no numerical bound is given. In yet another formulation, the problem is to find a controller that satisfies

$$\begin{array}{lll} \min & \{\phi_i(H)\}, & i \in \mathcal{S} \cup \mathcal{H} \\ \text{s.t.} & \phi_i(H) \leq d_i, & i \in \mathcal{H} \end{array} . \qquad (10)$$

This formulation is particularly nice since it not only seeks to satisfy the requirements, but it also seeks to find the best possible controller within those requirements. The constraints are there to enforce feasibility. Obviously, if the feasible set is empty, then there is no solution and the constraints must be relaxed.

Controllers that satisfy the above optimization problems (8), (9) and (10) are said to be *Pareto optimal*, which have the property that it is not possible to reduce any of the $\phi_i(H)$ without increasing at least one of the other $\phi_i(H)$. The problems generally do not have a unique solution, but a family of solutions which represent different trade-offs between the different objective functions.

There are a number of methods of finding solutions to (8), (9) and (10). Replacing the multiple objective minimization with the single objective minimization

$$\min \sum_{i=1}^{n} a_i\phi_i(H), \qquad a_i > 0$$

yields a Pareto optimal solution. Solving the weighted min-max problem

$$\min_H \max_i \{a_i\phi_i(H)\}, \qquad a_i > 0, \; i = 1, ..., n$$

yields a Pareto optimal solution. Fleming [10] suggests that the use of genetic search algorithms to solve (8), (9) and (10) is more robust in finding a solution than solving the nonlinear programs as above. Fleming mentions that due to the stochastic nature of the search, genetic algorithms are capable of searching the entire solution space with a greater likelihood of finding a global optimum.

Many of the examples in the literature for the above type optimization problems are mixed-norm problems. For example, minimizing a weighted sum of the H_2 and H_∞ norms, minimizing the l_1 norm subject to a constraint on the H_2 norm, etc. The solutions presented are very problem-specific. The following is a review of the more general methods presented in the literature.

Liu and Mills [17, 16] present a *convex combination* solution to the first problem (7) for LTI systems, where the objective functions are assumed to be convex. As shown in Boyd and Barrett [4], this assumption is reasonable since the majority of control specifications take the form of convex functions. The problem is then to find a controller $K(s)$ such that the specifications (7) are satisfied simultaneously. Liu and Mills solution is to find n sample controllers K_1, K_2,...,K_n, each of which satisfies at least one of the specifications. From these, closed-loop transfer functions $H_i(s)$ are derived to correspond to each K_i. At this stage, the following matrix is calculated, $\Phi \triangleq \{\phi_i(H_j)\}$. Liu and Mills show that the closed-loop transfer function

$$H^* \triangleq \sum_{i=1}^{n} \lambda_i H_i$$

satisfies all of the specifications simultaneously if the λ_i are solutions to the linear-programming problem

$$\Phi\Lambda \leq \Psi$$

where $\Lambda \triangleq \text{col}\ \{\lambda_i\}$ and $\Psi \triangleq \text{col}\ \{d_i\}$, provided this problem is feasible. It is shown that the necessary and sufficient conditions for feasibility of the above linear programming problem is that $\not\exists u \in \mathcal{R}^n \geq 0$ satisfying $u^T\Psi < \min_{1\leq j\leq n} u^T\psi_j$, where $\psi_j = \begin{bmatrix} \phi_{1j} & \phi_{2j} & \ldots & \phi_{nj} \end{bmatrix}^T$. Finally, the controller, K^* that solves the problem is derived from its closed-loop transfer function $H^*(s)$. The advantage of this method is that controllers need only be designed for one specification at a time.

Multiple Objective Parameter Synthesis (MOPS) [15] provides an automated way of tuning controllers to meet performance requirements. The method is very practical in that it does not have any underlying controller synthesis technique, and as such, it allows any controller synthesis technique to be used. Each controller synthesis technique has free parameters, p, which can be adjusted to tune the controllers. In PID laws, p consists of the gain parameters. In LQR, p consists of the entries of the Q and R matrices. The p could even be as general as the matrix entries in a state-space realization of a controller for a given controller order.

In MOPS, each performance objective is assigned a positive criterion $c_i(p)$, whose value is smaller the better the objective is achieved. For example, for the overshoot over a demanded steady-state value y_s, we have $c = \max_t(y(t)/y_s)$. The design criteria may be defined in terms of many things including, but not limited to, pole placements, time domain responses and frequency domain responses. Each c_i is assigned an upper bound demand value d_i. Some of the d_i are hard constraints, ie. $c_i \leq d_i$. Others are simply normalizing values so that all c_i/d_i may be compared reasonably. Defining $\phi_i(p) \triangleq {}_i/d_i$, the MOPS method involves choosing a controller synthesis method a-priori, and the solving the min-max problem

$$\min_p \max_i \{\phi_i(p)\}.$$

This can be solved using any general nonlinear optimization solver. Obviously, the solutions have the property that they are Pareto optimal. During the optimization procedure, some of the d_i may have to be relaxed to maintain feasibility of the solution. The procedure highlights which of the objectives are conflicting. The advantages of this procedure are that firstly, any controller synthesis technique may be used. Secondly, the objective functions may be defined in many ways. Thirdly, MOPS is applicable to nonlinear systems as well as linear systems. The disadvantage is that general nonlinear solvers are needed, and as such, MOPS may be computationally expensive to implement. This method was ranked best overall in the GARTEUR industrial assessment of the RCAM design challenge, in terms of control performance and industrial suitability.

A linear matrix inequality is a matrix inequality of the form

$$F(x) = F_0 + \sum_{i=1}^{m} x_i F_i > 0$$

where $x \in \mathcal{R}^m$ is the variable, and $F_i = F_i^T$ are given matrices. It is clear that the LMI presents a constraint on x (Boyd and Feron). For LTI systems, many performance specifications can be reformulated into linear matrix inequalities. The advantage of this is that the LMI's can be solved using convex optimization techniques, in fact there as a LMI Control toolbox in MATLAB. Thus, the problem can be solved (if a solution exists). LMI's are formulated in the state-space framework. The crux of the method is as follows. Since the closed-loop system is stable (we require it to be so), it must have the following Lyapunov function

$$V = x^T P x, \quad , P > 0$$

such that $A^T P + PA < 0$, where A is the closed-loop state matrix. The LMI method consists of imposing each performance specification as an additional constraint on the admissible Lyapunov functions, which results in and LMI for each performance specification (note that $A^T P + PA < 0$ in itself is also an LMI). There are many examples in the literature where LMI's are used to solve various types of multi-objective control problems [3, 8, 14, 18, 28, 31]. The disadvantage of the LMI approach is that transforming the specifications to LMI's introduces additional conservatism. The advantage is that LMI's may readily be solved using convex optimization techniques.

6 A Uniform Problem Formulation

We have seen the survey in integrated control and multi-objective control, in our opinion, two critical challenges in control systems development process. However, very little work has been done to consider both challenges together. In this paper, we would like to present a uniform framework such that the multi-objective and integration design can be treated simultaneously.

It is well known that any linear system can be formulated as a uniform framework [30], as shown in Figure

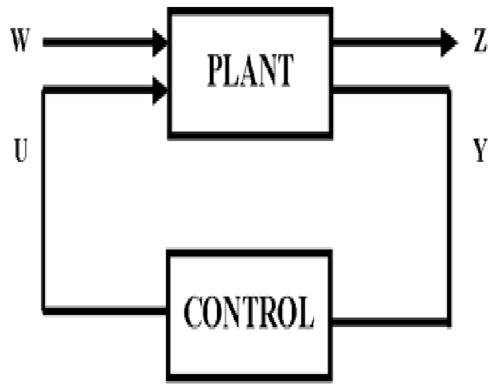

Fig. 1 System Framework

$$
\begin{aligned}
\begin{bmatrix} \boldsymbol{Z}(s) \\ \boldsymbol{Y}(s) \end{bmatrix} &= \begin{bmatrix} P_{zw}(s) & P_{zu}(s) \\ P_{yw}(s) & P_{yu}(s) \end{bmatrix} \begin{bmatrix} \boldsymbol{W}(s) \\ \boldsymbol{U}(s) \end{bmatrix} & (11) \\
\boldsymbol{U}(s) &= K(s) \cdot \boldsymbol{Y}(s) & (12) \\
\boldsymbol{Z}(s) &= H(s) \cdot \boldsymbol{W}(s) & (13)
\end{aligned}
$$

where $\boldsymbol{Z}$ represents the signals of interest, $\boldsymbol{Y}$ control outputs, $\boldsymbol{U}$ control inputs, and $\boldsymbol{W}$ external commands or disturbances. Therefore, the closed-loop transfer matrix $H(s)$ has the relationship with the open- loop matrix $P(s)$ and controller $K(s)$:

$$H = P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw} \tag{14}$$

We use this frame to treat integrated systems. Later, we will show that centralized and decentralized controllers can be derived from this uniform framework. Also, since the signals $\boldsymbol{Z}$ and $\boldsymbol{W}$ contain all the information of systems outputs and inputs respectively, we may be able to associate the design specifications to the closed-loop transfer matrix H, and the multi-objectives become:

$$\phi_i(H) \le d_i \quad i = 1, ..., n,$$

For example, consider the integrated flight/propulsion system (1), one can re-formulate into the framework:

$$
\begin{bmatrix} y_a \\ y_e \\ \cdots \\ e_a \\ e_e \end{bmatrix} = \begin{bmatrix} 0 & 0 & \vdots & G_a & G_{ae} \\ 0 & 0 & \vdots & G_{ea} & G_e \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ I & 0 & \vdots & -G_a & -G_{ae} \\ 0 & I & \vdots & -G_{ea} & -G_e \end{bmatrix} \cdot \begin{bmatrix} y_{ac} \\ y_{ec} \\ \cdots \\ u_a \\ u_e \end{bmatrix} \tag{15}
$$

or we can simply write

$$\begin{bmatrix} \boldsymbol{Z} = Y \\ \boldsymbol{Y} = E \end{bmatrix} = \begin{bmatrix} 0 & G \\ I & -G \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{W} = Y_c \\ \boldsymbol{U} = U \end{bmatrix} \tag{16}$$

With the controller

$$U = KE \tag{17}$$

We have the closed-loop transfer matrix

$$H = GK(I + GK)^{-1} \triangleq \begin{bmatrix} H_{aa} & H_{ae} \\ H_{ea} & H_{ee} \end{bmatrix} \tag{18}$$

Now, we can use this framework to treat the integrated control problem.

- The centralized controller, shown in (2), can be directly represented in (17).
- The decentralized controller, shown in (3), (4), and (5), can also be directly represented in (17).

As for the performance specifications, assume the overall multi-objectives are defined as functions of closed-loop transfer matrix H in (18), $\Phi(H) \triangleq [\phi_1(H)\ \phi_2(H) \ldots \phi_n(H)]^T$.

- For centralized controller, the controller is designed to meet the overall specifications.
- For decentralized controller, such as (3), one assume that the individual design has met their own multiple objectives, i.e., the closed-loop for plant G_{aa} and controller K_{aa} is $L_{aa} = G_{aa}K_{aa}(I + G_{aa}K_{aa})^{-1}$, and somehow the lower level specifications are defined as $\Psi(L_{aa}) \triangleq [\varphi_1(L_{aa})\ \varphi_2(L_{aa}) \ldots \varphi_{n_a}(L_{aa})]^T$. Similarly L_{ee} and $\Psi(L_{ee})$ are defined. Then the multiple objectives for centralized control system can be described as

$$
\begin{aligned}
H_{aa} &\sim L_{aa} & (19) \\
H_{ee} &\sim L_{ee} & (20) \\
H_{ae} &\sim 0 & (21) \\
H_{eq} &\sim 0 & (22)
\end{aligned}
$$

or one can directly address $\Phi(H) = \Psi(L_{aa})$ and $\Phi(H) = \Psi(L_{ee})$. Further, the derivations of Ψ from the overall performance specifications can also be addressed under the same framework.

There are several special cases that will further illustrate the usefulness of the uniform framework.

If the plant is decoupled, i.e. G matrix in (16) is diagonal, then a decentralized controller (3) will solve the design problem since in closed-loop H of (18) is also diagonal. In other words, design and their specifications are totally independent to each other.

Back into the decentralized implementation. When the K_{aa} is considered only, then it becomes

$$K = \begin{bmatrix} K_{aa} & 0 \\ 0 & 0 \end{bmatrix}$$

and the closed-loop transfer matrix becomes

$$H = \begin{bmatrix} H_{aa} = L_{aa} & H_{ae} \\ 0 & 0 \end{bmatrix}$$

It shows explicitly the interactions of closed-loop systems even with one individual controller.

In conclusion, the uniform framework presented here can be used for (centralized and decentralized) controller design, overall performance evaluation, and final simulation.

7 Conclusions

In this paper, we present the typical aircraft systems development process. The focus is placed on the control and integration technologies during this process. After a brief survey in this area, we decided to work on the multi-objective control for integrated systems. Based on literature review in these two areas, we propose a framework where the MOC and Integrated design are uniformly described.

The benefits and unique contributions of such framework are listed as follows.

- Decentralized and centralized design can be considered and evaluated together.
- Multiple objectives are described in a uniform format, which will be further allocated under the same framework.
- Simulation based on the framework can be used to evaluate the final design and verify the overall performance.

The work presented in this paper shows the initial stage of our research and development work in this area. A detailed study case problem formulation, as well as application to aircraft design, is under investigation.

Acknowledgment

The presented research work is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) Research Grant.

References

[1] anon S. *IEEE STD 1220-1998: IEEE Standard for Application and Management of the Systems Engineering Process (Revision of IEEE STD 1220-1994).* Software Engineering Standards Committee of the IEEE Computer Society, 1999.

[2] Bates D and Postlethwaite I. Integrated control system design for the automotive industry using modern robust control methods. *Proc Proceedings of IEE Colloquium on System Control Integration and Rapid Prototyping in the Automotive Industry*, pp 1–6, 1997.

[3] Boyd S, Balakrishnan V, Feron E, and Ghaoui L. E. History of linear matrix inequalities in control theory. *Proc Proceedings of American Control Conference*, June 1994.

[4] Boyd S and Barratt C. *Linear Controller Design: Limits of Performance.* Prentice-Hall, 1991.

[5] Bright M, Simon D, Garg S, and Mattern D. Piloted evaluation of an integrted methodology for propulsion and airframe control design. *Proc Proceedings of AIAA Guidance, Navigation and Control Conference*, 1994.

[6] Chen Z and Voulgaris P. Decentralized design for integrated flight/propulsion control of aircraft. *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 23, No 6, pp 1037–1044, 2000.

[7] D.Mattern and Garg S. Propulsion system performance resulting form and integrated flight/propulsion control design. *Proc Proceedings of AIAA Guidance, Navigation and Control Conference*, 1992.

[8] Dussy S and Ghaoui L. E. Measurement-scheduled control for the RTAC problem. *Proc Proceedings of American Control Conference*, 1997.

[9] Etkin B and Reid L. *Dynamics of Flight: Stability and Control.* 3rd ed. edition, John Wiley and Sons, 1996.

[10] Fleming P. Designing control systems with multiple objectives. *Proc Proceedings of IEE Colloquium on Advances in Control Technology*, 1999.

[11] Garg S. Partitioning of centralized integrated flight/propulsion control design for decentralized implementation. *IEEE Transactions on Control Systems Technology*, Vol. 1, No 2, pp 93–100, Jun 1993.

[12] Garg S and Mattern D. Integrated flight/propulsion control system design based on a centralized approach. *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 14, No 1, pp 107–116, 1991.

[13] Garg S, Ouzts O, Lorenzo C, and Mattern D. IMPAC - an integrated methodlogy for propulsion and airframe control. *Proc Proceedings of American Control Conference*, June 1991.

[14] Hindi H, Hassibi B, and Boyd S. Multi-objective

h_2/h_∞-optimal control via finite dimensional Q-parametrization and linear matrix inequalities. *Proc Proceedings of American Control Conference*, 1998.

[15] Joos H. A methodology for multi-objective design assessment and flight control synthesis tuning. *Aerospace Science and Technology*, Vol. 1, No 3, pp 161–176, 1999.

[16] Liu H. Design combination in integrated flight control. *Proc Proceedings of the American Control Conference*, pp 494–499, June 2001.

[17] Liu H and Mills J. Robot trajectory control system design for multiple simultaneous specifications: Theory and experimentation. *Journal of Dynamic Systems, Measurement, and Control*, Vol. 120, No 4, pp 520–523, December 1998.

[18] Mazabuchi I, Ohara A, and Suda N. Robust multi-objective controller design via convex optimization. *Proc Proceedings of 35th Conference on Decision and Control*, 1996.

[19] McRuer D, Ashkenas I, and Graham D. *Aircraft Dynamics and Automatic Control*. Princeton University Press, 1973.

[20] Moir I and Swabridge A. *Aircraft Systems: Mechanical, Electrical, and Avionics Subsystems Integration*. American Institute of Aeronautics and Astronautics, 2001.

[21] Neighbors III W and Rock S. Integrated flight/propulsion control - subsystem specifications for performance. *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 8, No 3, pp 572–578, 1995.

[22] Nelson R. C. *Flight Stability and Automatic Control*. second edition, WCB McGraw-Hill, 1998.

[23] Postlethwaite I and Bates D. Robust integrated flight and propulsion controller for the harrier aircraft. *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 22, No 2, pp 286–290, 1999.

[24] Pratt R. W. *Flight Control Systems*. The American Institute of Aeronautics and Astronautics, 2000.

[25] Rock S and et al. Integrated flight/propulsion control - system specifications. *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 17, No 1, pp 201–208, 1994.

[26] Schierman J and Schmidt D. Performance limitations of decentralized control laws for integrated fligth and propulsion control. *Proc Proceedings of AIAA Guidance, Navigation, and Control Conference*, Aug 1994.

[27] Schmidt P and Garg S. Decentralized heriarchical partioning of centralized integrated controllers. *Proc Proceedings of American Control Conference*, June 1991.

[28] Sherer C, Gahinet P, and Chilali M. Mutlobjective output-feedback control via LMI optimization. *IEEE Transactions on Automatic Control*, Vol. 42, No 7, pp 896–911, 1997.

[29] Stewart J, Burcham Jr. F, and Gatlin D. Flight-determined benefits of integrated flight propulsion control systems. *Proc Proceedings of 18th Congress of the International Council of the Aeronautical Sciences*, 1992.

[30] Stoorvogel A. *The H_∞ Control Problem*. Prentice Hall, 1992.

[31] Tokunaga H, Iwasagi T, and Hara S. Multi-objective robust control with transient specifications. *Proc Proceedings of 35th Conference on Decision and Control*, 1996.

[32] van Gelder P. Design of an integrated control system for flutter margin augmentation and gust load alleviation, tested on a dynamic windtunnel model. *Proc Proceedings of AIAA Guidance, Navigation and Control Conference*, 1986.

[33] Wang S and Davison E. On the stabilization of decentralized control systems. *IEEE Transactions on Automatic Control*, Vol. 18, No 5, pp 473–478, October 1973.