ENES 489P Hands-On Systems Engineering Projects

Multi-Objective Optimization and Trade Study Analysis

Mark Austin

E-mail: austin@isr.umd.edu

Institute for Systems Research, University of Maryland, College Park

Table of Contents

Part 1. Trade Study Analysis

- Tradeoff Analysis in Design
- Motivating Application: Route Selection in Transportation Engineering
- Preference Selection

Part 2. Multiobjective Optimization

- Problem Formulations for System Optimization
- Optimality Criteria / Visualization Techniques
- Sets of Noninferior Solutions and Two-Dimensional Problem

Part 3. Tradeoff Analysis with Multi-Criteria Optimization Tools

- Limitations of Trial-and-Error Analysis
- Assignment-Type Problems and use of ILOG CPLEX.

Part 1. Trade Study Analysis

Trade Studies

Sources of Tradeoff in Engineering Design

Engineering systems are typically designed to ...

... satisfy the needs of multiple stakeholder needs.

Each stakeholder will have:

- A set of functional requirements,
- Levels of performance that need to be met, and
- A budget.

Multiple objectives occur because ...

... a good design balances the attributes of economy, performance, reliability/quality, use of resources, details and timing of implementation.

Satisfying all of these criteria typically results in tradeoffs.

Trade Studies

Generation of Good Design Alternatives

Multiple (and possibly competing) design criteria implies that there could be ...

... many good design solutions and many bad design solutions.

Purpose of a Trade Study

The purpose of a trade study is to ...

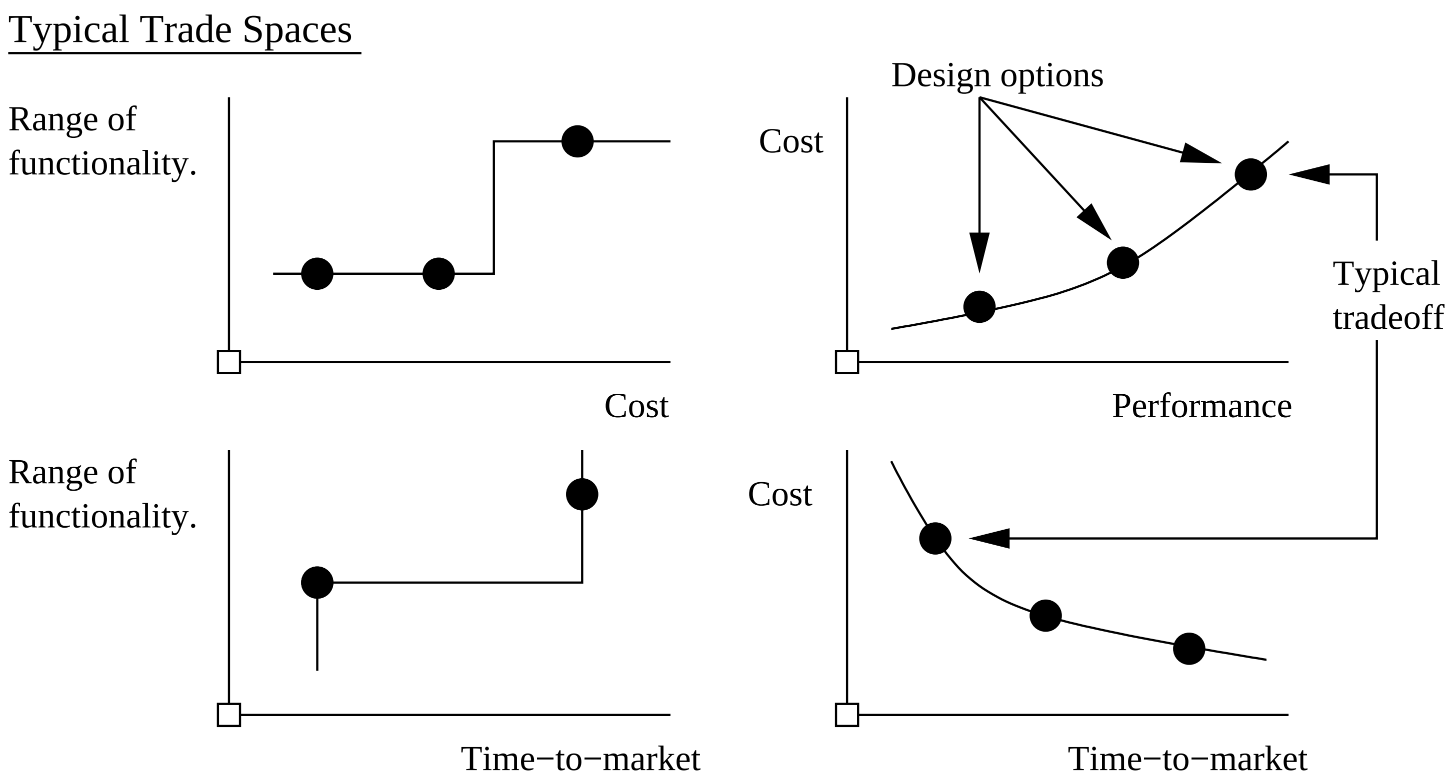
... examine the relative value and sensitivity of attributes associated with the design's measure of effectiveness.

This information is then used to ...

... guide decision making relating to the selection and treatment of design alternatives.

Typical Tradeoffs in Design

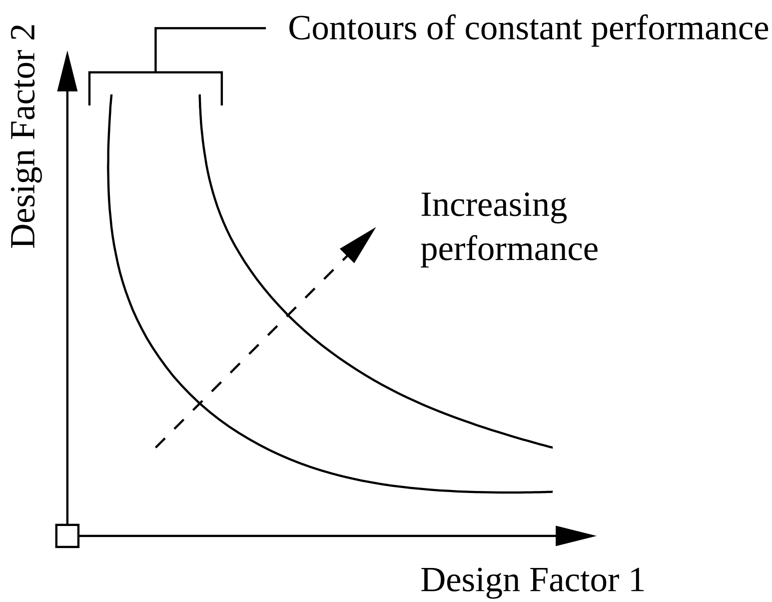
Typical Trade Spaces



Typical Tradeoffs in Design

A Few Observations

- More functionality usually means less economy (i.e., increases in system cost).
- Improved performance usually means less economy (i.e., increases in system cost).
- For systems having a fixed cost, improvements in one aspect of performance may only be possible with a decrease in other aspects of performance, i.e.,



Typical Tradeoffs in Design

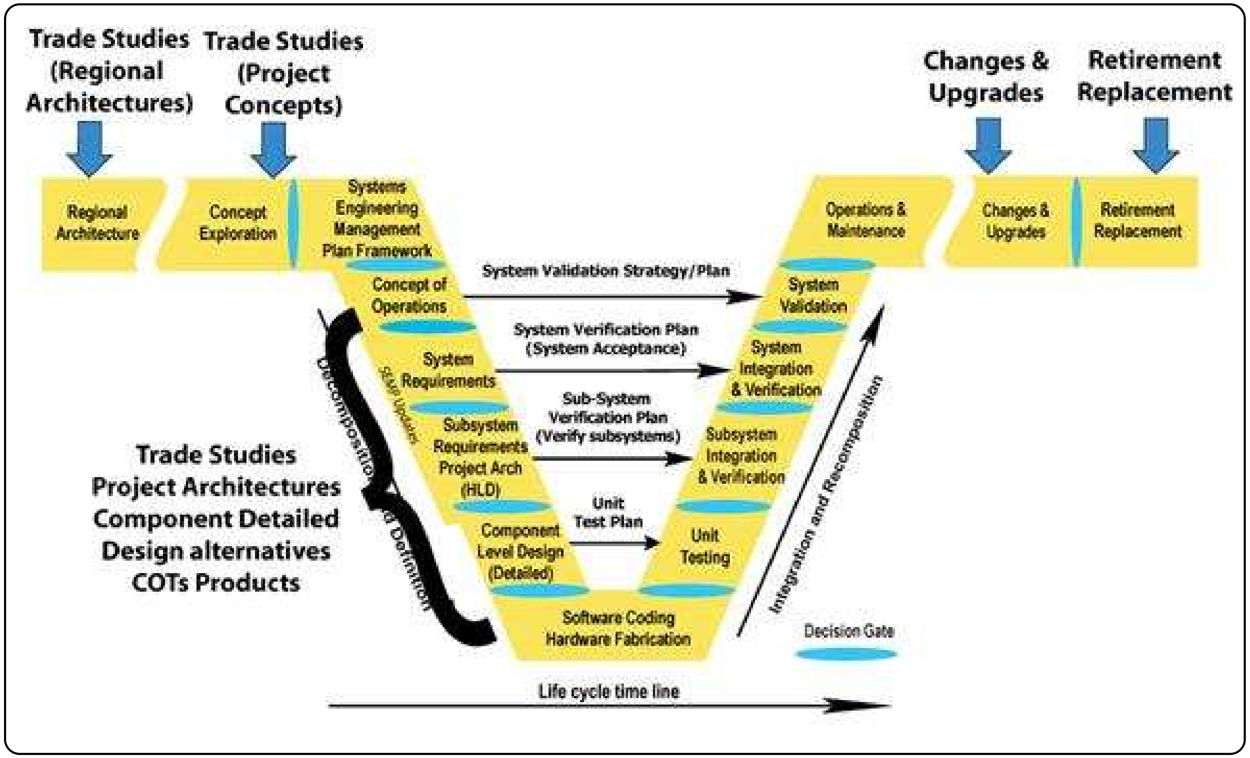
Decision Making in Typical Trades

For example:

- Serial versus parallel implementation of operations.
- Use of hardware versus software.
- Computation versus storage.
- Selection of hardware component performance versus component cost.
- Speed of system implementation versus cost.

Tradeoff Studies in System Development

Trade Studies at Various Stages of the V-Model



Source: Systems Engineering Handbook for ITS, Federal Highway Administration.

Motivating Application

Route Selection in Transportation Engineering

A fundamental problem in transportation engineering is ...

... the planning of routes for expansion of transportation networks.

Problem Statement

Suppose that we want to ...

... build a road from city A to city B, but that a mountain range spans the most direct route.

Is it better to ...

... build a road around the mountains,

or ...

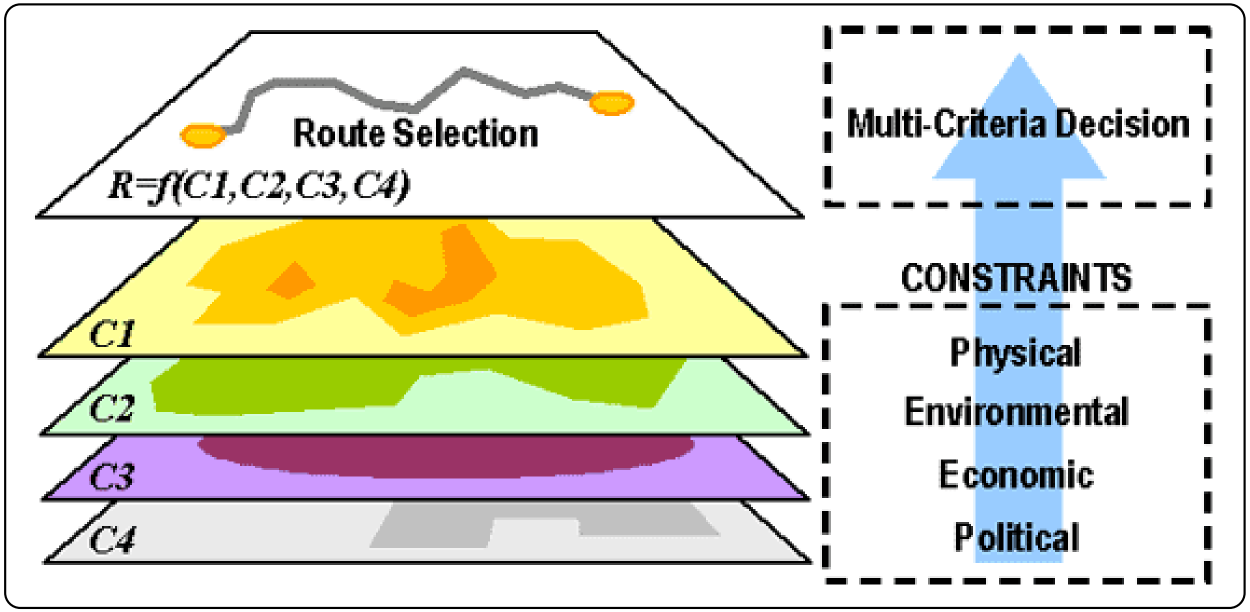
... pay more money upfront to build a tunnel through the mountains and provide a shorter route?

Motivating Application

Solution Procedure

The standard approach to problems of this type is to ...

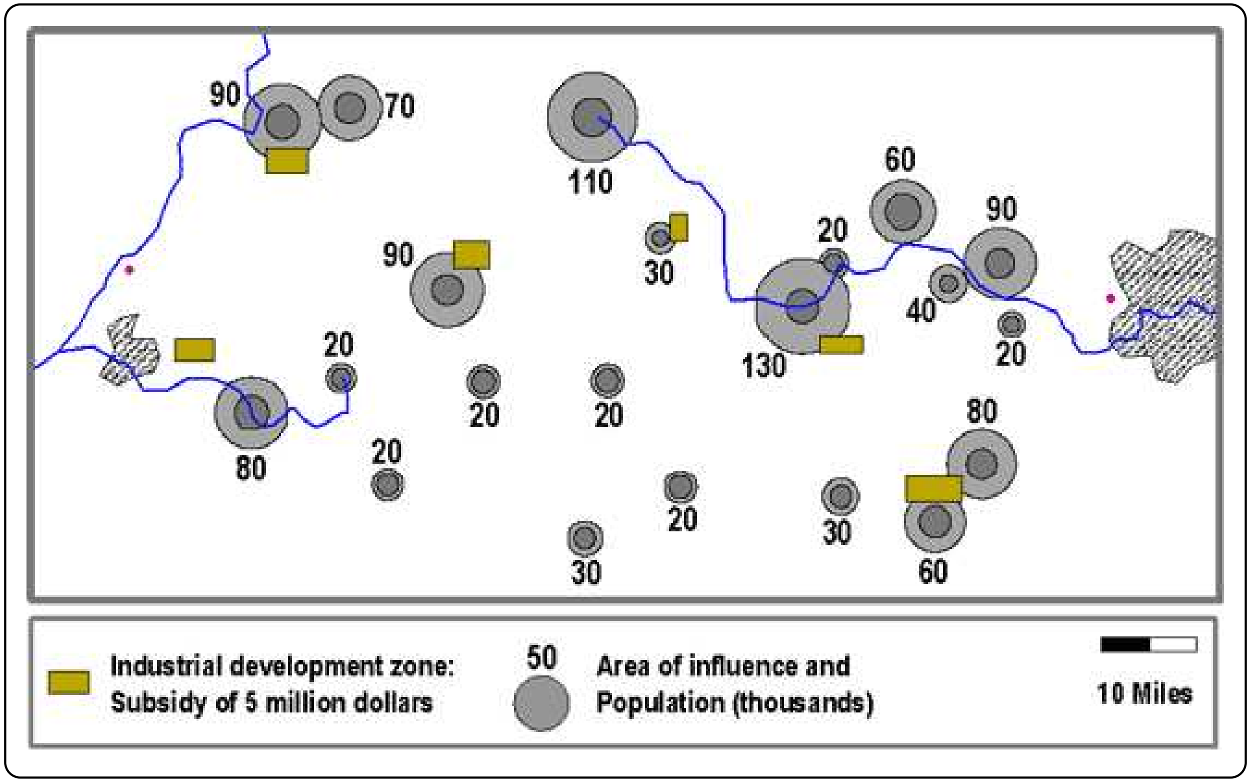
... deal with each concern separately, and then combine the results.



Motivating Application

Design Objective

Make sure that transportation routes need to go to the population centers...

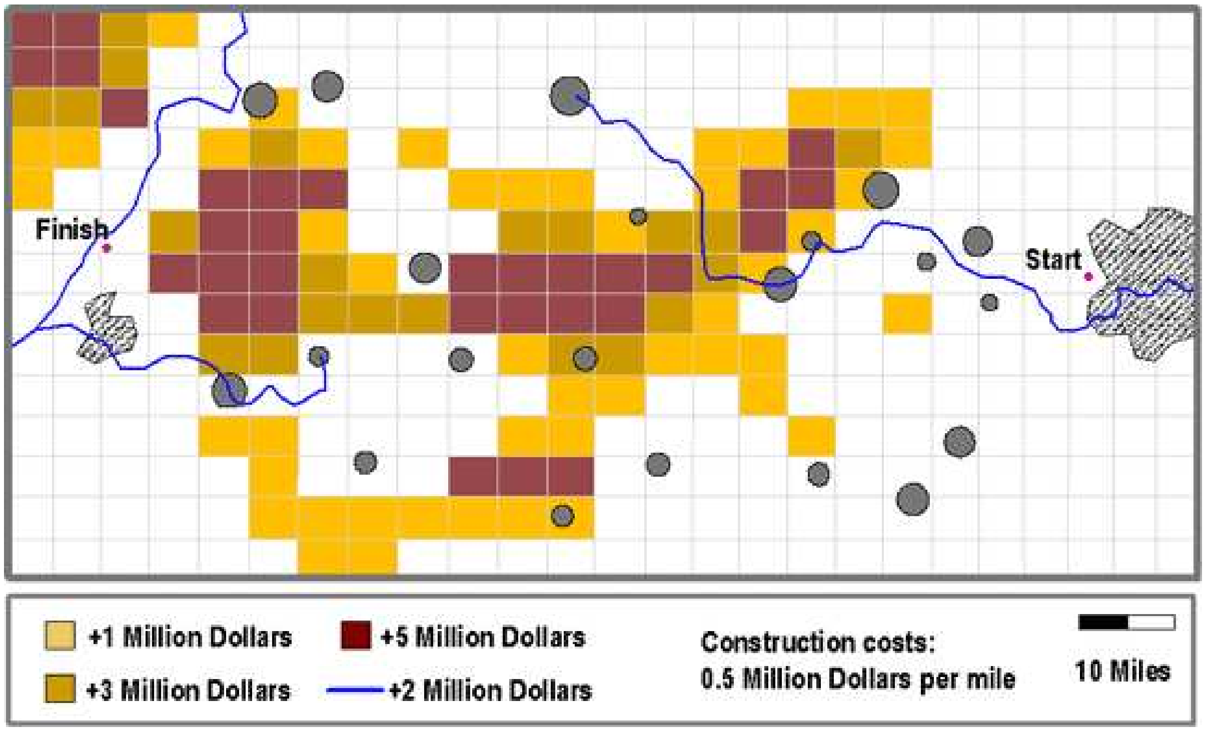


Motivating Application

Design Constraints

Try to minimize construction costs associated with physical constraints/mountains.

Construction Costs

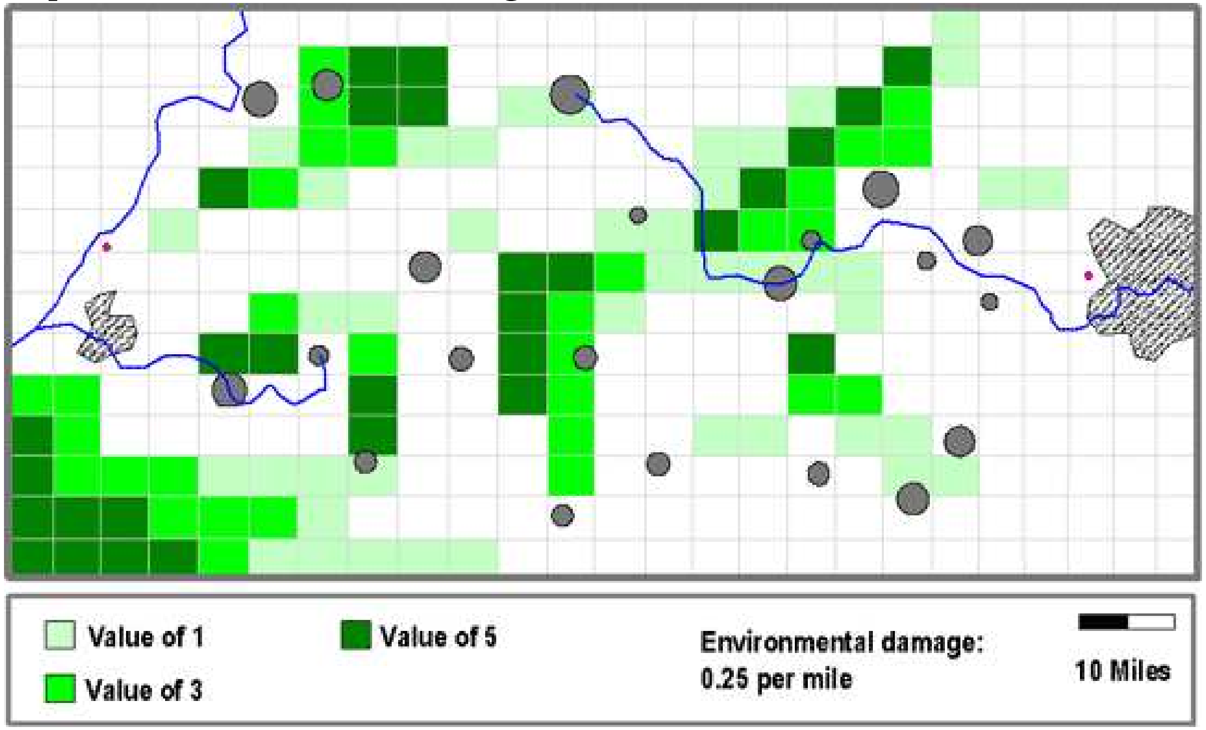


Tradeoff Studies in System Development

Design Constraints

Try to minimize environmental damage caused by the transportation route.

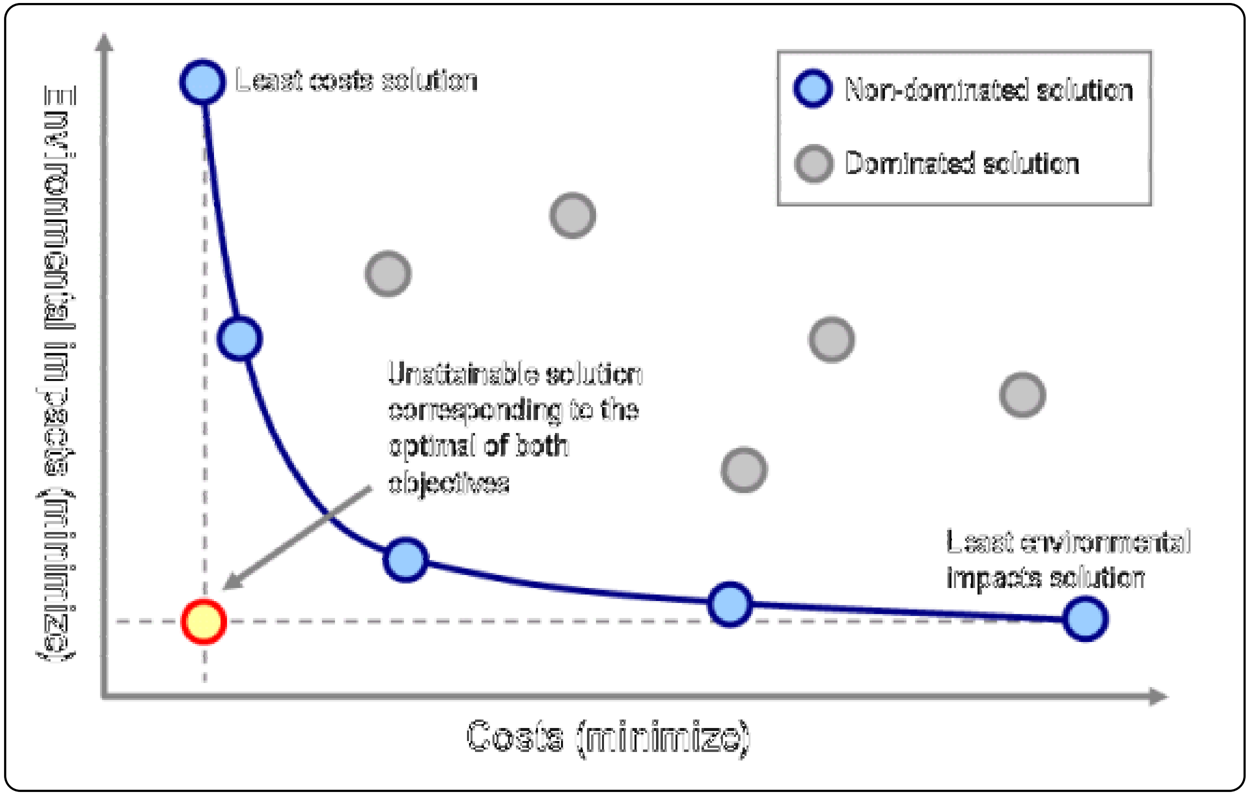
Impact of Environmental Damage



Tradeoff Studies in System Development

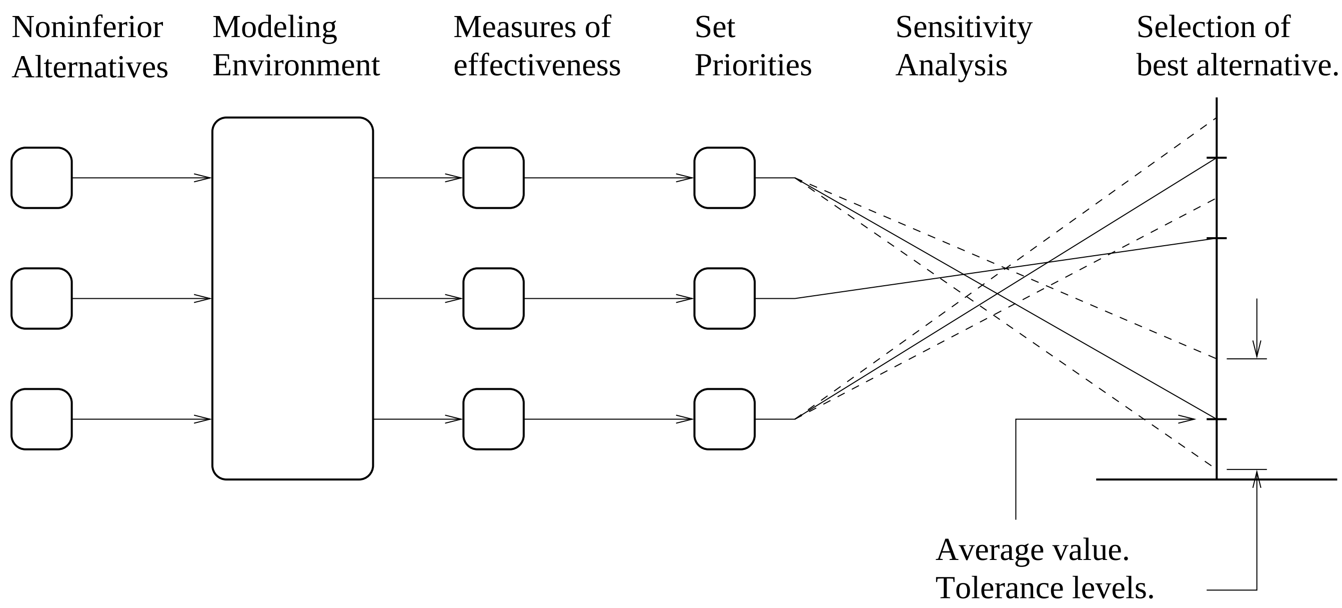
Typical Trade Space

The final result is always never a single point, but rather a family of good solutions:



Preference Selection

Evaluation and Ranking of Design Alternatives



For practical engineering problems, modeling system performance may be expensive and time consuming. These features ...

... place upper limits on the number of alternatives that can be considered within a limited time frame.

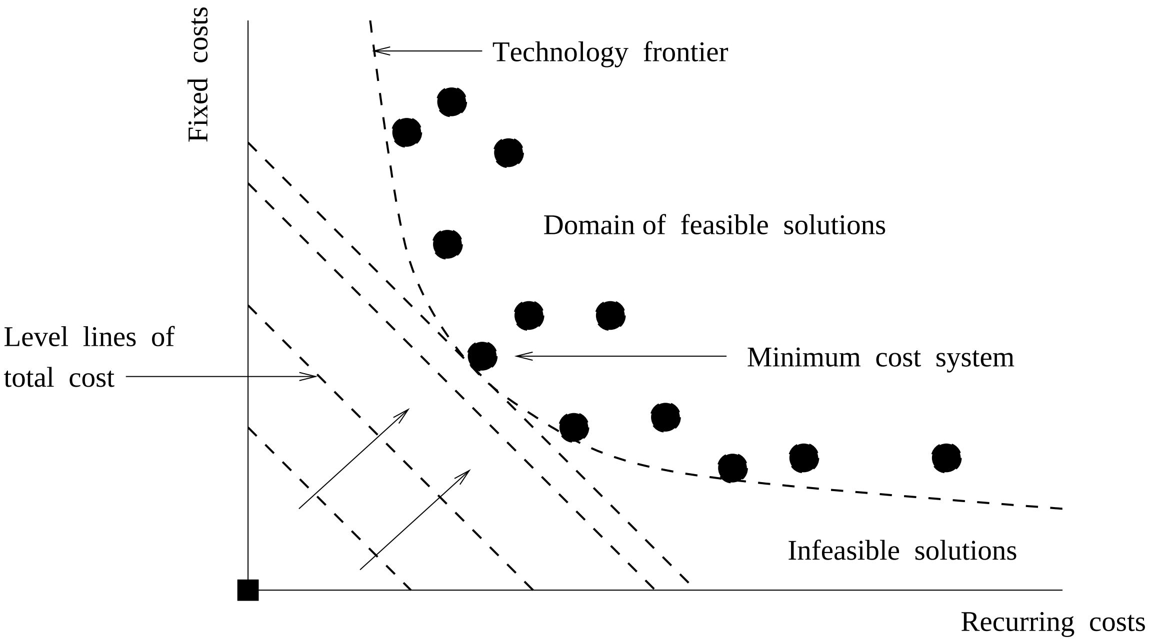
Preference Selection

Preference Selection based on Cost Alone ...

The best option is the design that is technically feasible, and has a total cost:

$$\text{Total cost} = \text{Fixed cost} + \text{Recurring cost} \tag{1}$$

that is minimized.



Part 2. Multi-Objective Optimization

System Optimization

Framework for System Optimization

System optimization is:

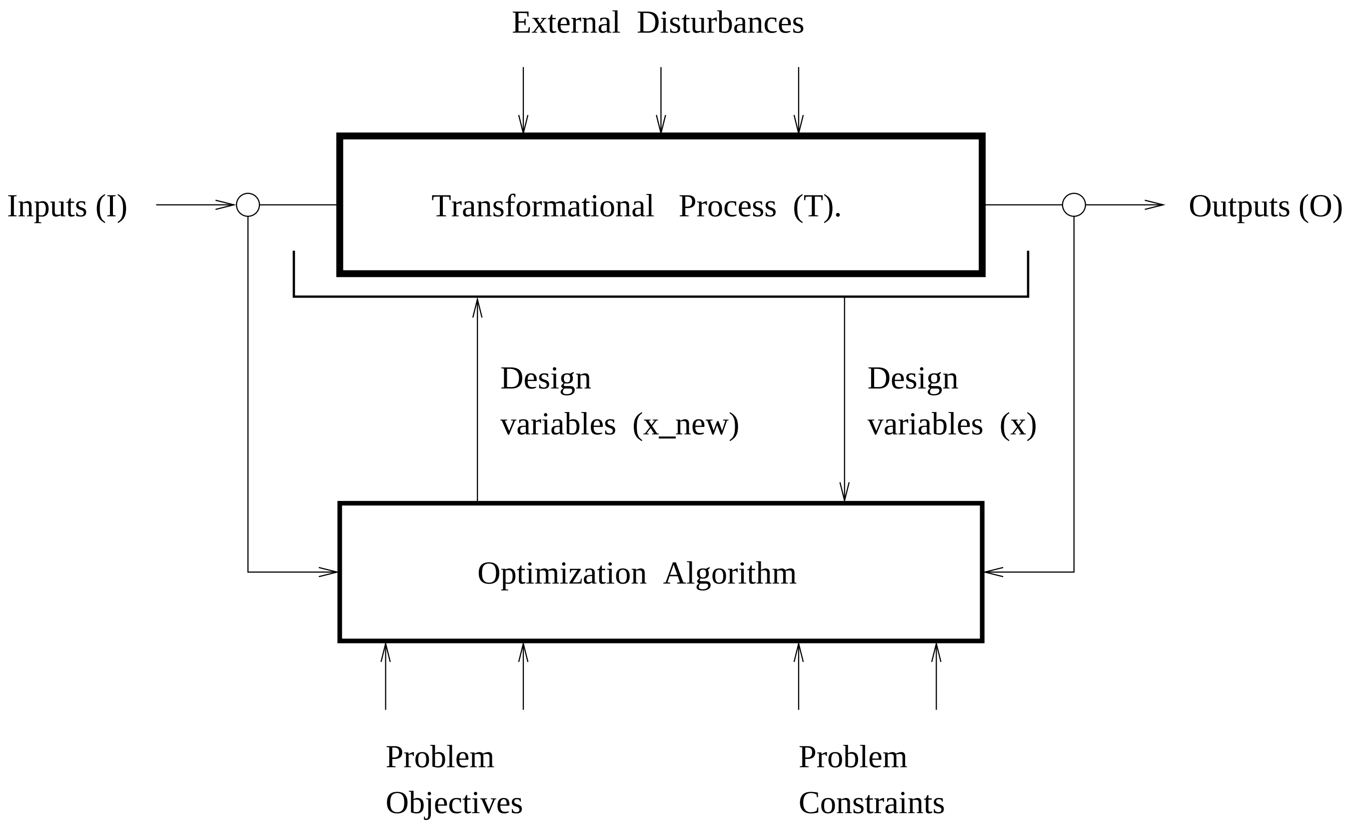
... a problem solving process that systematically looks for a set of design variables ”x” that will maximize (or minimize) a goal function.

Most optimization problems can be cast in terms of ...

... transformation models, where optimization may be interpreted as picking I, O, or T such that a specified evaluation criterion is optimized.

System Optimization

Components of a System Optimization Problem



System Optimization

System Optimization Pathway

Optimization algorithms receive as their input ...

... information on ”x”, the system inputs and outputs (I/O), the problem goals and constraints,

and generate ...

... a revised set of decision variables x_new.

Techniques

Techniques for selecting optimal values of "x" include:

- Simple trial-and-error search strategies,
- Mathematical programming techniques,
- Search procedures guided by combinations of heuristic/analytical information.

Problem Formulations for System Optimization

Method 1: Weighted Index Formulation

Convert multiobjective problems into a single objective optimization problem, i.e.,

$$f(x) = \sum_{i=1}^{r} w_i f_i(x) \qquad (2)$$

where $w_i > 0$ can be thought of as giving the relative importance of minimizing $f_i(x)$.

Problem Formulations for System Optimization

Procedure

Decision tables are an appropriate representation for ...

... problems where the number of alternatives is small enough that all decisions and outcomes can be enumerated (e.g., , cost, quality and schedule).

DESIGN ALTERNATIVE	DESIGN OBJECTIVES		
	COST	QUALITY	SCHEDULE
DESIGN A			
DESIGN B			
DESIGN C			

The design alternative with the highest worth is selected as the best option.

Otherwise ...

... use formal approaches to linear/nonlinear optimization.

Problem Formulations for System Optimization

Difficulties

- How to choose weighting coefficients in a rational manner?
- Preferences based on ecomomics alone may not reflect what and end-user really wants.

Problem Formulations for System Optimization

Method 2: Minimax Formulation

A second approach is to solve the following minimax problem:

$$\min_{r} \max_{i} [w_i f_i(x)] \tag{3}$$

where the w_i coefficients are selected as above.

Optimal Solution

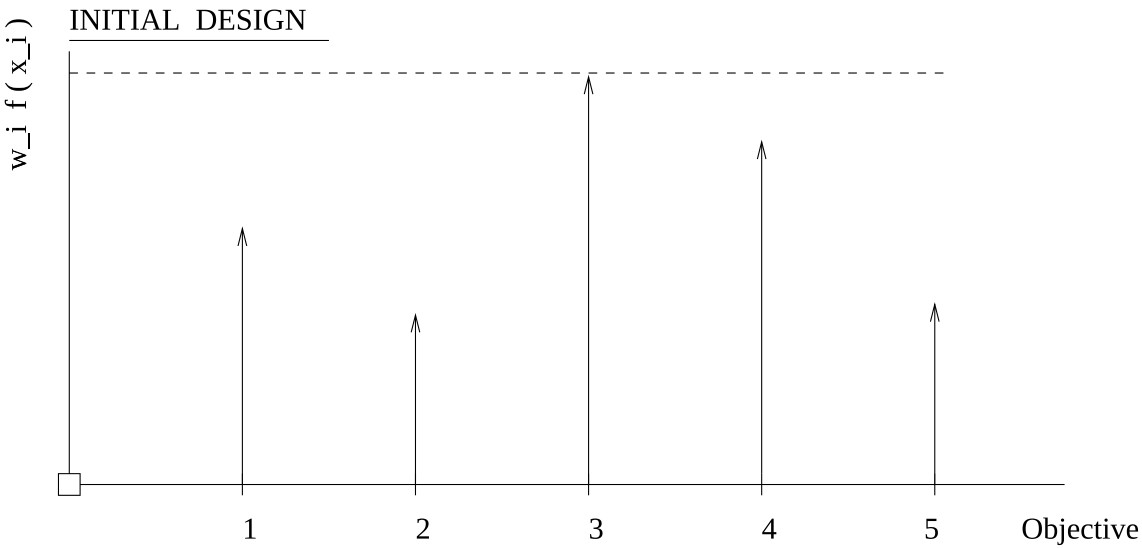
Typically the optimal solution x^* with involve a subset $\{i_k\}$ of the objectives where

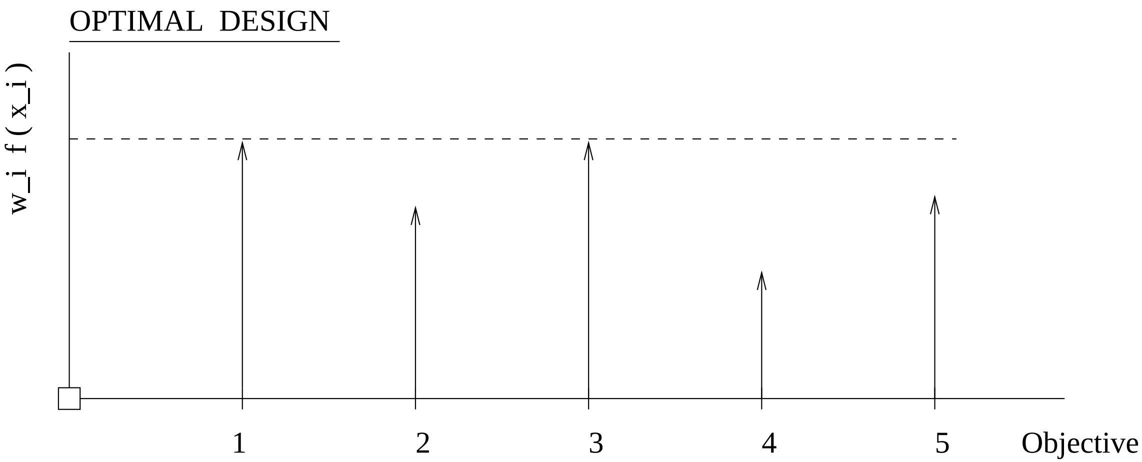
$$w_1 \cdot f_1(x^*) = \cdots w_s \cdot f_s(x^*). \tag{4}$$

with the other values of $w_i \cdot f_i(x^*)$ less than this value.

Problem Formulations for System Optimization

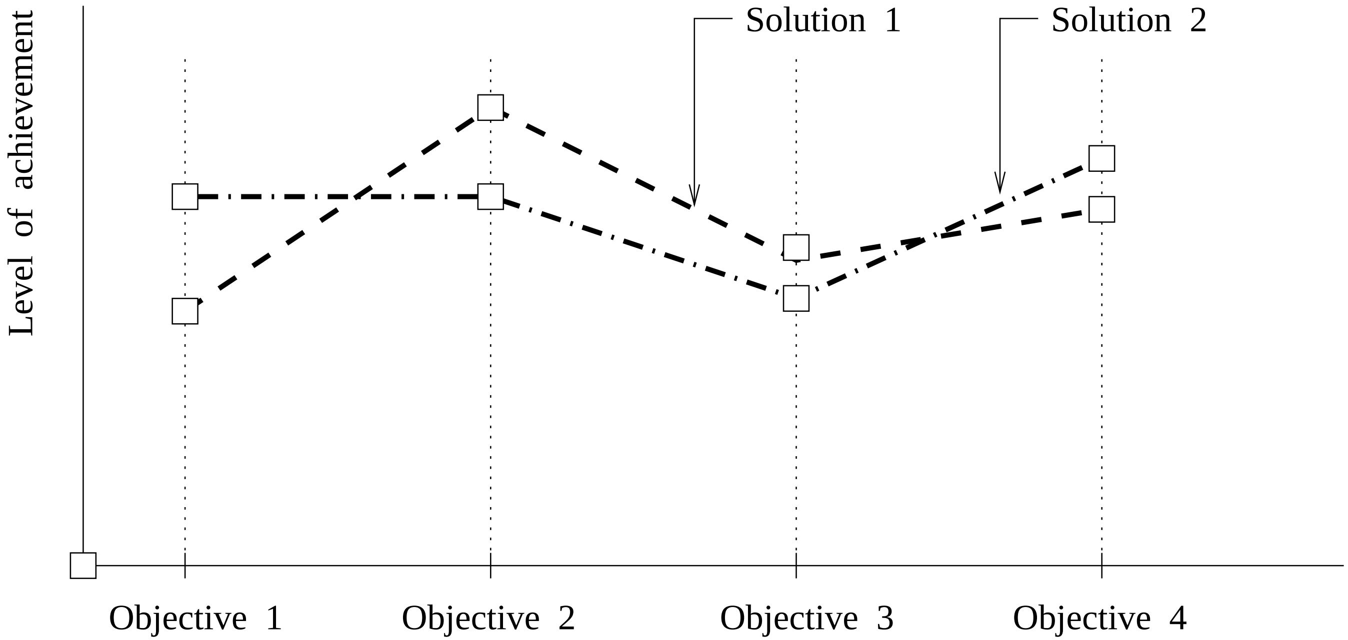
Initial and Final Designs for Minimax Formulation





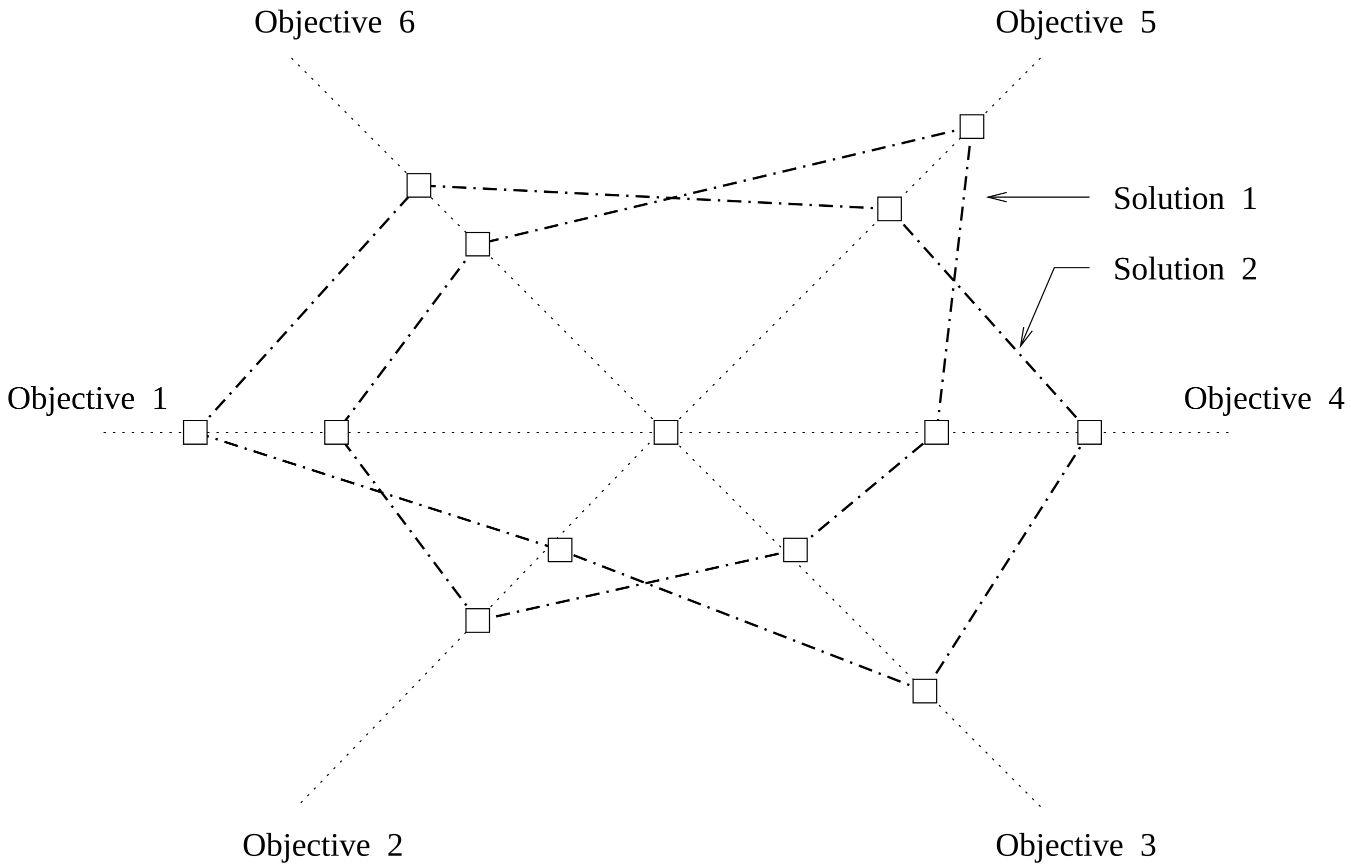
Visualization Techniques

Profile Display of MultiObjective Performance



Visualization Techniques

Star Display of MultiObjective Performance



Sets of Noninferior Solutions

Mathematical Definition

Given a set of feasible solutions X, the set of noninferior (or nondominated) solutions is denoted S and defined as follows:

> S = x : x $\in$ X, there exists no other $x^* \in$ X such that $f_q(x^*) > f_q(x)$ for some $q \in \{1 \cdots p\}$ and $f_k(x^*) \geq f_k(x)$ for all $k \neq q$.

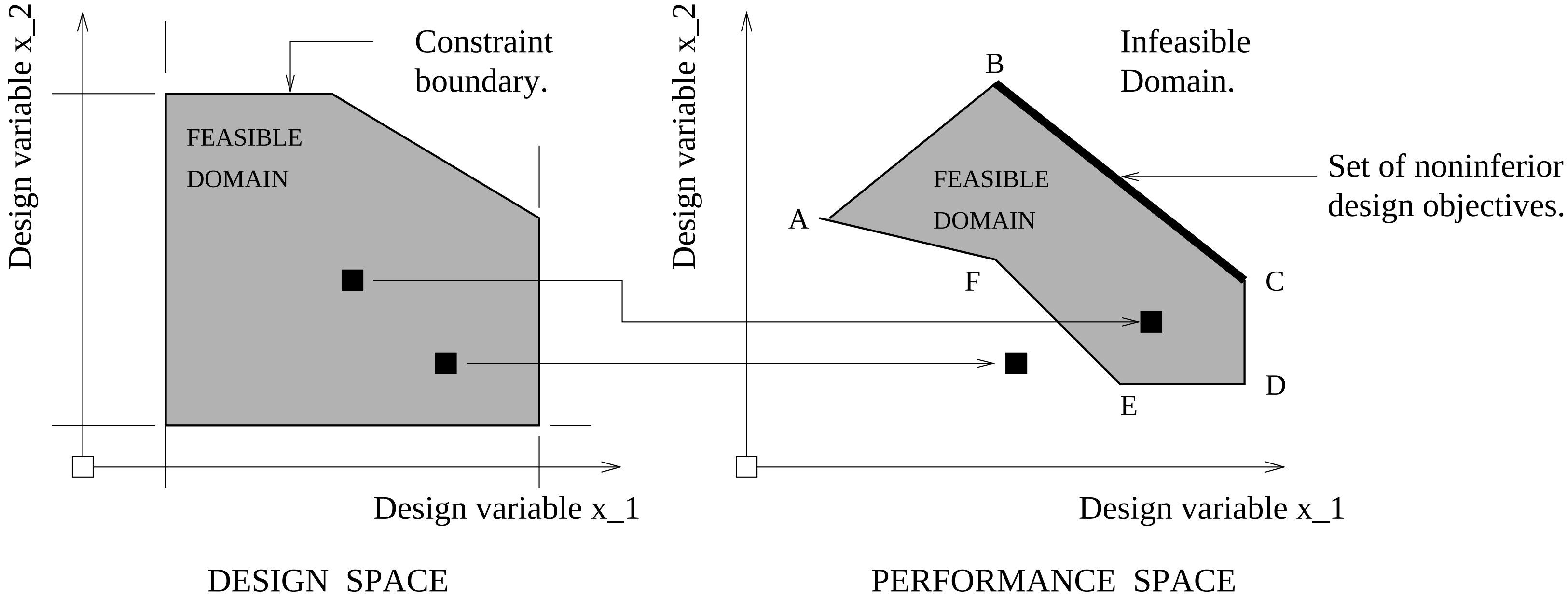
Plain English

- Let S be the set of solutions x for which we can demonstrate no better solutions exist.
- As one moves from one nondominated solution to another and one objective function improves, then ...

... one or more of the other objective functions must decrease in value.

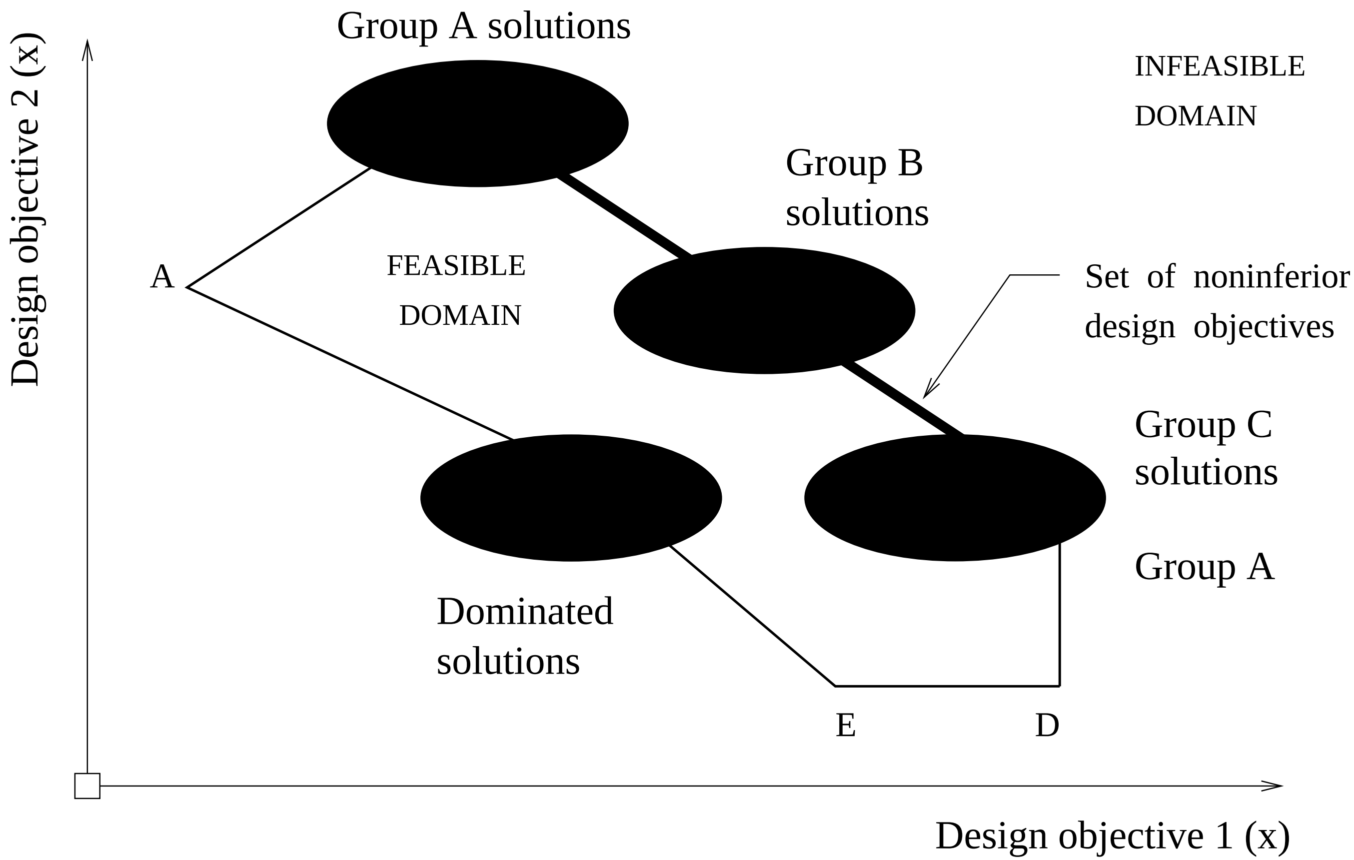
Sets of Noninferior Solutions

Optimization Design and Performance Spaces



Sets of Noninferior Solutions

Group Classification of Performance Space



Application 1. Two-Dimensional Problem

Problem Statement. Find the noninferior set for:

$$\begin{aligned} \text{Objective} &= [\, f_1(x),\; f_2(x) \,] \\ &= [\, x_1 - 3x_2,\; -4x_1 + x_2 \,]\,. \end{aligned}$$

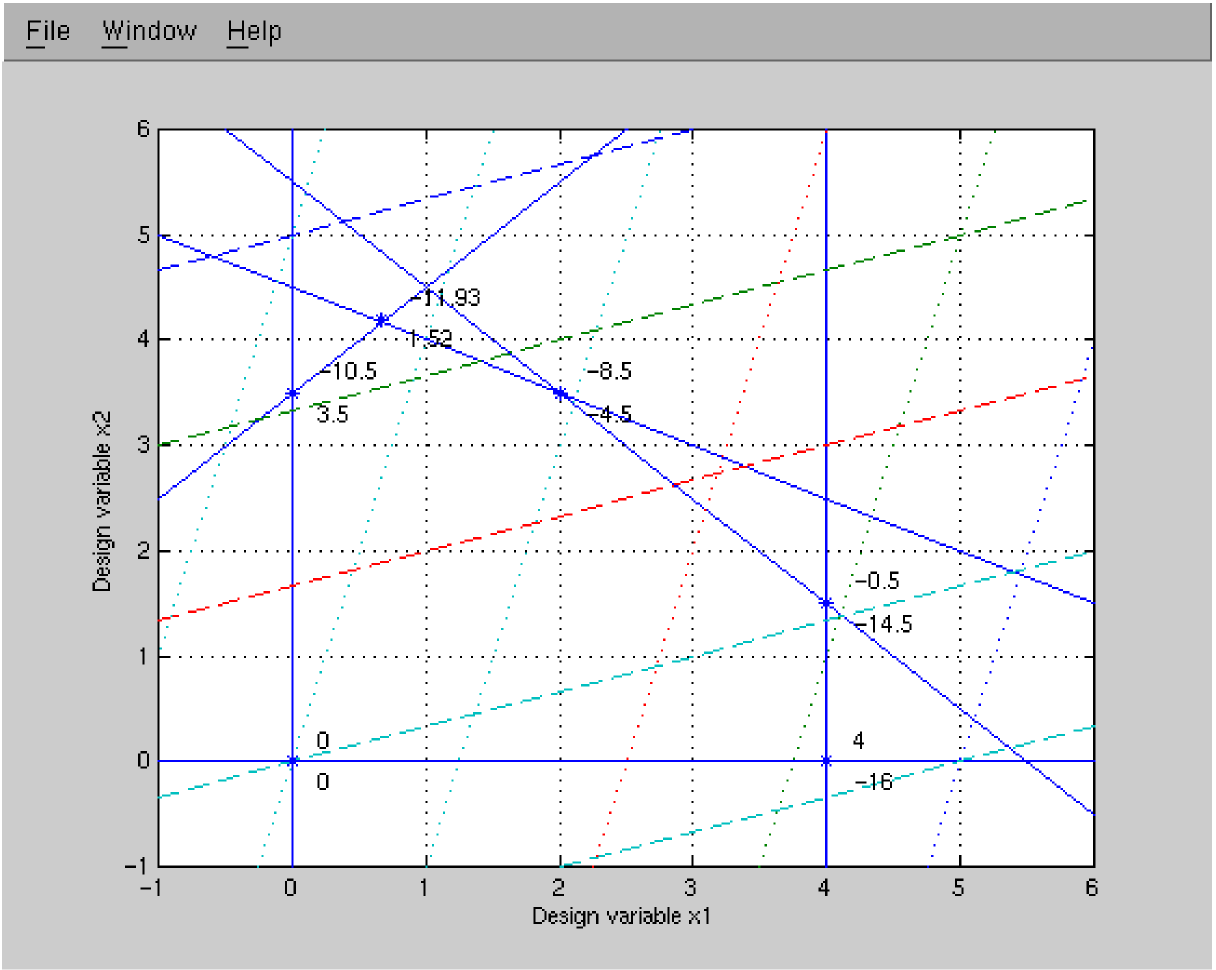
subject to the constraints:

$$\begin{aligned} g_1(x) &= -x_1 + x_2 - 7/2 \le 0 \\ g_2(x) &= x_1 + x_2 - 11/2 \le 0 \\ g_3(x) &= x_1 + 2x_2 - 9 \le 0 \\ g_4(x) &= x_1 - 4 \le 0 \end{aligned}$$

and $x_1 \ge 0$ and $x_2 \ge 0$.

Application 1. Two-Dimensional Problem

Feasible Domain and Level Sets for Objective Functions 1 and 2



Application 1. Two-Dimensional Problem

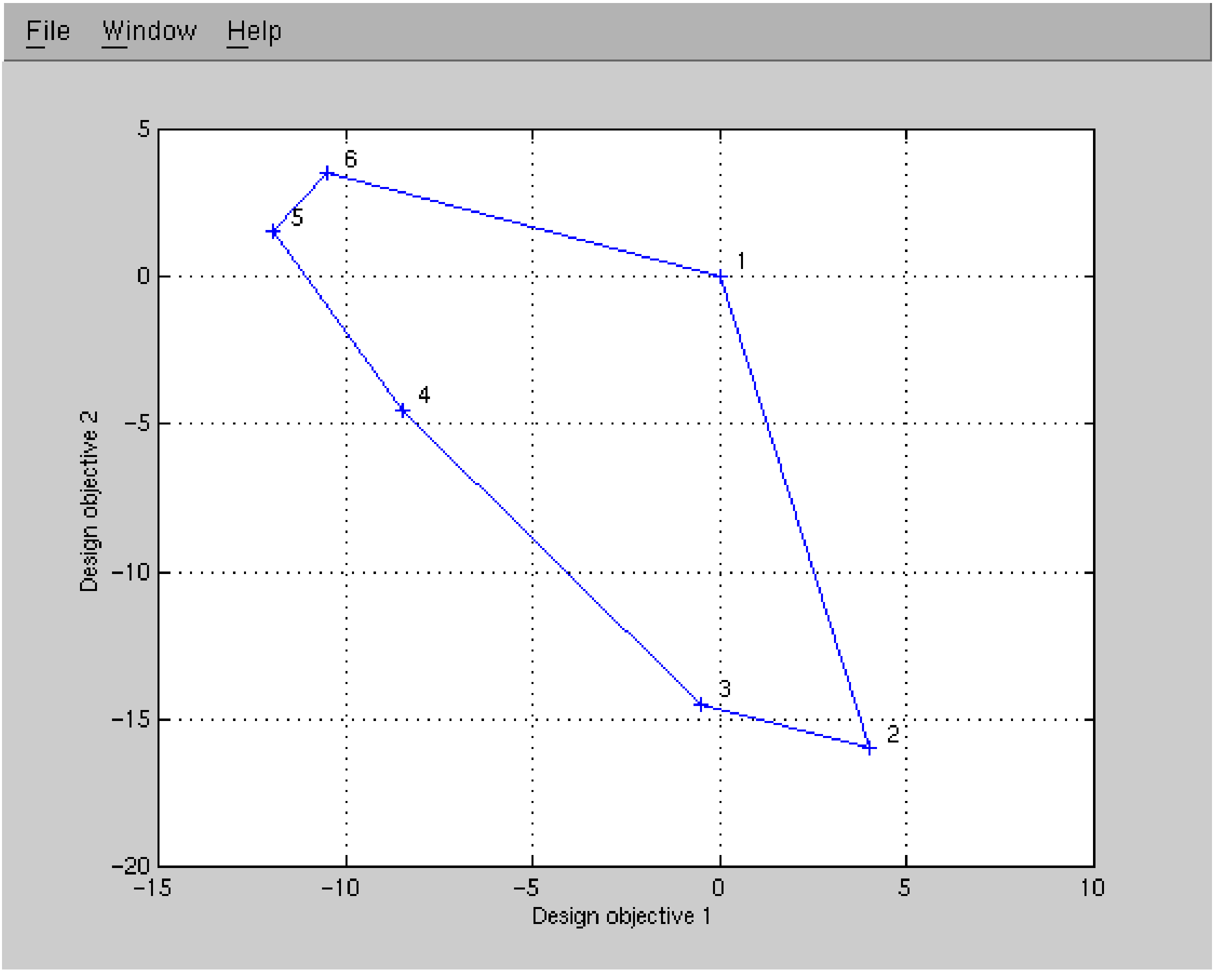
Corner Point Coordinates and Objective Function Values

The corner point coordinates and objective function values are as follows:

Corner Point	(x,y) coordinate	Objective 1	Objective 2
1	(0.0, 0.0)	0.0	0.0
2	(4.0, 0.0)	4.0	-16.0
3	(4.0, 1.5)	-0.50	-14.5
4	(2.00, 3.50)	-8.5	-4.5
5	(0.67, 4.20)	-11.93	1.62
6	(0.00, 3.50)	-10.5	3.5

Application 1. Two-Dimensional Problem

Design Objective View of Feasible Domain and Noninferior Set



Application 2. Architecture and Component Selection

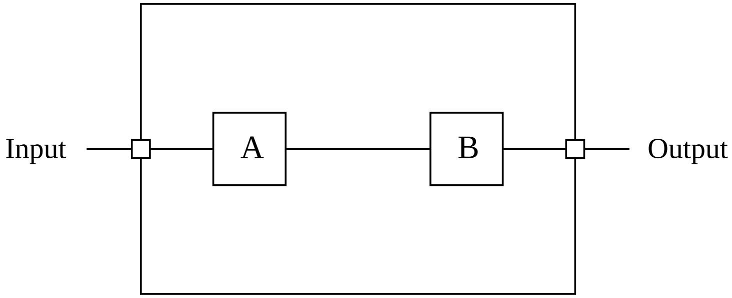
Problem Objective

We examine ...

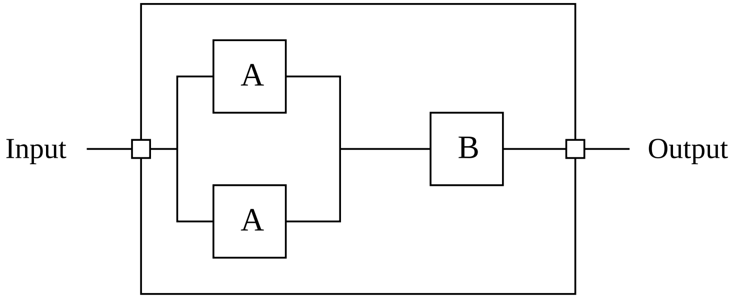
... tradeoffs in cost, performance, and reliability that occur when both the components and topology of component connectivity of a design can be selected.

Problem Setup

Architecture 1: Series Connectivity



Architecture 2: Mixed Connectivity



Properties of Architecture 1

From first principles of engineering we determine that:

$$\text{Architecture 1: Cost}\,(c_a, c_b) = c_a + c_b, \tag{5}$$

$$\text{Architecture 1: Performance}\,(p_a, p_b) = \min\,(p_a, p_b)\,, \tag{6}$$

and

$$\text{Architecture 1: Reliability}\,(r_a, r_b) = r_a r_b. \tag{7}$$

In equations 5 through 7, c_a and c_b are the costs of components A and B, p_a and p_b are the performance of components A and B, and r_a and r_b are the reliability of components A and B. min() is a function that returns the minimum value of the arguments, e.g., min(3,4) evaluates to 3.

Application 2. Architecture and Component Selection

Properties of Architecture 2

From first principles of engineering we determine that:

$$\text{Architecture 2: Cost}\,(c_a\,, c_b) = 2c_a + c_b\,, \tag{8}$$

$$\text{Architecture 2: Performance}\,(p_a\,, p_b) = \min\,(2p_a\,, p_b)\;, \tag{9}$$

and

$$\text{Architecture 2: Reliability}\,(r_a\,, r_b) = r_b\left(1 - (1 - r_a)^2\right). \tag{10}$$

Application 2. Architecture and Component Selection

Component Library

Let us assume that that there are two alternatives for component A:

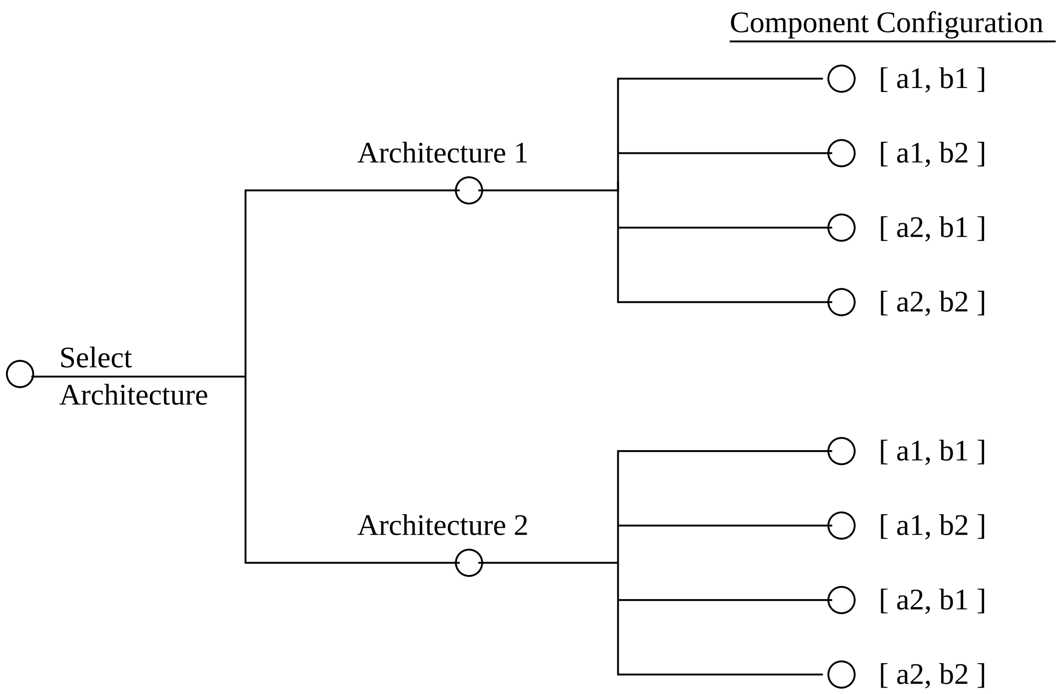
```
----------------------------------------------------------
Component Type A:     Cost    Performance      Reliability
==========================================================
Option a1:             2.0,            3.0,            0.8
Option a2:             4.0,            4.0,            0.9
==========================================================
```

and two alternatives for component B:

```
----------------------------------------------------------
Component Type B:     Cost    Performance      Reliability
==========================================================
Option b1:             5.0,            5.0,            0.8
Option b2:             7.0,            7.0,            0.9
==========================================================
```

Application 2. Architecture and Component Selection

Decision Tree and TradeOff Curves



First we need to select the system architecture, and then within that architecture, combinations of components that will minimize the system cost and maximize the system performance and reliability.

Application 2. Architecture and Component Selection

Cost, Performance, and Reliability in Architecture 1.

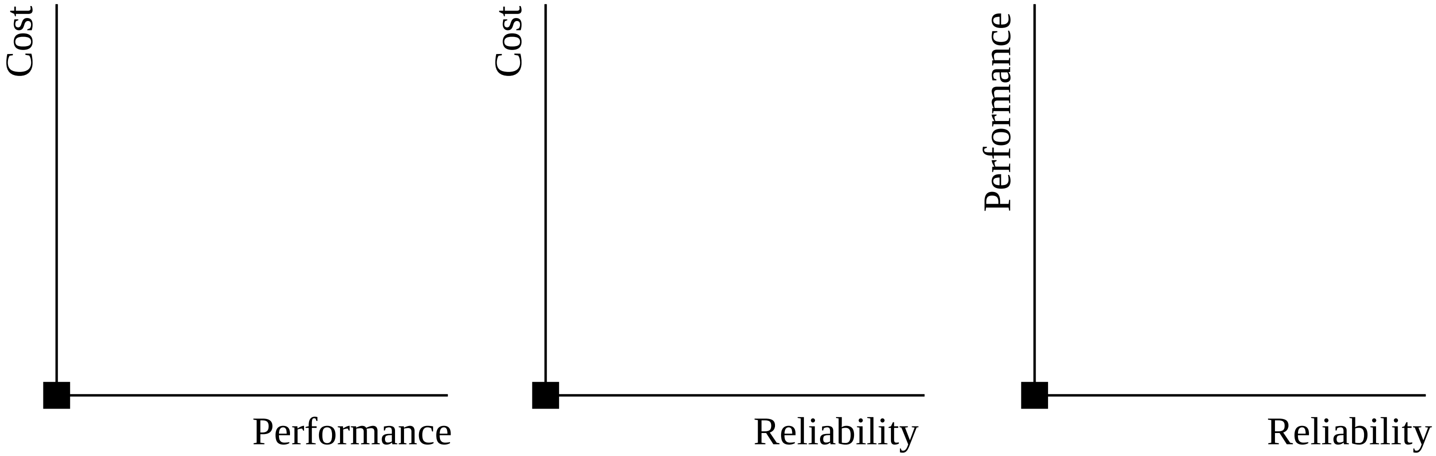
System Configuration	Component b_1			Component b_2		
	Cost	Perf.	Reliability	Cost	Perf.	Reliability
Component a_1	7	3	0.64	9	3	0.72
Component a_2	9	4	0.72	11	4	0.81

Cost, Performance, and Reliability in Architecture 2.

System Configuration	Component b_1			Component b_2		
	Cost	Perf.	Reliability	Cost	Perf.	Reliability
Component a_1	9	5	0.77	11	6	0.86
Component a_2	13	5	0.79	15	7	0.89

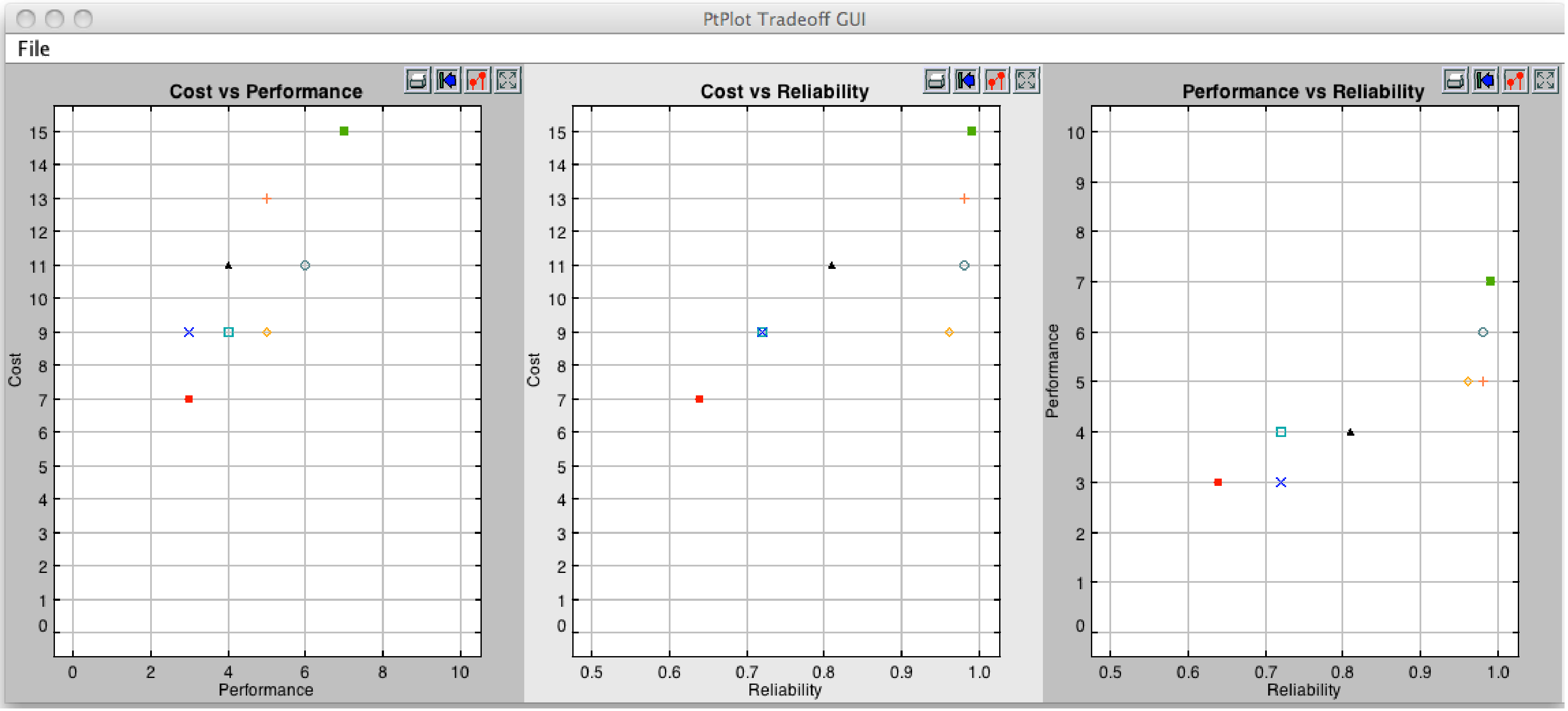
Application 2. Architecture and Component Selection

Identification of Non-Dominated Design Solutions



Application 2. Architecture and Component Selection

Screendump of TradeOff Software (Implemented in Java)



Application 2. Architecture and Component Selection

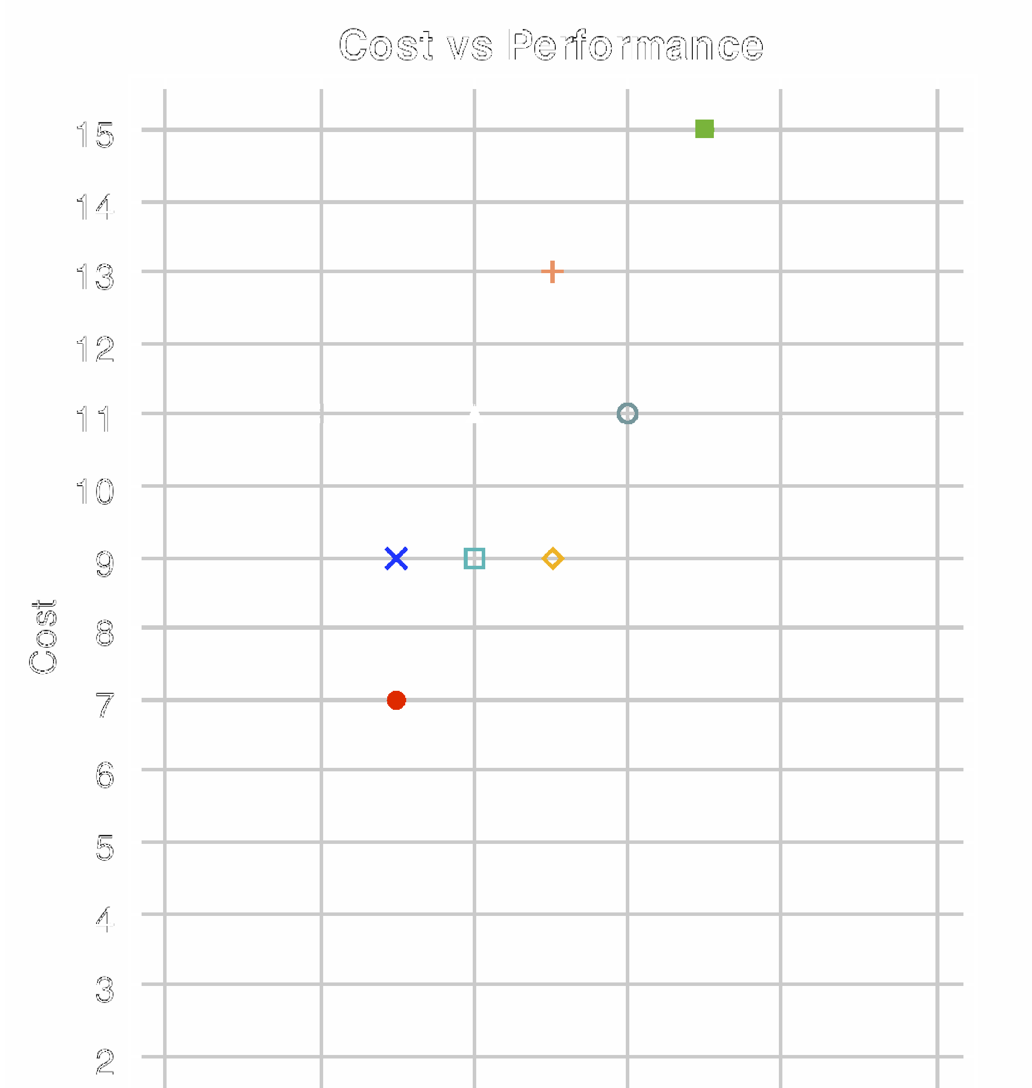
Tradeoff 1. Cost vs Performance

We wish to minimize cost and maximize performance. The Pareto optimal designs are:

```
Symbol                  Configuration    Component Selection
============================================================
Red dot.               Architecture 1             [ a1, b1 ]
Yellow diamond.        Architecture 2             [ a1, b1 ]
Cyan circle.           Architecture 2             [ a1, b2 ]
Green square.          Architecture 2             [ a2, b2 ]
------------------------------------------------------------
```

Application 2. Architecture and Component Selection

System Cost versus System Performance



Application 2. Architecture and Component Selection

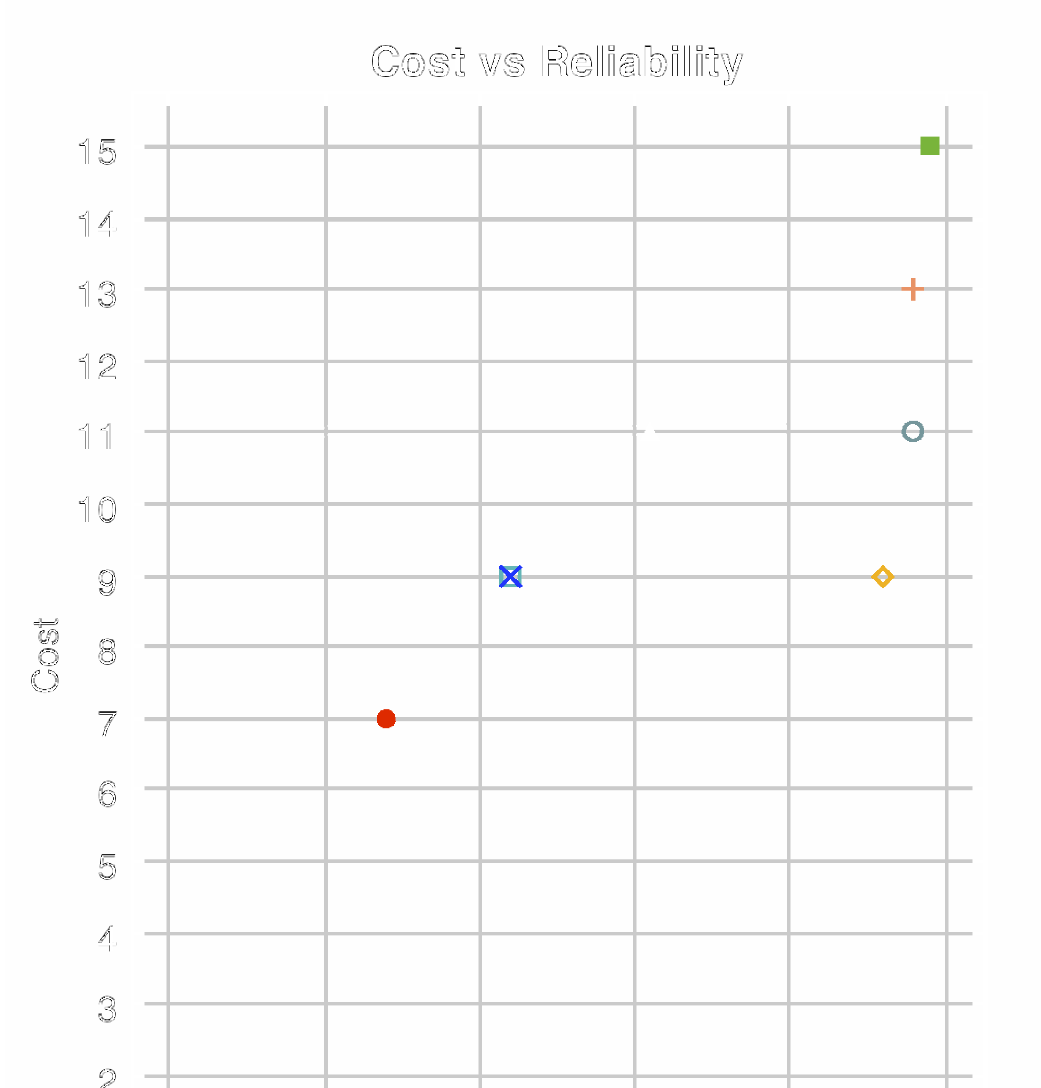
Tradeoff 2. Cost vs Reliability

We wish to minimize cost and maximize reliability. The Pareto optimal designs are:

```
Symbol                   Configuration    Component Selection
============================================================
Red dot.                Architecture 1           [ a1, b1 ]
Yellow diamond.         Architecture 2           [ a1, b1 ]
Cyan circle.            Architecture 2           [ a1, b2 ]
Green square.           Architecture 2           [ a2, b2 ]
------------------------------------------------------------
```

Application 2. Architecture and Component Selection

System Cost versus System Reliability



Application 2. Architecture and Component Selection

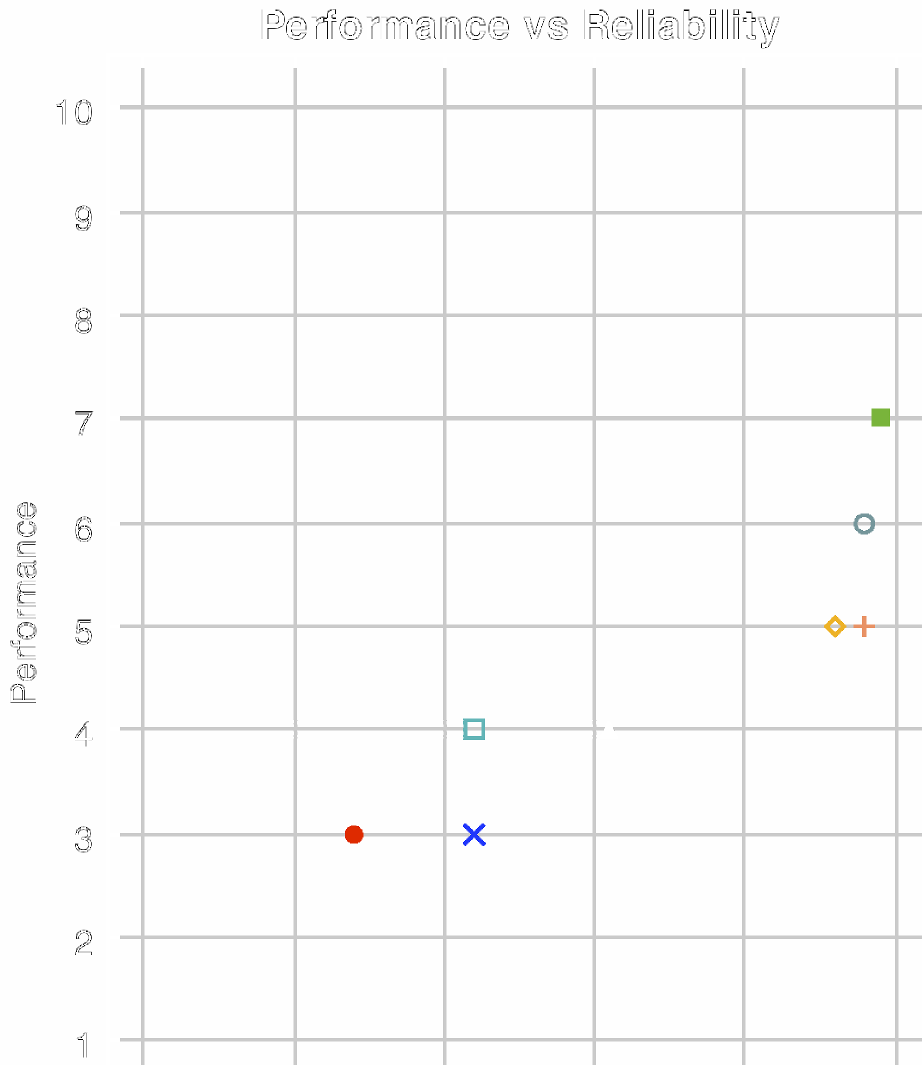
Tradeoff 3. Performance vs Reliability

We wish to maximize both performance and reliability. The Pareto optimal designs are:

Symbol	Configuration	Component Selection
Blue x.	Architecture 1	[a1, b2]
Cyan circle.	Architecture 2	[a1, b2]
Green square.	Architecture 2	[a2, b2]

Application 2. Architecture and Component Selection

System Performance versus System Reliability



Application 2. Architecture and Component Selection

Summary of Trades

1. The trade-space figures and the textual summaries for system configuration and component selection indicate that a system architecture and combination of component selections that is superior from all standpoints – cost, performance and reliability – does not exist.
2. Generally speaking both system performance and reliablity increase with system cost.
3. Architecture 2 is more expensive than architecture 1 because we use two A blocks instead of one. However, this allows for a refinement of the connectivity among components, which, in turn, improves the system level reliability.
4. Both the cyan circle (architecture 2; components a1 and b2) and green square (architecture 2; components a2 and b2) are part of the non-inferior design solutions in all three trade spaces.

Construction of Noninferior Design Solutions

Limitations of the Graphical Approach

The graphical approach to noninferior set identification ...

... works for problems having only two or three objectives.

Noninferior solutions for higher-dimensional problems can be computed by ...

... using the constraint method and the weighting method.

Both methods compute the set of noninferior solutions by ...

... transforming the multi-dimensional problem ...

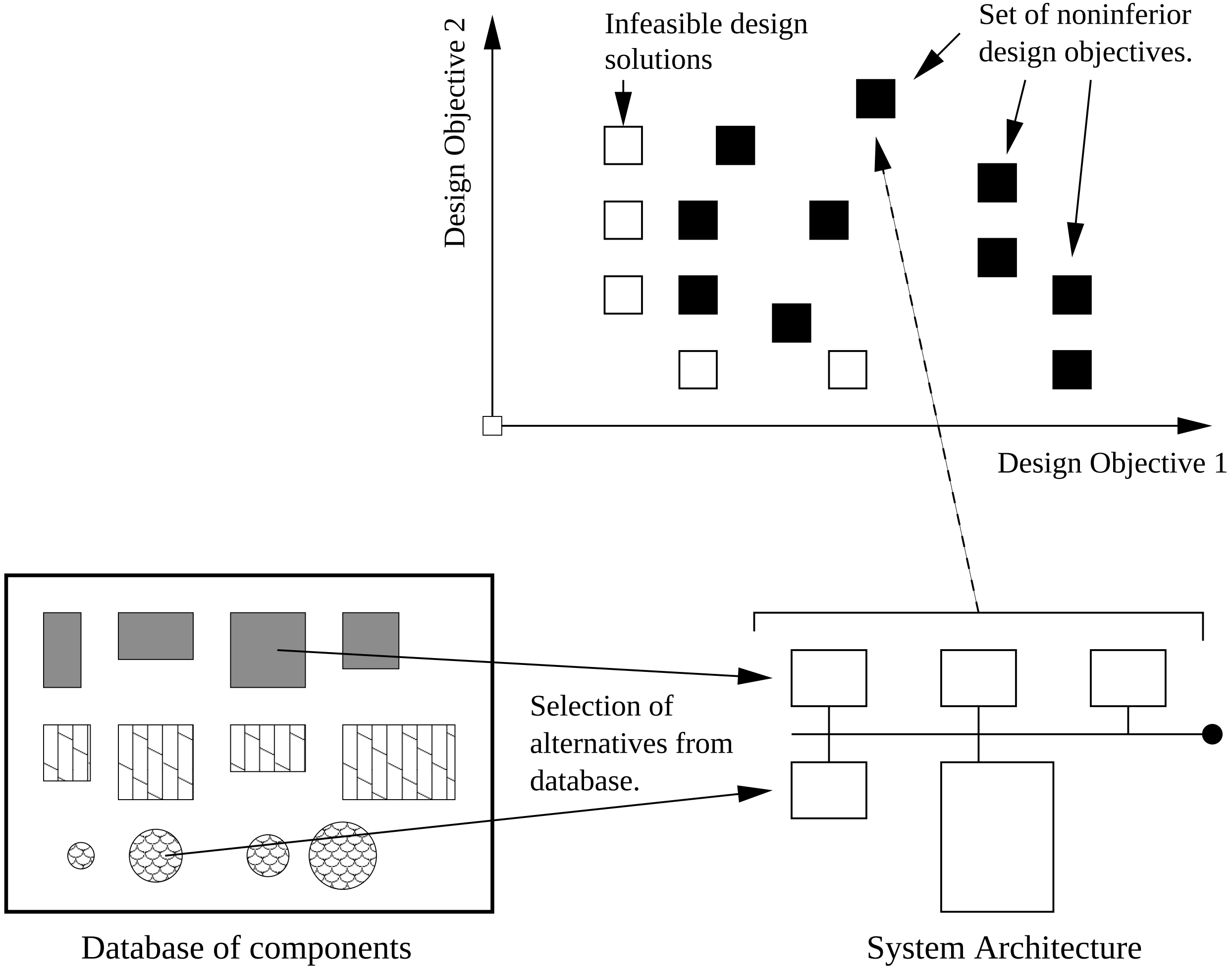
into

... a sequence of one-dimensional optimization problems.

Part 3. Tradeoff Analysis with Multi-Criteria Optimization Tools

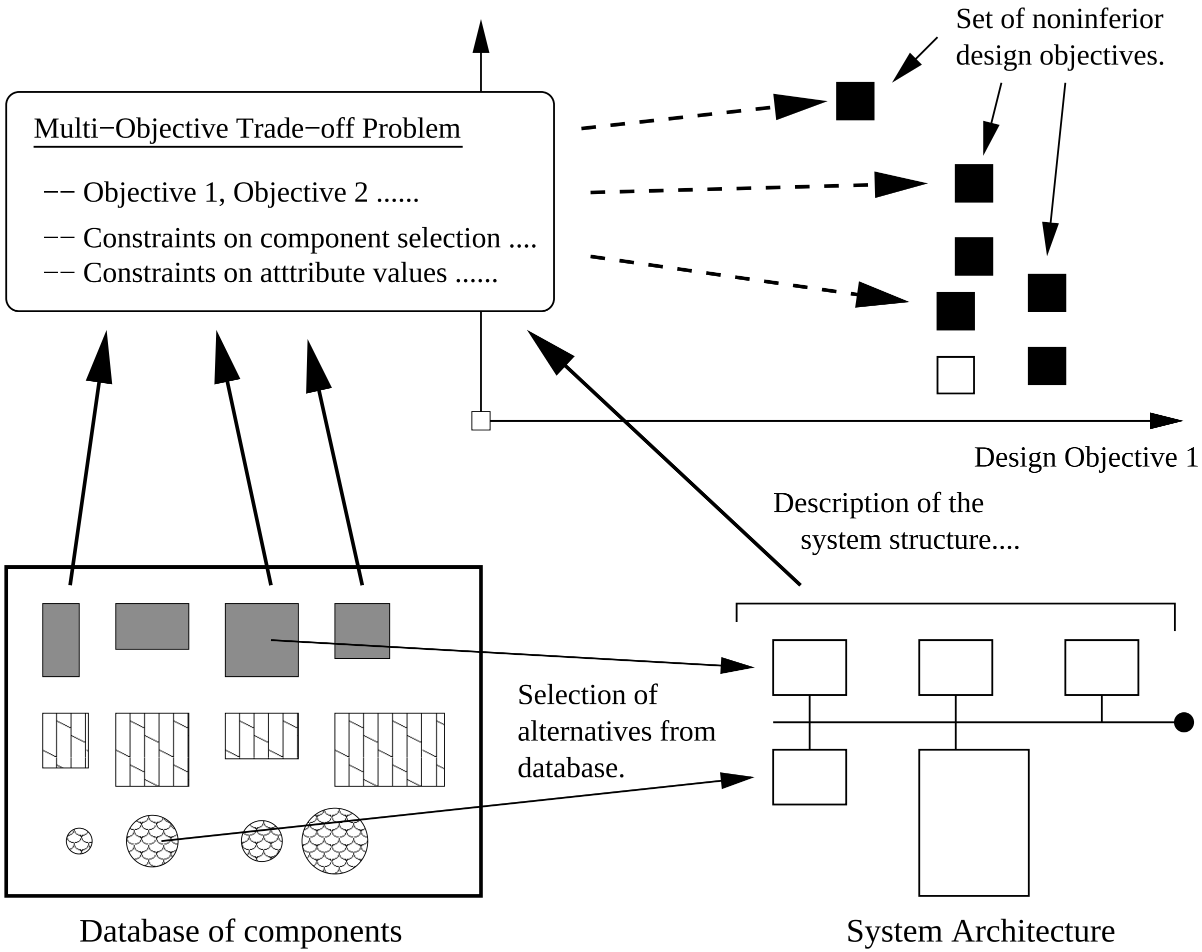
Generation of Designs from Component Alternatives

Method 1. Trial-and-Error



Generation of Designs from Component Alternatives

Method 2. Using Multi-Objective Trade-Off Analysis



Multi−Objective Trade−Off Formulation

Assignment-Type Problems

Assignment-Type Problems

Given N items and M resources, devise ...

> **... an assignment of items to resources such that a given cost function is optimized and ”K” restrictions are satisfied.**

The mathematical representation of ATPs is:

```
Minimize  F(x)  subject to:
          Sum  xij = 1 (1 <= i <= "N")
          G(x) <= 0 for k = 1 through "K"
          xij = 0 or 1 (1 <= i <= "N"; j in "J")
```

Here

- F(x) is the cost function ....
- G(x) are the imposed constraints ....
- "J" is the set of allowed resources for each item "i" ...

Assignment-Type Problems

Representation of Logical and Numerical and Specifications

Specifications can be numerical (e.g., 10 < x < 20) or logical (true/false).

Logical specifications can be converted to an equivalent numerical format, e.g.,

```
Select one of: Amplifier (A1), Amplifier (A2), Amplifier (A3),
               Amplifier (A4), Amplifier (A5), Amplifier (A6).
```

We can rewrite this problem as:

$$F(x) = x_1 A_1 + ... + x_6 A_6 \tag{11}$$

where

$$x_1 + x_2 + ... + x_6 = 1 \tag{12}$$

and x_i are constrained to be semi-positive integers (i.e., 0 or 1).