Part II – Decision Systems for Engineering Design
Lecture 1: Introduction

Robin Purshouse
University of Sheffield

Spring Semester 2023/24

1 Module overview

1.1 Module aims

In Part II of ACS6124 Multisensor and Decision Systems, students will develop an in depth knowledge and understanding of **decision systems** and the underlying mathematics and algorithms. Students will develop their confidence in solving complex problems requiring the application of decision techniques to a wide variety of applications.

1.2 Intended learning outcomes

After successful completion of Part II of the module, students will be able to:

1. Explain the importance of and need for decision systems in a wide range of industrial and research applications, and explain the relative merits and limitations of adopting such systems compared to other state-of-the-art solutions [M3];
2. Describe and explain the main components, architectures and design issues in decision systems, and future technological challenges and opportunities [M1];
3. Select and use appropriate architectures, and algorithmic, computational and experimental tools (including those from the research literature) to provide innovative solutions to complex, unfamiliar, open-ended decisions subject to a variety of technological constraints [M2];
4. Effectively present appropriate design methodology, analysis and critical evaluation of decision system solutions to engineering design problems [C17];

5. Use decision systems to evaluate the sustainability impact of solutions to engineering design problems and identify options that minimise adverse impacts [M7].

The codes in square brackets refer to Institute of Engineering and Technology (IET) learning outcomes – see `https://www.engc.org.uk/media/3464/ahep-fourth-edition.pdf` for further details.

1.3 Syllabus

A high-level overview of the syllabus for Part II of the module is shown below:

- Introduction to decision systems for engineering design, including problem formulation and real-world applications
- Sampling plans and knowledge discovery
- Multi-objective optimization: Pareto-based methods
- Multi-objective optimization: Decomposition and set-based methods
- Interactive multi-criteria decision-making
- Collaborative optimization

1.4 Recommended reading

Some elements of the course draw on topics and material from Forrester, Sóbester & Keane's *Engineering Design via Surrogate Modelling* (Wiley, ISBN 978-0-470-06068-1). A sequence of recommended journal paper readings will also be made available via Blackboard during Part II of the module.

1.5 Module pre-requisites

This module assumes that students have a basic understanding of calculus, linear algebra, and statistics. Students should also have some familiarisation with programming in Matlab.

2 Lecture overview

This lecture introduces the fundamental elements of an engineering design problem and motivates the need for a **decision systems** approach. Several real-world engineering design examples are included that illustrate these fundamental elements.

3 Elements of an engineering design problem

Consider a *complex engineered product*, such as a powertrain platform for use across a range of vehicle models. The *engineering design* for such a product is a complex process involving multiple engineering design teams, often working to different timescales and operating in different locations.

The design teams specialise in different areas—for a powertrain platform, this would be areas such as the engine system, the aftertreatment system and the transmission system—with each team responsible for a different aspect of the design and, potentially, different aspects of the overall performance of the product.

Through a carefully specified engineering design process, the design teams work together over time, expending their available resources, to realise an overall coherent design for the product—hopefully one that satisfies both customer and regulatory requirements. However, often the physics of the system is such that *conflict* naturally occurs between different aspects of performance (e.g. between fuel economy and emissions) making it difficult to find a single design that will satisfy all requirements simultaneously.

Decision systems play an important role in the design of a complex engineered product. They help the design engineers to:

1. Clarify the specifics of the design problem at hand;
2. Use resources efficiently in identifying promising *candidate designs*;
3. Understand the relationships between aspects of design and aspects of performance;
4. Communicate the designs and understanding to other teams and the Chief Engineer with overall responsibility for the product.

This second part of ACS6124 is about introducing these decision systems. But, before we get to the systems, let's think in slightly more formal terms about the engineering design problem they seek to address.

3.1 Design variables

The **design variables** are the parts of the engineering problem that are under the control (or at least partial control) of the design engineer. For complex engineered products, design variables typically exist at one of the following hierarchical levels:

1. *Architecture*—variables that define the overall concept for the architecture of the system. For a powertrain, these would be variables such as choice of power system technology (e.g. presence or absence of an exhaust gas recirculation system).

2. *Component*—variables that define the specific hardware aspects of components *within* a given architecture. Again, for a powertrain, this might be items such as the detailed geometry of the aftertreatment block or choice of precious metals within the catalyst.

3. *Calibration*—variables that can be defined *after* the hardware design has been realised, e.g. specification of injection timings in the engine, or choice of controller gains in the engine control system.

In what follows, we will denote a design variable using the conventional notation x_i, where i refers to the ith design variable in the problem. We refer to a set of design variables using vector notation, i.e. $\mathbf{x}$. We subscript vectors of design variables, $\mathbf{x}_o$, to indicate variables under the control of the oth design team within the overall engineering design organisation. In this module, we won't distinguish between design variables at different levels in the design hierarchy.

3.2 Parameters

Parameters are factors that can affect the performance of a design that are not under the control of the design engineer. Parameters can be categorised as follows:

1. *Environmental conditions*—factors that vary with the anticipated operating environment of the product, such as the ambient temperatures and pressures in the environment within which a powertrain could be operating.

2. *Physical constants*—empirical values identified as being important to the physics of the problem, e.g. the gravitational constant, g, or other empirically-derived relationships encoded in mathematical models of the system.

3. *Linking variables*—factors either directly or indirectly under the control of other design teams, e.g. from the perspective of the aftertreatment design team, the characteristics of the emissions generated by the engine.

Parameters are often characterised by **uncertainty**—arising from manufacturing tolerances, unknown operating conditions, errors in representing the physics of the system, or yet-to-be-determined choices (and their consequences) from other design teams.

In the module, we will denote a parameter using the conventional notation p_j, where j refers to the jth parameter in the problem. We refer to a set of parameters using vector notation, i.e. $\mathbf{p}$. We subscript vectors of parameters, $\mathbf{p}_o$, to indicate uncontrollable variables faced by the oth design team within the overall engineering design organisation.

3.3 Performance criteria

The **performance criteria** are the dimensions by which a candidate design is assessed. The criteria are closely related to the **requirements** for the product; however since requirements for a single product usually number in the tens of thousands and are organised in a hierarchy, performance criteria typically represent only the high-level requirements or overall **attributes** that the design is aiming to meet. A complex engineered product might typically have around 20 performance criteria (e.g. for a powertrain these would include attributes such as efficiency, performance, noise, durability, weight and cost).

Performance criteria are normally denoted using the notation z_k, where z_k refers to the kth criterion in the problem. We refer to a set of criteria using vector notation, i.e. $\mathbf{z}$. We subscript vectors of parameters, $\mathbf{z}_o$, to indicate criteria that are the responsibility of the oth design team within the overall engineering design organisation.

3.3.1 Constraints

Constraints are performance criteria for which a defined level of performance is necessary: either the criterion must be met exactly (an *equality constraint*) or a minimum level of performance must be achieved (an *inequality constraint*). The latter are the most common type of constraints found in engineering design problems (e.g. a maximum quantity of nitrogen oxide that can be released as part of a regulatory drive cycle).

Inequality constraints are usually expressed as $z_k \leq g_k(\mathbf{x}, \mathbf{p})$, although the functional form $g_k(.)$ may be implicit. In a similar fashion, equality constraints are expressed using $z_k = h_k(\mathbf{x}, \mathbf{p})$. Sometimes the z component of the notation is dropped for convenience.

3.3.2 Objectives

Objectives are criteria for which a desired direction of performance is expressed, but where constraints on the absolute level of performance do not exist—so objectives are either to be *maximized* or *minimized*. Usually the latter convention in adopted for convenience (without loss of generality, since minimisation of z_k is equivalent to maximisation of $-1 \cdot z_k$). In a typical design problem, cost and weight are criteria to be minimized.

Objectives are usually expressed as $z_k = f_k(\mathbf{x}, \mathbf{p})$, although the functional form $f(.)$ may be implicit. Again, sometimes the z part of the notation is dropped for convenience.

3.3.3 Preferences

Preferences relate to the desires of the *decision-maker* (e.g. the design engineer or the product Chief Engineer) for achieving particular levels of performance. Constraints, unless they relate to true physical bounds, can be considered a *hard* form of preference that must be met. There may also be *soft* forms of preference such as desirable but not essential levels of performance against the criteria.

Common forms of preference elicited directly with respect to the criteria include:

- *Goals*—preferred levels of performance against the criteria, e.g. "90g CO_2 per km". A design can be said to be *satisficing* if it meets the goals of the decision-maker. Goals are also sometimes known as *targets*.
- *Priorities*—some kind of ordering expressed over the criteria, e.g. "minimize weight first, then minimize cost".
- *Weights*—relative strengths of preference for performance against the criteria, e.g. "minimizing weight is twice as important as minimizing cost".

Preferences can also be expressed by direct comparisons between designs (where performance is implicit), e.g. "I prefer design A to design B". These types of preferences then need to be converted into expressions like the above that explicitly relate to performance against the criteria.

3.4 Evaluation functions

An important aspect of engineering design is to be able to estimate the performance of a candidate design. Formally, this is a mapping from the design variables $\mathbf{x}$ and parameters $\mathbf{p}$ to the performance criteria $\mathbf{z}$—i.e. the $f_k(.)$, $g_k(.)$ and $h_k(.)$ functions discussed above. Evaluations come in the form of physical experiments, mathematical models, and expert opinion. Mathematical models have long played a part in evaluating designs, with modern companies attempting to virtualise as much evaluation as possible in order to reduce design costs and timelines. Such models may be computationally fast to evaluate (e.g. controller design models based on ordinary differential equations) but may also be computationally expensive (e.g. crash simulations based on partial differential equations). Very often the models are sufficiently complex that the $f_k(.)$, $g_k(.)$ and $h_k(.)$ functions cannot be written down and the simulation model acts as a 'closed box' in a way similar to a physical experiment.

3.5 Putting it all together

Bringing all the aspects in this section together, the notion of decision systems for engineering design can be formally expressed as a **constrained multi-objective optimization problem**:

$$
\begin{aligned}
\underset{\mathbf{x}}{\text{minimize}} \quad & \mathbf{f}(\mathbf{x}, \mathbf{p}) \\
\text{subject to} \quad & \mathbf{g}(\mathbf{x}, \mathbf{p}) \leq \mathbf{0} \\
& \mathbf{h}(\mathbf{x}, \mathbf{p}) = \mathbf{0} \\
& \mathbf{x} \in \mathcal{D} \\
& \mathbf{p} \leftarrow \mathcal{P}
\end{aligned} \tag{1}
$$

where $\mathcal{D}$ is the space of design variables and $\mathcal{P}$ is the space of parameters. Here we have assumed the problem is solved across $\mathcal{D}$ for a single realisation from $\mathcal{P}$ (i.e. a nominal parameterisation of the problem), but other formulations are possible in which the full parameter space is considered.

If a *utility function* that encapsulates the decision-maker preferences, $U(.)$, is available then the first line of Equation 1 can be re-expressed as:

$$\underset{\mathbf{x}}{\text{maximize}} \quad U(\mathbf{f}(\mathbf{x}, \mathbf{p})) \tag{2}$$

4 Examples of engineering design problems

In this final section of the lecture, three engineering design problems are introduced, covering decision-making at architecture, component and calibration level. The articles relating to these problems are included as an accompaniment to the lecture notes on Blackboard.

4.1 Aerospace systems architecture design

4.1.1 Overview

As an example of an architecture-level problem, we consider the design of the overall control systems architecture for an aircraft. Over the past two decades, in the aerospace sector, there has been a move from traditional, centralised control boxes to more distributed control systems, featuring a network of sensors, actuators and 'smart' signal processing modules. Increasing moves toward electrification are also challenging existing hardwired architectures for aircraft control. A key question is how to design the overall architecture so that the functional requirements of the various control systems are met.

4.1.2 Design variables and parameters

A schematic for a generic distributed aircraft control systems architecture is shown in Figure 1. The main design variables $\mathbf{x}$ are:

- The number of signal processing modules in the system;
- The topology of the databus (ring, star, vertical, horizontal, and hybrid) and the level of redundancy (simplex, duplex or triplex);
- The types and corresponding numbers of sensors and actuators, together with their levels of redundancy (simplex, duplex or triplex).

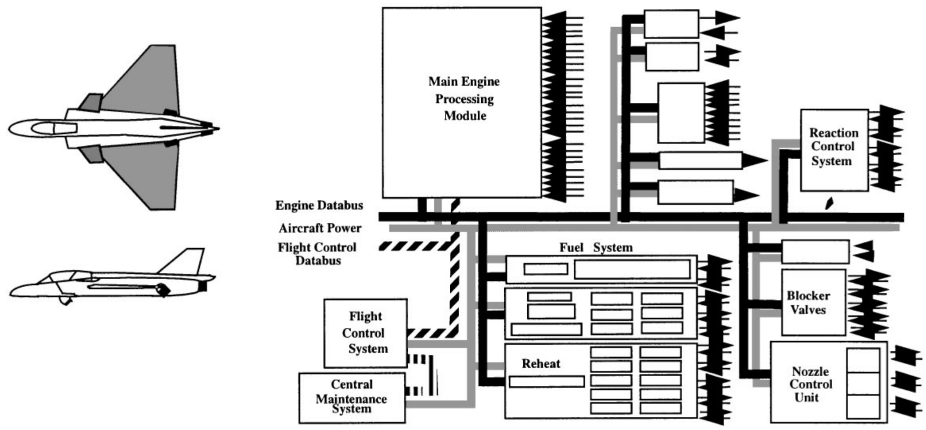

Figure 1: Schematic of a generic distributed control systems architecture for aircraft applications. Taken from Thompson et al.'s *Distributed aero-engine control systems architecture selection using multi-objective optimisation*, 1999

4.1.3 Performance criteria

The performance criteria $\mathbf{z}$ against which each possible architecture design is assessed are:

- Weight;
- Cost;
- Diagnostic capability;
- Risk;
- Maintenance.

Each criterion is treated as an objective, so there are no constraints in the problem formulation (this is common in conceptual design problems that are not directly assessing the detailed requirements specification of a product). Note that each criterion is itself a composite measure that combines, via a mathematical function, several aspects relating to the associated theme. For example, the diagnostic capability metric takes into account five distinct aspects of performance: local compensation, self-diagnosis, remote diagnosis, reconfiguration of failure and time-limited dispatch.

4.1.4 Evaluation of a candidate design

Since conceptual design problems address the performance of a system in its totality, it is often the case that there are few, if any, detailed physics models that can be harnessed to help with evaluating a design. Instead, high-level models are used,

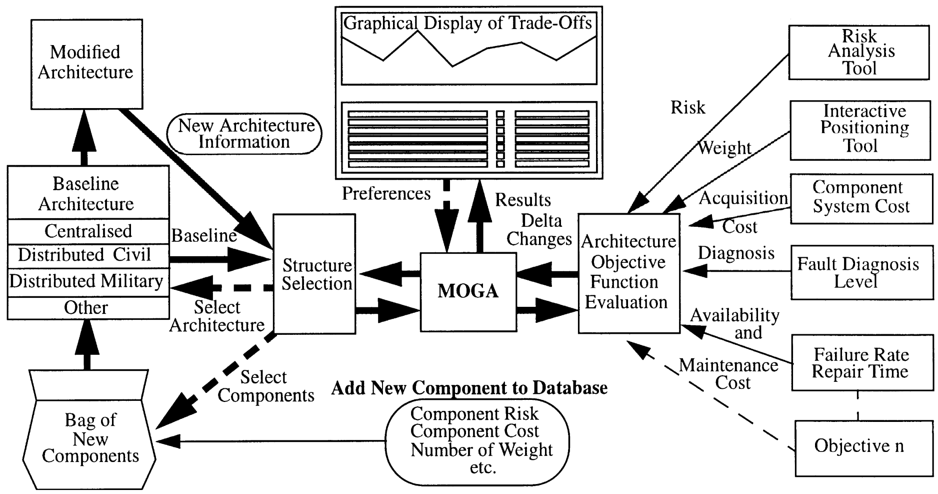

Figure 2: Components of a decision system. Taken from Thompson et al.'s *Distributed aero-engine control systems architecture selection using multi-objective optimisation*, 1999

which typically integrate expert opinion concerning the performance of building blocks of the architecture (e.g. the ability of a smart sensor to self-diagnose might have been captured by an expert expressing the judgement on an ordinal scale from 1 to 5, where 1 represents "no capability", and 5 indicates "excellent capability"). These models will contain crude simplifications but are often regarded as 'good enough' for use in comparing different architectures. An advantage of these models is that they are of low computational complexity and can be used to evaluate many thousands of candidate designs within typical resource budgets.

4.1.5 Example decision system

The Rolls-Royce University Technology Centre (UTC) in Control & Monitoring Systems Engineering housed in ACSE has been developing decision systems for systems architecture design for almost three decades. The work reported by Thompson and colleagues contains a good illustration of an overall decision system, which is reproduced in Figure 2. Over the coming lectures, we will be looking in detail at aspects of the decision system relating to the so-called 'MOGA' block at the heart of the diagram.

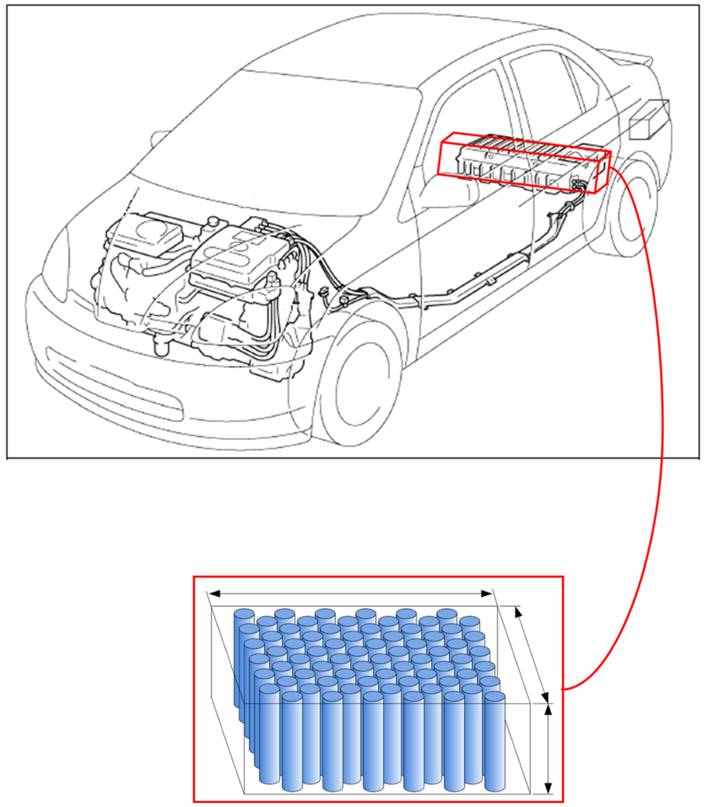

Figure 3: Battery hardware design problem for a hybrid electric vehicle. Taken from Dandurand et al.'s *Equitable multi-objective optimization applied to the design of a hybrid electric vehicle battery*, 2013

4.2 Automotive battery design

4.2.1 Overview

Hybrid vehicle engineers face particular challenges in how to package electric power technologies inside vehicles developed primarily without those features in mind. The choice of battery hardware and configuration (including its structural connections to the wider vehicle) is a good example of a hardware design problem. An example is shown in Figure 3 for a Lithium-ion battery pack in a mid-sized saloon hybrid vehicle.

4.2.2 Design variables and parameters

The design variables $\mathbf{x}$ in this problem are:

- Battery cell shape;

- Cell size;
- Cell layout;
- Spacing between cells.

The problem features parameters $\mathbf{p}$ relating to heat flow coefficients in the vehicle—these are not directly under the control of the battery design engineer but could be regarded as linking variables (that are affected by other packaging choices in the wider vehicle).

4.2.3 Performance criteria

As usual, weight and cost are important objectives to be minimized. Other objectives specific to the battery problem are:

- Deviation in cell temperature from a target value (40°C in this example);
- Evenness of the temperatures across cells.

In the problem formulation as presented by Dandurand and colleagues (working at BMW's research centre in Clemson University), there is a temperature deviation objective for every column of cells in the battery pack. No constraints were formulated, although we might imagine that these temperature criteria have hard limits, beyond which the battery will be thermally compromised. Note also the higher level of detail between these two objectives and those specified for the previous architecture example.

4.2.4 Evaluation of a candidate design

Designs are evaluated using a lumped-parameter physics model of the battery pack implemented in Simulink. The relatively low fidelity of this model, in comparison to a distributed-parameter model, has implications for how accurately the performance of each candidate design can be evaluated—this is why Dandurand's team evaluate temperatures for columns of cells rather than for individual cells. Note that the lumped-parameter model has a benefit though, in that its function evaluations will be several orders of magnitude faster than the equivalent distributed-parameter model.

4.3 Power generation controller design

4.3.1 Overview

Integrated gasification combined cycle (IGCC) power plants combine a gas and steam cycle to help provide cleaner and more efficient power from coal sources. The gasifier in the power plant converts a mix of pulverised coal and limestone, air and steam into a fuel gas. The gasification process is a challenging control problem and was promoted as an open benchmark problem by the company Alstom in the early 2000s. Griffin and colleagues in ACSE's Rolls-Royce UTC developed an H-infinity controller for the gasifier, using decision system technologies to help tune the controller gains.

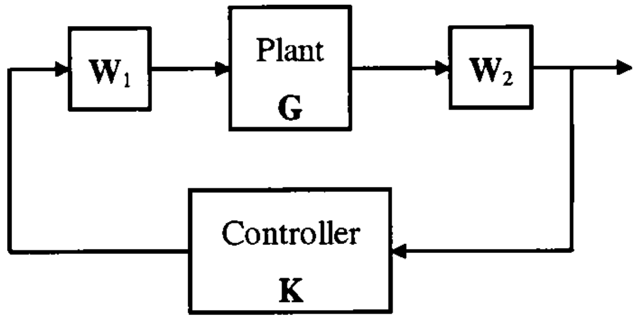

Figure 4: H-infinity robust control system for a power generation application with weighting function matrices $\mathbf{W_1}$ and $\mathbf{W_2}$. Taken from Griffin et al.'s *Multi-objective optimization approach to the ALSTOM gasifier problem*, 2000

4.3.2 Design variables and parameters

A high-level schematic of the H-infinity control system is shown in Figure 4. The design variables $\mathbf{x}$ are:

- The first order lags in the pre-plant diagonal matrix $\mathbf{W_1}$ (8 variables in total);
- The gains in the post-plant diagonal matrix $\mathbf{W_2}$ (4 variables in total).

From the perspective of the weighting function problem, parameters $\mathbf{p}$ relate to the coefficients in the plant model $\mathbf{G}$ and the realised state-space controller $\mathbf{K}$.

4.3.3 Performance criteria

The performance criteria $\mathbf{z}$ relate to the response of the system to a pressure disturbance and comprise four objectives to be minimized (all integrals of absolute error (IAE) of the response over 300 seconds) and four constraints to be satisfied (all peak fluctuations from the operating point):

- IAE of fuel gas calorific value;
- IAE of gasifier bed mass;
- IAE of fuel gas pressure;
- IAE of fuel gas temperature;
- Peak fluctuation of fuel gas calorific value;
- Peak fluctuation of gasifier bed mass;
- Peak fluctuation of fuel gas pressure;
- Peak fluctuation of fuel gas temperature.

4.3.4 Evaluation of a candidate design

Evaluation of a candidate pair of weighting function matrices $\mathbf{W_1}$ and $\mathbf{W_2}$ is performed using three linear state-space models that were developed through linearisation of a non-linear gasifier model at three distinct operating points. The models are implemented in Simulink and are fast to compute.

5 Further reading

1. Thompson H.A., Chipperfield A.J., Fleming P.J., Legge C. Distributed aero-engine control systems architecture selection using multi-objective optimisation. *Control Engineering Practice* 1999;7:655–664.

2. Dandurand B., Guarneri P., Fadel G., Wiecek M.M., Equitable multi-objective optimization applied to the design of a hybrid electric vehicle battery. *Journal of Mechanical Design* 2013;135:(041004):1–8

3. Griffin, I.A., Schroder P., Chipperfield A.J., Fleming P.J. Multi-objective optimization approach to the ALSTOM gasifier problem. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering* 2000;214:453–468.