TEL AVIV UNIVERSITY

THE IBY AND ALADAR FLEISCHMAN FACULTY OF ENGINEERING

The Zandman-Slaner Graduate School of Engineering

Search and Selection of Concepts in Multi-objective Engineering Problems Using Evolutionary Algorithms

By

Gideon Avigad

THESIS SUBMITTED FOR THE DEGREE OF "DOCTOR OF PHILOSOPHY"
SUBMITTED TO THE SENATE OF TEL-AVIV UNIVERSITY

March 2007

TEL AVIV UNIVERSITY

THE IBY AND ALADAR FLEISCHMAN FACULTY OF ENGINEERING

The Zandman-Slaner Graduate School of Engineering

Search and Selection of Concepts in Multi-objective Engineering Problems Using Evolutionary Algorithms

By

Gideon Avigad

THESIS SUBMITTED FOR THE DEGREE OF "DOCTOR OF PHILOSOPHY"
SUBMITTED TO THE SENATE OF TEL-AVIV UNIVERSITY

This research work was carried out at Tel-Aviv University
in The Faculty of Engineering

Under the supervision of Dr. Amiram Moshaiov and Prof. Neima Brauner

March 2007

ACKNOWLEDGMENT

This work could not have been done without the support of several dear people to whom I would like to acknowledge.

For the philosophical talks, care, support and love I thank my mother. For the understanding, the freedom to practice science, the love and support, I thank my wife Noa. To Ahiram, Shoham, Odeya and Amots my beloved children who endured years of busy evening and mumbling on a genetic code.

Prof. Neima Brauner deserves special thanks for her care, support, assurance, and patience.

I thank Prof. David Maron for encouraging me to continue my studies beyond the Bachelor degree.

Last but not least I thank my supervisor Dr. Ami(ram) Moshaiov for being a supervisor, a guide and above all a friend. His dedication, encouragement, and impact are of vital importance to this thesis.

To my father who would have enjoyed it the most

DECLARATION

Some parts of the work presented in this thesis have been published in several articles. Additional articles, containing unique algorithms and approaches, which are not addressed directly within this thesis, were also published during the course of preparing this thesis. Some of them include pioneering procedures and algorithms. All of these articles could have been a part of the literature survey, but they are excluded from it in order to prevent confusion. All of these articles are considered as a part of this thesis or its background. A short description of these papers and an explanation on their contribution to the progress of the work may be found in the appendix.

ABSTRACT

This thesis is a result of a motivation to advance the computational tools, which are used to support designers during the conceptual design stage of a multi-objective engineering design problem. The thesis involves contributions concerning four major topics including: a. simultaneous concept-based evolution of concepts towards and along a Pareto front, b. interactive concept-based evolution towards optimal solutions of preferred concepts, c. assessment of concepts in the multi-objective space, and d. supporting decision making with uncertainties due to delayed decisions. It should be noted that, with respect to the first contribution, the problem of simultaneous mechanics and control design, at the conceptual design level, is of a particular motivation to this study. Together the contributions of this thesis advance the state-of-the-art of methods to search, compare and select solution concepts in multi-objective problems.

In this thesis the concept-based multi-objective optimization problem is studied and its distinction from the traditional multi-objective problem is discussed. The concept-based problem involves concepts that are represented by particular designs (solution sets) which are associated with the concepts. The main assumption is that these concepts have reached a stage in which models are available for their evaluation. As a part of the presented study, novel evolutionary algorithms are developed, using a simultaneous search approach, to solve the concept-based multi-objective optimization problem. In particular, the suggested algorithms address the issue of resource sharing among concepts, and within each concept, while simultaneously evolving concepts towards a Pareto front by way of their representing sets. The introduced algorithms are compared with a sequential one from two major aspects: the computational time and the quality of the front's representation.

Next, the concept-based multi-objective optimization problem is extended to include interactivity, which is the main motivation to the development of the simultaneous algorithms. The interactive concept-based problem involves both model-based optimality and the subjectivity of the designers. A novel interactive concept-based multi-objective algorithm is presented. The articulation of designers' preferences towards concepts and sub-concepts, as suggested in this thesis, establishes a new approach to the integration of preferences within evolutionary based algorithms.

The results of both of the above suggested approaches are associated with the representation of the concepts solutions' performances within a multi-objective space. Selection between concepts based on these representations is the next step. In this thesis a new approach to support such a selection is introduced. Both aspects of optimality and variability, which are associated with concept selection, are taken into account. Furthermore, the uncertainty towards

the preferences of objectives, which is inherent to multi-objectives problems, is also considered by the new approach.

In addition, this thesis deals with an uncertainty involving delayed decisions. Such a situation could result from a temporary lack of information during a conceptual design. Here, for the first time, the delayed decision problem is introduced in the context of a MOP. Moreover, a computational tool to support concept selection with the presence of such uncertainty is suggested. To attend this problem, the proposed new selection approach is adapted to allow the assessment of the relative performance of concepts, which involves robustness to delayed decisions.

The described concept-based techniques set the stage for a general approach to conceptual engineering design, for cases, with available models. Academic examples as well as engineering examples are used to study and demonstrate the suggested techniques including examples from structural mechanics and from mechatronics. The latter also serves to demonstrate the aptitude of the suggested techniques to deal with simultaneous mechanics and control design at the conceptual design stage.

TABLE OF CONTENTS

CHAPTER PAGE

ACRONYMS

ACC	Associated Complete Concept
CBF	Concept-Based Front
CC	Complete Concept
C-MOP	Concept-based Multi-Objective Problem
DM	Decision Maker
EA	Evolutionary Algorithm
EC	Evolutionary Computation
EMO	Evolutionary Multi-objective Optimization
FU	Fronts Union
GA	Genetic Algorithm
GP	Genetic Programming
HBF	Human-Based Fitness
HLC	High Level Concept
HMF	Human Machine Fitness
IC-MOP	Interactive Concept-based Multi-Objective Problem
IEC	Interactive Evolutionary Computation
IGA	Interactive Genetic Algorithm
ISE	Integral of Square Error
MBF	Machine-Based Fitness
MMC	Multi-Model Concept
MOEA	Multi-Objective Evolutionary Algorithm
MOP	Multi-Objective Problem
MOO	Multi-Objective Optimization
OSF	Objective-Subjective Front
O&V	Optimality and Variability
P	Proportional controller
PD	Proportional Derivative controller
PID	Proportional Integral Derivative controller
PDF	Probability Density Function
SBC	Set-Based-Concept
SC	Sub-Concept
SPVBC	Single-Performance-Vector-Based Concept
WOI	Window-Of-Interest

NOMENCLATURE

A_B	The cross section of the B-th truss' bar
An	'AND' node
b	Width dimension
C_m	The m-th CC's counter
$C_m^{k,z}$	Boolean variable for the m-th CC, with respect to the z division on the k-th objective
$CD_{j,k}^{i,m}$	Objective-based distance of the j-th individual in the list $I_k^{i,m}$,belonging to the m-th CC with respect to the k-th objective
$CD_j^{i,m}$	Concept-based crowding distance of the j-th individual in the list $I_k^{i,m}$,belonging to the m-th CC
$CD_{min,i}$	The minimal crowding distance over all $CD_j^{i,m}$ (for all m and j of the i-th rank) in the i-th rank.
CFO_m	Concept front occupation of the m-th CC
$D1_{j,k}^{i,m}, D2_{j,k}^{i,m}$	Crowding distances of the j-th individual of the m-th concept along the k-th objective in the i-th rank.
D_m	Distance between the most distant representatives of the m-th concept on the CBF
$D_m{}^*$	The m-th CC spread indicator
D_m^{max}	Maximum distance between solutions of the m-th CC, computed analytically
D_{OS}^m	Euclidian distance between the most distant representatives of the m-th CC on the OSF
E	Yung modulus
f ( )	Function of
f_k	The k-th objective function
f_k^m	The k-th objective function of the m-th CC
$f_k^m(x_m)$	The k-th performance of a solution belonging to the m-th CC

fit_U^i , fit_L^i	Upper and lower boundaries for the fitness of the i-th rank
F	Vector of objective functions
F^m	The m-th CC vector of objective functions
FP_{os}^*	Objective-subjective front's set
Fr_i	The i-th front set
Fr_i^m	The m-th CC solutions in the i-th front
F_1, F_2	Forces acting on the truss arrangement
FU	Union set of all CCs fronts sets
h	Height dimension
H_m	m-th CC-weight
i	Index for ranks/levels of non-dominance
I	Inertia
$I_k^{i,m}$	List of sorted solutions of the m-th CC, with respect to the k-th objective in the i-th rank.
$I_{j,k}^{i,m}$	The j-th solution in the $I_k^{i,m}$ list
I_{HMF}	List of sorted solutions according to their HMFs
j	Index for individuals
k	Index for objectives
K_C,K_I	Controller gain and reset time
K	Number of objective functions
L	Length of manipulator arm
$m_{s,i}$	Number of surviving CCs on the i-th front
m	Index for CCs and ACCs
m_A	Arm's mass
M_L	Load's mass
$\max f_k^{i,m}$	Max value of performances over all solutions on the i-th rank, belonging

	to the m-th CC with respect to the k-th objective
$\max f_k^i$	Max value of performances over all CCs of the i-th rank, with respect to the k-th objective
$\min f_k^{i,m}$	Min value of performances over all solutions on the i-th rank, belonging to the m-th CC with respect to the k-th objective
$\min f_k^i$	Min value of performances over all CCs of the i-th rank, with respect to the k-th objective
$\min(f_k^{WOI})$	Lower bound of the WOI with respect to the k-th objective
$\min(FU_k)$	Minimal value of the performances of the fronts union with respect to the k-th objective.
M	Control force applied on the manipulator arm
$MBF_{j,i}^m$	The MBF of the j-th individual of the i-th non-dominance level belonging to the m-th CC
$MBF^{max,m}$	Maximal fitness of an individual belonging to the m-th concept
MBF_{max}	Maximal MBF in a generation
n	Population size
n_c	Number of CCs in a C-MOP and the number of sub-population
n_L	Number of the children's nodes
n_{AAC}^a	Number of ACCs associated with the a-th MMC
n_m	Dimension of the vector x^m
n_{max}	Maximal sub-population's size
n_m^i	Number of the solutions of the m-th concept in the i-th rank
n_{gen}	Total number of generations in an evolutionary run
np	Number of deviations for the variability and optimality axes
n_r	Number of ranks in a generation
n_{OS}^m	Average number of solutions from the m-th CC on the OSF over predefined number of runs

no_m	Number of solutions of the m-th CC that belong to the OSF that are dominated by P_m^*
n_{MMC}	Number of MMCs in a MOP
N	Size of the vectors u, v.
On	'OR' node
O_m	Optimality of th m-th CC
P_t	Parent population in the t-th generation
PF^*	Pareto front set
PF_C^*	Concept-based front set
PF_{CC}^*	Combined front
P^*	Pareto-optimal set
P_{CC}^*	Combined concept-based set
P_C^*	Concept-based Pareto set
P_m^*	Pareto set of the m-th CC
P_{os}^*	Objective -subjective set
P_{uc}^*	Union of the CCs Pareto sets
PF_m^*	The m-th CC Pareto front set
Q_t	Offspring population at generation t
r_c	Rank counter
R_t	Combined parent and offspring population in the t-th gneration
R^n	Space with dimension n
R^{n_m}	Space dimension n_m for the m-th CC
R^K	A space with dimension K
Rank(x)	Rank assigned to a solution x according to its level of non-dominance within the set X

$sol_j^{i,m}$	the j-th individual in the i-th front belonging to the m-th CC
sol_j^{HMF}	The j-th solution in the sorted list I_{HMF}
S	Search space
SDn_{OS}^m	Standard deviation, of n_{OS}^m
S_m	Search space of the m-th CC
SDD_{OS}^m	Standard deviation of the Euclidean distance between the boundaries of the CC found for each CC on the OSF over ten runs
t	Index for generations
t_{final}	Set simulation time
T_c	Total computational time for running an EC
T_{sq}	Computational time needed to evolve a CBF by the sequential approach
T_{sm-1}	Computational time needed to evolve a CBF using C_1-NSGA-II
T_{sm-2}	Computational time needed to evolve a CBF by C_2-NSGA-II.
u, v	Vectors
V_m	Variability of the m-th CC
VO_a^{worst}	Set of the worst performances associated with the a-th MMC
VO_m^a	The O&V values of the m-th ACC associated with the a-th MMC
VO^a	Set of all O&V values ACCs, which belong to the a-th MMC
w(pr)	Weight assigned to a parent node
w(ch)	Weight assigned to a children node
w(root)	Weight assigned or computed for the root node of the 'AND/OR' tree
W_s	Weight assigned to the s-th SC
x	Vector of decision variables
x^*	Optimal vector of decision variables (optimal solution)
x^m	m-th set for the m-th CC of all its feasible solutions
x_m	A solution of the m-th CC
X	Feasible set of solutions in the search space S

X_m	Vectors of decision variables of the m-th CC
X_t	Set of decoded vectors of solutions in the t-th generation
x^*_{uc}	A solution which belongs to P^*_{uc}
y	Performances of a solution
y^*	Performances of an optimal solution
Y	Set of performances of all x belonging to X
Y_t	Set of vectors of performances of X_t in the t-th generation

Greek

τ	Time ratio
τ_D	Controller derivative gain
η	Generation-based tuning factor
Δf	Deflection of the end-effecter.
θ	Angle between the axis of a robotic arm and the x-axis
θ_{SP}	Set point for the robotic arm angle
ε	Constant that separates between adjacent ranks
ρ	Density
Ψ	Utility function for the interactive concept-based problem

LIST OF FIGURES

FIGURE		PAGE

FIGURE		PAGE

FIGURE		PAGE

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Engineering conceptual design is perhaps the most crucial task in product development cycle (Bullinger *et al*., 1998). The significance of correctly choosing a concept has been reflected in an increasing effort to develop methodologies and computational tools to support concept selection. In particular, computer-supported methods are required to help designers during the conceptual design stage of Multi-Objective Problems (MOPs), which are common in engineering design (Mattson and Messac, 2005).

MOPs exist in a vast number of engineering and scientific applications (Coello, 2005). Solving such problems, which involve multiple and often conflicting objectives, is generally considered as a difficult problem. Evolutionary Algorithms (EAs) in general and Evolutionary Multi-Objective (EMO) algorithms specifically, possess several characteristics, which make them suitable for solving this type of problems (Zitzler *et. al*., 2003,). The main motivations for using an evolutionary approach to solve MOPs are: a. the possibility of EAs to simultaneously deal with a set of possible solutions, b. their ability to cope with hard problems involving many, mixed discrete/analogues design parameters, and c. EAs are less susceptible to the shape or continuity of the Pareto front (which are hard problems for mathematical programming approaches).

This thesis deals with a special type of MOPs, which are hereby termed Concept-based MOPs (C-MOPs). The concept-based approach is motivated by the way humans, such as engineers solve a problem (e.g., Mattson and Messac, 2005). According to this approach, a concept is an idea that reached a point where a model is available to calculate the performances of a family of solution alternatives, which represent the concept. In other words, a concept has multiple representations in the decision variable space, and consequently, it involves a set of associated points in the objective space. For example, consider two conceptual solutions to the problem of moving an object from one location to another. The first conceptual solution is a conveyor and the second one is a manipulator. The Decision Makers (DMs) may select a particular solution from either the first or the second concept.

Traditionally designers select a concept and only then search for a particular solution of the selected concept. If none of the preliminary solutions of the selected concept meets the design requirements, a new concept has to be chosen and the procedure is re-started. This iterative serial procedure is schematically represented in figure 1.1a. In figure 1.1a it is implicitly suggested that the evaluation of preliminary solutions is commonly done with the aid of computers whereas the selection of the concept is primarily done by humans. A major drawback of this tradition is the early exploitation of a successful concept that may hinder better concepts. Moreover this serial approach does not fit a Pareto-based approach to MOPs. The Pareto approach aims at representing a set of solutions to the DM for selection while the traditional approach guides the search towards a single solution.

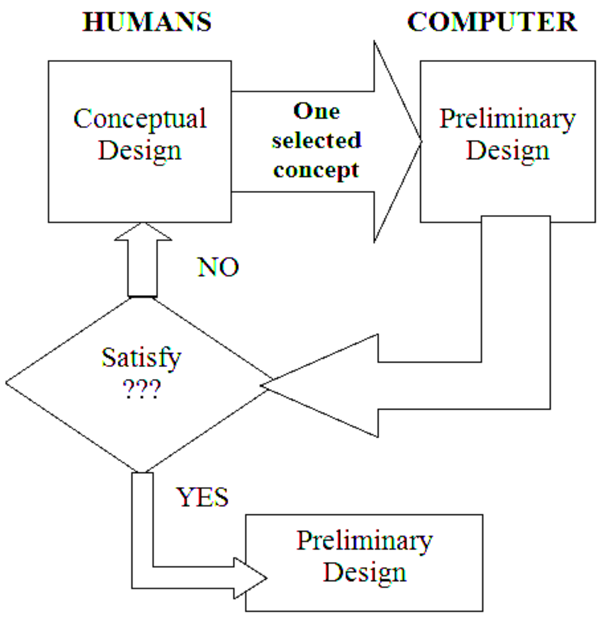


Figure 1.1a: The traditional approach to conceptual design

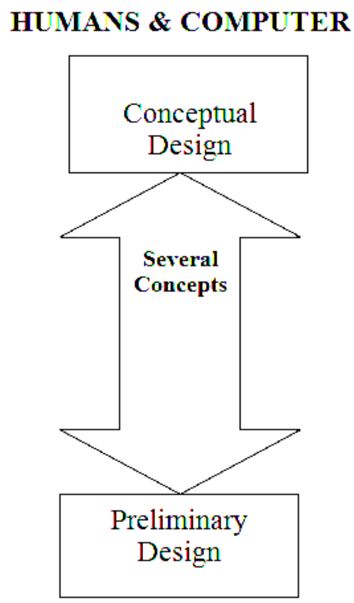


Figure 1.1b: Proposed approach to conceptual design

In this thesis, and similar works that deal with C-MOPs, such as Mattson and Messac (2003), a change of this traditional serial approach is assumed. According to the approach, which is adopted in this thesis, the selection of the final (particular) solution is supported by simultaneously comparing the performances of all the concepts by way of their associated particular solutions in the objective space. This simultaneous procedure, which is schematically depicted in figure 1.1b, ensures that the designers perform a multi-objective comparison of concepts and apply a search using a pressure towards a Pareto front with potentially obtaining several 'optimal' concepts (see definition in 3.2.1.2).

Classical algorithms of Evolutionary Multi-objective Optimization (EMO), such as the multi-objective genetic algorithm (Fonseca and Fleming, 1993), deal with problems in which each solution is treated as the basic member of the entire feasible set of solutions. In contrast, in the concept-based EMO (C-EMO), which is addressed here, the feasible set of solutions is divided into sub-sets of solutions (two in the above conveyor-manipulator example). Each such sub-set, which may belong to a different design space, represents a conceptual solution to the problem, or in short a 'concept.' Solving a C-MOP means finding all the *'optimal' concepts*, where each such concept has at least one member of its sub-set being a non-dominated solution with respect to the entire feasible set of solutions.

Figure 1.2 depicts (left side) three sets of solutions belonging each to a different concept. Let these sets constitute the entire design space. Different symbols, including circle, triangle and square, are used to designate the three concepts (concept 1, concept 2 and concept 3 respectively). For each of the solutions, the corresponding performances within a bi-objective space, are depicted in figure 1.2 (right side), marked by their concept-related symbols. Assuming a min-min problem, concepts 1 and 3 possess solution/solutions with performances, which are non-dominated with respect to the entire feasible set of solutions. Therefore, concept 1 and concept 3 are 'optimal' concepts. Concept 2 is not associated with any solution, which is non-dominate and therefore it is a 'non-optimal' concept.

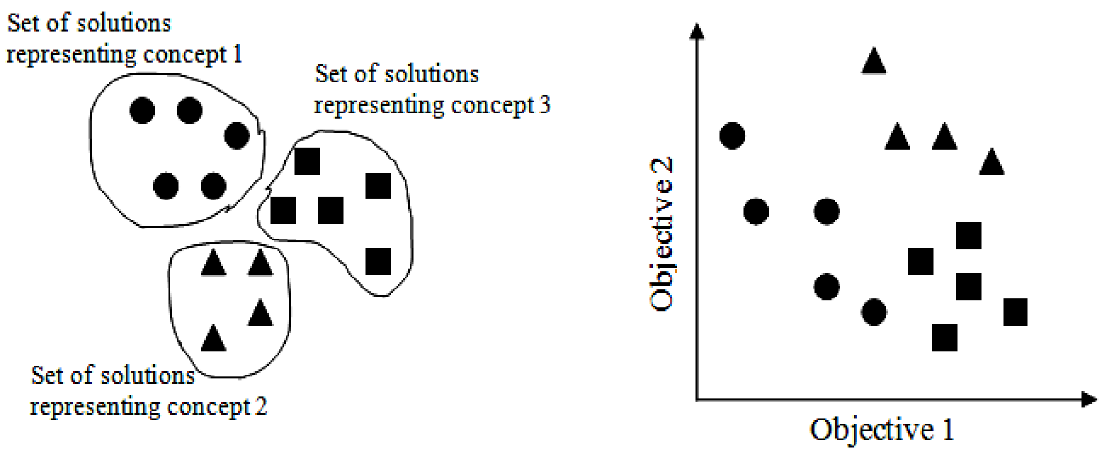


Figure 1.2: Optimal and non-optimal concept sets and their performances

In a C-MOP the assessment of concepts is done just by considering their computed performances. However, in ‘real-life’ design, models to assess the performances of conceptual solutions are not always available. Moreover the computation of merits might only partially reflect all issues that are involved in selecting a concept. Difficulties in realizing solutions associated with a particular concept, might not be modeled (e.g., difficulties of manufacturing and design problems). Therefore, humans rely on their experiences and preferences in choosing a conceptual solution, which is later realized by choosing a particular solution out of the concept's solutions.

In an interactive human-computer search, the advantages of the computer as a computational tool and the ability of humans to assess un-modeled ideas should be utilized. Commonly, the motivation for interactivity within MOPs is either to assign weights to objectives to aggregate them into a single function (e.g., Van Veldhuizen, 1999), or to direct the search towards preferred regions of the Pareto front (e.g., Cvetkovic', 2000). A systematic approach to direct the search towards preferred solutions has also been recently suggested by Deb and Chaudhuri, (2005). These interactivity approaches reduce the search space and by that efficiently use the available resources to direct the search to preferred regions. In this thesis a new type of interactivity is suggested as detailed in the methodology.

The problem in the concept-based search is not restricted to choosing a solution out of a set as in common MOPs, but moreover, is to initially choose the set (concept) itself. According to recent surveys (e.g., this thesis survey, Matson and Messac, 2005), there are just a few approaches, which are measure-based, to compare between sets of performances. The measures are primarily used to compare how well different optimization approaches generate Pareto sets (e.g., Van Veldhuizen, 1998).

When dealing with concepts selection, it is important to take into consideration the affects of uncertainties associated with the concept and with its particular solutions. Such uncertainties may be associated with the models of the concepts and the ability to realize their solutions by the available machinery and equipment (e.g., Mattson and Messac, 2002, Andersson, 2002). The conceptual design stage may be further associated with some unique uncertainties such as the delayed decision uncertainty. This uncertainty is associated with a need to postpone decisions within the conceptual design stage. Such a need is well known (e.g., Sobek and Ward, 1996). It could result from a lack of information to make an intelligent decision at that stage (e.g., availability of a particular manufacturing resource). When delaying a decision on certain parts of the involved concepts, the designers attempt to make a progress on other parts of the concepts. To elucidate the delayed decision problem, the following example is given. Suppose that the design concerns a manipulator to move an object from one location to the other. It involves conceptual decisions on the manipulator links and their control. Suggested concepts may include 'two-aluminum links controlled by a fuzzy controller' or 'one-steel-link controlled by a PID controller.' A delayed decision situation may occur if the designers are not certain about the availability of a proper hardware for implementing the fuzzy controller. However, they need to continue with conceptual decisions on the rest of the design of the links to ensure meeting the design deadline, e.g., ordering materials and preparing production infrastructure.

This thesis aims at advancing the existing computational methods that support searching, comparing, and selecting of concepts, which are represented by sets of particular solutions, in MOPs. The advances are associated with a. simultaneous development of concepts towards optimal concepts' solutions, b. interactivity towards concepts and sub-concepts, c. selection of concepts while taking into consideration different objectives of the conceptual design, and d. supporting decisions with the presence of uncertainties, resulting from delayed decisions.

The thesis is ordered as follows: Chapter 2 provides a general background to the main issues that this thesis is concerned with, such as conceptual design and MOPs. It also describes the State-Of-The-Art (SOTA) of the relevant particular topics and states the objectives of the thesis base on the SOTA shortages. Chapter 3 is concerned with the methodology, while chapter 4 includes case studies to demonstrate and study the methodology and algorithms. These include academic as well as engineering examples. Chapter 5 summarizes the thesis work, discusses its contributions with respect to the motivations, and suggests some directions for future research.

CHAPTER 2

LITERATURE SURVEY AND THESIS OBJECTIVES

This survey starts with section 2.1, which gives a general background on some issues that are important for understanding the SOTA and the contributions of this thesis. These issues include: engineering and mechatronic design, conceptual design, multi-objective problems and design space decomposition. The next section, 2.2, gives a background on evolutionary search and its applications in engineering design. This section includes brief reviews on genetic algorithms and their use for multi-objective optimization. In addition it includes a SOTA concerning resource sharing in single and in multiple objective problems. Moreover, section 2.2 outlines the use of evolutionary approaches for engineering designs. Section 2.3 reviews the search and selection approaches for three different notions, which are associated with concepts, including set-based concept, single-solution's performances-based concept and the notion of a family of designs. Section 2.4 reviews different methods for the articulation of preferences in evolutionary-based search including a-priori, progressive and a-posteriori methods. Section 2.5 surveys the treatment of robustness with respect to the three different notions of concepts that are described in section 2.3. The last section, 2.6, highlights the apparent shortages of the existing SOTA and consequently states the thesis objectives. Table 2.1 is provided to support focused reading and to highlight the relations between the issues of this thesis, the SOTA, the methodology, and the examples.

Table 2.1: SOTA and thesis aspects

Main issues	SOTA	Methodology	Examples
Simultaneous evolution	2.1.2; 2.1.3; 2.2.1- 2.2.4, 2.3	3.1; 3.2	4.1
Interactive concept- based evolution	2.1.2-2.1.4; 2.2.2-2.2.4; 2.4	3.1; 3.3	4.2
Supporting concept selection	2.1.2- 2.1.4; 2.2.2; 2.3; 2.5	3.4	4.3
Supporting decision making with delayed decisions uncertainties	2.1.2- 2.1.4; 2.2.2; 2.3; 2.5	3.5	4.4
Engineering applications	2.1.1-2.1.3; 2.2.2; 2.2.5;	-	4.1 - 4.5

In the table, the first column lists the main issues of this thesis. In the following three columns including literature survey, methodology and examples, each block indicates the associated (relevant) sections' numbers.

2.1 General Background

2.1.1 Engineering and mechatronic design

Engineering design is a major research field. This is reflected by its numerous books and research studies (e.g., Ullman, 1992; Pahl and Beitz, 1996; Ulrich and Eppinger, 2000). In this section just a few issues of relevance are shortly surveyed and discussed, as seems necessary for the completeness and clarity of this thesis.

As briefly explained in the introduction, a design process commonly begins with conceptual design, followed by preliminary and detailed designs. In such a procedure designers select a concept and only then search for a particular solution of the selected concept. If none of the preliminary solutions of the selected concept meets the design requirements, a new concept has to be chosen and the procedure is re-started. When a suitable preliminary design is found a detailed design stage commences, as depicted in figure 1a of the introduction chapter.

Mechatronics, which is related to the motivation for this thesis, is a relatively new engineering field that synergistically integrates mechanical engineering, electrical engineering, and software engineering. It is beyond the scope of this thesis to provide a literature review on mechatronics. The interested reader is referred to a comprehensive survey on mechatronics, including simulation packages, which was done by Diaz-Calderon (2000), as well as to existing journals such as Robotics and Mechatronics, The ASME/IEEE Transactions on Mechatronics, and related conferences. See also on-line: http://www.eng.morgan.edu/~salimian/courses/mechatronics/resources.html, and similar websites. The main engineering areas, which are associated with the design of a mechatronic artifact, are mechanics and control. Traditionally such a mechatronic design follows a sequential iterative procedure, which is schematically shown in figure 2.1.

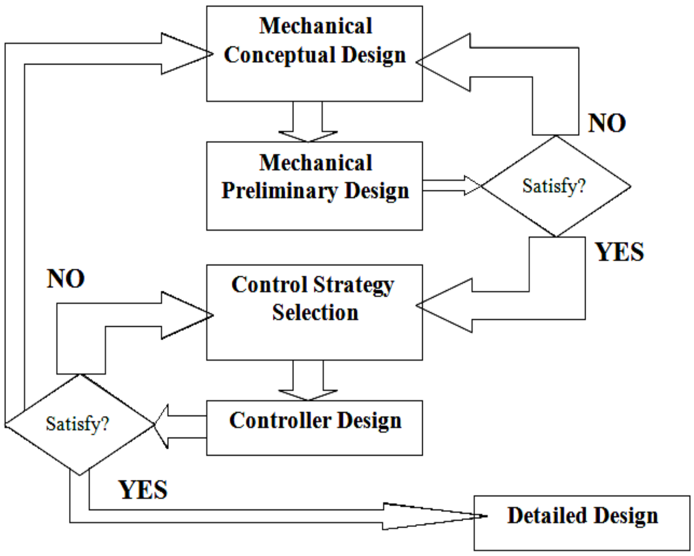


Figure 2.1: Traditional iterative mechanical-control design procedure

The first iterative procedure concerns mechanical design at the conceptual-preliminary levels. When a satisfying preliminary design is found the control has to be considered. Initially a control strategy has to be selected, followed by the controller design. The suggested controller has to be tested against the dynamic and steady state requirements. Moreover the robustness of the controller to the various uncertainties has to be checked (see section 2.5 for more details). If the controller does not meet these requirements a new controller or even a new strategy should be tested. If no adequate controller is found, it is possible that the iterative procedure would lead to reconsidering the mechanical concept.

The approach taken in this thesis aims, among other targets, at supporting designers in choosing a mechatronic concept, which is a combination of both a mechanical concept and a control strategy. This is done by allowing a simultaneous assessment of mechatronic concepts by way of their related preliminary designs. Such a preliminary design is a realization of the mechatronic concept by choosing a specific mechanical design and a specific controller. Such a simultaneous consideration may illuminate the possible iterations between the stages of the mechanical and control design. Other works, which deal with such a simultaneous consideration by applying EC, are surveyed in section 2.2.3. Examples of the application of the proposed methodology to engineering conceptual design and in particular to mechatronic design are given in chapter 4. Approaches to conceptual design may be found in the following section.

2.1.2 Conceptual design

According to Pahl and Beitz, (1996): '***conceptual design'** is that part of the design process in which, by the identification of the essential problems through **abstraction,** the basic solution path is laid down through the elaboration of a **solution principle.'*** An abstract concept description is a description that can be represented by a verbal description (e.g., Borgida and Brachman, 2002), a sketch (e.g., Lipson and Shpitalni, 2000), or a parametric model (*e.g.*, AL-Salka *et al.*, 1998). This early stage of the design process is dominated by the generation of ideas, which are then evaluated against a set of requirements. Conceptual design is perhaps the most crucial task in a product development cycle (Bullinger *et al.*, 1998).

It is a common knowledge that a large portion of the product cost results from decisions made during the conceptual design stage (e.g., Lotter, 1986, Ishii, 1995, AL-Salka *et al.*, 1998). Furthermore changing from one concept to another, during the lifetime of a company, commonly requires large resources, and may involve a change of machines, manufacturing lines, and employees. Modern manufacturing companies are facing an ever-increasing demand for customized products to be manufactured and delivered within ever shorter lead times. Choosing an incorrect concept lengthen the production by increasing the number of iterations and re-planning needed to complete the design. It could also lead to the re-initialization of the conceptual design due to the impossibility to compensate for the poor design within the preliminary stage (e.g., Hsu and Liu, 2000).

The significance of correctly choosing a concept has been reflected in an increasing effort to develop methodologies and computational tools to support concept selection. According to Mattson and Messac, (2005) the various concept selection methods found in the literature can be divided into two groups; (a) non-numerical approaches, and (b) numerical approaches. According to Pugh, (1996), mathematics is of little assistance early in the design process and in fact, hinders decision-making. This conviction is the main reason for the development of the non-numerical concept selection approaches. Among such methods are selection methods based directly on decision makers and voting (Ulrich and Eppinger, 2000), selection methods based on methodical/structured preference such as feasibility judgment/intuition (e.g., Otto, 1995; Ullman, 1992) and selection methods based on decision matrices such as concept screening (Ulrich and Eppinger, 2000) and the Pugh concept selection method (Pugh, 1996). The work of Ziv Av and Reich, (2005), where subjective human preferences are incorporated into a conceptual design is yet another example of non-numerical methods. In contrast, numerical concept selection is based directly on numerical computations and includes, among others, the following methods: Axiomatic Design (Suh, 1990), Decision Matrices (e.g., Pahl and Beitz, 1996; Ullman, 1992; Ulrich and Eppinger, 2000), Fuzzy Approaches (Wang, 2001), Knowledge Based Systems (e.g., Dieter, 1991; Chin and Wong, 1996), Utility Function Method, and Quality Function Deployment (Otto, 1995; Magrab, 1997).

It should be noted that there could be different interpretation to the notion of a design concept, and consequently it may affect the selection procedure. Three notions, as related to conceptual design, are of interest within the scope of this thesis. These notions are: Set-Based Concept (SBC), Single-Performance-Vector-Based Concept (SPVBC), and product family. The first is a concept, which is represented by a model that spans the performances of multiple solutions (e.g., a parametric model in AL-Salka *et al*., 1998), and therefore is regarded in this thesis as a ***set-based concept*** (SBC)**.** It is reminded here that a representation by a model is one of the possible abstract concept descriptions (see the beginning of this section; 2.1.2). More background regarding the set based concept is given in section 2.3.1. The second notion is associated with a concept, which is represented by one of its design alternatives' performances. Such a concept is regarded here as ***Single-Performance-Vector-Based Concept*** (SPVBC). This view appears to be more common than that of the set-based. This is not surprising, since when new conceptual ideas are mentally generated by the DMs, it is hardly expected that the DMs would have a multiple representations for each of the generated ideas. In other words new ideas are likely to be interpreted by the second notion of a concept. Presenting new ideas may be associated with the notions of ***Creativity and Innovation***. Bentley and Corne, (2002), defined creative evolutionary systems as systems possessing one or more of the following features: a) support human creativity b) independently solve problems that only creative people can solve. These features are discussed in the following.

The first feature ('a') may be related to Interactive Evolutionary Computation (IEC) methods, which are reviewed in section 2.4.2. The second feature ('b',) is related to systems that can automatically generate creative results. Systems that can automatically generate creative results have been proposed including: relaxing constraints, changing objectives and the boundaries of the design variables (e.g., Parmee, 1999). Gero and Kazakov, (2000), introduced a crossover mechanism that allows the enlargement of the design space, thus promoting the chance of new design alternatives (concepts according to this view). Additional examples for the automatic evolutionary search for creative designs may be found in: Bentley and Wakefield, (1997), Gero and Kazakov, (1996), Bentley (1999). Bentley and Corne, (2002), have investigated an approach where the parameters do not represent the solution itself, but the components from which the solution is constructed. Rules for constructing a solution have been evolved and mapped to a solution through 'embryogency' (Bentley, 1999). Embryogency refers to 'growing instructions' for the genotype. Such a rule-based mapping from genotype to phenotype, which may not be predicted by the designers, could be considered as creative. The resulting phenotype is then evaluated for fitness either by human guidance or automatically by a calculated fitness function. Creativity in evolutionary design can also be analyzed from a broader perspective, namely based on the theory of inventive problem solving (TRIZ) introduced by Altshuller (see Savransky, 2000). Altshuller discovered, through a study of patents databases, that the evolution of engineering systems is not a random process, but it is governed by a class of paradigms. These paradigms can be subsequently used to develop a system considering its technical evolution, i.e. by determining and implementing creativity. Altshuller introduced five levels of innovation in the context of an engineering design problem. All five, are associated with concepts viewed and evaluated as single solutions and not sets.

Innovative solutions are those extracted from the possible combinations that are pre-defined by the design space representation (Boden, 1992). Koza (1992) suggested a block diagram representation of elements of the solution. According to his approach, catalog elements are selected by a GA to be placed as blocks of the diagram. A population of structures of the block diagrams is evolved using special genetic operators. A more elaborative approach is the mixed discrete/continuous evolutionary GA search (e.g., Bäck and Schütz,, 1995) used by Parmee (1996a), where the search is conducted across design hierarchies of discrete design configurations further described by depended continuous variable sets. Some other examples of innovation-based searches can be related to using GAs for whole system design/optimization (Hillermeier *et al.*, 2000, Emmerich *et al.*, 2000).

The confusion between studies that view a concept as a SPVBC and those who view a concept as a set-based one is reflected in several studies and their terminology. According to Li and Azarm, (2000), Crossley *et. al.* (2001), and to Mattson and Messac, (2003), a design alternative, in contrast to a concept, is a specific design, resulting from unique values used in the parametric model of a concept. This may be interpreted such that a concept may be represented by the set of all possible design alternatives (preliminary designs) that are its realization. While the use of the term design alternative, in the above context, certainly distinguishes between design alternative and a concept, a design alternative is regarded in many other cases in the literature as a concept (e.g., Bentley and Wakefield, 1997, Cvetcovic, 2000).

The third notion is that of a ***product family,*** which might be related to the term concept as discussed below. A product family is a group of related products that share common components and/or subsystems – yet satisfy a variety of market niches (Simpson and D'Souza, 2004). The level in the product hierarchy at which commonality is pursued varies; it can be focused on common components (e.g., Fisher *et al*. 1999), on modules (e.g., Chakravarty and Balakrishnan, 2001), on product platforms (e.g., Gonzalez-Zugasti *et al.*, 2000) or on production processes (e.g., Siddique, 1998). The lines between these levels are often blurred (Fixon, 2004). As may be comprehended from these studies, a family of designs is actually a set of concepts that possesses some sort of commonality (see above).

Concepts that are represented by a set of solutions might be viewed as related to family of designs. An interpretation to this relation is depicted in Figure 2.2.

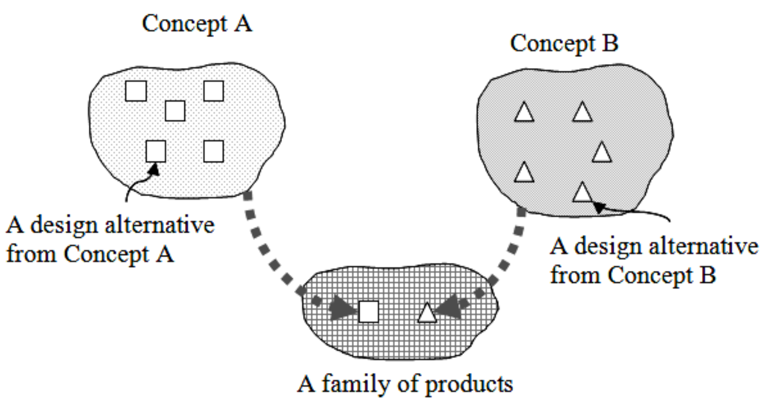


Figure 2.2: Concepts and a family of designs

This figure shows two concepts as two sets of alternative designs (encircled). It is noted that just several design alternatives are depicted for each concept. Assuming the existence of a commonality between the products (see above), then the products are considered as belonging to a family.

Search and selection of concepts, as related to the three concept-related notions are surveyed and discussed in section 2.3. The focus of this thesis is on the view of concepts as SBCs. Nonetheless, the other notions are further surveyed for their related search and selection approaches (see section 2.3) and especially as related to MOPs, which are discussed in the following.

2.1.3 Multi-objective problems

Multi-objective search is an important research topic. It concerns the search for solutions to many real world problems which are Multi-Objective Problems (MOPs). According to Mattson and Messac, (2005), successful engineering design generally requires the resolution of various conflicting design objectives. In case of contradicting objectives there is no universally accepted definition of an 'optimum' as in a single-objective optimization (Van Veldhuizen and Lamont, 2000). In such a case, there is no single global solution and it is often useful to determine a set of solutions that fits a predetermined definition for an optimum and let a DMs choose between them. The predominant concept in defining such a set point is that of Pareto optimality (Pareto 1896, 1906). By definition, Pareto solutions, which belong to the Pareto optimality set, are considered optimal because there are no other designs that are superior in all objectives (e.g., Steuer, 1986, Miettinen, 1999). The search for optimal solutions for a MOP is commonly termed Multi Objective Optimization (MOO).

Often, algorithms provide solutions that may not be Pareto optimal, but may satisfy other criteria, making them significant for practical applications. In such a case solving a MOP problem is not equivalent to solving the MOO of the problem. For example Parmee (1996b), introduced Cluster Oriented Genetic Algorithm (COGA), where the result of the MOP search is a set of solutions that are related to 'interesting regions'. Another approach is the goal attainment approach (Gembicki, 1974) that focuses on finding a solution (s) around a target (goal) in the objective space. A comprehensive survey and comparison between most multi-objective optimization techniques and algorithms can be found in Marler and Arora, (2004).

2.1.4 Design space representation

Representing the design space to support its search could be done in different ways. The hierarchical representation is of a special interest due to its potential in reducing the complexity of the search (Jacome, 1993). In the following, three approaches to hierarchically represent the design space are given. These include the use of abstraction spaces, the whole system hierarchical representation and the system and sub-systems hierarchical decomposition. These are discussed below;

Hierarchy of abstraction spaces: According to Bergmann and Wilke, (1995), "*Abstraction is one of the most challenging and also promising approaches to improve complex problem solving and it is inspired by the way humans seem to solve problems.*" Abstraction is a mechanism for representing things in a simplified manner, hopefully capturing their essence (Giunchiglia, 2003). The main reason for using abstraction for the representation of the design space is the possibility of concentrating, within a very large space, only on what is really crucial to the matter under consideration. Abstraction has many important applications in areas such as: natural language understanding, problem solving and planning, explanation, and reasoning by analogy (see Giunchiglia and Walsh, 1992, for a detailed discussion). At first, minimal abstract description of a solution to a given problem is considered. Then, step by step, more details are added (to the solution description) by taking an increasingly more detailed look at the problem and its solution. According to Armano *et al*. (2003), building an ordered set of abstractions for controlling the search has been proven to be an effective approach for dealing with the complexity of planning tasks. For each of the hierarchies there is a knowledge that allows assessing the plan (Jacome, 1993). The planning is followed by progressing from the highest to the lowest level of abstraction. Commonly, abstraction is performed by dropping sentences from the detailed description of the solution (e.g., Sacerdoti 1974, Tenenberg 1998, Knoblock, 1994). Giunchigilia and Walsh, (1992), and also Jacome (1993), have presented a comprehensive formal framework for abstraction and a comparison of the different abstraction approaches.

Whole system design: In engineering design a hierarchical descriptions of the entire design space has been used for the purpose of reducing the searched design space. This approach is commonly termed whole system design /optimization (e.g., Parmee 1998, Hillermeir, *et al*., 2000, Emmererich *et. al*., 2000). This approach searches the space for the optimal combination of sub-systems and their parameters. According to this approach the design space is constructed into a hierarchical 'OR' tree, where the nodes involve discrete design options, and the leaves are associated with continuous design sets.

System and sub-systems hierarchical decomposition: This hierarchical scheme deals with complex systems by decomposing the original problem into independent sub-problems. The sub-problems are solved separately and then they are coordinated in such a way that an optimal solution for the complete problem is achieved (e.g., Haimes *et al*., 1990). In engineering conceptual design, a DM may decompose a required function into sub-functions, which is the so-called functional decomposition (e.g., Kitamura *et. al.,* 2002). A solution approach to a multi-objective problem that produces Pareto optimal solutions using the hierarchical scheme of systems and sub-systems can be found in several studies (e.g., Caballero *et al*., 2002). Decomposing a large problem into several smaller ones, and its natural inspiration has been discussed by Goldberg (1992), in conjunction with EC.

2.2 EC-based search and its applications to engineering design

This section begins with a brief introduction to genetic algorithms and to Multi-Objective Evolutionary Algorithms (MOEAs) (sub-sections 2.2.1 and 2.2.2 respectively). Thereafter, sub-section 2.2.3 surveys the use of such algorithms to search solutions for engineering design problems. The last two sub-sections (2.2.4, 2.2.5) bring the state of the art as related to resource sharing in single and multi-objective optimization problems.

2.2.1 Genetic algorithms

Genetic Algorithms (GAs) are considered to be a part of Evolutionary Computation (EC) methods. They belong to a class of non-gradient methods that has grown in popularity following the original publication of Rechenberg (1973), and later Holland (1975). Goldberg, (1989) expanded the idea and helped its' popularization. GAs are stochastic search methods that mimic the natural biological evolution. GAs operate on populations of potential solutions applying the principle of 'survival of the fittest' to produce better and better solutions. A GA uses a population of individuals (solutions) instead of a single solution to perform a parallel search in the problem space. At each generation, a new set of approximations is created by a nature-inspired process. The natural processes, which are commonly mimicked by GAs are selection, breeding, mutation, migration, and survival of the fittest. The type of coding used, size of population, and other factors are much dependent on the problem treated (Goldberg, 1989). Usually some rules of thumb are used for their determination. Some attempts have been made to get an insight of GAs by modeling mathematically their dynamic behavior (e.g., Benet and Rogers, 1997). The success of GAs is highly influenced by a balance between the ability of the algorithms to explore the design space and their potential to exploit and find the optimum.

2.2.2 EMO algorithms and their evaluation

Searching a multi-objective design space, for optimal solutions, by EC approaches (such as GA) is commonly referred to as Evolutionary Multi-objective Optimization, (EMO). Multi-Objective Evolutionary Algorithm (MOEA) is an EMO algorithm, which searches for a solution in a multi-criteria space using some inspiration from evolutionary theories. Most MOEAs are using GAs for the evolutionary search.

Research on MOEA has grown considerably in the last few years (see: Coello's web site http://www.lania.mx/~ccoello/EMO/EMObib.html). A number of algorithms, such as the Multiple Objective Genetic Algorithm (MOGA) of Fonseca and Fleming, (1993) and the Non-dominance Sorting Genetic Algorithm (NSGA), introduced by Srinivas and Deb, (1994), are known to advance the use of EMO to solve MOPs. These algorithms use the non-dominance notation (Goldberg, 1989) to direct the search towards a Pareto front. They use niching to allow the spreading of solutions along the front. According to Coello (2005), the later generation of Pareto-based algorithms, such as SPEA2, (Zitzler *et al*., 2001), and NSGA-II, (Deb *et. al*., 2000, 2002), involve three major elements. The first element concerns the creation of a search pressure towards the Pareto set. This is commonly achieved by one of the known Pareto-based fitness assignment (dominance-based) techniques. The second element is set to avoid convergence to a single solution, and preserve diversity. The third element is elitism, which helps to prevent losing non-dominated solutions, which are diversified. Detailed descriptions of multi-objective evolutionary techniques could be found in Deb (2001), and a comparison between the various MOEAs is available in Tan *et al*. (2002).

According to a recent review by Coello (2005), EMO has reached a mature stage, and its development has consistently been followed by applications in engineering, product development, management, and science. In his survey paper, Coello defines some future research directions, which to his view are promising. These are: a) *Incorporation of preferences in MOEAs*, b) *Dynamic test functions*, c) *Highly-constrained search spaces*, d) *Parallelism,* e) *Theoretical foundations.* With respect to 'a', Coello states that "the incorporation of preferences within MOEA has not been fully investigated, although it is most important to do so, as such incorporation may focus the search towards a part of the front." In this thesis, this suggestion is embraced with an alternation. Instead of focusing the search towards solutions based just on optimality, here the interactivity directs the search towards optimal solutions, which belong to preferred concepts. With respect to 'd', Coello declares that: "Currently, there is a noticeable lack of research in this area and is therefore open to new ideas". This thesis also adopts this view and uses parallelism, but with a somewhat different interpretation. The parallelism here is referred to the introduction of simultaneity into the search of concepts and their solutions.

As a part of EMO development a number of test functions have been suggested. Van Veldhuizen and Lamount (1998) reviewed test functions, which were employed in MOEA research, and proposed three test functions as well as several combinatorial optimization problems. Deb (1999) identified and studied the MOEA common difficulties. Based on his findings he developed hard test problems for EMO. His suggested method, to construct such test functions, is based on transferring single objective problems into multi-objective ones. The method requires the choice of three auxiliary functions to control the aspects of difficulties. One function tests the ability to handle difficulties along the front, the second one tests such abilities with respect to lateral difficulties, and the third tests the ability to handle problems that arise due to the shape of the Pareto-optimal region. Using this approach a systematic study has been conducted to compare common MOEA (Zitzler *et al.*, 2000). Jensen’s study (Jensen, 2003) concentrates on computational complexity issues of MOEAs.

The approximation and diversity of the numerically obtained set are the main issues to be considered with respect to EMO algorithm performances. Laumanns *et al.*, (2002) discussed the important issue of convergence versus diversity of the solutions as attained by an EMO algorithm, and introduced the epsilon measure to improve MOEAs with respect to the above two goals. To analyze and compare MOEAs with respect to these goals, performance metrics and measures have been also suggested by others (e.g., Zitzler, 1999, Van Veldhuizen, 1999, and Bosman and Thierend, 2003). The applications of MOEAs with respect to engineering design, is surveyed in the following.

2.2.3 Applying GA/MOEA in engineering

GAs are being applied to many areas of engineering design including mechanical, electrical, aerospace, and civil engineering. A survey on the use of GAs in engineering design can be found in Renner and Ekárt, 2002. The following concentrates on the use of GA/MOEA for mechanical design, control design and mixed mechanical-control design.

Exploring mechanical design space by GAs has been treated extensively. Among such works are: Structural optimization of trusses including arrangement of bars (e.g., Steven *et al.*, 2000), area and material selection (e.g., Coello and Christiansen, 1998), and shape of bars (e.g., Erbatur *et al*., 2000). Others dealt with optimization of topology for beams and for products' shape involving finite element analysis (e.g., Cerrolaza and Annicchiarico, 1999). A modular robot has been designed with GA choosing 24 continuous and discrete parameters (Bi and Zhang, 2001). An extensive SOTA on mechanical design using EC may be found in Kicinger *et al.* (2005).

The application of GA for control can be broadly classified into two main areas: off-line control design and analysis, and on-line adaptive control. Although issues in adaptive control are related to control design, it is beyond the scope of this thesis. An updated survey on GA applications to control was done by Fleming and Purshouse, (2002). According to their review GAs were used to find control parameters in almost all famous control schemas. Typically PID and PI schemes were studied (e.g., Vlachos *et al*., 1999). Other used GA to tune H-infinity controller, (e.g., Chen and Cheng, 1998). The scheme of the controller has been also selected using GAs (e.g., Chipperfield and Fleming, 1996). GAs have been used in attempts to optimize various aspects of intelligent controllers. In Fuzzy control, it has been used to generate the fuzzy rule-base and to tune the associated membership functions parameters (e.g., Gurocak, 1999). In neural controllers, GAs can serve as an aid for weight learning and optimizing the topology of the net (Chung *et al*., 2000). GAs were also used for other control-related topology optimization (e.g., actuator/sensor placement by Brett and Edward, 1999). Han (1999), found the placement of piezoelectric sensors and actuators for vibration control, using GAs. Referring to control at large, Fleming and Purshouse, (2002) claim that: "*The potential of Evolutionary Algorithms is only starting to be realized in this area*".

The need for iterations between the design disciplines, in a mechatronic design process (see section 2.1.1), led to the motivation of simultaneously designing the mechanical structure and the controller. According to Hosam *et al.,* (2001), '*the simultaneous strategy furnishes better systems than the sequential approach.*' Such an approach has been adopted in several engineering related studies, such as for mechatronics (see Youcef-Toumi, 1996). Using EC, Sims initiated an approach (1994a, 1994b), to evolve body shapes and controllers simultaneously. Other researches followed this approach. Among these works are, Lund *et al*. (1997), who evolved robot's control programs with morphology parameters such as sensor number, body structure etc, Dellaert and Beer, (1996), studied the development of morphology and controllers for simple agents in discrete, two dimensional grid worlds. Lipson and Pollack, (2000), used energy minimization for the design of both the control and the structure of real robots from evolved virtual crawlers. Liu *et al*. (1998), and independently Begg (2000), used a GA to optimize the layout and actuator placement of smart structures. Zhu *et al*. (2001) integrated structure-control design of two-link flexible robot arm. They used a multi-objective utility function to determine 6 structural parameters (length and area of links), and 5 control parameters (a sliding mode controller). The results showed faster regulation and reduced arms' weight. Zhu *et al*. (2003) further introduced a simultaneous structural-control optimization of a coupled structural-acoustic enclosure. They have shown through a numerical example that the system performance has improved significantly by the simultaneous optimization.

2.2.4 Resource sharing and sub-populations in single objective problems

In biology the term species refers to the most basic biological classification. It is comprised of individuals that are able to breed with each other but not with others. In nature, a niche can be viewed as a subspace in the environment with finite resources that must be shared among the population of that niche, while competing to survive. In evolutionary algorithms (EAs) the term speciation (or 'niching') commonly refers to an automatic technique to overcome the tendency of the population to cluster around one optimal solution in a multi-modal function optimization. Speciation techniques help maintaining diversity to prevent premature convergence, while dealing with multi-modality. Yet, speciation has been employed for other purposes. For example, Darwen and Xin, (1997), have used speciation for automatic categorical modularization, where speciation is a way to avoid population near-identical strategies. Speciation, in its original sense, could be viewed as an automatic process, or an operator, that gradually divides the population into sub-populations (species). Each of these sub-populations deals with a separate part of the problem (niche of the search space). Commonly 'niche' refers to an optimum of the domain and the fitness represents the resources of that niche. The common process of speciation is also a niching process as it finds the niches, while dividing the population into the niches.

The sharing method, which was originally suggested by Holland (1975), is probably the most popular niching technique. Sharing is analogous to a situation in nature where the resources of a niche have to be shared. In mathematical terms this method penalizes solutions that are similar, by the division of the fitness of the niche among them. The sharing method possesses certain flaws; the most important of them is the need to set the dissimilarity threshold (Sareni and Krahenbuhl, 1998). A possible solution to this problem was proposed by Goldberg and Wang (1997), involving a combination of coevolution and sharing technique, and the implicit sharing technique by Smith *et al*. (1992).

Coevolution (e.g., Hillis 1990) is an important concept, which is often linked with sharing and sub-populations. The recent review on coevolution, done by Cartlidge, (Cartlidge, 2004), highlights the problem of defining what coevolution is, and suggests a new definition.

Coevolution, at least according to a restricted definition, involves species that are either competing or cooperating. Competitive coevolution, which is of a particular interest here, has been employed with single as well as with multi-populations. In contrast to niching, where species are automatically formatted, in coevolution of competing species, the species are commonly predefined (although their populations' relative size may be subject to automatic changes). This situation resembles that of a C-EMO, in which the association of sets of particular solutions with concepts is predefined. Moreover, finding the concept-based Pareto set, as suggested in this thesis, amounts to a competition among the concepts, or more precisely among the sub-populations that represent concepts. Competitive coevolution is usually employed when dealing with games. Different strategies are competing, and their success depends on the opponent strategies. A common approach to modeling games is by the use of a host-parasite analogy, where two sub-populations exist (e.g., Hillis, 1990). Several investigators have extended fitness sharing to coevolution including: competitive fitness sharing, (Rosin and Belew, 1995), implicit sharing (Darwen and Xin, 1997, Smith *et al*., 1992), and resource sharing (Juill'e and Pollack, 1998). A particularly interesting application of coevolution concerns the idea of evolving modules that are evaluated separately based on their ability to contribute to a larger whole. This results in dependencies between modules and complete individuals. Such setups are thus forms of coevolution, and many examples of this are available (e.g., Moriarty and Miikkulainen, 1999; DeJong, 2003).

Yet, another approach to sharing within EMO is taken within genetic programming (Koza, 1992). Genetic Programming (GP) searches a solution within a design space, which is represented by a tree or a graph. Within this representation niching is used to maintain diversity of tree structures. Several works introduced variants of diversity objective (e.g., DeJong *et al*., 2001). For example, Hu *et al*. (2002) introduced Structure Fitness Sharing (SFS) where the sharing maintained different tree structures by penalizing niches of tree structures according to a distance measure between the nodes of the trees. SFS aims to balance the search for a tree structure and its parameters by applying fitness sharing to each unique structure in a population. McKey (2000) used implicit fitness sharing in GP. Instead of calculating the distance between the structures of GP trees, the fitness is shared based on the number of individuals with similar behaviors, capabilities or functions.

Multi-populations exist also in parallel GA, which deals with the use of parallel hardware for evolutionary computations (Cantú-Paz, 1995). The division of the population, in parallel GA, is with respect to the hardware distribution. A popular model used in parallel implementation is the regional population model, which is often referred to as migration model, coarse-grained model, or island model (Cantú-Paz, 1995). In fact this population model is useful also in a serial implementation (Pohlheim, 2001). The island model involves a division of the whole population into subpopulations. An exchange of information (exchange of individuals) between the subpopulations, which is termed migration, takes place from time to time. Pohlheim (2001) developed a logical extension of the regional population model with the application of different strategies using the principle of competing subpopulations. When using competing subpopulations the size of a subpopulation is varying based on the success of its strategy. As noted by Pohlheim, this is in fact a division of resources. Schlierkamp-Voosen and Mühlenbein, (1996), suggested an adaptation technique, which varies the entire population size in addition to a change of the sizes of competing sub-populations. They have noted the relation of this extension to the Lotka-Volterra equation, which describes the development of competing species.

In summary, resource sharing among individuals and among sub-populations is a well know concept in EC, which has analogies to nature. It should be noted that the idea of resource sharing is also an essential part of EMO, as highlighted in the following.

2.2.5 Resource sharing and sub-populations in EMO

In EMO the algorithms should find all the trade-offs among the conflicting objectives. Therefore, ensuring diversity along the front, which depicts these trade-offs, is a must for any successful MOEA. The recent review by Zitzler *et al*. (2004), classifies existing methods for diversity preservation according to the three categories of statistical density estimation including: kernel methods, nearest neighbor, and histogram techniques. Fitness sharing, which is a popular technique for diversity preservation in MOEA, falls into the first category (e.g., Van Veldhuizen and Lamont, 1998). MOEAs commonly use sharing as a mean to equally distribute the vectors, which approximate the Pareto front. Nonetheless, the importance of preserving diversity within the design space has also been treated (e.g., Van Veldhuizen and Lamont, 2000, Chan and Ray, 2005). Different variants of preserving diversity have been employed, as reviewed in Van Veldhuizen and Lamont, (2000), with both genotypic and phenotypic distance measures. The later allows estimating the key sharing parameter, assuming known phenotypic extremes.

Models of sub-populations exist in parallel EA. The development of parallel MOEA is relatively new; only sporadic studies are available, and no accepted test suite exists (Deb, 1999). In general, research on parallel MOEA focuses on distributing the population as related to hardware considerations, such as communication.

Another resource sharing, which is treated within the framework of an EC search, is associated with Pareto-Coevolution (e.g., Ficici and Pollack, 2000, Watson and Pollack, 2000). In Pareto-Coevolution approach the search for optimal solution is directed by testing the individuals whose performances are optimized (called learners) by a testing set (called testers). To achieve the best progress towards optimality, a solution strategy should be chosen. An example of a solution strategy would be the learner that solves the largest number of tests. However, there may well be learners that solve tests not solved by such a learner, and may therefore be valuable. A solution concept that employs all information about learners provided by tests is given by the Pareto-Coevolution. Tests are treated as objectives in the sense of evolutionary multi-objective optimization. The resulting solution concept is the Pareto-front, containing all learners that are non-dominated as determined by their test outcomes. Sharing resources between learners and tests can be found in Werfel *et al*. (2000). In their work, the motivation behind resource sharing is to promote diversity, by rewarding strategies that can solve test cases that few other strategies are also able to solve. In this way, strategies receive less payoff for pursuing approaches that put them into 'niches' already heavily occupied. In other words, resource sharing is intended to preserve diversity, to prevent mediocre solutions from taking over the population, and to make more likely the emergence of exceptional new strategies through recombination of dissimilar, previously discovered, strategies. It is noted that the final goal of the approach is not to find different strategies but rather to make the search for optimal solution more efficient.

2.3 Search and selection of concepts in MOPs

In this section research efforts concerning the search of optimal concepts in the context of MOP are surveyed. The three concept-related notions, which were discussed in section 2.1.2, are considered here. These include: SBCs, SPVBCs and family of designs. It is noted that this thesis treats concepts according to the first view (set-based) and the review on the other two approaches is given for the sake of completeness and positioning the current thesis with respect to other approaches. In MOPs, the numerical-based assessments of designs, as related to the three notions of concept are associated with the solutions' performances within the objective space. Figure 2.3, depicts the thesis view of what is the relation between solutions and their performances representation as related to these three notions. A bi-objective problem is used in the figure for the sake of simplicity.

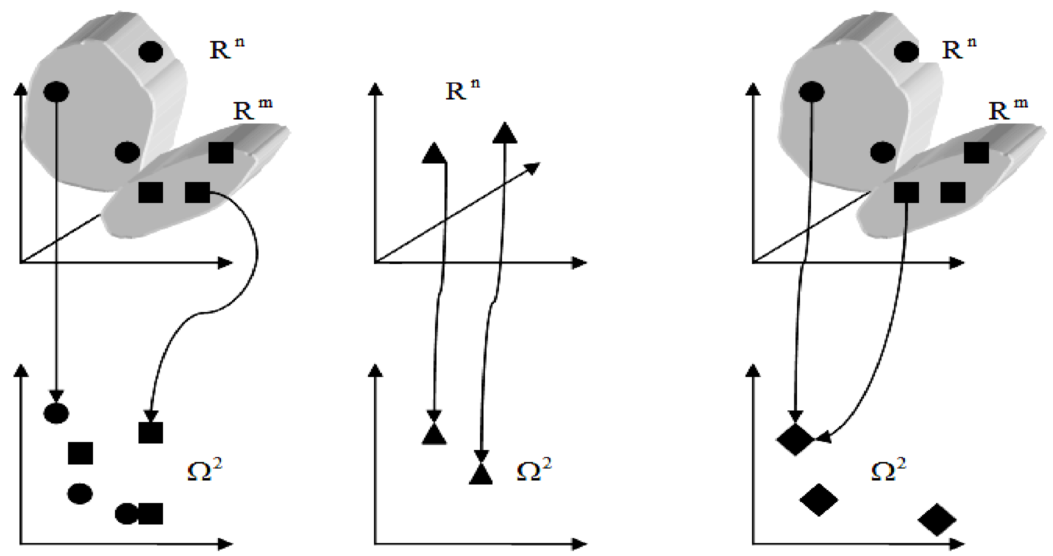


Figure 2.3a: SBC b: SPVBC c: Family of designs

In the SBC (figure 2.3a) each concept has its set of solutions, generally within its own design space. Each of the concepts is associated with a set of performances in the objective space (represented by a single point within that space). It is noted that the same performances might be associated with solutions from different concepts. The performances of solutions in the objective space are distinguished by different symbols according to their relation to a concept. The axes of the objective space are (as depicted in the figure) the design objectives, which are common to all concepts and their particular solutions. In the SPVBC (figure 2.3b), the design space is common to all solutions and each is associated with its related performances in the objective space. In this case the performances of the designs are also represented in a space which is directly associated with the design objectives. In the case of a family of designs (figure 2.3c) different designs of a product (concepts) have performances in the objective space. But in contrast to the former two approaches, the family of design approach focuses on points of the objective space where commonality exists. These points might not be a part of any of the concepts' fronts. In the following sections the search and selection of the three concept-related notions are surveyed. It is noted again that the focus of this thesis is the SBC. Nonetheless, the other two notions are also included in the survey here due to the above highlighted conflict and to their close relation with the SBC.

2.3.1 Set-based concepts

Multiobjective optimization has been used by some researchers to perform concept selection by posing the design as an optimization problem and choosing the designs that satisfy the optimization conditions. Pareto-based methods, which use SBCs, may be found in the literature e.g., Li *et al*. (1998), Das (1999), Balling (2000), Tappeta *et al*. (2000), Kasprzak and Lewis (2000), Crossley *et al*. (2001). In these works the focus is on the generation of the Pareto fronts and then on the support of a selection of a particular solution out of the set. Such works utilize different search techniques such as the Weighted Sum Normal Boundary Intersection method (Das and Dennis, 1998), the Physical Programming method (Messac and Mattson, 2002), and the Normal Constraint method (Messac *et al*., 2003). The main advantage of last three methods is their ability to generate even distributions of solutions of the concepts' fronts. According to Mattson and Messac, (2003) "*Although Pareto optimality has played a significant role in the advancement of methods for multi-objective optimization, its use as a tool for conceptual design has not yet been fully explored.*" In a more recent paper they stated (Mattson and Messac, 2005) that "*only a few concept selection methods use computational optimization techniques to evaluate concepts*." These that do use such methods are surveyed in the following.

Mattson and Messac, (2003) introduced the s-Pareto notion, as the solution of a C-MOP, involving a combined front of such sets of concepts. They have accompanied their solution method with a 'goodness' measure to compare concepts along the front (Mattson and Messac, 2005). The goodness of each concept is quantified by determining the intersection of each concept's Pareto front with the s-Pareto front. The assessment of a concept has been practiced within boundaries in the objective space which they have termed 'region of interest'. Exploring various regions of interest, the designer may collect information about the design space (i.e., which concepts occupy which parts of the design space); this information can be used to identify the most attractive design. In contrast to Mattson and Messac, who have used a normal constraint method to generate the front, Andersson (2001) used a MOEA approach to display and compare concept related fronts in a sequential approach. Each set of solutions that represents a concept is evolved separately and then introduced on a mutual graph to compare the concepts' fronts and to select a concept and eventually a particular solution.

The general motivation to develop an EMO approach to finding the s-Pareto is rooted in the general benefits of using EAs to solve MOPs (see introduction). A more specific motivation for using an EMO approach, which is brought within this thesis, is associated with the incorporation of preferences towards concepts (see methodology section 3.2). It seems that the incorporation of such interactivity is suitable for EMO. Examining the SOTA of computational tools, which incorporate human preferences towards designs (e.g., IEC) within a design space, it appears that the use of EC is by far more common than other approaches. This is especially true if EC is compared with the mathematical programming related methods. The sequential

EMO approach, such as used by Andersson (2001), has some deficiencies as discussed in section 4.1.2, and it is used here for the sake of comparison with the simultaneous approach, which is introduced in this thesis as an alternative search approach.

2.3.2 Single-Performance-Vector-Based Concepts

An EMO search approach to concepts that are SPVBCs has been suggested by Cvetkovic (Cvetkovic, 2000). He assumed that different performances are associated with different concepts. In his work, the number of solutions, which are presented to the DM, is limited by the use of ranking. The result of this procedure is a diluted well-spread Pareto front. It is noted that several other approaches exist for presenting just several solutions on the Pareto front (e.g., the clustering approach, Zitzler, 1999). A similar assumption that different locations within the objective space are of interest to the DM has been taken by Parmee (Parmee, 1996b). Parmee suggested that rather than identifying Pareto-optimal solutions, information that supports a better understanding of the multi-criterion aspects of the problem should be introduced to the DM. According to his approach, high performance regions are identified by the use of Cluster Oriented Genetic Algorithm (COGA). The work of Deb (2003) may also be related to such a clustering approach. Deb used a resulting Pareto front to identify, distinct and representing solutions along an evolved front.

2.3.3 Family of designs

Searching for family of designs by using EC may be found in several citations. The approaches to evolve the families are either sequential or simultaneous. In both approaches a Pareto front is developed. For example, Rai and Allada, (2003), introduced a sequential approach to tackle the modular product family design problem. The first step performs a multi-objective optimization using a multi-agent framework to determine the Pareto-design solutions for a given module set (SBC). The second step performs post-optimization analysis to determine the optimal platform level for a related set of product families and their variants. An example for the simultaneous approach is a MOEA approach, which has been taken by Simpson and D'Souza, (2004). They used NSGA-II to facilitate a structured GA (Dasgupta and McGregor, 1994), with a commonalty as a part of a one-stage optimization algorithm. The objectives of the problem in Simpson and D'Souza (2004) are the variation in design variables and a deviation function from a given goal. It is noted that in their approach the purpose of the optimization is different from that of this thesis, namely their focus is on commonality. As a result of the different motivation, the algorithm is different from those introduced in this thesis. A through SOTA of EC and non-EC approaches to search for family of designs may be found in Simpson (2003).

2.4 Handling preferences using MOEA

The main reason of preference articulation in the common MOEA literature is to limit the search space and direct the search towards preferred sub-spaces of the objective space. It is possible to classify MOEA techniques according to the approaches by which the DM is articulating the preferences (Van Veldhuizen and Lamont, 2000). These can be classified as: a-priori, progressive and a-posteriori.

2.4.1 A-priori methods

A-priori approaches are associated with the articulation of the DM preferences a-priori to the initialization of the evolution. One approach that belongs to this category is to transform a MOP into a single objective problem by specifying a utility function over all different objectives. A-priori preference articulation combines the objectives into a scalar cost function which is referred to as the fitness function. The approach is to predefine weights or their change interval. The weights reflect the importance of each objective in the fitness function, thus directing the search to a restricted region. Among such algorithms are the Weighted-Sum (Jakob *et al.*, 1992), goal programming (Coello, 1996) and guidance (Branke *et al.*, 2001). Another method is the Lexicographic approach (Fourman, 1985) where each goal is attended at a time. A comprehensive survey and explanations on all mentioned approaches and others can be found in Coello (1999, 2000, and 2005).

2.4.2 Progressive methods

The core of the progressive methods is interactivity between a DM (human) and a computer during the evolution. When applying such methods in conjunction with MOEA they are commonly referred to as Interactive Evolutionary Computation (IEC). IEC approaches might be categorized by the level of explicitness of the human intervention in the evolutionary process (Parmee and Abraham, 2004). Located on the implicit end of the spectrum are the automatic, model-based algorithms (no progressive articulation of preferences), and on the other (explicit) end are algorithms, where the evolution is solely driven by human evaluations. The explicit category corresponds to what has been referred to as the narrow sense of IEC (Takagi, 2001). In that end of the spectrum the DM is directly changing the fitness of the individuals for the sake of one evolutionary step at a time. IGA is suitable for systems that are ill-defined, or hard to program, and the fitness function is in doubt, or un-presentable (e.g., artistic design). The more implicit part of the spectrum holds methods where humans do not directly set the fitness, but rather influence the fitness indirectly.

An extensive survey of the use of IEC can be found both in Takagi's review of IEC (Takagi, 2001) and in Parmee (2001c). IGAs have been used for a wide range of applications such as fashion design (e.g., Sung-Bau 2002, Eckert, *et al.*, 1999), engineering design (e.g., Banzhaf 1997, Takagi, 1998), music (e.g., Biles, 1999), and for many more. Inoue *et al*. (1999) applied IEC to nurse scheduling support system. The head nurse interactively enters new constraints. This causes a change of the fitness function, and thus influencing the evolution. Parmee (2001a) and Cvetkovic (2000) allowed the DM to articulate fuzzy preferences towards the design objectives that change the fitness function weights. Furthermore in their works, an interactive tool was suggested allowing the DM to dynamically add constraints or goals. This was done by the introduction of scenarios. Moreover, Parmee used IEC for assisting the DMs in exploring the multi-objective design space (e.g., Parmee, 1996b). As a part of this approach the use of a Cluster Oriented GA (COGA) has been introduced, which allows the identification of high performance regions that are influenced by the DMs' objectives' preferences. Fonseca and Fleming, (1993), incorporated the goal attainment method, to their Multi-Objective Genetic Algorithm (MOGA). At each run the DM supplies goals, thus reducing the size of the search and allowing the learning of the trade-offs between the objectives. An interesting IEC work, which utilizes hierarchies of abstract descriptions (see section 2.1.4) for engineering design solutions is found in the work of Chan *et al*. (2000). In their work a shape of a wine glass is manipulated by articulation of preferences at different levels of abstraction. The fitness of the design is directly affected by the designers.

2.4.3 A-Posteriori and hybrid methods

In a-posteriori preference articulation the optimal solutions are presented to the DMs by the computer for a final manual selection of the preferred solution. For a related and extensive discussion on selection between alternative solutions, based on their performances, see section 2.3.

In the concept-based MOP, the a-posteriori selection of a solution is associated with two selections. Before selecting a particular solution, a concept has to be selected and only then a particular solution belonging to the chosen concept may be selected. In this thesis a new approach to support the selection of a concept is presented. It advances the SOTA of selecting concepts based on a comparison between concepts solutions' performances as detailed in section 3.4). The SOTA of a -posteriori selection of concepts may be found in section 2.3.

Hybrids between a-priori and a-posteriori approaches may be found in the literature (e.g., Branke *et al.*, 2001, Deb, 2005). In the hybrid approach, which is referred in Deb (2005) as I-EMO, the DM specifies the preferences as good as possible and provides imprecise goals. These are then used in a MOEA to bias, or guide the search towards the solutions that have been classified as 'interesting' by the DM. According to Deb (2005), *"this procedure, in addition to providing the ideal and nadir points of the problem, will also paint a good picture in the mind of the DM about the shape of the Pareto-optimal frontier which will help the DM later to concentrate on a particular region on the front*."

2.5 Robustness considerations

The survey on robustness is conducted with respect to the three concept-related notions (see section 2.1). These include: SBCs, SPVBCs and family of designs. It is noted that in the SOTA, the combination between a search for optimal and for robust SPVBC, which are declared as concepts, is scarcely found. Therefore, the common robustness considerations as related to particular solutions are alternatively surveyed (in the context of SPVBC). This is done in section 2.5.1. Section 2.5.2 focuses on the robustness of concepts that are represented by sets of designs (as the approach of this thesis) and finally section 2.5.3 surveys robustness as related to family of designs.

2.5.1 Robust design for particular solutions

Robust performance design tries to ensure that performance requirements are met and constraints are not violated due to system's uncertainties and variations (e.g., Mattson and Messac, 2005). Fundamentally, robust design is concerned with minimizing the effect of such variations without eliminating the source of the uncertainty or variation (Phadke, 1989).

Taguchi, (e.g., Taguchi *et. al*., 1999) has contributed tremendously to the development of this field of interest by introducing several approaches (e.g., Loss Function, Orthogonal Arrays and Linear Graphs). It is well known that optimality and robustness might be contradicting demands (e.g., Andersson, 2001). The importance of arriving at a robust design is well known and the interest at that field is reflected by the vast number of works, which are partially surveyed in the following.

Treating robustness by non-evolutionary approaches may be found in many related studies. A typical approach for handling uncertainties is to use safety factors. Although popular, the safety-factor approach generally results in overly conservative designs (Koch, 2002). Other robust design approaches focus on the probability of design failure, where failure is to occur when design constraints are violated. To minimize the likelihood of design failure, such approaches seek to reduce the area under the Probability Density Function (PDF) that lies outside the constraint boundaries. The area under the PDF that lies outside the constraint boundaries is reduced by optimizing the mean performance and minimizing its variation. Feasibility is maintained with probabilistic constraints.

Performing uncertainty analyses generally involves obtaining the variations in the response given variations in the input variables. Various approaches are typically used to obtain this output variation. Among others, these methods include Monte Carlo simulations, and sensitivity-based estimations (Koch, 2002). The Monte Carlo simulation is known to appropriately represent the behavior of the response variable when an adequately large number of samples are used. For computationally expensive problems, the number of samples needed to obtain the output variation may render the Monte Carlo approach prohibitive. In such a case, other reduced-sampling methods exist that can provide an adequate representation of the response function variation. These methods include Latin Hyper-cube sampling, Taguchi's orthogonal arrays, and importance sampling (see Du and Chen, 2000).

Robust performance approaches, involve situations in which either the design variables and/or the environmental parameters are subject to perturbations or changes. There are many possible ways to treat robustness by using EC, and a few possible heuristics have been suggested in Branke (2001b). The existing research work is commonly concerned with robustness as related to single objective problems, and the optimization of the expected fitness, given a probability distribution of the disturbance. Since it is usually not possible to calculate the expected fitness analytically, it has to be estimated. This, in turn, raises the question on how to estimate an expected fitness efficiently, and how to optimize based on such estimates. Evolutionary algorithms have been shown to be quite robust with respect to uncertainty in the fitness values, (e.g., Arnold and Beyer, 2003). Common evolutionary-based approaches to search for robust solutions include the following: a. explicit averaging over multiple samples (e.g., Thompson 1998, Wiesmann *et al*., 1998). b. evaluating important individuals more often (e.g., Branke *et al*., 1998b, Branke, 2001c) c. using other individuals in the neighborhood (e.g., Branke, 1998) d. meta-models (Andersson, 2001). An EMO approach to deal with robustness to a single objective problem has been taken by several researches (e.g., Das, (2000), Jin and Sendhoff, 2003). According to their approach performance and robustness were treated as separate objectives. The Pareto front contained a set of solutions with different relations between performance and optimality.

Dealing with robustness within MOPs is scarcely found. Just recently Deb and Gupta (2004, 2005) introduced a formulation for the different aspects of robustness and suggested an approach to evolve a robust front based on the mean of an effective fitness function. Luo *et al.* (2005) used an EMO approach to evolve robust fronts which are a result of taking into consideration possible market changes. In their work it has been assumed that, given a design space, the designer typically can specify a target point in terms of aimed design objective values. This target becomes the basis for determining the worst-case objective values and the best-case objective values under the variations in uncontrollable design parameters.

EC-based approaches are also used to treat other kinds of uncertainties including noisy fitness function, and dynamic fitness function. In the former approach noise in fitness evaluations may come from many different sources such as sensory measurement errors or numerical instabilities in the simulation. Examples of GAs that treat such uncertainties are: Sano and Kita, (2002), Branke (1998a) and Stagge (1998). In the later method a dynamic fitness function is considered. In such a case, it should be possible to continuously track the corresponding dynamic optimum rather than to repeatedly re-start the optimization process. Examples of GAs that consider dynamic fitness function are: Gobb and Grefenstette, (1993), and Naoki and Kita, (2000).

2.5.2 Set based Concepts' robustness

Studies on robustness concerning SBCs are scarce. Mattson and Messac, (2003) consider both uncertainties caused by stochastic design parameters and those associated with the design model. They have used a non-evolutionary method to produce separate Pareto fronts of concepts, and then combine them to a mutual front by the s-Pareto approach (see section 2.3.1). These fronts are then shifted according to non-deterministic approach to take into account uncertainties in design parameters and model structure (Mattson and Messac, 2002). Andersson (2002) has also addressed robustness of concepts. In his work, conceptual fronts are separately evolved by a MOEA, and then plotted on the same graph to obtain a final front. The sensitivity of different points on those fronts to different parameters combinations is evaluated. These evaluations serve the DMs for better insight and increase their capability to select good concepts. A similar approach has been taken by Olvander (2005). He presented a method where a multi-objective genetic algorithm is combined with response surface methods in order to assess the robustness of the identified optimal solutions. In his work, two different concepts of hydraulic actuation systems have been considered. The outcome from the optimization is a set of Pareto-optimal solutions that elucidate the trade-off between energy consumption and control error for each system. Based on these Pareto fronts, promising regions could be identified for each concept. In these regions, sensitivity analysis has been preformed to determine how the parameters affect the system at different optimal solutions.

Although optimality is the main focus of SBCs assessment, it is most important that DMs will choose a concept that might cope with market variability and uncertainty by the delivery of a large numbers of product variants (Tiihonen *et al.*, 1998). With respect to this view, it could be claimed that commonly the performance of a chosen design can be optimal in the Pareto sense, but considering the phenomenon of contradicting objectives, it might not be robust to objective variability and uncertainty. In contrast, concepts, which span a sub-space of preliminary designs, may possess such robustness.

2.5.3 Family of designs related robustness

When families of designs are concerned (see section 2.1.2), apart from the above issues, some new aspects of robustness may be considered. Product Family Based Robust Design (PFBRD) may accommodate for changes in the market, changes in customer needs/requirements, change in government/legislation (http://mail. chiangmai.ac.th/ ~apichat /Intro.htm in 'research objectives' 2004).

Gonzalez-Zugasti *et al*. (2000, 2001) assess the net value of a product family using real options to model the risks associated with factors, such as uncertainty, in funding levels and technology readiness. Blackenfelt (2000) uses robust design techniques to maximize profit and to balance the commonality and variety within a product family; he demonstrates the approach by using it to design a family of lift tables. Jiang and Allada, (2001) use a similar approach to design a family of vacuum cleaners that is robust to current and future market trends.

The SOTA as reviewed in sections 2.1-2.5, possesses some shortages as summarized in the following. This is followed by the statement of the objectives of this thesis.

2.6 Shortcomings of the existing SOTA and thesis objectives

Examining the SOTA as surveyed above, the following observations are made:

a. The surveyed studies, which deal with SBCs, do not incorporate interactivity with respect to the concepts.

b. The surveyed studies, which utilize EMO, do not combine model-based performances with human preferences of concepts and sub-concepts.
c. The surveyed approaches do not utilize preferences of designers towards abstract descriptions within a hierarchical representation of SBCs.
d. None of the existing EMO approaches to search for the s-Pareto are simultaneous.
e. The few studies, which support SBC selection, suggest utility based measures, which may hide valuable information concerning the tradeoffs between the different aspects of the conceptual design.
f. The delayed decision problem has not been introduced in the context of a MOP.
g. The solution to the delayed decision problem has not been supported by a computational tool.
h. The surveyed studies, which deal with SBCs, do not convey mechatronic examples to demonstrate the applicability of the approaches to a combined control-structure design.

The above list of limitations of the existing SOTA points at some needs in advancing computational methods to support conceptual design, and in particular with respect to multi-objective problems and their solution by evolutionary methods. The objectives of this thesis address the above shortages and are associated with the following four main issues: 1. simultaneous evolution of concepts' solutions towards and along a Pareto front (with respect to points 'd' and 'h'); 2. interactive evolution of concepts' solutions towards optimal solutions of preferred concepts (with respect to points 'a-d' and 'h'); 3. assessment and comparison of concepts in the multi-objective space (with respect to point 'e' and 'h'); 4. supporting decision making with uncertainties due to delayed decisions (with respect to points 'f-g' and 'h'). The following provides some details about the operative objectives of this thesis.

A. **Develop a simultaneous approach to evolve an s-Pareto**. This should be done by developing a MOEA to simultaneously search for concepts by way of evolving their particular solutions. Such an approach requires special considerations in terms of the utilization of the resources, to provide a good approximation to the associated Pareto front. The references given in sections 2.2.4, 2.2.5 do not compare easily with this requirement, which involves an inherent division of the population to sub-populations. During evolution, mating between sub-population should be avoided for reasons related to the nature of different concepts, leading to a need to consider the resource sharing between and within the sub-populations. The simultaneity objective imposes a need to treat the evolution in a new manner. The common use of phenotypic and genotypic niching, as surveyed above, is not applicable. The phenotypic niching may not be applied due to the possible existence of solutions from different concepts in the same location in the objective space. The latter can not be applied because the solutions of the different concepts may be from different design spaces, and/or have different number/boundaries of design variables (see section). Therefore, a distance between such solutions may not be rationally formulated.

It is noted that the main motivation for the simultaneous approach is the interactivity, as explained in the methodology chapter. Nonetheless, the apparent inefficient use of resources, which is involved with a sequential approach, serves as a further motivation for its development.

B. **Develop an interactive concept-based algorithm to evolve solutions, which are influenced by both model-based performances and by DMs preferences of the solutions' concepts.** This means that the search should be conducted towards optimal concepts' solutions, taking into consideration the preferences of the DMs towards the concepts themselves (either directly or by way of sub-concepts). With this respect, it is noted that a concept might be represented by a hierarchy of sub-concepts. The interactive concept-based evolution should therefore utilize the available computational resources to restrict the search towards the optimal solutions of preferred concepts and sub-concepts, while taking into consideration the hierarchical representation of the design space.

C. **Develop a method to compare concepts by highlighting the tradeoffs between different possible objectives of the conceptual design.** The method should incorporate measures that will allow a comparison between the concepts, thus supporting taking a decision on a concept. To highlight the tradeoffs, a utility function of the conceptual design's objectives should be avoided, as it merges these objectives into a scalar value, hiding the tradeoffs.

D. **State the delayed decision problem within the context of MOPs**.

Many engineering problems are MOPs and the need to postpone decisions in conceptual design is recognized. Therefore, the delayed decision within the context of MOPs should be introduced.

E. **Suggest a computational-based method to support decision making within the conceptual design stage in the presence of delayed decisions uncertainties.** Such a method may support decisions in the conceptual design stage, by supporting the decision on the basis of computations and not just based on human subjective evaluations.

F. **Demonstrate the applicability of the proposed approaches to mechatronic conceptual design.**

The scope of this thesis is wide and generic. It introduces new ideas, approaches, notions and algorithms as introduced in the following chapter. The contributions and their significances are highlighted and discussed in chapter 5.

CHAPTER 3

METHODOLOGY

In ‘real-life’ situations humans rely on their experiences and preferences in choosing a conceptual solution, and eventually they translate the chosen concept into a chosen preliminary/detailed design. This is usually done with or without the ability to explicitly evaluate the merits of the chosen solution. The aim of this work is to suggest a method to support DMs, by performing a concept-based multi-objective search, and by suggesting an approach for concept selection. The proposed search concerns conceptual solutions that can be represented by sub-sets of the set of particular solutions. In this thesis, which deals with applications to engineering design, these particular solutions are preliminary designs. Nevertheless, it is claimed here that the introduced approach is not limited to engineering design and therefore the notion of ‘particular solution’ is also used. It is further assumed that the performances of each preliminary design are computable via models (e.g., tables, parametric models). Each conceptual solution, and its associated particular solutions, may be characterized by different models, different design variables, and/or range of variables.

The following sections describe the suggested methodology. Section 3.1, describes the way concepts are represented in this thesis. Some tailored definitions linking between the design space and its representation are given. Based on this representation different types of concepts, which might be extracted out of the representation, are introduced. Section 3.2, describes the notion of a C-MOP. It also includes the formulation and introduction of several types of concept-related fronts as well as a suggestion for concept-based indicators that allow the assessment of concept-based algorithms. Furthermore two simultaneous EMO algorithms are introduced. These algorithms are utilized to evolve the s-Pareto front. The algorithms involve some new features that are introduced and explained within this section. Section 3.3, describes the interactive search of the design space for conceptual solutions, which are influenced by both solutions performances as well as by human preferences of the solutions' concepts. The interactive search problem is defined as well as the resulting fronts. An interactive MOEA and measures to assess its success in consistently evolving the front are introduced. In section 3.4 the selection of concepts for solving a MOP is treated. A new approach to compare between concepts, for supporting concept selection, is suggested. In section 3.5, the delayed decision problem is explained and restated as a MOP. Thereafter, the approach, which is introduced in section 3.4, is adapted to support conceptual decision making in the presence of such an uncertainty. Testing the methodology, which is presented in this chapter, is left for chapter 4.

3.1 Design space representation

In this thesis the design space is represented by an 'AND/OR' tree (a tree with nodes that are either of the type ‘AND’ or of the type ‘OR’). Such a representation is not new (see literature survey, chapter 2). Nevertheless some new relations between the tree and engineering conceptual design are introduced. These relations allow the introduction of new interactivities, between the DMs and the computers, during the conceptual design stage, as explained in this thesis. The description of the 'AND/OR' tree representation, in section 3.1.2, is followed by a discussion on the types of concepts that may be extracted from such a tree. The tree representation and its associated models allow the interrogation of the design space by the proposed methodology. It should be noted that in this thesis the focus is not on the construction of the design space representation, but rather on the search process itself, within the space representation. Therefore, issues concerning the space representation, such as: constraints, uniqueness, and extraction of sub-trees, are not discussed here. It is further noted that the suggested technique does not deal with the innovation stage of design, thus it is restricted to a search within a given representation of the design space (see section 2.1.2). This means that the generation of the concepts is not treated within the thesis.

3.1.1 The 'AND/OR' tree representation

This thesis deals with concepts that have computable models. To clarify the use of terms the following definition is given, with a subsequent example.

Definition # 1
A Complete Concept (CC) is a concept that has a computable parametric model, which allows the evaluation of the performances of its associated particular solutions.

The term CC designates a concept, which has a model, from concepts that do not. For example, an aircraft is a concept (idea) that could serve to solve an engineering problem of transporting goods and people from one continent to the other. The idea of an aircraft is not a CC, as it does not have a computable model to calculate its performance. In order to become a CC more details on the particular plane type should be provided (e.g., the type and shape of the wings). At a certain level of details, which depends on the particular design, a model can be obtained. Once a concept has all the necessary abstract descriptions of its details to obtain a parametric model it becomes a CC. Each CC is represented by an 'AND' tree, hereby termed *CC Tree*. The nodes of the CC tree represent *Sub Concepts* (SCs), which are defined below.

Definition # 2

A Sub Concept (SC) is an abstract description of one of the following types: 1. an abstract description of a part of a concept/CC, 2. a feature of a concept/CC, or 3. a concept that is insufficient to be a CC.

The following example is given to demonstrate the meaning of the three types of SCs. A wing is an example of the 1st type, as it is a description of a part of the concept 'plane.' Stiffness of the plane structure is an example of the 2nd type, and a 'plane' is an example of the 3rd type, due to its insufficiency for modeling (see example of a CC above). The SCs, of each branch of a CC tree, are descending as a result of the level of the abstraction of the details. Thus the SCs are ordered in a hierarchy of abstract descriptions. For example, a SC of 'PI controller' is a more detailed, yet abstract description of a 'controller' SC. Therefore the former is the son of the latter, or in other words it is located in a lower hierarchy with respect to the latter. The root node is the 1st hierarchy followed by increasing index of hierarchy till the leaves, which are in the lowest hierarchy (highest index).

Definition # 3

A design space tree is an 'AND/OR' tree which holds all the CC trees of the design problem.

The design space tree represents the conceptual design space. The CC trees, which are 'AND' trees, can be extracted from the design space 'AND/OR' tree. Selecting a particular CC (extracting it from the 'AND/OR' tree) involves decisions associated with the 'OR' nodes. The 'OR' nodes are designated by 'On'. The 'AND' nodes are nodes representing SCs that are made of a combination of other SCs (represented by the child nodes). Such nodes are designated by 'An'. Figure 3.1 depicts a design space tree with some nodes designated by their type (an 'On' or an 'An').

Figure 3.2, in conjunction with table 3.1, depicts an example of a part of such a design space tree. This space corresponds to the engineering problem of designing an artifact for the purpose of moving a given load from location 1 to location 2.

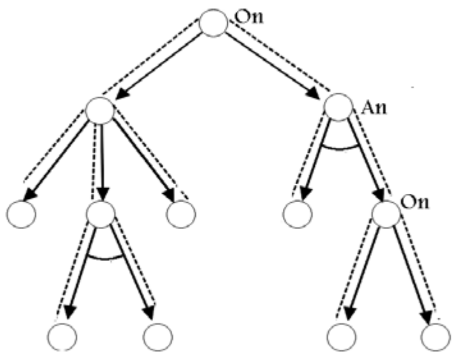


Figure 3.1: A design space tree

The 'A' node (not included in the table) is the root node (decision node) designating the 'OR' decision between 'B' and 'C' and is the 1st level of the hierarchy.

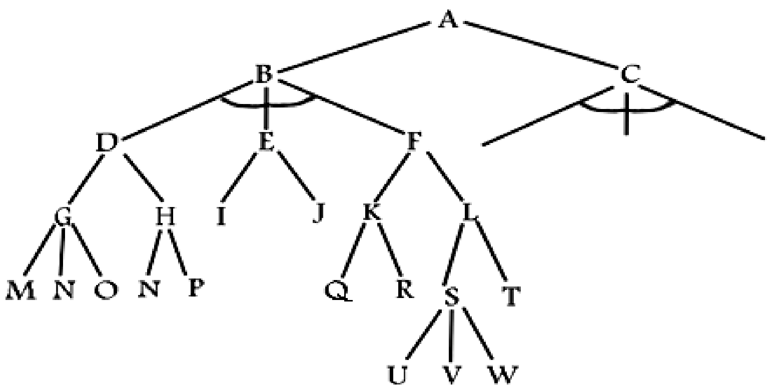


Figure 3.2: Example of a part of a design space tree

Table 3.1: Details of figure 3.2

SC	Description of SC	L	SC	Description of SC	L
B	One arm manipulator	2	M	Rectangular cross sec.	5
C	Conveyor	2	N	I- cross section	5
D	Arm's Shape	3	O	Circular cross section	5
E	Arm's Material	3	P	Tube cross section	5
F	Arm's Control	3	Q	PID controller	5
G	Variable cross section	4	R	PI controller	5
H	Constant cross section	4	S	Fuzzy controller	5
I	Aluminum-based link	4	T	NN controller	5
J	Steel-based link	4	U	Triangle membership fn.	6
K	Linear control	4	V	Trapezoid membership fn.	6
L	Non-linear control	4	W	Gaussian membership fn.	6

It should be noted that the SC 'B' of 'one arm manipulator' spans different CCs, which can be extracted out of the tree. A representation of one such a CC, which is extracted from the tree of figure 3.2, is given in figure 3.3. Its verbal description is: use a one-arm manipulator, with a constant I-cross-section steel arm, with a PID controller.

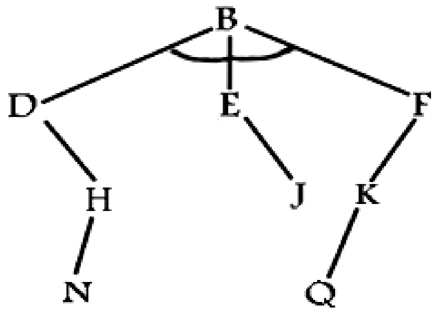


Figure 3.3: Tree representation of an extracted CC

A SC may have more than one detailed abstract description. For example, a PID and PI controllers are both more detailed descriptions of the same SC- 'linear control' (see node 'K' of fig 3.2). These additional details are represented by an 'OR' operation, as it makes two different abstract alternatives under the SC of a 'linear control'. SCs may also have 'AND' operator. For example the 'AND' operator between the sons of the SC located at node B of figure 3.2. It is noted that a SC may appear in several nodes (e.g., SC ‘N’ in figure 3.2). This is a result of the predefined structure of the tree representation that may include the same SCs in different branches of the tree.

A pruned 'AND' tree of a CC may still represent a concept. Yet, it is a more abstract description than the CC and it lacks a model.

Definition # 4

A Higher-Level Concept (HLC) is a concept which results from pruning the tree representation of a CC.

It is noted that the same HLC might be pruned from several different CC trees. In such a case the CCs that are associated with the same pruned tree are termed associated CCs.

Definition # 5

An Associated CC (ACC) of an HLC is a CC, which contains that HLC.

An HLC is a more abstract description than its ACC/ACCs. Figure 3.4 depicts an HLC (middle graph), and two ACCs (right and left graphs) of the design space. The linear control (‘K’ SC) may be implemented by either a PID controller (‘Q’ SC) or by a P controller (‘R’ SC).

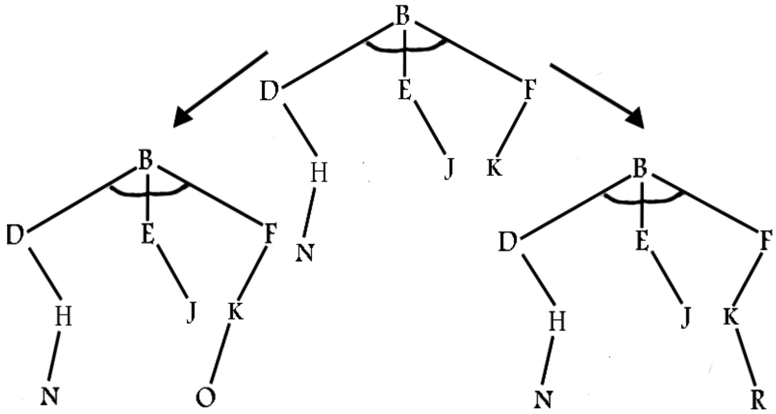


Figure 3.4: An HLC and its ACCs

The verbal description of the depicted HLC is a one-arm manipulator with a constant I-cross-section steel arm and a linear controller (less detailed description of the controller with respect to both ACCs). The different notions used in this section are summarized in table 3.2.

Table 3.2: Summary of notions

Definition #	Notion	Abbreviation	Tree representation
1	Design space		'AND/OR' tree
2	Sub-Concept	SC	Node
3	Complete Concept	CC	'AND' tree
4	Higher Level Concept	HLC	Pruned 'AND' tree
5	Associated Complete Concept	ACC	'AND' tree

3.1.2 Concept types and the 'AND/OR' tree

In this thesis, it is assumed, that when a description of a concept, by way of its SCs, becomes sufficiently detailed, it may be related to a model, involving some variables. Here, a model is a vector of objective functions (see section 3.2.1.2 for details). Solving the model by using values for the functions' variables allows the computation of a concept's solution performances. Therefore for each of the 'AND' trees that are extracted from the 'AND/OR' tree there is such a model.

In this thesis it is assumed that no other concepts other than the CCs have computational models. In other words, in contrast to CCs, HLCs are assumed not to have a unique model, due to their insufficient descriptions. Nevertheless an HLC might be related to one or more CCs, depending on the pruned branches. Multiple models, that are multiple vectors of objective functions, are associated with an HLC whenever an 'OR' operation exists in the original

‘AND/OR’ tree under the pruning location. For example, a fuzzy controller, ('S' SC of figure 3.2) may be implemented by either triangular membership functions ('U' SC of figure 3.2) or by Gaussian membership functions ('W' SC of figure 3.2). These different SCs that are related to a fuzzy controller are associated with different models. Therefore a possible description of such an HLC is: use a one-arm manipulator, with a constant I-cross-section steel arm, with a fuzzy controller.

Definition # 6

Multi-Model Concept (MMC) is an HLC that is associated with more than one ACC

Each MMC has a set of CCs’ models to compute its performances. This means that for an MMC there is more than one vector of objective functions to compute its performances. MMCs are used in this thesis in conjunction with the delayed decision uncertainty (see section 3.5). There, the uncertainty dictates a need to take conceptual decisions while considering the performances of all the MMCs' ACCs,

3.2 Concept-based search and optimization

Commonly the search of 'optimal' solutions within a multi-objective space is referred to as Multi-Objective Optimization (MOO). Usually, the optimization purpose is to find the Pareto front (see section 2.1.3). In the concept-based problem, the front may be associated with solutions from more than one concept. Therefore the search has to consider not just optimality but also the relation of the solutions of the optimal set, to concepts, which are in this section; CCs (see definition #1 in section 3.1.1). In the following, a formal representation of a C-MOP is given. This formalization is equivalent to the one in Mattson and Messac, (2005) and is given here for the sake of clarity and completeness. Thereafter, this formulation is discussed with respect to both a simultaneous and a sequential approach. Next, two new algorithms, which are hereby termed C_1-NSGA-II and C_2-NSGA-II are presented and tested. They involve a new approach to the resource-sharing problem between and within concepts, while evolving towards and along a Pareto front. It is important to note that, in contrast to any former concept-based work, here mating among concepts is avoided, which leads to a possible comparison between concepts and species. Such a comparison is suggested in recent papers (Moshaiov 2006a, b).

3.2.1 Problem definition and solution approach

In this section the distinction between a C-MOP and the classical MOP is highlighted. Furthermore, two alternatives for generating the concept optimality related front (s-Pareto) are discussed including: a sequential and a simultaneous approach. It is noted that the current investigation is based on the second approach.

3.2.1.1 Classical MOPs

In the classical multi-objective search problem, such as dealt with in Coello (2005), the set of Pareto optimal solutions is sought from the set of all possible particular solutions. Any particular solution is characterized by specific values of the problem decision variables representing a point in the problem decision space. The set of Pareto optimal solutions is found by comparing the performances of all particular solutions in the objective space for non-dominance. The representation, in the objective space, of the set of non-dominated solutions is known as the Pareto front. A classical MOP is commonly formalized as follows:

$$\min F(x) \tag{3.1}$$

$$\text{s. t.} \qquad x \in X \subseteq S \subseteq R^n$$

where x is the vector of decision variables. In general x might be subjected to equality and/or inequality constraints, which commonly include some bounds on the decision variables. A solution $x \in X \subseteq S \subseteq R^n$, which satisfies all the constraints, is called a feasible solution. The set X, of all feasible solutions, is called the feasible region in the search space S. A MOP deals with minimizing of F(x), which is the vector of K objective functions where,

$$F(x) = [f_1(x), f_2(x), ..., f_K(x)]^T \qquad K \geq 2 \tag{3.2}$$

It can be shown that problems involving maximization, or a mixture of both min and max with respect to different objectives, may be easily transformed into the above problem. Furthermore, it should be noted that usually, due to contradicting objectives, there is no single solution to the above problem. In the classical MOP the focus is therefore on the trade-offs with respect to the objectives. The well-known concept of Pareto dominance supports exploring such trade-offs. The development of an optimality-based Pareto front in the objective space is based on a comparison between solutions using the idea of vector domination. Under minimization a vector $u = (u_1,...,u_N)$ is said to dominate $v = (v_1,...,v_N)$, denoted by $u \preceq v$, iff u is partially less than v, i.e., $\forall j \in \{1,...,N\}, u_j \leq v_j \wedge \exists j \in \{1,...,N\} : u_j < v_j$. If u dominates v in the objective space then the corresponding solution of u is considered a better solution than the one corresponding to v (with respect to the minimization problem).

The Pareto optimal set, P^*, is the set of optimal solutions to the classical MOP such that:

$$P^* := \{x^* \subseteq X \mid \neg \exists\, x' \in X : F(x') \preceq F(x^*)\} \tag{3.3}$$

As declared above, the performances of the optimal solutions constitute a set of points within the objective space, which is termed Pareto front. The Pareto front set PF^* is defined as:

$$PF^* := \{y^* \in Y \mid y^* = F(x^*) \quad : x^* \in P^*\} \tag{3.4}$$

A solution that its performances are included in the Pareto front set is a Pareto optimality solution.

3.2.1.2 Concept-based MOPs

In a C-MOP, which is equivalent to the s-Pareto definition of Mattson and Messac 2003, the interest is not on the performances of particular solutions, as done in the classical MOP. The focus here is rather on obtaining the set of performances of each of the conceptual solutions that are distributed along the Pareto-front, or at least a representative set for each such concept. The notion of a conceptual solution, as understood in engineering design, (e.g., Mattson and Messac 2003, Andersson 2001), is associated with abstract ideas, which are generated by humans, describing a generic solution to a problem. Due to its inherent lack of details it is difficult, and often impossible, to evaluate a conceptual solution in the regular sense of performances. In fact the selection of a conceptual solution is commonly done based on human experience and preferences, without the use of computational means. Traditionally, during a conceptual planning or design stage, no particular solution is explored. In comparison with conceptual solutions, particular solutions are sufficiently detailed such that each has a one-to-one relationship with a point in the objective space (as in the classical MOP). In such problems, the association of multiple particular solutions with a concept, together with the association of each particular solution with a point in the objective space, constitutes a one-to-many relationship between a conceptual solution and the objective space. In such a case, a C-MOP can be defined, in which the distribution of solutions that represents concepts along a Pareto front is sought.

The concept-based search concerns conceptual solutions that can be represented by sub sets of the set of particular solutions of the problem. It is assumed that the performances of each particular solution are computable. Each conceptual solution, and its associated particular solutions, may be characterized by different models and/or range of variables. It is noted that the models, either identical or not, map the concepts solutions' performances to an objective space, which is mutual to all concepts. It is recalled that such a concept has been termed a CC (see definition # 2 in section 3.1.1). For example, the problem of storing some powder is considered. Two possible concepts are examined. The first is of cylindrical containers and the second is of prismatic containers, with rectangular base. The decision variables of the first concept are: the radius of the base, and the cylinder height. The second concept involves the width and the breadth of the rectangular base as well as the height of the prismatic container. Obviously, these concepts are not represented in the same decision variable space, yet they are examined with respect to the same objectives, hence they share the same objective space. For example, consider the MOP, which involves the maximization of the volume, while minimizing the weight of the empty containers. It is clear that the models to calculate the performances differ from one concept to the other. Another way of posing a C-MOP is to consider two concepts for the above MOP, which may share the same decision variable space and models. One may define two different concepts, both involving prismatic containers. The first is of large base and short height and the second is of small base and medium height. It should be noted that the first MOP example might be reformulated as a constrained problem within the 'same' variable space. In other words, a circle equation could serve as a constraint for the base variables of the cylindrical container, within a Cartesian coordinate system. Yet, this is not the general case as concepts to solve a MOP may be as remote as a plane and a car; both are valid concepts to solve a traveling problem between two cities.

Here, n_c sets of decision variables are used; one set for each CC. The m-th set for the m-th CC, of all its feasible solutions, is denoted X_m, where $X_m \subseteq S_m \subseteq R^{n_m}$, and S_m is the search space of the m-th CC. X_m contains the decision variable vectors, x^m, of the m-th CC, as follows:

$$x^m \in X_m \quad m = 1,\ldots,n_c \tag{3.5}$$

where the dimension n_m of the vectors, x^m, is in general concept dependent. The set X is the union of these n_c sets, i.e.,

$$X = \bigcup_{m=1}^{n_c} X_m \tag{3.6}$$

The vector of objective functions $F: \ X \rightarrow Y$ is given as follows

$$F(x) = F^m(x), \qquad \text{for } x = x^m, \ m = 1,\ldots.., n_c \tag{3.7}$$

where $F^m(x^m) = [f_1^m(x^m), f_2^m(x^m), ..., f_K^m(x^m)]^T$: $X_m \rightarrow Y$, for m = 1,..., n_c , is the mapping, into the objective space, of the particular solutions that are associated with the m-th concept and

$$\left\{ \; y \in Y \;\middle|\; y \in R^K \; \right\} \tag{3.8}$$

The mapping of the m-th concept is done by using a set of concept related objective functions with $f_k^m(x^m)$ (or in short f_k^m) as the k-th objective function. The above exposition allows defining a C-MOP similarly to the classical problem. A C-MOP is defined as the problem stated in (3.1), with the minimization of F(x) as defined in (3.7) and (3.8), subject to (3.5) and (3.6). C-MOPs involve finding the Pareto-optimal concepts, where a Pareto-optimal concept is defined as a CC with at least one member of its sub-set being a non-dominated solution with respect to the entire feasible set of solutions. The solution to a C-MOP is termed the concept-based Pareto set, and is designated as P_C^*. Similarly, the associated front is termed the Concept-based Pareto Front (CBF), and is designated as PF_C^*. These are defined as follows:

$$\begin{gathered} P_C^* := \{x_m^* \in X | \neg\exists\, x_i' \in X_i : F^i(x_i') \preceq F^m(x_m^*),\; m \in \{1,...,n_c\} \text{ and } i = 1,...,n_c\} \\ PF_C^* := \{y^* \in Y | y^* = F^m(x_m^*) \quad : x_m^* \in P_C^*\} \end{gathered} \tag{3.9}$$

These definitions coincide with that of the s-Pareto (Mattson and Messac 2005), and are given here for the sake of completeness and clarity.

A C-MOP differs from the classical MOP in several aspects, as indicated in the example of the containers given above. First, a C-MOP involves the possibility of several decision variable spaces of different dimensions, whereas the classical problem is commonly defined with one such space. Second, even under a restricted case of one decision space the search is inherently divided into different regions to explore the behavior of concepts rather than just specific solutions. Finally, the end result, which is a Pareto set and its associated front, should provide an understanding of the distribution of CCs' representatives on the front, rather than just specific solutions. The resulting fronts could be categorized as suggested in the following section. The introduced categorization is important when considering resource sharing within EMO (see section 3.2.2.2).

3.2.1.3 Types of Concept Based Fronts

CBFs could be categorized into two different types of fronts. In the first type any point of the front is associated with a solution or solutions of one CC only: $\forall x_m^* \rightarrow y^* \neg\exists x_{m'}^* | x_{m'}^* \rightarrow y^*$ for all $m \neq m'$. This type of front is hereby termed: 'non-intersecting front.' In the second type, there exists at least one point of the front, which represents the performances of solutions that are related to more than one CC: $\exists x^*_m \rightarrow y^* | \exists x^*_{m'} \rightarrow y^*$ for all $m \neq m'$. This front is hereby termed 'intersecting front.' The two types of fronts are depicted in figure 3.5 for a bi-objective space, where performances of different CCs are designated by different colors (black and gray).

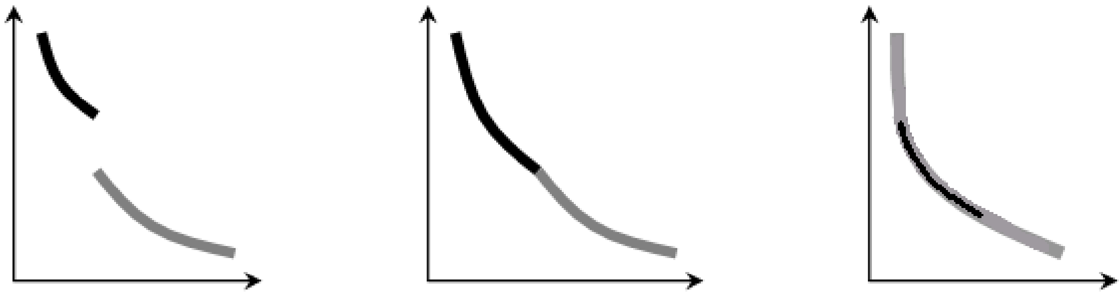


Figure.3.5: a. Non-intersecting front b. Intersecting front: case 1 c. Intersecting front: case 2

A primitive case of the non-intersecting front is the uni-concept front in which there is only one optimal CC. Two approaches to obtain CBFs are discussed below.

It is noted that in the C-MOP, it is important to find all 'optimal' CCs even that some may cover a smaller sub-space of the front than others (such as the case in figure 3.5.c). This is due to the fact that the DMs may prefer any of the 'optimal' CCs based on un-modeled hazards or as a result of subjective preferences. The reader is referred to section 3.2.2.2 for explanations on how the proposed algorithms apply a search pressure towards the representation of all of the 'optimal' CCs on the CBF.

3.2.1.4 The sequential approach

The sequential approach to solve a C-MOP consists of auxiliary problems. According to this approach the problem is reformulated into two stages. In the first stage each CC is considered independently, in a fashion similar to the classical MOP. In other words, for each of the m-th CCs (m=1,…., n_c), the MOP is solved to independently minimize $F^m(x^m)$. This results in n_c Pareto sets and associated fronts. The Pareto set of the m-th CC, P^*_m , and its associated Pareto front, PF^*_m, are defined below:

$$
\begin{aligned}
P^*_m &:= \{x^*_m \in X_m | \neg\exists\, x_m{}' \in X_m \;: F^m(x_m{}') \preceq F^m(x_m)\} \\
PF^*_m &:= \{y^* \in Y | y^* = F^m(x^*_m) \quad : x^*_m \in P^*_m\}
\end{aligned}
\tag{3.10}
$$

Finding P_m^* for each of the CCs is an auxiliary stage in solving a C-MOP sequentially. In the second stage the union of the CCs Pareto sets $P_{uc}^* = \bigcup_{m=1}^{n_c} P_m^*$ is obtained. Thereafter, the Fronts Union (FU) set, is found by $FU = \bigcup_{m=1}^{n_c} PF_m^*$ and dominance sorting among the members of FU is performed to finally obtain the combined Pareto set, P_{CC}^* and the related combined front, PF_{CC}^*. These are formulized as follows:

$$\begin{gathered} P_{CC}^* := \{x_m^* \in P_{uc}^* \,|\, \neg\exists\, x_i^{'} \in P_{uc}^* \;: F^i(x_i^{'}) \preceq F^m(x_m^*),\; m \in \{1,\ldots,n_c\} \text{ and } i = 1,\ldots,n_c\} \\ PF_{CC}^* := \{y^* \in Y \,|\, y^* = F^m(x_m^*) \quad : x_m^* \in P_{CC}^*\} \end{gathered} \tag{3.11}$$

Given the nature of the sequential approach, which is based on auxiliary problems, it has to be proven that the solution (3.11) is equal to the solution (3.9) of a C-MOP. In other words it has to be proven that the CBF is identical to the combined front. The equality of the sets is intuitively true, as the 'order of the sorting' should not affect the results. In fact (3.9) does not suggest any particular order and therefore the order of sorting, which is inherent to the definition of (3.11), is not in any contradiction with the definition of (3.9). The sequential approach has a primitive form, which is suitable for problems with graphical representations (objective space up to 3-D). In this case only the auxiliary problems are computed and the resulting fronts are graphically combined (e.g., Andersson 2001).

Figure 3.6 depicts the independent fronts of three CCs, CC_1, CC_2, CC_3, and their CBF/combined front in a bi-objective space for a min-min problem.

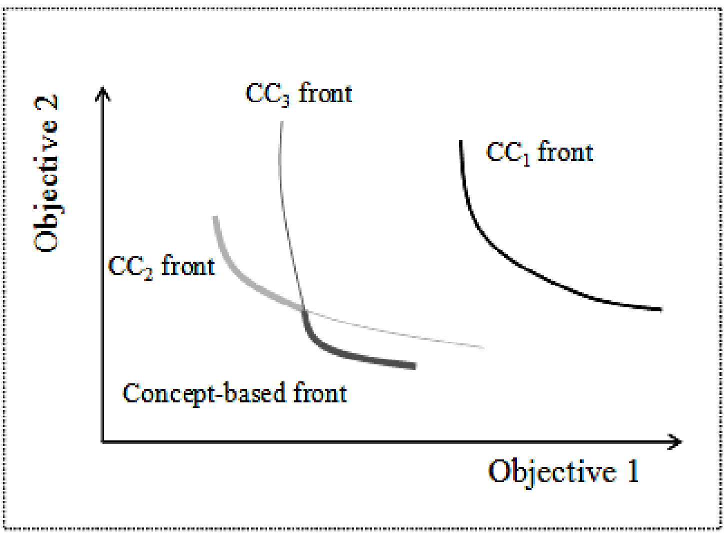


Figure 3.6: CCs' fronts and a CBF/combined front

The Pareto front of CC_1, is designated by the black color while the Pareto fronts of CC_2 and CC_3, are designated by light gray and dark gray, respectively. The CBF/combined front is distinguished from the CCs' fronts by thicker curves. It is constructed from parts of the fronts of CC_2 and CC_3. Using the sequential approach, for the problem of figure 3.6, seems to involve a waste of resources. This is due to the fact that the sought front, namely the combined front, does not involve any parts of the front of CC_1, and large sections of the obtained fronts of CC_2 and CC_3 are excluded as well.

3.2.1.5 The simultaneous approach

The purpose of the simultaneous concept-based approach is to reach, by a simultaneous evolution of the concepts, the Pareto-optimal set or at least its approximation, with adequate representation of the concepts. The term 'simultaneous' indicates that all investigated concepts are participating during the same evolution process. The simultaneous approach aims at reaching the sets defined in (3.9) without reaching first those of (3.10). A major advantage of the simultaneous approach is that it avoids the development of fronts of ‘non-optimal concepts’. The term 'non-optimal concept' refers to CCs that have no representatives on the CBF, whereas 'optimal concept' refers to a CC with at least one such representative. It should be noted that the simultaneous approach should not be considered just as an alternative to the sequential approach, but mainly as an approach that supports progressive interactivity. In section 3.2.2 the simultaneous C-EMO is dealt with, and two algorithms are presented.

3.2.2 Simultaneous concept-based MOEA

In this section two algorithms, C_1-NSGA-II and C_2-NSGA-II, for the simultaneous approach to the solution of a C-MOP are introduced and explained. In additions, comparison measures to compare the proposed algorithms, and that of the sequential approach, are introduced. The aim of the concept-based search is to find a front as defined in equation 3.9. In implementing a MOEA approach care should be given such that the algorithm will search for a proper representation of the optimal concepts and their spread on the CBF. Section 3.2.2.1 provides an outline of the requirements that aim at reaching an adequate approximation to the problem solution. In section 3.2.2.2 the basic algorithm C_1-NSGA-II is presented, and subsequently modified in section 3.2.2.3 to C_2-NSGA-II, for the purpose of saving computational time.

3.2.2.1 Requirements

While searching simultaneously for the CBF, attention should be given to the following requirements:

1. Maintaining a search pressure towards optimal solutions
2. Ensuring representation of all optimal concepts on the CBF in both potential types of fronts (see section 3.2.1.3).
3. Ensuring diversity of solutions within each CC on the front
4. Maintaining a transverse search pressure to ensure a balanced representation of CCs (see explanation in section 3.2.2.2)
5. Minimize computational efforts.

These requirements are addressed by the proposed algorithms, and are detailed and explained in sections 3.2.2.2 and 3.2.2.3.

The following discussion explains the ability of the suggested algorithms to allow the simultaneous evolution of CCs and presents the way by which the above requirements are met.

3.2.2.2 The basic algorithm – C_1-NSGA-II

The suggested algorithm, which is termed C_1-NSGA-II, is a modification of NSGA-II, (Deb *et al*., 2002) that allows the simultaneous evolution of CCs. The pseudo code of the algorithm and its explanation by steps are given below:

C₁-NSGA-II

a. Initialize populations P_t with n_c equal sub-populations, each per CC, and set the population size as n= $|P_t|$. Also, create $Q_t = P_t$
b. Combine parent and offspring populations and create $R_t = P_t \cup Q_t$.
c. Decode all individuals to obtain a population of solutions X_t and compute their performances Y_t, using their concept-related objective functions, $X_t \rightarrow Y_t$.
d. Perform a non-dominated sorting for Y_t and find fronts, Fr_i, i=1,…,n_r where n_r is the number of fronts in a generation.
e. Initialize a new parent population $P_{t+1} = \varnothing$. Set a non-dominance level counter i=1. While $|P_{t+1}| + |Fr_i| \le n$, include the i-th front in the new parent population: $P_{t+1} = P_{t+1} + Fr_i$ and set i=i+1.
f. Perform the 'Concept-based Crowding Sort' procedure (as outlined below), and complete the filling of P_{t+1} with the most widely spread $n - |P_{t+1}|$ solutions using the 'Concept-based Crowding Distance' (as outlined below).
g. Create offspring population Q_{t+1}^{*} from P_{t+1} by a 'Concept-based Tournament Selection.'
h. Perform 'In-concept Crossover' (as outlined below) to obtain Q_{t+1}^{**}

from Q_{t+1}^{*}.

i. Perform 'Two Regimes Mutation' (as outlined below) to obtain Q_{t+1} from Q_{t+1}^{**}.

j. Go to b.

The following provides an explanation to the algorithm steps.

Step 'a': In the simultaneous approach the initial generation should contain individuals from all the CCs. An initial population, P_t is generated in step 'a'. The population is constructed out of n_c sub-populations (n_c is the number of CCs of the problem). In the initial populations, the size of each of the sub-populations is the same. Each of the sub-populations has its CC's individuals. The values of the CC parameters are encoded within the individuals of the CC's sub-population. Another initial population Q_t is created by duplicating P_t .

Step 'b': The two populations, P_t and Q_t are combined into one population, R_t. Due to the fact that the cross-over is done within each sub-population, the population, Q_t, holds the same number of solutions from each sub-population as P_t. Therefore R_t has a size which is twice the size of the population of P_t or Q_t and its sub-populations are twice the size of those of P_t or Q_t .

Step 'c': At this step the individuals of R_t are decoded into a set of solutions, X_t, and their performances, are computed in accordance with their CC, and memorized in Y_t. It is noted that (see section 3.2.1.2) a CC may differ from the other CCs by the models that are used for the calculations of the performances in accordance with the problem's objectives.

Step 'd' and 'e': In step 'd' the combined population R_t is sorted into levels of non-dominance , with a front Fr_i for the i-th level. In step 'e' an elitist population, of size n, is constructed out of the ordered non-dominance levels. This sorting and elitism causes a pressure towards optimal solutions. This supports the fulfillment of requirement 1.

Step 'f': In this step the Concept-based Crowding-sort is implemented. For this purpose two crowding distances, $D1_{j,k}^{i,m}$ and $D2_{j,k}^{i,m}$ are defined. These are defined for each objective k, for each front, Fr_i, for each of the CCs, and for each non-boundary individual j (where $2 < j < n_m^i - 1$, and n_m^i is the number of solutions of the m-th CC in the i-th front) as follows:

$$D1_{j,k}^{i,m} = \frac{f_k^m(I_{j+1,k}^{i,m}) - f_k^m(I_{j-1,k}^{i,m})}{\max f_k^{i,m} - \min f_k^{i,m}} \qquad D2_{j,k}^{i,m} = \frac{f_k^m(I_{j+1,k}^{i,m}) - f_k^m(I_{j-1,k}^{i,m})}{\max f_k^{i} - \min f_k^{i}} \tag{3.12}$$

where $\max f_k^{i,m}$ and $\min f_k^{i,m}$ are the max and min performances respectively, over all the individuals of the m-th CC in the k-th objective of the i-th front. In addition, $\max f_k^{i}$ and $\min f_k^{i}$ are the max and min performances respectively, in the k-th objective of of the i-th front over all CCs. The terms of the numerators, in the above equation, are the performances of the individuals that are located in the sorted lists $I_k^{i,m}$, which are obtained by the procedure of the concept-based crowding-sort. This procedure and its explanation are given below:

Concept-based Crowding Sort

1. Set $n_m^i = \left|Fr_i^m\right|$.
2. For each j-th individual, of the m-th surviving sub-population in the i-th rank, initially assign an objective-based distance in the k-th objective, $CD_{j,k}^{i,m} = 0$.
3. For each objective k=1,2,..K, sort the set of solutions, in the i-th front, in each of the m-th surviving CC (sub-population) m=1,2…$m_{s,i}$, by f_k^m : $I_k^{i,m} = \text{sort}(f_k^{i,m}, >)$.
4. Find the Concept-based Crowding Distance as follows:

   a. For each objective k, in each of the m-th CC of each front, Fr_i, assign a large distance to the boundary solutions, $CD_{1,k}^{i,m} = CD_{n_m^i,k}^{i,m} = \infty$, and for all other solutions of the m-th CC, j=2 to ($n_m^i - 1$), compute two distances, $D1_{j,k}^{i,m}$ and $D2_{j,k}^{i,m}$, using equation 12.

   b. Assign at each generation, t, all solutions with $2 < j < n_m^i - 1$, of the m-th CC of the i-th front, with the following concept-based crowding distance, $CD_j^{i,m}$:

$$CD_j^{i,m} = \eta \sum_{k=1}^{K} D1_{j,k}^{i,m} + (1-\eta) \sum_{k=1}^{K} D2_{j,k}^{i,m} \tag{3.13}$$

$$\text{and} \qquad CD_{1}^{i,m} = CD_{1,k}^{i,m} = CD_{n_m^i}^{i,m} = CD_{n_m^i,k}^{i,m} = \infty$$

where $\eta = (\frac{t}{n_{gen}})$, $\eta \in [0,1]$, is a generation-based tuning factor and n_{gen} is the total number of generations in the evolutionary run.

In step 'f' of the pseudo-code of C_1-NSGA-II, above, the requirements 2, 3, and 4 are all addressed as explained in the following. The concept-based crowding distance is influenced by two terms that involve η. Both compute distances between a solution neighbors in the objective space, yet the normalization is different. While η is small (at the beginning of the evolution) the normalization is based on the boundaries of the performances of the CC. While η grows the computation of the distance is becoming increasingly dependent on a normalization that is based on the boundaries of performances of the entire front. The effect and role of these generation dependent terms is explained below, using figures 3.7a, 3.7b and 3.7c.

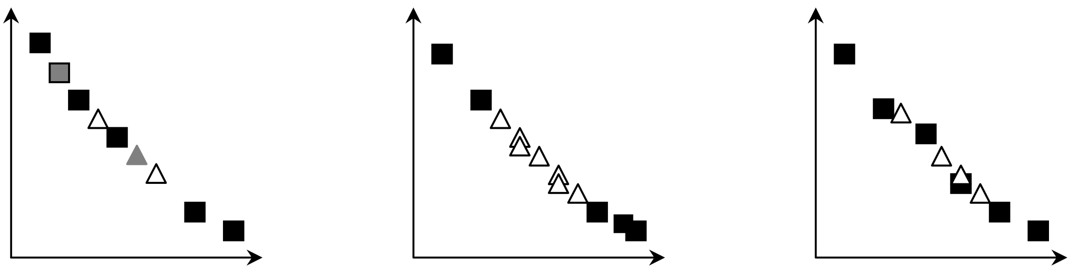


Figure 3.7: Crowding in the concept-based multi-objective evolution

a: CCs' in a front b: Over-crowded CC c: Similarly-crowded CCs

Figure 3.7a depicts performances of solutions, belonging to two CCs (designated by squares and triangles). The CCs representatives are located at the same level of non-dominance. It is noted that the squares have a much wider spread of solutions than that of the triangles. In other words the difference between the $\max f_k^i$ and the $\min f_k^i$ is larger for the 'square' CC. Considering two particular representatives, marked by a gray color, the following observations are made. First, the gray square has two close neighboring squares, and similarly the gray triangle has two triangle neighboring points. Second, the Manhattan distances between the neighbors of the gray square are similar to those of the neighbors of the gray triangle. Nevertheless, the gray triangle representative will have a larger concept-based crowding distance at the beginning of the evolution (η<<1). This effect, by the first term of equation 3.12, is due to the normalization that is based on the boundaries of the CC, which are closer for the 'triangle' CC than those of the 'square' CC. In such a case the normalization has two effects. The first effect is that niched solutions of a CC, which occupies a small section of the front, will not be penalized as much as those of a CC that occupies a larger section of the front. This prevents hindrance of the development of the former CC and helps meeting requirement 2. The second effect of that normalization is the increased number of the representatives of the CC that occupies the smaller section in the next generation. This helps the search of solutions, which are related to that CC. Such an increase of the number of representatives supports the fulfillment of requirement 4 by increasing the CC resources, hence allowing expansion as needed by the CC boundaries. Furthermore, the different crowding of the individuals of the same CC will enhance their spread within the CC section on the front (see requirement 3). In other words, the first crowding distance, of equation 3.12, ensures a pressure towards the sharing of the fronts by several CCs and the spread of CCs solutions on the front. Yet, referring to figure 3.7b, in the case where the 'triangle' CC occupies a smaller part of the CBF than that of the 'square' CC, an over-crowding of representatives from the 'triangle' CC may be caused. Therefore as generations progress a better share of resources should take place. This is achieved by the second expression of the crowding distance of equation 3.12, which depends on the normalization by the front boundaries. It supports a balanced spread of solutions on the front in accordance with their CC true part on the front. The desired result is depicted in figure 3.7c, where similar density of representation is achieved for both concepts.

It is noted that step 'f' for the calculation of the concept-based crowding distance is based on the distance between neighboring individuals within the same CC. In other words, no penalization is taken place due to neighboring solutions from different CCs. Hence, requirement 2 is fulfilled not only in non-intersecting fronts but also in cases in which optimality involves intersecting CCs.

It is noted that steps 'e' and 'f' guarantee the survival of low ranked solutions, which are less crowded with respect to other solutions of the same ranks. It is most likely that in later generations the entire population involves solutions, which are all in the lowest level of non-dominance. In such a case step 'e' is avoided (automatically) and step 'f,' allows the elitist population to include just the most uncrowned solutions. It is further noted that in the current work, which uses NSGA-II with less as possible alternations, the crowding in the objective space is considered, while crowding within the design space is ignored and left for future work (see chapter 5).

Step 'g': At this step a concept-based tournament selection (finding Q_{t+1}^{*} from P_{t+1}**)** is performed as follows:

Concept-based Tournament Selection

Choose randomly two solutions from P_{t+1} and compare them by tournament as follows:

1. If their non-dominance levels are different the solution with the lower non-dominance level is the winner and is placed in Q_{t+1}^{*}.
2. If their non-dominance levels are equal the solution with the larger concept-based crowding distance is the winner and is placed in Q_{t+1}^{*}.

Step 'h': Individuals, which belong to different CCs, might have differences that are outlined in section 3.2.1.2, as well as in the explanation of steps 'a' and 'b' of C_1-NSGA-II. Therefore, mating between individuals is restricted to individuals belonging to the same CC. This intra-sub-population mating, which is performed in the current step, is termed in-concept crossover.

Step 'i': Following the in-concept crossover, mutation is implemented. The mutation rate, used here, is altered during the evolution. Initially a high mutation rate is implemented, and after a predefined percentage of the generations a smaller mutation rate is used. The initial mutation regime ensures a diverse search of the space to prevent a premature convergence to CCs that have initial representatives at lower levels of non-dominance on the expense of 'optimal concepts' that might have initial representatives at higher levels of non-dominance. This premature convergence, which might hinder the evolution of 'optimal concepts,' is avoided by the proposed mutation regimes and thus supports requirement 2.

C_1-NSGA-II addresses requirements 1-4. Addressing requirement 5 needs a special consideration and a modification of the algorithm as detailed below.

3.2.2.3 Saving computational resources – C_2-NSGA-II

The computational complexity that influences the search run-time, using a traditional MOEA, depends primarily on the number of generations, and the size of the population (e.g., Jensen 2003). In such a traditional case, it is assumed that there is no substantial difference in the computational time of the performances from individual to individual. In the simultaneous C-EMO the run-time is influenced by the possible differences, from CC to CC, among the computational efforts that are associated with calculating their solutions performances. In C_1-NSGA-II the total number of individuals, which are divided among all CCs in the initial population, is likely to be unequally redistributed among the CCs during the run and in the final population. This means that the number of individuals of an 'optimal concept' may increase with generations. In such a case the representation of that 'optimal concept' would be denser than its representation by the sequential approach (see section 3.2.3), when taken with the same initial population size, per CC, as in the simultaneous case. This may be viewed as an advantage however, there is no free lunch!! There is an associated computational cost to this 'advantage.' Assuming that there is no need for a CC representation by more individuals than in the initial sub-population, the extra individuals should be viewed as a waste of resources. A possible solution is to limit the growth of the sub-populations of the CCs to include no more individuals than the number of the initial sub-population. Such a modification is given in the following pseudo-code:

C_2-NSGA-II

a. Initialize populations P_t with n_c equal sub-populations, each per CC, and set the population size as n= $|P_t|$. Also, create $Q_t = P_t$ and Set $n_{max} = n/n_c$ as the maximal CC population.

b. Combine parent and offspring populations and create $R_t = P_t \cup Q_t$.

c. Decode all individuals to obtain a population of solutions X_t and compute their performances Y_t, using their concept-related objective functions, $X_t \rightarrow Y_t$.

d. Perform a non-dominated sorting for Y_t and find fronts, Fr_i, i=1,…,n_r where n_r is the number of fronts in a generation.

e. Initialize a new parent population $P_{t+1} = \emptyset$. Set a non-dominance level counter i=1. While $|P_{t+1}| + |Fr_i| \leq n$, include the i-th front in the new parent population: $P_{t+1} = P_{t+1} + Fr_i$ and set i=i+1.

f. Initialize a new parent population $P_{t+1} = \emptyset$. For each of the m-th CC initialize a counter C_m =0.

g. While $n - |P_{t+1}| > 0$ or all individuals are removed from R_t. For each non-dominance front Fr_i , (i=1,…., n_r):

   1. Include the most widely spread solution, $sol_j^{i,m}$, in the new parent population: $P_{t+1} = P_{t+1} + sol_j^{i,m}$ if its CC counter $C_m \leq n_{max}$, else go to 3.
   2. Update the CC counter, $C_m = C_m + 1$, of the individual j.
   3. Remove the individual from R_t
   4. Continue to the other individuals of the front.

h. Update the size of the population for the next generation: $n = \sum_{m=1}^{n_c} | C_m |$

i. Create offspring population Q_{t+1}^{*} from P_{t+1} by a 'Concept-based Tournament Selection,' (as outlined in section 3.2.2.2).

j. Perform 'In-concept Crossover' (as outlined in section 3.2.2.2) to obtain Q_{t+1}^{**}.

k. Perform 'Two Regimes Mutation' (as outlined in section 3.2.2.2) to obtain Q_{t+1}.

l. Go to b.

The following provides an explanation to the steps of the C_2-NSGA-II algorithm.

Step 'a' –'d': These steps are similar to the C_1-NSGA-II algorithm, except from setting the maximal size of a CC sub-population, at step 'a'.

Step 'e': In contrast to steps 'e' and 'f' of C_1-NSGA-II, this step includes only the 'Concept-based Crowding Sort.' In other words, here while $|P_{t+1}|+|Fr_i| \leq n$ the individuals of the fronts, in the modified algorithm, are not added automatically. The number of individuals from each CC, which are allowed to continue through the evolution, is limited by their initial number in the population.

Step 'f': Apart from preparing a new null population (as done in C_1-NSGA-II), in the current step counters are initialized for each CC. These counters allow limiting the growth of the sub-populations to their initial size (upper limit).

Step 'g': At this step individuals are added one by one to the new population (the elitist population). The inclusion starts with the lower non-dominance level based on the order of crowding. Less crowded individuals are added first, with a consideration to the limit of the allowed maximal amount of individuals per CC, as achieved by the CC counter. The procedure in step 'g' guaranties that a solution, which is relatively un-crowded, survives the evolution and passes to the next generation provided its CC does not have too many representatives in the population. By this modification, in comparison with C_1-NSGA-II, requirement 5 on computational efforts is addressed.

Step 'h': At this step, the number of individuals for the new population is counted. This number might be either the same or less than the size of the initial population.

Step 'i', to 'l': These steps are similar to steps 'g' to 'j' of the C_1-NSGA-II algorithm.

3.2.2.4 Performance indicators for concept-based representations

Comparing between different MOEAs is commonly done using performance indicators, (e.g., Bosman and Thierens, 2003). These indicators involve values that represent the success of an algorithm to find the Pareto front. Proximity and diversity are the main issues concerning this representation. Here we modify known indicators to fit the special nature of the concept-based search. The indicators that are used in the current investigation are:

1. Number of representatives of a CC on the first non-dominance level (front), $n_m^{Fr_1}$. This indicator is a modification of the 'front occupation indicator' of Bosman and Thierend, (2003). When using such an indicator it is assumed that a better result is associated with

more alternatives. For each of the m-th CC, the indicator CFO_m, which is the Concept Front Occupation indicator, is computed as:

$$CFO_m = \left|Fr_1^m\right| \tag{3.14}$$

A high CFO is preferable.

2. The front spread indicator (Van Veldhuizen, 1999) is adapted to the concept-based approach. The Euclidian distance between the most distant representatives of each CC, D_m, is measured on the front, in the objective space. It serves as an indicator to the ability of the algorithm to find the bounds of the CCs. A large D_m is preferable.

To better understand the results, shear numerical results for the second indicator is avoided, and the indicator is normalized into D_m^*, which is termed the CC spread indicator. The indicator is computed by:

$$D_m^* = \frac{D_m}{D_m^{max}} \tag{3.15}$$

where D_m^{max} is the maximum distance between solutions of the m-th CC on the CBF, which is computed analytically. A higher D_m^* means that the algorithm better succeeds to find the analytical front and hence is a better representation.

3.2.2.5 Computational time

The total computational time, T_c, includes both the time to compute the performances and the time to run all the rest of the algorithm steps. Often, in real life situations, evaluating the performances is the most significant computational time effort. Moreover, as discussed in section 3.2.2.3, the computational time of the C-EMO is influenced by the differences between the computational efforts that are associated with calculating the solution performances of different CCs. In such a case this time might strongly vary from CC to CC depending on the complexities of their models. Therefore a comparison of the computational time of the three presented algorithms should take it into account. The reader is referred to the indicative situations, which are presented in the study case section (see section 3.2.4.3), for a further discussion on the comparison.

The comparison of the computational time is based on the following measure. Let T_{sq} be the time to conclude the development of a CBF by the sequential approach (see section 3.2.3). Also, let T_{sm-1} be the time to find that front using C_1-NSGA-II and T_{sm-2} be the time needed to evolve the front by C_2-NSGA-II. To compare the results a time ratio, τ, is computed for each of the algorithms by:

$$\tau = \frac{T_c}{\min(T_{sq}, T_{sm-1}, T_{sm-2})} \tag{3.16}$$

where $\min(T_{sq}, T_{sm-1}, T_{sm-2})$ is the shortest computational time among all algorithms for the same problem. The simultaneous approach is compared to a sequential one which uses the following algorithm.

It is noted that intuitively it seems that a simultaneous approach should outperform the sequential approach especially as related to the computational time. Nonetheless as investigated and demonstrated in section 4.1.3, this is not as straightforward.

3.2.3 A Sequential MOEA

In this thesis the sequential search, which is used for a comparison purpose, is achieved by the following algorithm:

The sequential algorithm (based on NSGA-II)

a. For each of the m-th CCs of the problem run NSGA-II to find the concept's fronts PF_m^* m=1,…, n_c.
b. Combine all solutions performances of the CCs' fronts to find $FU = \bigcup_{m=1}^{n_c} PF_m^*$.
c. Perform a non-dominated sorting to FU and find the first level of non-dominance, P_{CC}^* which is the CBF.

The sequential approach algorithm is used to perform comparisons with the simultaneous algorithms (see section 4.1). This is done based on the measures, which are presented in section 3.2.2.4.

3.3 Interactive concept-based search and optimization

In this thesis human interactivity is associated with the articulation of preferences towards CCs and SCs. Such preferences influence the survival of CCs together with the model-based performances. In the following, a formal presentation of the Interactive C-MOP (IC-MOP), is provided. The solution to the IC-MOP is addressed with a consideration to both a simultaneous and a sequential approach. Next, human interactivity with respect to conceptual ideas is explained and the concept weight, which represents the DMs' preferences, is introduced. The calculation of the concept weight is explained for two design space decompositions including a hierarchical and non-hierarchical representation. This is followed by the presentation of, a new algorithm, which is hereby termed IC-NSGA-II. It is based on the C_1-NSGA-II algorithm that is modified to incorporate the DMs' preferences by using the concept's weight. This incorporation allows a search that is based on both, performance calculations and DMs preferences articulations. Inherent to the interactive case is the lack of an analytical solution. In such a case validation of the results could be done by checking if different runs result in similar representations. Performance indicators for such testing are presented in section 3.2.2.4.

3.3.1 Problem definition and solution approach

In this section the distinction between the IC-MOP and the classical interactivity, while solving a MOP, is highlighted. The solution to the IC-MOP is presented and discussed.

3.3.1.1 Classical interactive MOP

As surveyed in the literature review (see section 2.4), interactivity is used in conjunction with MOPs for several reasons, commonly driving the evolution to focus the search to sub-spaces of the entire search space. When optimality is considered, the resulting optimal set obtained by the interactive search, is expected to include a sub-set of solutions from the optimal set (from the Pareto-set). In contrast to classical interactive MOPs, the main interactivity element in this thesis concerns directly CCs and SCs as discussed in the next sub-section (see also section 3.3.2 for extra details).

3.3.1.2 Interactive concept-based MOP

In contrast to a C-MOP, IC-MOP takes into account not just computed performances, which are calculated (objectively) by the computer, but also DMs' preferences that are subjective in nature. The concept-related interactivity aims at an efficient utilization of resources to search regions that are not just associated with optimality, as done in the common approaches, but rather with both human preferences towards concepts and optimality. To elucidate the interactive concept-based approach, the following simple C-MOP is discussed. Suppose that the design concerns a manipulator to move an object from one location to the other. It involves conceptual decisions on the manipulator links and their control. For example both an aluminum and steel links are considered as well as using PID or fuzzy controllers. One possible concept that might be selected by the DMs is the use of an aluminum link controlled by a PID controller. Other such combinations may also be feasible concepts. If no preferences are articulated and just optimality is concerned, then the Pareto optimality set should be found considering the entire concepts' sets of solutions. But in another scenario, the DMs might not be interested in using an aluminum link, as they might be worried about its time of supply that could influence meeting the production deadline. As a result it seems that it is reasonable to allocate the available search resources towards conceptual solutions that are associated with steel links. In spite of this preference towards using steel, the concepts that are associated with aluminum links should not be neglected as they might be profoundly superior in the overall comparison with steel links. The suggested interactive concept-based approach, which is described in section 3.3.1.2, aims at directing the search using both model-based optimality and the DMs preferences. This means that the search should focus on optimal concepts' solutions, taking into consideration the preferences of the DMs towards the concepts themselves (either directly or by way of sub-concepts). In the following the IC-MOP is defined and its solution, which is associated with optimality and subjectivity, is given.

An IC-MOP is defined, as follows:

$$\max \ \Psi(x) \qquad (3.17)$$

$$\text{s. t.} \quad x \in X \subseteq S \subseteq R^n$$

where X is defend in equation 3.6 and

$$\Psi(x) = f(\text{Rank}(x), H(x)) \quad \text{and } H(x) = H_m \text{ for } x \in X_m \qquad (3.18)$$

The function Ψ is actually a utility function of both human preferences and optimality of solutions. The Rank(x) is a rank assigned to a solution x according to its level of non-dominance within the set X (see equation 3.6) sorted based on F (see equation 3.7), following a procedure of non-dominance sorting (e.g., Deb et. al., 2002). H_m is the m-th CC weight representing the DMs' preferences towards the CC (see section 3.3.2.4 for explanations). It is noted that the function f, should be selected such that it increases with decreasing Rank(x) and increasing H(x).

3.3.1.3 Objective-subjective fronts

Interactivity should, in an IC-MOP, direct the search according to the DM preferences. The interactivity suggested in this section is based on the preferences of a DM towards the concepts'

SCs (see section 3.1). The interactivity promotes the survival of some CCs while hindering others based on the articulation of the DMs' preferences. The front, which is obtained using such interactivity, may include sections of the CBF of the non-interactive problem, or none at all! This constitutes a major difference from the common MOP, and moreover from a C-MOP that has been dealt with in section 3.2.

The solutions resulting from the interactive concept-based search belongs to the 'objective-subjective set' P^*_{os} and its representation in the objective space is termed the 'Objective-Subjective Front' (OSF), FP^*_{os} , which are both defined as follows:

$$\begin{aligned} P^*_{OS} &:= \{x^*_m \in P^*_{uc} \subseteq X \mid \neg\exists\, x_i^{'} \in X : \Psi(x_i^{'}) \le \Psi(x^*_m),\ m \in \{1,\dots,n_c\}\ \} \\ FP^*_{os} &:= \{y^* \in Y \mid y^* = F^m(x^*_m) \quad : x^*_m \in P^*_{OS}\} \end{aligned} \tag{3.19}$$

where P^*_{uc} has been defined in section 3.2.1.4. In equation 3.19 the non-dominance sign, which has been used till now to define fronts, is substituted by an inequality sign. This is due to the fact that the optimality does not depend solely on the non-dominance of the non-interactive problem, but also on the scalar function of equation 3.18, which is a result of the DMs preferences.

Alternatively the objective-subjective set and the OSF may be defined as follows:

$$\begin{aligned} P^*_{OS} &:= \{x^*_m \in X_m \subseteq X \mid \neg\exists\, x_i^{'} \in X : \Psi(x_i^{'}) \le \Psi(x^*_m),\ m \in \{1,\dots,n_c\}\ \} \\ PF^*_{os} &:= \{y^* \in Y \mid y^* = F^m(x^*_m) \quad : x^*_m \in X_m\} \end{aligned} \tag{3.20}$$

The difference between the definitions is the replacement of P^*_{uc} in equation 3.19 by X_m as seen in equation 3.20. These are related to two approaches to attain the OSF as discussed in the following section.

3.3.1.4 OSF Development

According to the definition of the OSF in equation 3.19, the set P^*_{uc} has to be initially found. Finding P^*_{uc} is possible by sequentially developing all the CCs' Pareto fronts. Thereafter the CCs' weights are incorporated. On the other hand equation 3.20 defines the same front, with no demand for a predevelopment of the CCs' fronts. Therefore, a direct approach to obtain the solution to the IC-MOP is to simultaneously sort solutions without creating the fronts of the CCs sequentially. The simultaneous approach aims at reaching the front defined in 3.20 without reaching first those that are defined in equation 3.10. A major advantage of the simultaneous approach is that it applies a search pressure towards the OSF, avoiding the need to initially develop all CCs' fronts. It is noted that a comparison between the simultaneous and the sequential approaches is avoided with respect to this section on IC-MOP (in contrast to what is done in the section on C-MOP) and the focus here is just on the presentation of the approaches. It should be noted, however, that one of the measures, which is introduced in section 3.3.3.3, is aimed at demonstrating that the resulting solutions actually belong to P^*_{uc} .

To elucidate the idea of the OSF, a hand calculation example is used. Refer to figure 3.8a, where three CCs' fronts are depicted in a bi-objective space. These are a result of the non-interactive problem. The CCs are designated by different symbols (circle, triangle, and square for CC_1, CC_2, and CC_3 respectively).

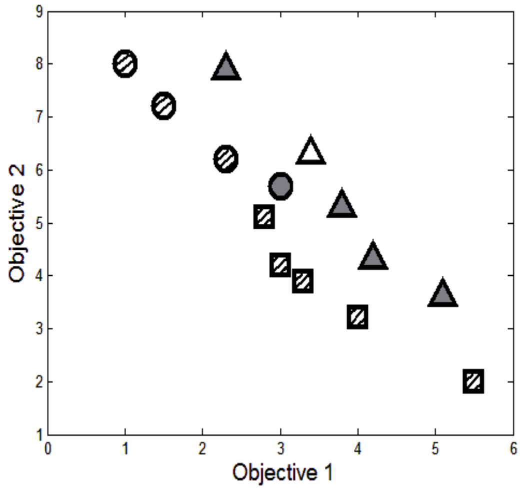


Figure 3.8a: Three ranks of CCs' solutions

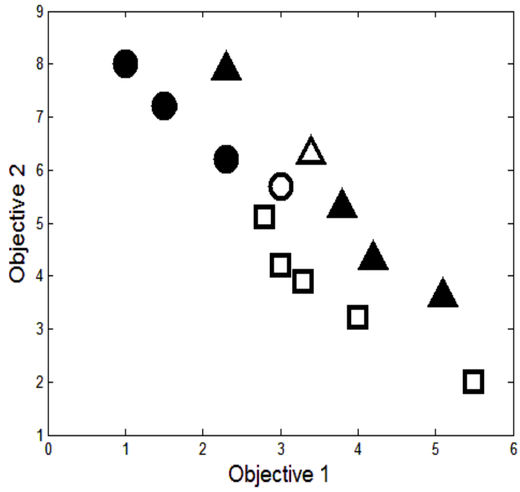


Figure 3.8b: First rank after Ψ computation

In figure 3.8a the first rank is designated by diagonal lines and the performances of solutions that belong to the second rank are designated by gray filling of the symbols, and in the third rank, there is just one solution (designated by a blank triangle.) Let the CC-weights (see section 3.3.2.4) assigned by the designers be: H_1=0.5, H_2=1, H_3=0.2 for the three CCs respectively. Now suppose that, for the sake of this explanation, the function Ψ of equation 3.18, is given by $\Psi = \frac{10}{\text{rank}} \cdot H_m$. Using the weights and the ranks (as depicted from figure 3.8a) the values for Ψ are:

Ψ for the circles of the 1^{st} rank is : 5

Ψ for the circle of the 2^{nd} rank is : 2.5

Ψ for the squares of the 1^{st} rank is : 2

Ψ for the triangles of the 2nd rank is : 5

Ψ for the triangle of the 3rd rank is : 3.33

According to equation 3.19, the OSF is the one that is associated with the maximal values of the function Ψ. In this example it is associated with the circles of the first rank as well as with the triangles of the 2nd rank, as they both have a value of 5 (which is the maximal value). The OSF is depicted in figure 3.8b designated by filled black symbols. It is noted that the obtained OSF holds solutions that are optimal based on equation 3.19, nevertheless some of them are non-optimal, based on equations 3.9. In fact an OSF may overlap a CBF, may contain a part of it or be totally different from it. Figure 3.9a depicts CCs' fronts designated by black, gray and dashed curves. For these fronts, figure 3.9b depicts the CBF (designated by a bold black curve). Figures 3.9c, d show two possible OSFs designated by bold black curves, whereas the CCs' Pareto parts, which are not a part of the OSFs are designated by gray.

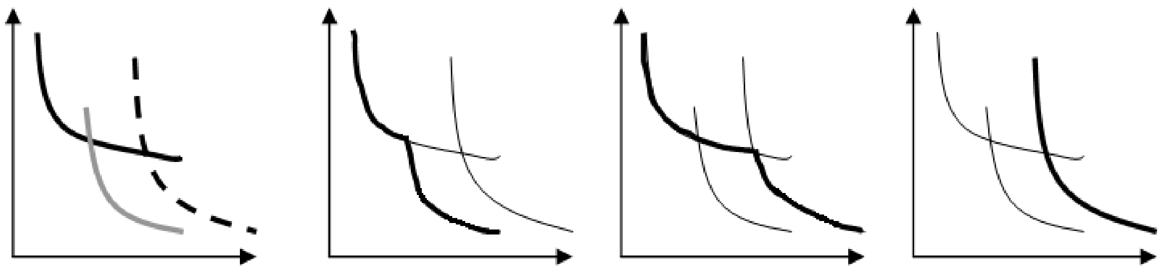


Figure 3.9: Different concept related fronts

a: CCs' fronts b: CBF (also a possible OSF) c: Possible OSF case 1 d: Possible OSF case 2

It is noted that during a simultaneous evolution, issues of resource sharing between CCs and among the CCs' solutions has to be considered and the procedure of finding Ψ is not as straightforward as in the above example, as further explained in 3.3.3.2.

3.3.2 Human interactivity – preferring SCs and concepts

Human-machine interaction has been acknowledged to be important for product development (see literature survey in chapter 2.4.2). Interaction between humans and computers during conceptual design has also been surveyed (see sections 2.3.1, 2.4.2). In this section some new interactivities, within the conceptual design stage, are presented.

3.3.2.1 Articulation of preferences towards SCs

In many problems the availability of models allows the computation of merits, which might only partially reflect all issues that are involved in selecting a concept. For example, difficulties in realizing solutions, associated with a particular concept, might not be modeled (e.g., difficulties of manufacturing the design solutions, un-modeled hazards associated with contracts, employees and so on). Further difficulties to model a concept are associated with the early stages of the design, were DMs rely on their cognitive 'models' of the concepts. At these stages, human experience and intuition has a major influence on the development of concepts.

In this thesis the human attitude towards SCs may support or hinder the development of CCs. The facilitation of these attitudes allows the development of concepts that are not just influenced by shear computations but also by human preferences. This is explained below.

3.3.2.2 Hierarchy of SCs and human preferences

The human cognitive process, which concerns conceptual design, commonly involves decisions in a hierarchical fashion. The DMs' attitude towards the SCs influences the search process. This is subjected to the hierarchical nature of the decisions. Referring to the design space representation of section 3.1, preferences of SCs are not to be considered whenever preferences exist at ancestors' nodes. For example the DMs might strongly reject using non-uniform cross-section area for a robotic arm (see figure 3.2). This means that all preferences towards the alternatives under node 'G' of figure 3.2 are not to be accounted for as they become irrelevant. It is emphasized that a SC may be related to more than one CC. Therefore preferring such a SC is equivalent to a preference towards more than one CC. This multi-CCs' preference by a SC's preference is accounted for by the proposed algorithm (see section 3.3.3), that enhances an accelerated (or decayed) development of several CCs that are related to the preferred (or opposed) SC.

3.3.2.3 Weighting DMs' preferences

In this thesis the attitude towards the SCs is facilitated through weights. DMs' preferences towards SCs are inherently fuzzy. In the early stage of design, team members may be quite uncertain about their weighting. This is not just because they are unconfident if the SC is good or not, but also due to the difficulty to relate their view to precise weights. Moreover, assuming each DM has a strong personal opinion about the discussed SCs, the team is likely to have a considerable variability of opinions. It is assumed here that a discussion may take place leading to a resolution of the differences of opinion. In other words, precise weights of the SCs are determined. The team of DMs may assign weights to some SCs of the problem, with values in the interval [-1, 1], where -1 designates pure dislike, and 1 stands for highest preference. SCs, with no preference, are automatically assigned with zero weights. The weights assigned to the SCs are accumulated to a CC-weight. Alternatively the CC-weight may be assigned directly to the CC. In the context of an evolutionary search the weights affect the development of the CCs fronts by changing the probability of their reproduction.

3.3.2.4 Articulation of preferences – the CC-weight

Taking into account all SCs' preferences, as related to a CC, results in the CC-weight, which represents the CC preference. A suggested procedure of attaining the CC-weight is explained in the following. The DMs may assign weights to some SCs of the problem, with values in the interval [-1, 1], where -1 designates pure dislike, and 1 stands for highest preference. For each 'AND' tree, the nodes, which represent weighted SCs, are assigned with the respective weights. Branches, below any of the nodes, which represent SCs with assigned preferences, are pruned. The SCs of the pruned tree, which have no preference, are automatically assigned with zero weights. The weights, of the resulting pruned tree, are used to obtain the m-th CC-weight, H_m, representing the CC preference. Starting from the leaves of the pruned tree, the weight, w(pr), of each parent node, is calculated by averaging the weights of its children, w(ch). The weight of a parent node is:

$$w(pr) = \frac{1}{n_L}\sum_{n=1}^{n_L} w(ch) \tag{3.21}$$

$$\text{and } H_m = w(root)$$

where n_L is the number of the node's children. The calculation of the weight of the m-th CC, $H_m = w(root)$, is obtained by calculating the weights of the ancestors up to the root node of the 'AND' tree of the m-th CC. Finally, each j-th individual of the m-th CC is assigned with $H_j = H_m$. The following example illustrates the procedure. Figure 3.10a depicts a CC tree, with some of its nodes assigned with weights (the numbers by the nodes). Figure 3.10b shows the pruned tree. The weight of the CC is obtained as follows: $H_m = ((0.6 + 0.0 + 0.0)/3 +0.3)/2 = 0.25$.

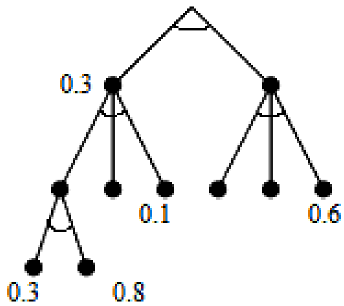


Figure 3.10a: CC tree

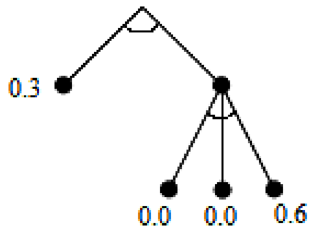


Figure 3.10b: The pruned tree

It is noted that the DMs may assign directly the CC-weight by assigning it to the root node of the CC tree. The CC-weight is used to evolve the objective-subjective front (see section 3.3.3.2).

3.3.3 Interactive concept-based MOEA

The preferences of the DMs towards SCs, which results in CCs-weights (see section 3.3.2.4), should influence the resulting front in accordance with the IC-MOP definition (equations 3.17-3.18). The MOEA that is introduced in the following enhances the development of the OSF, which is a result of both the solutions performances as well as their related CCs' preferences.

Commonly in EC (not IEC), the fitness is directly influenced by the performances of the designs within the objective space. The fitness, which is based on model-based performances, is hereby termed Machine-Based Fitness. This fitness is associated with the rank to which the individual belongs, but it is also influenced by the nature of the EC search, which involves differences in the crowding among the individuals. Therefore it is different from the rank as was the case in section 3.1.2. A fitness that is influenced just by the preferences of humans, which is common in IEC approaches, is hereby termed Human Based Fitness (HBF). This fitness is set to be the concept preference weight H_m (see section 3.3.2). The fitness, which results from considering both influences, is hereby termed Human Machine Fitness (HMF),

$$HMF = f\,(MBF, HBF) \tag{3.22}$$

The HMF is a version of Ψ (see equation 3.18) where MBF is associated with the rank and the crowding of solutions, while and HBF is the influence of the human preferences on the fitness. Figure 3.11 provides a schematic description of the interactive concept-based MOEA. The process begins with the initialization of a population, with representatives for each of the concepts. The genetic codes of these individuals are decoded to produce particular solutions which are related to the CCs. The performances of these solutions are assessed by computations to produce their MBF based on rank and crowding. The human (DM) articulate his preferences through the HBF. Such an articulation of preferences is practiced at the beginning of an evolutionary run and the preferences stay constant as until the run stops and the program shows the front.. A change of the preference initializes a new run. The MBF and the HBF are combined to compute the HMF. The solutions' HMFs are used for the evolutionary cycle, which includes reproduction cross-over and mutation.

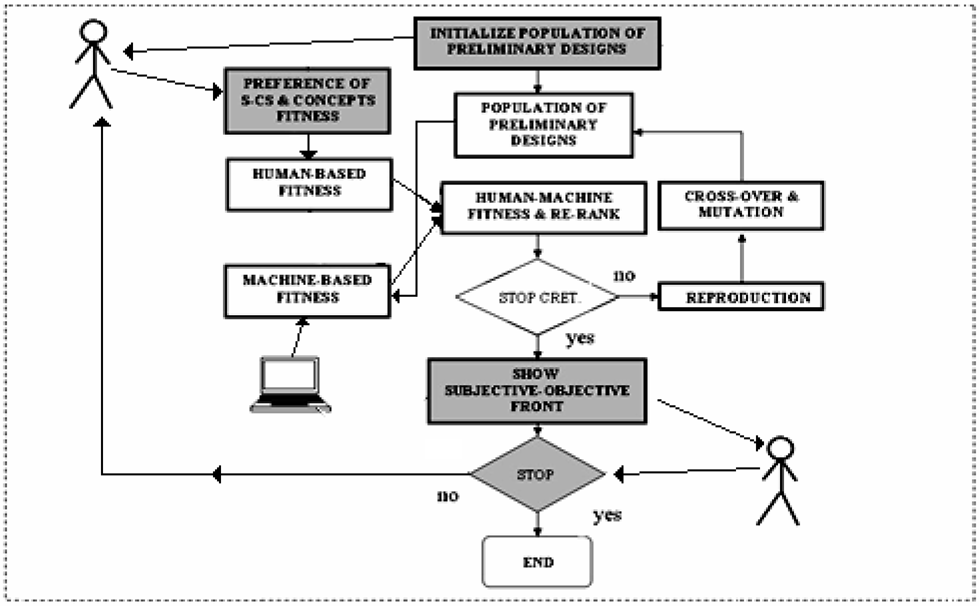


Figure 3.11: Interactive concept-based approach

The un-darkened blocks of figure 3.11 constitute an inner loop, which represents the evolutionary run. When a stopping criterion is reached the objective-subjective front is presented to the DMs. A discussion of the results may lead to reassigning new preferences, followed by a reexamination of their effect on the front. The darkened blocks, in figure 3.11, belong to the outer loop, which activates a new evolutionary run with new preferences.

3.3.3.1 MOEA requirements

In designing the algorithm for the search of an OSF, the following requirements are considered:

1. Maintaining a pressure towards the 'objective-subjective' solutions based on the problem definition in equation 3.17
2. Maintaining a pressure towards the representation of all CCs on the OSF
3. Ensuring diversity of solutions within each CC on the front
4. Maintaining a transverse search pressure to ensure a balanced representation of CCs on the OSF.

These requirements are addressed by the proposed algorithm. Requirement 1 is treated by using non-dominance sorting and assigning optimal solutions with lower ranks (lower level of dominance). Thereafter the CC weights are incorporated to calculate Ψ . Requirements 2, 3 and 4 are addressed in the same manner as in the C_1-NSGA-II (see section 3.2.2.2). In the following section a presentation of the IC-NSGA-II is given, accompanied by a detailed explanation.

3.3.3.2 MOEA algorithm for the evolution of OSFs

A presentation of the IC-NSGA-II is given, followed by a detailed explanation.

IC-NSGA-II

a. Initialize populations P_t with n_c equal sub-populations, each per CC, and set the population size as n= $|P_t|$. Also, create $Q_t = P_t$

b. Combine parent and offspring populations and create $R_t = P_t \cup Q_t$.

c. Decode all individuals to obtain a population of solutions X_t and compute their performances using their CC-related objective functions, $X_t \rightarrow Y_t$.

d. Perform a non-dominated sorting to Y_t and find fronts Fr_i , i=1, 2,…, n_r.

e. Perform the ***'Interactive Concept-based Crowding Sort'*** procedure (*see below).*

f. Compute each of the individual's HMF by using the ***'HMF procedure'*** *(see bellow)*

g. Initialize a new parent population $P_{t+1} = \varnothing$. Sort the solutions according to their HMFs $I_{HMF} = \text{sort}(HMF_j, >)$.

While $n - |P_{t+1}| > 0$

1. Include the first n solutions of I_{HMF}, in the new parent population: $P_{t+1} = P_{t+1} + sol_j^{HMF}$, where $sol_j^{HMF} \in I_{HMF}$

h. Create offspring population Q_{t+1}^{*} from P_{t+1} by an '***Interactive Concept-based Tournament Selection,'*** (*see below*).

i. Perform 'In-concept Crossover' (as outlined in step 'h' of C_1-NSGA-II) to obtain Q_{t+1}^{**}

j. Perform 'Two Mutation Regimes' (as outlined in step 'i' of C_1-NSGA-II) to obtain Q_{t+1} .

k. Go to b.

The following provides an explanation to the algorithm steps.

Step 'a' to 'd': these steps are similar to those of the C_1-NSGA-II.

Step 'e': in this step the Interactive concept-based crowding is implemented as explained in the following:

Interactive Concept-based Crowding Sort

1. Set, the number of solutions in the i-th front, Fr_i that belong to the m-th concept as $n_m^i = \left|Fr_i^m\right|$. For each j-th individual, of the m-th surviving sub-population and objective k, initially assign $CD_{j,k}^m = 0$
2. For each of the n_r fronts compute an upper level and lower level of fitness according to the ***'limits assignment procedure'*** :

Limits assignment procedure

Computed, the following upper (U) and lower (L) bounds for each rank according to:

$$fit_U^i = n_r(1+\varepsilon), \quad \text{for } i=1 \quad \varepsilon \ll 1 \tag{3.23}$$

$$fit_U^{i+1} = fit_U^i - 1 - \varepsilon, \quad \text{for } i=1,\ldots,n_r-1$$

$$fit_L^i = fit_U^{i+1} - 1, \quad \text{for } i=1,\ldots,n_r \tag{3.24}$$

where ε is a constant that separates between adjacent ranks. As a result, each rank has an available fitness span of 1. Figure 3.12, depicts the ranked based fitness assignment and the notions of equations 3.23, 3.24. The available span is reserved for distributing the fitness of the individuals, of the rank, according to their Interactive concept-based crowding (see bellow)

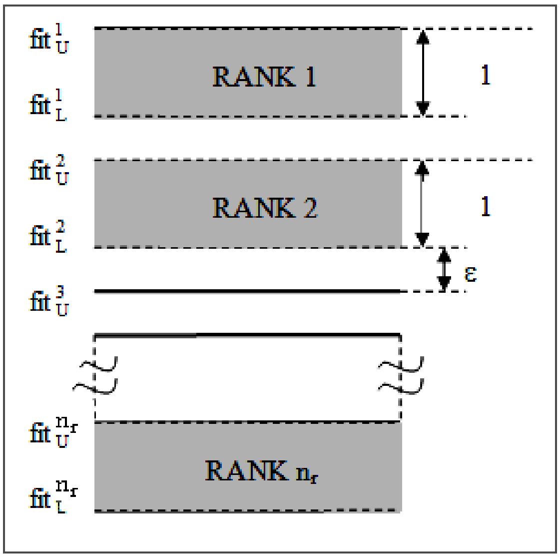


Figure 3.12: The non-dominance ranking

The procedure continues as follows:

1. For each objective k=1,2,..K, sort the set of solutions of each surviving concept

(sub-population) m=1,2…$m_{s,i}$, of the i-th front, by $f_k^{i,m}$: $I_k^{i,m} = sort(f_k^m, >)$.

2. Find the Concept-based Crowding Distance as follows:

   For each objective k, in each m of each front, Fr_i, assign the upper fitness value of the front to the boundary solutions of all concepts in the front, $CD_1^{i,m} = CD_{n_m^i}^{i,m} = 1$, and for all other solutions, j=2 to ($n_m^i - 1$), compute distance, $CD_j^{i,m}$ by: using equation 3.12 and 3.13.

Step 'f': in this step the human machine fitness is computed according to the following procedure:

<u>HMF procedure</u>

The MBF (Machine based Fitness) of all individuals of each rank is calculated. The MBF of the j-th individual of the i-th non-dominance level belonging to the m-th CC is calculated as follows:

$$MBF_{j,i}^m = fit_L^i + \frac{CD_j^{i,m} - CD_{min,i}}{fit_U^i - fit_L^i} \qquad (3.25)$$

where $CD_{min,i}$ is the minimal crowding distance over all $CD_j^{i,m}$ (for all m and j of the i-th rank) in the i-th rank. According to equation 3.25, the most crowded solution will be assigned with an MBF, which is the lowest fitness of the rank. On the other hand the most un-crowded solutions (boundary solutions of the m-th CC in the i-th rank) are assigned with an MBF, which at its limit is equal to the maximal fitness of the rank. Equation 3.25, transforms the sorted list (I_k^m) into a set of fitness values that represent both the level of non-dominance and the crowding of the individuals. The boundaries of the concepts representatives within a front are assigned with the upper limit of the fitness of the rank while the other individuals are assigned fitness between the lower and the upper limits of the rank's fitness according to their crowding distance. Next, the weights of all CCs are utilized to compute the HMFs for all individuals. The HMF of the j-th individual of the i-th non-dominance level belonging to the m-th CC is calculated by

$$HMF_{j,i}^m = \qquad (3.26)$$

$$\begin{array}{ll} MBF_{j,i}^m \cdot (H_m + 1) & \text{for } -1 \le H_m \le 0 \\ MBF_{j,i}^m + (MBF_{max} - MBF^{max,\ m}) \cdot (H_m) & \text{for } \ 0 < H_m \le 1 \end{array}$$

where MBF_{max} is the maximal machine fitness over all individuals within the generation, and

$MBF^{max,m}$ is the maximal fitness of an individual belonging to the m-th CC of the generation. Thus, the fitness of an individual is scaled according to the team preferences. The implementation of equation 3.26 is depicted in figure 3.13, for H_m=1.8. It is noted that equation 3.26 is an example for a possible HMF. This means that other expressions for Ψ may be used. For example, the use of an exponential expression instead of a linear one may be considered.

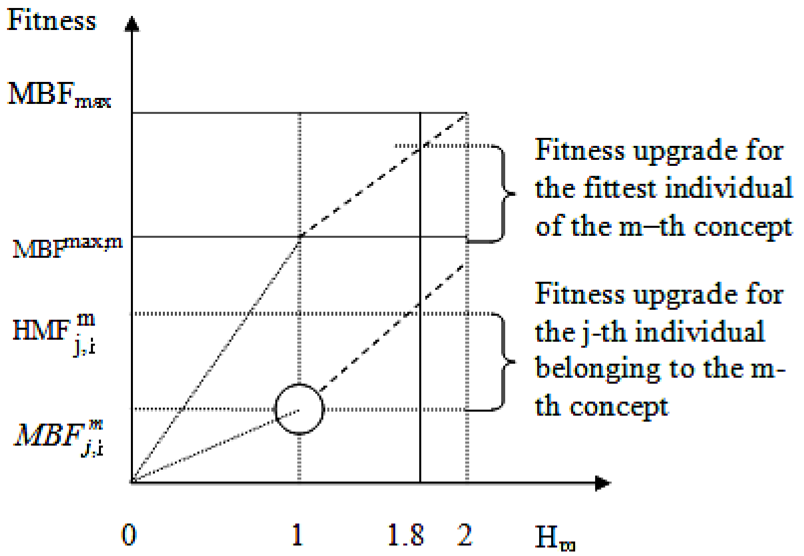


Figure 3.13: HMF assignment based on MBF and HPM

In the example of figure 3.13 the fitness of the solutions related to the m-th CC is up-scaled. The up-scaling is determined based on the best individual of the concept solutions. It is noted that both unchanged as well as degraded fitness may result from the proposed transformation (depending on H_m). It is further noted that when such shifts of fitness, which are associated with parts of the population, occur, a shuffling of the ranks of figure 3.12 may take place. This means that an individual belonging to the second rank may have, after an HMF related up-scaling, a fitness that is within the boundaries of the first rank. The modified set, belonging to the first rank, is introduced to the DMs as the OSF.

Step 'g': In this step the elitist solutions are preserved. In the interactive concept-based algorithm an elite solution is not necessarily associated with a low non-dominance level. This is due to a possible shift of the levels of non-dominance as a result of the interactivity, which is implemented by equation 3.26. Therefore the elite solutions are those with the higher HMFs, which are not necessarily the solutions with low level of non-dominance. This establishes a variant to the traditional approach to EMO.

Step 'h': In this step a tournament selection is implemented using the '***Interactive Concept-based Tournament Selection,***' as follows:

Interactive Concept-based Tournament Selection

Choose randomly two solutions from P_{t+1} and compare them by tournament as follows:

a. If their fitness is different the solution with the higher fitness is the winner and is placed in Q_{t+1}^{*}.

Step 'i' to 'k': These steps are similar to the steps in C_1-NSGA-II (see section 3.2.2.2)

3.3.3.3 Assessing the validity of the resulting OSF

The assessment of the resulting OSF has to be considered based on the requirements from the algorithm. The resulting front might not be a CBF and thus the measures introduced in section 3.2.2.1, are not applicable to the current general case. Therefore an adequate set of measures are introduced below to allow the assessment of the OSF validity. The measures are:

1. Optimality Measure – All solutions of the OSF should be a part of the fronts union FU (see section 3.3.1.3). For each CC which has representatives on the OSF, a non-dominance test is conducted between these representatives and the CC front, P_m^* . A measure of optimality, no_m, is the number of solutions of the m-th CC that belong to the OSF that are dominated by P_m^*, averaged over a fixed number of runs,
2. Repeatability Measures – Because the OSF results from a utility function of both human preferences and optimality of solutions its repeatability is investigated. This is done by finding:
   a. The average number of solutions, n_{OS}^m, of each concept on the OSF over predefined number of runs, and the standard deviation of this number, SDn_{OS}^m.
   b. The Euclidian distance between the most distant representatives of each CC on the OSF, D_{OS}^m and its standard deviation, SDD_{OS}^m, over predefined number of runs.

The first measure guarantees the fulfillment of the optimality demand, while the others are associated with the validity.

3.4 Assessing concepts in MOPs

MOPs are commonly associated with an inherent uncertainty of the DMs' preferences towards the relative importance of the objectives. In such a case no exact preferences are assigned and a decision on a preferred solution is taken posterior to the search. This is usually done by inspecting the performances of the resulting Pareto-optimal solutions. Avoiding a pronounced preference to objectives may also be associated with uncertainty of market demands (e.g., Avigad *et al.*, 2005c). As discussed in section 3.1, in this thesis a CC is associated with a set of solutions and therefore with a cluster of performances in the objective space. In contrast to a particular solution, a CC is commonly associated with a sub-space of the objective space and hence may fit a variety of preferences. Therefore a decision on a CC should take into account not just the performances associated with the CCs but also their coverage of the objective space, or more precisely a pre-defined region of it. For this purpose a Window Of Interest (WOI) is assumed to be chosen by the DMs, which is a subspace, $\mathrm{WOI} \subseteq \mathrm{Y}$, of the objective space. It limits the search to a limited region of the objective space based on the DMs' interest and therefore defines a region where satisfying solutions are sought. Solutions, located outside of this subspace are of no interest to the DMs. It is noted that the term WOI, which has been originally used in Avigad et al., (2005c), possess a similarity to the 'region of interest' introduced independently by Mattson and Messac (2005). The difference is that the WOI is a bounded region whereas the 'region of interest' is not. In this section, the WOI is adjusted to allow the assessment of the CCs performances. The upper boundaries of the WOI are assigned by the DMs (the same as in the case of the 'region of interest'). The lower bounds are determined either by the rationality of the problem objectives (it is impossible to aim at a negative cost or fuel consumption), or automatically set for all k = 1,…K as: $\min(\mathrm{WOI}_k) = \min(\mathrm{FU}_k)$ where $\min(\mathrm{FU}_k)$, is the minimal value of the fronts union in the k-th objective.

In this thesis both the variability and the values of the CCs' performances within a WOI are used to assess the 'goodness' of a CC. Here, a high 'goodness' is associated with both the optimality of the CCs' solutions, as well as their ability to cover the WOI. In the following, an approach to assess the performances of CCs, with respect to a WOI, is presented. It presents the problem of selecting a CC as an auxiliary MOP (section 3.4.2).

3.4.1 Optimality vs. variability

The importance of both optimality and variability of concepts in MOPs is reflected in the review given in section 2.3. It is possible to consider measures for either optimality or variability, and also to consider measures, such as the 'goodness' measure of Mattson and Messac, (2005), which merge both issues. Here the problematic nature of the latter type of measure is demonstrated by the use of two examples. The examples, 1 and 2, are depicted in figures 3.14a and 3.14b respectively.

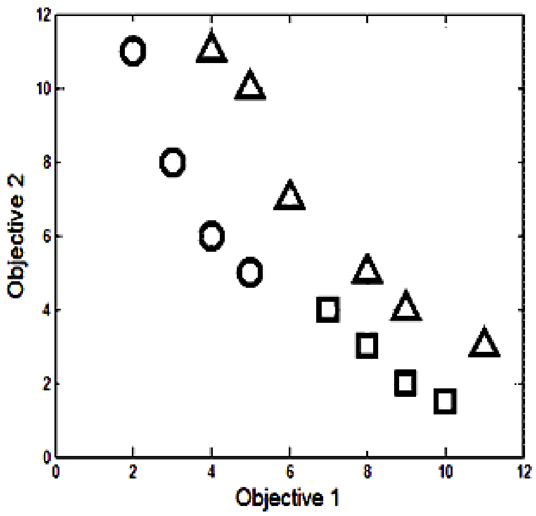


Figure 3.14a: Example 1

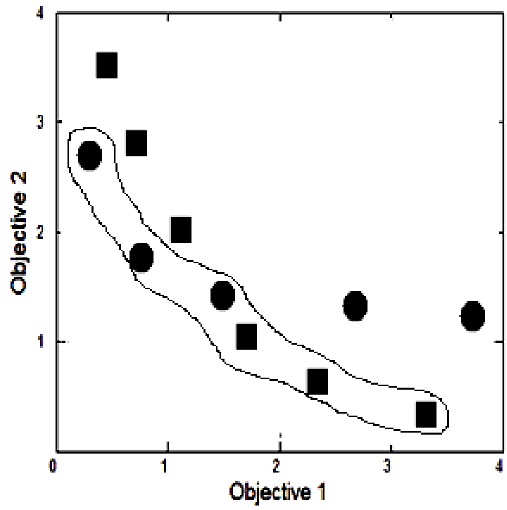


Figure 3.14b: Example 2

In figure 3.14a, three CCs' fronts are depicted within a WOI. These are designated by circles (CC_1), squares (CC_2) and triangles (CC_3). Solutions from two CCs constitute the CBF (CC_1 and CC_2). The use of the 'goodness' measure (see section 2.3.1) introduced in Mattson and Messac, (2005), would yield approximately a value of 0.65 for CC_1 and a value of 0.35 for CC_2. Due to the fact that the 'goodness' measure is calculated based on the s-Pareto solutions only, no value is assigned to CC_3. Nonetheless, CC_3 has the highest variability of solutions within the region and therefore should also be considered for selection. In figure 3.14b two 'optimal' CCs representatives are depicted in a bi-objective space. The CCs representatives are designated by different symbols; circles and squares for CC_1 and CC_2 respectively. The solutions' representatives, which constitute the CBF, are encircled. Using the 'goodness' measure of Mattson and Messac, (2005), would result in values of approximately 0.5 for each of the CCs. Based on these values both CCs are equally good. Nonetheless, observing the CC fronts, it can be seen that CC_2 has a higher variability within the WOI as it spans a higher range of the objectives.

Following the problematic observations from both examples a new approach is suggested. In the proposed approach, Optimality & Variability (O&V) are calculated separately for each CC.

The selection of a CC can be posed as an auxiliary max-max MOP in the O&V bi-objective space, in which the m-th CC is represented by a single point (V_m, O_m). For example figure 3.15 shows four such point representations for four CCs designated by a circle, a squares, a triangle, and a plus. The objectives of the auxiliary MOP are to maximize optimality and to maximize the variability. The decision space of the auxiliary problem concerns the selection of the CCs based on their performance in the bi-objective space.

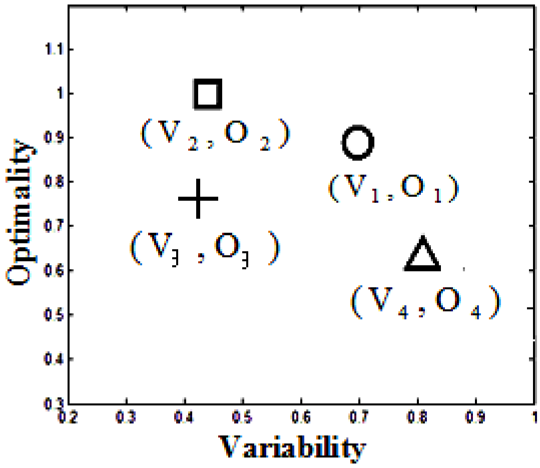


Figure 3.15: Representatives of CCs in the Auxiliary MOP

The optimality of the m-th CC (O_m) is calculated by utilizing the CCs' fronts and implementing the following pseudo code:

<u>Pseudo code for the CCs' optimality measure - O_m</u>

Sort all solutions of P^*_{uc} (see section 3.2.1.4), within the WOI, to obtain each solution non-dominance level, which is also its rank.

For all CCs with no representatives within the WOI assign O_m=0

Initialize a rank counter by setting r_c=1;

While not all CCs are assigned with rank

For each of the un-assigned CCs which have at least one representative within the WOI and with a rank equal to r_c

Assign the m-th CC with $O_m = \frac{1}{r_c}$

Remove all solutions belonging to the assigned CC

$r_c = r_c+1$

End While

The higher the value of optimality O_m the more optimal the m-th CC is.

The variability, V_m of the m-th CC, is calculated by utilizing the CCs fronts and implementing the following pseudo code:

Pseudo code for the CCs' Variability measure - V_m

Divide each of the K objective axes into np divisions, where np is equal to the highest number of representatives any CC possesses on its front within the WOI.

Set to zero a Boolean variable $C_m^{k,z}$ for each of the CCs, with respect to each of the $K \cdot np$ divisions (m=1,.., n_c and z=1,.., np, k=1,…., K).

For all m=1,.., n_c

For all k=1,…., K objectives

For all $x_j \in P_m^*$

For z=1,…, np

If $\{\min(WOI_k) + (z-1)np < f_k(x_j) \le \min(f_k^{WOI}) + z \cdot np\} \wedge f_k(x_j) \in WOI$

$C_m^{k,z} = 1$

Compute for each of the m-th CC the variability measure $V_m = \frac{\sum_{k=1}^{K}\sum_{z=1}^{np} C_m^{k,z}}{K \cdot np}$

Here, $\min(f_k^{WOI})$ is the lower bound of the WOI with respect to the k-th objective. The above procedure checks the performances of each solution in the population. It adds a count to a CC for each objective and for the relevant division in which the solution performance resides. The Boolean variable is equal 1 when at least one representative of the CC has a trajectory on the division. It is noted that the suggested procedure is not applicable to an infinite set of representatives. Yet, given that EMO is an inherently discrete approach, this limitation is not significant here. It is further noted that CCs, which their entire front is outside the WOI, are assigned by the above procedures by 0 for both measures.

The O&V measure, which has been introduced in this section, might be used not just for comparing between the CCs' fronts that are obtained separately. They can be directly used to compare CCs, which are a part of a CBF or an OSF.This may be achieved by changing the algorithm of the variability from using $x_j \in P_m^*$ to $x_j \in P_C^*$ and $x_j \in P_{OS}^*$ respectively. The O&V is adapted in the following section, to support decision making under uncertainties which are associated with delayed decisions.

The final selection of a CC, could be based on its representation in the auxiliary MOP objective space, is left for the DMs. It is suggested that the choice of a CC will be made out of CCs, which possess representatives on the Pareto front of the auxiliary MOP. Referring to figure 3.15 the CC that should be considered for selection are CC_1, CC_2, and CC_4 as they are located on the auxiliary MOP front.

3.5 Supporting conceptual decisions under delayed decisions

In this section the delayed decision problem is introduced. As a part of the introduction, the relation between the new types of concepts, which are introduced in section 3.1, and the delayed decision problem is clarified (section 3.5.1). This is followed by the problem statement as a MOP. Next, the problem is re-stated (section 3.5.2) as a sequence of sub-problems. A possible approach to solve the re-stated problem is suggested based on the O&V measures (see section 3.4).

3.5.1 Introduction to the delayed decisions problem

The delayed decision problem is a known problem, which has been investigated by others. Here the problem is restated in the context of the conceptual design space representation of this thesis (see section 3.1). To highlight the problem as related to the tree representation of the design space the following example is used. Figure 3.16 depicts an 'AND/OR' tree of a conceptual design space for a one-link manipulator. Node 'a' is a SC 'the link', and node 'b' is a SC 'controller'.

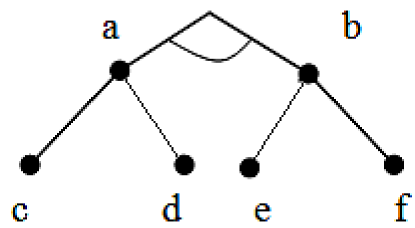


Figure 3.16: Conceptual design space tree

The link (node 'a') is associated with either 'I-shaped link made of aluminum' (node 'c') or 'Square-shaped link made of steel' (node 'd'). The decision on a controller (node 'b') is associated with a decision on either a 'PD controller' (node 'e') or a 'fuzzy controller' (node 'e'). There are four CCs in the space represented by the above tree. An 'I-shaped aluminum link with a PD controller' is an example of such a CC that can be extracted from the tree.

Now suppose that the DMs wish to postpone the decision on the control (the SC of node 'b'). Nevertheless, it is considered essential to continue with the decisions on the link to allow material ordering on time. As a result the search is limited to just two concepts including 'I-shaped aluminum link with unspecified controller,' and 'Square-shaped steel link with unspecified controller.' The pruned trees describing these concepts are depicted in figure 3.17.

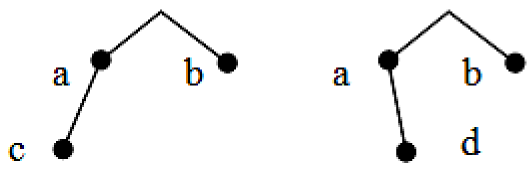


Figure 3.17: HLCs pruned trees

It is noted that both of these concepts are not associated with models as their description is not complete (unspecified controller). As defined in section 3.1.2, such concepts are HLCs (see definition # 4). It is also recalled that the same HLC might be pruned from several different CC trees. In such a case the CCs that are associated with the same pruned tree have been termed associated concepts, ACCs, (see definition # 5 in section 3.1.2) and the HLC has been termed MMC (see definition # 5 in section 3.1.2).

The delayed decision problem calls for a comparison between these MMCs. For the comparison the multi-models can be used. Such a comparison may support the selection of an MMC. In the context of the above example, this will allow the order of either an 'I-shaped aluminum link' or a 'square-shaped steel link,' resolving the delayed decision problem. In using the multi-models for comparing the MMCs, it is assumed that if an MMC is selected, anyone of its ACCs might be later selected. The ACC selection is assumed to be based on uncertain external restrictions, and that this choice can not be controlled. The underlining assumption is that the design has to continue with the given delayed decisions before the external uncertainties are resolved (see the above manipulator example). Therefore, when a decision is delayed, choosing an MMC should ensure that no matter which of the ACCs of the MMC are later chosen, there exists a satisfactory solution from that ACC to be selected by the DMs. A satisfactory solution is a solution with performances within the problem WOI. If more than one MMC is associated with such satisfying solutions a comparison between the MMCs should be conducted to choose the best one. To support MMCs' comparison and selection, the delayed decision problem is restated as a sequence of sub-problems, which are described in the following section.

3.5.2 The delayed decision problem

The delayed decision problem is to select an MMC in a MOP, under the uncertainties of which ACC will be eventually selected. This problem statement is in the context of the design space representation of section 3.1. This problem can be restated as a sequence of sub-problems, which are listed in the following.

a. ***Given*** X (as in section 3.2.1.2) ***find*** P_m^* m=1,..,n_c ***using*** the sequential algorithm (see section 3.2.3).

b. ***Given*** P_m^* m=1,..,n_c ***find*** (V_m, O_m) for m=1,.., n_c in the O&V space by ***using*** the pseudo codes for the CCs' optimality and variability measures (see section 3.4.1).

c. ***Given*** (V_m, O_m) for m=1,.., n_c , ***find*** VO_a^{worst} for all MMC^a a=1,…, n_{MMC} , ***using*** the worst-case sorting procedure, which is introduced in the following.

d. ***Given*** all worst sets, VO_a^{worst} a=1,…, n_{MMC} and all the (V_m, O_m) for CCs that are not ACCs ***present*** these sets in the auxiliary MOP objective space for selection.

e. ***Select*** one of the robust MMCs/CCs ***using*** the presentation from the above sub-problem based on the robustness definition (see definition # 7 below).

Worst-case sorting procedure

a. For each MMC^a a=1,…, n_{MMC} use $VO_m^a = (V_m^a, O_m^a)$ m=1,… n_{ACC}^a to find

$$VO^a, \text{such that } = VO^a = \bigcup_{m=1}^{n_{AC}^a} VO_m^a$$

b. For a= 1,…, n_{MMC} MMCs, sort their ACCs' sets ACC_m^a m= 1,…, n_{ACC}^a, to find each of the MMCs' worst set VO_a^{worst}, in the O&V space by using equation 3.27

$$\begin{gathered}\text{for all } m \in \{1,...,n_{ACC}^a\} \text{ and } m' \in \{1,...,n_{AC}^*\} \text{ and } m \neq m' \\ VO_m^a \in VO_a^{worst} \mid \neg\exists VO_{m'}^a \in VO^a \mid VO_{m'}^a \preceq V_m^a\end{gathered} \tag{3.27}$$

where the domination in equation 3.27 is for finding the min – min front for the m-th MMC. In other words, the auxiliary O&V problem, which involves the maximization of both O&V, is reversed to find the worst case. The worst case has to be considered as it might be the performances of the MMC after the related ACC has been selected. It is noted that in MOPs, 'worst' in the same manner of 'best,' may be associated with a set of performances (front). Using the worst cases representing the MMCs the concepts (MMCs and CCs) should be compared based on their robustness to the delayed decisions uncertainty.

Definition # 7

Robust concept to delayed decisions is:

1. *An MMC for which* $\forall \mathrm{VO}_m^a \in \mathrm{VO}_a^{worst} \neg\exists \mathrm{VO}_m^a = (0,0)$ *where a is the index of the MMC and m is the index for its ACCs,*

2. *A CC for which its* (V, O) $\neq$ (0, 0).

Figure 3.18a, depicts three MMCs, each associated with three ACCs. The MMCs are designated by different symbols: MMC_1 (circles), MMC_2 (squares) and MMC_3 (triangles). The ACCs of the MMCs are designated by black filling, grey filling and blank symbols.

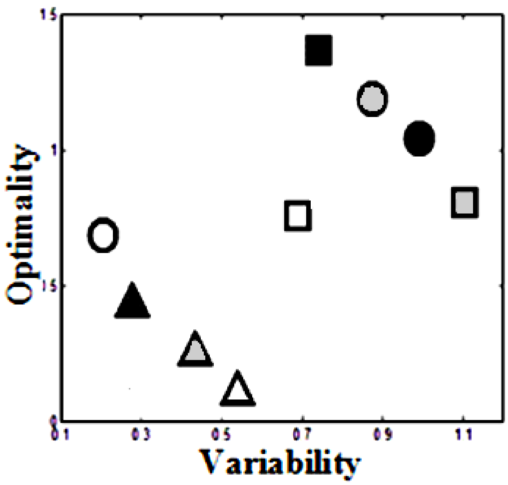


Figure 3.18a: MMCs' ACCs' performances

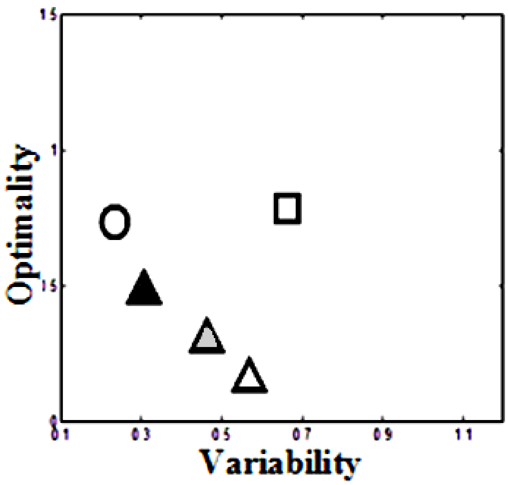


Figure 3.18b: The MMCs representatives

Figure 3.18b, depicts the MMCs representatives sorted according to equation 3.27. The resulting set holds one representative from MMC_1 and one from MMC_2, while MMC_3 has three representatives. In this example, based on definition # 7, all MMCs are robust to the delayed decisions MOP as each holds worst set representatives in the auxiliary MOP, which are not at the origin. Based on these worst cases a decision on an MMC should be taken. Observing the results, which are depicted in figure 3.18b, it may be wise to select MMC_2 as it is associated with the most optimal performances in the worst case within the auxiliary MOP. Nonetheless not all representations in the auxiliary MOP may lead to such a conclusive decision. For example consider the worst sets of four MMCs, which are depicted in figure 3.19.

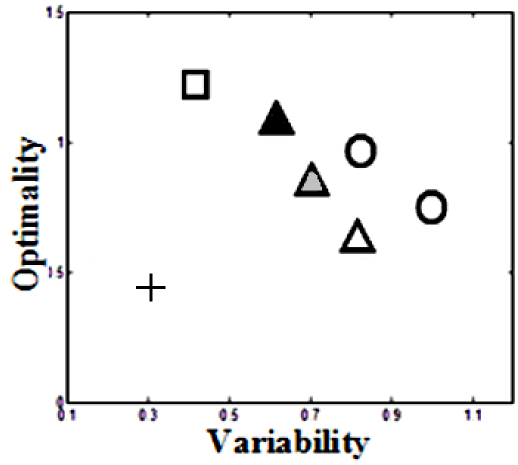


Figure 3.19: Worst sets of four MMCs in the V&O space

Although all MMCs are robust to the delayed decision uncertainty, choosing an MMC out of the circle, triangle and square MMCs, is not trivial, whereas the plus related MMC is clearly not as good as they are. In such a case set-based non-dominance sorting should be considered (left for future work).

CHAPTER 4

CASE STUDIES AND EXAMPLES

In this chapter the methodology presented in chapter 3 is examined through the use of examples. Section 4.1 contains examples for the methodology, which has been presented in section 3.2 on concept-based search and optimization. Section 4.2 contains examples for the methodology, which has been presented in section 3.3 on interactive concept-based search and optimization. Section 4.3 contains examples for the selection methodology, which has been presented in section 3.4 on assessing concepts in MOPs. Section 4.4 contains examples for the methodology, which has been presented in section 3.5 on supporting conceptual decisions under delayed decisions. Section 4.5 contains, a compound example, demonstrating the applicability of the methodology to mechatronic conceptual design.

4.1 Case studies for a C-MOP

The following section is divided into four sections. The first section, 4.1.1, demonstrates the importance of the main features of the C_1-NSGA-II algorithm (e.g., the two mutation regimes and the two expressions of the crowding distance). The second section 4.1.2 provides examples that are used to compare the introduced simultaneous approach with the sequential approach. The comparison employs the concept-based indicators (see section 3.2.2.4). The third section, 4.1.3, compares the computational times of the two introduced algorithms and that of a sequential algorithm. In sections 4.1.1 to 4.1.3 academic examples are used. It is noted that the CCs in the academic examples are not explicit. Only their reflection in the models for the evaluation of the objectives and the search within different search spaces are used. In examples 4.1.1-A, 4.1.2-A -4.1.2-D, only different models of the objectives are used to distinguish between CCs, while the search space is identical. In the other academic examples, both the models and the search space are distinct to each CC. Please refer to section 4.1.4 provides a real-life example on structural design. Finally, section 4.1.5 provides a mechatronic engineering example.

In all of the following test cases an 8 bit binary code is used to code all design parameters. A two point crossover with probability of 50% is used over the entire population. Also employed, over the entire population, are mutation rates of 5% and 1% for the first and second mutation regimes respectively, unless otherwise specified. It is noted that the tuning, of the two regimes mutation probability, has been based on the results obtained with the suggested binary encoding. A different approach may be needed when real encoding is used (see section 5 for future work with respect to the encoding approach). All the results that are presented in the tables of this section are an outcome of averaging 10 runs. In all the examples a min-min bi-objective problem is considered. In examples 4.1.1A-4.1.1.C, 4.1.4 a sub-population size is 25 (n_c=25). In example 4.1.2-D it is 15. In examples 4.1.2A- 4.1.2C, and 4.1.3 n_c=50. It is noted that in all of the examples the same objectives are used for all the CCs, and the apparent differences in the objectives' expressions are due to the different models to compute the performances of solutions for different CCs.

4.1.1 Main features of C_1-NSGA-II

Example 4.1.1-A

This first example demonstrates the ability of the C_1-NSGA-II algorithm to evolve a CBF. The design space is related to solutions from two CCs that differ in their objective functions. The two vector functions, both suggested in Deb 2001, which are related to the two CCs are:

CC_1:
$$\begin{aligned} f_1 &= x \\ f_2 &= 1 + y^2 - x - 0.2\sin(\pi x) \end{aligned} \qquad -2 \le x \le 2 \text{ and } -2 \le y \le 2 .$$

CC_2:
$$\begin{aligned} f_1 &= x \\ f_2 &= 0.75 + y^2 - x - 0.1\sin(3\pi x) \end{aligned} \qquad -2 \le x \le 2 \text{ and } -2 \le y \le 2 .$$

Solving the min-min MOP just for CC_1 results in a convex front, while solving the MOP of CC_1, results in a concave front. The initial population contains representatives from the two sub-populations as depicted in figure 4.1a. The CCs are designated, in both figures 4.1a, and 4.1b by pluses and circles for CC_1 and CC_2 respectively. The result of the simultaneous evolution, using C_1-NSGA-II, is depicted in figure 4.1b. For a reference, the analytical fronts are shown by curves in figure 4.1b. It can be observed that the evolution by the C_1-NSGA-II algorithm has developed a CBF with some distinct parts. The upper part of the front, which is involved with lower values for objective f_1 and higher values for objective f_2, holds solutions from both CCs. The middle part of the front contains just solutions of CC_1. The lower part (bottom-right of the front) holds solutions just from CC_2. It should be noted that, referring to the lower part of the front, even though the front of CC_1 is very close to the front of CC_2, the evolution finds the optimal one (CC_2). Several other aspects of the results are considered in the subsequent examples.

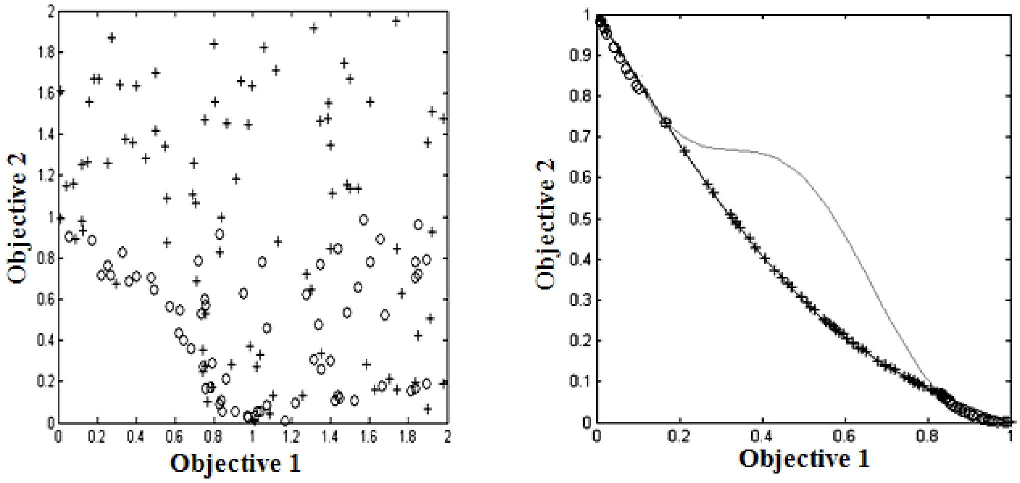


Figure 4.1a: Initial population Figure 4.1b: The resulting CBF
CC_1 is designated by circles and CC_2 is designated by pluses

Example 4.1.1-B

This example demonstrates the importance of the two mutation regimes that are introduced in section 3.2.2.2. Initially the C_1-NSGA-II is used with equal mutation rate of 1% throughout the evolution. In this example the CCs' objective functions are:

$$CC_1: \quad \begin{aligned} f_1 &= x \\ f_2 &= (x-1)^2 \end{aligned} \qquad 0 \le x \le 5,$$

$$CC_2: \quad \begin{aligned} f_1 &= x \\ f_2 &= (x-1)^2 + (y-1)^2 \end{aligned} \qquad 0 \le x \le 5,\ -0 \le y \le 5$$

In the following figures (4.1- 4.7), CC_1 is designated by circles and CC_2 is designated by pluses. The two CCs differ from one another by both the second objective and by the number of decision variables. Figure 4.2 depicts the initial population with 25 representatives from each sub-population. The final front of this example is designated by a curve in the figure. It is noted that the initial sub-population, which is related to CC_1, occupies parts of the final CBF. On the other hand, as seen in figure 4.2, no initial solution of CC_2 is positioned on the front. It is noted that the CBF of this problem contains both CCs. Nevertheless, as depicted in figure 4.3a, no solutions of CC_2 appear on the front at the end of the evolution, which was obtained by using C_1-NSGA-II with just 1% mutation rate throughout the entire run.

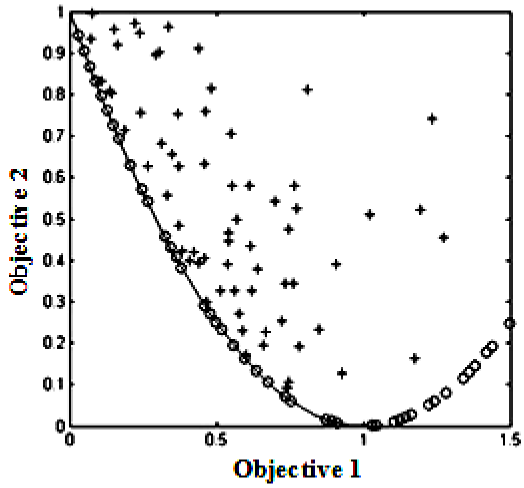
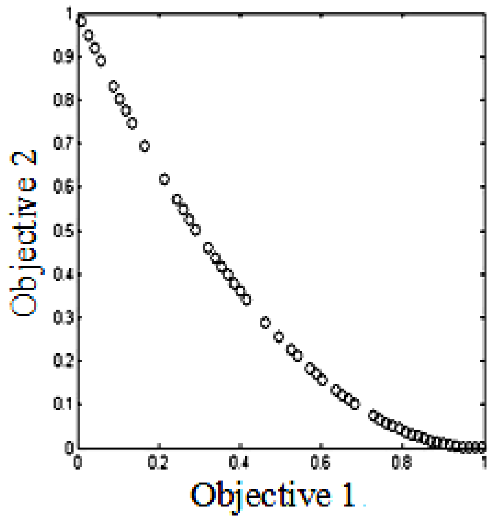


Figure 4.2: Initial population of Example 4.4.1-B

Figure 4.4 depicts the change of the number of solutions on the first level of non-dominance (shown once every 10 generations) as related to each CC. The number of solutions of CC_1 increases rapidly, and dominates the entire population, while the number of solutions of CC_2 stays zero.



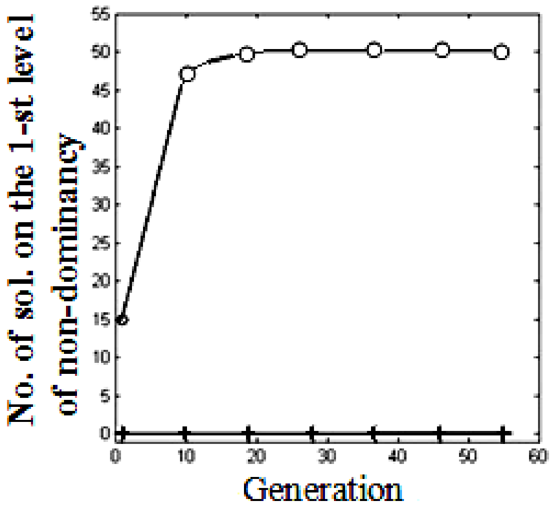


Figure 4.3a: The resulting front (no mutation regimes)

Figure 4.3b: Number of front solutions vs. generation

The reason for the appearance of just one CC on the front has been discussed in section 3.2.2.2. It is noted that the observed problem occasionally happens once in several runs. When the two regimes approach is used, the evolution using C_1-NSGA-II results in the front, which is depicted in figure 4.4a. The front shows a mix of solutions from both CCs (circles and pluses are mingled).

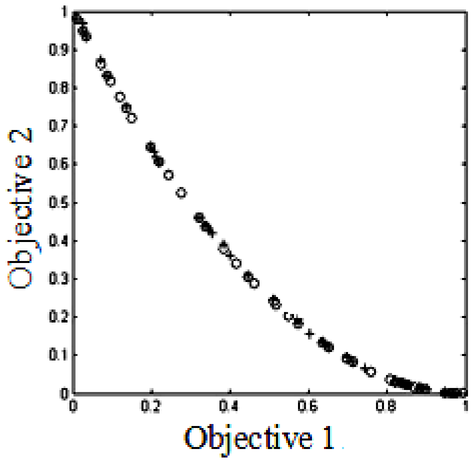


Figure 4.4a: The resulting front (two mutation regimes)

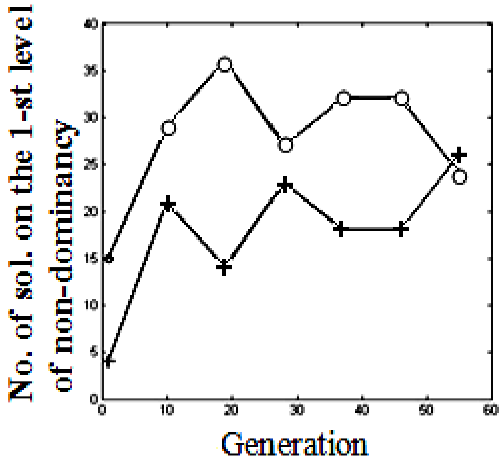


Figure 4.4b: Number of front solutions vs. generation

The front, which results using the two mutation regimes, is occupied by both CCs. Figure 4.4b shows the change in the number of solutions, once every 10 generations, for each CC, on the first level of non-dominance. It is noted that when using the low mutation rate, of 1%, throughout the evolution either the front of figure 4.3a or the front of 4.4a could result. Using the two regimes approach, in all runs that were tested, the resulting front is as depicted in 4.4a.

Example 4.1.1-C

This example demonstrates the importance of each of the expressions of the concept-based crowding (see equation 3.13). The CCs, used here, differ in both the objective functions and in the number of the decision variables. It is noted that the decision variable, x, is mutual to both CCs, but it is searched within different boundaries for each CC. In this example n_c=50.

CC_1: $$f_1 = x \qquad f_2 = (x-1)^2 \qquad 0 \le x \le 5$$

CC_2: $$f_1 = 0.4 + x \qquad f_2 = 0.1 + \frac{y}{x} \qquad 0.01 \le x \le 0.08\,,\ -2 \le y \le 5$$

The C_1-NSGA-II is employed, for the above C-MOP, with three variations of the value of η in equation 3.13. In the first case the calculation of the crowding distance is based solely on the second expression of the equation (setting $\eta = 0$). The resulting front is depicted in figure 4.5a. It is noted that the part of the front where CC_2 (pluses) is the 'optimal concept' is not fully developed. The reason is explained in section 3.2.2.2. It is related to the unwanted disappearance of CC's solutions due to their being niched in a small part of the CBF. The change in the number of the CCs' solutions, which are a part of the evolving front, is depicted in figure 4.5b where the circles and the pluses are designating CC_1 and CC_2 respectively.

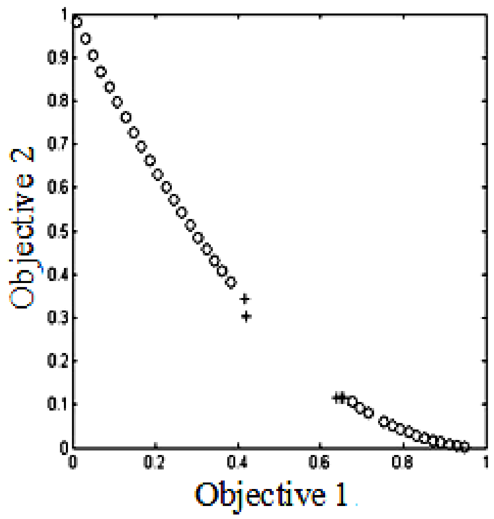


Figure 4.5a: The resulting front $\eta = 0$

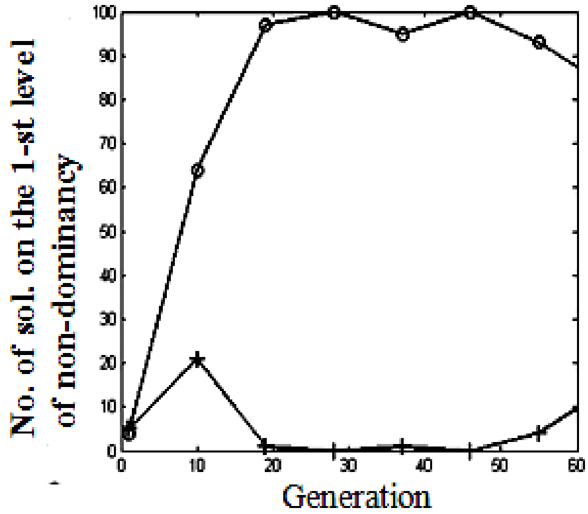


Figure 4.5b: Number of front solutions vs. generation

The second run is executed with $\eta = 1$. This means that the concept-based crowding is influenced just by the first expression of equation 3.13. The resulting front is depicted in figure 4.6a.

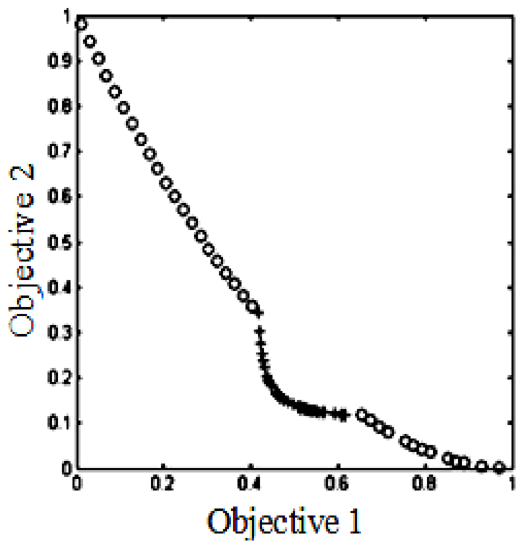


Figure 4.6a: The resulting front with $\eta = 1$

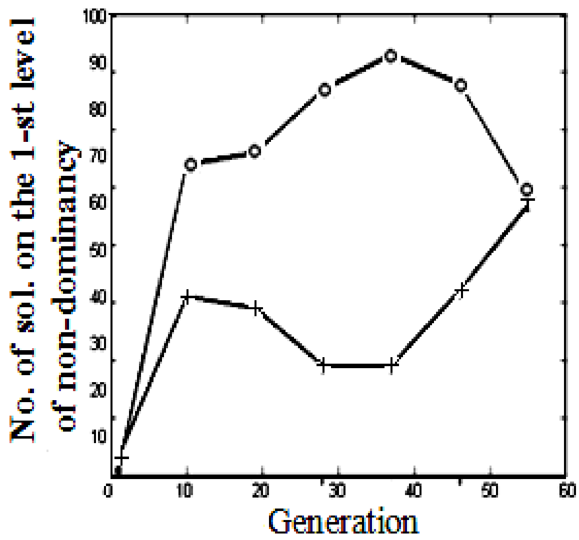


Figure 4.6b: Number of front solutions vs. generation

In this case the entire CBF has evolved but the part that is associated with CC_2 is more crowded than that of CC_1. This is due to the normalization, which is done on the basis of the boundaries within each CC (see equation 3.13). This is also reflected in figure 4.6b, where the number of solutions on the final front is equally split between the sub-populations (note that CC_1 occupies a much smaller section of the front). It is noted that such an equal division is only an example and 40% to 60% is common. The phenomenon, which is demonstrated here, is not desired since that CC_2 has too many representatives when considering its relative part in the entire CBF.

The third run is executed using both expressions of equation 3.13 for the calculation of the concept-base crowding. This means that $\eta = gen / N_{gen}$. In other words η is changing from 0 to 1 as the evolution progresses.

The result of the evolution is depicted in figure 4.7a. It is observed that the density of the front is similar along its parts (in contrast to figure 4.6a).

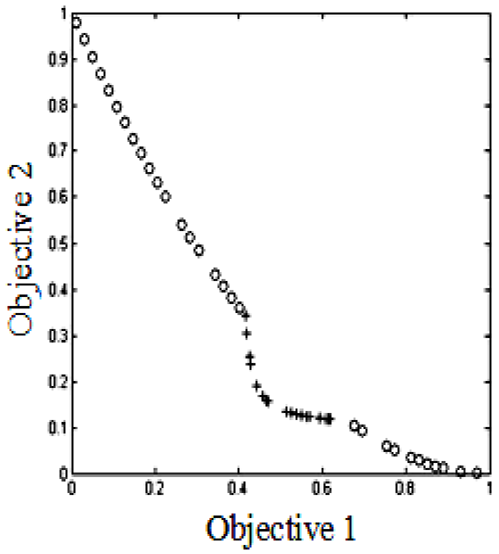


Figure 4.7a: The resulting front with $\eta = gen / N_{gen}$

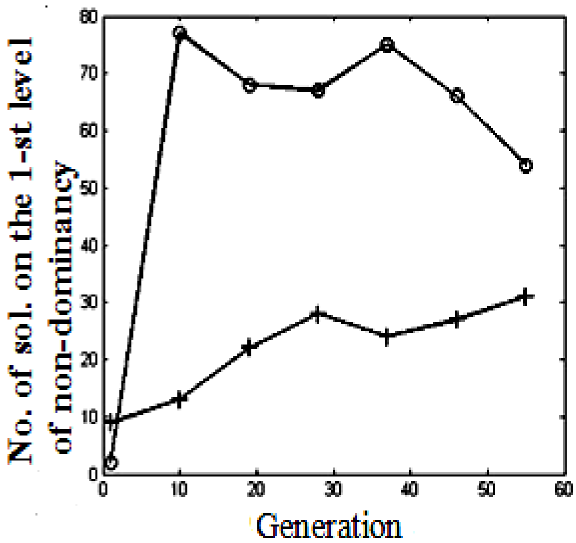


Figure 4.7b: Number of front solutions vs. generation

This is a result of the shift of the normalization of the concept sharing from in-concept boundaries to the entire front boundaries. The change of the distribution of the CCs' solutions on the front is depicted in figure 4.7b. It shows a better division in comparison with that of figure 4.6b.

4.1.2 Simultaneous vs. sequential EMO

In this section a comparison between the simultaneous and the sequential approaches is carried out using C_1-NSGA-II. The comparison is performed using the concept-based indicators (see section 3.2.2.4). In the examples below the total number of individuals summed over all CCs (sequences) of the sequential approach equals the population size used in the simultaneous evolution with C_1-NSGA-II. It is noted that an equal number of generations is used for both approaches. In the current section 4.1.2 two performance indicators are considered, whereas the third indicator is used in the subsequent section 4.1.3.

Example 4.1.2-A

In the first example the CBF involves two CCs such that each CC shares about half of the CBF. The bi-objective problem is used with the following objective functions:

$$CC_1: \quad \begin{aligned} f_1^1 &= x \\ f_2^1 &= 1 + y^2 - x - 0.2\sin(\pi x) \end{aligned} \qquad -2 \le x \le 2, \qquad -2 \le y \le 2$$

$$CC_2: \quad \begin{aligned} f_2^1 &= x \\ f_2^2 &= 0.87 + y^2 - x - 0.1\sin(3\pi x) \end{aligned} \qquad -2 \le x \le 2, \qquad -2 \le y \le 2$$

CC_1 is associated with a convex front while CC_2 is associated with a concave front. Figure 4.8 depicts the analytically obtained fronts of CC_1 by a dashed curve and that of CC_2 by a continues curve.

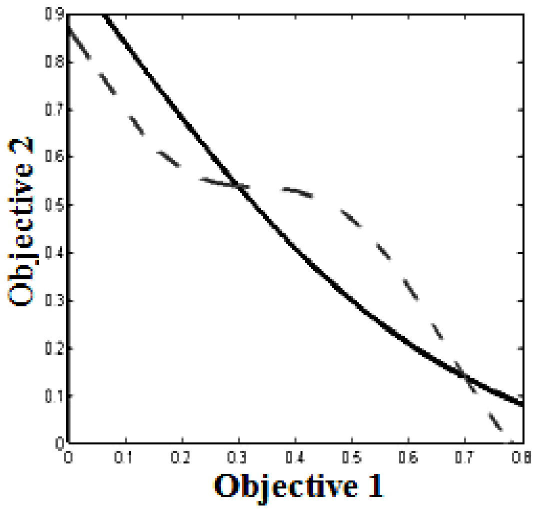


Figure 4.8: Analytically obtained CCs' fronts_for Example 4.1.2-A

The results of running the simultaneous and the sequential algorithms are depicted in figures 4.9a and 4.9b respectively.

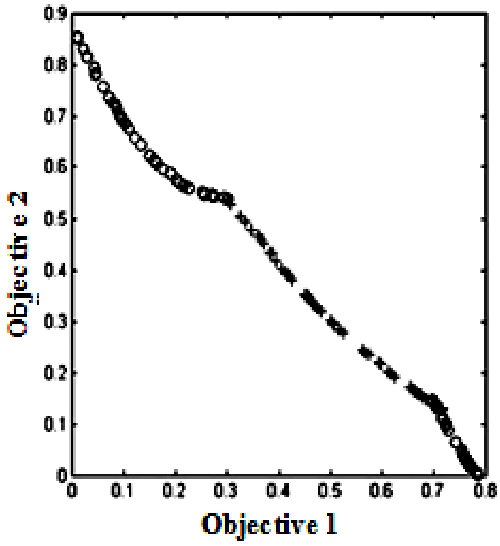


Figure 4.9a: Results of C_1-NSGA-II

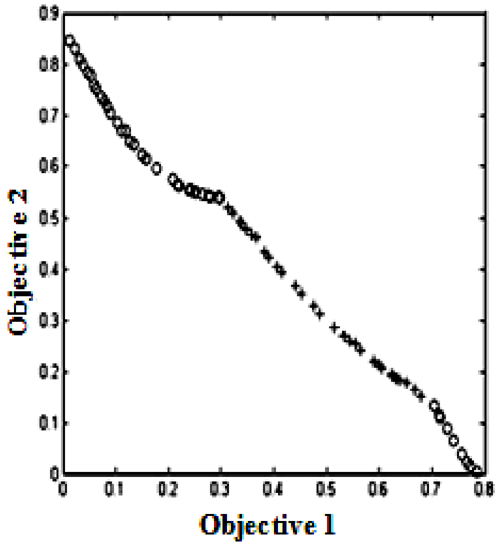


Figure 4.9b: Results of the sequential algorithm

The comparisons between the results of the simultaneous and the sequential approaches, based on the indicators, are depicted in table 4.1.

Table 4.1: Comparing results of example 4.1.2-A

	Simultaneous		Sequential	
Indicator	CC_1	CC_2	CC_1	CC_2
CFO_m	110	90	42	38
D^*_m	0.99	0.997	0.87	0.998

The results that are shown in table 4.1 indicate that for this example:

1. The simultaneous approach introduces much more solutions, from each CC, on the CBF.
2. The distance between the most apart solutions found for CC_1 by the C_1-NSGA-II is greater than the one that is found by the sequential approach. For CC_2 the resulting distances are similar. This is a result of the ability of C_1-NSGA-II to apply a pressure towards the boundaries of each CC on the front and not just towards the front boundaries.

The results obtained in section 4.1.2, demonstrate that the C_1-NSGA-II commonly finds more CC-related solutions on the CBF than those found by the sequential approach. This may be better understood by examining figure 4.10 where the CCs' fronts, which have been evolved separately, are depicted.

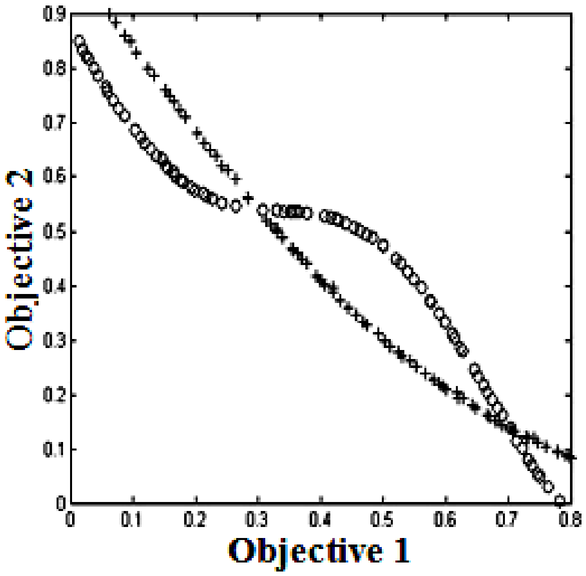


Figure 4.10: Separately evolved concepts' fronts of example 4.1.2-A

Figure 4.10 shows that about half of the initial population of each CC is 'wasted' on solutions that are not located on the CBF. On the other hand, in the simultaneous algorithm the individuals' resource is better utilized to search solutions on the front.

Example 4.1.2-B

In this example the objectives are slightly changed from those of ***4.1.2***-A as follows:

CC_1: $$f_1 = x \qquad f_2 = 1 + y^2 - x - 0.2\sin(\pi x) \qquad -2 \le x \le 2, \quad -2 \le y \le 2$$

CC_2: $$f_1 = x \qquad f_2 = 0.75 + y^2 - x - 0.1\sin(3\pi x) \qquad -2 \le x \le 2, \quad -2 \le y \le 2$$

The employed change of f_2 of CC_2 shifts its front to the right and by that changes the ratio between the relative sizes of the parts of the CCs on the front. Now CC_2 occupies a larger part of the CBF. The results of the C_1-NSGA-II and the sequential algorithm are depicted in figures 4.11a and 4.11b respectively.

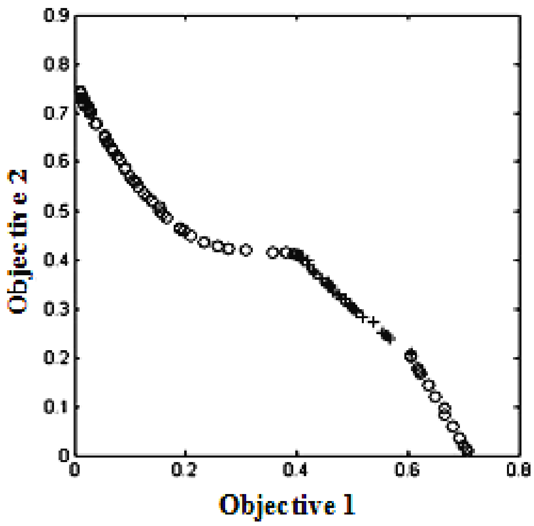


Figure 4.11a: Results of C_1-NSGA-II

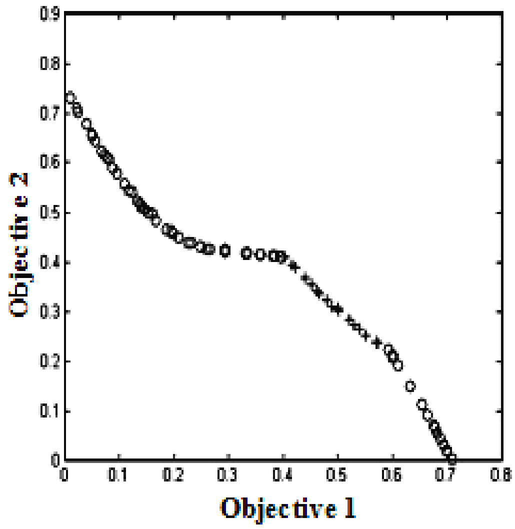


Figure 4.11b: Results of the sequential algorithm

The comparison between the fronts is achieved by the concept-based indicators and is summarized in table 4.2.

Table 4.2 - Comparing results of example 4.1.2-B

	Simultaneous		Sequential	
Indicator	CC_1	CC_2	CC_1	CC_2
CFO_m	130	70	80	38
D^*_m	0.99	0.98	0.84	0.99

The results of this example are consistent with the results of example 4.1.2-A and the advantages of the C_1-NSGA-II are well observed again.

Example 4.1.2-C

In this example one CC occupies the entire CBF, while the other one shares a part which is rather small relatively to the entire front. Here two CCs are used, with the following objective functions:

CC_1: $$f_1 = x \qquad f_2 = 1 + y^2 - x - 0.2\sin(\pi x) \qquad -2 \le x \le 2, \quad -2 \le y \le 2$$

CC_2: $$f_1 = x \qquad f_2 = 0.70 + y^2 - x - 0.1\sin(3\pi x) \qquad -2 \le x \le 2, \quad -2 \le y \le 2$$

The CCs fronts are shown in figure 4.12a. The front of CC_1 is marked by a continues curve, whereas the front of CC_2 with a dashed curve.

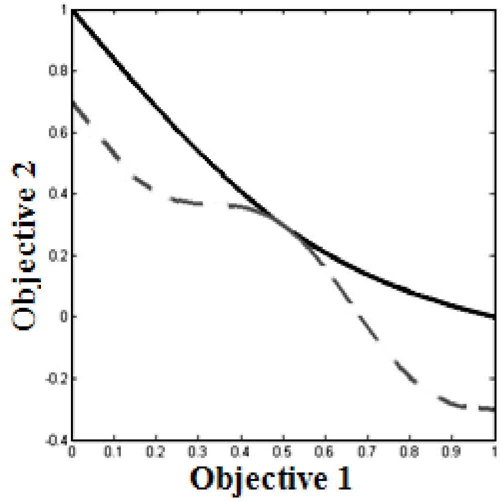


Figure4.12a: The concepts fronts

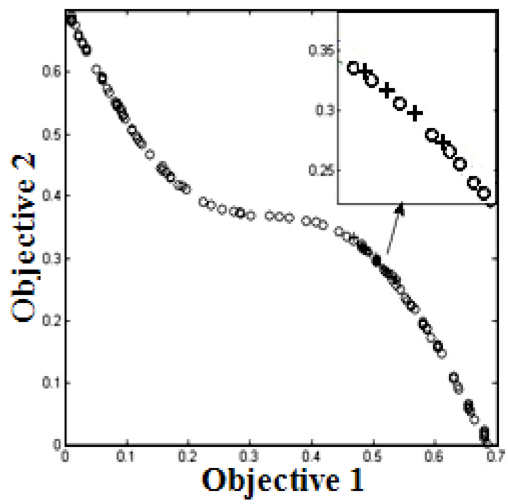


Figure 4.12b: Results of C_1-NSGA-II

Figure 4.12b, depicts the C_1-NSGA-II resulting front, with a zoomed frame designating the region where the intersection takes place. The zooming depicts that the intersecting part of the front is well covered by both CCs. In contrast, when observing the results of the sequential algorithm, as shown in figure 4.13, the intersecting part contains just a few solutions that are related to CC_1 (marked by arrows.) It is noted that once in several runs no representatives of CC_1 appear on the CBF at all! This phenomenon is profound when the size of the populations is decreased.

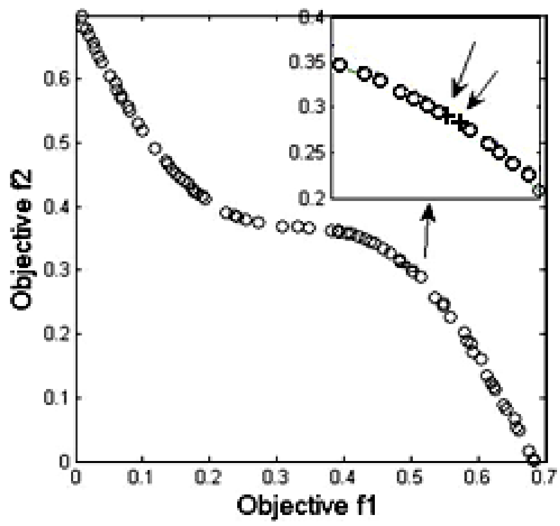


Figure 4.13: The results of the sequential algorithm

The comparison between the fronts is done using the concept-based indicators and is summarized in table 4.3

Table 4.3 - Comparing results of example 4.1.2-C

	Simultaneous		Sequential	
Indicator	CC_1	CC_2	CC_1	CC_2
CFO_m	170	30	96	4
D_m^*	0.90	0.995	0.2	0.996

The results that are shown in table 4.3 indicate the following:

1. The simultaneous approach introduces much more solutions, for each CC, on the CBF. This is extremely evident for CC_1.
2. The distance between the most apart solutions, as found for CC_1 by C_1-NSGA-II, is greater than that found by the sequential approach. It is noted again that there are runs where the sequential approach produces no solutions for CC_1. The superiority of C_1-NSGA-II is evident for such examples. For CC_2 the results are similar.

Example 4.1.2-D

All of the above examples involve two CCs. Here a case of eight CCs is used to further compare the simultaneous vs. the sequential approach. The objectives in the current example are:

$$\begin{aligned} f_1 &= x \\ f_2 &= a + y^2 - x - 0.1\mathrm{fn}(b\pi x) \end{aligned} \qquad -2 \le x \le 2\,;\ -2 \le y \le 2\,.$$

The function 'fn', used here, is either sin or cosine. This function together with the parameters 'a' and 'b,' dictate different models for calculating the performances of the different CCs. The different parameters' values the related concepts and their legends are summarized in table 4.4.

Table 4.4 Summary of CCs for Example 4.1.2-D

Concept	a	b	function	Legend
1	1	1	sin	○
2	1	1	cosine	△
3	0.8	1	sin	□
4	0.8	1	cosine	✱
5	1	3	sin	+
6	0.8	3	cosine	▽
7	1	3	cos	◇
8	0.8	3	cosine	☆

Figure 14a shows a representative part of an initial population. The initial population contains 15 individuals per each CC. The eight CCs are distributed in the objective space according to their performances. Figure 4.14b depicts the resulting CBF, as evolved using C_1-NSGA-II, and the analytical fronts of the 'optimal concepts' (CC_3, CC_6 and CC_8).

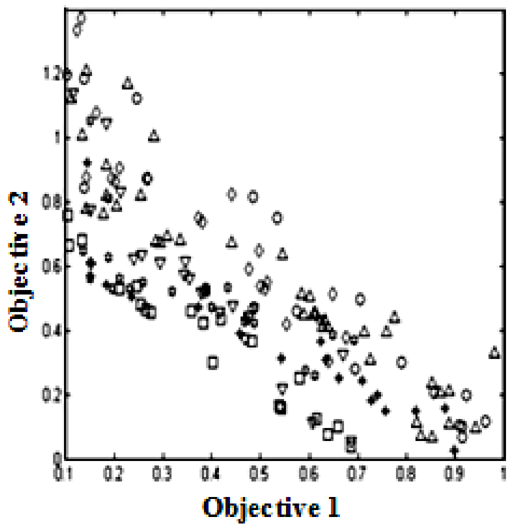


Figure 4.14a: Part of initial population

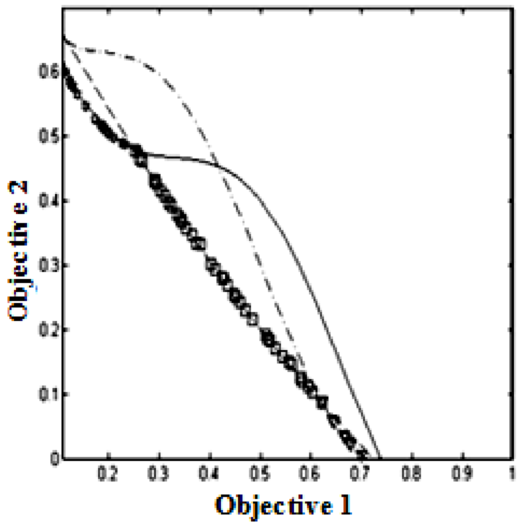


Figure 4.14b: The CBF

Evolving sequentially the eight populations, of 15 individuals each, results in the CBF depicted in figure 4.15.

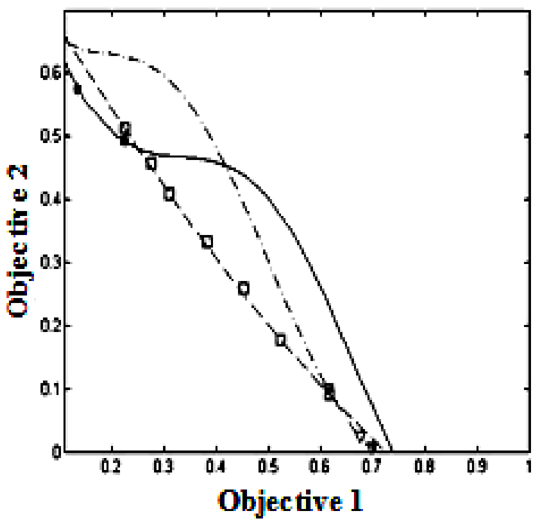


Figure 4.15: Sequential based front

Table 4.5 provides a comparison between the results as obtained by the simultaneous and the sequential approach for the three 'optimal concepts.'

Table 4.5: Simultaneous vs. Sequential Approach for example 4.1.2-D

	Simultaneous			Sequential		
Indicator	CC_3	CC_6	CC_8	CC_3	CC_6	CC_8
CFO_m	65	12	28	7	3	4
D^*_m	0.98	0.96	0.97	0.88	0.92	0.74

The obtained results are in accordance with those obtained in the other examples of this section. It is noted that using the sequential approach the obtained particular solution of CC_3 (squares) with the minimal f_1 value, as shown in figure 4.15, is not optimal. This is due to the lack of particular solutions from CC_8 in that vicinity that would have dominated those of CC_3. The results, which have been obtained by the simultaneous approach, do not involve the above disadvantage. This is a reflection of the better use of the available resources (individuals) in the simultaneous approach.

The above phenomenon of the inclusion of non-optimal solutions in the obtained CBF, which occurs in the sequential approach, is less evident with increasing size of the CCs populations (not shown here). While increasing the size of populations improves the expected resolution and the values of the performance indicators for the concept-based representation, it is associated with a larger computational time.

Example 4.1.2-E

This example demonstrates an evolution of a non-intersecting front and in particular the evolution of a uni-concept front. In this case the bi-objective MOP, (slightly changed from the 'KUR' test problem, see Deb 2001) is to find the CBF for the following objectives:

CC_1

$$f_1 = \sum_{i=1}^{2}\left[-10\exp(-0.2\sqrt{x_i^2 + x_{i+1}}\right] + 20$$
$$f_2 = \sum_{i=1}^{3}\left[|x_i|^{0.8} + 5\sin(x_i^3)\right] + 20 \qquad -5 \le x_i \le 5, \quad i = 1,2,3.$$

CC_2

$$f_1 = 3 + x$$
$$f_2 = 30 + (x - 3.5)^2 \qquad 0 \le x \le 5$$

The CCs differ from one another both by the models of the objective functions and by the number of decision variables. The resulting CBFs achieved by C_1-NSGA-II and by the sequential algorithm are depicted in figures 4.16a and 4.16b respectively. A summary of the results for comparing between the simultaneous and the sequential approach are depicted in table 4.6.

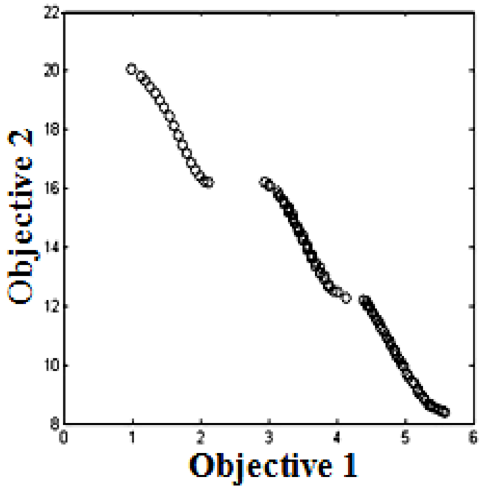


Figure 4.16a: Results of C_1-NSGA-II

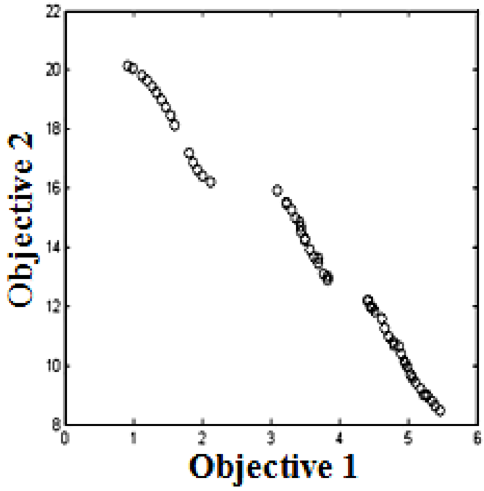


Figure 4.16b: Results of the sequential approach

Table 4.6: Comparing results of example 4.1.2-E

	Simultaneous	Sequential
Indicator	CC_1	CC_2
CFO_m	190	95
D^*_m	0.99	0.93

The results that are shown in table 4.6 indicate the following:

1. The simultaneous approach introduces much more solutions, belonging to the surviving CC on the front.
2. The distance between the most apart solutions found, for the surviving CC, by C_1-NSGA-II is greater than the one found by the sequential approach.

C1-NSGA-II produces more solutions on the front (in case of the uni-concept front it doubles). In addition it has a better chance of finding the edges of a front with discontinuous regions. This can be realized by comparing figure 4.16a with 4.15b.

4.1.3 Comparing computational time

This section provides a comparison of the simultaneous and sequential approaches using the third indicator (computational time). To demonstrate such a comparison the objective functions of each of the CCs are accompanied with a delay function that simulates the assumed computation time needed to conclude the computation of their performances (see section 3.2.2.5).

Four basic states are considered as follows:

1. Both CCs equally share the front.
2. Both CCs share the front but one CC occupies a much larger part.
3. One CC survives on the front and the front of the non-optimal CC is far from the 'optimal concept's front. Here the front of the non-optimal concept is considered 'far' if its sub-population disappears within 5% of the generations for both running by C_1-NSGA-II and C_2-NSGA-II (see remark below).
4. One CC survives on the front and the front of the non-optimal concept is close to the 'optimal concept's front. Here we refer to 'close' if the sub-population of the non-optimal concept disappears after at least 80% of the generations for both running by C_1-NSGA-II and C_2-NSGA-II (see remark below).

Remark - It should be noted that although the last two states (3 and 4) are related to distance in the objective space, a generational-based disappearance measure is used. In the following examples this generational measure correlates well with the location. Nevertheless, such a relation is not always true and care in using the generational measure should be taken.

An example for the four basic states is depicted in figure 4.17a-d.

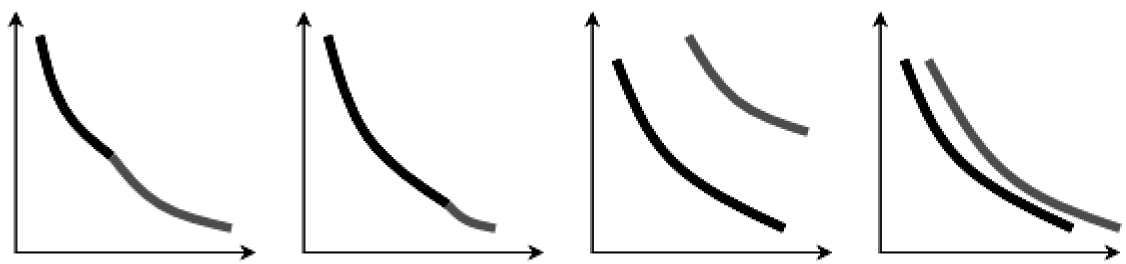


Figure 4.17 a: State 1 b: State 2 c: State 3 d: State 4

These four states are combined with the delay, which is explained above, to introduce the following investigated 7 situations:

A. State 1 with no delay.
B. State 2 with no delay.
C. State 3 with no delay.
D. State 4 with no delay.

E. State 2 with a delay to the CC occupying the larger section of the front.
F. State 2 with a delay to the CC occupying the smaller section of the front.
G. State 3 with a delay to the 'optimal concept'.
H. State 3 with a delay to the non-optimal concept.
I. State 4 with a delay to the 'optimal concept'.
J. State 4 with a delay to the non-optimal concept.

The following results correspond to the four problems, which represent the four states. The problem representing state 1 is the problem of example 4.1.2-A. The problem representing state 2 is the problem of example 4.1.2-A using 0.72 instead of 0.87 in the second objective of CC_2. The problem representing state 3 is problem 4.1.2-E. The problem representing state 4 is the problem of example 4.1.2-A using 0.68 instead of 0.87 in the second objective of CC_2.

The parameters for the evolutionary computations were kept as described in the beginning of section 4.1. Table 4.7 summarizes the results as obtained for the above A-J situations. For each algorithm three columns are given including the time ratio, and the number of the CC representatives for each CC (CC_1 and CC_2) on the obtained front. The later two columns contain the values as obtained by the first performance indicator (see section 3.2.2.4). The use of the second performance indicator, which is not shown here, does not change the conclusions. Fifty individuals were initially used for each sub-population in the simultaneous algorithms, as well as per CC population in the sequential approach.

Table 4.7: Comparing Simultaneous approaches vs. sequential approach

Situation	C_1-NSGA-II			C_2-NSGA-II			Sequential		
	τ	CC_1	CC_2	τ	CC_1	CC_2	τ	CC_1	CC_2
A	2.4	47	48	2.2	46	45	1.0	25	35
B	2.35	85	15	1.4	49	20	1.0	40	10
C	3.8	97	0	1.0	49	0	1.7	49	0
D	2.3	97	0	1.8	47	0	1.0	48	0
E	4.2	85	15	3.6	49	20	1.0	40	10
F	4.0	85	15	1.0	49	20	1.2	40	10
G	4.5	97	0	1.3	48	0	1.0	49	0
H	3.8	97	0	1.0	49	0	4.2	49	0
I	4.0	98	0	1.2	47	0	1.0	47	0
J	4.0	97	0	1.8	47	0	1.0	48	0

The following can be observed from table 4.7.

a. C_2-NSGA-II involves less computational time than C_1-NSGA-II.
b. Comparing the computational time of C_2-NSGA-II vs. the sequential approach reveals that it depends on the situation. There are situations (C, F, and H) where the time of using C_2-NSGA-II is shorter than that of the sequential approach. This is most evident in case 'H'. This is a result of the non-optimality of the CC which has the delay. In this case the sequential approach has a wasting time of producing a non-optimal front.
c. Comparing the computational time of C_1-NSGA-II vs. the sequential approach it is noted that only in situation H, for similar reason as above, the simultaneous approach is shorter. Moreover, in this situation the representation (as indicated in the CC_1 columns) obtained by the simultaneous approach is better, based on the representation performance indicator.
d. When considering the first performance indicator alone, C_1-NSGA-II generally outperforms the other two alternatives. This is in agreement with the results of section 4.1.2.
e. Also, for the first performance indicator alone, C_2-NSGA-II is at least as good as the sequential approach.

Table 4.7 appears to show the generic behavior of the algorithms. This includes the following major observations:

1. C_1-NSGA-II produces better representation performance at the cost of higher computational time.
2. C_2-NSGA-II involves shorter computational time than C_1-NSGA-II at the cost of lower representation performance.
3. The results of comparing the computational time of the sequential approach vs. the simultaneous approaches depend on the nature of the problem.

4.1.4 Structural mechanics example

The following example demonstrates the applicability of the suggested approach to real-life applications. For this purpose the concepts used by Mattson and Messac 2005 are adopted, with a modification of the unit system and loadings. The conceptual design space includes three different CCs that are depicted in figure 4.18.

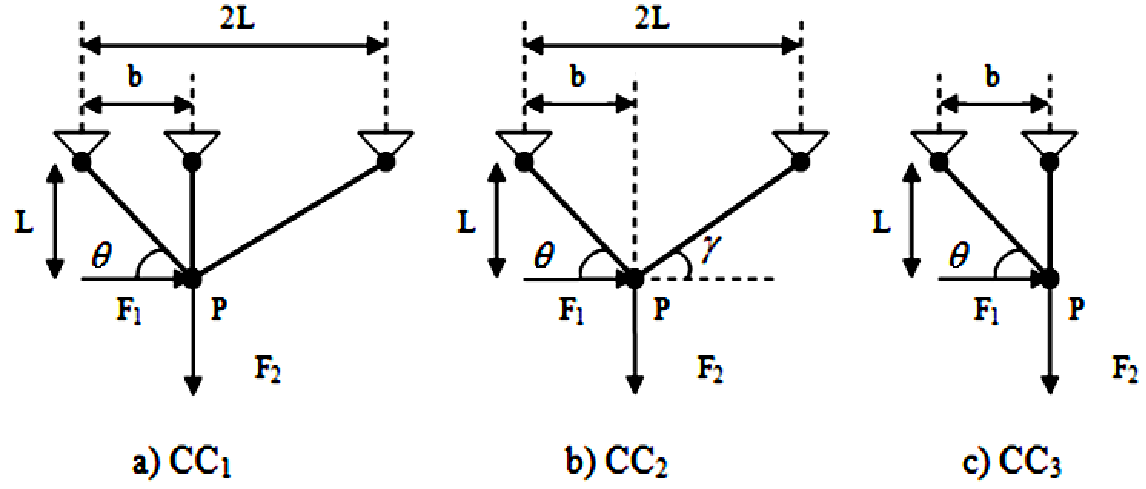


Figure 4.18: Three CCs truss problem

The bi-objective problem involves the minimization of both the total volume of the truss and the deflection of junction P. The design variables of this problem are the cross section of the bars, A_B, and the width variable, b, within the following design constraints: $3\text{cm}^2 \leq A_B \leq 17\text{cm}^2$ and $9\text{m} \leq b \leq 27\text{m}$. The length parameter is kept constant at L= 18m. The design CCs differ by the number of design variables and by a difference in the models for the objective functions. In the following figures CC_1, CC_2 and CC_3 are designated by squares, triangles and circles respectively.

Using C_1-NSGA-II with 20 individuals for each sub-population of the initial population, and the loads F_1=1N and F_2=10N, results in the CBF as depicted in figure 4.19. The resulting CBF contains representatives from CC_1 and CC_3. A good qualitative correlation between the optimal concepts obtained here and those of Mattson and Messac, (2005) is achieved. Yet, it is beyond the scope of this thesis to compare the methods here with theirs, and this is left for future work.

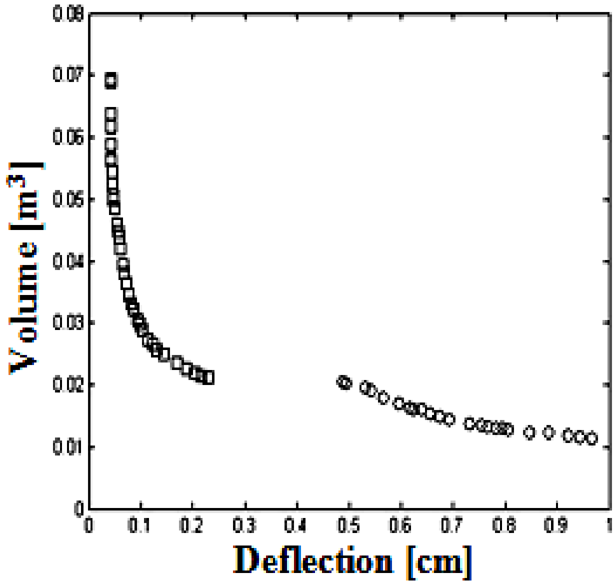


Figure 4.19: CBF with F1=1N, F2=10N

Changing the loads to F_1=10N, F_2=15N results in a new CBF, as depicted in figure 4.20.

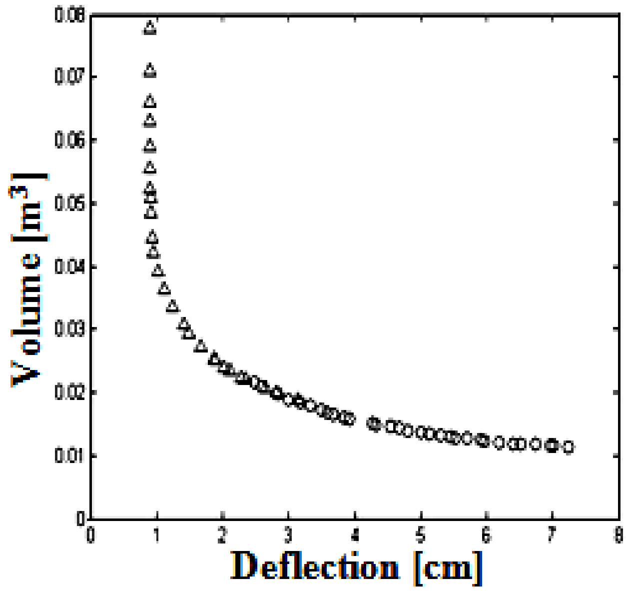


Figure 4.20: CBF with F_1=10N, F_2=15N

The load changing results in a substantial change of the fronts including a change of the front location, the number of 'optimal concept's and the front type (see section 3.2.1.3). Comparing figures 4.19 and 4.20, the latter involves the three rather than two CCs, with an intersection of C_2 and C_3 at the middle part of the front.

4.1.5 Effects of MOEA parameters

In addition to the study on the effect of mutation, which is discussed in section 4.1.2, the following changes were made to assess the sensitivity of the proposed algorithms to such changes:

a. The examples of sections 4.1.2 and 4.1.3 were run with 20 bit encoding instead of 8 bit encoding. Excluding the expected increase of computational time, there was no reportable difference from the results of sections 4.1.2 and 4.1.3 (with 8 bits).

b. In the examples of section 4.1.2A- 4.1.2C the sub-population size was changed from 50 to 25 and also to 75. Results obtained when increasing the sub-populations size to 75 appeared consistent with those of section 4.1.2. When using 25 individuals within a sub-population the results of the simultaneous approach became even more superior with respect to both performances measures. This is in accordance with the results obtained in example 4.1.3.

4.2 Case studies for the IC- MOP

The methodology introduced in sections 3.3.1-3.3.3 on interactivity is investigated here by the use of several academic examples. In section 4.2.1 non-hierarchical problem decomposition is used to demonstrate the working of the interactive concept-based evolution algorithm. In section 4.2.2 a hierarchical 'AND/OR' tree description is utilized to demonstrate the influence of the interactivity towards SCs on the resulting OSF. In section 4.2.3, an engineering problem is used to demonstrate the applicability of the proposed algorithms to engineering conceptual design. In all of the following examples the evolutionary parameters are kept as detailed in section 4.1 unless stated otherwise.

4.2.1 Interactivity with no hierarchies

For this case, with no hierarchies, the weights assigned are the CCs' preferences. Therefore, no weight calculations are needed and the assigned weights are directly employed.

Example 4.2.1-A: Uni-front and interactivity

In this example the concepts' objectives and the parameter search limits are:

$$\begin{aligned} & f_1 = x^2 + b \\ & f_2 (x-2)^2 + b \end{aligned} \qquad -5 \le x \le 5$$

Two SCs are used, one corresponds to b=0 and the other to b=5. This means that there are two CCs, CC_1 and CC_2, differing one from each other by the models of the objective functions. The interactive concept-based MOEA (see section 3.3) uses a population of 20 individuals for each sub-population while other parameters (mutation rate etc.) are kept as in section 4.1. The effect of human preferences towards the CCs is examined in the following. When there are no SCs' preferences, then $H_1 = H_2 = 1$ for both CCs (see section 3.3.2.3). In such a case no upgrading of the MBF occurs. The resultant Pareto front, corresponding to the winning CC, CC_1 (b=0), designated by squares, is shown in figure 4.21. This front is in fact the CBF.

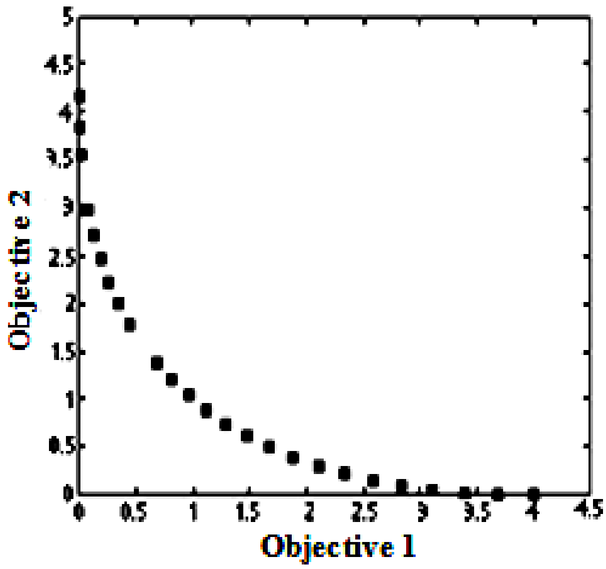
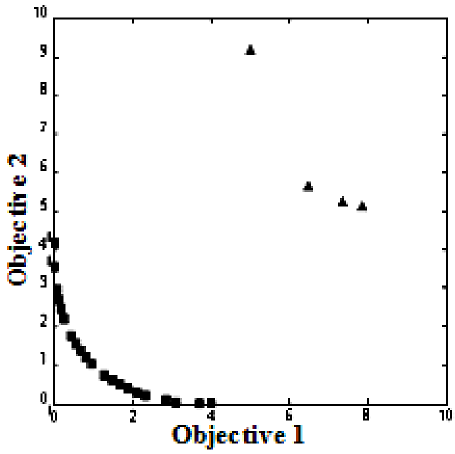


Figure 4.21: CBF of Example 4.2.1-A (no preferences)

Increasing the preference of the disappearing CC, CC_2 (b=5), by assigning it with a preference weight of 0.4, while assigning no preference to CC_1, leads to the OSF depicted in figures 4.22a. It is depicted that along the initial front, there are several representatives from CC_2 designated by triangles. Increasing further the preference weight of CC_2, to 0.7, results in the OSF depicted in figure 4.22b. It can be observed from figures 4.22a and 4.22b that as the preference weight of CC_2 is increased more of its front is reviled. It is noted that as the preference towards CC_2 raise, more of the resource of individuals is transferred towards a search of solutions belonging to CC_2. Assessing the results of figure 4.22a and 4.22b is done here based on the measures, which have been introduced in section 3.3.3.3. The results are summarized in table 4.8a and 4.8b respectively.



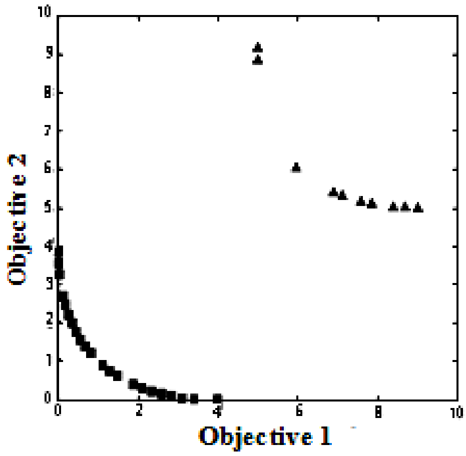


Figure 4.22a: OSF with $W_1 = 0$, $W_2 = 0.4$ Figure 4.22b: OSF with $W_1 = 0$, $W_2 = 0.7$

Table 4.8a: Measures of successes for example 4.2.1-A with $W_1 = 0$, $W_2 = 0.4$

Measure	CC_1	CC_2
no_m	0	0
n^m_{OS}	25	5
SDn^m_{OS}	1.78	0.88
D^m_{OS}	5.62	5.5
SDD^m_{OS}	0.03	0.08

From the table it can be concluded:

1. All the CCs representatives are optimal within each CC.
2. The OSF is consistent and similar fronts result in all runs (as observed from the low values of the two standard deviations measures (SDn^m_{OS}, SDD^m_{OS}).

Table 4.8b: Measures of successes for example 4.2.1-A with $W_1 = 0$, $W_2 = 0.7$

Measure	CC_1	CC_2
no_m	0	0
n^m_{OS}	22	8
SDn^m_{OS}	1.3	0.9
D^m_{OS}	5.62	6.01
SDD^m_{OS}	0.02	0.05

From the table it can be concluded:

1. All the CCs representatives are optimal within each CC.
2. The OSF is consistent and similar fronts result in all runs (as observed from the low values of the two standard deviations measures ($SDn^{m}_{OS}, SDD^{m}_{OS}$).

It is noted that using a weight of 0.35, causes un-decisive results. This means that representatives of CC_2 may or may not survive. Therefore the front may be the same as the CBF or as an OSF. This should not be considered as a disadvantage of the approach as the DM dilemma stated in undeceive preference is reflected in the results.

When the preference towards CC_2 is evident and is expressed by assigning W1 = -0.5, W_2 = 0.9 the resulting concepts weights are: H_1 = -0.5 and H_2 = -1.9. The evolution based on this assignment results in the OSF depicted in figure 4.23 (for reference the CBF is shown by a continuous curve). It is observed that most of the computational resources turned towards the preferred CC, with no CC's solutions on the CBF.

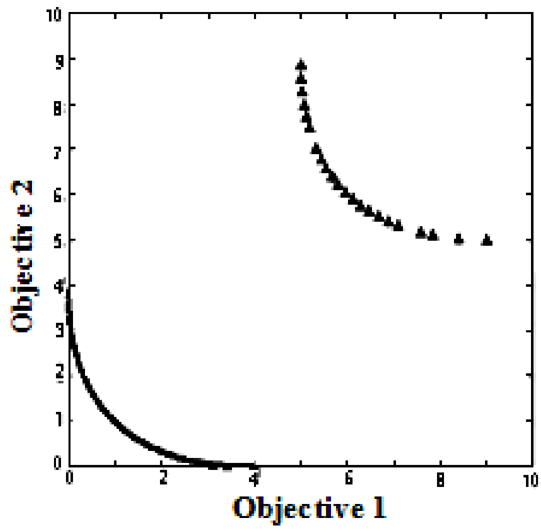


Figure 4.23: OSF with W_1 = -0.5, W_2 = 1.9

Example 4.2.1-B: Intersecting front and interactivity

The objectives, used in this academic example, are:

$$f_1 = x^2 + 5b + 10$$
$$f_2 = (x-4)^2 + c(b-1)x \qquad -10 \le x \le 10;$$

The space consists of four CCs. The CCs are composed of four SCs, which are characterized by: c=-3, c= -4, b=+ 2 and b=+3. Table 4.9 summarizes the CCs and their related SCs as well as the CCs designating legends which are used in the following figures.

Table 4.9: Summary of CCs and their legend for example 4.2.1-B

CC #	c values	b values	Legend
1	- 3	+2	circle
2	- 4	+2	star
3	- 4	+3	rhombus
4	- 3	+3	plus

In the first simulation, which provided the results shown in figure 4.24, there are no human preferences assigned to the SCs. It can be seen that CC_4 (plus) did not survive at all, while the others share the resulting front with intersecting section at the top part of the front. Descending along the front, it contains only one dominating CC (CC_2), and then again, at the lower part, it contains another CC (CC_3).

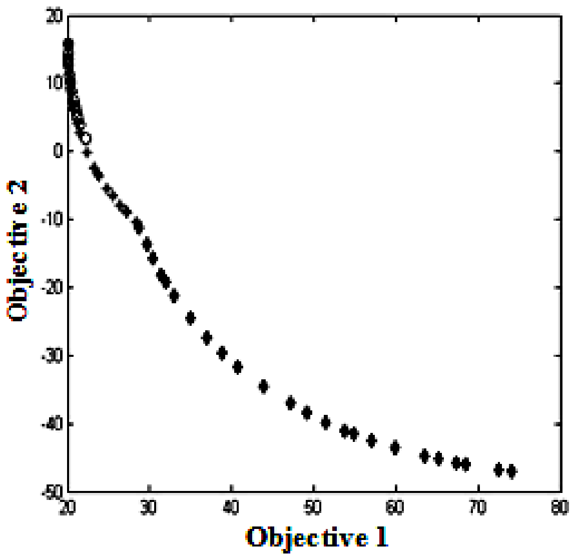


Figure 4.24: The CBF

To demonstrate the effect of human preferences the following situation is examined. The weight of the SC characterized by c= -3 is assigned with a value of 1.0 (the highest preference possible), and the weight for c= -4 is chosen as -0.6. These preferences cause the CCs, associated with c=-3 (CC_1 and CC_4), to be preferred over those associated with c=-4 (CC_2 and CC_3). The results of the evolution are shown in figure 4.25.

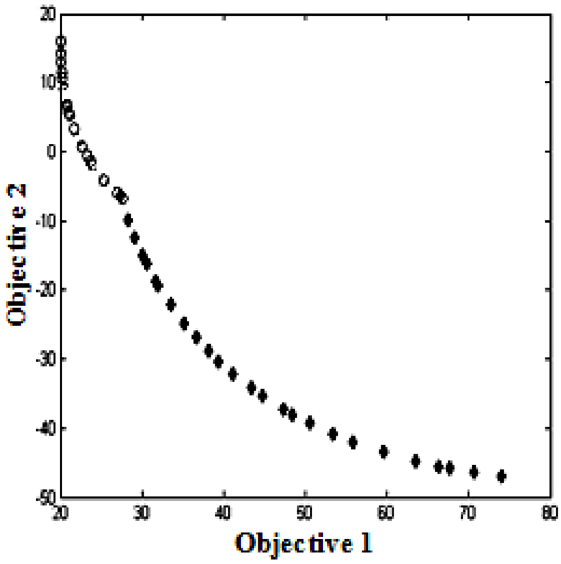


Figure 4.25: The OSF

Comparing figure 4.24 with figure 4.25, it can be seen that the winning CCs are not all the same. This is due to the influence of the subjective preference articulation. Moreover the front is no longer an intersecting front and it is a shared one. Table 4.10 summarizes the numerical values of the performance measures of this example.

Table 4.10: Measures of successes for example 4.2.1-B

Measure	CC_1	CC_2
no_m	0	0
n_{OS}^m	25	13
SDn_{OS}^m	1.1	0.9
D_{OS}^m	61.7	27.3
SDD_{OS}^m	0.06	0.11

The validity of the OSF as calculated by the proposed measures of validity is high as further reflected by the low values of the standard deviation measures. It is also notes that the relation between the D_{OS}^m of both CCs is 2.26. The relation between the n_{OS}^m of the CCs is 1.92. The resemblance between these values implies on the transverse pressure towards a balanced representation of the CCs on the OSF.

4.2.2 The hierarchical case

The purpose of the following bi-objective academic example is to demonstrate the affect of the hierarchical preferences on the resulting front. The bi-objectives minimization problem used here is similar to the one used in example 4.1.2-D.

$$\begin{aligned} f_1 &= x \\ f_2 &= a + y^2 - x - 0.1\mathrm{fn}(b\pi x) \end{aligned} \qquad -2 \le x \le 2 \text{ and} -2 \le y \le 2\,.$$

This time the CCs are extracted out of a hierarchical AND/OR tree representation. Eight SCs associated with the parameters, 'a' and 'b,' and with a trigonometric function, 'fn,' are used to describe the conceptual design space. The SCs, within the hierarchal arrangement of the 'AND/OR' tree, are depicted in figure 4.26. Each of the 8 CCs can be represented by an 'AND' tree that is extracted from the 'AND/OR' tree. Table 4.11 provides a list of CCs and their associated symbols (legend) as used in the following figures.

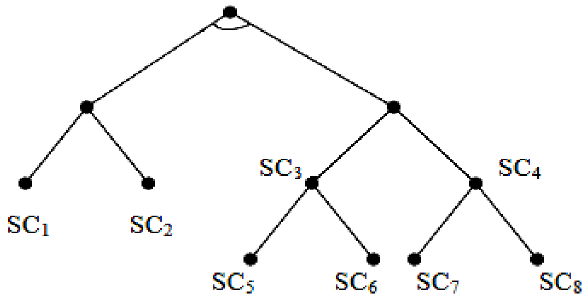


Figure 4.26: The 'AND/OR' tree representation

Table 4.11: The hierarchical case - summary of CCs and their legend

CC	S-Cs	a	b	function	Legend
1	1,3,5	1	1	sin	○
2	1,4,7	1	1	cosine	△
3	2,3,5	0.8	1	sin	□
4	2,4,8	0.8	1	cosine	✳
5	1,3,6	1	3	sin	+
6	2,4,8	0.8	3	cosine	▽
7	1,4,8	1	3	cosine	◇
8	2,3,6	0.8	3	sin	☆

The parameters a, b, and the trigonometric function, sinus or cosine, which are associated with the S-Cs, dictate different models for the objective functions, (as related to equation 5), for the various CCs. It should be noted that SC_1 to SC_8 are related to a=1.0, a=0.8, sin, cosine, b=1, b=3, b=1, b=3 respectively. It is also noted that the equal values of b=1.0 for SC_5 and SC_7 as well as the equal values of b=3.0 for SC_6 and SC_8 mean that these SCs are actually the same, and the different indices is a result of the representation. The decision on the trigonometric function and the decision on the value of the parameter 'a' characterize the highest SCs of the hierarchy and the decision on the value of the parameter 'b' is associated with the SCs of the lower hierarchy. Deciding on the branch at each 'An,' leads to an ‘AND’ tree of SCs, which corresponds to a CC.

Figure 4.27a shows a part of the initial population, which is associated with eight sub-populations of 15 individuals each. The eight CCs are distributed in the objective space according to their performances. Figure 4.27b depicts the resulting CBF, which is the same as in figure 4.14 (re-shown here for ease of reading). The surviving front, including CC_3, CC_6 and CC_8, is the CBF that is achieved by an evolution, which is influenced by the MBF alone.

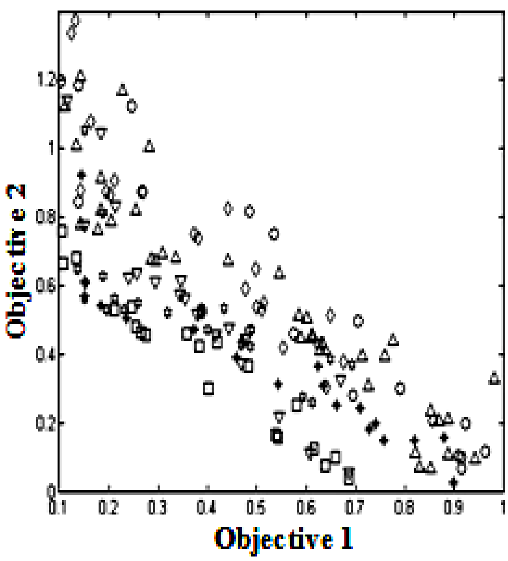


Figure 4.27a: The initial population

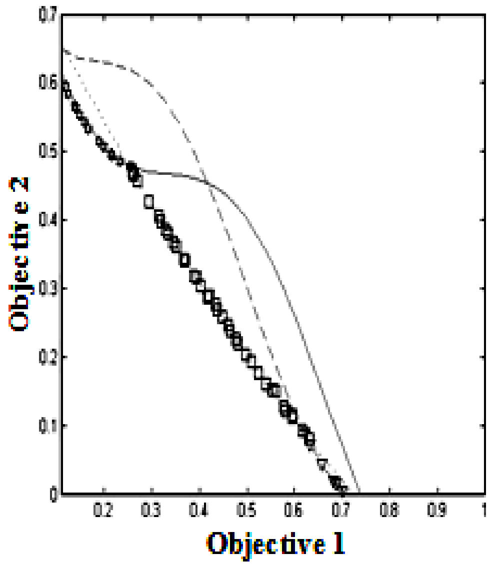


Figure 4.27b: The CBF

It is noted that all three optimal CCs are associated with SC_2 (for which a=0.8), which is associated with the highest level of the hierarchy. Now suppose that a preference weight of 0.6 is assigned to that SC. The resulting OSF is depicted in figure 4.28a. It is seen, that apart from the CBF there is 'another' front. It is the front of CC_4. This added front is not a part of the problem CBF, but it survived due to the preference of a high located SC of the CC. Now suppose that the same SC is assigned with a lower weight of 0.5. The resulting front is depicted in figure 4.28b, where less representatives of CC_4 has survived due to its lower preference. In other words, a reduced search pressure towards CC_4 is taking place. A similar OSF to the one depicted in 4.28b is achieved if SC_2 has no weight assignment and SC_4 and SC_8 are each assigned with a weight of 0.25, which is half the weight that was assigned initially to SC_2. This is due to the lower location of the preferred SCs within the hierarchy. It is noted that the last assignment did not result in a survival of solutions of CC_2 or CC_7 although they are associated with those preferred SCs.

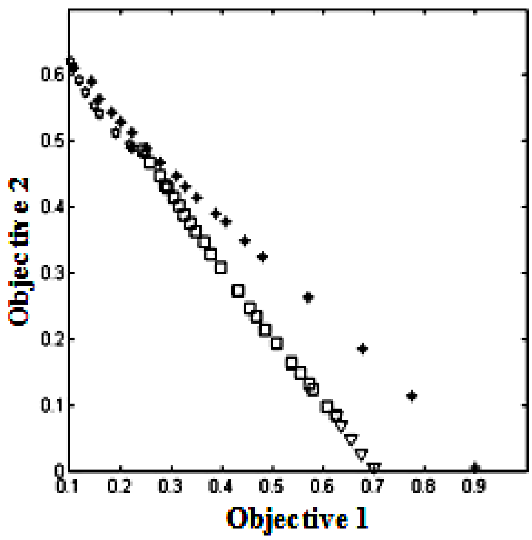


Figure 4.28a: OSF for W_2 =0.8

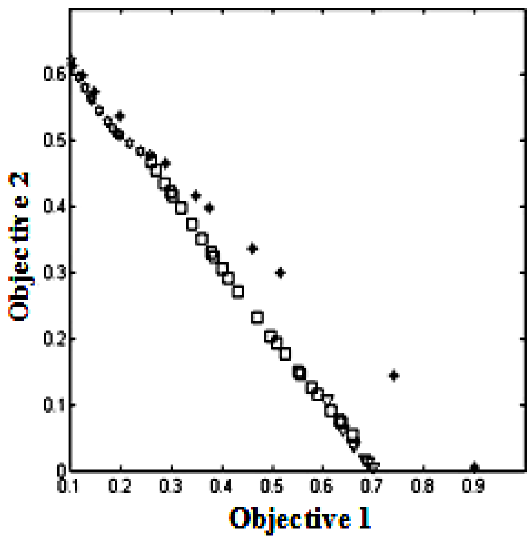


Figure 4.28b: OSF for $W_4 = W_8 = 0.5$

This is due to the insufficient upgrade of their fitness. Now let the DMs' discussion result in assigning W_2 = - 0.8, and W_1 = 0.9. Running the interactive concept-based evolutionary algorithm with this assignment results in the OSF depicted in figure 4.29.

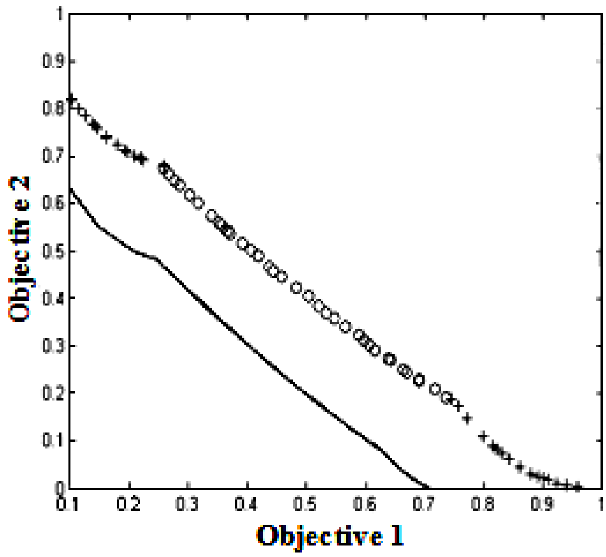


Figure 4.29: OSF for: W_2 = - 0.8, and W_1 = 0.9

The resulting OSF is a shared front, containing representatives of CC_1 and CC_5. It does not include any representatives of the CBF, which is depicted in the figure by continues curve. Table 4.12 summarizes the measures (see) for this example.

Table 4.12: Measures of successes for the hierarchical case

Measure	CC_1	CC_5
no_m	0	0
n^m_{OS}	60	50
SDn^m_{OS}	6	5
D^m_{OS}	0.85	1.08
SDD^m_{OS}	0.04	0.09

The measures values as summarized in table 4.12 leads to the same conclusions expressed following tables, 4.8a, b and table 4.10.

To enhance a search of solutions belonging to a specific CC the preferences assigned should guarantee its solo evolution. For example, the following assignment is attempted with $W_4 = W_5 = W_7 = W_8 = -0.5$, $W_2 = -0.8$, $W_3 = W_6 = 0.7$. The resulting OSF is depicted in figure 4.30. The only surviving representatives are those of CC_5.

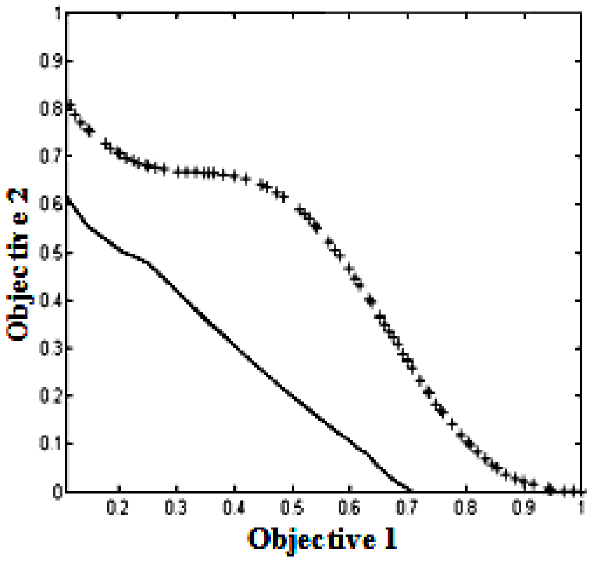


Figure 4.30: Concept elevation

4.2.3 Structural mechanics example

To demonstrate the effect of interactivity in an engineering setting, the example described in section 4.1.4, with loads of F_1=10N and F_2=15N is considered. The CBF for this case is depicted in figure 4.20. It is an intersecting front that holds representatives of all CCs. Now suppose that the DMs prefer the use of two bars truss over the use of a three bar truss. Such a preference may be expressed by assigning $W_1 = -0.5$ and W_4=0.8. The resulting OSF using the interactive concept based evolution is depicted in figure 4.31.

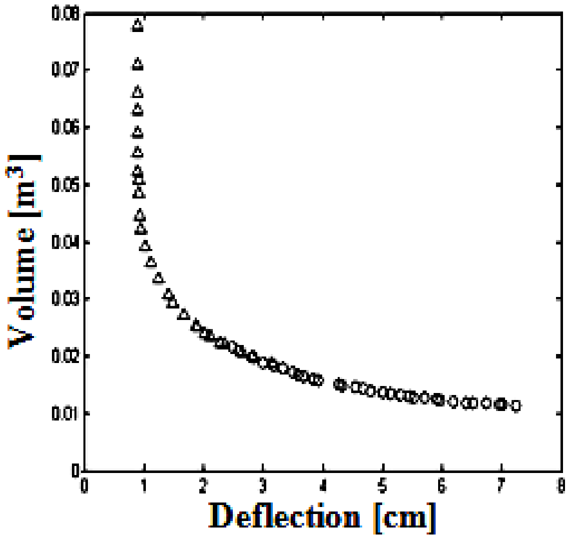


Figure 4.31: OSF for W_1=-0.5 W_4=0.8

It is observed that the resulting OSF is still an intersecting front but now it holds just representatives from CC_2 and CC_3. The representatives of CC_1 did not survive due to the low preference assigned to it and the high preference assigned to the other CCs through their highest level in the hierarchy.

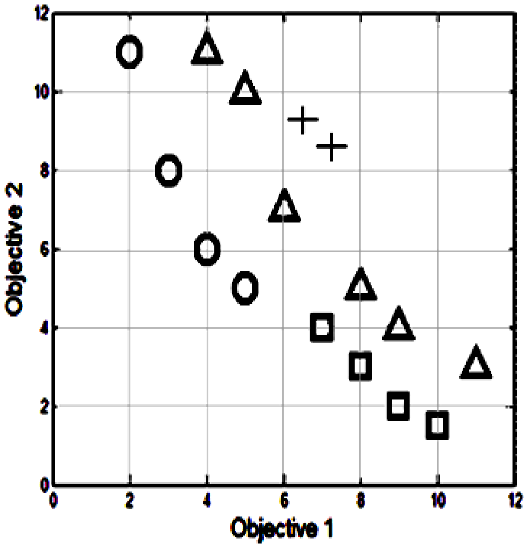
From the examples of section 4.2, the following may be concluded:

1. The OSF representation is mostly consistent as concluded from the results, which are summarized in tables 4.8a, b, 4.10 and 4.12.
2. There may be certain values of weights that may cause boundary preference that may cause the front not to be consistent. This means that the values for the standard deviations measures cross the 0.1 value and even reach the value of 0.5! Such perturbations in the representation are therefore associated with boundary preferences where the subjectivity dictates uncertainty.

4.3 Case studies for the conceptual selection support approach

In this section, hand calculation examples are given to demonstrate and to further explain the O&V approach for the support of concept selection.

The problems of figure 3.14a and 3.14b are further used here, with an added CC, CC_4 (pluses), to demonstrate the proposed method. The WOIs of examples 1 and 2 are divided as depicted in figures 4.32a and 4.32b respectively. The division of each axis in these examples is 6. This is in accordance with the maximal number of representatives any CC has in each problem.



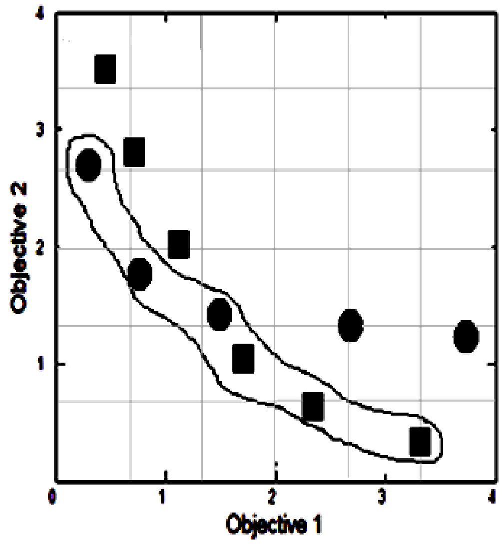


Figure 4.32 a: Axes partitioning for ex. 1 4.32b: Axes partitioning for ex. 2

The O&V values for the CCs of the examples are summarized in table 4.13.

Table 4.13: O&V values

Measure	Example 1	Example 2
V_1	0.5	0.66
O_1	1	1
V_2	0.33	0.83
O_2	1	1
V_3	0.91	---
O_3	0.5	---
V_4	0.166	---
O_4	0.33	---

The plots of these results, in the bi-objective space of the auxiliary MOP, are depicted in figure 4.33a and 4.33b.

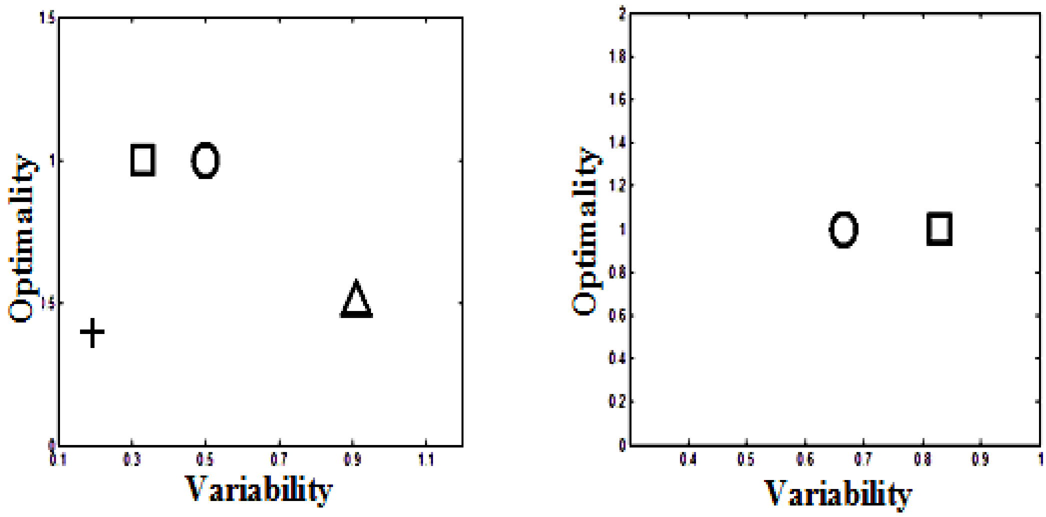


Figure 4.33: a: O&V for example 1 b: O&V for example 2

In contrast to the 'goodness' measure, the results of the O&V approach are more adequate in the sense that they highlights the trade-offs between the O&V objectives. In example 1, which is associated with the auxiliary MOP representation of figure 4.33a, there are two out of the four CCs which have O&V values on the auxiliary MOP front (circles and triangles). Considering Pareto-optimality of the auxiliary MOP, each one of these two CCs may be selected. These results show the importance of considering CCs, which are not a part of the CBF (the triangles in this example), as they might be 'optimal' in the O&V sense. In the second example the superiority of CC_2 over CC_1, which is associated with variability, is highlighted by the O&V approach. This demonstrates the other aspect of the auxiliary MOP, which is not considered by the 'goodness' measure, as explained in section 3.4.1.

4.4 Case studies for the delayed decision problem

The procedure introduced in section 3.5 is demonstrated here for two cases. The first is an academic example, while the second is associated with an engineering conceptual design problem.

4.4.1. Academic example

The academic example is depicted in figure 4.34.

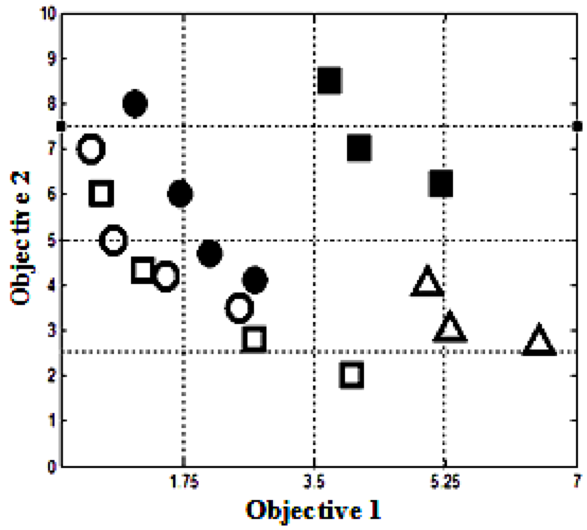


Figure 4.34: Demonstration for the delayed decision problem

In Figure 4.34, two MMCs are shown. MMC_1 involves the circles designating its ACCs (ACC_1 and ACC_2) by the blank and filled circles respectively (this is kept throughout the figures below). Similarly the other MMC, MMC_2 involves ACC_3 and ACC_4, which are designated by blank and filled squares respectively. In addition a CC which is not an ACC, and is not affected by the delayed decision, is designated by triangles.

For the above example the axes divisions (4 for each axis) are also depicted in figure 4.34. The numerical results for the O&V are given in table 4.14. The table also includes the numerical values of the worst cases.

Table 4.14: O&V worst case results

Concept	Optimality	Variability	Worst case/s	
			Opt.	Var.
ACC1	1	4/8	1	4/8
ACC2	0.5	5/8	0.5	4/8
ACC3	1	6/8	1/3	3/8
ACC4	1/3	3/8		
CC	0.5	3/8	0.5	3/8

The MMCs and the CC representatives in the auxiliary bi-objective space are depicted in figure 4.35a. Following the worst-case sorting procedure (see section 3.5.2), figure 4.35b, depicts the obtained MMCs sets for both cases.

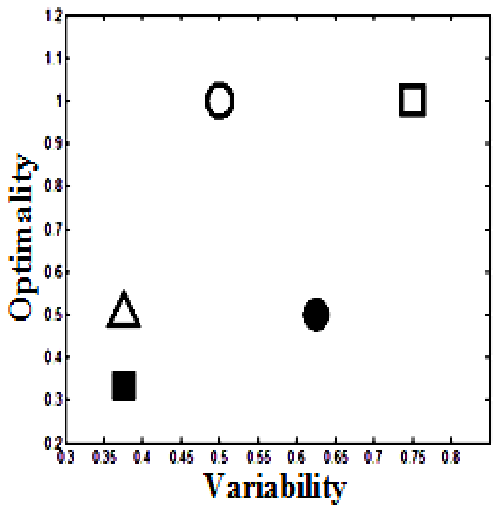


Figure 4.35a: The CCs representatives

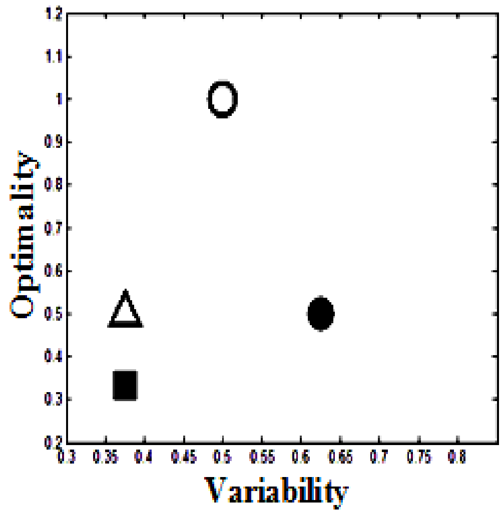


Figure 4.35b: MMCs' representatives

It is noted that although MMC_2 is associated with the CC, which has the highest O&V measure (ACC_3), it has poor performances when robustness to delayed decisions is considered. According to the results, which are depicted in figure 4.35b, MMC_1 has the best of the worst performances in the auxiliary MOP and should be preferred over the other MMCs.

4.4.2 Structural mechanics example

The truss arrangement (see section 4.1.4) is reused here. A NSGA-II algorithm with 50% crossover, 1% mutation, tournament selection and a population of 50 individuals is run for each of the truss' models. The resulting CCs fronts are depicted in figure 4.36, designated as in section 4.1.4, by squares, triangles and circles for CC_1, CC_2 and CC_3 respectively.

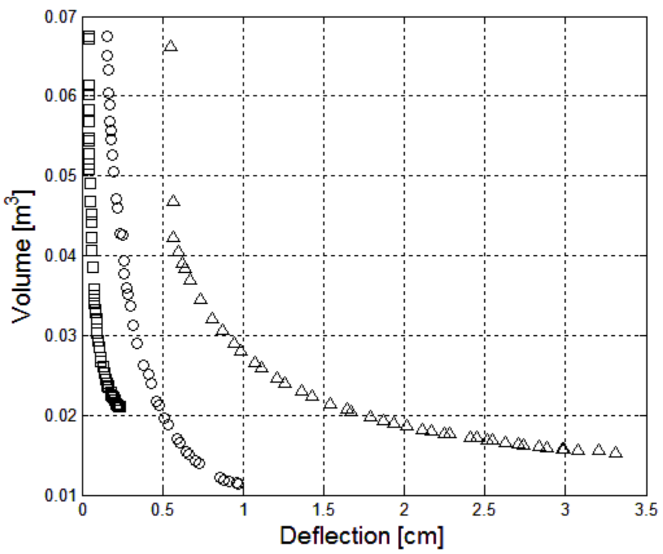


Figure 4.36: The CCs fronts

In this example it is assumed that the decision on the exact configuration of the two bar truss is delayed. Figure 4.37 depicts the tree representation of the conceptual design space of the current problem, which is an OR tree. It is noted that in this example SC_0 and CC_1 are the same, $CC_2 = SC_1 \wedge SC_2$, and $CC_3 = SC_1 \wedge SC_3$.

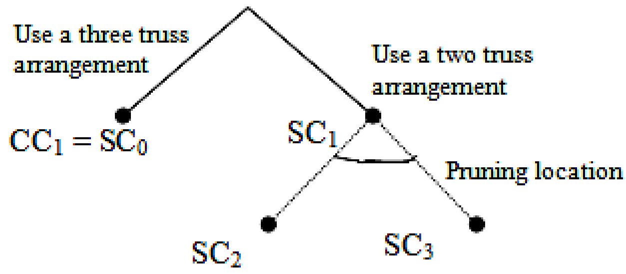


Figure 4.37: Design space tree with the pruning location

Now suppose that a decision on whether to continue with a three bar truss or a two bar truss should be taken (e.g., to decide between ordering two or three hanging devices). In addition, suppose that the architect of the structure approved the three truss arrangement but is unsure about which of the two two-bars truss arrangements she would approve. This means that a decision under this uncertainty should be taken, and the decision on the exact two bar arrangement is delayed. The effect of delaying the decision is shown as the pruning below SC_1. This pruning results in one CC (CC_1) and one MMC with its ACCs (CC_2, CC_3). The DMs are interested in solutions with a WOI defined by the entire boundaries of the figure.

Following the procedure outlined in sections 3.4 and 3.5, the bi-objective O&V selection space is depicted in figure.4.38.

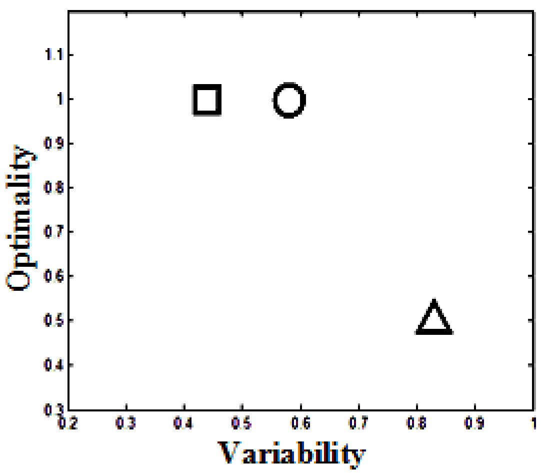


Figure 4.38: O&V selection space

In this case both ACCs (circles and triangles) are the worst cases of the MMC. Both are robust concepts. Considering optimality within the auxiliary MOP and in the Pareto sense the two bar MMC is better. In any case the decision on a final bars arrangement may be postponed.

4.5 Mechatronic design example

In this section a simple mechatronic example is used to demonstrate the applicability of the approaches, introduced in the thesis, to mechatronic design. The conceptual decisions that have to be made in this example belong to both mechanical and control disciplines. Namely, a concept for both the structure and the control of a one-arm manipulator is to be found. The arm is shown in figure 4.39 (side view).

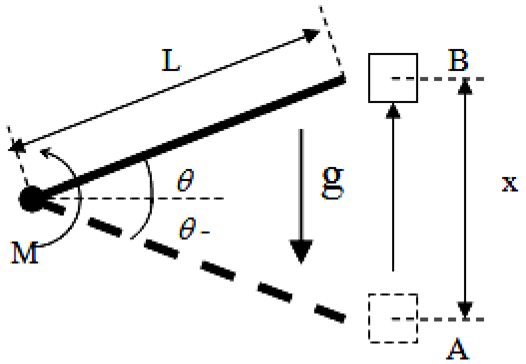


Figure 4.39: One arm manipulator (side view)

A load of mass M_L is to be raised to a location B at a height x above an initial location A. An arm of length L, and mass m_A, is used for the lifting. The arm is manipulated by a torque M at its base determined by a controller. The bi-objective problem involves the minimization of:

$$f_1 = ISE$$
$$f_2 = \Delta f$$

Where, $ISE \equiv \int_0^{t_{final}} error \cdot dt$, $error = \theta_{SP} - \theta$. Δf , is the deflection of the end-effecter. θ_{SP} is determined by the length of the arm. The dynamic equation of the arm is: $(\frac{m_A}{6} + M_L)L^2 \frac{d^2\theta}{dt^2} = M - (\frac{m_A}{2} + M_L)gL$ where $M = K_C \cdot error + \tau_D \frac{derror}{dt}$ for a PD control and $M = K_C \cdot error + K_I \int_0^{t_{final}} (error)\,dt + \tau_D \frac{d(error)}{dt}$ for a PID control. K_C, K_I and τ_D , are controller gain, reset time, and derivative gain, respectively. The weight of the arm is calculated by: $m_A = \rho g L$ where g is the gravity and $\Delta f = \frac{m_A g L^3}{8EI} + \frac{m_L g L^3}{2EI}$, where $I = bh^3$.

The following SCs are used: 'use a long arm with a large cross section' (SC_1), 'use a short arm with a small cross section '(SC_2), 'use a PD controller' (SC_3), 'use a PID controller' (SC_4). It is noted that SC_1 and SC_2 might be a result of a requirement to use in-stock arms. The design space tree is depicted in figure 4.40.

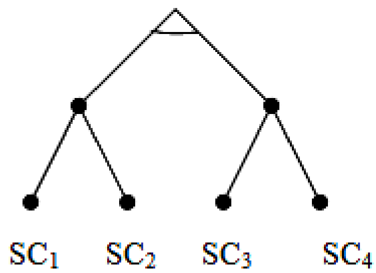


Figure 4.40: Conceptual design space tree – mechatronic example

SC_1 differs from SC_2 by the bar cross section, the material, and the search space limits at which the values for the arm length are searched for. SC_3 differs from SC_4 both by the model to compute the controller force and by the number of the design parameters. Tables 4.15 and 4.16 summarize the SCs parameters and constants.

Table 4.15: SC_1 and SC_2 related parameters and constants

SC #	Density ρ	Module of elasticity, E	Cross section, $b \cdot h$	Length L
1	2700 kg/cm^3	70000 MPa	5cm x 10cm	$1m \le L \le 3m$
2	2700 kg/cm^3	70000 MPa	10cm x 10cm	$0.8m \le L \le 1.3m$

Table 4.16: SC_3 and SC_4 related parameters and constants

SC #	P	I	D
3	$1 \le K_C \le 300$	$1 \le K_I \le 100$	
4	$1 \le K_C \le 300$	$1 \le K_I \le 100$	$1 \le \tau_D \le 300$

Four CCs may be extracted from the conceptual design space. Table 4.18 summarizes the CCs their related SCs, their description, and their designating legends.

Table 4.17: Summary of SCs, CCs and their legends

CC #	SCs #	Description	Legend
1	1,3	long arm with large cross section PD control	star
2	1,4	long arm with large cross section PID control	plus
3	2,3	short arm with small cross section PD control	triangle
4	2,4	short arm with small cross section PID control	square

A load of 0.8kg is used in the following simulations. The distance, x is chosen to be 1.5m, which is rather large in comparison with the maximal length of the bar. This is selected to examine the control away from liner conditions.

4.5.1 Mechatronic example – CBF

The MOEA used here is the C_1-NSGA-II algorithm (see section 3.2.2.2). Each sub-population of the CCs is initialized with 25 individuals. The algorithm parameters are kept as in section 4.1. A part of the initial population is depicted in figure 4.41a. The CBF location is marked by black points. The change of the angle θ with time for several solutions, belonging to the initial population, is depicted in figure 4.41b. It is noted that the initial population does not contain optimal solutions (in the concept-based sense). This is not general as such solutions may occasionally appear in the initial population. It is further noted that longer arms are associated with a smaller absolute value of the initial angle θ (see the mathematical model for explanation).

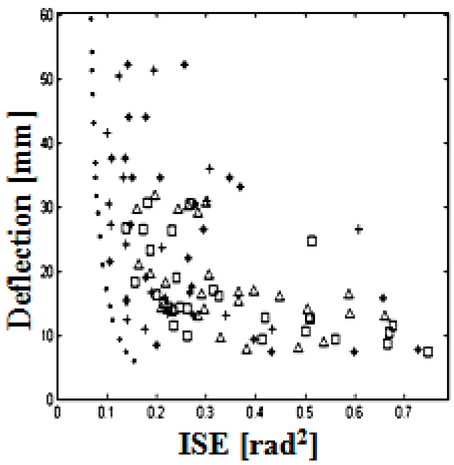


Figure 4.41a: Initial population performances

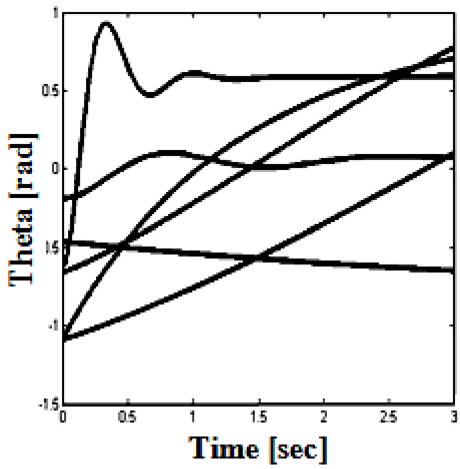


Figure 4.41b: Representing time responses of the initial population

In the first simulation, no preferences are assigned to the SCs. The result of using C_1-NSGA-II is depicted in figure 4.42a. It is noted that, as expected, the resulting front is similar, when alternatively using the interactive concept-based MOEA with no assignment of preferences.

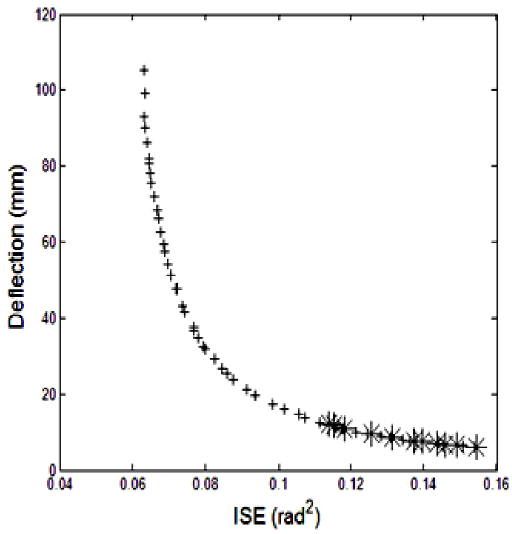


Figure 4.42a: The CBF

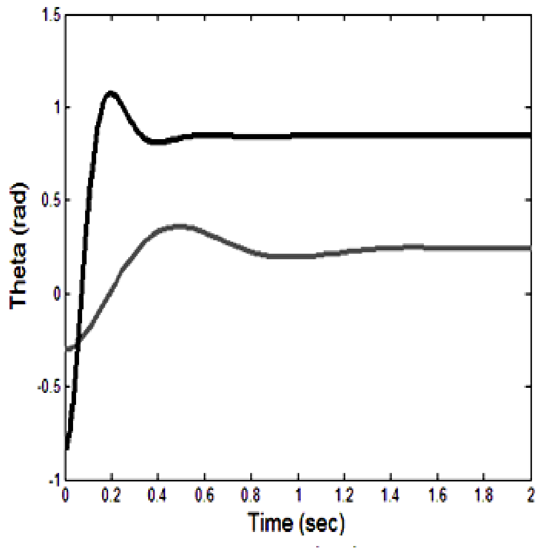


Figure 4.42b: Time responses of the boundary solutions

The figure depicts the CBF, which is an intersecting one. It holds along all its length, solutions belonging to CC_2 and in its lower part it holds also solutions from CC_1. Both CCs are associated with the larger cross section of the arm. Figure 4.42b depicts the time responses of the boundary solutions of the front. The minimal ISE with maximal deflection is associated with a PID controller that uses the larger cross section and long arm (2.7m). The time response for that solution is depicted in figure 4.45b, designated by grey. It is noted that the use of a long arm results in a shorter movement (less angle to travel). A shorter travel means a smaller change in the angle theta. A linear controller may fit better to such smaller changes of theta as the model is associated with nonlinearity. On the other end of the front, there are solutions that belong to the intersecting concepts. Both solutions use the CCs shortest possible arms of the longer arms (1m). The use of a shorter arm (with larger cross section) allows a fast response with low deflection for both the PID and the PD controllers. The time response of CC_1 is designated in figure 4.42b by the black response curve. Table 4.18 summarizes the results of comparing the results using C_1-NSGA-II and a sequential algorithm (see section 3.2.2.2) based on the first concept-based indicator (see section 3.2.2.4). Results for the second concept-based indicator can not be found since that an analytical front is not available.

Table 4.18: Comparing simultaneous vs. sequential approach

	Simultaneous		Sequential	
Indicator	CC_1	CC_4	CC_1	CC_4
CFO_m	30	65	6	23

The results shown in the table are consistent with the results of the academic example (see section 4.1.2).

4.5.2 Mechatronic example – Interactivity

The interactive EC algorithm is examined by using the problem of section 3.2.4.4. The preferences towards the SCs are changed. Now the DMs assign the preference weights of W_3= 0.4 and W_4= -0.6. These weight assignments express that the DMs prefer a PD controller over a PID controller. The resulting OSF is depicted in figure 4.43a. It holds just the solutions of CC_1. Now suppose that the DMs' preference is increased towards the PD controller by assigning the preferences weights as: W_3= 0.8 and W_4= -0.8. The resulting OSF is depicted in figure 4.43b. It now holds two CCs fronts (CC_1 and CC_3), which are both associated with the preferred SC of the PD controller.

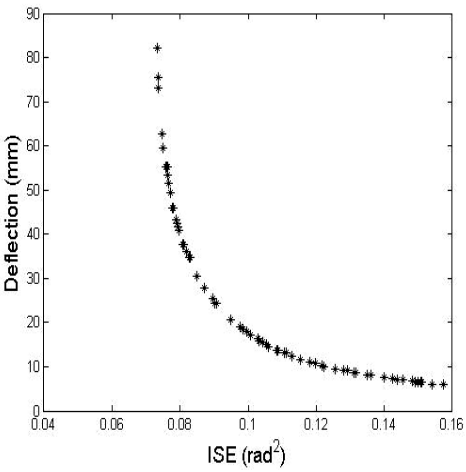


Figure 4.43a: OSF for W_3= 0.4 and W_4= -0.6

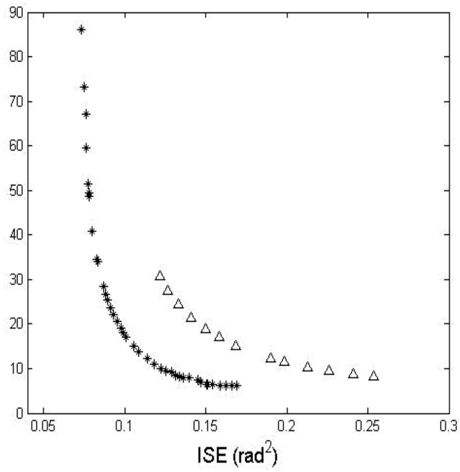


Figure 4.43b: OSF for W_3= 0.8 and W_4= -0.8

It is noted that the obtained values for the measures introduced in section 3.6.2.2, which are not shown here, do not indicate a change from the results shown in tables 4.8a, b, 4.10, and 4.12.

4.5.3 Mechatronic example – Supporting decisions

To demonstrate the applicability of the algorithms introduced in section 3.4 and 3.5, to the mechatronic example, two different delayed decisions are considered. The first case involves a delayed decision on the mechanical SCs. This means that the decision is postponed on using a larger or a smaller cross section for the arm. In the second case the decision on the controller is delayed. This means that the decision on using a PD or a PID is postponed. The pruned trees, which are depicted in figures 4.44a and 4.44b, are for the first and second case respectively.

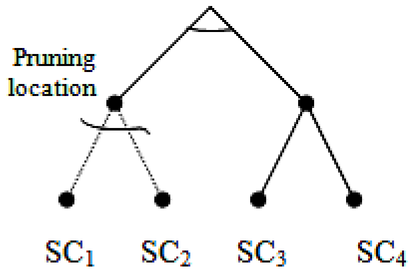


Figure 4.44a: Delaying structure decision

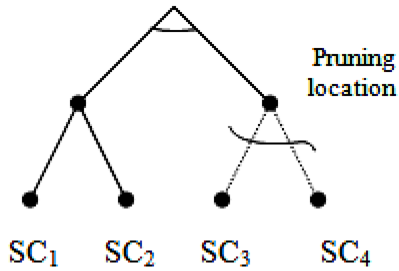


Figure 4.44b: Delaying control decision

For both cases there are two MMCs, which are associated with two ACCs each. Table 4.19 summarizes the MMCs and the ACCs for the two cases.

Table 4.19: MMCs and their ACCs

	Case 1 ACCs	Case 2 ACCs
MMC_1	CC_1 and CC_3	CC_1 and CC_2
MMC_2	CC_2 and CC_4	CC_3 and CC_4

Running NSGA-II for each of the CCs result in the CCs fronts depicted in figure 4.45.

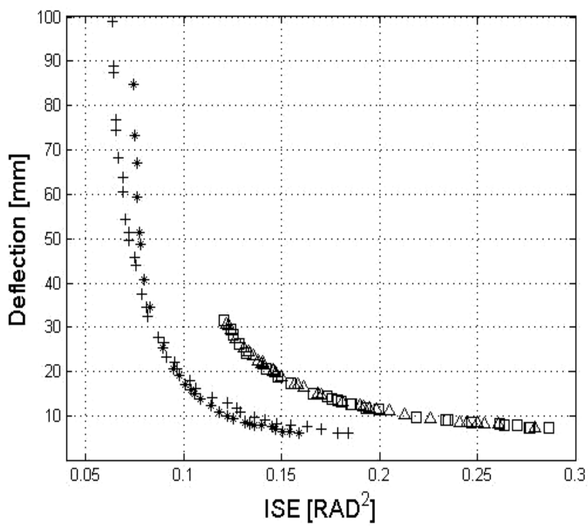


Figure 4.45: Four mechatronic CCs' fronts

For the ACCs of the problem the representations in the auxiliary MOP objective space, is depicted in figure 4.46.

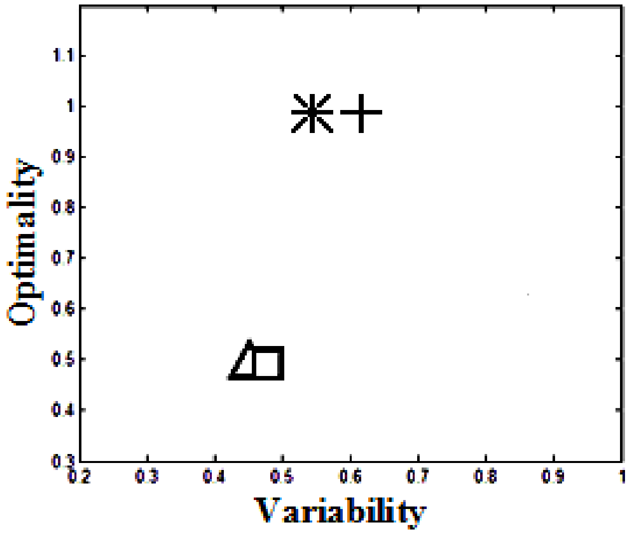


Figure 4.46: Auxiliary objective space for the mechatronic example

Considering the two cases (case 1 and case 2) of table 4.19, the worst set for each MMC and for each case are depicted in figure 4.47a and 4.47b respectively.

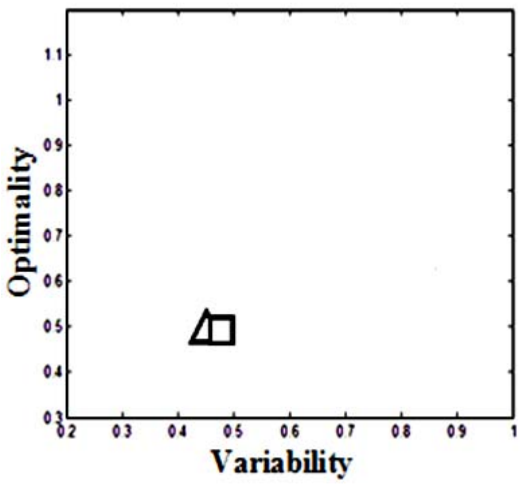


Figure 4.47a: Auxiliary objective space for case1

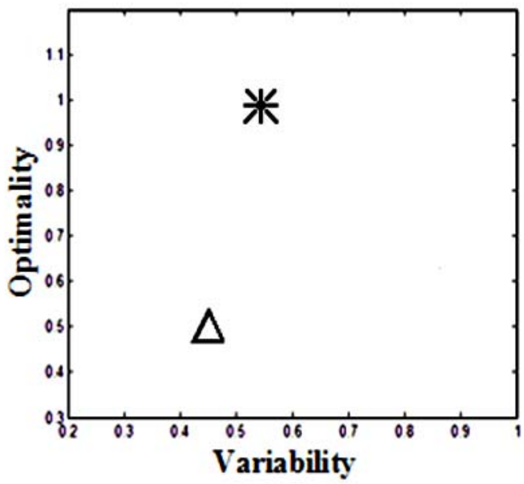


Figure 4.47b: Auxiliary objective space for case2

It is observed from the results that in case 1, both MMCs are robust. However, their O&V values are similar hence are quite as good for selection. In contrast, in the second case although both MMCs are robust, MMC_1 is 'optimal' in the auxiliary objective space.

In another scenario a different WOI may be considered by the DMs. Suppose that in the new scenario the designers set the WOI to be bounded by a deflection of 70 mm and an ISE of 0.11

rad^2. The lower bounds are determined as (0, 0). For this scenario the auxiliary MOP would result in the two MMCs sets at the origin for case 1. This is due to the fact that both CC_3 and CC_4 have their entire fronts located outside the WOI. In case 2, the representative of MMC_2 has values for O&V that are (0.54, 1), and (0, 0) for MMC_1. Therefore the decision on the structure may not be postponed, whereas postponing the control is possible. This means that MMC_2 is the only robust concept. In this case (case 2) the long arm with large cross section structure may be chosen (including order of material and adequate manufacturing related items).

CHAPTER 5

SUMMARY CONCULUSIONS and FUTURE WORK

The main goal of this thesis is to advance the development of methodologies and computational tools, which support conceptual design, and in particular with respect to multi-objective problems. This study encompasses four main issues including: a. simultaneous evolution of concepts' solutions towards and along a Pareto front, b. interactive evolution of concepts' solutions towards optimal solutions of preferred concepts, c. assessment and comparison of concepts in the multi-objective space and d. supporting decision making with uncertainties due to delayed decisions.

The development of the simultaneous approach for the evolution of concepts (handling the first issue) serves as a base towards handling of the second issue - interactivity. Yet, the simultaneous approach has some merits on its own, when compared with the sequential approach that has been used by others. The proposed simultaneous evolution of solutions from different concepts towards a Pareto front (CBF), involves several computational issues, including the management of resources among sub-populations and within each sub-population. To enhance such an evolution, a basic and a modified algorithm are suggested. The basic algorithm, C_1-NSGA-II, performs a simultaneous evolution of CCs, while sharing the computational resources in accordance with the CCs' optimality. This has been achieved by introducing concept-based crowding sort and concept-based tournament. The need to balance between the concept resources is highly desired not only from the need to reach all optimal concepts, but also from the need to balance their representations on the front. Furthermore, in contrast to the case of a traditional MOP, a simultaneous evolution of concepts requires considering potential situations in which the time to compute the fitness of a solution may strongly differ from one CC to the other.

To improve the time performance of the simultaneous approach a modification of C_1-NSGA-II has been introduced, namely the C_2-NSGA-II. In the modified algorithm, the transfer of computational resources is restricted by limiting the CCs' subpopulations to no more than their initial size. To compare the introduced algorithms with the existing sequential approach, some performance measures have been adapted from EMO studies to the C-EMO needs. Moreover, a measure for comparing between the simultaneous approach and the sequential one, on the base of computational time, has been introduced. Academic as well as engineering examples are used to demonstrate the simultaneous approach and its applicability to conceptual engineering design.

The proposed simultaneous C-EMO has been proven to be a good foundation to deal with interactivity within MOPs , which has been the initial motivation for the simultaneous approach. As mentioned above, the simultaneous approach became a significant approach on its own. This is evidence from the enclosed comparison with a sequential approach. The comparison also includes an interrogation of the two proposed simultaneous evolution algorithms. It shows that: a. C_1-NSGA-II produces better representation performance than the C_2-NSGA-II, or the sequential approach, with a cost of computational time; b. C_2-NSGA-II involves shorter computational time than C_1-NSGA-II at the cost of representation performance; c. The results of comparing the computational time of the sequential approach vs. the simultaneous approaches depend on the nature of the problem.

The search of the conceptual design space for promising concepts usually involves situations in which the designers should take into account not only the optimality of the concepts' solutions, as based on available models, but also the influence of their subjective preferences towards the concepts. To incorporate such situations, this thesis includes the formulation and a solution approach to the interactive concept-based MOP (IC-MOP). This formulation involves the introduction of the objective-subjective front (OSF). The OSF is an optimality-based front. It includes just solutions that belong to the CCs' fronts, which are based on their available models. Due to the incorporation of subjectivity, it may coincide fully or partially with the CBF, or be totally different from it.

To deal with interactive problems, the conceptual design space may be represented by either a non-hierarchical or a hierarchical representation, depending on the problem decomposition. In both cases the DMs' preferences are expressed towards the SCs, or alternatively towards the CCs, by assigning weights. A measure, which accumulates the SCs weights in the hierarchical case into a concept-weight, has been suggested. An interactive concept-based simultaneous EC search algorithm has been introduced to allow solving the IC-MOP. In the interactive algorithm the evolution of an individual solution is influenced both by the rank of the solution performances based on a model, and by its preference in accordance with its concept association. The simultaneous nature of the interactive evolution means that resource sharing within a rank and within each CC has to be considered. For this purpose C_1–NSGA-II algorithm has been modified into the IC-NSGA-II algorithm. The evolutionary search is aimed at finding a set of solutions such that their performances constitute the OSF. Assessing the validity of the resulting OSF is done by introducing validity measures. Academic examples as well as engineering examples demonstrated the use and the applicability of the interactive approach. It is evident from the examples that the interactivity, as related to SCs, concentrates the computational resources towards a search of optimal solutions that are related to the preferred SCs.

It is apparent that choosing an EMO approach, to serve as the foundation for the IC-MOP, has been a good decision.

Obtaining the CBF or the OSF in a concept-based problem is just a step towards the selection of a concept, and subsequently the selection of a particular solution. The CBF and the OSF are valuable representations, yet they lack some important information concerning the variability of the concepts' performances. It has been shown in this thesis that a CC, which is not a part of a CBF (or an OSF) might still be considered for selection when the variability is also considered. This thesis suggests a method to support the selection of a concept by comparing between the concepts, using an auxiliary multi-objective problem with optimality and variability (O&V) as the objectives. It is noted that the introduced approach is not limited to presentations that include all CCs' fronts within a WOI. They may be also used to compare CCs based on a CBF or on an OSF representation.

Conceptual design is characterized with the presence of a variety of uncertainties, which may influence the selection of a concept. In this thesis one type of uncertainty is treated. It involves an uncertainty that is associated with delayed decisions. In this thesis the delayed decision problem has been stated with respect to a MOP. It has been shown that delaying a decision may be involved with a concept that is associated with more than one CC (ACC). Such a concept has been termed an MMC (multi-model concept). As such, it is associated with more than one model, and therefore, with more than one Pareto front. This means that in the case of contradicting objectives the location of the Pareto front is uncertain, as either of the ACCs of the representation may be later selected. In this thesis an O&V approach to select an MMC, based on its multi fronts representations has been suggested. The O&V method is adapted by sorting the CCs representatives for non-dominance in a reversed auxiliary MOP. In this case the MMC is represented by its worst ACC performances in the auxiliary MOP objective space. It has been shown that the proposed approach may allow the continuation of a design, under delayed decisions, while ensuring the selected concept robustness to later decisions.

In this thesis the support of DMs' conceptual decisions is concerned with engineering related problems. The engineering examples, and in-particular the mechatronic one, demonstrate the applicability of the introduced approaches and algorithms to engineering design. The mechatronic design example highlights the applicability of the proposed techniques to a simultaneous search for mechanical-concepts and control-strategies.

Based on the conducted study, the contributions of this thesis with respect to former investigations and approaches, as related to the four main issues, are summarized below. The reader is also referred again to table 2.1 to link the topics of contribution to the proper sections of the thesis.

A*. *Simultaneous concept-based search

1. Simultaneous concept-based evolution

   The proposed simultaneous evolution is novel. There are cited studies that employs existing MOEAs to deal with concept-based MOPs (see section 2.3.1). Yet, none of the available studies deals with a simultaneous concept-based ***evolution of concepts towards and along a Pareto front*** as suggested in this thesis. Observing the SOTA, as presented in section 2.2, it is concluded that the proposed simultaneous evolution is not only new, but also involves a substantially different algorithm from what has been used in past studies.

2. Managing resources while striving for optimality

   The simultaneous approach requires special considerations with respect to the utilization of resources such that a good approximation of the Pareto front. Here, during the evolution mating between the sub-population is avoided, and an inherent division of the population into sub-populations exists. This leads to a need to consider resource sharing between and within the sub-populations. The sharing between design sets, which is associated with design concepts, while striving for a multi-objective optimality (Pareto front) as done in this thesis, is unique. Observing the SOTA, as presented in section 2.2, it is concluded that this is the first time that such a concept-based evolution mechanism is introduced. Neither genotypic niching nor a-posteriori learning of concepts or clustering are equivalent to it. This is due to the fact that here concepts are pre-defined, may belong to different design spaces and may possess solutions at the same location of the CBF.

3. Comparing the simultaneous approach with a sequential approach

   The simultaneous approach, which is introduced in this thesis, calls for a comparison with a sequential approach. The comparison which is carried out in this thesis is based on measures that are adopted to assess the success of the algorithms to find a set that well represent concepts' optimality and to compare between their computational efforts. Observing the SOTA presented in sections 2.2, 2.3, it is concluded that a comparison between the approaches, which is based on measures is novel to this thesis. It has been shown that the simultaneous approach, although motivated by its support of interactivity, possesses by itself some major advantages over the former sequential evolutionary approach.

4. Developing and comparing different simultaneous approaches

   The simultaneous approach highlights the difference between the times to compute the solutions within the different concepts. Highlighting this issue is important, as it has an important influence on the assessment of the different C-EMO algorithms. No study has treated this issue before. Observing the SOTA as presented in sections 2.2, 2.3, it is concluded that the comparison between different simultaneous approaches is done here for the first time. Moreover the consideration of concept-related computational time within an EMO approach is novel to this thesis.

B. Interactive concept-based evolution

1. Introduction of the IC-MOP and its solution

   The interactive concept-based approach is novel. No former works (EC or non EC introduced such a problem or its solution. This is true for both the goal attainment (see Avigad *et. al.,* 2004) or for the development of the Objective subjective front.

2. Introduction of the IC-NSGA-II

   A novel EMO algorithm has been introduced to solve the IC-MOP problem. Neither IEC, nor other interactivity related EMO studies are to be confused with these thesis suggestions. This is due to the fact that here the interactivity is expressed towards concepts and not towards specific solutions/objectives. In the suggested approach not all the solutions are always a part of the non-dominant set of the entire objective space.

3. Hierarchical influence of preferences within a human-computer concept-based search-

   Utilizing preferences of designers towards SCs within a hierarchical representation has been found in a few works (see section 2.4.2). However, the use of such a hierarchical representation for the purpose of developing SBC related fronts, which are effected by both preferences and performances, is unique to the approach taken in this thesis.

C. Supporting selection of concepts

1. The introduced approach poses the decision on a concept as an auxiliary MOP, which opens the way to the establishment of a new paradigm for concept selection. Here, the selection support approach allows a selection of concepts, while taking into consideration the contradicting aspects of optimality and variability. Such an approach has not been introduced before.
2. An extension to the SOTA as related to the selection of SBCs is introduced. In

contrast to Mattson and Messac (2005), this thesis proposes a measure to select concepts within a window of interest, which allows a comparison between concepts that do not necessarily belong to the s-Pareto frontier. Therefore, it establishes a more general measure that may support designers in taking a conceptual decision.

D. Selection of concepts under uncertainty

1. The introduction of the delayed decision problem as a MOP is novel to this thesis.
   The use of EMO to support conceptual decisions with the presence of delayed decisions uncertainties has not been done before. The treatment of this sort of uncertainty is performed by the representation of the design space as a hierarchical tree, and the extraction of some unique concepts. This treatment expands the use of computational tools and specifically EMO to treat conceptual design related uncertainties.
2. Computer-based approach to support conceptual decisions with delayed decision
   None of the existing publications has suggested a computer-based support for taking conceptual decisions with the delayed decision uncertainty.
3. A new design space tree representation is proposed, which is fundamental to the treatment of the delayed decision problem by computers.
4. Robust worst case sorting
   This sorting is a new approach to treat robustness by considering a set of worst cases.

E. Conducting a SBC search approach to mechatronic design

1. The proposed application of the simultaneous evolution for mechatronic concepts and the associated approach for selecting an optimal and robust mechatronic concept are original to this thesis. Although others have noted and studied the importance of simultaneous evolution of morphologies (mechanics) and behavior (control). Only a few previous studies have incorporated it as a MOP. Moreover, no other study involves a selection framework, which is similar to what is proposed here. As such, this thesis includes an important example of simultaneity in mechatronic design.

It is clear that the study conducted within this thesis has reached the objectives set by the motivation to expand computational methods for supporting conceptual design. Yet, there is still a lot to investigate with respect to all the issues of this thesis, as discussed in the following.

Simultaneous C-EMO- Future work:

a. At this stage no accepted test functions for C-EMO are available. Such hard case functions should be constructed to allow future comparisons of different concept-based MOEAs, and for comparing them with other methods, such as those used by Mattson and Messac (2005). It is noted that a comparison between this thesis approach with that of Mattson and Messac (2005), should be practice on the basis of agreed test functions and problems.

b. The C-EMO algorithms should be further investigated and modified. These should include:

1. Adapting the algorithms to cases where a concept may be hindered due to computational factors. Such factors may include the existence of a local Pareto or bad initial population. One way to tackle the problem is to amalgamate the current algorithms with the algorithm, which was suggested in Avigad *et. al.,* (2005a). Such an algorithm allows the reappearance of concepts by way of mutation of their code. Another approach may be to save, during the evolution process, fronts of all concepts, and to always keep them within the population. This may allow concepts to improve by way of evolution of the kept concepts' solutions.
2. Apart from saving representatives from 'non-optimal' concepts, near optimal solutions should be also considered. With this respect, Moshaiov and Avigad, 2007, have introduced a modified version of the CBF to include such near optimal solutions.
3. A major influence on the balanced representation of concepts on the CBF is associated with the linear function, which involves η (see equation 3.13). With this respect other functions should be tested. Alternatively automatic tuning mechanism should be investigated. For example a mechanism that changes the resources according to the size of the different levels of non-dominance' hyper surfaces.

c. Although the C-EMO algorithms and the performance measures are all designed to allow their use within MOPs involving more than two objectives, examples of many objectives should be studied, and improvement of the algorithms to incorporate high dimensionality should be investigated. In-particular, the effect of the need to take larger populations in many objectives problems (when using NSGA-II) on the computational time should be investigated. It is noted that a major problem, in selecting a concept in many objectives problems, is the visualization problem. Nonetheless the approach of posing the problem as optimality versus variability may substantially assist in such situations. This should be demonstrated. It is further noted that some algorithmic issues should be further investigated, such as using a different encoding system (e.g., real encoding) and using a different MOEA as the base for the approach (e.g., SPEA2).

d. Another important issue that should be also investigated is the robustness within the framework of the C-MOP. In this thesis just robustness to delayed decisions has been treated. As stated in appendix, several other investigations concerning the robustness of concepts have been carried out (Avigad *et al.,* 2005c, Moshaiov and Avigad, 2006).

Nonetheless the effects of uncertainty of the concepts models as well as towards the design parameters have not been treated. Treating robustness in EMO is scarcely found and existing studies concerning robustness within the framework of C-MOP are seldom (as surveyed in sections 2.5.2, 2.5.2). The author of this thesis is currently working on several approaches to amalgamate uncertainties within the framework of C-EMO. It is clear that considering robustness within C-MOP is a crucial task and should be explored.

e. As stated several times, this thesis does not deal with the generation of concepts. In the future a way to insert creativity) should be searched for. Such approaches could be related embryonics (e.g., Bentley 1999). Moreover the search may be linked to a search within the internet. Such modifications should allow a much wider search for concepts.

f. In this thesis only academic examples as well as simple engineering examples are given. In the future more compound and sophisticated real life engineering problems should be utilized to demonstrate the benefits of the approach to industry.

g. The simultaneous C-EMO seems to possess a generic nature and might be applicable to fields other than engineering design (e.g., economics). This has been demonstrated by the employment of some of the ideas of this thesis to a robotic path planning problem (Moshaiov and Avigad, 2004). Moreover, the possible relation between the simultaneous evolution of concepts and the simultaneous evolution of species is an attractive idea that should be investigated, as recently suggested by Moshaiov (2006a, b). Such further studies are needed to strengthen any claim on the generic nature of the methods that are introduced in this thesis.

Future work on Interactive C-EMO::

With relation to the interactivity issue, future work may involve a comparison of the simultaneous and the sequential approach, for IC-MOPs, based on the measures of section 3.2. This might lead to a necessity to develop an interactive EMO algorithm that is based on C_2-NSGA-II rather than on C_1-NSGA-II. Moreover, a comparison with the evolution of OSF that has been initially investigated (e.g., Avigad *et al.*, 2005a) is needed. This comparison may be based on computational complexity, as well as on the quality of the OSF representation.

Future work on Selection of Concepts:

The O&V approach should be further investigated for different auxiliary spaces, and/or with more than two auxiliary objectives. For example, adding to the O&V space the dimension of commonality measure for family of designs. Further work may concentrate on improving the efficiency of the search algorithm. Such algorithms may adopt a simultaneous rather than a sequential approach by aiming at developing the CCs' fronts directly to be bounded by the WOI, or developing CCs that are optimal in the sense of the auxiliary MOP.

With this respect it is important to note that optimality in the sense of the CBF is different than that of optimality as related to the auxiliary MOP. Moreover these two different optimality aspects might rely on different representations of the concepts. Therefore, as suggested in point 'b' of C-EMO future work, saving fronts from all concepts during the evolution could be useful. This will allow a comparison between concepts based on the two aforementioned optimality aspects.

The delayed decision uncertainty, which has been treated in this thesis, is just one uncertainty typical to the conceptual design stage. Other uncertainties should also be incorporated within an EMO concept-based search.

The potential of EMO approaches to support designers within the conceptual stage is not yet fully explored; further investigations and development work should be conducted as discussed above. This might lead to the establishment of a practical concept-based EMO framework and tools, which will be used to actually support DMs in taking conceptual decisions. As a part of such studies, a comparison of the evolutionary approach with the solution approach taken by Mattson and Messac (2003, 2005), and other potential solution approaches, should be done. It is noted that regardless of the attractively of using bio-inspired methods, what really counts in computer-supported engineering work is the applicability and optimality of the chosen method. As many engineering problems involve time consuming evaluations of solutions, optimizing the computational efforts with respect to available hardware should be a major concern when comparing such methods.

The scope of this thesis is wide. It introduces new approaches and algorithms. The ideas introduced in the thesis seem to possess a generic nature and as such might be applicable not just for engineering conceptual design.

Chapter 6

Bibliography

1. AL-Salka M.A., Cartmell M.P., Hardy S. J. A Framework for generalized computer-based support environment for conceptual design, *Journal of Engineering Design* 9(1): 57-88, 1998.
2. Andersson, J. Multiobjective optimization in engineering design applications to fluid power systems, *Ph.D. Thesis submitted to Linkoping University*, Linkoping, 2001.
3. Andersson J. Sensitivity analysis in Pareto optimal design, *Proceedings. of the 4th Asia-Pacific Conference on Simulated Evolution and Learning, (SEAL 02),* Orchid, Singapore, 2002.
4. Armano, G., Cherchi, G., and Vargiu, E. Experimenting the performances of abstraction hierarchical planner, *Proceedings of IASTED, International Conference on Artificial Intelligence and Applications,* (*AI, 2003*), pp: 399-404, Innsbruck, Austria, 2003.
5. Arnold, D.V., and Beyer, H.G. A comparison of evolution strategies with other direct search methods in the presence of noise, *Computational Optimization and Applications*, 24:135–159, 2003.
6. Avigad, G., multi-multi objective optimization problem and its solution by a multi objective evolutionary algorithm. In the proceedings of EMO 2007, Japan, 2007. (Student award), 2007a.
7. Avigad, G., Solving the multi-multi objective optimization problem by a simultaneous MOEA, to appear in the proceedings of CEC 2007.
8. Avigad, G., Moshaiov, A., and Brauner, N. Towards a general tool for mechatronic design, *Proceedings of the 2003 IEEE CSS Conference on Control Applications, (CCA 2003),* pp: 23-25, Istanbul, Turkey, 2003.
9. Avigad, G., Moshaiov, A., and Brauner, N. Concept-based interactive brainstorming in engineering design, *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 8(5): 1-6, 2004.
10. Avigad, G., Moshaiov, A., and Brauner, N. Interactive concept-based search using MOEA: The hierarchical preferences case*, International Journal of Computational Intelligence,* 2(3): 182-191, 2005a.
11. Avigad, G., Moshaiov, A., and Brauner, N. MOEA-based approach to delayed decisions for robust conceptual design, In *Applications of Evolutionary Computing,* volume 3449 of LNCS, pp: 584 – 589. Springer-Verlag, 2005b.
12. Avigad, G., Moshaiov, A., and Brauner, N. MOEA for concept robustness to variability and uncertainty of market's demands, *Proceedings of the first AI*IA workshop on evolutionary computation*, (GSICE 2005), pp 62-68, Milan, Italy, 2005c.
13. Avigad, G. and Deb, K. The sequential optimization-constraint multi-objective problem and its applications for robust planning of robot paths, to appear in the proceedings of CEC 2007.
14. Baader, D., McGuinness, D., Nardi, and Patel-Schneider P. F., editors. *Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, 2002.
15. Bäck, T., and Schütz, M. Evolution strategies for mixed-integer optimization of optical multiplayer systems. In: McDonnell, J. R., Reynolds, R. G. and Fogel, D. B. (Eds.),

Proceedings of the Fourth Annual Conference on Evolutionary Programming, MIT Press, pp: 33–51, 1995.

16. Balling, R.J. Pareto sets in decision-based design, *Engineering Valuation and Cost Analysis,* 3: 189–198, 2000.
17. Banzhaf, W. *Interactive Evolution*, Handbook of Evolutionary Computation, 1997.
18. Begg, D.W. On simultaneous optimization of smart structures, algorithms and examples, *Computer Methods in Applied Mechanics and Engineering*, 184: 25-37, 2000.
19. Benet, A.P., and Rogers, A. The dynamics of a genetic algorithm for simple randomizing systems, *Physica*, D, 104: 75-114, 1997.
20. Bentley, P.J (Ed.) *Evolutionary Design by Computers*, Morgan Kaufman Pub., 1999.
21. Bentley, P. J., and Corne, D.W. *An introduction to creative evolutionary systems*. Creative Evolutionary Systems, New York: Academic Press, pp: 1-75, 2002.
22. Bentley, P.J, and Wakefield J.P. Conceptual evolutionary design by a genetic algorithm, *Engineering Design and Automation* 3(2): 119-131, 1997.
23. Bergmann, R., and Wilke, W. Building and refining abstract planning cases by change of representation language, *Journal of Artificial Intelligence Research* 3: 53-118, 1995.
24. Bi, Z.M., and Zhang, W.J. Concurrent optimal design of modular robotic configuration, *Journal of Robotic Systems*, 18(2): 77-87, 2001.
25. Biles, J.A. Life with GenJam: Interacting with a musical IGA, in *Proceedings of IEEE International Conference on Systems, Man and Cybernetics (SMC'99)*, 3: 652-656, Tokyo, Japan, 1999.
26. Blackenfelt, M. Profit Maximization while considering uncertainty by balancing commonality and variety using robust design–The Redesign of a family of lift tables, *ASMED design Engineering Technical Conferences–Design for Manufacturing*, Baltimore, MD, ASME, Paper No.DETC2000/ DFM-140.13, 2000.
27. Boden, M. *The creative mind: myths and mechanisms*, London Cardinal, 1992.
28. Borgida, A., and Brachman, R. J. *Conceptual modeling with description logics*. In Baader, D. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. Description Logic Handbook: Theory, Implementation and Applications. Cambridge University Press, 2003.
29. Bosman, P.A.N., and Thierend, D. The balance between proximity and diversity in multiobjective evolutionary algorithms, *IEEE Transactions on Evolutionary Computation*, 7 (2):174-188, 2003.
30. Branke, J. Creating robust solutions by means of evolutionary algorithms, *Proceedings of Parallel Problem Solving in Nature (PPSN 98)*, 5:119-128, 1998a.
31. Branke, J. *Creating robust solutions by means of an evolutionary algorithm*. In Eiben, A. E., B¨ack, T., Schoenauer, M., and Schwefel, H.P., editors, Parallel Problem Solving from Nature, volume 1498 of LNCS, pp: 119–128, Springer, 1998b.
32. Branke, J., Kaußler, H., and Schmeck, H. Guidance in evolutionary multi-objective optimization, *Advances in Engineering Software*, 32:499-507, 2001a.
33. Branke, J. Reducing the sampling variance when searching for robust solutions. In et al., L. S., editor, *proceedings of Genetic and Evolutionary Computation Conference (GECCO'01)*, pp: 235–242. Morgan Kaufmann Publishers. 2001b.
34. Branke, J. *Evolutionary optimization in dynamic environments*, Kluwer Academic publishers, 2001c.
35. Brett, P., and Edward, F. Evolutionary design of controlled structures. *Journal of Aircraft*, 36(1): 209-217, 1999.

36. Bullinger, H.J., Ohlhausen P., Prieto, J., and Bröcker, A. Concurrent simultaneous engineering (CSE) – New orientation in product development, *Proceedings 1st CME-Congress* Bremen, Germany, 1998.
37. Caballero, R., Nunez, T. G., Luque, M., Garcia, F. M., and Ruiz, F. Hierarchical generation of Pareto solutions in large-scale multiobjective systems, *Computers and Operation Research*, 29(11): 1537-1558, 2002.
38. Cantú-Paz, E. A Summary of research on parallel genetic algorithms. Technical Report IlliGAL No. 95007, July 1995, University of Illinois at Urbana-Champaign, 1995.
39. Cartlidge, J.P. Rules of engagement: competitive coevolutionary dynamics in computational systems, *PhD. submitted to the, University of Leeds*, Leeds, 2004.
40. Cerrolaza, M., and Annicchiarico, W. *Genetic algorithms in shape optimization: finite and boundary element application*, Evolutionary algorithms in engineering and computer science, Edited by K. Miettinen, M. Makela, P. Neittaanmaki, J. Périaux, Jonh Wiley & Sons LTD., Capítulo 14: 283-326, 1999.
41. Chakravarty, A.K., and Balakrishnan, N. Achieving product variety though optimal choice of module variations,' *IIE Transactions*, 33(7): 587-598, 2001.
42. Chan, K.H., Frazer, J., and Tang, M.X. Handling the evolution and hierarchy nature of designing in computer-aided design support systems, *Proceedings of the third international conference Computer-aided industrial design and conceptual design (CAID & CAD, 2000)*, pp: 447-454, Hong Kong, China, 2000.
43. Chan, K.P and Ray, T. An Evolutionary algorithm to maintain diversity in the parametric and the objective space, *Proceedings of the third International Conference on Computational Intelligence, Robotics and Autonomous Systems,* (*CIRAS 2005*), Singapore, 2005.
44. Chen, B.S., and Cheng, Y.M. A structure-specified H-infinity optimal control design for practical applications: A genetic approach, *IEEE Transactions on Control Systems Technology*, 6(6): 707-718, 1998.
45. Chen, W., Allen, J.K., Tsui, K.L., and Mistree, F. A Procedure for robust design, *Journal of Mechanical Design*, 118: 478-485, 1996.
46. Chin, K.S., and Wong, T. Knowledge-based evaluation for the conceptual design development of injection molding parts, *Engineering Applications in Artificial Intelligence,* 9(4): 359–376, 1996.
47. Chipperfield, A., and Fleming, P.J. Multiobjective gas turbine engine controller design using genetic algorithms, *IEEE Transactions on Industrial Electronics*, 43(5): 1-5, 1996.
48. Chung, I.F., Lin, C.J., and Lin, C.T. A GA-based fuzzy adaptive learning control network, *Fuzzy Sets and Systems*, 112: 64-84, 2000.
49. Coello, C.A.C. An Empirical study of evolutionary techniques for multiobjective optimization in engineering design, *Ph.D. Thesis submitted to the Tulane University, New Orleans, LA*., 1996.
50. Coello, C.A.C., and Christiansen, A.D. Two new methods for multiobjective optimization, *Civil Engineering Systems*, 15(3):207-243, 1998.
51. Coello, C.A.C. A comprehensive survey of evolutionary-based multiobjective optimization techniques, knowledge and information techniques, *Knowledge and Information Systems*, 1(3): 269-308, 1999.
52. Coello, C.A.C. Handling preferences in evolutionary multiobjective optimization: A survey, *Proceeding of the 2000 congress on evolutionary computation*, *(CEC 2000)*, 1:30-37, San Diego, USA, 2000.

53. Coello, C.A.C. List of References on evolutionary multiobjective optimization, http:/www.lania.mx/~ccoello/EMOO/EMOObib.html 2001.
54. Coello, C.A.C. Recent Trends in Evolutionary Multiobjective Optimization, in Ajith Abraham, Lakhmi Jain and Robert Goldberg (editors), *Evolutionary Multiobjective Optimization: Theoretical Advances and Applications*, pp: 7-32, Springer-Verlag, London, 2005.
55. Cristiano, J.J., White, C.C., and Liker, J.K. Application of multiattribute decision analysis to quality function deployment for target setting, *IEEE Transactions on SMC – Part C - Applications and Reviews,* 31(3): 366-382, 2001.
56. Crossley, W. A., Martin, E. T., and Fanjoy, D. W. Conceptual aircraft design environment: case study evaluation of computing architecture technologies, *AIAA Paper* 2001-5247, Oct. 2001.
57. Cvetkovic', D. Evolutionary multi-objective support systems for conceptual design, *Ph.D. Thesis submitted to the university of Plymouth*, 2000.
58. Darwen, P.J., and Yao, X. Speciation as automatic categorical modularization, *IEEE Transactions on Evolutionary Computation*, 1(2): 101.108, 1997.
59. Das, I. Nonlinear multicriteria optimization and robust optimality: a truss optimization problem, *Ph.D thesis submitted to Rice University, Houston*, TX, 1997.
60. Das, I. A preference ordering among various Pareto optimal alternatives, *Structural Optimization,* 18(1): 30–35, 1999.
61. Das, I. Robustness optimization for constrained nonlinear programming problems, *Engineering Optimization*, 32(5): 585–618, 2000.
62. Dasgupta, D., and McGregor, D.R. A more biologically motivated genetic algorithm: The model and some results, *Cybernetics and Systems: An International Journal*, 25(3): 447–469, 1994.
63. Deb, K. Multi-objective genetic algorithms: problem difficulties and construction of test problems, *Evolutionary Computation* 7(3): 205-230, 1999.
64. Deb, K. *Multi-objective optimization using evolutionary algorithms*. J. Wiley & Sons, Ltd., 2001.
65. Deb, K. Unveiling innovative design principles by means of multiple conflicting objectives, *Engineering Optimization,* 35(5):445–470, 2003.
66. Deb, K., Agrawal, S., Pratap, A., and Meyarivan, T.A. Fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II, In*: Proceedings of the Parallel Problem Solving from Nature VI Conference*, pp: 849–858, Paris, France, 2000.
67. Deb, K., and Chaudhuri, S. I-EMO: an interactive evolutionary multi-objective optimization tool, *KanGAL Report NO. 2005003*, 2005
68. Deb, K., and Gupta, H. Introducing robustness in multi-objective optimization. *KanGAL Report No. 2004016*, October, 2004.
69. Deb, K., and Gupta, H. Searching for robust Pareto-optimal solutions in multi-objective optimization, *Evolutionary Multi-Criterion Optimization*, volume 3410 of LNCS, pp: 150–164. Springer-Verlag, 2005.
70. Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. A Fast and elitist multiobjective genetic algorithm: NSGA–II, *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, April 2002.
71. DeJong, E.D., Watson, R.A., and Pollack, J.B. Reducing bloat and promoting diversity using multi-objective methods, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)* , pp: 11-18, Morgan Kaufmann Publishers, 2001.

72. De Jong, E.D. Representation development from Pareto-coevolution, In E. *Proceedings of the Genetic and Evolutionary Computation Conference, (GECCO-03),* pp: 265-276, Springer-Verlag, Heidelberg, 003.

73. Dellaert, F., and Beer, R.D. A *Developmental Model for the evolution of complete autonomous agents*. Maes, P. *et al.* (eds.). From Animals to Animates, 1996.

74. Diaz-Calderon, A. A composable simulation environment to support the design of mechatronic systems, *Ph.D. Thesis submitted to the Carnegie Mellon University,* Pittsburgh, Pennsylvania, 2000.

75. Dieter, E. *Engineering design: a materials and processing approach*, 2nd edition, McGraw-Hill, Inc., 1991.

76. Du, X., and Chen, W. Methodology for managing the effect of uncertainty in simulation-based design, *AIAA Journal,* 38(8): 1471-1478, 2000.

77. Duda J.W, and Jakiela M. Generation and classification of structural topologies with genetic algorithm speciation, *Journal of Mechanical Design*, 119: 127–31, 1997.

78. Eckert, C., Kelly, I., and Stacey, M. Interactive generative systems for conceptual design: An empirical perspective, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 13: 303-320, 1999.

79. Emmerich, M., Grotzner, M., Gross B., and Schutz, M. Mixed integer evolution strategy for chemical plant optimization with simulators, evolutionary design and manufacture, *Proceedings of the fourth international conference on adaptive computing in design and manufacture*, pp: 55-68, Springer-Verlag, 2000.

80. Erbatur, F., Hasançebi, O., Tutüncü, I., and Hakan, K. Optimal design of planer and space structures with genetic algorithms, *Computers and Structures*, 75(2): 209-224, 2000.

81. Fixon, S.K. Assessing product architecture costing: product life cycles, allocation rules, and cost models, *Proceedings of DETC'04 ASME 2004 Design Engineering Technical Conferences and Computers and Information in Engineering Conference,* pp: 1-12, Salt Lake City, Utah, 2004.

82. Fisher, M.L., Ramdas, K., and Ulrich, K.T. Component sharing in the management of product variety: a study of automotive braking systems, *Management Science*, 45(3): 297-315, 1999.

83. Ficici, S.G., and Pollack, J.B. *A game-theoretic approach to the simple coevolutionary algorithm*, In M. Schoenauer *et al*., editor, *Proceedings of Parallel Problem Solving from Nature, (PPSN,VI, 2000)*, volume 1917 of LNCS, pp: 467-476, Springer-Verlag, 2000.

84. Ficici, S.G., and Pollack J.B. A game-theoretic memory mechanism for coevolution. In E. Cant_u-Paz et al., editor, Genetic and Evolutionary Computation GECCO-2003, volume 2723 of LNCS, pp: 286-297, Springer-Verlag, 2003.

85. Fleming, P.J., and Purshouse, R.C. Evolutionary algorithms in control systems engineering: a survey, *Control Engineering Practice*, 10: 1223–1241, 2002.

86. Fonseca, C.M, Fleming P.J. Genetic algorithms for multiobjective optimization: formulation, discussion and generalization. *Proceedings of the Fifth International Conference on Genetic Algorithms, (ICGA 93),*pp: 416–23, Urbana-Champaign, USA, 1993.

87. Fourman, M.P. Compaction of symbolic layout using genetic algorithm, Genetic algorithm and their applications, *Proceeding of the First International Conference on Genetic Algorithms*, pp: 141-153, Pittsburgh, USA, 1985.

88. Gembicki, F.W. Vector optimization for control with performance and parameter sensitivity indices. *PhD. thesis, Submitted to Case Western Reserve University*, Cleveland, Ohio, 1974.
89. Gero, J.S., and Kazakov, V. An exploration-based evolutionary model of a generative design process and its contributions to design process, *Microcomputers in Civil Engineering* 11: 209-216, 1996.
90. Gero, J.S., and Kazakov, V. Adaptive enlargement of state spaces in evolutionary designing, *Artificial Intelligence in Engineering Design, Analysis and Manufacturing* 14:8-31, 2000.
91. Giunchiglia, G. A semantics for abstraction *Technical Report* # T03−12−10, December, 2003.
92. Giunchiglia, F., and Walsh, T. A theory of abstraction, *Artificial Intelligence*, 57: 323-389, 1992.
93. Gobb, H.G., and Grefenstette, J.J. Genetic algorithms for tracking changing environments, *Proceedings of the Proceedings of the Fifth International Conference on Genetic Algorithms*, *(ICGA 93),* 5: 523-529, Urbana-Champaign, USA, 1993
94. Goldberg, D.E. *Genetic algorithms in search, optimization and machine learning*, Addison-Wesley, 1989.
95. Goldberg D.E. Genetic algorithm as a computational theory of conceptual design, *Proceedings of Applications of Artificial Intelligence in Engineering*, 6: 3-16, 1991.
96. Goldberg. D.E. *The Design of Innovation,* Kluwer Academic Publishers, 2002.
97. Goldberg, D. E., and Wang, L. Adaptive niching via coevolution sharing, *IlliGAL Report,* No. 97007, August 1997.
98. Gonzalez-Zugasti, J.P., Otto, K.N., and Baker, J.D. A Method for architecting product platforms, *Research in Engineering Design*, 12: 61-72, 2000.
99. Gonzalez-Zugasti, J.P., Otto, K.N., and Baker, J. D. Assessing value for platformed product family design, *Research in Engineering Design*, 13(1): 30–41, 2001.
100. Gurocak, H.B. A Genetic-Algorithm-Based method for tuning fuzzy logic controllers, *Fuzzy Sets and Systems*, 108: 39-47, 1999.
101. Haimes, Y.Y., Tarvainen K., Shima, T., and Thandanthil, H. *Hierarchical-multi-objective analysis of large-scale systems*, New York: Hemisphere, 1990.
102. Han, J.H. Optimal placement of piezoelectric sensors and actuators for vibration control of a composite plate using genetic algorithm, *Smart Materials and Structures*, 8(2): 257-267, 1999.
103. Hillermeier, C., Huster S., Marker W., and Sturm T.F. Optimization of power plant design: stochastic and adaptive solution concept, evolutionary design and manufacture, *Proceedings of the fourth international Conference on adaptive computing in design and manufacture*, pp: 3-18, Springer-Verlag, 2000.
104. Hillis, D.W. Co-evolving parasites improve simulated evolution in an optimization procedure, *Physica* D, 42: 228-234, 1990.
105. Holand, J.H. *Adaptation in neural and artificial systems*, Ann Arbor: The University of Michigan Press, 1975.
106. Hosam, K.F , Reyer J. A.,Papalambros, P.Y., and Ulsoy, A.G., On the coupling between the planet and controller optimization problems, *Proceedings of the American Control Conference*, pp: 25-27, Arlington, VA , 2001.
107. Hsu W., and Liu, B. Conceptual design: issues and challenges, *Computer Aided Design*, 32: 849-50, 2000.

108. Hu, J., Seo, K., Li, S., Rosenberg, R.C., and Goodman, E.D. Structure fitness sharing (SFS) for evolutionary design by genetic programming, *Proceedings of the Genetic and Evolutionary Computation Conference, (GECCO-2002*), pp: 780-787, Morgan Kaufmann Publishers, 2002.

109. Inoue, T., Furuhashi, T., Fujii, M. Development of nurse scheduling support system using interactive EA., in *IEEE International Conference on Systems, Man, and Cybernetics* (SMC'99), , 5: 533-537, Tokyo, Japan ,1999.

110. Ishii, K. Life-cycle engineering design, *Journal of Mechanical Design*, 117: 42-47, 1995.

111. Jacome, F.J. Design process planning and management for CAD frameworks, *A PhD. submitted to the Carnegie Mellon University*, Pittsburgh, Pennsylvania, 1993.

112. Jakob, W., Gorges-Schleuter, M., and Blume. C. Applications of genetic algorithms to task planning and learning, In R. Manner and B. Mandric, editors, *Proceedings of Parallel Problem Solving from Nature (PPSN'2)*, LNCS, pp: 291-300, 1992.

113. Jensen M. T. Reducing the run-time complexity of multi-objective EAs: The NSGA-II and other algorithms, *IEEE Transactions on Evolutionary Computation,* 7(5): 503 – 515, 2003.

114. Jiang, L., and Allada, V. Design for robustness of modular product families for current and future markets, *ASMED design Engineering Technical Conferences–Design for Manufacturing Conference*, Pittsburgh, PA, ASME, Paper No. DETC2001 /DFM-21177, 2001.

115. Jin, Y., and Sendhoff, B. Trade-off between performance and robustness: An evolutionary multiobjective approach. In Evolutionary Multi-criterion Optimization, volume 2632 of LNCS: 237–251, Springer-Vrlag, 2003.

116. Juill´e, H., and Pollack, J. B. Coevolutionary learning: a case study, *Proceedings of the 15th International Conference on Machine Learning,* pp: 251-259, Morgan Kaufmann Publishers, 1998.

117. Kasprzak, E.M., and Lewis, K.E. An approach to facilitate decision tradeoffs in Pareto solution sets, *Engineering Valuation and Cost Analysis*, 3: 173–187, 2000.

118. Kicinger, R., Arciszewski, T., and De Jong, K. A. Evolutionary computation and structural design: a survey of the state of the art. *Computers & Structures*, 83(23-24): 1943-1978, 2005.

119. Kitamura, Y., Sano, T., Namba, K., and Mizoguchi, R., A functional concept ontology and its application to automatic identification of functional structures, *Advanced Engineering Informatics*, 16(2):145-163, 2002.

120. Kitamura, Y., Sano, T., Namba, K., and Mizoguchi, R., A functional concept ontology and its application to automatic identification of functional structures, *Advanced Engineering Informatics*, 16(2):145-163, 2002.

121. Knoblock, C.A. Automatically generating Abstractions or planning, *Artificial Intelligence*, 68(2): 243-302, 1994.

122. Koch, P.N. Probabilistic design: Optimizing for six sigma quality, *AIAA 43rd AIAA/ASME/ASCE/AHS Structures, Structural Dynamics, and Materials Conference*, Paper Number AIAA–2002–1471, 2002.

123. Koza, J.R. *Genetic programming – on the programming of computers by means of neutral selection*, MIT Press, 1992.

124. Laumanns, M., Thiele, L., Deb, K., and Zitzler, E. Combining convergence and diversity in evolutionary multi-objective optimization. *Evolutionary Computation*, *10*(3): 263-282, 2002.

125. Li, H., and Azarm, S. Product design selection under uncertainty and with competitive advantage, *Journal of Mechanical Design*, 122: 411–418, 2000.

126. Lipson, H., and Shpitalni, M. Conceptual design and analysis by sketching, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 14: 391-401, 2000.

127. Lipson, H., and Pollack, J.B. Automatic design and manufacture of robotic lifeforms. *Nature* 406: 974-978, 2000.

128. Liu, X., Begg, D.W., and Fishwick, R.J. Genetic approach to optimal topology/controller design of adaptive structures, *International Journal for Numerical Methods in Engineering*, 41(5): 815-830, 1998.

129. Lotter, B. *Manufacturing Assembly Handbook*. Butterworth, Boston, 1986.

130. Lund H., Hallam J., and Lee W. Evolving robot morphology. *Proceedings of the IEEE Fourth International Conference on Evolutionary Computation*, pp: 197-202, NJ: IEEE Press, 1997.

131. Luo, P.K Besharati, B.B., and Azarm, S. Design of robust new products under variability: marketing meets design, Journal of Product innovative Management, 22: 177-192, 2005

132. Magrab, E.B. *Integrated Product and Process Design and Development: The Product Realization Process*, 1st edition, Chapt. 6, CRC Press: New York, pp: 129–136, 1997.

133. Marler, R.T., and Arora, J.S. Survey of multi-objective optimization methods for engineering, *Structural Multidisciplinary Optimization*, 26: 369–395, 2004.

134. Mattson, C.A., and Messac, A. A non-deterministic approach to concept selection using s-Pareto frontiers., *Proceedings of ASME DETC 2002, ASME 2002 Design Engineering Technical Conference and Computers and Information in Engineering conference*, Design Automation Conference , Paper No. DETC2002/DAC-34125, Montreal Canada 2002.

135. Mattson, C.A., and Messac, A. Concept selection using s-Pareto frontiers, *AIAA Journal*, 41(6): 1190-1198, 2003.

136. Mattson, C. A., and Messac, A. Pareto frontier Based concept selection under uncertainty, with visualization, *Optimization and Engineering*, 6: 85–115, 2005.

137. McKay, R.I. Fitness sharing in genetic programming, *Proceedings of Proceedings of the Genetic and Evolutionary Computation Conference*, *(GECCO-2000)*, pp: 435-442, Morgan Kaufmann, 2000.

138. Meittinen, K. M. *Non-linear Multiobjective optimization*, International Series on Operations Research and Management Science. Kluwer Academic publishers, 1999.

139. Moriarty, D.E., and Miikkulainen, R. Forming neural networks through efficient and adaptive coevolution. *Evolutionary Computation*, *5*(4): 373.399, 1998.

140. Moshaiov, A., Multi-objective design in nature and in the artificial, (Invited keynote paper), in *Proceedings of the 5th International Conference on Mechanics and Materials in Design*, (Eds S.A. Meguid and J.F. S. Gomes), July, 2006a.

141. Moshaiov, A., Multi-objective cybernetics and the concept-based approach: will they ever meet?, Presented at the PPSN 2006 workshop on Multiobjective Problem Solving from Nature, (PPSN 2006), available at: http://dbkgroup.org/knowles/MPSN3/Moshaiov-MO-cybernetics.pdf , 2006b.

142. Moshaiov, A., and Avigad, G. Multi-objective Path Planning by the Concept-based IEC Method, *Proceedings of the 2004 IEEE International Conference on Computational Cybernetics, (ICCC 2004*), Vienna, Austria, 2004.

143. Moshaiov, A., and Avigad, G. Concept-based IEC for multi-objective search with robustness to human preference uncertainty, *Proceedings of Congress on Evolutionary Computation (CEC 2006),* pp: 1893-1900, Vancouver, Canada, 2006.

144. Moshaiov A., Avigad, G. The Extended Concept-based Multi-objective Path Planning and its A-life implications, to appear in proceedings of ALIFE, 2007.

145. Mullur, A.A., Mattson, C.A., and Messac, A. Pitfalls of the typical construction of decision matrices, *AIAA 41th Aerospace Sciences Meeting and Exhibit*, Paper Number AIAA–2003–0466, 2003.

146. Naoki M., and Kita H. Genetic-Algorithms for Adaptation to Dynamic Environments—A Survey, *Proceedings SEAL 2000*, pp: 2947-2952, 2000.

147. Olvander,J. Robustness considerations in multi-objective optimal design, *Journal of Engineering Design*, 16(5): 511--523, 2005.

148. Otto, K.N. Measurement analysis of product design methods, *Research in Engineering Design* 7(2): 86–101, 1995.

149. Pagie, L., and Hogeweg, p. Evolutionary consequences of coevolving targets, *Evolutionary Computation*, 5(4):401-418, 1998.

150. Pahl, G., and Beitz, W. *Engineering design: a systematic approach*, Springer-Verlag, 1996.

151. Palli, N., Azarm, S., McCluskey, P., and Sundararajan, R. An interactive multistage inequality constraint method for multiple objectives decision making, *ASME Journal of Mechanical Design,* 120: 678–686, 1998.

152. Pareto, V. *Cours D'Economic Politique*, Volume 1 and 2 Frouge, Lausanne, 1896.

153. Pareto, V. 1906: Manuale di Economica Politica, Societa Editrice Libraria. Milan; translated into English by A.S. Schwier as Manual of Political Economy, edited by A.S. Schwier and A.N. Page, 1971. New York: A.M. Kelley

154. Parmee I. C., The development of a dual-agent strategy for efficient search across whole system engineering design hierarchies. *Proceedings of Parallel Problem Solving From Nature. (PPSN IV).* Lecture Notes in computer science 1141 Springer-Verlag pp: 523-532, 1996a.

155. Parmee I.C., The maintenance of search diversity for effective design space decomposition using cluster oriented genetic algorithms (COGAs) and multi-agent strategies (GAANT), *Proceedings of Adaptive Computing in Engineering Design and Control II*, University of Plymouth, pp: 128-138, 1996b.

156. Parmee, I.C., and Parchase, G. Integrating computational intelligence technologies with design and manufacturing team practice, *In Proceedings of Intelligent Design in Engineering Applications Symposium (IDEA'97),* pp: 95-99, Aachen, Germany, 1997.

157. Parmee, I., C., Evolutionary and adaptive strategies for efficient search across whole system engineering design hierarchies, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing Journal*, 12: 431-445, 1998.

158. Parmee, I., C., Towards the support of innovative Conceptual design through interactive designer/ evolutionary computing strategies, *Artificial intelligence for engineering design, Analysis and Manufacturing Journal*, 14: 3-16,1999.

159. Parmee, I.C., Cvetkovic, C., Watson, A. H., Bonham, C. R. Multi-objective satisfaction within an interactive evolutionary systems for ill-defined design environments, *Evolutionary Computation*, 8(2): 197-222, 2000.

160. Parmee, I.C. Poor-definition, uncertainty, and human factors- satisfying multiple objectives in real-world decision-making environments, E.Zitler *et al*. (Eds.): *EMO,* LNCS, pp: 52-66, 2001a.

161. Parmee, I.C., Cvetkovic, D., Bonham, C.R., and Packham, I.S. Introducing prototype interactive evolutionary systems for ill-defined multi-objective design environments, *Journal of Advances in Engineering Software*, 32 (6) :429 – 441, 2001b.

162. Parmee, I. C. *Evolutionary and Adaptive Computing in Engineering Design*, Springer Verlag, 2001c.

163. Parmee, I.C. Improving problem definition through interactive evolutionary computation, *Journal of Artificial Intelligence in Engineering Design, Analysis and Manufacture-Special Issue: Human-computer Interaction in Engineering Contexts* 16(3): 185-202, 2002.

164. Parmee, I.C., Abraham, J.A. User-centric evolutionary design, *International Design Conference- Design 2004*, Dubrovnik, May 2004.

165. Phadke, M.S. *Quality engineering using robust design*, Englewood Cliffs, New Jersey, Prentice Hall, 1989.

166. Pohlheim, H. Competition and Cooperation in Extended Evolutionary Algorithms, In Spector, L. (ed.): *Proceedings of the Genetic and Evolutionary Computation Conferenc*e – *(GECCO 2001),* Late Breaking Papers. San Francisco, CA: Morgan Kaufmann Publishers, 2001.

167. Pugh, S. *Creating Innovative Products Using Total Design: The Living Legacy of Stuart Pugh*, 1st ed., pp: 167–182, Addison-Wesley, 1996.

168. Rai, R., and Allada, V. Modular product family design: agent-based Pareto-optimization and quality loss function-based post-optimal analysis, *International Journal of Production Research*, 41(17), 2003.

169. Rechenberg, I. Evolutionsstrategie: *Optimierung technischer systeme nach prinzipien der biologischen evolution*, Stuttgart: Fommann-Holzbook, 1973.

170. Renner, G., Ekárt, A. Genetic algorithms in computer-aided design, *Computer-Aided Design* 35: 709–726, 2003.

171. Robertson, D., and Ulrich, K. Planning product latforms, *Sloan Management Review*, 39(4): 19–31, 1998.

172. Rosin, C. D., and Belew, R. K. New methods for competitive coevolution. *Evolutionary Computation*, *5*(1): 1.29, 1997.

173. Sacerdoti, E.D. Planning in a hierarchy of abstraction spaces, *Artificial Intelligence*, 5: 115-135, 1974.

174. Sanderson, E., and Cameron, T.M. Robust design optimization of structures through considerations of variations, *Computers and Structures*, 80: 1605-1613, 2002.

175. Sano, Y. and Kita, H. Optimization of noisy fitness functions by means of genetic algorithms using history of search with test of estimation, In *Proceedings of Congress on Evolutionary Computation, (CEC 2002*), Honolulu, Hawaii 2002.

176. Sareni, B., and Krahenbuhl, L. Fitness sharing and niching methods revisited, *IEEE Transactions on Evolutionary Computation*, 2(3): 97-106,1998.

177. Schlierkamp-Voosen, D., and Mühlenbein, H. Adaptation of population sizes by competing subpopulations, In *Proceedings of the 1996 IEEE Conference on Evolutionary Computation*, IEEE Press, pp: 330-335, Piscataway, New Jersey, USA., 1996.

178. Schnier, T., Yao, X., and Liu, P. Digital filter design using multiple Pareto fronts, *to appear*.

179. Siddique, Z., Rosen, D.W., and Wang, N. On the applicability of product variety design concepts to automotive platform commonality, *ASME Design Engineering Technical Conferences*, Atlanta, GA, ASME, DETC/DTM-5661, pp: 1998.

180. Simpson, T.W., Product Platform design and optimization: status and promise, In *Proceedings of Design Engineering Technical conferences and Computers and Information in Engineering Conference*, *(DETC’03 ASME 2003)*, Chicago, Illinois USA, 2003.

181. Simpson, T.W., Maier, J.R.A. and Mistree, F. Product platform design: method and application, *Research in Engineering Design*, 13(1): 2–22, 2001.
182. Simpson, T.W., Nanda, J., Halbe, S., Umapathy, K. and Hodge, B. Development of a framework for web-based product platform customization, *ASME Journal of Computing and Information Science in Engineering*, 3(2): 119–129, 2003.
183. Simpson , T.W., and D'Souza, B.S. Assessing variable levels of platform commonality within a product family using a multiobjective genetic algorithm, *Journal of Concurrent Engineering: Research and Applications*, 12(2): 119-129, 2004.
184. Sims, K. Evolving 3D morphology and behavior by competition. Brooks, R. & Maes, P. (eds.) Artificial Life IV: *Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems*, pp: 28-39: MIT Press, 1994.
185. Sims, K. Evolving Virtual Creatures. Computer Graphics (SIGGRAPH 94 Proceedings), pp: 15-22: ACM SIGGRAPH, 1994.
186. Smith, R.E., Forest, S., and Perelson, A.S. Searching for diverse, cooperative populations with genetic algorithms, *Evolutionary Computation*, 1(2): 127-149, 1992.
187. Sobek, D.K., Ward, A.C. Principles from TOYOTA'S set-based concurrent engineering process, *Proceedings of The 1996 ASME Design Engineering Technical Conferences and Computers in Engineering Conference,* Irvine, California, 1996.
188. Srinivas, N., and Deb, K. Multi-objective function optimization using non-dominated sorting genetic algorithms, *Evolutionary Computation* 2(3): 221-248, 1994.
189. Stagge P. Averaging efficiently in the presence of Noise, *Proceedings of Parallel Problem Solving in Nature, (PPSN 98),* pp:188-197, Amsterdam, Netherlands,1998.
190. Steuer, R. E. *Multiple Criteria Optimization, Theory Computations and Applications*, John Wiley & Sons, Inc.: New York, 1986.
191. Steven, G., Querin, O., and Xie, M. Evolutionary structural optimization (ESO) for combined topology and size optimization of discrete structures, *Computer Methods in Mechanics and Engineering*, 188(4): 743-754, 2000.
192. Sturges R.H. O'Shaughnessy K., and Reed R.G. A Systematic approach to conceptual design, *Concurrent Engineering: Research and Applications*, 1: 93-105, 1993.
193. Suh, N. P. *The Principles of Design*, Oxford University Press: New York, 1990.
194. Sung-Bau, C. Towards creative evolutionary systems with interactive genetic algorithm, *Applied Intelligence*, 16: 129-138, 2002.
195. Savransky S.D., Engineering of creativity: introduction to TRIZ methodology of inventive problem solving, CRC Press, 2000.
196. Taguchi, G., Taguchi, S., and Tugulum, R., Taguchi robust engineering: worlds' best practice for achieving competative advantage in the new millennium, McGraw –Hill, 1999.
197. Tan, K.C., Lee, T.H., and Khor, E.F., Evolutionary algorithms for multi-objective optimization: performance assessments and comparisons, *Artificial Intelligence Review* 17: 253-290 2002.
198. Takagi, H. *Interactive Evolutionary Computation*, World Scientific, October pp: 16-20, 1998.
199. Takagi, H. Interactive Evolutionary Computation: Fusion of the Capacities of EC optimization and human evaluation, *Proceedings of the IEEE,* 89(9): 1275-1296, 2001.
200. Tappeta, R.V., Renaud, J.E., Messac, A., and Sundararaj, G.J. Interactive physical programming: tradeoff analysis and decision making in multicriteria optimization, *AIAA Journal,* 38(5): 917–926, 2000.

201. Tenenberg, J.D. Abstraction in planning., *Ph.D. thesis, submitted to the University of Rochester*, 1988.
202. Thompson, A. On the automatic design of robust electronics through artificial evolution. In Sipper, M., Mange, D., and Peres-Urike, A., editors, *International Conference on Evolvable Systems*, pp: 13 – 24, 1998, Springer.
203. Tiihonen J., Lehthonen T., Soininen T., Pulkkinen A., Sulonen R., and Riitahuhta A. Modeling configurable product families. In Tichem M., Storm T., Anderson M. and Duffy A.H.B editors, *Proceedings of the 4th WDK Workshop on Product Structuring*, WDK Product Structuring Workshop Series, Technical University of Delft, 1998.
204. Tjornfelt-Jensen, M., and Hansen, T. K. Robust solutions to job shop problems, *proceedings of the Congress on Evolutionary Computation*, *(CEC 1999)*, 2:1138– 1144, Washington, DC, USA, 1999.
205. Ulrich, K. T., and Eppinger, S. D. *Product Design and Development*, 2nd ed., McGraw–Hill, New York, pp: 138–159, 2000.
206. Ullman, D.G. *The Mechanical Design Process*, McGraw–Hill, New York, pp: 168–184, 1992.
207. Van Veldhuizen, D., and Lamont, G.B. Multiobjective evolutionary algorithm test suites. *ACM*, 1998.
208. Van Veldhuizen, D. A. Multiobjective evolutionary algorithms: Classifications, analysis, and new innovations, *Ph.D submitted to Air Force Institute of Technology*, 1999.
209. Van Veldhuizen, D.A., and Lamont, G.B. Multiobjective evolutionary algorithms: analyzing the State-of-the-Art, *Evolutionary Computation*, 8(2): 125-147, 2000.
210. Van Veldhuizen, D. A., Zydallis, J. B., and Lamont, G.B. Considerations in engineering parallel multiobjective evolutionary algorithms, *IEEE Transactions on Evolutionary Computation*, 7(2), pp 2003.
211. Ventrella, J. *Animated artificial life*. Chapt. 3 in Heudin, J. C. (ed.) Virtual Worlds: Synthetic Universes, Digital Life and Complexity: Perseus Books, 1999.
212. Vlachos, C., Williams, D., and Gomm, J.B. Genetic approach to decentralized PI controller tuning for multivariable process, *IEE Proceedings –Control theory and Applications*,146 (1):58-64, 1999.
213. Watson, R. A., and Pollack, J. B. Symbiotic combination as an alternative to sexual recombination in genetic algorithms. In M. Schoenauer *et al*., editor, *proceedings of Parallel Problem Solving from Nature*, PPSN-VI, volume 1917 of LNCS: pp: 425-434 Springer-Verlag, 2000.
214. Wang, J. Ranking engineering design concepts using a fuzzy outranking preference model, *Fuzzy Sets and Systems*, 119: 161–170, 2001.
215. Wang Q., Rao M. and Zhou, J. *Intelligent systems for conceptual design of mechanical products*, In: Mital A., Anand S., editors. Handbook of expert systems applications in manufacturing: structures and rules, New York: Chapman & Hall, 1994.
216. Wang, Q., and Stengel, R.F. Robust control of nonlinear systems with parametric uncertainty, *Automatica*, 38: 1591-1599, 2002.
217. Wang, L., Shen, W., Xie, H., Neelamkavil, J. and Pardasani, A. Collaborative conceptual design state of the art and future trends, *Computer-Aided Design*, 24: 981-996, 2002.
218. Wiegand, R.P. An analysis of cooperative coevolutionary algorithms, *PhD. thesis, submitted to George Mason University*, Fairfax, Virginia, 2003.

219. Wiesmann, D., Hammel, U., and Bäck, T. Robust design of multilayer optical coatings by means of evolutionary algorithms, *IEEE Transactions on Evolutionary Computation*, 2(4):162–167, 1998.

220. Werfel, J., Mitchell, M., and Crutchfield, J.P. Resource sharing and coevolution in evolving cellular automata [Letter], *IEEE Transactions on Evolutionary Computation*, *4*(4): 388-401, 2000.

221. Youcef-Toumi, K., Modeling design and control integration: a necessary step in Mechatronics, *IEEE/ASME Transactions on Mechatronics*, 1: 78-86, 1996.

222. Zhu, Y., Qiu, J., and Tani, J. Simultaneous optimization of a two – link flexible robot arm, *Journal of Robotic Systems*, 18(1): 29-38, 2001.

223. Zhu, Y., Qiu, J., Du, H., and Tani, J. Simultaneous structural-control optimization of a coupled structural-acoustic enclosure, *Journal of Intelligent Material Systems and Structures*, 14(4-5): 287-296, 2003.

224. Zitzler, E. Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications. *Ph.D. thesis* submitted to the Swiss Federal Institute of Technology (ETH) Zurich, Switzerland, 1999.

225. Zitzler, E., Deb, K., and Thiele, L. Comparison of multiobjective evolutionary algorithms: empirical results, *Evolutionary Computation* 8(2): 173-195, 2000.

226. Zitzler, E., Laumanns, M., and Thiele, L. SPEA2: Improving the strength Pareto evolutionary algorithm. *Technical report 103*, Computer Engineering and Networks Laberatory (TIK), Swiss Federal Institute of Technology (ETH) Zurich, Gloriastrasse 35, CH-8092 Zurich, Switzerland, May 2001.

227. Zitzler, E., and Thiele, L. Multiobjective optimization using evolutionary algorithms – A comparative case study, *Parallel Problem Solving from Nature (PPSN 98),* V: 292-301, 1998.

228. Ziv-Av, A. and Reich, Y., SOS - subjective objective system for generating optimal product concepts, *Design Studies*, 26:509-533, 2005.

Appendix

The following outlines the articles which have been published as a part of the work towards this thesis. A simultaneous evolution of solutions, which are associated with SBCs, has been initially introduced in ***Avigad et al. (2003)***. In that work the concepts involved both mechanical SCs as well as control SCs. The search for concepts has been performed to find concepts for solutions with performances around a target goal. This approach has been extended in ***Avigad et al. (2004)***, to include interactivity. In the latter study a progressive concept-based goal attainment approach has been taken. The introduced interactivity includes two types. The first is the movement of the target goal from one location to another, and the second, which is the major one, is the incorporation of preferences towards conceptual ideas. These preferences that have been represented by weights, allow a simultaneous search for solutions of preferred concepts around a goal.

In both studies of ***Avigad et al. (2003, 2004)***, the concepts were evolved simultaneously, by way of their sets of solutions. In the simultaneous evolution of concepts, which has been introduced for the first time in these works, all concepts and preliminary designs have been encoded using a structured individual, which has been termed 'Compound Individual' (CI). Decoding such a CI results in a decoded pointer, which points to a concept and to its design variables as well as to its related model (objective functions). A CI may be decoded to any solution of the concepts' design spaces. In addition to the use of CIs, 'concept-sharing' has been introduced. Concept sharing enhances a pressure towards the representation of all concepts (which have feasible solutions around the goal). This is done by sharing the computational resource of the entire population between successful concepts based on their solutions' optimality. The CI allows searching the design space innovatively by searching not just preliminary designs but also concepts by way of their SCs.

In a later study by ***Avigad et al. (2005a)***, a simultaneous Pareto-directed, rather than a progressive goal approach has been introduced. Ensuring the simultaneous survival of successful concepts and the spread of their solutions on the front has been achieved by using in-front concept sharing and in-concept front niching, respectively. In addition, a unique robust sorting has been introduced. It involves the incorporation of worst case sorting within an EC algorithm.

Additional investigations, which are related to this thesis, have widened the use of interactivity to treat other issues of the conceptual design stage within MOPs, including delayed decisions (***Avigad et. al., 2005b***) and uncertainty and variability of market demands

(***Avigad et. al., 2005c***). In these latter works some unique EC approaches were introduced including worst case sorting, robustness-based evolution and a simultaneous evolution of concepts based on concepts' sets trajectories on the objective axis.

It is important to note that in all the above studies, the population is not separated into distinct sub-populations as done in this thesis. In the cited studies, concepts are formed or disappeared, during the evolution, based on a code, which codes the concepts.

The approaches presented in the above studies, and in this thesis, possess a generic nature and they might be applied to other fields of interest. For example, in ***Moshaiov and Avigad (2004, 2006),*** the applicability of the ideas to path planning has been demonstrated. A further extension of their work, to include near Pareto solutions, may be found in Moshaiov and Avigad, 2007. Finally, it is noted that following this thesis the author has published additional studies, which are not directly associated with the theme of the thesis. These studies include: Avigad 2007a, Avigad 2007b, Avigad and Deb 2007.