

Applying Genetic Algorithms to Multi-objective Land-Use Planning

Keith B. Matthews

A thesis submitted in partial fulfilment
of the requirements of
The Robert Gordon University
for the degree of Doctor of Philosophy

© October 2001, Keith B. Matthews

Collaborating establishment: Macaulay Land Use Research Institute,
Craigiebuckler, Aberdeen

Abstract

This thesis explores the potential of applying genetic algorithms (GAs) to multi-objective land-use planning. Rural land managers, in the UK and elsewhere in Europe, are faced with an increasingly complex decision-making environment where a varied mix of goals have to be achieved.

To address this problem, GA-based land-use planning tools have been developed that interface with a decision support system. The thesis first evaluates the options for representing the land-use planning problem within a GA framework. Two genotype representations are proposed: a genotype mapping land-use directly to land parcels (Land-Block), and a representation making allocations indirectly via a greedy algorithm (P&P). The P&P representation requires novel breeding operators and post-evaluation processing, to identify and remove duplicate or defective genotypes.

The performance of the two GAs' was evaluated for a single-objective land-use planning problem. Since both GAs found acceptable solutions, two multi-objective genetic algorithms (mGAs) were implemented based on the proposed representations. The goal of these mGAs was to search for a set of solutions defining the structure of the relationship between two or more objectives. To achieve this goal, modifications are required to the calculation of selection-fitness and the implementation of parent selection. The mGAs were tested for a problem with two conflicting objectives, with the Land-Block mGA found to have superior performance.

Given the non-standard nature of the GAs, it was necessary to investigate their parameterisation, in particular the balance between the GA-operators over the course of the run. Two online-optimisation approaches were tested. The use of online-optimisation was successful in significantly improving the efficiency of the GAs.

To investigate the usefulness of the GAs and mGAs for decision support, two further investigations were carried out. In the first the scalability of the two representations was tested for a single-objective problem. The P&P GA outperformed the Land-Block, since the complexity of the P&P genotype depends on the number of land-uses present in the optimal solution, not on the number of land parcels to be allocated a land use. Second, the sets of solutions found by the mGAs were compared with allocations collected, using soft-systems methods, from professional land-managers with a range of backgrounds. The mGAs found solutions as good as those proposed by the land managers. Additional factors that needed to be accounted for by the mGA were identified, but the mGA solutions were seen by the land-managers as forming a useful basis for practical land-use planning.

Acknowledgements

I wish to thank my director of studies Professor Susan Craw, for her well-considered advice and constructive criticism over the course of the research. I hope that her standards have now become mine. I would also like to thank the other members of my supervisory team, Dr. Stewart Elder and Mr. Iain MacKenzie, of the Robert Gordon University, for their specialist assistance and Mr. Alan Sibbald, of the Macaulay Land Use Research Institute, for his encouragement, diplomacy and for providing the definitive answer to the “land use” or “land-use” question. Thanks are also due to the land-managers who contributed enthusiastically to the evaluation of the tools developed.

The research was funded by grant-in-aid from the Scottish Executive Rural Affairs Department. Financial and logistical assistance for conducting the evaluation workshop was provided by Grampian Enterprise Limited.

Finally I would like to thank my parents for “occasionally” asking when my PhD would be finished.

Contents

1	Introduction	1
1.1	Land-Use Planning	1
1.1.1	Strategic whole-farm planning	1
1.1.2	Changing context for strategic whole-farm planning	3
1.1.3	Test application: Hartwood Research Station	3
1.2	Objectives	6
1.3	Land Use Decision Support System	7
1.4	Software Platform	9
1.5	Approach	10
2	Previous Research	12
2.1	Existing Approaches to Land-Use Planning	12
2.1.1	Linear programming	12
2.1.2	Heuristic search methods	14
	Simulated-annealing	15
	Tabu-search	16
2.1.3	Multi-criteria decision making	17
	Compensatory MCDM	18
	Non-compensatory MCDM	18
2.1.4	MCDM applications in land-use planning	19
2.2	GA Applications in Similar Domains	20
2.2.1	Static parameter optimisation	20
2.2.2	Dynamic algorithm control	20
2.2.3	Direct analysis	21

2.2.4	Summary of GA applications in similar domains	22
2.3	Genetic Algorithms	23
2.3.1	Representation	24
2.3.2	Replacement	24
2.3.3	Operators	25
2.3.4	Parameterisation	26
2.4	Multi-objective GAs	26
2.4.1	Pareto-optimal sets	27
2.4.2	Pareto-based selection	28
	Niche-based fitness sharing	29
	Sharing spaces	30
	Sharing Parameterisation	31
2.4.3	Alternative mGA approaches	32
	Pareto tournament	32
	Strength Pareto Evolutionary Algorithm	32
	Pareto Archived Evolutionary Strategy	33
2.4.4	Replacement	34
2.4.5	Mating restrictions	34
2.4.6	Summary of mGA themes	35
2.5	Conclusions	35
3	Representation of the Problem	37
3.1	Introduction	37
3.2	Spatial and Non-spatial Representations	37
3.2.1	Grid representation	37
3.2.2	Quad-tree representation	39
3.2.3	Polygon representation	40
3.3	Land-Block Representation	41
3.4	Percentage and Priority Representation	43
3.4.1	The DSS land-allocation greedy-algorithm	44
3.5	Summary	45

4	Single-Objective GAs	46
4.1	Introduction	46
4.2	Generic SOGA Features	48
4.2.1	Reproduction	48
4.2.2	Replacement	48
	Uniqueness testing	48
	Fitness Evaluation	49
	Replacement of least fit	50
4.2.3	Selection fitness calculation	50
4.3	Representation-Specific Features	51
4.3.1	Population initialisation	51
	Land-Block population initialisation	51
	P&P population initialisation	52
4.3.2	Non-fitness feedback to the P&P SOGA from the DSS	52
	Causes of functionally identical genotypes	52
	Detection and elimination of functionally identical genotypes	54
4.4	Operators	55
4.4.1	Operators used by both representations	55
4.4.2	Land-Block specific operators	56
4.4.3	P&P specific operators	56
	Binary Operators	56
	Unary Operators	57
4.5	Initial SOGA Evaluation	58
4.5.1	Evaluation metrics	58
4.5.2	Results from initial testing	59
	Quality of solutions for the Land-Block and P&P SOGAs	60
	Efficiency of learning for the Land-Block and P&P SOGAS	61
4.6	Summary and Conclusions	61
5	Multi-Objective GAs	63
5.1	Introduction	63
5.2	The mGA Goals	64

5.3	mGA Implementation	65
5.4	Determining mGA selection-fitness	65
5.4.1	Dominance ranking	65
5.4.2	Fitness sharing	68
5.5	Mating restrictions	71
5.5.1	Mating restriction objectives	71
5.5.2	Mating restriction implementation	72
5.6	Initialisation	73
5.6.1	Population sizing	73
5.6.2	Population doping	74
5.7	Features in common with SOGA	74
5.8	Initial mGA Evaluation	75
5.8.1	mGA evaluation metrics	76
5.8.2	Results and discussion of initial testing	77
	Parametrisation used in the mGA testing	77
	Effectiveness of mating restrictions	77
	Land-Block and P&P mGAs' performance for the individual objectives	79
	Multi-objective performance	81
	Efficiency of mGA learning	81
5.9	Summary and Conclusions	83
6	Parameterisation	85
6.1	Introduction	85
6.2	Evolutionary Operation	85
6.2.1	Conventional Evolutionary Operation	85
6.2.2	EVOP-GA	86
	Running the EVOP cycle	87
	Analysing the EVOP cycle	88
	EVOP-LB testing	91
	EVOP-LB results	92
6.2.3	EVOP summary and conclusions	93
6.3	Online-Parameterisation GA	93

6.3.1	The OP-GA algorithm	94
	Operator application	95
	Operator adaptation	96
	Operator Initialisation	98
6.3.2	Land-Block OP-GA	99
	Land-Block OP-GA testing	99
	Land-Block OP-GA results	100
6.3.3	P&P OP-GA	103
	P&P OP-GA initialisation	103
	P&P OP-GA testing	103
	P&P OP-GA results	104
6.3.4	OP-GA summary and conclusions	107
6.4	Parameterisation Conclusions	107
7	Evaluation	108
7.1	SOGA Scaling	108
	7.1.1 Materials and methods	108
	7.1.2 Scaling results	109
	Quality of solution	109
	Absolute efficiency	109
	Relative efficiency	110
	7.1.3 Scaling conclusions	111
7.2	mGA - Expert Evaluation	111
	7.2.1 Methods and materials	112
	7.2.2 Assumptions and metrics	113
	7.2.3 Delegate and subgroup allocations	115
	7.2.4 Comparing the mGA and expert allocations	117
	7.2.5 Evaluating the choice of planning objectives	118
	7.2.6 Effectiveness of subgroups in finding compromise allocations	118
	7.2.7 Expert evaluation conclusions	119
8	Conclusions	121
8.1	Contributions to Planning Knowledge	121

8.1.1	Effective methodology	121
8.1.2	Evaluation approach	122
8.2	Contributions to Genetic Algorithm Knowledge	123
8.2.1	Representation	123
8.2.2	Operators and Non-fitness feedback	124
8.2.3	Multi-objective GAs	124
8.2.4	Parameterisation	125
8.3	Future research	126
8.3.1	Improvements	126
8.3.2	Additional applications	127
8.3.3	Integration with AI planners and schedulers	127
8.4	Summary of outcomes	128
A	GA Class Hierarchies	138
A.1	GA	138
A.2	Population	138
A.3	Genotype	140
A.4	Gene	141
A.5	Operator	141
B	Workshop Materials	143
C	Published Papers	150

List of Figures

1.1	Spatio-Temporal Land-Use Planning Scales	2
1.2	Hartwood Research Station	4
1.3	Sub-components of the DSS	7
2.1	A simulated annealing algorithm.	15
2.2	A tabu search algorithm.	17
2.3	Goldberg's and Fonseca and Fleming's Pareto-ranking schemes.	29
2.4	Niche size and its relationship to population size	32
2.5	SPEA dominance ranking	33
3.1	Grid, Quad-tree and Polygon Representation	38
3.2	Grid-based toroidal two-point crossover and mutation	39
3.3	Quad-tree example.	40
3.4	Land Block Representation	42
3.5	An example of the P&P Representation	44
3.6	The greedy allocation algorithm	45
4.1	Land-Block and P&P SOGA Structure	47
4.2	Detail of the replacement phase (from Figure 4.1)	49
4.3	P&P Non-Fitness Feedback Mechanisms	53
4.4	Fitness Evaluation and Non-Fitness Feedback	54
4.5	PRR Operator	56
4.6	Land-Block and P&P GA performance graphs	60
5.1	Two-objective maximisation example application	65
5.2	mGA Structure	66

5.3	Pareto-optimal ranking example	67
5.4	Genotypes distributed within niches	68
5.5	Niche-based sharing example	70
5.6	Mating restrictions example	71
5.7	Counts of genotypes for Land Block and P&P mGAs	80
5.8	Population plots for Land Block and P&P mGAs	82
6.1	EVOP cycles for one and two parameters	86
6.2	EVOP-GA algorithm	87
6.3	Running the EVOP cycle	88
6.4	EVOP cases	90
6.5	Analysing the EVOP cycle	91
6.6	Averaged EVOP Operator Probabilities	92
6.7	OP-GA algorithm	94
6.8	OP-GA operator application	95
6.9	OP-GA operator adaptation	96
6.10	OP-GA example	98
6.11	OP-GA operator initialisation	99
6.12	Average Land-Block operator probability trajectories	101
6.13	Land-Block operator probability trajectories for all runs	102
6.14	Average P&P operator probability trajectories	106
7.1	Example of the allocation maps prepared by delegates	114
7.2	mGA and Land Manager Allocations	116
7.3	Individuals contributing to the subgroup allocations	119
8.1	Future development of the land-use planning tools	128
A.1	GA class hierarchy	139
A.2	Population class hierarchy	139
A.3	Genotype class hierarchy	140
A.4	Gene class hierarchy	141
A.5	Operator class hierarchy	142

List of Tables

1.1	Land uses for the Hartwood Research Station application.	5
4.1	Rank-based linear normalisation example	50
4.2	Operator sets for each representation	55
4.3	Land-Block and P&P GA performance	61
5.1	Selection fitness using dominance rank for population in Figure 5.3	67
5.2	Fitness sharing, for population in Figure 5.5	70
5.3	∞ - <i>norm</i> distances, for genotype pairs in Figure 5.5	70
5.4	Mating-fitness calculation for population in Figure 5.5	72
5.5	Population sizes for combinations of niche size and number of objectives	73
5.6	The effect of mating restrictions on mGA performance	78
5.7	Land-Block and P&P mGA performance	79
6.1	Relative performance of fixed parameterisation Land-Block GAs	91
6.2	Relative performance EVOP-LB vs. LB-Best-Fixed	92
6.3	Land-Block initial operator probabilities	100
6.4	Relative performance LB OP-GA to LB-Best-Fixed	100
6.5	P&P initial operator probabilities	103
6.6	Relative performance of fixed parameterisation P&P GAs	104
6.7	Relative performance OP-PP to PP-Best-Fixed	105
7.1	Land-Block and P&P GA performance for scaling test problems	110
7.2	Delegates by subgroup	112
7.3	Current and Delegate Allocations (ordered by NPV)	116

Chapter 1

Introduction

1.1 Land-Use Planning

The aim of the thesis is to investigate how useful genetic algorithms (GAs) are as a tool to assist land-managers in creating plans to achieve multiple objectives. Most land-use decision support systems focus on the tactical or operational aspects of land management, paraphrased as “how to get the job done”, rather than “which job to do” (Jones, El-Swaify, Graham, Stonehouse & Whitehouse 1996).

This thesis addresses the application of GAs to multi-objective land-use planning. The form of planning conducted is *marketing planning* which investigates the *goal state* that will best meet the objective(s) of the decision maker (Barták 1999). This contrasts with classical AI planning that determines the *intermediate states* required to transform from current to goal state and with scheduling which allocates the *resources* required. A continuum exists from marketing planning through AI planning to scheduling with each stage producing plans with increasing specificity and resolution.

1.1.1 Strategic whole-farm planning

Strategic whole-farm planning is the application domain chosen for the investigation. Figure 1.1 shows how this domain fits with several other land-use planning domains characterised by their spatial and temporal scales. Spatial scales range from individual plants and animals to global ecologies. Temporal scales range from hour to decade. Strategic whole-farm planning is concerned with determining the best mix and spatial pattern of

land uses for a single land management unit. The temporal scale is typically focused on year-to-year plans. Within-year considerations may have to be taken into account, particularly for the marketing of produce. While the planning horizon for perennial crops such as forestry may be anything up to a century. The information on which the strategic whole-farm plans are based is typically one order more highly resolved, with individual fields represented and information on monthly production used.

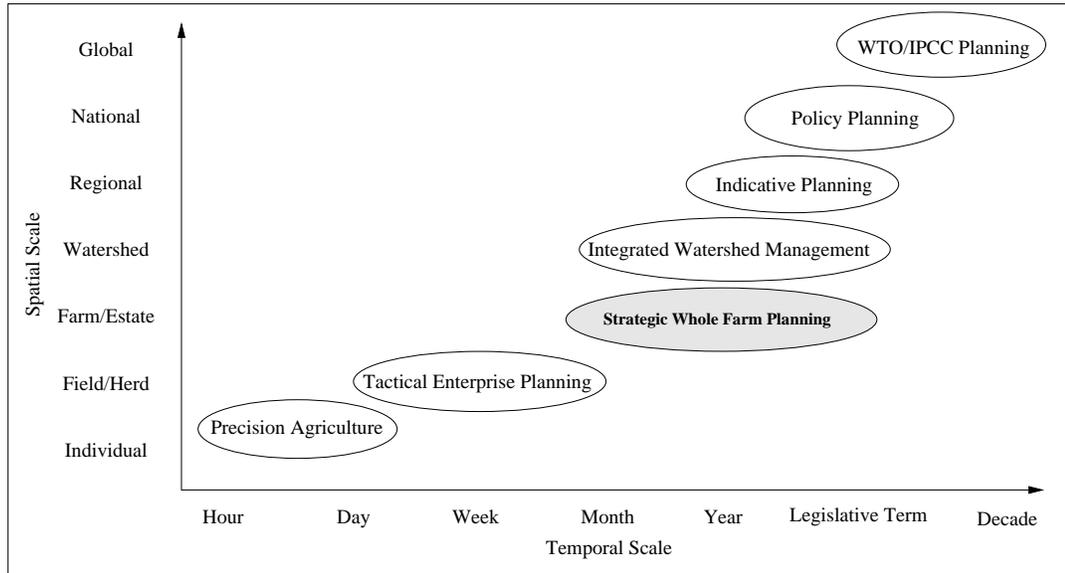


Figure 1.1: Spatio-Temporal Land-Use Planning Scales

Whole-farm planning was chosen as it is the largest spatial unit where it is possible to identify a single decision maker. The decisions made at this level have a crucial influence on the form of the rural landscape. Below this level *tactical enterprise planning* is concerned with planning individual land uses and *precision agriculture* with optimising the schedule of management interventions. At smaller spatio-temporal scales, for example *watersheds* and above, it is the interactions between multiple decision makers that becomes the key to understanding the pattern of land use. The regional, national and global scale land-use planning decisions are also important as they set the context within which strategic whole-farm planning takes place. It is changes in this context that has meant multi-objective planning is increasingly relevant.

1.1.2 Changing context for strategic whole-farm planning

It is possible to identify several factors that result in land-managers increasingly having to undertake multi-objective land-use planning tasks.

- Existing agricultural production has to be achieved within narrowing environmental and social limits. Examples include statutory requirements to reduce pollution or to increase animal welfare standards.
- Alternative crops and land-use systems are proposed, based on scientific research. Such systems may be more complex, and often lie outwith the experience of individual land-managers or their peer group. The performance of novel systems needs to be transparently evaluated against existing systems.
- Public awareness of land-use planning is increasing, particularly regarding land-use change. The general public also expects to influence the decision-making process since significant sums of public money are invested in rural land use via grants and subsidies.
- Land ownership changes, with trusts and conservation bodies becoming increasingly significant land owners in the UK, means that more diverse management goals are being pursued with social and environmental regeneration goals balanced against financial returns.

This demand for multi-objective land-use planning requires sufficient information to be provided to enable the land-manager to explore the options and impacts of alternative land-use strategies and the structure of the trade-offs between the various objectives. Exploring such trade-offs is a complex problem especially when spatio-temporal interactions between land uses require to be addressed.

1.1.3 Test application: Hartwood Research Station

The application chosen for the testing of the land-use planning GAs was the Hartwood Research Station in Lanarkshire Scotland. The farm is 300ha in area and divided into 95 fields, see Figure 1.2. The farm is located in wet, exposed uplands with significant limitations imposed by both climate (low winter temperatures), and soils (poor drainage),

on the range and productivity of land-uses. Environmentally, the area has been substantially degraded, including a period of open-cast coal-mining but has significant potential for environmental improvement, building on existing remnant features. The importance of land management for providing employment is significant in an area of high unemployment. The research station, while managed for research purposes, is typical of a class of farms on the margins of financial viability but with significant environmental and social potential. This made it an ideal application for the GA-based land-use planning tools, offering the opportunity to explore the range of allocation options available to the land manager to improve performance for any of the impact dimensions. Using a real farm, even though one not managed on a purely commercial basis, also meant that it was easier for land-management experts to evaluate the outputs from the land-use planning tools for their practical utility as the basis for decision making.

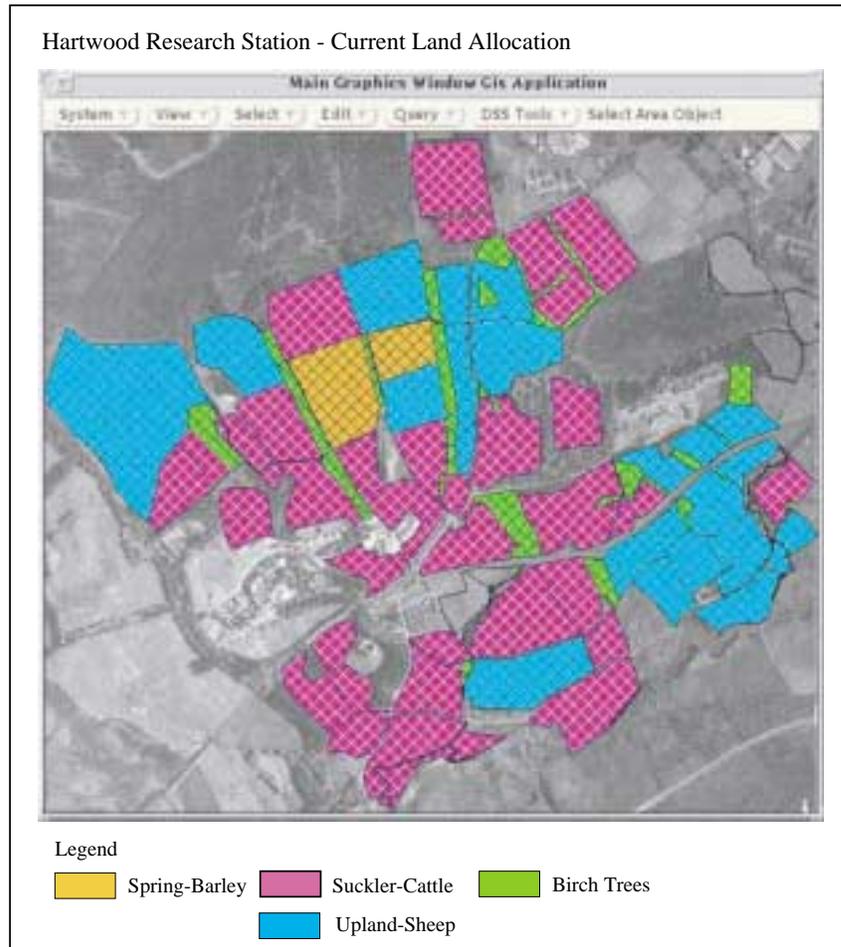


Figure 1.2: Hartwood Research Station

The ten land uses available for allocation by the land-use planning tools were those typically encountered in the uplands of the UK. Table 1.1 lists the land uses.

Table 1.1: Land uses for the Hartwood Research Station application.

Arable	Livestock	Woodland
Spring-barley	Suckler-cattle Upland-sheep	Alder Beech Birch Sycamore Wild-cherry Scots-pine Sitka-spruce

Spring-barley is an example of an arable crop; it has significant suitability limitations but potentially high financial returns. If the crop is to be included then it must, in general, be allocated to the most productive land in the management unit. Upland-sheep and suckler-cattle provide examples of the dominant, livestock-based, land uses of the uplands. Both can be allocated across a wide range of conditions. Cattle, at the prices, costs and levels of subsidies used, achieve the better financial returns. The remaining land uses are five broad-leaved and two conifer-tree species. These land uses have significant environmental benefits but their financial returns are both lower and delayed (up to 100 years for Beech trees). Broad-leaved species in general have stricter suitability criteria and are less productive than the conifers, but have a higher financial value per unit produced, based on higher timber quality.

The fitness metrics used in the testing of the land-use planning GAs were chosen with the expectation that they would ultimately be used within a multi-objective analysis. Therefore the metrics had firstly to be non-commensurable, that is, it should not be possible to combine the metrics into a single measure. Secondly, the metrics should be in conflict so that the structure of the trade-off between the objectives could be investigated. Finally there was the practical concern that any metrics used should be able to be computed at an acceptable speed to be used within the iterative process of GA evolution. The two metrics chosen were financial and environmental.

- The financial metric was the *Net Present Value* (NPV) of the farm (Boehlje & Eidman 1984). This is calculated from the gross margins per hectare, including

available grants and subsidies, discounted at a fixed rate (3%), over the period of a typical forestry rotation (60 years). NPV is used to enable comparison of the performance of land uses making annual returns, such as livestock systems, with those making intermittent returns over an extended period, such as woodland.

- The environmental metric used is the Shannon-Weiner Index (SW) (Forman & Godron 1986). The SW index is maximised when the largest number of available land uses are present in equally sized areas. This metric was chosen because it has an optimum value that could be predicted *a priori* and because it was known to conflict, in the case of Hartwood, with the financially optimum land-use pattern, a near-mono-culture of cattle.

1.2 Objectives

To successfully tackle multi-objective land-use planning for the strategic whole-farm application it is possible to identify several essential features of the planning tools.

- A *robust and efficient* search and optimisation algorithm is needed since the search space for a spatially explicit land-use planning application is large and complex. The algorithm should scale sufficiently well to cope with real-world applications.
- The land-use planner must form part of a *flexible problem solving environment*. It should allow the modification of evaluations in the light of experience by adding new parameters or changing the assumptions embodied within the land-use evaluation (Densham 1991, Moon, Jeck & Selby 1996). The objective is to support the process of decision making not to provide one “perfect” answer that would eliminate the decision maker from the process (Jones et al. 1996).
- When there are *multiple, non-commensurable objectives* the land-use planning tool should allow the investigation of the structure of the *trade-off* between objectives to enable the best compromise to be selected (El-Swaify & Yakowitz 1998).
- Given the possibility of *spatially explicit* land-use plans the land-use planning tool should be closely integrated with a geographic information system. This will permit

the most effective communication of the land-use plans as maps (Zhu, Aspinall & Healey 1996).

- The land-use planning tool must be integrated with an appropriately responsive simulation model of the system being analysed.

1.3 Land Use Decision Support System

The GA-based land-use planning tools implemented in the thesis are interfaced with a land-use decision support system (DSS). The DSS provides a framework within which the consequences of alternative land-use strategies may be evaluated. The sub-components of the DSS are illustrated in Figure 1.3, and their functionality in relation to the GA-based land-use planning tools is outlined below.

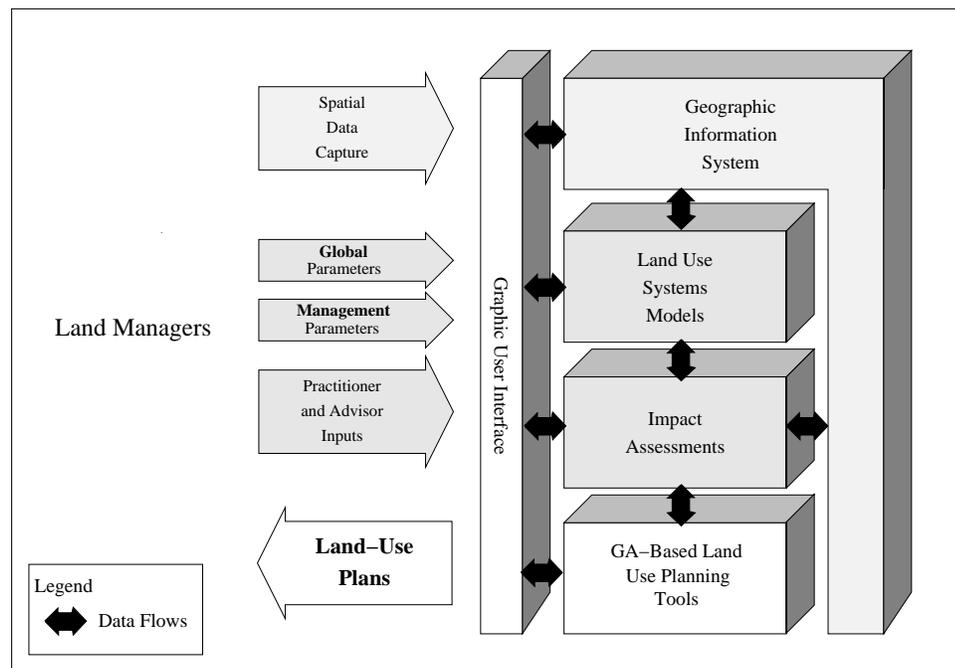


Figure 1.3: Sub-components of the DSS

- The *geographic information system* (GIS) provides all the biophysical information for the DSS, this includes climate, soils and topographic data and information on the farm infrastructure. This information is derived from ground or airborne survey and from published digital sources. The GIS component of the DSS is implemented

in the object-oriented Smallworld GIS on a Sun/Solaris platform. The GIS is also the principal means of visualising, as maps, the allocations proposed by the land-use planning tools. The GIS can bring in a wealth of contextual information, such as aerial photographs, as part of the visualisation process and this greatly aids the interpretation of the output from the planning tools. It is also possible, via the GIS interface, to interrogate sub-components of the allocation to query or override allocations. This allows a significant amount of customisation of allocations, enabling users to engage in further “what-if” style analysis using the GA allocation as a starting point.

- The information from the GIS is passed to a series of *land-use systems models*. These, along with all the non-GIS components of the DSS, are implemented in G2, a knowledge-based systems development environment, also on a Sun/Solaris platform. The land-use systems models each represent an individual land use and share the same generic functionality. The land-use modules first assess the biophysical *suitability* of a land parcel. The exact nature of the parcel considered is elaborated in Chapter 3. If the parcel is suitable a module then proceeds to estimate its *productivity* under a given set of *management parameters*, defined by the land manager. Finally each land-use model makes an assessment of the financial *profitability*. This is the gross margin of the system, for a given set of costs and market prices, termed the *global parameters*. The suitability, productivity and profitability analyses, conducted at the level of the land parcel, provide the fundamental data that are used to evaluate the individual components of the land-use allocations.
- The *impact assessment* component, brings together the information derived by the land-use systems models and provides the means of analysing, for the whole management unit, the allocations proposed by the land-use planning tools. The impact assessments thus act as the fitness-evaluation functions for the land-use planning tools. Three dimensions of impact assessment are considered by the DSS, financial, social and environmental. In the test applications to date, the financial and environmental impact assessments have been used.
- The GA-based land-use planning tools are the core of an iterative system of anal-

ysis. The land-manager or analyst specifies a scenario, principally by choosing the fitness metric(s) to be employed, but also by defining the management and global parameters or by limiting the planning to a subset of the available land parcels. Within the limits of the scenario specified, the land-use planning tools search for allocation(s) to achieve the best performance for the metrics chosen. The terminal population members can then be passed to the DSS for visualisation in the GIS or for further analysis by impact assessments not used by the land-use planning tools. An example of this would be a social-impact analysis on the amount of labour required or employment provided by an allocation. Individual allocations may also be customised using expert knowledge unavailable to the DSS. The allocation may then be accepted as it stands or may inform the specification of further runs for the land-use planning tools.

1.4 Software Platform

G2 was chosen as the implementation environment for the GAs. This was to a large extent, influenced by the use of G2 for the DSS implementation. By adopting the G2 environment, the integration of the land-use planning tools with the remainder of the DSS was greatly simplified. G2 also supports an object-oriented (OO) approach to software development, which over the course of the research has proved to have several advantages. The process of design, experiment and redesign associated with a research application such as the land-use planning GAs is greatly assisted by an OO approach. The specialisation of methods, to add further functionality or the abstraction, into generic classes, of shared functionality, minimises the amount of code that has to be maintained or modified in the light of testing. Extensive use was also made of the multiple-inheritance supported by G2. This was particularly effective in allowing the use of *mixin* classes, for example, methods providing parameterisation functionality were integrated with the existing GA classes with minimal disruption.

The GAs were implemented as a stand-alone component of the DSS. The GAs are thus run as a separate G2 process, linked to the DSS using the proprietary G2-to-G2 communication protocols. This was initially done as part of the development process to avoid having to restart the DSS every time a restart was required by the GA component.

The DSS restart process is prolonged by the need to import the biophysical data from the GIS and to undertake the calculations within the land-use modules. The use of the G2-to-G2 communications strategy has, however, been maintained because it makes it possible to employ either two workstations or two processors within the same workstation, thereby increasing the speed of operation of the land-use planning tools. Communications between the two systems are based on remote-procedure calls. The communications protocols allow any of the G2 objects to be passed across the interface as long as there are compatible object definitions on both sides. For the evaluation of GA allocations, genotypes are passed from the GA to the DSS for evaluation, with fitness values and non-fitness information returned.

1.5 Approach

This thesis proposes that, using genetic algorithms, it is possible to develop effective tools for assisting decision makers with multi-objective land-use planning problems. The thesis is presented in the following chapters.

- *Previous research* first reviews existing approaches that have been applied to land-use planning and identifies their limitations, particularly for spatial and multi-objective applications. The important preconditions for the success of GA applications are then set out. The review then focuses in detail on approaches to the use of GAs for multi-objective optimisation.
- *Representation of the problem* describes the range of options considered for how best to represent the land-use planning problem within a genetic algorithm framework.
- *Single-Objective GAs* details the implementation of two single objective GAs using the representations proposed in the previous chapter. One representation directly allocates land uses to individual fields while the other supplies the parameters to a greedy-allocation algorithm within the DSS
- *Multi-Objective GAs* takes the two single-objective GAs and modifies them for application to multi-objective applications. This involves the GA not in searching for a single optimum solution but a population of solutions defining the trade-off between objectives.

- *Parameterisation* looks at two approaches to adapting GA parameters over the course of the GA run to improve the performance of the GA.
- *Evaluation* compares the two single-objective GAs for how well each can find solutions for problems with increasingly large numbers of land parcels. For the multi-objective GAs the results of a workshop-based analysis of the GA's performance relative to that of an expert land management group is presented.
- *Conclusions* are drawn from the implementation and analysis of single- and multi-objective GA performance for the strategic whole-farm planning application and proposals made for future directions of the research.

Chapter 2

Previous Research

Existing approaches to single-objective land-use planning have frequently been based on the use of linear programming (LP). While this approach has much to recommend it, there are serious limitations on the method when applied to explicitly spatial land-use planning problems. Heuristic-search methods such as simulated-annealing and tabu-search have more recently been substituted for or combined with LP, to address spatial optimisation tasks. For multi-objective land-use planning the conventional approach to date has been the use of multi-criteria decision making, but as with LP this is known to have significant limitations. These existing land-use planning approaches are reviewed in Section 2.1.

Evolutionary computational methods, especially GAs, are increasingly employed in land-use planning and related fields. Examples of existing roles for GAs in these domains are outlined in Section 2.2. The features of GAs that make them particularly suitable as the basis for land-use planning tools and the essential steps required to exploit their functionality are presented in Section 2.3. Finally the approaches taken to extending GAs for application to multi-objective problems are detailed in Section 2.4.

2.1 Existing Approaches to Land-Use Planning

2.1.1 Linear programming

LP is a well-established optimisation method that has been integrated with GIS to undertake spatial land-use planning (Chuvieco 1993, Arthur 1997). The method seeks to optimise an *objective-function*, subject to a series of constraints, usually on the resources

available. An example of an objective function would be:

$$\text{Maximise } O\{1.5Ag + 1.1Fo\}, \quad (2.1)$$

where O is the objective to be maximised, usually the economic returns and Ag and Fo are the *decision-variables*, or *alternatives*, in this case the areas of **A**griculture and **F**orestry. The coefficients for the decision-variables are the economic returns for a unit area. A series of constraints are also defined, within which the objective-function is optimised. For example, if the total area available for allocation to agriculture and forestry were 200 hectares a constraint of the form $Ag + Fo \leq 200$ would be used. Collectively the LP constraints are referred to as the *Right Hand Side* (RHS). Both the objective-function and the RHS must be specified as linear equations and the coefficients for both are assumed to be known. The LP algorithm returns the optimum decision-variable values for the objective-function and any *slack* or *surplus* in the RHS, indicating the degree to which an individual constraint affected the optimisation. The LP algorithm also returns *dual-prices* for the decision-variables and the RHS. These dual-prices indicate the change in the objective-function value for a unit change in decision-variable or constraint, assuming such a change does not alter the structure of the objective-function.

The frequent use of LP means that it represents a benchmark against which it is useful to judge other proposed land-use planning methodologies. One criticism that can be levelled at LP is that it is limited to deriving only one solution in a single run, whereas decision makers seem to prefer several alternatives to compare and choose from. By manipulation of the constraints and coefficients LP can, however, be used to sequentially investigate a number of alternative scenarios. The additional sensitivity information provided by the dual-prices can also be used to guide an iterative investigation of alternative scenarios. The most frequent criticism of LP is its underlying assumption of linearity for the objective-function and RHS. While this assumption may be invalid, it is necessary to ask whether it is demonstrable that this approximation introduces larger errors into the analysis than, for example, errors in estimating the constant values of the RHS constraints? In any case, if relationships are demonstrably non-linear then variants of LP exist that can accommodate such relationships.

While LP is guaranteed to find the globally optimum solution, subject to a RHS

specification that allows an optimum to exist, there are significant problems in using LP for spatial land-use planning. LP is often applied without incorporating any spatially-explicit constraints within the objective-function or RHS. This means that the optimum solution found by LP indicates how much land should be allocated to a specific land use but not where. The process of disaggregating the global solution often requires the use of secondary data, and may well introduce additional criteria not explicitly considered in the LP model. Indeed the disaggregation process may have a much larger impact on the pattern of land-use than the LP modelling. Furthermore, many of the *constants* employed within an LP model are in reality highly spatially-variable. For example levels of economic returns for a given land use may vary dramatically across even relatively small distances. It is possible to incorporate explicitly-spatial constraints using integer constraints (Tarp & Helles 1997) but problems can quickly become intractable, for example when spatial adjacency for multiple land management units is considered (Roise 1990).

A further potential limitation of LP is the practical effects of using a specialised model formulation. That is, an LP model of a land use system is structured to make it compatible with the optimisation method employed. This severely limits the ability of the land-use planner to take advantage of existing simulation models, as sources of information, without a re-implementation process. This duplication of effort could be especially severe if an incremental process of model refinement is being used. The re-implementation process, with its associated need for debugging and validation, means that a LP model may take significantly more time to specify than it does to run. By contrast, a GA-based approach requires only that a fitness evaluation can be made for the solution represented by the genotype. It is thus possible to substitute updated versions of the land-use models without disrupting the operation of the GA.

2.1.2 Heuristic search methods

Given the limitations of LP for spatial optimisation tasks, heuristic search methods have been used to augment or replace LP. Two popular methods, particularly in forestry, are *simulated-annealing* (SA) and *tabu-search* (TS). Both of these methods employ an *acceptance function* as part of their search to avoid the algorithm becoming stuck in local optima, a problem experienced when employing simple hill-climbing approaches.

Simulated-annealing

SA is based on the analogy of heating up a solid and then slowly cooling it until it crystallises as a perfect lattice. SA combines local search with an acceptance-function whose operation is controlled by a simulated *temperature* regime (Tarp & Helles 1997). The temperature state of the algorithm controls the probability of accepting a worse solution. The structured acceptance of worse solutions eliminates the problem of the search becoming prematurely fixed in a local optimum. The operation of SA is shown in Figure 2.1.

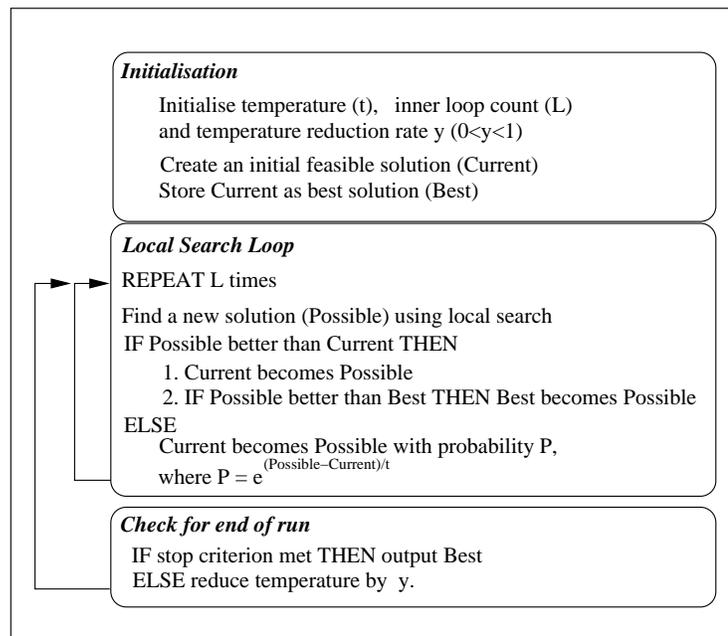


Figure 2.1: A simulated annealing algorithm.

A worse solution is accepted with probability $e^{\Delta/t}$, where Δ is the difference between the current and newly generated solutions and t is the temperature value. When the temperature is high, worse solutions are more frequently accepted and the search space is explored. As the temperature is reduced over the course of the SA run, the probability of accepting a worse solution drops and SA converges towards a near-optimal solution. Despite the acceptance of worse solutions as part of the search process, the SA algorithm is elitist as it stores the best solution to date, in addition to the solution from which the local search is being conducted. A stop criterion is used as a safety feature to end the SA run once the temperature has reached a certain level, the rate of increase in the

objective-function drops below a specified level or a defined number of iterations has been completed.

An example of the use of SA in land-use planning is provided by Tarp & Helles (1997), where LP is used to define a strategic level plan and SA is used to generate a local plan incorporating significant limits on operations that can be carried out on adjacent blocks. Comparisons between the two plans are carried out by the forest-management team. Financial costs of deviations from the LP plan and environmental or management risks for deviations from the SA plan can be assessed and workable compromises produced. While this LP/SA hybrid does not directly create a single plan integrating both the strategic and local factors, it has proved useful for decision support. SA has much to recommend it as a method for solving large spatial optimisation problems but it is limited to finding a single solution which makes it less attractive as the basis of a multi-objective land use planning tool.

Tabu-search

TS is an example of a sophisticated acceptance function that may be added to heuristic search algorithms, including local search and GAs (Bettinger, Sessions & Boston 1997, Ibaraki 1997). In contrast to SA, however, TS *always* accepts the new solution found by the search algorithm, even if the evaluation of the new solution found is worse than that of the current. This forces the algorithm to continue the search, even after a local optimum has been found. To prevent the algorithm from cycling over a small number of solutions and returning to the local optimum, TS keeps a *tabu-list* of solutions whose acceptance is forbidden. The tabu-list is a short-term memory of recently accepted solutions. In addition to the tabu-list a long-term memory list can also be used to guide a *diversification* strategy. This directs the search to unvisited regions of the search space, by the long-term prohibition of solutions previously found. The run is terminated if it fails to find a better solution after a fixed number of search operations. An example of a steepest ascent hill-climbing algorithm with TS is presented in Figure 2.2.

TS has been employed successfully in the planning of timber harvesting schedules where financial goals of reliable flow of timber are balanced by the need to preserve wildlife habitat and to observe restrictions on the harvesting of adjacent blocks (Bettinger et al. 1997). Starting from an initial state where all the constraints were satisfied, the TS was used

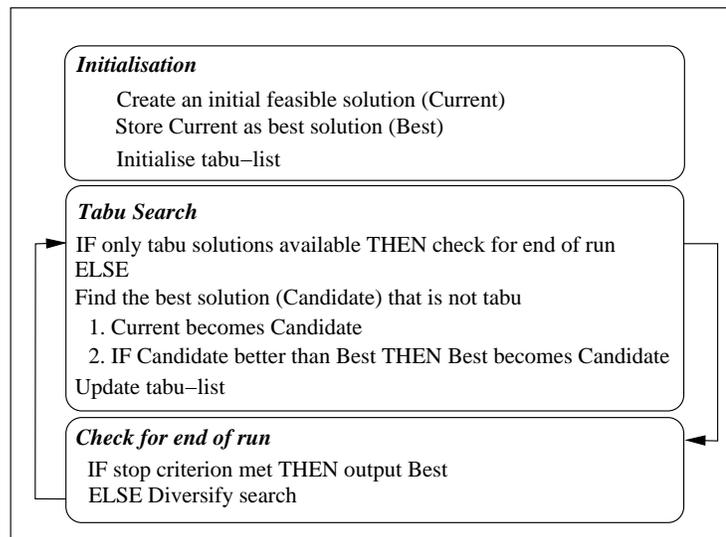


Figure 2.2: A tabu search algorithm.

to schedule blocks for harvesting. New states are generated using *moves*. The legitimate moves were to schedule, unscheduled or reschedule a block for harvest. The range of moves available is restricted by soft constraints, such as not harvesting two adjacent blocks and by hard constraints such as not harvesting a block twice in the same growth period. The solutions found by the TS were spatially and temporally feasible and respected the environmental constraints. The advantage of heuristic search algorithms with TS lies in the structured control of the search process, but this is obtained at the expense of a significant book-keeping overhead. While TS is demonstrably able to assist with complex spatio-temporal planning tasks, it has not to date been seen as necessary for multi-objective planning.

2.1.3 Multi-criteria decision making

Multi-objective planning, where one decision maker seeks, within a single plan, to achieve more than one objective, is dominated, in a land-use context, by methods collectively known as *multi-criteria decision making* or MCDM (El-Swaify & Yakowitz 1998). MCDM recognises that there are often multiple, conflicting criteria underlying a land-use decision. These conflicting criteria are brought together using a variety of methods to derive a single recommended alternative, a reduced set of acceptable alternatives or a ranking of all possible alternatives (Jankowski 1995). Two classes of MCDM approach may be

usefully distinguished, *compensatory* and *non-compensatory*. Compensatory approaches are *scalarisation* methods where high performance for one criterion may be traded-off against poorer performance for another. These methods require the decision maker to define the relative importance of criteria using weighting schemes. Non-compensatory approaches are order-based methods where alternatives are compared using single criteria in an order defined by the decision maker. Examples of both approaches are outlined below.

Compensatory MCDM

Within compensatory MCDM there are *additive* and *ideal point* methods. The most commonly used additive method is the *weighted sum* where the performance of an alternative is determined by summing the normalised values for each criterion multiplied by the decision-maker supplied weighting. A more sophisticated alternative method, *concordance-discordance analysis* uses a pairwise comparison of alternatives, based on the degree to which one solution is superior to another for the individual criteria. From the pairwise comparisons all alternatives are ranked.

Ideal point methods evaluate alternatives relative to a pre-defined ideal solution that, in reality, is not possible to achieve. For the *target vector* method the evaluation of an alternative is the Euclidean distance between the alternative and the ideal point. The coordinates used for the distance calculation are the criterion scores. The target vector method is known to be highly sensitive to the ideal point used and therefore an arbitrary choice of ideal point is undesirable. The ideal point should be chosen based on knowledge of the single objective optimum values. An alternative ideal point method is *min-max* in which alternatives minimising the difference between their criterion scores and the optimum criterion scores are preferred. This method is employed to seek the best compromise between objectives.

Non-compensatory MCDM

Several approaches to ranking are possible using direct comparison of individual criterion scores.

- *Dominance* uses pairwise comparison, where any alternative better for one criterion and equal for all others dominates, and the other alternatives are eliminated from consideration.
- *Conjunctive-cut-off* sets a minimum level for all criteria and any that fail to exceed all cut-off values are eliminated.
- *Disjunctive-cut-off* requires that an alternative exceeds at least one criterion cut-off value or it is eliminated.
- *Lexicographic ordering* requires the decision maker to order the criteria. Alternatives are then compared using the most important criteria first. Only in the case of equally good alternatives are further criteria employed.

Non-compensatory criteria are significantly less sophisticated than the compensatory approaches but may be used in geographic contexts as a first sieve to reduce the number of alternatives being considered. The conjunctive, disjunctive and lexicographic methods may also be used a part of an iterative process with cut-off values and criteria orderings modified in the light of results from the MCDM analysis.

2.1.4 MCDM applications in land-use planning

Land-use planning applications of MCDM have typically been in site selection and regional-scale zoning problems. For site selection, the goal is to use the MCDM to provide a ranking of sites from which the decision maker may choose (Carver 1991). For land-use zoning applications the output from the MCDM analysis is one or more land-use suitability maps, with MCDM scores mapped as a grid for the area of interest (Eastman, Weigen, Kyem & Toledano 1995, Beedasy & Whyatt 1999). While the suitability map(s) may be sufficient for some applications, the creation of an indicative land-use plan may need to take into account adjacency constraints or the desire to allocate zones with fixed minimum sizes. The process of arriving at a final allocation based on the suitabilities derived by MCDM may itself be a complex process of optimisation.

MCDM provides a useful method to support decision making where there are many factors and is helpful in structuring the decision making process. It is open to question how well the decision maker can define the relative importance of more than eight crite-

ria, though consistency checking techniques are available. Weights and orderings used in MCDM may be refined as part of the process and visualisation of the results as maps significantly aids the decision maker in this regard. MCDM methods are notably dependent on the weightings or orderings used, but sensitivity testing for weightings and input data accuracy may also be included to increase the reliability of the results achieved. A fundamental limitation of MCDM is that it limits the understanding gained by the decision maker of the underlying trade-offs. MCDM is, if not a black box, a grey one.

2.2 GA Applications in Similar Domains

GAs have increasingly come to be recognised as useful tools for applications in land-use and geographic domains. GAs have taken on three roles: static parameter optimisation for existing models, dynamic control of other algorithms and direct analysis, either as stand-alone tools or as part of larger DSS.

2.2.1 Static parameter optimisation

Numerical optimisation of parameter values is a role for which GAs can be successfully adapted. Mechanistic models of land-use systems, developed and tested for one agro-ecological zone, may correctly model the key biological processes, but still require re-calibration if they are to be applied in a novel situation (Sequeira, Olson, Willers & McKinion 1994). The parameterisation of spatial interaction models, predicting the numbers of trips between locations, given the intervening opportunities and travel costs, has been problematic for conventional non-linear optimisers (Diplock & Openshaw 1996). This leads to the use of sub-optimal parameters or less sophisticated models. The complex non-linear interactions between model parameters makes re-calibration a non-trivial problem. GAs have been used successfully to re-calibrate both of the models cited in this section using the fit achieved between model predications and validation data as the fitness metric.

2.2.2 Dynamic algorithm control

Dynamic algorithm control is a significant role for GAs in geographic applications. For example, the product of an MCDM analysis using spatially gridded data, is one or more *raster maps*, defining the suitability for each grid cell for each land use (Eastman et al.

1995). The goal of the planner, however, is often to allocate a land use to a spatially contiguous region, of particular size, and possibly shape. The raster suitability maps are at best a partial solution to this problem. One approach to the region definition problem has been the use of GAs to control the operation of a *parameterised region growing* (PRG) algorithm (Brookes 1997a). The PRG defines regions of spatially contiguous cells, balancing how close the region is to the ideal shape against the cumulative suitability of the cells contained within the region. The ideal shape is application specific and is a parameter of the PRG. GAs have been used successfully to optimise the regions(s) grown (Brookes 1997b).

In a similar application, a GA was successfully employed to control the re-engineering of census zones (Openshaw & Schmidt 1996). The re-zoning required a reduction in the number of zones, with larger zones created by merging spatially-contiguous existing units. In this case the GA operated as the controller for a parallel implementation of the re-zoning algorithm. The re-zoning algorithm used a simulated annealing approach and the GA supplied the zone initialisations and the temperature settings. Over the course of the run the GA tracked the best solutions and temperatures regimes and assigned processors for further analysis.

2.2.3 Direct analysis

Direct analysis of problems with a spatial aspect has long been pursued by the GA community. Classical examples of such problems include *travelling salesman*, *graph colouring*, and *location-allocation* problems (Goldberg 1989, Davis 1991, Haupt & Haupt 1998). GAs have also been used as part of a DSS to generate feasible alternatives within the constraints of a planning problem (Dibble & Densham 1997).

Increasingly there are examples of GAs being used for explicitly geographical applications. One such application is the *map labelling problem* (van Dijk, Thierens & de Berg 1999). This is a cartographic problem of how to optimise the location of labels for features on maps, particularly point features such as cities. The problem is NP-complete and also incorporates aesthetic soft constraints. The approach taken to the problem is to use a local optimiser to both generate partial solutions and to repair conflicts when these partial solutions are combined to form complete maps. The GA is unusual as it uses a crossover operator that can only split the genotype at the locations between the partial

solutions generated by the local optimiser. The GA also has no mutation operator, with diversity being generated by the local optimiser. It is, however, a very good example of the specialisation of a GA for a geographic application, employing a novel representation, modified operators and hybridising the GA with an existing tool.

GA-based models of the adaption of biological systems to dynamic environments have also been proposed (Kirley & Green 2000). The GAs employed in these investigations are *cellular GAs*, with the population of genotypes occupying a toroidal pseudo-landscape, and with mating restricted by topological connectivity constraints. These GAs have proved interesting tools for the investigation of population responses to environmental disruptions such as fires. Cellular GAs, while sharing features of selection, crossover and mutation common to GA, have such a radically different representation and so many additional features that they may be more appropriately considered examples of agent-based modelling.

GAs have also been used for spatially-explicit multi-objective planning (Feng & Lin 1999). In this application the GA was used to generate *sketch layout models*, the first stage in city planning in Taiwan. The GA was used to generate possible layouts that maximised the *efficiency*, defined as the cost/benefit ratio from public investment and the *harmony*, measured by the compatibility of contiguous land uses. The sketch layouts generated by the GA were superior to those generated by the conventional planning methods and also presented the planning authorities with a range of alternative solutions.

2.2.4 Summary of GA applications in similar domains

From the examples of applications in similar domains presented above it is clear that GAs have significant potential in land-use and geographical domains. They can be adapted to perform numerical optimisation tasks in situations where conventional approaches have proven inadequate. Integration of GAs with existing methods has also been successful, either using the GA as a controller for a second algorithm or in hybridising a second algorithm with the GA, usually a local optimiser. GAs have proven to be flexible with many different representations and operator sets developed, and to be sufficiently robust and reliable to allow their use in real world applications.

2.3 Genetic Algorithms

The features of GAs that have led to their successful use, across the diverse range of applications identified in the previous section, can be traced to their earliest origins in Holland's (1975) investigations of adaptive systems. In essence GAs are a crude representation of natural evolution with a population changing its genetic make up to best survive an environment. Emerging from the simple reproduction rules is an efficient and robust optimisation behaviour.

GAs have proved to be consistently efficient as optimisers across a range of applications (Biethahn & Nissen 1995). Given the assumptions of *no-free-lunch*, GA optimisation cannot be more efficient than a specialised algorithm developed for a specific application (Wolpert & Macready 1997, Spears & de Jong 1999). GAs do, however, provide a basis on which it is possible, via specialisation or hybridisation, to build very efficient optimisation algorithms (Davis 1991). Efficiency is desirable as it allows larger scale planning problems to be analysed and for the interactive use of the planning tools by land managers. Both of these factors are significant for the operational use of the land use planning tools.

The robustness of GA performance can be seen in their consistent ability to find good solutions particularly in very large and complex search spaces (de Jong 1990). The population based nature of their search also means that they are less likely to become trapped in local optima. The solutions found by GAs tend to be "good enough" (Goldberg 1989). For the land-use planning application the base line for "good enough" is defined relative to the existing pattern of land use. In cases where globally optimal solutions are desirable, a successful approach has been to use the GA to find the region where the optimum exists and employ a local optimiser such as hill-climbing to find the peak (Cox, Davis & Qiu 1991). Consistent performance is of particular importance for land-use planning tools as they may well be employed in situations where there is conflict about the best course of action or where there is skepticism of their utility.

Exploiting this efficient and robust performance is dependent on the existence of an appropriately responsive model to evaluate solutions, an effective representation of the search space, and a compatible and well-parameterised set of operators (Davis 1991). The developments that have been incorporated within the land-use planning GAs in pursuit of these goals are reviewed below.

2.3.1 Representation

For GAs, each genotype in the population encodes a complete specification for a single solution. The genotypes in turn are made up of genes that represent the individual components of the solution that may be manipulated by the GA operators. The representation used in the earliest GA implementation was the fixed-length binary string. This remains the most flexible representation, as interpretation is entirely dependent on the decoding used. While binary encoding was once thought to be the superior encoding for use with GAs, this interpretation was challenged by Antonisse (1989). Subsequently a diverse range of alternative representations have been proposed to improve GA performance for particular applications. The representations that influenced the land-use planning GAs are set out below.

- Real-coded genes have been successfully employed for numerical optimisation with specialised operator sets (Michalewicz 1992).
- Order-based representations with reordering operators have been used for applications where permutations of sets are a natural representation. (Goldberg 1989).

In investigating problems that had proved unexpectedly difficult for GAs to optimise, Goldberg, Korb & Deb (1989) proposed the use of a *messy* representation. With a messy representation genes could be combined together in any order as each gene was tagged so that it could be decoded irrespective of its location on the genotype (Goldberg, Deb, Kargupta & Harik 1993). The messy GA searched both for an optimum solution and an optimum ordering for the genes. The messy genotypes could be of variable length with the possibility of *over-specification* with too many genes or *under-specification* with too few. Over-specification resulted in redundant genes that required to be eliminated while under-specification was dealt with by filling gaps from the current best solution, termed the *competitive template*.

2.3.2 Replacement

Replacement controls the survival of the fittest process in GAs. For classical GAs an new population of offspring genotypes is *generated* at each iteration of the GA. The parent population is then eliminated and replaced by the offspring population. This *generational*

replacement strategy can, however, result in the loss of fit genotypes from the population due to the finite size of the GA population and the stochastic nature of the selection, crossover and mutation operators. While these losses can be minimised by the careful design of the selection algorithm (Baker 1989) the GA may still need to *regenerate* lost genotypes making the GA less efficient.

To eliminate such losses replacement strategies have been proposed where parent and offspring populations overlap, with parent genotypes passing unmodified into the offspring population.

- The minimum overlap occurs with *elitist* replacement, where the best genotype from the parent population is copied into the offspring population. While this ensures that the best performing genotype cannot be lost, a case can be made for protecting larger proportions of the population by increasing the degree of overlap.
- Overlap is maximised for *steady-state* or *individual replacement* strategies, where for each genotype added to the population, one is immediately eliminated, usually the least fit (Whitley 1988). Use of such a strategy eliminates the need to regenerate genotypes as they will only be removed from the population once no longer fit. While this replacement strategy was not found by Goldberg, Korb & Deb (1991) to have any inherent advantage, Davis (1991) indicates that the superiority of individual replacement may depend on other aspects of GA formulation such as enforcing genotype uniqueness within the population and the performance metrics employed. Since genotypes are not re-evaluated at any point with an individual replacement strategy, it should not be employed where the fitness evaluation is dynamic or noisy.

2.3.3 Operators

New representations such as real-coding and permutations have required the modification of the GA operators, such as the use of non-uniform mutation for real-coded genes (Michalewicz 1992). Beyond representation-specific modifications, the original single-point crossover has been succeeded by multi-point and eventually uniform crossover, where individual genes not genotype segments are exchanged (Syswerda 1989). The exploratory power of crossover was found to be maximised when uniform crossover was employed and the *crossover proportion*, of genes from each parent, was set to 0.5 (Eshelman, Caruana

& Schaffer 1989). Eshelman et al. also noted that this caused maximal genotype disruption, that is, genotypes without the genetic features required for high fitness were created. Spears & de Jong's (1999) analysis points to a no-free-lunch theorem for exploratory power: increased exploratory power is exactly balanced by crossover disruption. It is possible, however, that choosing an elitist replacement strategy may mitigate the disruptive effects of the high crossover proportion. With such a replacement strategy, crossover disruption results only in a failure to make a fitness gain for the single operation, not in reduced fitness genotypes within the population, which could be propagated by further reproduction.

2.3.4 Parameterisation

Parameterisation of the GA can represent a significant challenge due to the non-linear interactions between the components of the GA. While GA performance can be robust in the face of sub-optimal parameterisation, it may also be seriously degraded. Approaches to parameterisation have included: systematic investigation of parameter settings, theoretical analysis and the use of *meta-optimisation* approaches (Freisleben 1997). Meta-optimisation is the setting of GA parameters using either another GA or some other optimisation tool. The meta-optimisation approaches were developed because parameterisation was increasingly seen to be algorithm-dependent and possibly application-specific. For certain GA parameters this meta-optimisation needs to be conducted during the course of the run as the parameters follow varying trajectories (Davis 1989). Davis proposes a methodology which uses information on the fitness gains made by individual reproductive events as the basis for adjusting parameterisation.

2.4 Multi-objective GAs

In extending the application of GAs to multi-objective problems the key distinction is between *a priori* and *a posteriori* approaches (van Veldhuizen & Lamont 2000).

- With *a priori* the model of operation is *decide-then-search*, with the decision maker defining a weighting or ordering scheme for the objectives and a solution then generated. This model uses the scalarisation or ordering approaches seen in MCDM. Solutions found by the decide-and-search model are known to be highly sensitive to

the weightings and orderings employed (Fonseca & Fleming 1994). This means that in certain situations, particularly where there is conflict over a decision, it may be impossible to agree *a priori* weightings or orderings. It may also be alleged that the decide-then-search model is employed corruptly to find weightings or orderings that result in the solution desired.

- In *a posteriori* the model is *search-and-decide*, with the decision maker presented with a range of alternatives, defining the trade-off between objectives. The search-and-decide model is clearly superior for a land-use planning application as it emphasises the interactive and transparent nature of the process. The search-and-decide model is less of a black box. While it is probably not entirely transparent to the decision-maker how the range of possible solutions is derived at least the nature of the conflict between objectives is presented.

Several important elements for implementing a search-and-decide model can be traced back to the development of the Vector Evaluated Genetic Algorithm (Shaffer 1985). Most significant among these is Shaffer's observation that no Utopian solution is likely to exist, where all objectives are simultaneously optimal. The population-based nature of the GA search can therefore be exploited to find multiple solutions defining a range of compromises between the conflicting objectives in a single run.

2.4.1 Pareto-optimal sets

The nature of the compromise between objectives was formalised by Shaffer using the *dominance* relation between genotypes. One genotype *dominates* another only if superior, in pairwise comparison, for at least one objective, and with equal or better performance for the other objectives. A genotype that is *non-dominated* is frequently referred to as *Pareto-optimal*. In fact only genotypes that are not dominated by any solution within the whole search space are true members of the set of Pareto-optimal solutions (van Veldhuizen & Lamont 2000). Since it is not possible in most cases to enumerate all solutions, non-dominance or Pareto-optimality for a member of a GA population is, in practice, defined relative to either the current members of the population or to an archive of best-solutions-to-date. The GA may thus not find solutions that are globally Pareto-optimal but solutions that are near Pareto-optimal. Furthermore, in most cases the set of

Pareto-optimal solutions is much larger than the GA population, so it is only possible for the GA to find a subset. Thus the output from the GA is an approximation of a subset of the Pareto-optimal set.

Despite these limitations, comparison of the individual members of the GA population provides the decision maker with an explicit representation of the conflict between objectives. It is therefore highly desirable that the subset of non-dominated solutions found by the GA be as representative of the Pareto-optimal set as possible. This means that the GA subset should encompass the ranges of fitness values that are Pareto-optimal. Furthermore the individual genotypes should be evenly spread across that range. Achieving these two goals greatly assists the decision-maker by providing the best characterisation of the trade-off between objectives. The sections that follow review the alternative approaches taken to achieve these goals.

2.4.2 Pareto-based selection

The implementation of selection provides the means of biasing the evolution towards the creation and maintenance of a population of non-dominated solutions. Several variants of the Pareto-based selection strategies have been proposed reflecting the integration of the dominance relationship with existing GA selection strategies. This approach has become the *de facto* standard in the mGA literature.

Pareto ranking was first proposed by Goldberg (1989), as a means of eliminating the bias in favour of genotypes with excellent performance in only one objective, seen in Shaffer's pioneering work. Pareto-ranking uses the dominance relationship to rank the population, with the fitness values used in parent selection assigned using a linear normalisation function (Whitley 1989).

Two alternative Pareto-ranking schemes have been proposed by Goldberg and Fonseca & Fleming (1995b). The schemes are illustrated, for a maximisation problem, in Figure 2.3. The dominated genotypes of the GA population are marked with the open circles and the non-dominated with closed. For Goldberg's ranking the non-dominated genotypes are assigned the best ranking of one. These are then removed from consideration and the remaining genotypes compared with the dominance relation. The new non-dominated genotypes are assigned rank two and the process repeated until all genotypes are ranked. For Fonseca & Fleming's scheme the rank of a genotype is the count of the number of

genotypes in the population that dominate it. Non-dominated genotypes thus receive rank zero. It is thus possible to have missing ranks for example no genotypes with rank two or four in Figure 2.3.

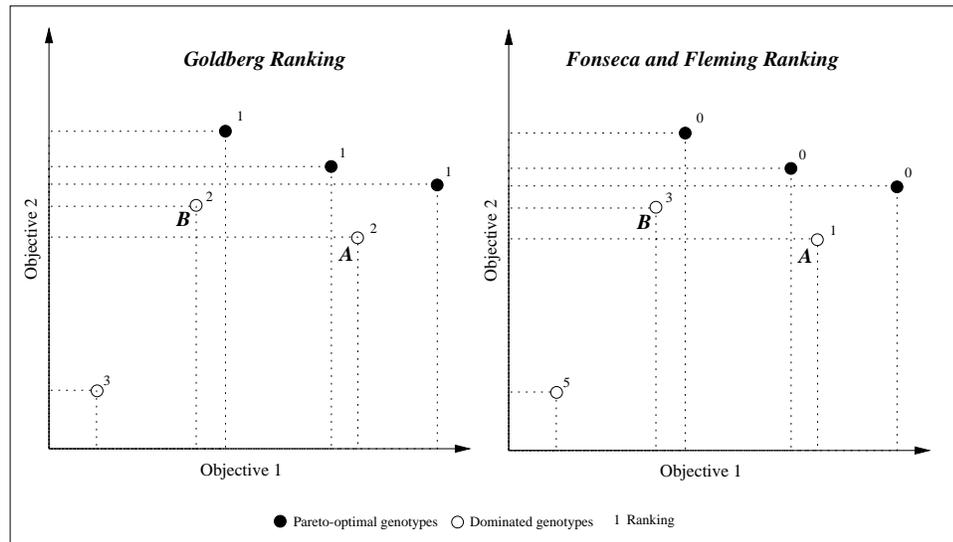


Figure 2.3: Goldberg's and Fonseca and Fleming's Pareto-ranking schemes.

The ranking approach proposed by Fonseca & Fleming can be considered superior to that of Goldberg since it differentiates better between genotypes that, while dominated, exist in less densely sampled parts of the search space. For example the Fonseca & Fleming scheme ranks genotype A higher than B because genotype A is dominated by more genotypes. This finer-grained ranking is particularly useful if an individual replacement strategy is adopted because it ensures that the most dominated genotypes are identified and can be removed first from the population.

Niche-based fitness sharing

While the use of dominance-based selection guarantees selection pressure in favour of non-dominated solutions, it is not sufficient to ensure that the population will be evenly distributed. A common approach to achieving this goal is to apply an additional selective pressure against genotypes occupying the same regions of the search space (Goldberg 1989, Fonseca & Fleming 1995b, Srinivas & Deb 1995). The selective pressure is applied by reducing the probability that a genotype will be selected as the number and proximity of other members of the population increases. This reduction is termed *fitness sharing*. The

distance beyond which other genotypes do not cause sharing is the *niche-size*, a parameter of the mGA. The niche-size defines the average expected spacing of genotypes in the final population.

Differences in selection probability due to niche-based sharing guide the mGA evolution as the proportion of non-dominated genotypes in the population increases. Depending on the replacement scheme adopted, niche-based sharing will also affect the order in which genotypes are replaced. In a generational replacement scheme, genotypes which have their fitness reduced by sharing will tend to fail to reproduce themselves in the next generation. With an individual-replacement approach, sharing will determine the genotypes being replaced, with those in the most densely sampled parts of the search space replaced first.

As part of its niche-based fitness sharing, the Non-dominated Sorting GA (NSGA) ensures that the maximum selection probability for a lower-ranked genotype is always less than the minimum shared fitness of the genotype(s) ranked above it (Srinivas & Deb 1995). This method raises the issue of whether a non-dominated individual in a densely sampled area of the search space should be retained at the expense of a dominated solution in a less densely sampled region. This probably depends on the usefulness of the genetic diversity being provided by the dominated solution, in the sparsely sampled part of the search space, and the extent to which the dominated genotype can be cross-bred to form non-dominated solutions. The NSGA may be being unnecessarily dogmatic in its insistence that dominance must on all occasions be paramount.

Sharing spaces

The space within which the sharing takes place can be either *genotypic* or *phenotypic*. Distances in genotypic space represent differences at a genetic level between individual genotypes, regardless of how the genotypes may be evaluated. An example of a genetic distance is the Hamming distance computed between two binary-representation genotypes (Goldberg 1989). Phenotypic space, conversely, is the search space defined by the mGA objectives. Genotypes have coordinates within that space determined by their fitness evaluation values.

Phenotypic sharing is used by both Fonseca & Fleming and Horn, Nafpliotis & Goldberg (1994). Horn et al.'s intuition is that phenotypic space is the space “we care more about”, and that sharing should be used to maintain diversity along the trade-off defined

in phenotypic space. Srinivas & Deb argue that phenotypic sharing may fail to maintain the diversity of input parameter values and it is these input parameter values that interest the decision maker.

The choice of sharing space in reality depends on the objective of the mGA. In most cases for the land-use planning application, the goal will be to find solutions across phenotypic space, showing the possible range of impacts as measured by the objective functions. It is possible to conceive of applications, however, where it would be very useful to find different land-use plans that result in similar outcomes. For these applications genotypic sharing would be appropriate.

Sharing Parameterisation

The successful employment of niche-based fitness sharing is dependent on setting an appropriate combination of niche- and population-size (van Veldhuizen & Lamont 2000). A successfully applied approach to this problem is to set the population size with the following formula (Fonseca & Fleming 1995b):

$$N = \frac{\prod_{i=1}^n (\Delta_i + \sigma_{share}) - \prod_{i=1}^n \Delta_i}{\sigma_{share}^n} \quad (2.2)$$

where Δ_i is $M_i - m_i$, the maximum minus minimum value for the objective i , and σ_{share} is the niche-size. The population size N is calculated as the difference in volume between two hyper-parallelipeds with edges $\Delta_i + \sigma_{share}$ and Δ_i , divided by the volume of the hypercube of edge σ_{share} . This formulation is based on the observation that the area of the trade-off cannot be larger than the sum of the areas of the Δ_i hyper-planes. The population size is thus the number of hypercubes required to cover this surface with centres σ_{share} apart, using the ∞ -norm distance metric. This is illustrated for two- and three-objective applications in Figure 2.4. Fonseca & Fleming also provides the converse formulation setting niche-size given a population size.

In practice, the best results with this formula have been achieved using normalised values for the objectives and with σ_{share} chosen to give an adequate graphical visualisation of the trade-off (Fonseca & Fleming 1995a). The values for the normalisation are updated based on the maximum and minimum values in the current population.

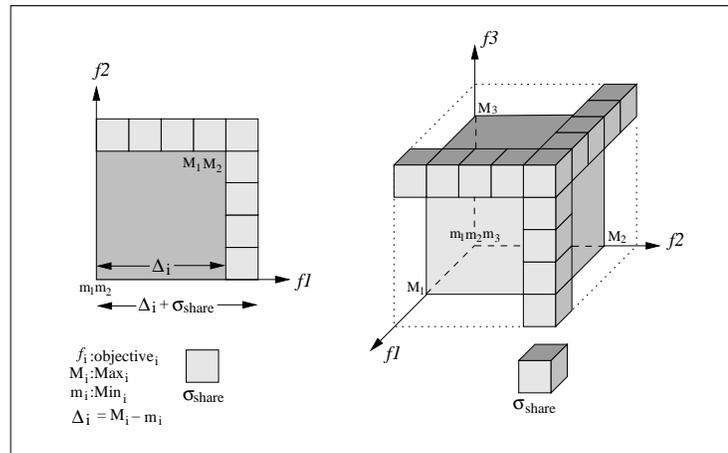


Figure 2.4: Niche size and its relationship to population size

2.4.3 Alternative mGA approaches

Pareto tournament

The Niched Pareto GA (NPGA) uses the dominance relation as part of a tournament selection scheme (Horn et al. 1994). Rather than Pareto-ranking the whole population, NPGA uses Pareto-domination tournaments. In these tournaments two candidate parents, selected at random, are compared using the dominance relation with a comparison-set also drawn at random from the population. If one only parent is non-dominated then it is chosen. Otherwise, when neither or both are dominated, the tie is broken using the count of the number of genotypes within the niches surrounding the parents. The parent with the lower count is accepted. NPGA is an elegant means of combining dominance-based selection and population distribution, but is known to be highly sensitive to the parameterisation of the comparison-set size.

Strength Pareto Evolutionary Algorithm

Strength Pareto Evolutionary Algorithm (SPEA) uses a secondary population, termed the *Pareto-set* to store the non-dominated solutions found to date (Zitzler & Thiele 1998). The secondary population is elitist, with only non-dominated solutions allowed to enter; any solutions that become dominated are removed. Clustering is used to reduced the size of the Pareto-set if it grows too large. The clustering approach is simpler than niching as it does not require the balancing of niche and population sizes. Both the population

and Pareto-set participate in reproduction, with a generational replacement strategy used. Parent selection is conducted using binary tournament selection.

SPEA is notable for using a novel dominance ranking. A genotype's ranking, for the members of the Pareto-set, is the proportion of the population it dominates, this is the genotypes *strength*. For the genotypes in the population, ranking is the sum of the strengths of the dominating genotypes in the Pareto-set +1. Figure 2.5 presents an example of SPEA ranking.

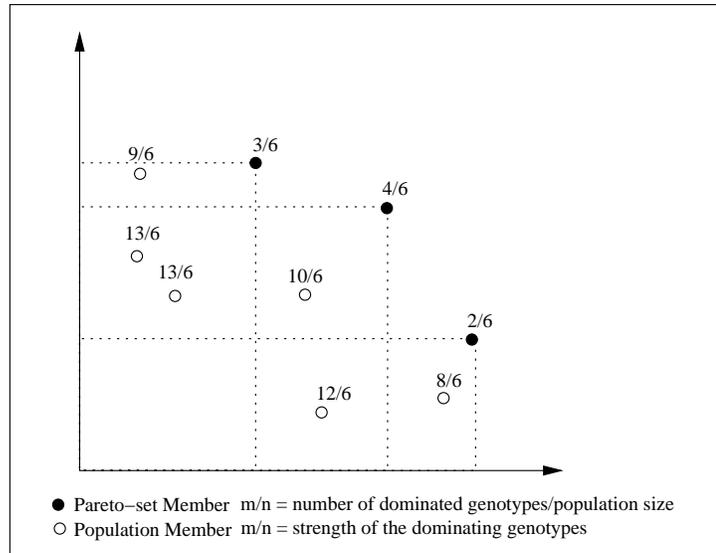


Figure 2.5: SPEA dominance ranking

The effect of this ranking scheme is to bias selection in the population so as to encourage the creation of genotypes across and close to the trade-off defined by the Pareto-set. This is desirable as it provides a diversified pool of genotypes for recombination/mutation.

Pareto Archived Evolutionary Strategy

The Pareto Archived Evolutionary Strategy (PAES) is a radical departure from the approaches to multi-objective optimisation so far discussed (Knowles & Corne 1999). In its simplest form, PAES is a (1+1) *evolutionary strategy*, with a single parent mutated to produce a single offspring (Bäck & Schwefel 1993). Other forms include a (1+ λ) and a ($\mu + \lambda$) evolutionary strategy. The algorithm is essentially a hill-climber with the parent replaced if dominated by the offspring. When neither is dominant a comparison set is used to break the tie. The comparison set in this case is an archive of the non-dominated solu-

tions found to date. Solutions are preferred that occupy less crowded areas of the search space. Crowding is measured using a recursive subdivision of space with the number and identity of the genotypes recorded.

The (1+1), variant of PAES has proved to be the most effective. It combines an aggressive elitist search strategy with an efficient archive-maintenance method. PAES speed of operation is of notable benefit in time dependent applications, such as the telecommunications domain in which it was developed. In Knowles & Corne's testing PAES was out-performed, however, by a modified version of NSGA, with an elitist replacement strategy and an archive of non-dominated solutions added. This may be because the hill-climbing-based strategy finds it difficult to find both ends of the Pareto-front, since it is unable to backtrack.

2.4.4 Replacement

The replacement strategy of mGAs is consistently identified as a crucial element in determining mGA performance. Comparison of mGAs, in all cases indicates that performance is enhanced by the use of elitism (Zitzler, Deb & Thiele 1999). A reason for the importance of elitist strategies may be that mGAs, like learning classifier systems (LCS), are searching for a co-adapted set of genotypes (Valenzuela-Rendón & Uresti-Charre 1997). The LCS is seeking a set of rules encoded as genotypes, whereas the mGAs are searching for non-dominated genotypes across the trade-off. For LCS, individual replacement, the strongest form of elitism is the norm, because it is recognised that a co-adapted set of rules will only be found incrementally. Genotypes are replaced only when they cease to fit in the context of the current population. Elitism ensures that fit individuals cannot be lost by chance, with the mGA having to rediscover them subsequently.

2.4.5 Mating restrictions

An element of mGA design on which there is less unanimity is the use of mating restrictions. Two forms of mating restriction have been identified (Shaffer 1985). *In-breeding* is the recombination of genotypes occupying similar locations within the search space defined by the objectives, whereas *cross-breeding* recombines genotypes from distant locations. Cross-breeding of non-dominated genotypes from separate parts of the search

space was found to result in non-dominated solutions at intermediate locations (Horn et al. 1994). Fonseca & Fleming (1998), conversely use in-breeding to attempt to reduce the number of non-viable genotypes produced by the GA. Others report no benefit from the use of mating restrictions (Zitzler et al. 1999). A possible conclusion is that the utility of mating restrictions may be problem dependent (van Veldhuizen & Lamont 2000).

2.4.6 Summary of mGA themes

An mGA applied to the problem of multi-objective land-use planning should adopt an *a posteriori* approach, finding populations of genotypes defining the trade-off between objectives. This allows the decision maker to learn from the mGA results and possibly to modify the underlying premises on which the mGA is operating. Given an *a posteriori* approach, the use of the dominance relation for comparisons between genotypes has been seen to be effective. This biases selection in favour of non-dominated, and possibly Pareto-optimal solutions. To provide an adequate characterisation of the trade-off between objectives, the population found by the mGA should be evenly distributed. With a single population this requires the use of niche-based fitness sharing or with a secondary-population, a clustering or crowding mechanism. In all cases it is essential that elitism be employed to ensure that useful genotypes are not lost by chance and are only replaced once superseded by fitter offspring. Diversity within the mGA population, defined in either genotypic or phenotypic terms is vital with both cross-breeding and in-breeding reproduction seen as appropriate.

2.5 Conclusions

Optimisation tools must be flexible, robust and acceptably efficient in order to tackle strategic land-use planning problems. In particular there is the need to be able to tackle spatially-explicit applications without the need for a *post hoc* process of disaggregation to translate optimal solutions into patterns of land use. The tools implemented must also be capable of tackling problems large enough to represent real world management units. Analysis of problems with multiple, non-commensurable objectives should also be possible. For multi-objective land-use planning, the goal should be to enhance the understanding of the conflicts between objectives and thus assist in making rational compromise decisions.

Of the existing tools reviewed in this chapter, all fulfill some of these criteria but none

met all. LP and its related methods are efficient but are significantly limited in their capacity to deal with spatially-explicit problems of sufficient size. LP is also inflexible as a component of a decision support system. The heuristic search methods, while capable of dealing with large-scale spatial problems, cannot be readily applied to multi-objective planning. MCDM methods are adept at presenting a range of compromise alternatives for multi-objective problems, but do so without making clear the underlying structure of the conflicts between objectives, thus making the justification of decisions based on such results difficult. MCDM methods may also be limited by their reliance on decision-maker-defined weighting or orderings of criteria or the need for significant *post hoc* processing.

Strategic land-use planning tools using GAs for search and optimisation meet the criteria identified above and present a significant opportunity to assist decision makers.

Chapter 3

Representation of the Problem

3.1 Introduction

In Chapter 1, GAs were identified as potentially useful approaches to land-use planning. The key to the successful application of these methods, however, is the translation of the problem into a genotype design that may be manipulated successfully by the GA operators. This chapter outlines the representations considered; the basis of the decision to proceed with two contrasting representations is also presented.

3.2 Spatial and Non-spatial Representations

Given the background to the research, in GIS-based decision support, the initial representations considered were those commonly used in the GIS community: the grid, the quad-tree and the polygon (Samet 1989). These three representations are illustrated in Figure 3.1 and their suitability as the basis for a GA representation discussed in this section.

3.2.1 Grid representation

The grid representation is probably the simplest, being a two dimensional array of values. For the application considered here, the values are land uses. Each cell in the array represents a geographic area of fixed size. Selecting the size of cells determines the level of detail and the size of data structure that must be manipulated.

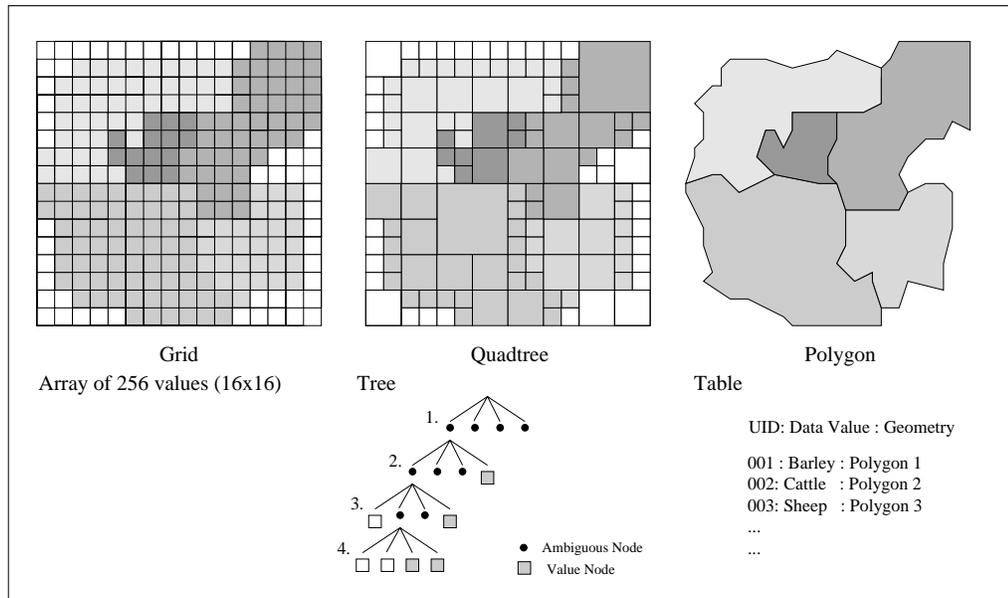


Figure 3.1: Grid, Quad-tree and Polygon Representation

As a genotype representation for the land use planning application, the grid has advantages. It is a simple structure that can be easily manipulated. GA operators may be readily translated for two dimensions (Cartwright & Harris 1993). Figure 3.2 shows examples of grid-based crossover and mutation operators implemented in the initial phases of this research. A further reason for considering the grid representation is that spatially-adjacent land parcels are likely to have similar underlying biophysical properties, and thus their suitability and productivity are likely to be correlated. It is probable, therefore that adjacent grid cells will form *building-blocks* from which the overall solution can be constructed.

A significant problem with the grid representation, however, is redundancy. This is particularly likely when using categorical data such as land use types. For example, compare the five table rows required by the polygon representation to store the land uses for the five land parcels in Figure 3.1, with the 256 values needed for the grid representation. Indeed even with this level of redundancy, the grid provides a relatively coarse view of the landscape compared with that of the polygon. This redundancy is undesirable on two levels. First it increases the size of the optimisation task for the GA and second it makes the evaluation of allocations by the DSS more demanding. This is of course in addition

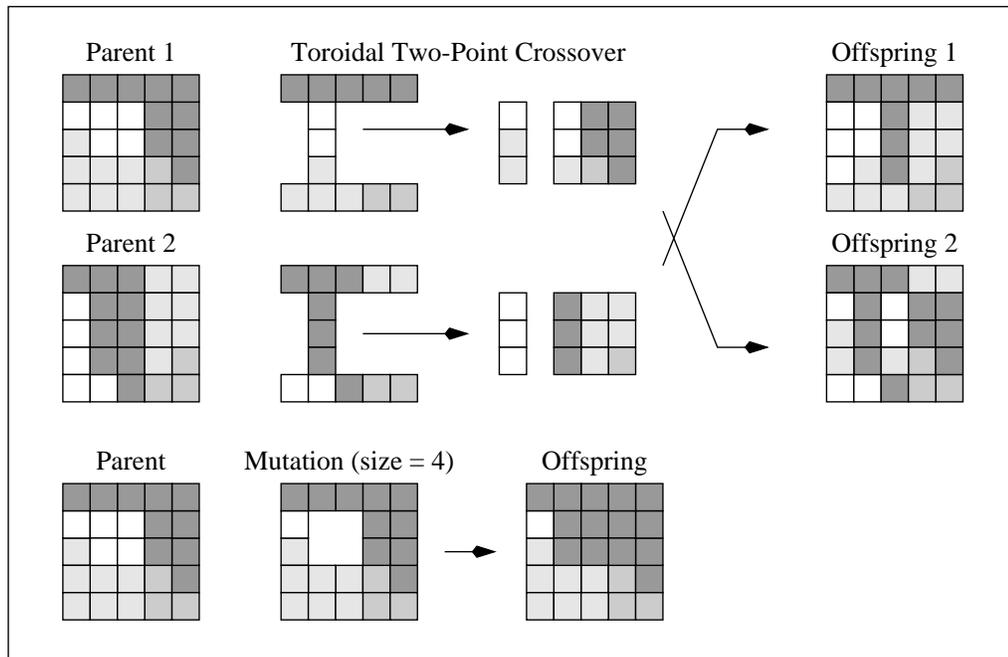


Figure 3.2: Grid-based toroidal two-point crossover and mutation

to the computer resource difficulties that may be caused by the need for a population of genotypes with such large data structures.

3.2.2 Quad-tree representation

The issue of redundancy in the grid representation has long been recognised and the quad-tree is one data structure proposed to address the issue. The quad-tree recursively subdivides space into progressively smaller spatial units (quad-blocks), indexed using a tree-structure. Subdivision proceeds until a node in the tree is homogeneous, with only one land use represented or the maximum depth of recursion has been reached. In Figure 3.3, four levels of recursion are presented. At the bottom level, if the quad-block is still not homogeneous, then the land use with the largest area is used. In the partial tree shown, no node is homogeneous at level one (the whole grid). The example then presents the level-two subdivision of the lower left block, which has one homogeneous block (quad-block 3). At level three (again considering only the division of the lower left quad-block) there are two homogeneous blocks (0 and 3). Finally, the example shows the subdivision of the lower right block at level three into four majority land use blocks. By this mechanism the quad-tree representation focuses detail where it is necessary, at the boundaries of the

real world features. The quad-tree has the same building-block advantages as the grid and GA operators have been implemented for use with tree-structures, for example in genetic programming (Koza 1993).

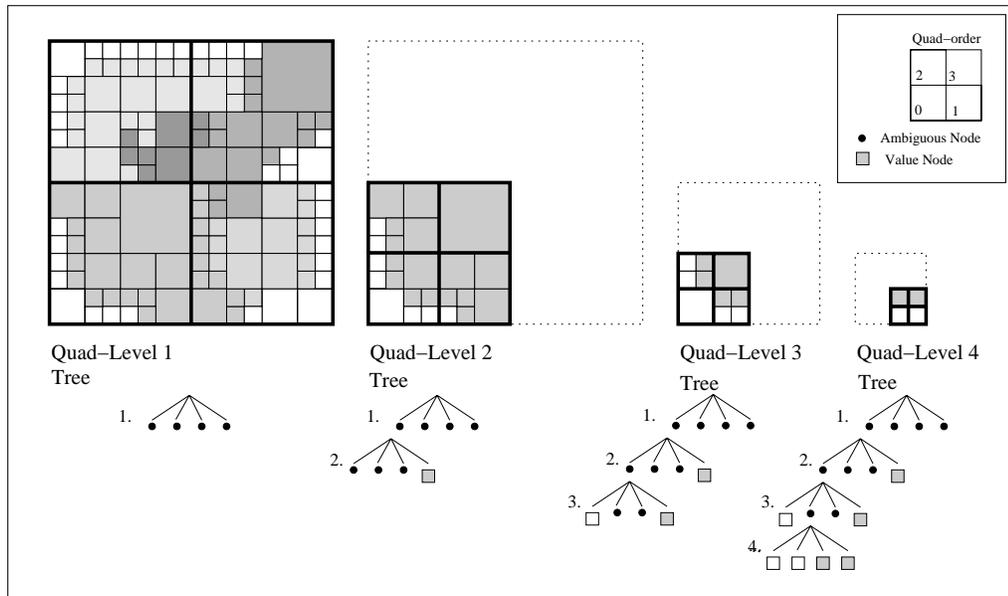


Figure 3.3: Quad-tree example.

The reduction in data redundancy of quad-trees is, however, at the expense of algorithmic and computational complexity. Building and repairing the quad-tree structure also adds an overhead to the optimisation process. The quad-tree is still a relatively inefficient means of representing management units as there is the need to translate any pattern of land use defined by the quad-tree into a real world field-system. This quad-tree to field-system translation process is a further significant overhead that may, in fact, be more complex than the optimisation of the land uses.

3.2.3 Polygon representation

The final representation from those initially considered is the polygon. In this representation, the focus is on real-world management units rather than more abstract divisions of space. Each polygon represents the boundary of an object, in this case the land parcel to which a land use is allocated. The part of the polygon data structure manipulated by the GA can be seen as a-spatial as each polygon is linked to a record in a relational database table, or an instance in an object store. The land use is simply a field of the

record or a slot of the object, the value of which can be modified by the operations of the GA. Any analysis that depends on the geometric properties of the real-world object can be handled by the GIS. If each management unit forms a single gene then the genotype is simply a list of those genes. This does, however, mean that spatial relationships between management units cannot be maintained in the genotype. The use of a representation that maintains building-block proximity is, however, only effective when building-block disruption by operators is position dependent (Eshelman et al. 1989). One- and two-point crossover operators exhibit such position dependence. The desire to use more exploratory crossover operators such as uniform crossover, however, means that building-block disruption depends on order, (the number of genes in the building block,) rather than on position. Operator design choices are further discussed in Chapter 4. An a-spatial representation based on a list structure can thus be considered with individual genes referring to a property of the polygons.

3.3 Land-Block Representation

Within the DSS, the polygon class used to store all the information on land allocation is the *land-block*. The land-block is defined as an area of uniform land use, such as a field or forest compartment. Within the land-block there may be differences in management protocols or biophysical conditions but it is at this level that land allocations are made. The Land-Block representation is thus a mapping from individual genes on the genotype to individual land-blocks, Figure 3.4. The representation has similarities to that adopted for other multi-parameter optimisation problems (Bramlette & Bouchard 1991). It is a fixed-length, fixed-order genotype with each gene defining a parameter, in this case the land use. The mapping between the the individual genes and the land-blocks is set up as part of the GA initialisation process. Fitness evaluation for a land-block is a matter of looking up the appropriate values, determined by the allocated land use.

The case for adopting of a Land-Block representation is further strengthened when one considers that land-blocks represent real-world enterprise infrastructures. Existing land-blocks have proven practical utility as management units because of their size and biophysical characteristics and their alteration entails significant capital costs. Use of these units removes the need for the potentially difficult problem translating grid or quad-

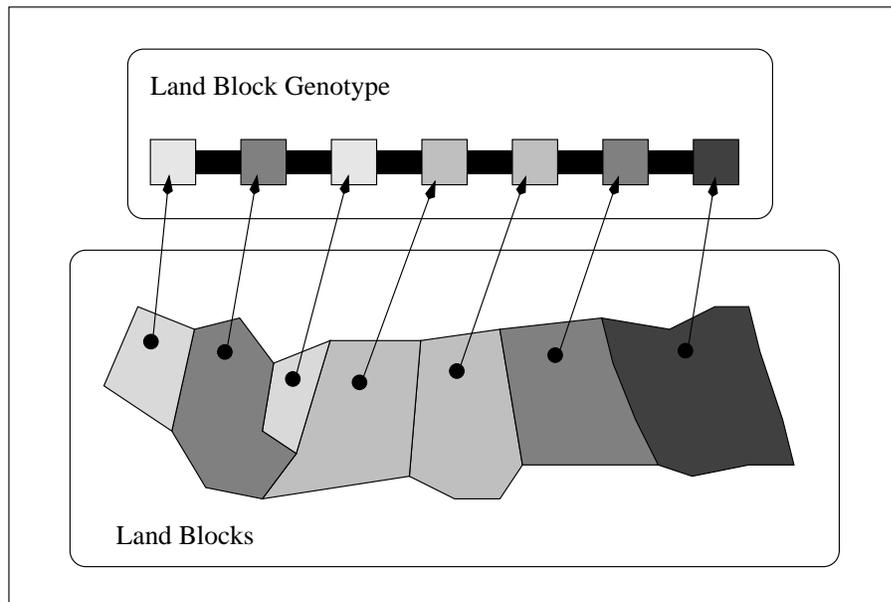


Figure 3.4: Land Block Representation

tree allocations into an acceptable management framework. The use of land-blocks also accords with Beurden & Padding's (1994) advice that, whatever the units used for any analysis, the results must be presented to decision makers using recognisable management units.

Two limitations of the representation are apparent. The first is that the framework of land-blocks may not be suitable, without modification, for a new system of land use, for example splitting a field in order to plant a woodland at one end, or combining two fields in order to increase the efficiency of farm machinery. The ability to combine, split or otherwise modify land-block shapes is potentially desirable. For example Voronoi tessellations could be used to subdivide land-blocks (Barrett 1997). It was decided, however, that incorporating modifications to field structure lay beyond the scope of this research. The second, and possibly more fundamental potential limitation, was how well the Land-Block representation would scale with increasing numbers of land-blocks. While the DSS applications that predate the GA research were of the order of 65 blocks, could a Land-Block GA cope with 650 blocks? This concern, prompted by Davis (1991), led to the development of an alternative representation less affected by the number of land-blocks to which land uses are to be allocated.

3.4 Percentage and Priority Representation

Given the success of hybridising GAs with other algorithms (Cox et al. 1991) or in using the GA to parameterise a second algorithm, it was decided it was thought appropriate to investigate such an approach for the land-use planning problem. Within the DSS there was an existing land-allocation algorithm that made allocations based on user priorities and areas of land use to be achieved, with the areas expressed as percentages of the whole management unit. An example of such a specification would be 10% arable, 40% trees and the remainder into livestock. This *Percentage and Priority* (P&P) representation of allocations had been found to have similarities with that employed by expert land-managers. Using the DSS-algorithm to make the allocation to individual land-blocks, the GA genotype would encode a prioritised list of *target-percentages*. The magnitude of the genotype would therefore depend on the number of land uses to be allocated, usually less than 10, rather than the number of blocks, 100 or more.

The P&P gene representation is more complex, each gene having three components: *land use*; *priority* and target-percentage, Figure 3.5. The land use to be allocated is encoded in the object-class of the gene. The target-percentage to be allocated is encoded as a real-coded parameter of the gene with the range and granularity of the permitted values explicitly controlled. Finally the priority for each gene is determined by its position on the genotype. The interpretation of a gene is thus position-dependent. As with messy genotypes, the P&P representation may be either *under-specified* or *over-specified* (Goldberg et al. 1993). In the first case the target-percentages sum to less than 100%. This is permitted, with unallocated blocks considered as appropriate within the fitness evaluation used, usually being ignored. Under-allocated genotypes will tend to be selected against, and will be eliminated during the course of the GA run, as most evaluation-functions are maximised by complete allocations. For over-allocation, the prioritising of genes simply means that the later target percentages fail to allocate any land blocks. Genes failing to make allocations, or allocating less than the target value, are dealt with using *non-fitness feedback* mechanisms detailed in Chapter 4. Having defined the P&P representation it is now useful to outline the operation of the DSS-algorithm

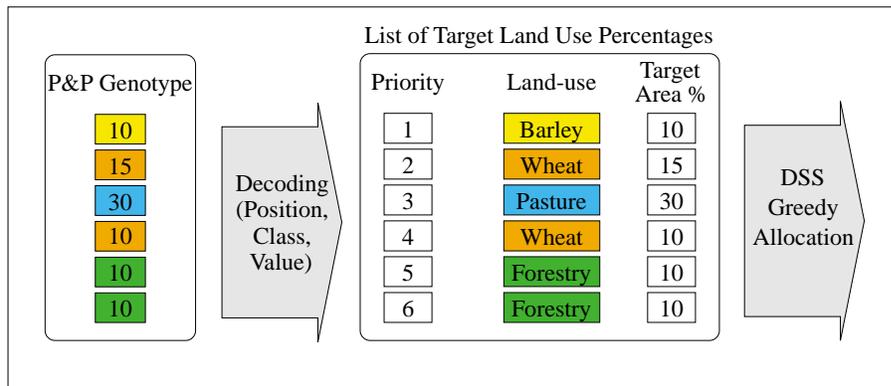


Figure 3.5: An example of the P&P Representation

3.4.1 The DSS land-allocation greedy-algorithm

The algorithm used to make the allocations is a simple *greedy-algorithm*, illustrated in Figure 3.6. The algorithm begins by creating a list of the land-blocks which may be allocated. For each land use in priority order, the algorithm sorts the land-blocks according to their productivity per hectare. The biophysical productivity is used as this is influenced by the management regime of the land-block but does not explicitly take account of financial or other factors that could form part of the fitness evaluation. The algorithm then iteratively allocates the most productive land-block to the land use until either the target-percentage is exceeded, there remain no land-blocks for which the land use is suitable or all land-blocks are allocated. The algorithm repeats the sorting and greedy-allocation until all land-blocks are allocated or there are no more P&P genes.

It is apparent from the nature of the greedy-algorithm that the allocations made will depend greatly on the operation of the sorting function. The use of productivity per hectare, while outwardly a neutral sorting function, may, by allocating the most productive land first, bias allocations in favour of production maximisation. This may be undesirable where the fitness evaluation is based on environmental impact. The bias may also affect how well the P&P representation can perform as part of a multi-objective land use allocation tool.

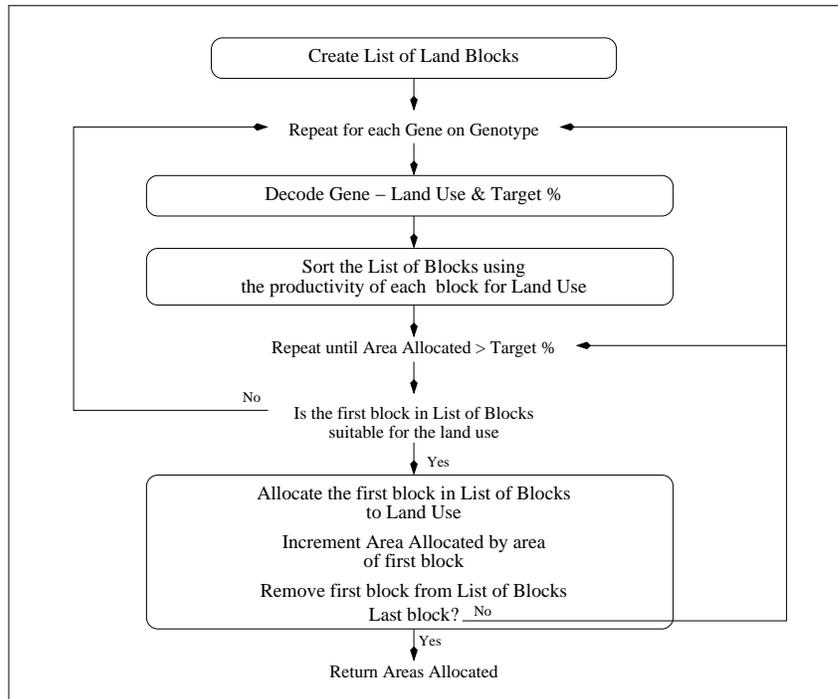


Figure 3.6: The greedy allocation algorithm

3.5 Summary

This chapter has presented the development of two GA representations for land-use planning problems. The representations are a-spatial, with the Land-Block representation being a fixed-length, fixed-order mapping from gene to land-block and the P&P representation using a order-dependent, variable-length representation that makes allocations via a greedy-algorithm. The following chapter presents the implementation of two single-objective GAs using these representations.

Chapter 4

Single-Objective GAs

4.1 Introduction

This chapter presents the single-objective GA (SOGA) land-use planning tools implemented using the two representations (Land-Block and P&P) defined in the previous chapter. The goal for both SOGAs was to find land allocations acceptably close to the single-objective optimum. If both SOGAs found acceptable solutions, their relative efficiency would be considered. The chapter focuses on the GA design choices that were made.

Both the Land-Block and P&P SOGAs use a common underlying structure based on Davis (1991) object-oriented genetic algorithm. Six *phases* may be identified in the operation of the SOGAs as illustrated in Figure 4.1.

- *Initialisation* sets the control parameters for the SOGA and generates the starting population. Population initialisation is specialised for each representation.
- *Run Control* checks if any of the criteria for ending the run have been met; for example the total number of reproductive events or the count of successive reproductive events without a fitness gain having been made.
- *Reproduction* has three operations: selecting the operator, selecting one or two parents depending on the operator to be used and applying the operator to the parents to create the offspring. The P&P SOGA has a larger set of GA-operators to select from, reflecting its more complex representation.

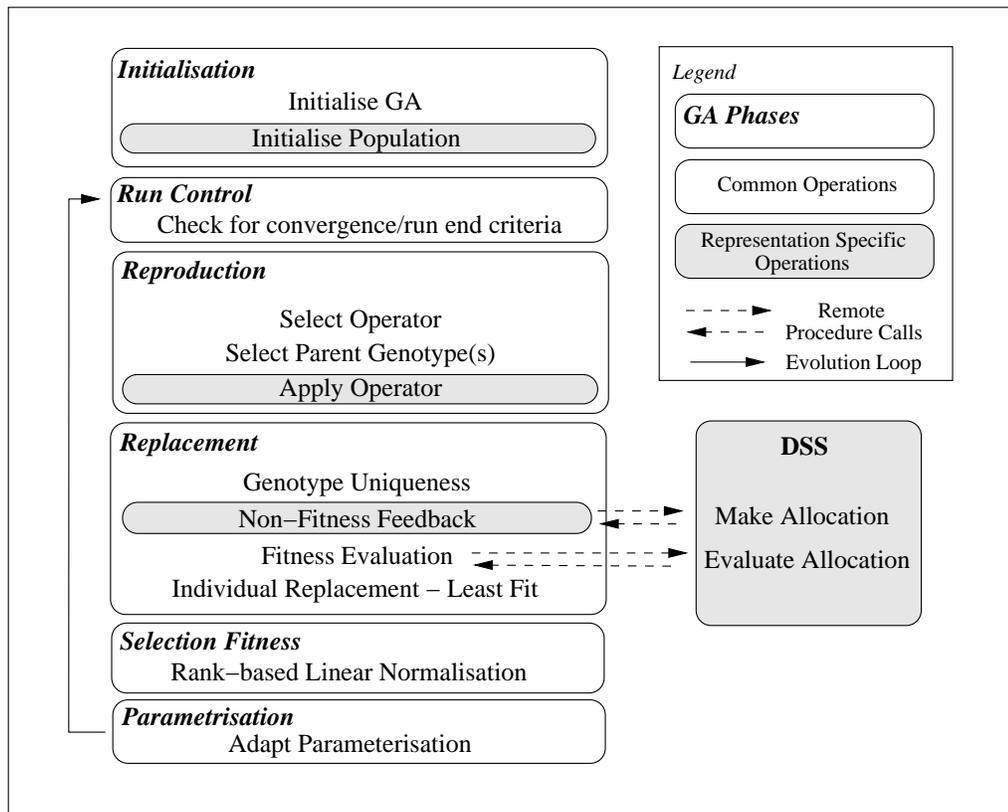


Figure 4.1: Land-Block and P&P SOGA Structure

- *Replacement* is the defining feature of the SOGA's design. Offspring, created in the reproduction phase, only enter the population if they are not duplicates of an existing population member and if they are fitter than the least-fit member of the current population. Generically, the replacement phase has three operations: checking uniqueness, evaluating the fitness of the offspring using the DSS and the replacement, if appropriate, of the least-fit individual. For the P&P SOGA there is a further operation where additional non-fitness information, on the allocations made by the DSS, is fed back to assist the P&P SOGA's search.
- *Selection Fitness* calculation determines a genotype's probability of selection as a parent. The fitness values returned from the DSS are transformed using rank-based linear normalisation.
- *Parameterisation* adapts the control parameters of the SOGA or the GA operators if required.

The following section details the implementation of the generic operations common to both SOGAs. The representation-specific operations are examined in Section 4.3 and the GA-operator sets for the two SOGAs appear in Section 4.4. Section 4.5 details the metrics used in development and evaluation and presents the results of the initial testing. From these initial results, conclusions are drawn which set the context for subsequent chapters.

4.2 Generic SOGA Features

4.2.1 Reproduction

The SOGAs' individual offspring are the product of a single GA operator. By contrast, in classical GAs, reproduction first creates two offspring using crossover, each of which is then subject to mutation (Goldberg 1989). The SOGA approach is termed *independent operator application* (Davis 1991). In this approach each GA-operator stores as a parameter its probability of application. The choice of GA-operator for each reproductive cycle is made using roulette wheel selection. The advantage of independent operator application is that it is possible to track the gains being made by individual GA-operators thus allowing the investigation of the effectiveness of any new GA-operators proposed.

4.2.2 Replacement

The replacement strategy used was elitist individual replacement with genotype uniqueness enforced. This is illustrated in Figure 4.2 and its component operations explained below.

Uniqueness testing

Offspring genotypes produced by the application of the GA operators are first tested to ensure that they do not duplicate an existing member of the population. This test for genotype uniqueness does impose a computational overhead on the operation of the GA. This overhead is, however, offset by the reduced number of genotype evaluations required, since the check for duplication can be performed before fitness evaluation. As will be seen in Section 4.5.2 this saving of fitness evaluations is a significant efficiency gain. Throughout the GA testing considered to date, no additional information would be

gained from the repeated evaluation of duplicate genotypes as there are neither temporal changes nor stochastic noise in the fitness functions.

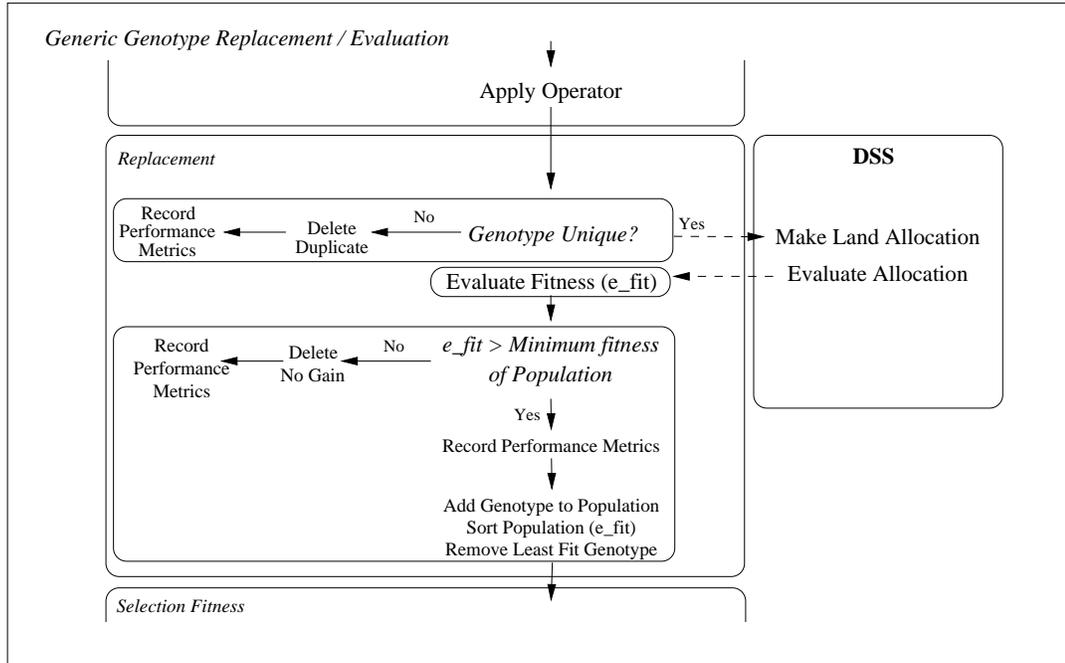


Figure 4.2: Detail of the replacement phase (from Figure 4.1)

In addition to reducing the number of genotype evaluations required, a further benefit of the adoption of a replacement strategy with genotype uniqueness enforced is that such a strategy better preserves genetic diversity within the population. Genetic diversity is significant as crossover-type operators depend on recombining genetically dissimilar genotypes to make fitness gains.

For the land-use planning application, genotype uniqueness is also useful in enabling a single GA run to find not only the best solution but the n best solutions where n is the population size. While finding the n best solutions does involve a significant extra period of search and optimisation after the single objective optima has been found, it was hypothesised that users of the DSS would find it useful to have available a range of high-fitness solutions on which to base their final choice of allocation.

Fitness Evaluation

Following the uniqueness testing, two remote-procedure calls are made to the DSS. The first passes the genotype to the DSS and decodes it into a land allocation. The second

performs the fitness evaluation for the allocation and returns a fitness value. This returned value is the *evaluation-fitness* of the genotype.

Replacement of least fit

Following fitness evaluation, if the evaluation-fitness of the offspring genotype is greater than the minimum fitness of the population, the genotype is added to the population, the population is sorted and the the least-fit genotype removed. This elitist approach ensures that no genotype in the population is displaced by a less-fit individual, eliminating the need to repeatedly recreate genotypes. Having completed replacement, the GA proceeds with setting the *selection-fitness* values used in the subsequent reproductive cycle.

4.2.3 Selection fitness calculation

The evaluation-fitness values returned from the DSS are translated into selection-fitness values using rank-based linear normalisation. This process is illustrated in Table 4.1 where a linear increment of 10 is used to set the selection-fitness values for the genotypes based on their rank.

Table 4.1: Rank-based linear normalisation example

Genotype	A	B	C	D	E	F
Evaluated Fitness	£0.50M	£0.51M	£0.52M	£1.00M	£2.00M	£100.00M
Rank	1	2	3	4	5	6
Selection Fitness	10	20	30	40	50	60

Rank-based linear normalisation was chosen as it provides explicit control on the selection pressure applied to the population and reduces the likelihood that the population will converge prematurely on a sub-optimal solution. Problems caused by super-individuals dominating the selection process are also avoided. For example, based solely on evaluation-fitness, genotype F in Table 4.1 would be selected 96% of the time compared with 29% using rank-based linear normalisation. This is important for both SOGAs, but particularly for the P&P SOGA, as it is possible to have sub-optimal yet highly-fit individual genotypes after initialisation. Linear normalisation also enables the GA to exploit even small differences between genotypes, such as between genotypes A, B and C in Table 4.1. This avoids *genetic-drift*, where the differences in evaluated-fitness between individual

population members are too small to provide adequate bias in the selection process. The number of genes in the Land-Block representation means that the GA must continue, as the population converges, to exploit small differences in evaluation-fitness, for example a change in land use for one land-block of one hundred or more. The selection-fitness values thus calculated, are used to control parent selection via a simple roulette wheel procedure.

The Land-Block and P&P SOGAs can thus be summarised as elitist individual replacement GAs, using rank-based selection with reproduction via independent application of single operators. Having defined the general structure of the Land-Block and P&P SOGAs, the specialisations required by each representation can be examined.

4.3 Representation-Specific Features

Both GAs use specialised population-initialisation methods. The P&P GA is further specialised by its use of non-fitness information fed back from the DSS as part of the evaluation of the allocation.

4.3.1 Population initialisation

Land-Block population initialisation

The number of genes in the genotype and the land use values permitted at each locus are both set as part of population initialisation. The number of land-blocks that make up the land management unit and the land uses suitable for each land-block are returned to the GA using a remote-procedure call to the DSS. The number of land-blocks considered may be less than the total number of land-blocks within the land management unit, since land-blocks with fixed or non-commercial land uses are excluded from the analysis. The suitability information for each land-block is used to restrict the range of initialisation values for each gene. Restricted initialisation ensures that all Land-Block genotypes define biophysically possible patterns of land use for the land management unit. Efficiency of the Land-Block GA is thus increased as it does not have to discover the suitable land uses as part of the optimisation process. Suitability information is also used by the Land-Block mutation operator.

P&P population initialisation

Initialisation for the genotypes of the P&P population creates a fixed number of genes per genotype (five genes were used in testing). Gene initialisation selects:

- Land Use - any of the 10 land uses represented by the DSS.
- Target Percentage - randomly chosen from {5%, 10%, 15%, ... 100%}.

This initialisation tends to over-specify the allocation but ensures that, in general, complete allocations are considered. Over- or under-allocation of initialised P&P genotypes is dealt with using the non-fitness information returned from the DSS as part of genotype evaluation and is discussed next.

4.3.2 Non-fitness feedback to the P&P SOGA from the DSS

Non-fitness feedback is the return from the DSS of information in addition to the usual evaluation-fitness values. For the P&P SOGA the additional information returned is the percentage of the land area allocated by the DSS for each P&P gene. This information is used to determine if an offspring genotype is *functionally identical* to an existing member of the population. Functionally identical genotypes occur when, despite being genetically distinct, a genotype results in the same allocation as an existing member of the population. In certain GA applications, for example some multi-objective GAs, it is possible and useful to have distinct genotypes resulting in solutions with identical features. In the case of the P&P GA, however, the causes of functionally identical genotypes mean that their retention within the population would simply reduce the effective genetic diversity of the population. If an offspring genotype is determined to be functionally identical then it is eliminated.

Causes of functionally identical genotypes

Functionally identical genotypes occur for the following reasons.

- Over-allocation occurs when a gene specifying, for example, 5% of a land use results in an allocation of 10%. This occurs because the greedy-algorithm iteratively allocates fields, ordered by productivity, until the target-percentage is exceeded. Allocation of a large field near completion of the target-percentage can result in over-allocation.

- Under-allocation, this usually arises because there are insufficient land-blocks available to the greedy-algorithm for which the land use is suitable. Under allocation may also result from initialisation when the sum of the target-percentages is greater than 100%.

An example of under-allocation is presented in Figure 4.3. Suppose the first **Wheat** gene can achieve only 7% of a target of 15% when 10%:Barley has already been allocated. If we consider for the moment only the first two genes of the genotype, any genotype with a 10+% **Wheat** target percentage in the second gene after an initial gene specifying 10%:Barley, (e.g., 10%:Barley, 15%:Wheat or 10%:Barley, 90%:Wheat), will result in an allocation of 12%:Barley and 7%:Wheat and therefore be functionally identical.

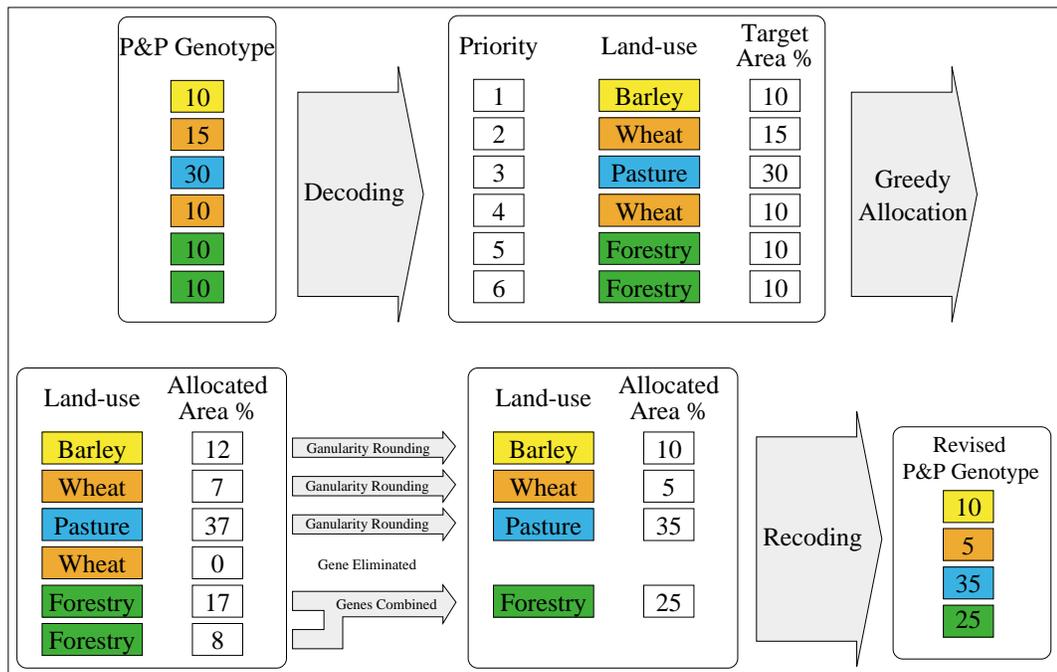


Figure 4.3: P&P Non-Fitness Feedback Mechanisms

- Failure to make any allocation, is the extreme case of under-allocation, for example the second wheat gene in Figure 4.3. These zero-valued genes are termed *parasitic* as they contribute nothing to the fitness of the genotype.
- *Gene-pairs* are two consecutive genes with the same land use, for example the two

forestry genes in Figure 4.3. Such gene-pairs are functionally identical to a single gene with the sum of their target-percentages. Gene-pairs may be created at initialisation or by the actions of GA operators.

Detection and elimination of functionally identical genotypes

To detect and eliminate functionally identical genotypes four operations are added to the replacement phase of the P&P SOGA, see Figure 4.4.

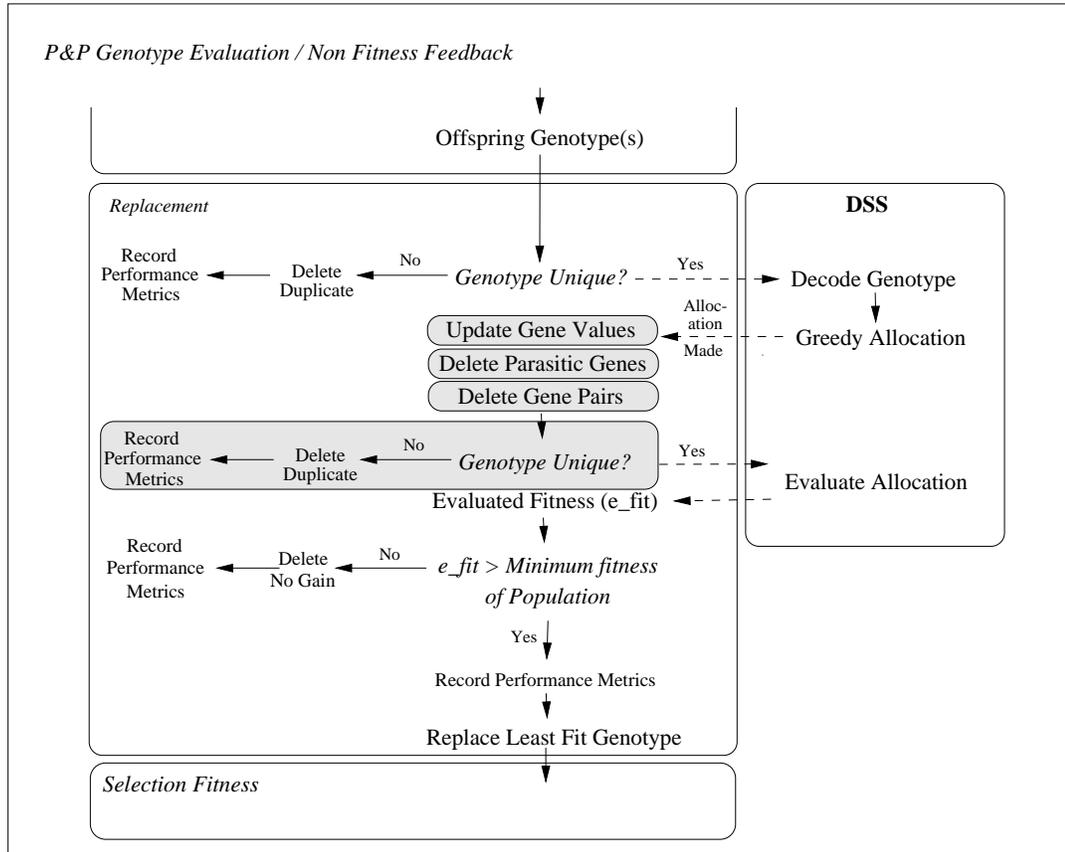


Figure 4.4: Fitness Evaluation and Non-Fitness Feedback

First the target-percentages in each gene are updated using the percentage values returned from the DSS. This is sufficient to allow the P&P SOGA to detect, using the second uniqueness check, functionally identical genotypes caused by over- or under-allocation. Parasitic-genes and gene-pairs, however, require structural changes to the genotype. Parasitic-genes are deleted and the genes to either side concatenated to form a new genotype. Gene-pairs are merged by updating the target-percentage of the first gene

to be the sum of the pair’s target-percentages. The second gene is then eliminated as a parasitic-gene. The second test for uniqueness, using the updated and repaired genotype, is made before evaluation and possible insertion into the population.

4.4 Operators

Having set out the framework of the two GAs it is now possible to examine the two sets of operators presented in Table 4.2. The operator set for the Land-Block representation GA reflects its relative simplicity with just two operators. The P&P representation, by contrast, with its variable-length, order-dependent interpretation of genes and more complex individual genes was implemented with eight operators.

Table 4.2: Operator sets for each representation

Type	Land-Block	Percentage and Priority
Binary	Uniform-Crossover	Uniform-Crossover Partial-Relative-Reordering Splice
Unary	Mutation	Type-Mutation Non-uniform Mutation Pair-Swap Insert-Gene Delete-Gene

4.4.1 Operators used by both representations

The only operator common to both representations is *uniform crossover* (Syswerda 1989). This was chosen for its reported robust high-performance across a range of practical applications (Davis 1991). The operator was implemented using a *crossover mask* defining the source of genes for each offspring. For the P&P representation, where genotypes may be of unequal length, the crossover is performed only between the initial segment common to both parents. Crossover is thus performed between the highest priority genes.

4.4.2 Land-Block specific operators

The Land-Block mutation operator simply replaces the current land-use value of the gene with one chosen at random from the set of suitable land uses for the land-block to which the gene is mapped.

4.4.3 P&P specific operators

Binary Operators

Based on order-based uniform-crossover (OBXO) (Davis 1991) a binary permutation operator (*Partial-Relative-Reordering, PRR*) was implemented. As the P&P representation is order-dependent, but not a permutation of an identical set of genes, PRR performs its reordering only on the genes common to both parents. For the P&P representation, common genes are those seeking to make an allocation to the same land use. The operation of PRR is illustrated in Figure 4.5. For the two parent genotypes, the genes common to both are B,C and D. These genes are reordered, to the order that they appear in the other parent, leaving the other genes unaltered. As with OBXO, the reordering is relative rather than absolute and seeks to exploit similarities in the relative priorities of genes rather than their absolute priority as determined by their positions on the genotype.

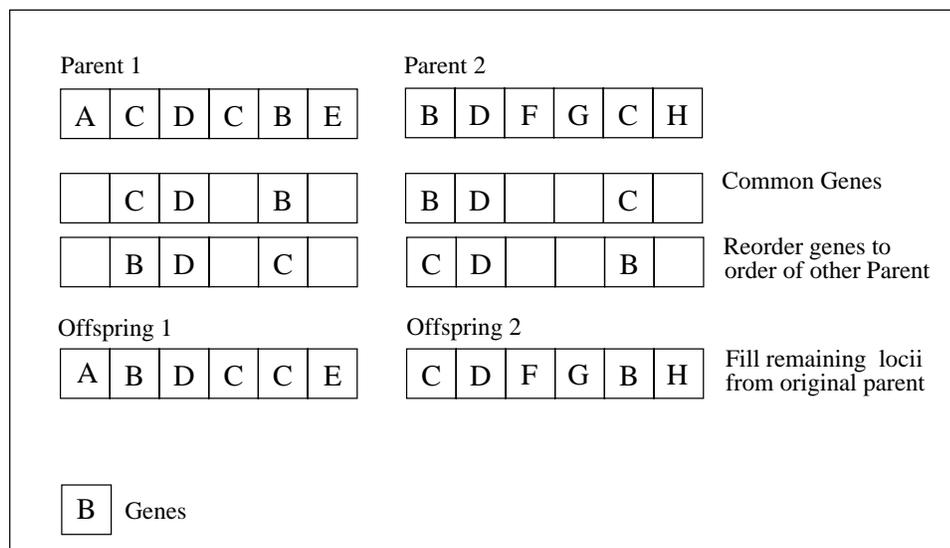


Figure 4.5: PRR Operator

The final binary operator used in the P&P operator set is *Splice*. This operator em-

ployed by (Goldberg et al. 1993) concatenates two genotypes to form a single offspring. This provides a mechanism for genotypes to grow in length and thus is a means for partial allocations to be completed. In Goldberg et al.'s messy GA, a second operator *Cut* was also employed. This was not thought necessary for the P&P representation as it would result in partial allocations that would not be permitted to enter the population due to the elitist replacement strategy adopted.

Unary Operators

The P&P gene has three elements: position determining the priority of the land use; gene class determining the land use and a float-parameter determining the target-percentage. A mutation operator was implemented for each of these elements.

- Priority is mutated using *Pair-Swap* based on the *2-Opt* operator (Cox et al. 1991). This exchanges the genes at two loci chosen at random.
- The gene class, and thus the allocated land use, is mutated using *Type-Mutate*. This operator creates a new gene from the list of available land uses, sets the target-percentage of the offspring from the parent-gene and replaces the parent-gene at the same locus.
- The target-percentage is mutated using *Non-Uniform Mutation* (Michalewicz 1992). This operator increases or decreases the target-percentage by an increment. The minimum size of the increment is the granularity of the target-percentage, a parameter of the representation. In testing a granularity of 5% was used. The maximum and minimum values for the target-percentage, also parameters of the representation, set the range of values within which the target-percentage may be mutated. The direction and magnitude of the increment are randomly generated, with the probability of larger mutations being reduced over the course of the run. This reflects the need for larger exploratory mutations earlier in the run and fine-tuning of the target-percentages later. The magnitude of a mutation μ is a function of the range of values y and when the mutation occurs, t . The value of y is either *Maximum - Current* or *Current - Minimum* depending on the direction of mutation. The time when a mutation occurs, t , is measured using the current count of fitness gaining events.

The size of mutation is thus given by equation 4.1,

$$\mu(t, y) = y * \left(1 - r^{(1 - \frac{t}{T})^b}\right), \quad (4.1)$$

where r is a random number from $[0 \dots 1]$, T is the maximum number of fitness gaining events and b is a parameter determining the degree of non-uniformity in the distribution of mutation values. Increasing values of b mean more small magnitude mutations.

- While the splice operator serves to increase genotype length it cannot add genes other than at the end of the genotype. A further operator *Insert-Gene* adds a single gene at a random location on the genotype thereby potentially increasing the number of land uses present within the land-use plan. Complementary to insert-gene is the final operator *Delete-Gene*. This removes a single gene from a random location on the genotype. This operator was implemented to promote lower-priority land uses that were being blocked by higher priority but under-performing genes.

4.5 Initial SOGA Evaluation

The criteria chosen for the evaluation of the performance of the GAs were the quality of solutions found and the GA's efficiency of learning. The test application used for this initial testing was a financial optimisation problem, maximising the net-present-value (NPV), for 65 of the land-blocks of the Hartwood Research Station presented in Chapter 1.

4.5.1 Evaluation metrics

Comparison of the quality of solutions found by the GAs is made using the evaluation-fitness of the highest-fitness genotype in the population (*MaxFit*) and the average fitness for the population (*AvgFit*). AvgFit is useful as it gives an indication of overall quality of the population and, compared with MaxFit, is a pointer to the degree of convergence towards the optimum that has been achieved. Five further metrics were used to look at the efficiency of learning for the two GAs. The first four are counts of reproductive operator events.

- All events (*All*).

- Events that result in fitness gains (*Gain*).
- Events where no fitness gain is made (*NoGain*).
- Events where duplicate genotypes are created (*Duplicates*).

The fifth efficiency metric was the *CPU* time used. This includes both the GA operations and the genotype fitness evaluations by the DSS. Both are included, as the allocation process for the P&P genotype is considerably more CPU-intensive than the Land-Block, requiring repeated sorting of the land-blocks to be allocated while the Land-Block allocation process is simply a matter of updating parameter values.

4.5.2 Results from initial testing

The initial test runs were conducted using populations of size 30 and with a balance between binary and unary operators initially set at 0.65/0.35 and adapting over the course of the run to 0.5/0.5. The probabilities adapt both to how far the run had progressed towards the maximum number of reproductive events and the number of reproductive events since a fitness gain was made. This ensured that, should the population converge well before the maximum run length, the population would still have been exposed to the higher levels of mutation. For the P&P GA, where more than one binary and unary operator exists, the probability of application is equally divided between operators.

The evaluation metrics were collected over the course of 25 runs to provide the time series presented in Figure 4.6. The values at termination of the GA runs were averaged and are presented in Table 4.3. The significance (*Sig*) of the differences (*Diff*) between the values for the two GAs were tested using a *Monte-Carlo sample-difference test* (Noreen 1989). This test estimates the probability that the Diff value occurred by chance. The estimate is made by repeatedly drawing two equal-sized samples, at random, from a population incorporating both the Land-Block and P&P runs. After each draw the Diff value between the samples is calculated and compared with that observed between the two representations. Sig is the proportion of the randomly-drawn samples whose Diff value exceeds the observed Diff. The Monte-Carlo sample difference test was used since the distribution of fitness and other metrics was not normal.

Quality of solutions for the Land-Block and P&P SOGAs

The graphs in Figure 4.6 show that the two GAs achieved similar results but via different routes. This was borne out by the results in Table 4.3. For MaxFit and AvgFit values achieved there is no significant difference ($\text{Sig} \leq 0.005$) between the Land-Block and P&P SOGAs. Both SOGAs for MaxFit achieved 95% of the known optimum value. The optimum returns are known since they are simply the sum of the best returns for each land block. The lack of inter-block dependencies for the financial returns metric means that the best performing land use for each block can be determined by simply sorting the possible land uses by their performance for the evaluation metric.

The number of reproductive events required to achieve this performance for financial returns was, however, significantly different with the P&P SOGA outperforming the Land-Block for the count of All events. As there was no statistically significant difference between the average fitness values of the initialised population, the P&P SOGA was making fewer but larger gains.

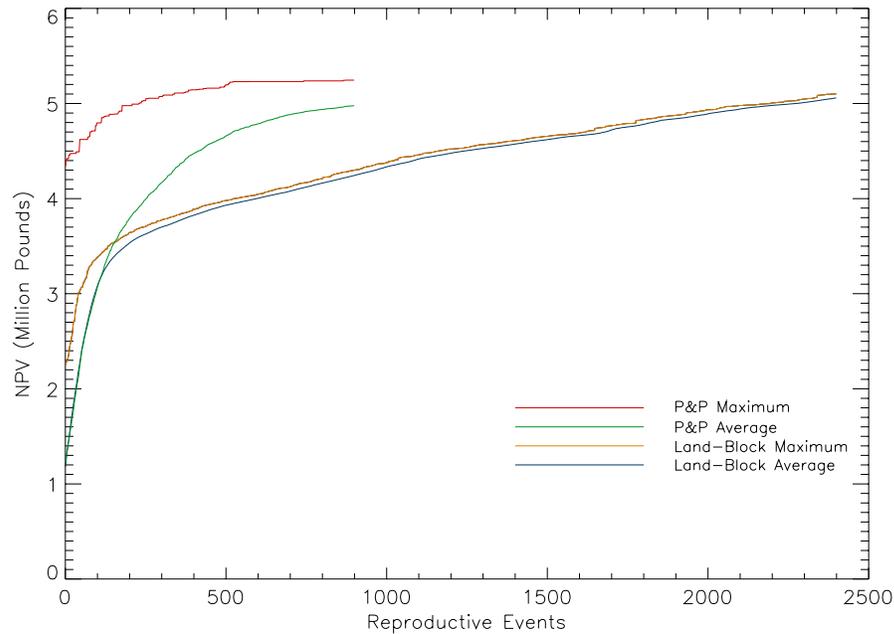


Figure 4.6: Land-Block and P&P GA performance graphs

Table 4.3: Land-Block and P&P GA performance

Metric	Land-Block	P&P	Diff	Sig
MaxFit(£M)	5.10	5.24	-0.14	0.093
AvgFit(£M)	5.06	4.98	0.08	0.219
All	2431	931	1500	0.000
Gain	1312 (54%)	156(17%)	1156	0.000
NoGain	391 (16%)	354(38%)	37	0.196
Duplicates	726 (30%)	421(45%)	305	0.000
CPU (sec)	1762	556	1206	0.000

Efficiency of learning for the Land-Block and P&P SOGAS

For computational efficiency the P&P GA again outperformed the Land-Block with significantly lower cpu time required to achieve comparable levels of fitness. Examining the breakdown of the proportions of reproductive events in Table 4.3 provided insights into the operation of the two GAs.

- The Land-Block GA has a larger proportion of fitness gaining events. This reflects the utility of restricting gene values to those land uses that are suitable for each land-block. There are no such restrictions on the initialisation or mutation of the P&P genes.
- The proportion of duplicating events was larger for P&P. This was an anticipated result of using a 5% granularity for the target land-use percentage and the repair mechanisms where any genotype over-allocating a land use has that percentage reduced to the maximum possible. This will often result in the creation of genotypes already existing in the population, these being eliminated at the second uniqueness check.

4.6 Summary and Conclusions

This chapter has detailed the basis for the design and implementation of two GAs based on the representations proposed in Chapter 3. Both GAs share a common GA framework of rank-based, individual replacement with independent-operator application. The GAs are differentiated by their initialisation and use of non-fitness information and by the operator sets they employ. The metrics used in the initial testing of the performance of

the two GAs were detailed and the results of this testing presented. Given the acceptable performance of both GAs for quality of solutions found it was decided to pursue both representations as the basis for multi-objective GAs, in Chapter 5. While the P&P GA was more efficient the utility of the P&P greedy allocation algorithm when applied to multi-objective optimisation was uncertain. It was also concluded that the larger number, and non-standard nature of the operators employed by the P&P GA meant that there was uncertainty about the effectiveness of individual operators and their parameterisation, this is addressed in Chapter 6. Finally the need to test how well both GAs handle larger scale problems was identified. The results of this further testing are presented in Chapter 7.

Chapter 5

Multi-Objective GAs

5.1 Introduction

In this chapter the Land-Block and P&P SOGAs are modified and extended to address land-use planning problems with two or more objectives. For multi-objective GAs (mGAs) the goal is no longer to find a single near-optimum solution but to evolve a population of land allocations that define the structure of the relationship between the objectives, in most cases a trade-off. While the Land-Block and P&P mGAs share many features with their SOGA equivalents, the new goal requires the specialisation of several procedures.

- The calculation of selection-fitness is still rank-based, but the basis of ranking is changed and the process also incorporates a mechanism to ensure that the individual genotypes of the final population are evenly-spaced to ensure an adequate characterisation of the trade-off. The size of the mGA population determines the distance between genotypes along the trade-off, assuming the genotypes are evenly-spaced. Setting the mGA population size, during initialisation, is thus a balance between the desire for detailed characterisation of the trade-off and computational load.
- To reduce the number of non-viable offspring produced, the parental selection process, for binary operators, is modified to bias selection in favour of similar genotypes.
- To assist the mGA in finding the single-objective optima, located at the extremes of the trade-off, a further initialisation phase, termed *population doping* is added.

As the goals of the mGAs differ from those of the SOGAs, a wider range of evaluation metrics is needed to compare their performance. The metrics adopted are set out in this chapter followed by the test application used for development and initial testing. The results of this testing and conclusions complete the chapter.

5.2 The mGA Goals

The goal of the land-use planning mGAs is to establish the structure of the relationship between two or more non-commensurable objectives. For applications in the land-use planning domain, the relationship will be some form of trade-off since, in nearly all cases, objectives will be in conflict. For example, a more diverse pattern of land use may have environmental benefits but these are often gained at the cost of reduced financial performance. Conflict can be established *a priori*, either intuitively or by comparing the optimum solutions found by a SOGA for the individual objectives. Since for land-use planning, objectives are influenced by the spatial organisation of natural resources, the degree of trade-off, or conflict, is often unclear.

An idealised terminal population, resulting from an mGA applied to a two-objective problem, is illustrated in Figure 5.1. The Utopian solution, where all objectives are optimal, lies in the infeasible region, and thus cannot be realised. In the absence of further preference information, the best the mGA can achieve is to find non-dominated (or Pareto-optimal) solutions. Non-dominated solutions are each evaluated as no less fit, for all objectives, than any existing member of the population.

The number of possible non-dominated solutions is usually larger than the size of the mGA population, therefore the goal of the mGA is to find a representative subset that defines the shape of the trade-off between the objectives. The two single-objective optima are by definition non-dominated, each maximising one of the objectives at the expense of the other. These points define the ends of the trade-off between the objectives and should also be found by the mGA. To best indicate the form of the trade-off between these points, the subset of non-dominated solutions should be evenly spaced within the range of objective values found to be Pareto-optimal. The mGAs exploit the population-based nature of GA search to evolve the subset of evenly-spaced non-dominated solutions, across the trade-off in a single run.

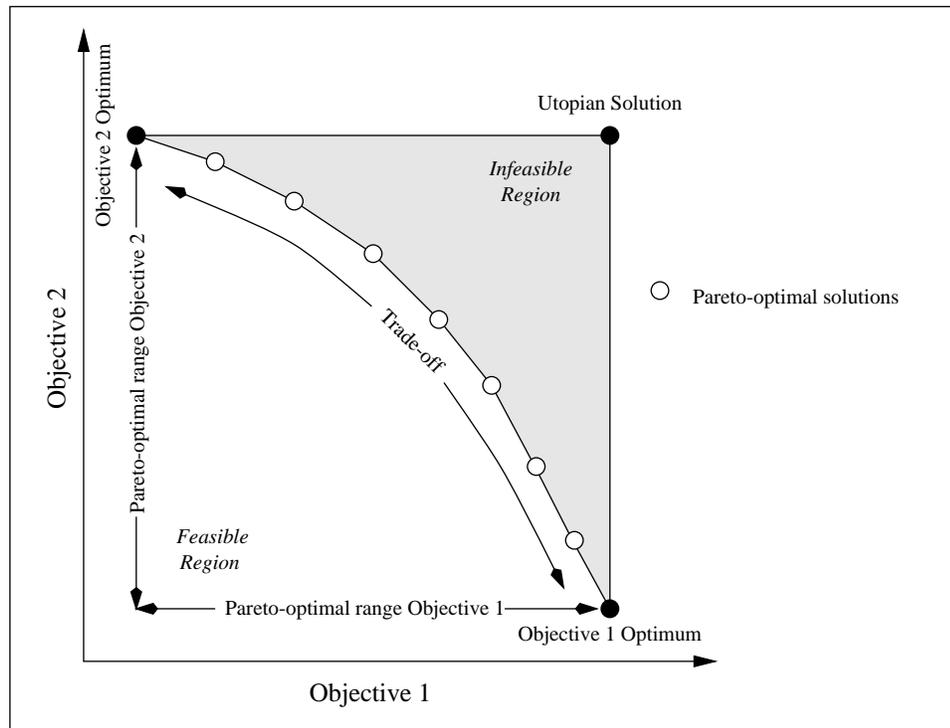


Figure 5.1: Two-objective maximisation example application

5.3 mGA Implementation

Figure 5.2 shows the structure of the mGAs. The new or modified elements required to achieve the mGA goals are highlighted. These components are dealt with in the following sections: Selection Fitness in Section 5.4; Parent Selection in Section 5.5 and Initialisation in Section 5.6. The features inherited from the SOGAs and the integration of the new components completes the discussion of the mGA implementation in Section 5.7

5.4 Determining mGA selection-fitness

5.4.1 Dominance ranking

The fundamental change between the SOGAs and the mGAs is how selection-fitness is determined. In the case of the SOGA, selection-fitness is calculated by ranking the genotypes of the population using their single evaluated-fitness value and then using a linear normalisation function to set the selection values. The rank-based approach to setting fitness values is maintained in the mGAs, but uses *dominance-ranking* (Fonseca & Fleming 1995b).

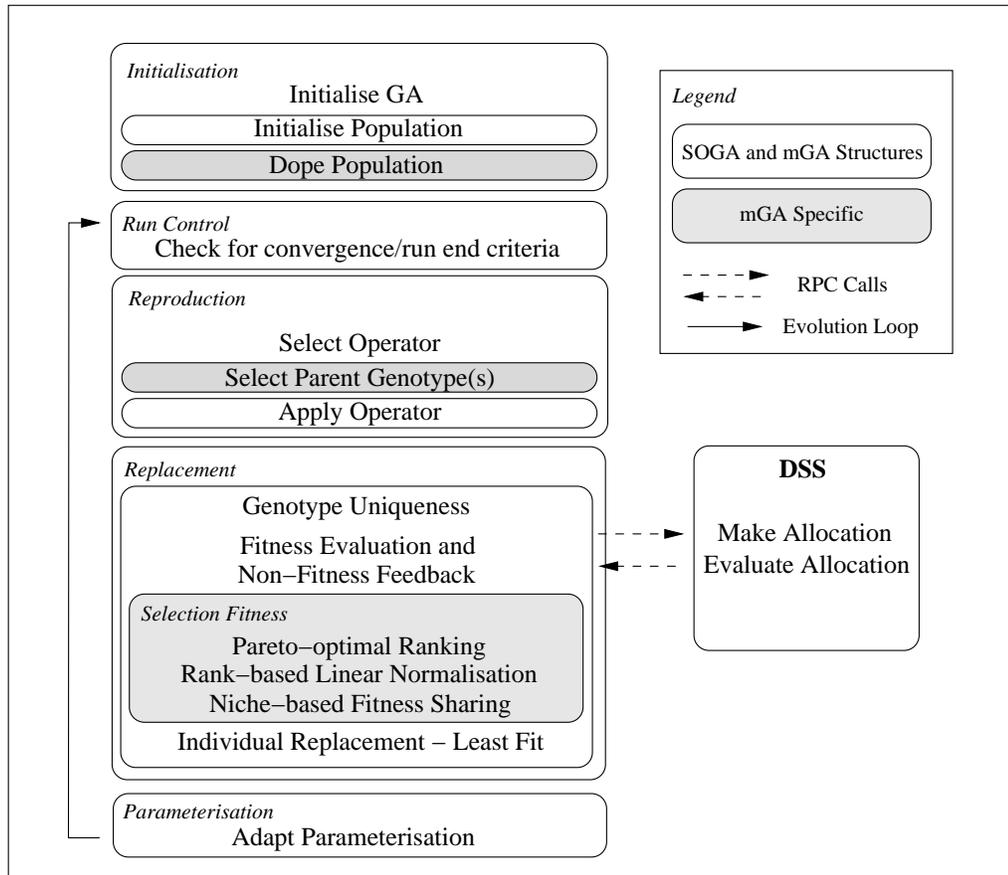


Figure 5.2: mGA Structure

For dominance-ranking, the rank of a genotype is the count of the genotypes in the population evaluated as performing better for at least one objective, with equal or better performance for all other objectives. Figure 5.3 shows six members of a population. The three unshaded genotypes are dominated to a greater or lesser degree (ranks of five to one). The three shaded genotypes are not dominated and receive the best rank of zero. The genotypes with rank zero are Pareto-optimal. Using dominance-ranking enables the multiple-evaluation fitness values to be reduced to a single fitness metric that is compatible with the GA, without the need to resort to weighting-based methods.

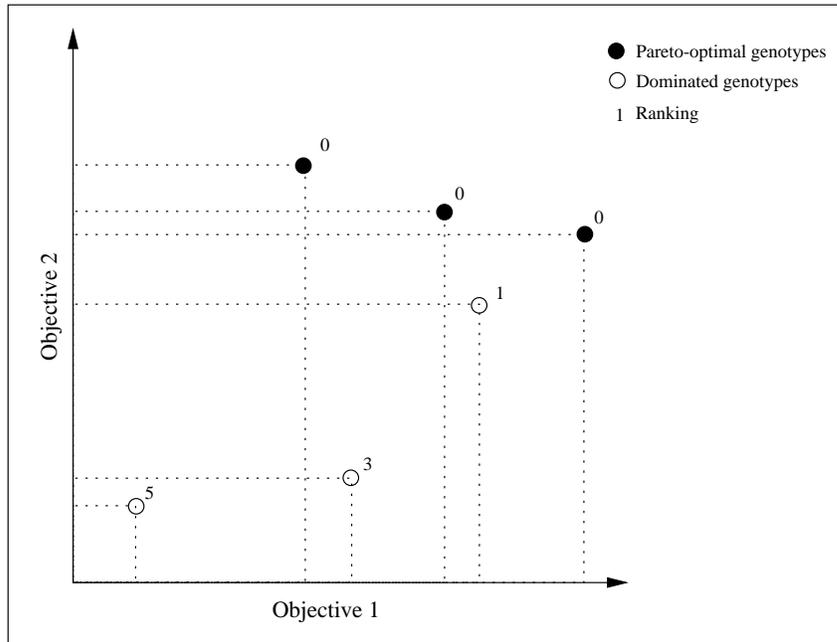


Figure 5.3: Pareto-optimal ranking example

The dominance-ranks are translated into selection-fitness values using the same rank-based linear normalisation methods employed with the SOGAs. As with the SOGAs, genotypes sharing the same rank have their fitness values averaged. Table 5.1 presents this process for the population in Figure 5.3 using a linear increment of 10.

Table 5.1: Selection fitness using dominance rank for population in Figure 5.3

Dominance-rank	5	3	1	0	0	0
Raw Linear-normalised Fitness	10	20	30	40	50	60
Selection Fitness	10	20	30	50	50	50

5.4.2 Fitness sharing

The use of dominance-ranking will, over the course of the GA cycle, result in a population composed of non-dominated solutions. It will not on its own, however, ensure that the population is evenly spread across the trade-off. To achieve this goal the selection-fitness calculation is augmented with *niche-based fitness sharing*. The concept of niches is borrowed from ecology, with an ecological niche being the environmental space within which an organism exists. For the mGA it is the genotypes in the population that occupy niches defined in the search space. Figure 5.4 shows five niches across the range of values for a single objective. The dots indicate five genotypes unevenly distributed across the niches.

In the ecological case, too many organisms in a niche tends to weaken the individuals until numbers are reduced to a sustainable level. If two or more genotypes exist in a niche then their fitness is reduced (or shared), making it more likely that one of them will be eliminated as the least-fit member of the population. By this mechanism, mGA evolution is biased in favour of genotypes that are progressively more evenly distributed across the trade-off front. Indeed once all genotypes within the population are non-dominated the bias in selection-fitness produced by sharing is the driving force behind the continued evolution of the population.

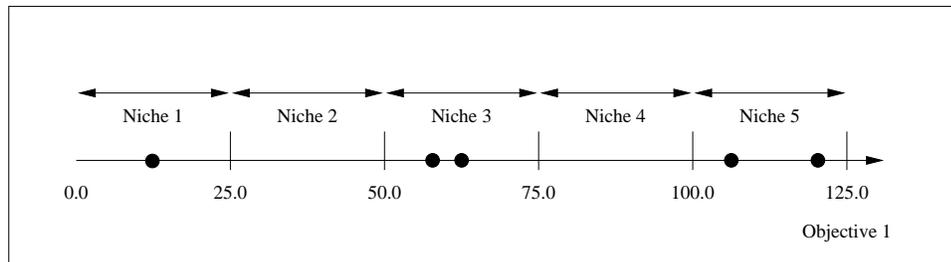


Figure 5.4: Genotypes distributed within niches

The fitness sharing implemented within the mGAs is that proposed by Horn et al. (1994). In this approach the distance between genotypes determines the degree of mutual fitness sharing. The distance measure used is the ∞ -norm distance, the maximum difference between the individual coordinates, as proposed by Fonseca & Fleming (1995b). The coordinates used in the ∞ -norm distance calculation are derived by normalising the evaluated-fitness values for each objective, across the population. The coordinates are thus defined in terms of phenotypic units, differences in fitness evaluations, rather than

the alternative genotypic units, differences at the genetic level. This choice was made since the primary interest was finding a range of alternative land-use patterns across the trade-off front, rather than alternative land-use patterns achieving the same performance.

The number and size of niches is set as part of the mGA initialisation and is detailed in Section 5.6. Sharing is conducted pairwise between genotypes with the same dominance rank. Within the distance defined by the niche-size, genotype pairs mutually reduce each other's fitness, using the triangular sharing function proposed by (Horn et al. 1994).

$$P_{AB} = 0.5 + 0.5 * \frac{D_{AB}}{\sigma_{share}} \quad (5.1)$$

P is the proportion by which the linear-normalised fitness values, of the genotype pair A and B, are multiplied, to give their respective selection-fitness values. D_{AB} is the ∞ -norm distance between the pair and σ_{share} is the niche-size chosen. Genotypes beyond this distance do not affect each others selection fitness.

An example of this niche-based fitness sharing is shown in Figure 5.5 and tabulated in Table 5.2. The genotypes phenotypic-coordinates are indicated on the outer axes in Figure 5.5 with their normalised coordinates shown in brackets beside each genotype. The phenotypic and normalised coordinates are also tabulated as coordinate pairs in Table 5.2. The ∞ -norm distances between the genotypes are shown in Table 5.3.

Since the four genotypes ABCD in Figure 5.5 are all non-dominated, dominance-rank zero, they have their rank-based selection fitness values, based on a linear increment of 10, averaged to 25. Only genotypes B and C are within the niche size of 0.2 chosen for the example. Using their ∞ -norm distance of 0.1, and the niche size of 0.2 in Equation 5.1 results in a sharing proportion between genotypes B and C of 0.75, and selection-fitness values of 18.75, Table 5.2

The role of selection-fitness calculation in guiding the evolution of the mGA population is therefore clear. Dominance-ranking ensures that selection is biased in favour of solutions that, in the absence of further preference information, can be regarded as equally acceptable. Niche-based sharing of the selection-fitness, meanwhile, biases selection in favour of genotypes evenly distributed across the trade-off. This is accomplished by reducing the fitness of those genotypes located at distances below the niche size, making them more likely to be eliminated as the least-fit genotype in the population.

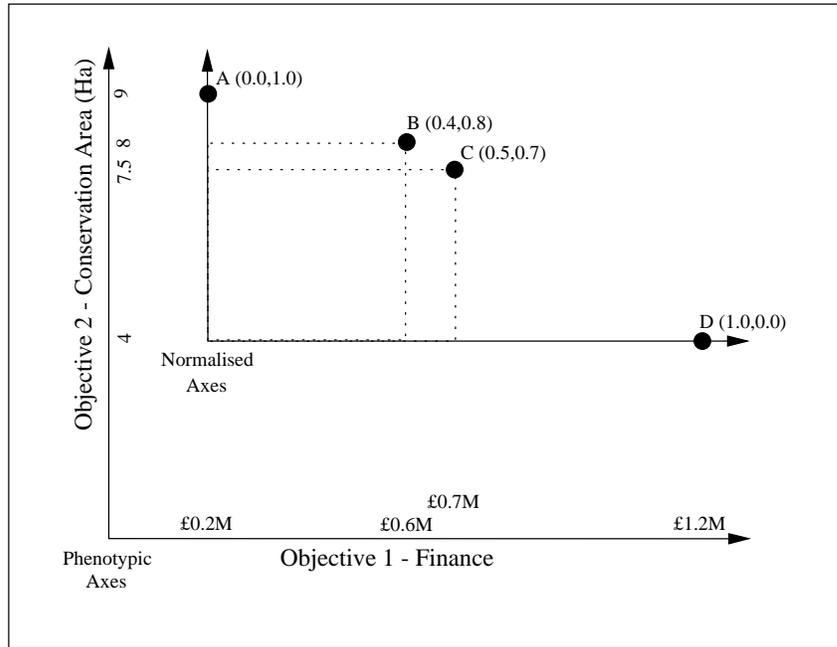


Figure 5.5: Niche-based sharing example

Table 5.2: Fitness sharing, for population in Figure 5.5

Genotype	A	B	C	D
Phenotypic Coordinates	(0.2,9.0)	(0.6,8.0)	(0.7,7.5)	(1.2,4.0)
Normalised Coordinates	(0.0,1.0)	(0.4,0.8)	(0.5,0.7)	(1.0,0.0)
Dominance-rank	0	0	0	0
Raw Linear Normalised Fitness	10	20	30	40
Linear Normalised Fitness	25	25	25	25
Sharing Proportion	-	0.75		-
Shared Fitness	25	18.75	18.75	25

Table 5.3: ∞ - norm distances, for genotype pairs in Figure 5.5

A	B	C	D	
-	0.4	0.5	1.0	A
	-	0.1	0.6	B
		-	0.5	C
			-	D

5.5 Mating restrictions

5.5.1 Mating restriction objectives

A further difference between the SOGAs and the mGAs is the use of mating restrictions on parent selection when employing binary GA-operators. This specialisation of the parent selection methods is highlighted in Figure 5.2. Mating restrictions are used to bias selection of the second parent towards genotypes in the same region of the trade-off as the first, termed *in-breeding*.

The use of in-breeding is based on the heuristic that a niche between existing genotypes is more likely to be filled by recombining parental genotypes from niches close to the unfilled niche than those more distant from it. Figure 5.6 shows four non-dominated members of the population, marked W to Z and the locations of three genotypes, on the trade-off, as yet not found by the mGA, marked 1 to 3. If genotype X is selected as a parent, then mating restrictions mean that it is more likely that parent Y will be selected as the second parent. This is arguably beneficial as it is more likely that genotype 2 will be discovered by crossing X and Y than in crossing X with either W or Z.

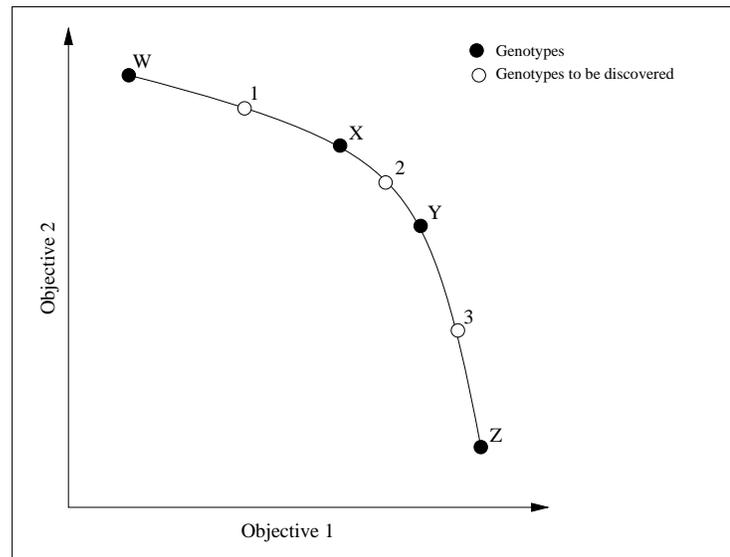


Figure 5.6: Mating restrictions example

Mating restrictions should thus result in both a better characterisation of the trade-off, with more of the niches filled, and a more efficient mGA, since fewer dominated offspring should be created.

5.5.2 Mating restriction implementation

Mating restrictions are implemented as follows. The first parent is selected normally using the selection-fitness values determined by the dominance-ranking and niche-based fitness sharing. Having selected the first parent, a second fitness value, the *mating-fitness*, is calculated for each of the other genotypes as follows. Using the normalised coordinates, calculated in the niche-based fitness sharing operation, the distances from the first parent genotype to all the other genotypes in the population are calculated. If the distance is less than the mating-distance, a parameter of the mGA, then the mating-fitness is simply the selection-fitness. If the distance is greater than the mating-distance the mating-fitness value is reduced proportionally as follows:

$$M_A = S_A * (1.0 - D_{AB}^2), \quad (5.2)$$

where M_A is the mating-fitness of the genotype A being considered as the second parent, S_A is it's selection-fitness, and D_{AB} is the $\infty - norm$ distance between genotype A and genotype B , the genotype selected as the first parent.

Table 5.4 presents a mating-fitness calculation, extending the selection-fitness example from Table 5.2. Supposing genotype B is selected as the first parent and the mating-distance limit is 0.2. The mating-fitness value for genotype C is not affected since the $\infty - norm$ distance, from Table 5.3, is 0.1. Genotypes A and D have their selection-fitness reduced by applying Equation 5.2.

Table 5.4: Mating-fitness calculation for population in Figure 5.5

Genotype	A	B	C	D
Phenotypic Coordinates	(0.2,9.0)	(0.6,8.0)	(0.7,7.5)	(1.2,4.0)
Normalised Coordinates	(0.0,1.0)	(0.4,0.8)	(0.5,0.7)	(1.0,0.0)
Shared Fitness	25	18.75	18.75	25
$\infty - norm$ Distance from B	0.4	-	0.1	0.6
Proportional Reduction	0.16	-	-	0.36
Mating-Fitness	21	-	18.75	16

The effect of the mating restrictions in this example is to bias selection away from genotype D which suffers the largest reduction in selection probability. For the remaining genotypes there is a balance between the reduction in selection-fitness suffered by genotype

A as it is outwith the mating-distance and the previous reduction in the selection-fitness of genotype C by niche-based sharing. Genotype A is the most likely second parent and this is likely to result in a genotype to fill a niche between B and A. Crossing of B and C may also be desirable as it could result in a genotype that outperforms either of its parents and go on to dominate the niche currently shared by B and C.

More generally, mating restriction as implemented here has the effect of biasing the selection in favour of genotypes within the region defined by the mating-distance parameter but does not preclude the pairing of those more widely spaced. The mating-distance used to date has been 0.25, a compromise value based on the need not to excessively restrict the initial exploration of the search space by the mGA while encouraging the in-breeding of similar genotypes later in the mGA run.

5.6 Initialisation

5.6.1 Population sizing

The populations are sized using the Fonseca & Fleming (1995a) formulation, detailed in Chapter 2. For the mGAs considered here, the population-size is determined based on the niche-size chosen. Table 5.5 shows the increase in population-size resulting from the increasing number of objectives or from reducing the niche-size. For the two-objective applications tested, a niche-size of 0.1 was chosen with a population-size of 21. This was consistent with the need for both an adequate visualisation of the trade-off while remaining within the limits of the computer resources available.

Table 5.5: Population sizes for combinations of niche size and number of objectives

Niche Size	Number of Objectives		
	2	3	4
0.25	9	61	369
0.1	21	331	4641
0.025	81	4921	265761

The populations for the mGAs are fixed-size and unstructured with genotype uniqueness enforced. As with the SOGAs, the enforcement of genotype uniqueness both increases efficiency, by eliminating redundant genotype evaluations, and increases the genetic diversity within the population that may be exploited by the mGA operators.

5.6.2 Population doping

In addition to random initialisation of the SOGAs it was decided to add a further phase to the initialisation, prior to the evolution of the population. In this phase, genotypes previously found to have high performance for one of the objectives are added to the population as candidate solutions. This is termed the *doping phase*, Figure 5.2. To date, the genotypes used for doping have simply been mono-cultures of cattle, a land use with few suitability restrictions and one of the best financial performances. Other sources of genotypes for doping could include: the current pattern of land use on the farm, thus ensuring that the solutions found by the mGA are no worse than the existing strategy; solutions found by SOGAs or patterns proposed by land-management experts. The mGAs do not depend on the doping process but it does assist the mGA search by adding points at the extremes of the trade-off that may be recombined with other members of the population to form intermediate solutions.

5.7 Features in common with SOGA

The three additional or modified operations, selection fitness, mating restriction and population doping can now be placed in the context of the GA structure inherited from the SOGAs, see Figure 5.2.

The mGAs share the SOGAs' strategies for replacement, operator application and non-fitness feedback. The most significant of these, for the mGA design, is the use of individual replacement. For the mGAs, the new genotype may make a fitness gain in two ways. In the initial phases of the run by achieving a better dominance ranking, or later in the run by being located in a less densely sampled area of the trade-off. Individual replacement is particularly effective for mGAs as it allows a single population to act both as the store of best-to-date genotypes and as the source of parents for further evolution.

While the mGAs use the same replacement strategy as the SOGAs, the value used to determine if a fitness gain has been made is the selection-fitness, since there are multiple evaluation-fitness values. This means that the selection-fitness calculation has to occur after the fitness-evaluation and non-fitness feedback and before individual-replacement. Compare Figure 4.1 showing the selection-fitness as an operation carried out at the end

of the evolution loop with Figure 5.2 where selection-fitness is calculated prior to the replacement of the least-fit genotype.

With the SOGAs, it was possible to determine whether a fitness gain had been made without inserting the genotype into the population simply by comparing the fitness of the newly created genotype to that of the least-fit genotype in the population. For the mGAs, genotype selection-fitness is a product both of the dominance-rank achieved and the degree of fitness sharing experienced, with fitness sharing being determined by the number and proximity of genotypes within the niche. Both can be determined only by including the new genotype within the population. All non-duplicate genotypes are therefore inserted into the population and selection-fitness determined. Only then is it possible to check if a fitness gain has been made. Genotypes cannot remain in the population without making a positive contribution to the population as a whole since their selection-fitness is determined relative to the current population.

The Land Block and P&P mGAs use the same operator sets as their single-objective equivalents, with the GA operators again applied individually.

5.8 Initial mGA Evaluation

The goal in the initial evaluations was to determine the relative performance of the two mGA representations for a two-objective application. The mGA performance was tested using an extension of the Hartwood application used for the SOGA testing. The number of land blocks was increased to 95, by adding more fields for which data had become available, while the land uses available remained at 10. In addition to the financial returns objective (NPV over 60 years), the Shannon-Weiner (SW) index was added with the aim of maximising the diversity and evenness of the pattern of land use (Forman & Godron 1986). SW is calculated as follows:

$$SW = - \sum_l P_l * \log_e P_l, \quad (5.3)$$

where P_l is the proportion of the management unit area devoted to land use l .

SW is maximised when the largest number of available land uses are present in equally sized areas. This second objective was chosen as it has an optimum value that could

be predicted *a priori* and because it was known to conflict with the financially optimum land-use pattern of a near-mono-culture of cattle.

5.8.1 mGA evaluation metrics

The mGA evaluation employs many of the same metrics used for the SOGA testing: the learning efficiency measures that count all, gain, no gain and duplicating events and the CPU time. To investigate the effectiveness of the mGAs in finding the single-objective optima at the ends of the trade-off, the maximum fitness value for *each* objective was tracked, *MaxNPV* for the financial objective and *MaxSW* for diversity. The population average values for each objective, used for the SOGA evaluation, were not employed because the mean value, taken over a population evolving to form a trade-off is not meaningful.

Comparison of the mGAs' performance is more complex than for SOGAs with several factors influencing the quality of solutions (Zitzler et al. 1999, Srinivas & Deb 1995).

- The members of the terminal population should include only non-dominated solutions, unless the population size is larger than the number of non-dominated solutions. The first metric used is the count of non-dominated solutions (*NonDom*) as any dominated solution cannot be Pareto-optimal.
- The mGA population should be spread evenly across the trade-off front. Evenness (*Even*) is measured as the absolute difference between the actual and expected number of genotypes per niche.
- The mGA should find solutions across as much of the trade-off front as possible. The coverage (*Cover*) criterion is measured for the two-objective test applications by summing the areas of the rectangles defined by the origin and coordinates defined by the individual objectives.

The values of these metrics are all recorded during the course of the mGA run.

The final metric used in the evaluation is the extent to which the terminal populations with one representation dominate those of the other. This provides a means of determining if one representation is consistently better at finding solutions over either part or all of the trade-off. Individual dominance (*IDom*) for the Land-Block mGA is calculated for each Land-Block mGA population as the mean proportion of all P&P populations

dominated. IDOM for the P&P mGA measures the equivalent dominance over the Land-Block populations. The mean of the IDOM values for each representation is then calculated and used as the dominance criterion (*DOM*).

Given the uncertainty about the impact on performance of mating restrictions, it was decided to first test the performance of the two mGAs with and without mating restrictions imposed. This would ensure that a fair comparison between the two representations was being made, since it was possible that mating restrictions were effective for only one of the mGAs.

5.8.2 Results and discussion of initial testing

Parametrisation used in the mGA testing

Average performance values were derived for each mGA from 25 runs with and without mating restrictions. The mGAs were parameterised as follows:

- the populations size is 21, since a niche-size of 0.1 is used in a two-objective problem, see Table 5.5,
- the mating-distance, when used, was set to 0.25,
- the operators retained the parameterisations used for the SOGA application,

The runs were terminated after 200 gaining events or 20 consecutive events without making a gain (no gain or duplicate events). The results of these runs are tabulated in Table 5.6, with the difference (*Diff*) and significance (*Sig*) measures as used for the SOGA analysis.

Effectiveness of mating restrictions

Table 5.6 shows the effect of mating restrictions on the Land-Block and P&P mGAs.

For the Land-Block mGA there is no significant difference in performance with or without mating restrictions in place. For the P&P mGA the significant difference in performance is the greater efficiency of the P&P mGA with mating restrictions imposed. This was an anticipated effect of mating restrictions but is not explained by differences in the numbers of gain, no gain and duplicate reproductive events since their proportions are

Table 5.6: The effect of mating restrictions on mGA performance

Metric	Land-Block-Restricted	Land-Block	Diff	Sig
MaxNPV (£M)	6.123	6.123	-	-
MaxDIV	2.111	2.095	0.016	0.009
NonDOM	21	21	-	-
Cover (1e8)	1.172	1.180	-0.008	0.742
Even	15.6	15.7	-0.1	0.929
Dominance	0.445	0.429	0.016	0.977
All	400	430	-30	0.028
Gain	200 (50%)	200 (46%)	-	-
NoGain	193 (48%)	226 (53%)	33	0.018
Duplicates	7 (2%)	4 (1%)	3	0.017
CPU (sec)	1037	1126	-89	0.019
Metric	P&P-Restricted	P&P	Diff	Sig
MaxNPV (£M)	6.123	6.123	-	-
MaxDIV	2.166	2.167	-0.01	0.488
NonDOM	16.4	16.3	0.1	0.906
Cover (1e8)	0.936	0.970	0.034	0.150
Even	35.5	37.3	-1.7	0.344
Dominance	0.415	0.395	0.020	0.982
All	422	472	-50	0.000
Gain	103 (24%)	107 (23%)	-4	0.000
NoGain	146 (35%)	168 (35%)	-22	0.000
Duplicates	173 (41%)	197 (42%)	-24	0.000
CPU (sec)	1553	1698	-145	0.000

all but identical. The P&P mGA with mating restrictions simply required less reproductive events to achieve comparable performance.

If further analysis were to be conducted on the use of mating restrictions it would be useful to examine a range of parameter values for mating distance and the effect of adapting mating distance over the course of the mGA run.

For the evaluation of the relative performance of the Land-Block and P&P mGAs the mGAs with mating restrictions were used.

Land-Block and P&P mGAs' performance for the individual objectives

Table 5.7 presents the comparison of the performance for the best performing Land-Block and P&P mGAs.

Table 5.7: Land-Block and P&P mGA performance

Metric	Land-Block-Restricted	P&P-Restricted	Diff	Sig
MaxNPV (£M)	6.123	6.123	-	-
MaxDIV	2.111	2.166	-0.005	0.577
NonDOM	21	16.4	4.6	0.000
Cover (1e8)	1.172	0.936	0.236	0.000
Even	15.6	35.5	-19.9	0.000
Dominance	0.818	0.155	0.663	0.000
All	400	422	-22	0.089
Gain	200 (50%)	103 (24%)	97	0.000
NoGain	193 (48%)	146 (35%)	47	0.000
Duplicates	7 (2%)	173 (41%)	-166	0.000
CPU (sec)	1037	1553	-516	0.000

For the single objective optima, MaxNPV and MaxSW, both mGAs performed well, with each achieving 99% of the known optimum values, pointing to the utility of the doping process. The P&P representation achieves better results for MaxSW. This is statistically significant, but of marginal practical significance because it is smaller than the error with which the DSS can make real world-predictions. Further examination of these results does, however, point to differences between the operation on the two mGAs. Figure 5.7 shows the counts of genotypes within a matrix 100x100 defined across the terminal range of phenotype values, presented as a three dimensional surface. The P&P mGA is more consistent in finding the single objective optima, seen as the larger peaks at the ends of the trade-off in Figure 5.7.

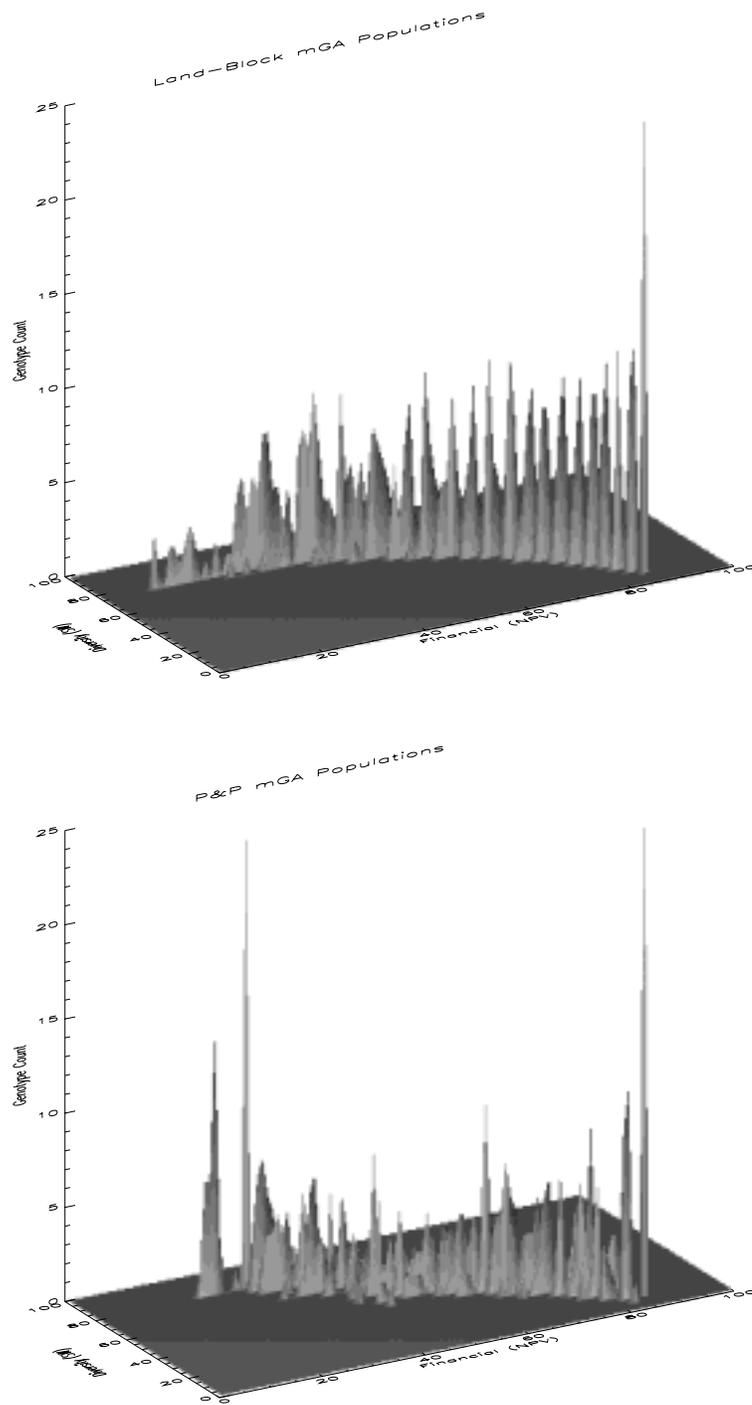


Figure 5.7: Counts of genotypes for Land Block and P&P mGAs

The greedy algorithm assists the P&P mGA in finding the financial optimum by allocating the most productive land uses first. The greedy algorithm is, however, also of utility for the diversity objective. Whatever the land-use priority or target-percentage is, the allocation made by the P&P mGA will have the best possible financial returns since the most productive land is allocated first. The P&P mGA thus has only to find the order and magnitude of the target-percentages whereas the Land-Block mGA must find the the best locations as well.

Multi-objective performance

For the non-dominance criterion, the Land-Block mGA performs well with no dominated individuals in the population. This reflects the success of the individual replacement strategy in ensuring that dominated solutions are eliminated from the population, allowing the recombination of non-dominated solutions to achieve coverage of the trade-off. For the P&P mGA the average count of non-dominated solutions is reduced by occasional runs where significant numbers of dominated solutions are present.

For evenness and coverage, the Land-Block mGA performs significantly better than the P&P mGA. The Land-Block mGA populations on average also dominate 80% of the P&P mGA populations. The cause of this superior performance was the Land-Block mGAs' more consistent ability to form high-fitness intermediate solutions. Figure 5.8 illustrates this by plotting, in phenotype space, the locations of the genotypes for the 25 terminal populations of each mGA. The Land-Block populations can be seen to be more tightly concentrated at the edge of the feasible space. The cause of this poorer P&P performance in finding intermediate solutions may be the use of the 5% granularity in defining the target land-use percentages. While this had been agreed as an acceptable level of accuracy for presenting results to land managers it perhaps compromised the mGA's ability to recombine genotypes to form intermediate solutions.

Efficiency of mGA learning

The CPU used, a metric indicating the overall efficiency of the mGA learning, also points to the superior performance of the Land-Block mGA. The problem for the P&P mGA can be seen in the breakdown of the events. While the difference in the total number of events is not significant the Land-Block mGA is consistent in achieving 200 gaining events

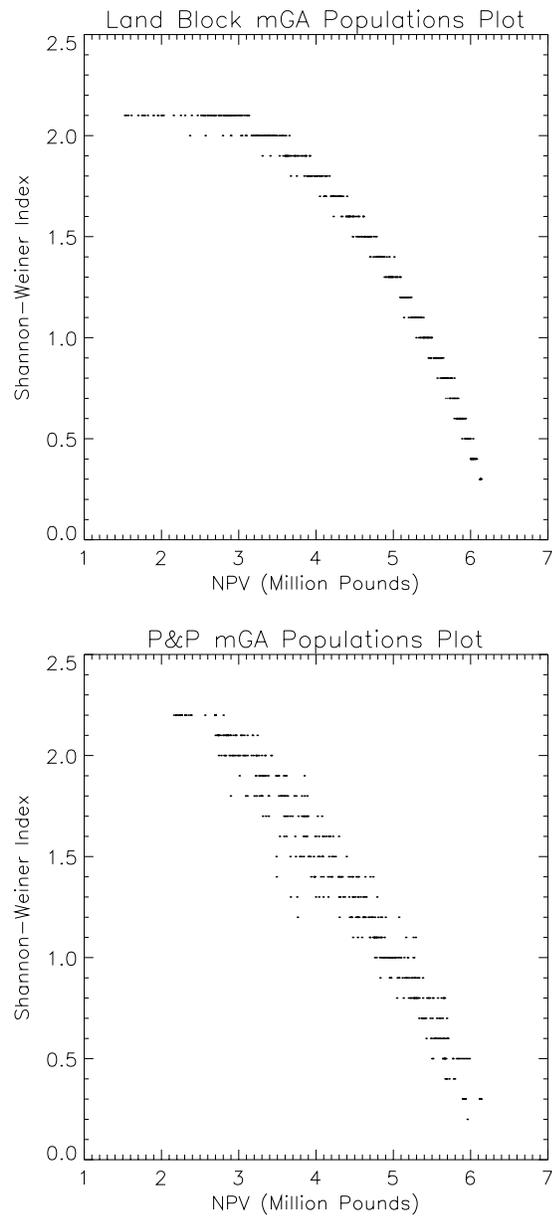


Figure 5.8: Population plots for Land Block and P&P mGAs

while the P&P mGA achieves only 103. This means that in most cases the P&P mGA is terminating due to its failure to make gaining events. It would be possible to increase the threshold for consecutive non-gaining events but this would only decrease further the efficiency of the P&P mGA. The problem for the P&P mGA is the large number of duplicate genotypes being created. While duplicates will tend to result from the use of the non-fitness feedback mechanism, the use of the 5% granularity and the relatively

small population may be aggravating the problem. The population size of 21 provides an adequate visualisation for the trade-off front, but may provide insufficient genetic diversity for the P&P representation.

A further parameterisation issue that may have affected the P&P mGA was the parameterisation of the operator probabilities. While the operator parameterisation may be suitable for the Land-Block representation it may be necessary to adjust the parameterisation of the P&P operators.

5.9 Summary and Conclusions

This chapter has presented the implementation and initial testing of two mGAs based on the Land-Block and P&P SOGAs. The mGAs required the following specialisations. For setting the selection-fitness values, the mGAs employ dominance-ranking, rank-based linear normalisation and niche-based fitness sharing. This allows the mGA to search for a population of non-dominated solutions defining a trade-off between objectives. For binary reproduction, a bias in selection of the second parent towards genotypes within the neighbourhood of the first parent was added. This sought to encourage the creation of intermediate genotypes required to define the trade-off. A further phase of population initialisation, doping the population with genotypes known to have high single-objective fitness, was added to assist the mGAs in finding the single-objective optima.

From the evaluation undertaken it was possible to conclude the following.

- The mGAs provide a useful means of quickly establishing the structure of the trade-off between non-commensurable objectives.
- Doping assisted the mGAs to find the individual optima for each objective and thus locate the limits of the trade-off front.
- Mating restrictions increased the efficiency of the P&P but not the Land-Block mGA. In neither case were mating restrictions seen to have a significant effect on the quality of the solutions found.
- The superior performance of the Land-Block mGA pointed to the need for care when parameterising the mGAs. For the P&P mGA the niche size must be chosen such

that it results in a population large enough to maintain sufficient genetic diversity. The P&P mGA may also be sensitive to the granularity used in specifying the target land-use percentages, if these are set too high, the ability of the mGA to find genotypes evenly spread across the trade-off front may be compromised

It was decided that based on these conclusions it would be profitable to compare the results achieved by the mGAs with those proposed by expert land managers.

Chapter 6

Parameterisation

6.1 Introduction

The parameterisation used in the initial evaluations of the Land-Block and P&P SOGAs was defined either by GA-design decisions or used values from the GA literature (Davis 1991). Given the novel representation, operators and feedback-mechanisms employed in the land-use planning GAs, it was necessary to investigate appropriate, application-specific, parameterisations. Of particular importance to GA performance is the parameterisation of the probability with which operators are applied. Adaptation of these probabilities over the course of the run is necessary as the optimum operator probabilities follow nonlinear trajectories over the course of a GA run. Two approaches to adaptive parameterisation were tested, *evolutionary operation* (EVOP) and *online-parameterisation* (OP).

6.2 Evolutionary Operation

6.2.1 Conventional Evolutionary Operation

EVOP is a well-established method for incrementally improving the performance of manufacturing processes and maintaining near-optimal performance in the face of possible changes in conditions (Hunter & Kittrell 1966, Lochnar & Matar 1990).

In each time-increment EVOP runs a number of structured experiments, the complete set being referred to as an *EVOP cycle*. Examples of EVOP cycles for one and two pa-

parameters are shown in Figure 6.1. The filled circles are the *base-scenarios*, representing the current parameter settings. The open circles represent the *variation-scenarios*. These are the experiments used to test the direction in which the parameter(s) should be modified to improve performance. For the single-parameter EVOP there is one base- and two variation-scenarios and for the two-parameter EVOP one base- and four variation-scenarios.

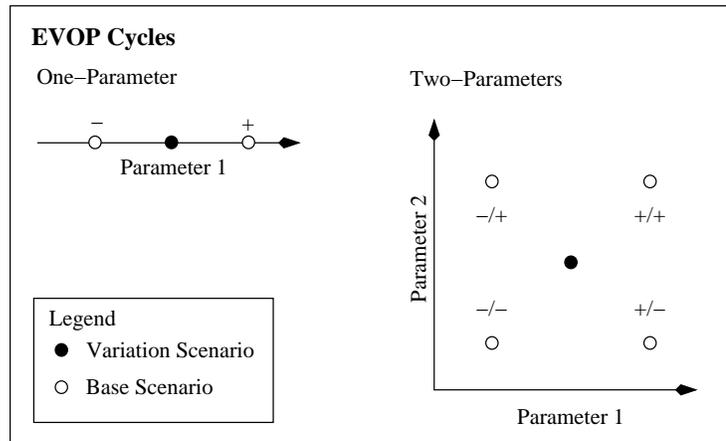


Figure 6.1: EVOP cycles for one and two parameters

In a process-control setting these experiments are run online and sequentially. For the variation-scenarios, the deviations above or below the base value are kept relatively small, to ensure that the product remains within specification. The deviations must, however, be large enough that, despite noise in the system, they provide information on the parameterisation that will provide the best improvement in quality. To reduce the possible effect of noise, the EVOP cycle is run several times until the uncertainty is less than the effect of the variation-scenarios. The size of deviation may also be adapted over the course of a series of EVOP cycles. Once the experiments have been conducted they are analysed and the new values are set for the parameters and/or variation-scenarios. The *EVOP-GA* parameterisation makes use of an EVOP-like strategy but with significant modifications that are detailed in the following section.

6.2.2 EVOP-GA

The aim of the EVOP-GA is to determine the value of parameters during the course of the GA run. The integration of the EVOP strategy with the GA is shown in Figure 6.2.

Following initialisation, the EVOP-GA runs and analyses EVOP cycles until stopped by the convergence criteria.

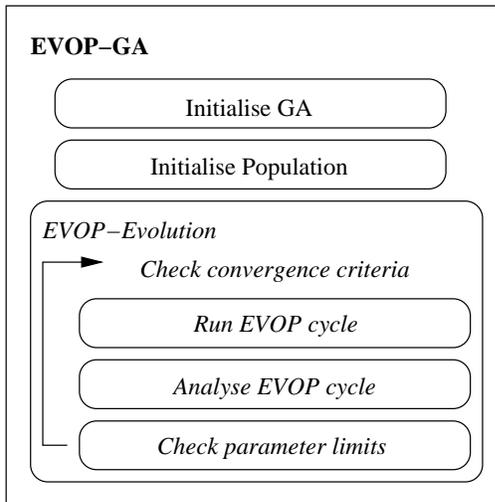


Figure 6.2: EVOP-GA algorithm

Running the EVOP cycle

The main difference between EVOP-GA and classical EVOP can be seen in the way that the EVOP cycle is run. In classical EVOP, testing is conducted as part of a continuous process, with individual variation-scenarios tested sequentially. For the EVOP-GA the variation-scenarios are run using copies of the same starting population. The EVOP-GA thus tests the scenarios in parallel.

The details of the *Run EVOP Cycle* method are illustrated in Figure 6.3. In classical EVOP, the cycles are run repeatedly to distinguish the signal of improved performance from the stochastic noise associated with the manufacturing process. For the EVOP-GA each scenario is run for a number of GA evolution cycles (reproduction, replacement and selection-fitness). The number of reproductive events that EVOP is run for is termed here the EVOP *epoch*. If the threshold for consecutive non-gaining events is broken before the EVOP epoch is complete the scenario run will also terminate, guarding against parameterisations that make very infrequent gains. The size of the EVOP epoch is user-defined and is adjusted to ensure the number of GA iterations is large enough to distinguish between the variation-scenarios. The value used to date is 50 events. The metric used to assess the scenarios performance can be chosen from the metrics available to the GA. In the example

below the average population fitness was used as maximum population fitness changes too infrequently to guide the adaptation.

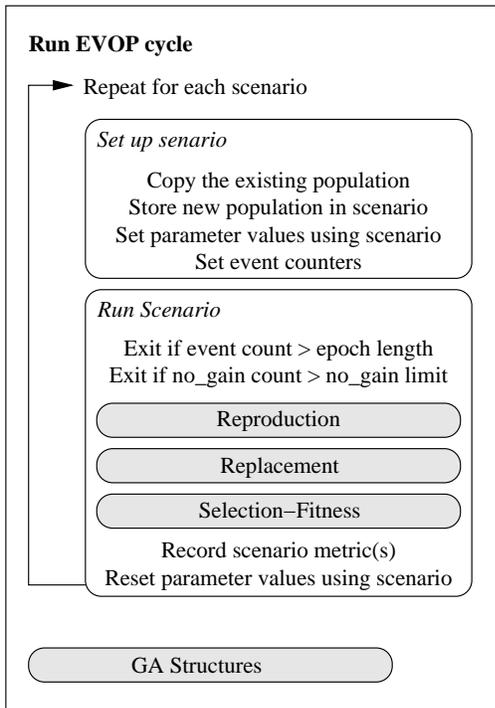


Figure 6.3: Running the EVOP cycle

Analysing the EVOP cycle

EVOP analysis determines the parameter value(s) that maximise the fitness metric. The form of this analysis depends on the number and relationship between the parameters being manipulated. The EVOP-GA was initially applied to the parameterisation of the operator probabilities for the Land-Block operators (EVOP-LB).

Since the Land-Block GA has only two operators whose probabilities sum to one, it is possible to estimate the parameter values that maximise the fitness metric by fitting a quadratic equation to the base and two variation scenarios as follows:

$$Y = a_1 Q_1 + a_2 Q_2 + a_{12} Q_1 Q_2, \quad (6.1)$$

where Y is the fitness metric to be maximised, Q_1 is the probability of crossover and Q_2 is the probability of mutation with the operator probabilities constrained so that $Q_1 + Q_2 = 1$.

The three scenarios are:

Scenario	Fitness Metric	Crossover, Mutation
Base Scenario	y_0	q_1, q_2
Variation Scenario 1	y_1	$q_1 + h, q_2 - h$
Variation Scenario 2	y_2	$q_1 - h, q_2 + h$

where h is the *variation-size*, the magnitude of deviation from the base scenario parameter values.

By substituting the values from each scenario into Equation 6.1 we can calculate the a_{12} coefficient of the quadratic as

$$a_{12} = \frac{2y_0 - y_1 - y_2}{2h^2}. \quad (6.2)$$

The sign of the a_{12} coefficient determines whether the quadratic curve is concave or convex.

If $a_{12} \leq 0$ then it is concave and thus has its maximum at one of the end points, see Figure 6.4(a)-(b). In this case a simplifying assumption is made that the base scenario for the next EVOP epoch should not have parameter values beyond the range of those tested in the current epoch. Consequently the variation-scenario with the better performance becomes the base-scenario for the next epoch with the pre-epoch population replaced by that stored in the variation-scenario, see Figure 6.5. EVOP-GA then proceeds with the next cycle.

The quadratic is convex if $a_{12} > 0$, see Figure 6.4(c)-(e). In these cases the crossover probability q_1^o where Y is maximised can be calculated as:

$$q_1^o = 0.5 + \frac{q_1 - q_2}{2} + \frac{y_1 - y_2}{4ha_{12}} \quad (6.3)$$

The mutation probability for this location is simply $1 - q_1^o$.

In using Equation 6.3 three cases must be handled as illustrated in Figure 6.4(c)-(e).

- For case (c) the modelled optimum lies within the range of parameter values tested and is applied directly as the base scenario of the next EVOP epoch.
- For case(d) the modelled optimum lies outwith the range of parameter values tested but within the allowed parameter range from 0.1 to 0.9. To be consistent with the concave cases, the values of the next epoch's base scenario could have been restricted

to the range tested, and hence the larger variation scenario would have been chosen. It was decided, however, to permit the parameter values of the new base scenario to be set outwith the tested range. This decision was based on the observation, during development of the EVOP algorithm, that the modelled optimum values rarely lay far outwith the range of parameter values tested.

- For case (e) the modelled optimum lies outside the permitted range and so the base scenario parameter is set to the closest permitted value; i.e. 0.9 in the example shown.

In all these cases, as there exists no actual population created with the modelled parameterisation, the next EVOP cycle uses the population with the highest-fitness, Figure 6.5. This makes the assumption that the highest-fitness population is genetically most similar to a population which would be created by running the GA using the modelled parameterisations.

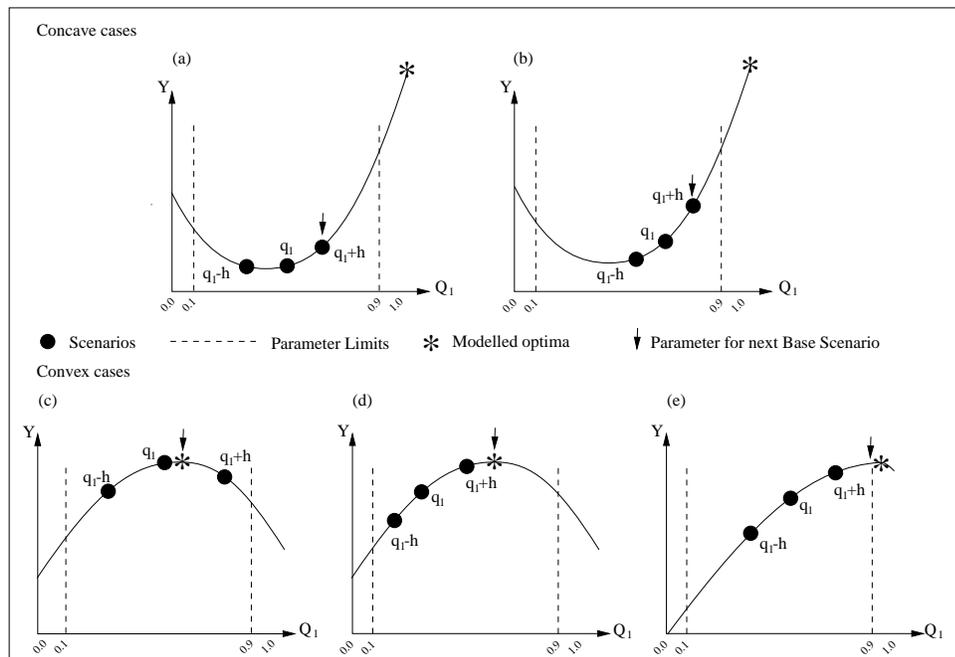


Figure 6.4: EVOP cases

EVOP-LB results

Figure 6.6 shows the average trajectories for the crossover and mutation probabilities. These show the expected decline in the probability of crossover over the course of the GA run. The rate of adaptation for the parameters is, however, less than expected. This reflects the inconsistency in the pattern of adaptation for the individual EVOP-LB runs.

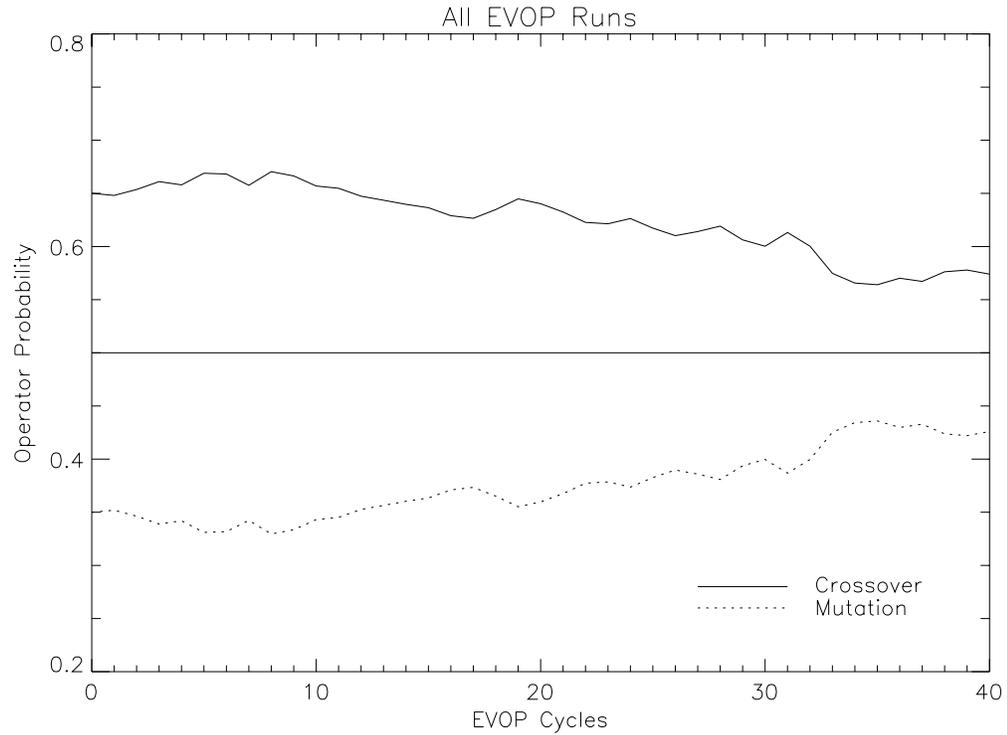


Figure 6.6: Averaged EVOP Operator Probabilities

Table 6.2 compares the performance for the evaluation metrics of the EVOP-LB and fixed parameterisation LB-Best-Fixed. The difference and significance values are calculated as previously.

Table 6.2: Relative performance EVOP-LB vs. LB-Best-Fixed

Metric	EVOP-LB	LB-Best-Fixed	Diff	Sig
AvgFit (£M)	5.29	5.34	-0.05	0.434
MaxFit (£M)	5.32	5.38	-0.06	0.354
All	2000	2000	-	-
Gain	1151 (57%)	1416 (71%)	-265	0.000
No Gain	216 (11%)	427 (21%)	-211	0.000
Duplicates	633 (32%)	157 (8%)	476	0.000

From Table 6.2 it can be seen that EVOP-LB is more efficient in finding the same quality of solution. For both AvgFit and MaxFit there is no significant performance difference fitness. While both GAs are terminating on reaching 2000 events, all the differences in the breakdowns of reproductive event types are significant. The EVOP-LB has fewer no-gain events but creates more duplicates. The EVOP-LB thus requires fewer gaining events than the LB-best-fixed to achieve the same performance. The EVOP-LB also requires less fitness-function evaluations, the sum of the gain and no-gain event counts, again improving efficiency.

6.2.3 EVOP summary and conclusions

The EVOP-GA strategy is applicable to any GA parameter that is adapted over the course of the GA run. The parameter adaptation is based on structured experiments that systematically investigate the effect of parameterisation. The method as tested, however, was inconsistent in its pattern of adaptation and consequently produced an inadequate improvement in performance relative to the fixed parameterisation GA. Further development of the approach, possibly using an adaptive setting of the size of the variation-scenarios, may improve performance. One inherent limitation in the method, however, is the rapid growth in the number of variation-scenarios required as the number of parameters being optimised increases (2^n , where n is the number of parameters). This limitation is of particular importance for the parameterisation of the P&P operator probabilities. It was thus necessary to consider an alternative approach for the parameterisation of the P&P SOGA.

6.3 Online-Parameterisation GA

Given the limitations of the EVOP-GA it was necessary to implement an alternative online-parameterisation strategy that would more effectively optimise the operator probabilities for the Land-Block GA and allow the investigation of P&P GAs parameterisations. The online-parameterisation GA (*OP-GA*) is based on that proposed by Davis (1989) and is particularly suited to analysing the interactions between multiple GA operators. For the OP-GA, an operator's probability of application adapts based on the ratio of the average fitness gained by the operator to the average fitness gained by all operators. If the operator

is making greater gains than the average for all operators then its probability is increased otherwise it is reduced.

The OP-GA uses independent operator application. This permits the tracking of the fitness gained for each application of an operator. The OP-GA also recognises that it is possible for operators to act in a facilitating role. For example, a mutation operator may introduce a new feature to a sub-optimal genotype that, when later crossed, results in a further fitness gain. A credit pass-back method is therefore employed to ensure that facilitating operators receive adequate credit. The implementation of the OP-GA algorithm is set out in Figure 6.7.

6.3.1 The OP-GA algorithm

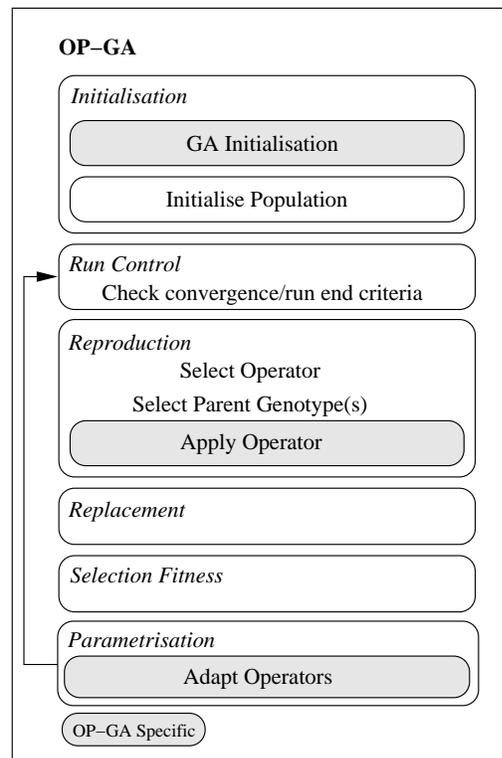


Figure 6.7: OP-GA algorithm

The OP-GA is fully integrated within the operation of the GA, contrasting with the EVOP-GA approach of running the GA inside an EVOP framework. The OP-GA specialises three parts of the GA: GA initialisation; operator application and operator pa-

parameterisation, in this case operator probabilities. These specialisations are discussed below.

Operator application

After GA initialisation, dealt with separately below, the OP-GA runs the normal GA evolutionary cycle but, during *Apply-Operator* in addition to reproduction, the method records the ancestry of the offspring genotypes and the operator that created them, Figure 6.8. Ancestry is recorded by inserting a reference to the parent genotype(s) into a list within the genotype. Over the course of the GA run, this builds a family tree with genotypes produced later in the GA run potentially having several levels of ancestors. The deletion of genotypes from the population automatically prunes the family-trees by eliminating genotypes from the lists of ancestors. The family-trees are used to control the process of passing back credit to operators that created ancestor genotypes.

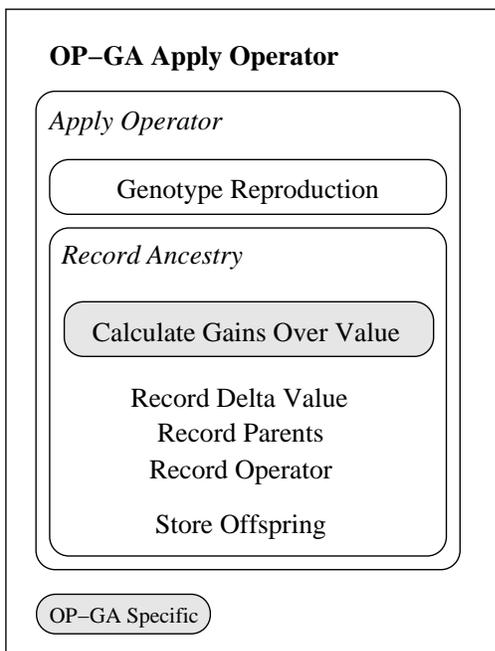


Figure 6.8: OP-GA operator application

The value of the *gains-over* criterion, used later by the *adapt-operators* method to calculate the fitness gain made by the offspring, is also recorded in the genotype. Gains over minimum-fitness of the population is the metric used in the OP-GA. This contrasts with Davis' use of gains over maximum-fitness. The choice of this alternative metric was

based on the extra information it provides. During the course of a typical run, the value for the maximum-fitness of the population changes very infrequently compared with that of the minimum-fitness. Gains over minimum-fitness provides a richer source of information for OP-GA on the performance of the GA operators, and places less reliance on the passback mechanism to ensure that operators receive rewards for facilitating actions. Since the GAs employ individual replacement, the OP-GA uses information from all genotypes inserted into the population.

Operator adaptation

The *adapt-operators* method is called at each iteration of the GA evolution and performs two actions, see Figure 6.9. The first is the calculation of the fitness gained, termed by Davis the *delta* value. This is done for each of the viable offspring from the current cycle; those inserted into the population. The second action is to update the operator probabilities. This action is performed after a fixed number of delta value calculations. The number of calculations, a parameter of the OP-GA, is termed the *adaptation window*.

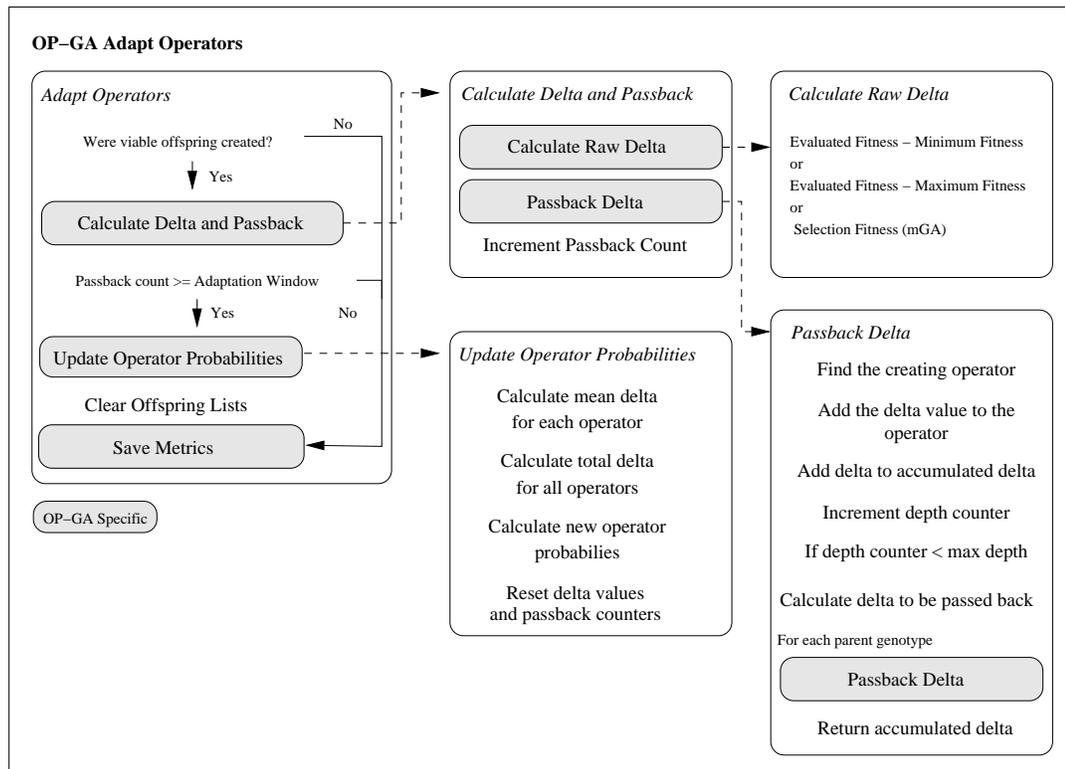


Figure 6.9: OP-GA operator adaptation

Delta value calculations. The delta value calculation is a two step process, with the *raw-delta* for each offspring genotype calculated first, and then a proportion of the raw-delta passed-back to the genotype's ancestor(s). The raw-delta calculation simply takes the difference between the evaluation-fitness and the gains-over value of the genotype. The subsequent *pass-back-delta* method takes as its input the raw-delta value and credits this to the creating operator of the genotype. Pass-back-delta is then called recursively for each parent genotype, up to a recursion depth specified by the OP-GA. A recursion depth of 5 was adopted. This is smaller than that used by Davis, but given the size and turnover in the population, it is unusual for a genotype to possess such a deep family tree. The *passed-back-delta* value used at each successive recursion level is reduced by a fixed proportion (0.9). For genotypes created by binary operators, the passed-back-delta is equally divided between the ancestors.

Update operator probabilities. The operator probabilities are updated according to how well each operator has performed during the course of the adaptation-window. This is accomplished in two stages.

1. Each operators current probability of application is first reduced by multiplying it by the *adaptation proportion*, (A). The value of A , a parameter of the method, determines the change in operator probabilities permitted in a single adaptation window. A value of 0.15 was proposed by Davis (1989) and is used here.

The operator application probabilities sum to one since independent operator application is being used. Reducing each operator's application probability by A , leaves a share of operator application probability, A . A is reallocated between the operators in the second stage.

2. The reallocation of A is determined by the relative performance of the operators. The metric used to compare performance is the average delta per application of the operator, Δ . The average delta includes that credited to the operator by the passback mechanism. Each operator receives a share of A equal to the ratio of the operator's Δ to the sum of Δ for all operators.

These two stages can be formalised as shown in Equation 6.4, below. The new operator probability P_n for the i^{th} operator O_i is

$$P_n(O_i) = [P_c(O_i) * (1 - A)] + \left[\frac{\Delta_{O_i}}{\sum_{j=1}^N \Delta_{O_j}} * A \right] \quad (6.4)$$

where $P_c(O_i)$ is the current operator probability, Δ_{O_i} is the average delta per operation for the i^{th} operator and $\sum_{j=1}^N \Delta_{O_j}$ is sum of Δ values for all operators. The update method is illustrated for a two operator case in Figure 6.10.

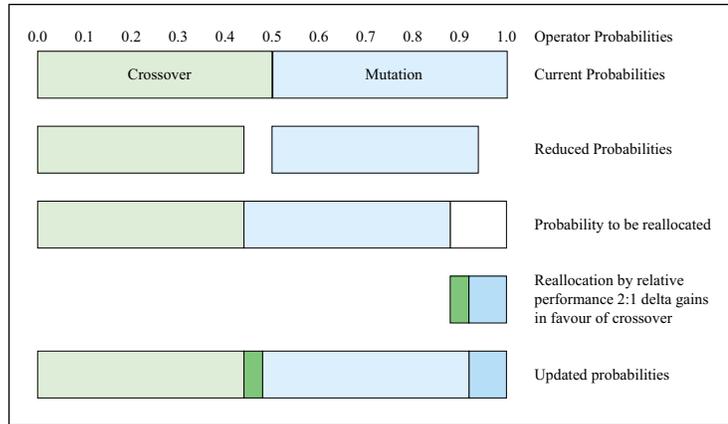


Figure 6.10: OP-GA example

After the update operator method has completed, the evaluation metrics are saved. The metrics employed are the average and maximum fitness of the population and the event counts.

Operator Initialisation

To complete the description of the OP-GA it is now possible to return to the initialisation of OP-GA, and in particular the setting of the initial operator probabilities. The approach adopted is, shown in Figure 6.11. The method iteratively calculates average values for the operator probabilities after a single adaptation window. The average calculation uses the same population as the basis for each of the runs and all the operators start with equal probability. The average operator probabilities thus found are fed back as the starting operator probabilities in the next iteration. This repeated calculation of the averages continues until there is only a small difference between the new average values and the current operator probabilities ($< 5\%$ for all operators). All operators have a

minimum probability to ensure that they will be tested during the course of the OP-GA run. The final operator probabilities are therefore adjusted so that the minimum operator probability is 0.05.

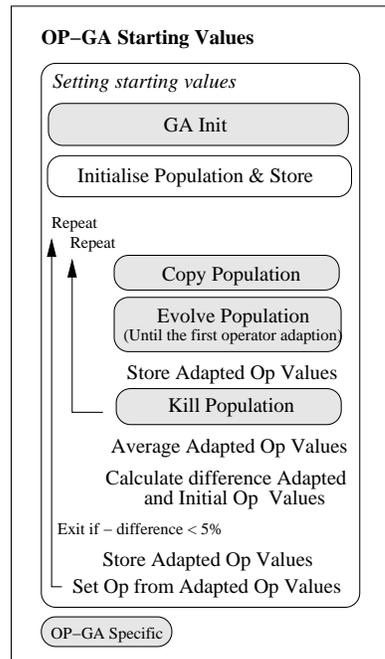


Figure 6.11: OP-GA operator initialisation

6.3.2 Land-Block OP-GA

The first application of the OP-GA was to the Land-Block SOGA (LB OP-GA). This was done to investigate whether it was possible for OP-GA to improve on the results found by the EVOP-GA.

Land-Block OP-GA testing

The same problem was used as for the EVOP-LB testing, with the performance of the LB OP-GA compared to that of the best fixed parameterisation SOGA. The LB OP-GA was parameterised as follows:

- the delta values metric was gains over minimum population fitness;
- the adaptation window was 50 gaining events;
- the adaptation proportion was 0.15;

- the pass-back recursion depth was 5, with a pass-back proportion of 0.9.

The initial operator probabilities are shown in Table 6.3.

Table 6.3: Land-Block initial operator probabilities

Operator	Probability
Uniform Crossover	0.626
Mutation	0.374

Land-Block OP-GA results

The performance of the LB OP-GA is presented in Table 6.4 and compared with the best performing fixed parameterisation Land-Block GA (LB-Best-Fixed). For AvgFit and MaxFit the LB OP-GA performs significantly better than LB-best-fixed. The efficiency of the two GA's search is not significantly different.

Table 6.4: Relative performance LB OP-GA to LB-Best-Fixed

Metric	LB OP-GA	LB-Best-Fixed	Diff	Sig
AvgFit	5.52	5.34	0.18	0.000
MaxFit	5.54	5.38	0.16	0.000
All	2000	2000	-	-
Gain	1432 (72%)	1416 (71%)	16	0.301
No Gain	442 (22%)	427 (21%)	15	0.136
Duplicates	126 (6%)	157 (8%)	-31	0.079

Figure 6.12 plots the average operator trajectories for the two Land-Block operators. There are three phases for the LB OP-GA. Crossover starts with high probability and then declines as the diversity of the population is reduced and mutation plays an increasingly important role. There then follows a period when both operators are roughly in balance. Crossover then begins to make a comeback as the population converges towards the optimum, with crossover operating by passing the best genes on to all of the population.

The operator trajectories for each of the 25 individual runs, shown in Figure 6.13, are consistent with the average trajectories. There are certain runs when the operators diverge from the .5/.5 balance in the second phase with periods of mutation followed by crossover, for example runs 7 and 16. This seems to indicate the ability of the online parameterisation

to increase mutation, potentially diversifying the population, with crossover later able to recombine mutated genotypes with other members of the population to make further gains.

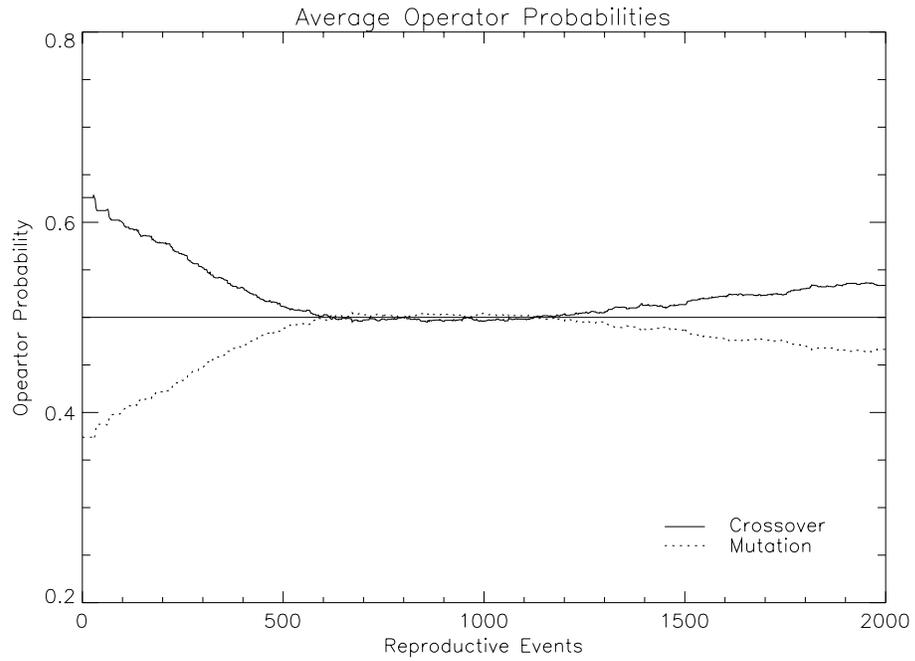


Figure 6.12: Average Land-Block operator probability trajectories

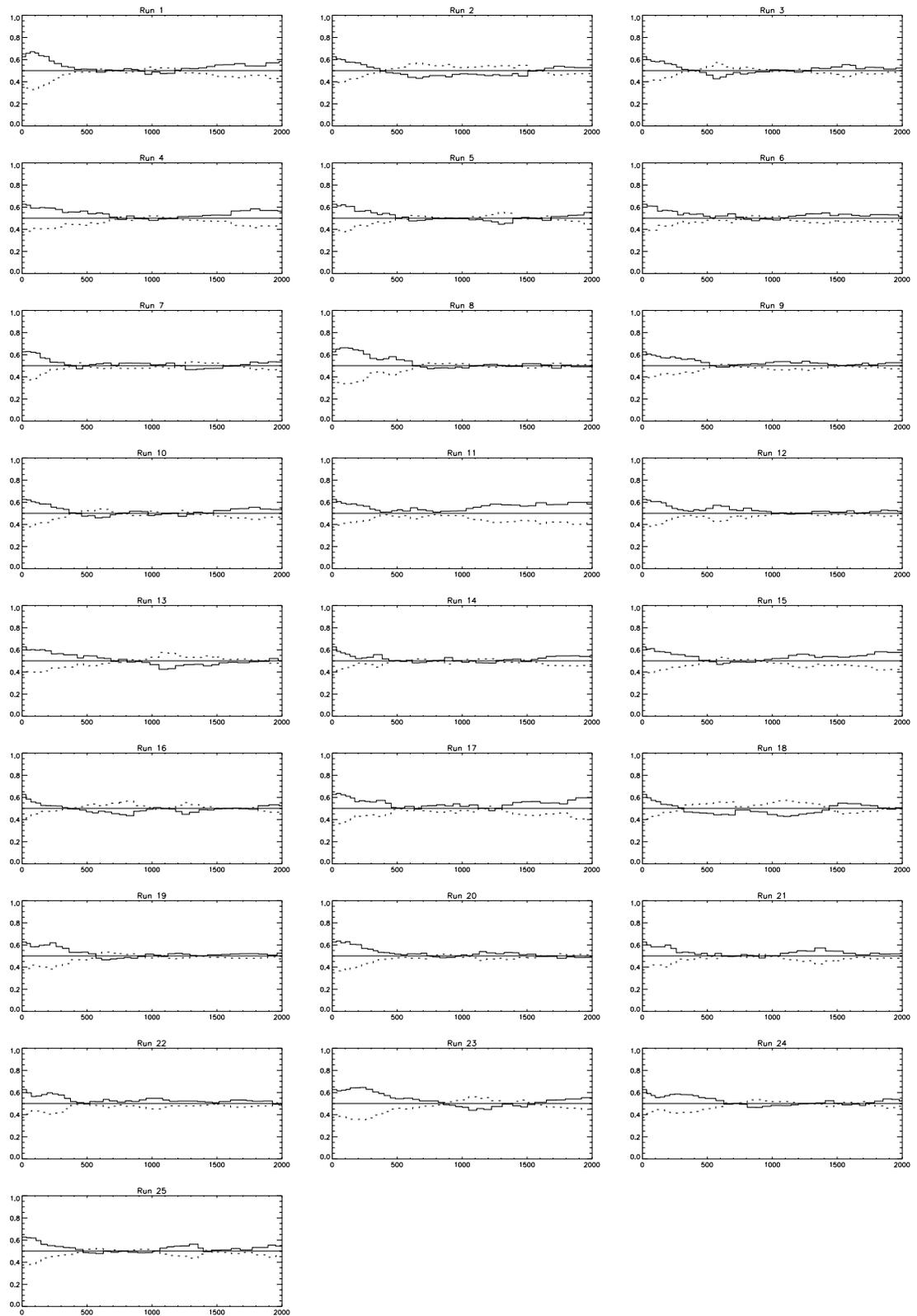


Figure 6.13: Land-Block operator probability trajectories for all runs

6.3.3 P&P OP-GA

Following the testing of the LB OP-GA the method was then applied to parameterising the P&P SOGA (P&P OP-GA). This would allow the effectiveness of the proposed operators to be assessed.

P&P OP-GA initialisation

The initial values for the operator probabilities are shown in Table 6.5. It is very interesting to note that for the P&P representation the balance between binary and unary contrasts with the land-block GA. The binary operators are apparently much less effective for P&P, with uniform crossover only achieving a probability of 0.1 and the other binary operators being assigned the minimum value of 0.05. The binary P&P operators have a combined probability of 0.2 compared with 0.626 for the land-block representation. The reason for this difference may be that since the P&P genotypes are much shorter than those of the land-block representation, crossover style operators act for P&P as a form of mutation. Since there are more forms of mutation possible with the P&P representation the individual probabilities for the operators are smaller than those for the land-block GA. Beyond initialisation values the parameterisation of the OP-GA remained the same as used for the Land-Block OP-GA.

Table 6.5: P&P initial operator probabilities

Operator	Probability
Uniform Crossover	0.10
Partial Relative Reordering	0.05
Splice	0.05
Type Mutation	0.17
Non-uniform Mutation	0.15
Pair-Swap	0.23
Insert-Gene	0.13
Delete-Gene	0.14

P&P OP-GA testing

The P&P OP-GA testing used the same problem as the Land-Block OP-GA. The P&P OP-GA performance was compared to the best performing fixed parameterisation P&P

GA found. A number of options for the balance of operator probability between binary and unary operators were tested, ranging from a binary to unary ratio of 0.6/0.4 to 0.2/0.8. These probability values are, as previously, divided equally between the individual operators. The OP-GA initialisation values were also used as a fixed parameterisation (OP-Init).

Table 6.6 shows that the best performance for quality of solution was achieved with a balance of binary to unary operators of 0.3/0.7. This was significantly better than the next best performing parameterisation 0.4/0.6. The 0.4/0.6 parameterisation was, however, significantly more efficient. This efficiency gain is seen in the reduced number of no gain events. The 0.3/0.7 parameterisation was chosen as the fixed parameterisation GA to be compared with P&P OP-GA based on the superior quality of the solutions found.

Table 6.6: Relative performance of fixed parameterisation P&P GAs

Metric	0.6/0.4	0.5/0.5	0.4/0.6	0.3/0.7	0.2/0.8	OP-Init
AvgFit (£M)	5.32	5.36	5.40	5.43	5.40	5.35
MaxFit (£M)	5.59	5.66	5.67	5.70	5.68	5.60
All	1945	1809	1587	1693	1169	1656
Gain	183	185	184	186	183	191
No Gain	513	526	522	641	446	710
Duplicates	1249	1098	881	866	540	755

As a footnote to the testing of the fixed parameterisation P&P GAs it is interesting that OP-Init performs relatively poorly. This seems to indicate that the the P&P OP-GA initial values while achieving the greatest gains at the start of the run do not remain the best for the course of the entire run. This lends support to the idea that the optimal operator probabilities are dynamic.

P&P OP-GA results

Table 6.7 presents the performance metrics for the P&P OP-GA and PP-Best-Fixed. There is no significant difference between the AvgFit and the MaxFit values achieved. Both GAs achieve very close to the optimum value of £5.72M. The difference between the GAs is in the speed with which they find the optimum. While both the PP-Best-Fixed and the P&P OP-GA are terminating on average before the 2000 event limit, the P&P OP-GA takes one third fewer events to achieve the same performance. The proportion of fitness

gaining events for the P&P OP-GA is very close to that of PP-Best-Fixed. The number of fitness gaining events is however, significantly fewer indicating that the gains per event being made by the P&P OP-GA are larger. It is possible that the OP-GA algorithm is ensuring individual operators are being applied when most appropriate and thus making larger gains. The P&P OP-GA also has significantly fewer no-gain and duplicating events. The effect of the online-parameterisation for the P&P GA is to dramatically increase the efficiency of the search.

Table 6.7: Relative performance OP-PP to PP-Best-Fixed

Metric	P&P OP-GA	PP-Best-Fixed	Diff	Sig
AvgFit (£M)	5.40	5.43	-0.03	0.112
MaxFit (£M)	5.69	5.70	-0.01	0.596
All	994	1693	-699	0.000
Gain	117 (12%)	186 (11%)	-69	0.000
No Gain	403 (40%)	641 (38%)	-238	0.000
Duplicates	474 (48%)	866 (51%)	-392	0.000

The graphs of the P&P OP-GA operator probabilities are presented in Figure 6.14. From the graphs it can be seen that two of the operators are of marginal utility. Both `partial-relative-reordering` and `splice` were initialised to 0.05, the minimum initial probability, and steadily lose probability over the course of the run. It is likely that both of these operators could be eliminated for this particular application. `Uniform-crossover` maintains itself at approximately the 0.1 level throughout the run. The utility of `uniform-crossover` may be being limited by the shortness of the P&P genotype (typically between 2 and 5 genes for the financial optimisation).

Three of the mutation operators increase their probability during the course of the run, `non-uniform mutation`, `type-mutation` and `pair-swap`. For `non-uniform mutation` there seem to be three phases with an initial increase followed by a period of consolidation and then a rapid increase towards the end of the run. This late increase in probability is probably due to the fact that, by late in the run, the correct number, type and order of genes has been found and the allocations are being fine-tuned by changes to the target-percentages. Both `type-mutate` and `pair-swap` will be aided by the DSS greedy-algorithm – with the target-percentages perhaps being modified by the non-fitness feedback.

Finally, it is interesting to note the patterns of the `insert-` and `delete-gene` opera-

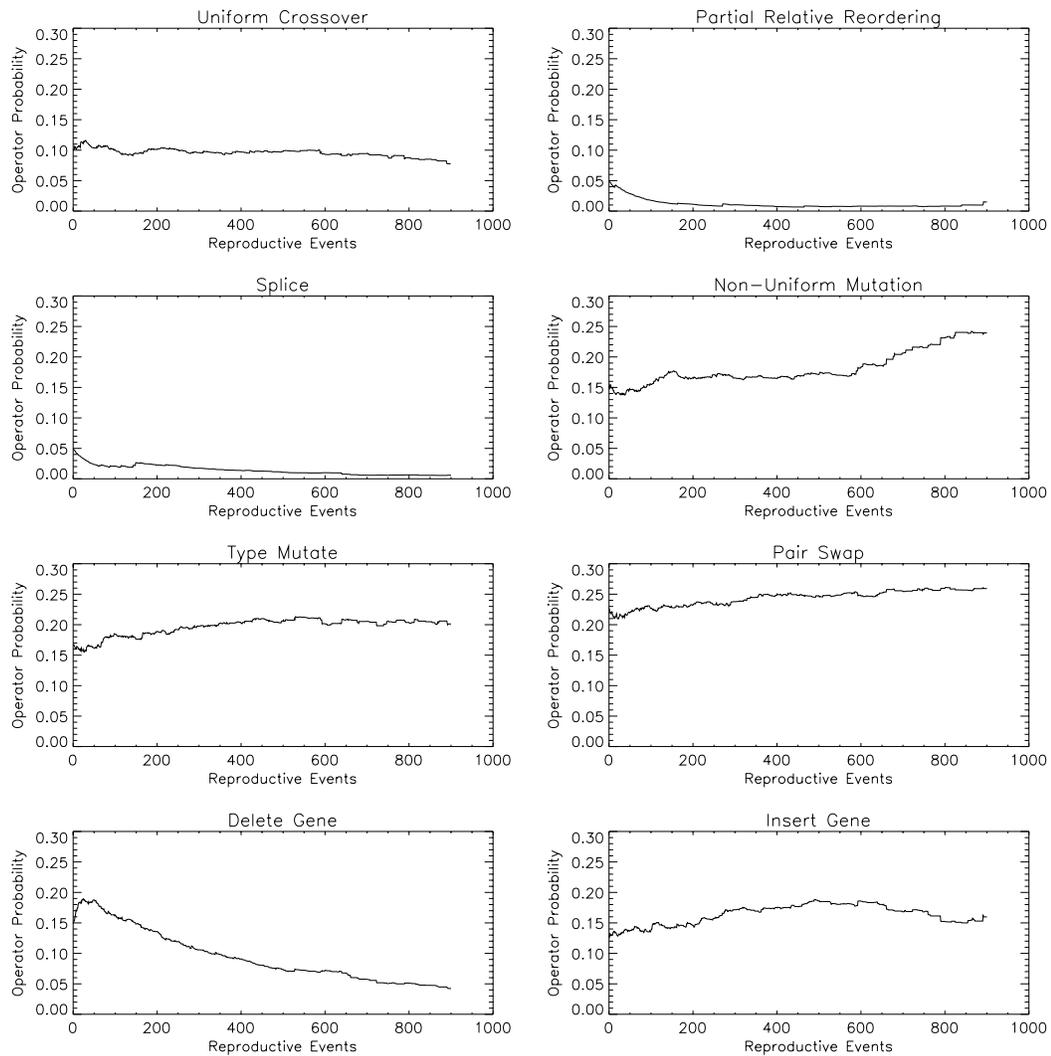


Figure 6.14: Average P&P operator probability trajectories

tors. **Delete-gene** makes gains initially when genotypes are longer, the P&P genotypes being initialised at length five. There are thus significant numbers of genes that may be in blocking locations and that can be removed by the **delete-gene** operator. After this initial period, however, there is a steady decline in the effectiveness of **delete-gene** as the genotypes have fewer suboptimal genes, and deletion tends to result only in partial allocations. **Insert-gene** follows a unique trajectory, increasing gently until the middle of the run and declining thereafter. This probably reflects the utility of introducing new genes to genotypes until the population begins to converge on the optimum genetic makeup. From this point inserting new genes is of less utility than modifying the existing genes.

6.3.4 OP-GA summary and conclusions

The OP-GA approach adapts the parameterisation based on the fitness gains made by the GA. The approach is particularly suited to balancing the operator probabilities, especially for the P&P SOGA with its larger number of operators. The OP-GA was also effective in investigating the initial values for operator probabilities. For both the Land-Block and P&P OP-GAs the performance of the GA was significantly improved using OP-GA. For the Land-Block OP-GA this was seen in the improvement in the quality of solution found. For the P&P OP-GA, the improvement in performance was in the efficiency of the GA with a greatly reduced number of events needed to find near-optimal solutions.

6.4 Parameterisation Conclusions

Given the success of the OP-GA approach in improving performance, it is possible to conclude that it should be used within the land-use planning GAs as the mechanism to control operator probabilities. The overhead of extra computation needed for an OP-GA approach is more than offset by the increased efficiency of the GAs. The particular methods of the OP-GA cannot readily be translated for other parameters, but it is probable that if similar online adaptive methods were devised, performance could be further improved. The outputs from the OP-GA, especially the graphs of the operator trajectories, are helpful in identifying ineffective operators, and allow interpretation of the actions of the GA over the course of the run. A further source of information that could prove useful is the delta and passed-back-delta for each operator. This would allow the assessment of the effectiveness of the pass-back mechanisms and the facilitating roles of individual operators. The OP-GA approach should also be applied to the mGAs to determine the effectiveness of and trajectories for the operators when the goals of the GA are significantly different. Use of OP-GA would require a different gains-over metric with the genotypes selection-fitness the most promising candidate.

Chapter 7

Evaluation

7.1 SOGA Scaling

One of the design goals for the P&P representation was to enable problems with larger numbers of blocks to be tackled than would be possible with the Land-Block SOGA. To investigate SOGA performance for larger-scale problems three aspects were considered.

- *Quality of the solution(s) found:* the fitness of the best genotype and the mean fitness of the population.
- *Absolute efficiency:* the number and type of events required to find the solution.
- *Relative efficiency:* the CPU-time needed.

As the size of the test-problem increased it was expected, from initial testing, that the two SOGAs would find solutions of the same quality, but that the P&P would do so with a greater absolute efficiency than the Land-Block. In initial testing, there had been no significant difference in relative efficiency for a 65-block problem. For larger problems, the P&P SOGA was expected to be relatively more efficient.

7.1.1 Materials and methods

In order to investigate how well each of the SOGAs performs as the magnitude of the land-use planning problem increases, test farms with 50, 100 and 150 land-blocks were generated. The new farms were created by randomly selecting from the list of 95 land-blocks that make up Hartwood Research Station. It was not desirable to generate the

test farms entirely randomly as there are significant correlations between the bio-physical parameters of individual land-blocks. Random generation would result in combinations of bio-physical characteristics that cannot exist and thus cannot be evaluated by the land-use systems models or impact assessments. Land-blocks could occur more than once in the test farms without disrupting the operation of the greedy algorithm or fitness functions.

The Land-Block and P&P SOGAs used for the testing of scaling were those incorporating online-parameterisation for the operator probabilities. Other GA parameters remained the same as for testing of the OP-GAs. The fitness evaluation used was the financial impact. In addition to the metrics recorded for the OP-GA testing, cpu-time (CPU) was added, for this evaluation, to allow relative efficiency comparisons to be made. The two SOGAs were run 25 times for each test-farm and the results averaged. The difference and significance testing methods are those used previously.

7.1.2 Scaling results

Table 7.1 presents the average performance of the two SOGAs for the three test-farms.

Quality of solution

In each case the P&P SOGA achieves a higher MaxFit value but the difference is not significant. In all cases the SOGAs' MaxFit values are greater than 95% of the optimum fitness. For AvgFit the performance of the Land-Block SOGA is significantly better than the P&P SOGA for all but the 150-block test. The average population fitness is lower for the P&P SOGA because the minimum differences between genotypes are larger for P&P representation than for Land-Block. This is the result of enforcing genotype uniqueness and using a 5% granularity for the target land-use percentage.

Absolute efficiency

The scaling of the total number of reproductive events fits the expected pattern. The count of all events for the P&P SOGA remains nearly constant and the breakdown of the event types for each test-farm is nearly identical. This indicates that for each test-farm the P&P SOGA is solving a problem with the same complexity. The complexity of a problem for the P&P SOGA depends on the number of land uses present in the optimum

Table 7.1: Land-Block and P&P GA performance for scaling test problems

Metric	Land-Block-50	P&P-50	Diff	Sig
MaxFit (£M)	3.31	3.43	-0.12	0.154
AvgFit (£M)	3.29	3.27	0.02	0.000
All	2361	1006	1355	0.000
Gain	1077(46%)	176(18%)	901	0.000
NoGain	479(20%)	420(41%)	59	0.184
Duplicates	804(34%)	409(41%)	395	0.000
CPU (sec)	1503	398	1114	0.000
Metric	Land-Block-100	P&P-100	Diff	Sig
MaxFit (£M)	5.46	5.61	-0.15	0.156
AvgFit (£M)	5.33	5.29	0.40	0.000
All	2484	1042	1442	0.000
Gain	1531(62%)	187(17%)	1344	0.000
NoGain	384(15%)	439(42%)	-55	0.169
Duplicates	568(23%)	415(40%)	153	0.011
CPU (sec)	3262	641	2621	0.000
Metric	Land-Block-150	P&P-150	Diff	Sig
MaxFit (£M)	8.82	9.08	-0.26	0.159
AvgFit (£M)	8.86	8.71	0.15	0.156
All	2828	1024	1804	0.000
Gain	1854(66%)	180(17%)	1674	0.000
NoGain	393(13%)	453(44%)	-60	0.030
Duplicates	581(21%)	389(38%)	192	0.000
CPU (sec)	5917	1036	4935	0.000

allocation. This defines the number of genes for which the P&P SOGA must find the correct order and target-percentage.

For the Land-Block SOGA the total number of events (All) scales with the number of land-blocks. The break-down of the Land-Block SOGA events is interesting as it shows an increase in the number of gaining events. It is not possible to definitively explain this result, but it is possible to speculate that crossover remains effective for longer in the GA run when genotypes contain more genes.

Relative efficiency

Comparing the relative efficiency of the two SOGAs, there is a significant difference, with P&P consistently out-performing Land-Block. For the Land-Block SOGA, the value of the CPU metric approximately doubles for each additional 50 blocks. For the P&P SOGA, the increase in CPU required is determined almost entirely by the increased work being done by

the DSS greedy-algorithm, since the number of reproductive events is almost unchanged. The superiority of the P&P SOGA's relative efficiency is greater than expected. There are two reasons for this difference.

- *Improved parameterisation* of the P&P SOGA reduces the total number of reproductive events required to find the optimum. With a larger number of non-standard operators, the P&P SOGA has benefited more than the Land-Block SOGA from the use of the online-parameterisation. The adaptation of the P&P operators means that the most appropriate type of operator is applied at the correct phase of the GA run. This means the size of fitness gains made is increased.
- *Improved DSS greedy-algorithm efficiency* has also had an impact on the relative efficiency of the two SOGAs. While the changes made to the implementation of the greedy-algorithm during the course of the DSS development lie outwith the scope of the thesis they have had a positive impact on the relative efficiency of the P&P algorithm.

The P&P SOGA is thus consistently more efficient than the Land-Block for problems where the allocation of the most productive land-blocks first does not run counter to the fitness function being evaluated.

7.1.3 Scaling conclusions

As was expected, both SOGAs find solutions with similar quality but the P&P SOGA is more efficient. It is thus possible to conclude that the hybridisation of the land-use planning SOGA with the greedy-algorithm of the DSS has resulted in significant performance improvement. Since there is no-free-lunch for search and optimisation algorithms the increase in performance is bought at the expense of reducing the range problems for which the GA suitable. It is also likely that for problems where the optimum genotype has a larger number of genes, the difference in performance would be less marked.

7.2 mGA - Expert Evaluation

In addition to the hard evaluations of the mGA performance set out in chapter 5 it was decided that the sets of land allocations found by the mGAs should be compared with

those made by expert land-managers with a range of land-use perspectives. This would provide an opportunity to evaluate the performance of the mGAs and provide feedback on their potential as practical land-use planning tools.

7.2.1 Methods and materials

The approach taken to the generation of expert allocations was based on soft systems methods previously applied to the evaluation of decision support tools (van Beek 1995). These workshop-based methods are typically used in complex, value laden and conflict-prone situations (van Beek & Nunn 1995). They bring together stake-holders with differing perspectives and priorities and use facilitated subgroups to identify possible solutions. For evaluating the mGAs, the delegates were each tasked with producing an individual solution to a multi-objective land use allocation problem. These solutions would then serve as the basis for the “compromise” solutions “agreed” by two subgroups.

The delegates were selected to reflect the range of stake-holders typical for land use problems. The delegates thus included: land-owners, land-managers, non-governmental interest groups, rural investment institutions (banks) and academics. Table 7.2 lists the delegates and the two sub-groups.

Table 7.2: Delegates by subgroup

Subgroup 1	Subgroup 2
BA1 - bank adviser	SA2 - systems analyst
AG1 - agriculturalist	AG2 - agriculturalist
B1 - biologist	C2 - conservationist
E1 - estate manager	E2 - estate manger
F1 - farm manager	F2 - farm manager

The delegates were tasked with the same application used in the initial testing of the mGAs. For the Hartwood Research Station the delegates were asked to produce a *workable compromise* between the two conflicting objectives of diversity and financial returns. The degree of compromise would depend on the delegates’ particular priorities. It was hoped that, given the range of delegate backgrounds, the individual solutions would occur close to, and across the trade-off. The allocations made by the delegates would be based on a previously circulated information pack (see Appendix B). The materials included in this pack were those used by the DSS in its analysis: climate, soils and topographic data. In

addition to this data, it was decided to also provide information that would be readily available to planners such as aerial and ground level photographs, land capability and soil maps. All this information was designed to allow informed decisions to be made without prejudicing the range of allocations made.

Once the individual allocations were completed the delegates were divided into two subgroups each with a facilitator and reporter (members of the DSS development team). The role of the facilitator was to ensure that all participants had equal opportunity to contribute to the design of the group allocation. The reporter recorded the allocation(s) proposed and noted the factors considered by the delegates in the planning process, the heuristics used and how the different points of view were reconciled. Each delegate presented their individual allocation plan with an explanation of how it had been derived. The subgroup as a whole then sought to answer the following questions.

- Is the plan workable a whole?
- Are there parts of the plan that must be kept/dropped?
- Are there elements that can be added to improve the plan?

Following the presentations the groups were asked to agree a single plan by first defining the elements that were:

- fixed and non-negotiable,
- consensus allocations,
- land uses/locations that must not occur
- areas where any land use would be acceptable.

For both the individual and the subgroup allocations the delegates were offered the choice of producing the allocation either as maps or as tables. In all case the delegates used maps, an example is presented in Figure 7.1.

7.2.2 Assumptions and metrics

To ensure that it was possible to analyse any of the land-allocations proposed by the delegates or the subgroups, it was necessary to impose a number of constraints and assumptions on the possible allocations. Agreeing the constraints and assumptions with the

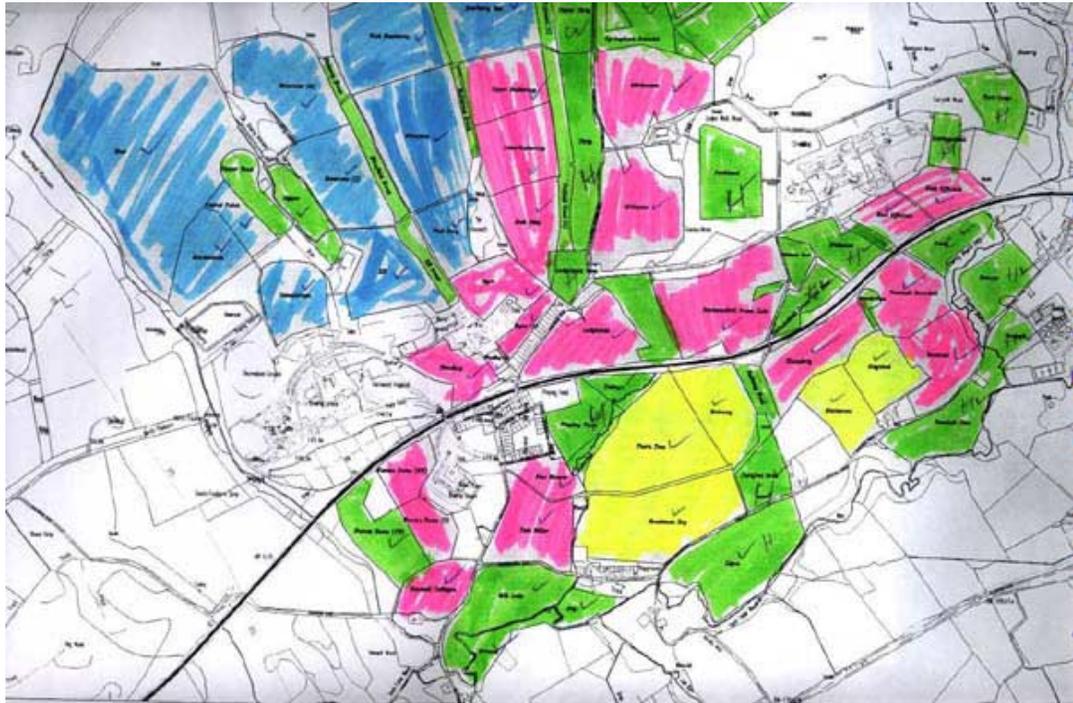


Figure 7.1: Example of the allocation maps prepared by delegates

delegates is central to the use of using soft systems methods and is a process of negotiation. The delegates are required to accept limitations imposed by the available analytical capabilities of the mGAs and the DSS. In turn the researcher must be prepared to modify the experimental set up, if possible, to make the scenario being tested as realistic as possible. Realism ensures the delegates engage with the task and that the solutions produced are credible reflections of their preferences. In the best case, the delegates will challenge the assumptions made or make explicit additional assumptions previously hidden within the experimental design.

The proposed assumptions and the results of the negotiation process are outlined below.

- The land-allocation is defined using a land-block based representation with each existing land parcel assigned one of the range of possible land uses. This assumption was maintained, but for the test application only a subset (5) of the possible (10) land uses were considered of practical value (trees were allocated under a general classification of broad-leaved and conifer species rather than by individual species). This restricted set of land uses (arable, cattle, sheep, broad-leaved and conifer-trees)

was also imposed on the mGAs to simplify the process of comparing results. The potential for diversification into non-agricultural activities on the Hartwood Research Station was noted but lay beyond the DSS' scope and several alternative land-use strategies were therefore disregarded.

- No changes to the existing pattern of field boundaries. This was accepted but noted as limiting for certain delegates' plans.
- No land may be bought or sold. One solution was proposed (G1-2 in following section) with land assumed to be rented to other land managers.
- The existing land uses do not limit future potential. This was accepted but all delegates went further and retained all existing woodland thus fixing 5% of the farm as common to all allocations.
- Capital and infrastructure are not limiting. While accepted, this was highlighted as one of the key constraints on real-world change of land use.

Over the course of the workshop it also became clear that, in discussing allocations, additional metrics were being used. These included total stock numbers, and summaries of the arable and forestry land use by percentage of the total enterprise area. These were adopted as secondary metrics for comparing delegates' land allocations.

7.2.3 Delegate and subgroup allocations

The workshop delegates produced 13 allocations, nine individual allocations and three from the subgroups (G1-1, G1-2 and G2). A fourteenth allocation (the current pattern of land use) was added to allow direct comparison with the *status quo*. The performance of the delegate allocations are tabulated in Table 7.3, with both the primary metrics (NPV and SW index) and the secondary descriptive metrics used by the delegates.

The delegate, subgroup and current allocations are also plotted in the search space defined by the two objectives in Figure 7.2. The figure also presents the two trade-offs found by running the Land-Block and P&P mGAs.

The theoretical range of SW values is from 0.0 (a mono-culture) to 1.6 (with a 20% allocation to each of the five possible land uses), the latter is shown by the upper dotted

Table 7.3: Current and Delegate Allocations (ordered by NPV)

	NPV (£M)	SW	Sheep	Cattle	B'leaf(Ha)	C'fer(Ha)	A'ble(Ha)
Current	3.71	1.048	1223	348	23.3	0	15.2
E1-2	4.00	0.869	1187	401	21.8	0	0
E2	3.77	1.146	1016	355	26.8	0	32.6
G1-1	3.56	1.135	967	347	49.0	0	14.4
AG1	3.36	1.234	544	329	78.5	0	33.9
AG2	3.16	1.439	597	307	42.9	40.7	33.9
SA2	3.08	1.271	1098	251	36.4	0	45.1
G2	3.05	1.31	802	271	76.0	0	38.6
BA1	2.74	1.098	1150	246	103.0	0	0
C2	2.69	1.454	591	255	54.9	21.7	31.5
F2	2.31	1.525	898	187	76.1	36.5	30.6
G1-2	2.31	0.961	1982	137	71.4	0	0
E1-1	2.12	1.327	1651	133	57.7	34.3	8.3
B1	1.58	1.508	768	110	113.6	47.9	29.7

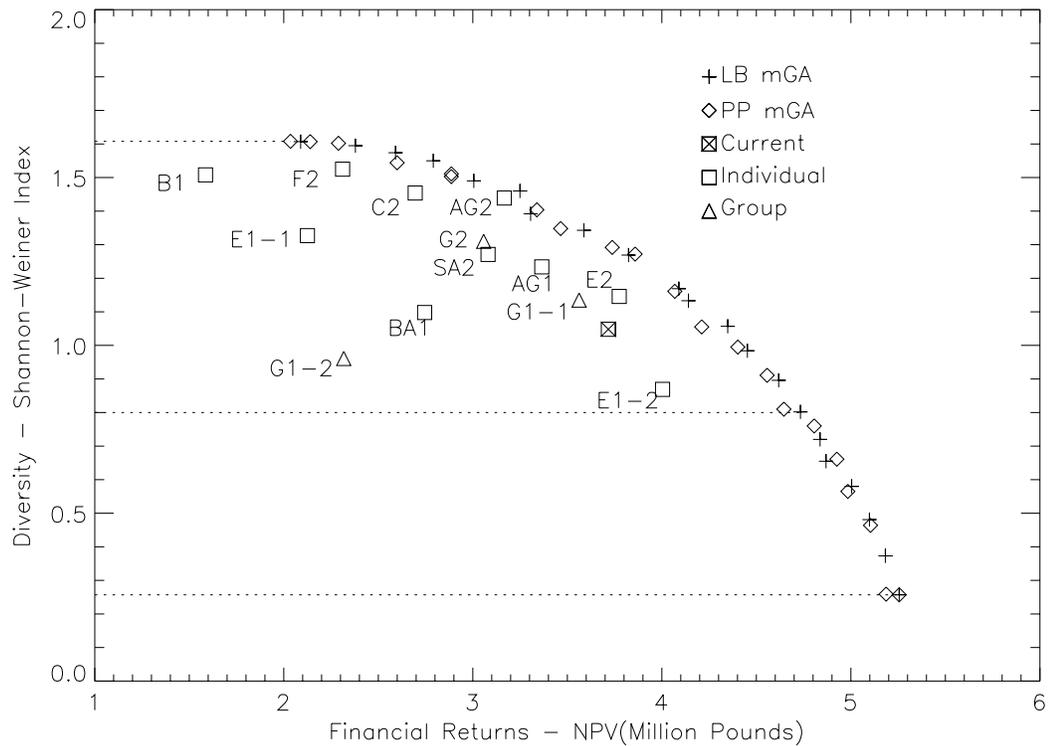


Figure 7.2: mGA and Land Manager Allocations

line in Figure 7.2. By preserving all the existing broad-leaved woodland, the delegates imposed a minimum diversity value of 0.25 (shown by the bottom dotted line in Figure 7.2). The mGA analysis also accepted the existing woodlands as fixed.

Beyond these explicit limitations, the delegates still defined allocations over only half of the possible range of SW values. The explanation for this lies in factors not explicitly taken account for in the mGA search. For all allocations in Table 7.3 both sheep and cattle are included as there are good animal welfare reasons based on rotational cleaning of grazing. The amount of land devoted to the two systems can vary, with the area of sheep typically less than that devoted to cattle. The area for sheep must however be large enough that it can integrate with the cattle rotation. This balance of sheep and cattle (with the fixed broad-leaved woodland) imposes a minimum SW diversity of approximately 0.8 (the middle dotted line in Figure 7.2).

7.2.4 Comparing the mGA and expert allocations

Inside these bounds the delegates proposed allocations with financial performance between £1.59M and £4.00M. The allocations B1, F2, C2, AG2, G2, SA2, Ag1, G1-1, E2 and E1-2 form a trade-off sitting just inside that found by the mGAs. The degree of financial under-performance is about 10-15%. The probable reason for this under-performance was the choice of the delegates to block together fields with the same land use in order to form management units, for example keeping all the cattle fields contiguous. While there are good management reasons for doing so, this does mean that the performance of the land-blocks in the financial terms defined is not optimised.

Not all proposed delegate allocations occur within the group forming the trade-off front. While sub-optimal, the allocations B1, BA1 and E1-1 are useful in indicating that solutions are not necessarily only located close to the trade-off front. For B1 and BA1 the financial performance of the allocation is harmed by the large extent of the low valued woodland areas (113 and 103 ha. out of a total area of 300 ha.) For E1-1 the reason for sub-optimality was the failure to allocate the arable land uses to land-blocks defined as suitable by LADSS. While the information pack did provide the information on which suitability could be judged, it did not provide either the criteria used by the DSS or maps of the land-blocks suitable for particular land uses. This was probably a fault in the analysis as it confuses the ability of the land managers to come up with compromise

land allocations with their ability to determine the suitability of land for particular land uses in unfamiliar situations. It was decided that, to minimise the possible impact of mis-allocation, the suitability criteria would be relaxed for arable crops; this can be rationalised through possible management intervention such as drainage. While this eliminated almost all of the illegal arable allocations from the delegates (those in E1-1 being exceptional in this regard), it also meant that the allocations found by the mGAs changed in character with much larger areas of arable crops resulting. The need for care in experimental design even in such soft systems analysis is apparent.

7.2.5 Evaluating the choice of planning objectives

The subgroup G1-2 allocation illustrates how carefully the mGA metrics have to be chosen, and the degree of multi-dimensionality in land management issues. The G1-2 allocation proposed the inclusion of available labour as an additional precondition. In this case the allocation assumed a single, full-time labour unit available with a pattern of land use dominated by sheep, but with some forestry and the remainder of the land rented to other farmers for seasonal grazing. The financial analysis of such a system depends not only on the gross margins but also on the ratio of input costs (including labour and machinery) to output revenue. The system proposed had very low input costs and thus could be much closer to the financial optimum than indicated by the metric used for the current analysis. There are also potential environmental benefits not measured by the SW index from the less intensive farming regime but social costs because of the reduced levels of local employment provided. There is thus significant potential for adding further objectives when considering current land-use planning applications.

7.2.6 Effectiveness of subgroups in finding compromise allocations

The process of compromise within the delegate subgroups is illustrated in Figure 7.3. The diagrams each show the subgroup delegates' allocations and the subgroup allocations they contributed to. It is clear from these diagrams that the group allocations sit at the mid-point between the individual allocations, indicating a willingness of the delegates to compromise. The degree of compromise, had real incomes or environmental impacts been at stake, is less certain. It is also clear that in both cases one delegate's allocations

formed the basis for the group allocation, for subgroup one this was provided by F1 (so the individual allocation of F1 is not shown) and for subgroup two the group allocation is very close to that of SA2.

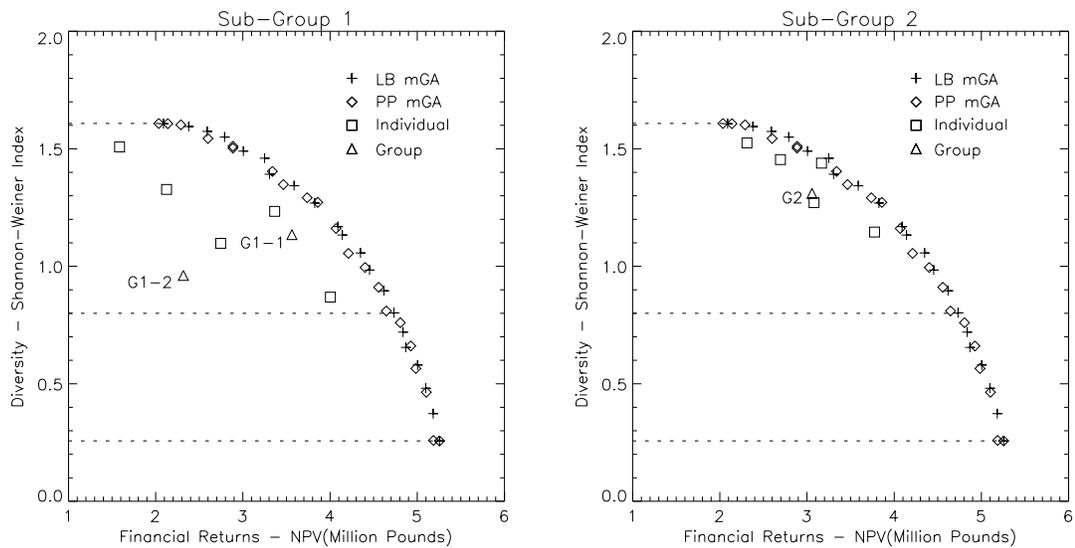


Figure 7.3: Individuals contributing to the subgroup allocations

7.2.7 Expert evaluation conclusions

Workshop-based soft systems methods were used to collect allocations made by land management specialists that could be compared with the Pareto-sets found by the mGAs. The comparison revealed that the practitioners operated within an agreed set of constraints that limited the range of allocations considered, but that, within those limits, solutions were found across the search space, and in the majority of cases, close to the Pareto-front defined by the mGAs. Practical management concerns, such as the desire for land-blocks of some land uses to be spatially contiguous, was hypothesised as the most likely reason for differences between the practitioner allocations and those of the mGAs. The utility of the mGAs would be improved by the use of spatial contiguity information (provided by the GIS), either as a constraint or as an explicit optimisation goal. The allocations found by the mGAs were, however, agreed by the land managers to be capable of forming the basis of management plans with modifications to individual land-blocks to ensure real world practicality.

The soft systems analysis also provided a wide range of qualitative evaluations for both

the mGAs and the DSS. These insights suggested improvements to: the range of analyses the DSS should provide; the metrics used by land managers in planning and comparing land allocations; heuristics that could be added as default allocation strategies and the key constraints required to ensure that the allocations found by the mGAs are workable. The workshop also provided anecdotal backing for the view that land management professionals faced with complex multi-objective planning problems want interactive decision support tools where a range of options can be examined and conclusions drawn on the trade-offs in costs and benefits. A wide choice of evaluation metrics and constraints that can be customised to the specific conditions of a particular land management unit are desirable.

The combination of hard metrics backed by soft systems-based analysis has proved effective both in evaluating the performance of the mGAs and in suggesting improvements to the range of analyses supported by the DSS.

Chapter 8

Conclusions

This thesis has investigated the usefulness of GAs as tools to assist land managers in creating land-use plans to achieve multiple-objectives. The application domain was strategic whole-farm planning, focusing on land management units with a single decision-maker and providing marketing-level plans. Plans to achieve multiple objectives are particularly relevant because of: tighter regulation of land management; the need to evaluate novel science-based, land-use strategies; increased public interest in land-use change and new land-owners with non-financial goals. The land-use plans provide information on both the mix and spatial pattern of land use that best achieves one or more objectives. The thesis has therefore contributed to both planning and genetic algorithm knowledge.

8.1 Contributions to Planning Knowledge

The contributions to planning knowledge were in creating and evaluating multi-objective land-use use planning tools.

8.1.1 Effective methodology

Land-use planning has lacked an effective methodology for tackling spatial, multi-objective problems. This thesis makes a significant original contribution by adapting and enhancing existing GA-based methodologies to create tools useful for real-world land-use planning tasks. The resulting tools are capable of tackling problems with spatially-explicit input parameters and may be applied to both single and multi-objective planning problems.

Existing land-use planning methods are often required to make simplifying assumptions that reduce the utility of their results. For example, non-spatial biophysical inputs may ignore significant constraints on the possible patterns of land use. Robust non-GA methods for land-use planning do exist, such as simulated-annealing but these methods cannot readily be adapted for use in multiple-objective applications.

For multi-objective applications the tools developed within the thesis provide the first method that allows the decision-maker to characterise the trade-off between conflicting non-commensurable objectives in a single use of the tool. Planning applications where objectives are non-commensurable and in conflict are common in strategic whole-farm planning. This is particularly true when financial and environmental objectives are considered. An *a posteriori* approach allows a search to be conducted, the structure of the trade-off between objectives established, and a rational decision made. The individual allocations found by the mGAs may also be passed to the DSS for further analysis with additional impact assessments. A decision made in this manner is transparent and can be quantitatively justified. The decision does not depend on weightings or orderings that may be corruptly manipulated to provide the solution desired. This is highly significant since the objectives in real-world multi-objective land-use planning problems are not only conflicting but value-laden. The solutions found by both mGA's were considered as acceptable marketing-plans by the land management experts who were consulted.

The efficiency of the methods, for problems typical in scale and complexity of those likely to be encountered in real-world applications, was also tested. The land-use planning GAs must be able to tackle real-world applications. The scaling testing for the two SOGAs indicated that while both can find acceptable solutions to problems of increasing size, the P&P does so more efficiently, both in terms of the number of reproductive events required and the computational effort required. The number of reproductive events required by the more complex P&P SOGA to find a near-optimum solution scales with the number of genes required to define the pattern of land use. For the Land-Block SOGA the number of reproductive events is determined by the number of land-blocks in the genotype.

8.1.2 Evaluation approach

The use of the workshop-based soft systems methods for evaluating the performance of the land-use planning tools takes a methodology employed within the social sciences and

applies it in a novel way to the evaluation of the planning tools developed. This approach provided both a quantitative evaluation of the effectiveness of the tools developed, by comparing their performance with that of an expert group, and a qualitative assessment of the usefulness of the tools outputs.

The use of the mGAs in the workshop drew out many of the assumptions and self-imposed constraints that land managers operate within, such as the need to preserve existing woodlands and the importance of using systems that ensure animal welfare. The additional constraints and secondary performance metrics highlighted by the soft-systems evaluation increased the usefulness of the land-use planning tools in a decision support context. In particular the visualisation of the land-use plans allowed the land-management experts to identify the need for the allocations to take account of the need for fields devoted to the same land use to be spatially contiguous. Incorporating such explicitly spatial constraints on the patterns of land-use will be possible in future developments of the land-use planning tools. The soft-systems approach is also very effective in engaging the interest of potential users of the tools where there may be significant reservations about the use of computer-based methods.

8.2 Contributions to Genetic Algorithm Knowledge

In pursuing the planning goals it was necessary to propose new approaches and test new methods particularly for the genetic representation used.

8.2.1 Representation

The analysis of the options for representing the land-use planning problem within a GA framework demonstrated that despite the spatial nature of the planning problem it was not necessary to maintain the spatial relationships between elements of the plan within the genotype structure. Indeed, doing so could be counter-productive in increasing the magnitude of the optimisation problem.

Two representations were tested. The Land-Block representation directly allocates land uses to individual fields, while the P&P GA makes allocations indirectly via the DSS greedy-algorithm. Existing management units were chosen over abstract divisions of space, for example grids or quad-trees, since to be useful for land-management purposes

the grid or quad-tree optimum solutions would have to be translated into individual fields, adding a further level of complexity to the problem.

The use of the an indirect representation (P&P), however, highlighted the utility of combining GAs with other algorithms, in this case a greedy algorithm within the decision support system. Use of this hybridisation approach reduced the number of reproductive events required by land-use planning GA to find near-optimal allocations for single-objective problems, especially for larger scale problems. For the multi-objective problems the poorer performance of the P&P representation pointed to the need for care when choosing the fitness sharing niche-size as this may result in a population-size with insufficient genetic diversity.

8.2.2 Operators and Non-fitness feedback

To support the P&P representation it was necessary to implement an extended operator set with operators modifying the class, parameter value and position of P&P genes. Of the eight operators modified to support the P&P representation, six were found to be of utility. The feedback of non-fitness information used to identify and eliminate genetic defects (parasitic genes and gene-pairs) or genotypes that were functionally identical was successful. The non-fitness feedback when combined with the enforcement of genotype uniqueness is effective in reducing problems associated with loss of genetic diversity.

8.2.3 Multi-objective GAs

The mGA implemented takes features from a range of existing GAs and mGAs, integrating them to create an effective land-use planning mGA.

- The mGA employs individual replacement with genotype uniqueness enforced, ensuring that a genetically diverse population is maintained and genotypes are only removed once they cease to be fit relative to other members of the population.
- Pareto-ranking based fitness assignment, niche-based fitness sharing and mating restrictions apply the appropriate selection biases to ensure the mGA population characterises the trade-off.
- The effectiveness of mating restrictions was examined. In contrast with other mGAs,

the window beyond which mating restrictions are imposed was set larger than the fitness sharing niche-size. The larger window was chosen as this would not exclude the possibility of recombining widely spaced genotypes but would bias selection in favour of inbreeding. The use of mating restrictions was seen to be effective in increasing the efficiency of the P&P mGA but for neither mGA was there an improvement in the quality of solution found.

- The use of population doping was effective in assisting the mGA to find the ends of the Pareto trade-off corresponding to the individual objective optima. This is significant as the mGA is less effective at finding the trade-off end points since they cannot be created by simply recombining Pareto-optimal solutions.

8.2.4 Parameterisation

The non-standard nature of the operator set of the GAs required that their parameterisation be investigated. EVOP, a method using structured experimentation, was used to set the probability of operators being applied. The GA was run in an EVOP framework, with parallel experiments conducted each using the same base population. The population associated with the best performing experiment then used as the start point for the subsequent set of experiments. While the EVOP approach is applicable to any GA parameter the improvement in GA performance achieved was insufficient to justify the methods operational use.

A second parameterisation method, online-parameterisation was also implemented. The method was modified to use all fitness-gaining information rather than only that provided when new maximum-fitness genotypes are created. Applied to the land-block GA, online-optimisation improved the GA's performance and replicated the operator-probability trajectories reported previously. For the larger number of P&P operators the online-parameterisation identified ineffective operators and adapted operator probabilities to significantly improve efficiency. The trajectories followed by the operators were non-linear and problem dependent pointing to the need to use online-optimisation in the operational use of the land-use planning GAs.

8.3 Future research

There are three areas of future research that can be identified. The first are *improvements* to the existing algorithms that would be worth pursuing, to increase their robustness, efficiency and scalability. Second there are *additional applications* that have relevance to land managers. Third there is the potential for the *integration* of the marketing-level planning tools with AI planners and schedulers to investigate the problems faced by land managers in implementing plans.

8.3.1 Improvements

The performance of the mGAs could be improved by the use of online-parameterisation. Of particular interest is how the dynamics of the mGA operator probabilities differ from those of the SOGAs. As part of online-parameterisation it would also be useful to track the raw delta gains and passed-back-delta values for individual operators. This would allow an investigation of the relative importance of the primary and enabling roles for operators.

It is also possible, based on the most recent mGA research, to make further improvements to the efficiency of the mGAs (Deb, Agrawal, Pratap & Meyarivan 2000). Options exist for alternative implementations of Pareto-ranking and the use of a locally defined crowding measure instead of the niche-based fitness sharing.

For most land-managers characterising the trade-off between two objectives is the primary concern. Testing the mGAs for three or more objectives is possible but presents a significant challenge of how best to communicate results. For three objectives, visualisation of the mGA population as a three-dimensional rendered surface is a possible solution.

Based on the experience of the expert evaluation of the mGAs it will be necessary to incorporate spatial constraints on the permitted allocations. In particular there is the need to constrain allocations of individual land uses to be spatially contiguous. This will be implemented by using either information derived from the GIS to constrain initialisation and operator actions or GIS-based fitness evaluations that penalise allocations that fail to respect the contiguity constraints. The GAs will also be tested using problems where there are significant spatial interactions between the land blocks.

8.3.2 Additional applications

For land-use planning in areas of semi-natural pasture, where land-cover is heterogeneous but where there are few existing management boundaries such as fences, it would be useful to build a land-use planning GA based on the quad-tree representation. Indeed it is possible that where year-to-year patterns of land use are important then the use of oct-trees defining the pattern of land use over time could be an appropriate approach. These alternative representations would of course also require specialised operators and possibly modified parameterisation. The underlying GAs could, however be based on the existing land-use planning tools.

With the increasing sophistication of the land-use systems models within the DSS, it is possible to consider incorporating risk in the planning of the patterns of land use. While an optimal pattern of land use may, under average conditions, produce the best returns, it may not do so over a time series of year-to-year weather variability around that mean. Using the more sophisticated land-use system models coupled with the mGAs, it would be possible to investigate the structure of the trade-off between risk and financial returns.

Given the success of the workshop-based evaluation of the mGAs, it is intended that the land-use planning tools will be applied to a wider range of real land management units. The application of the land-use planning tools for real problems will prioritise the developments undertaken.

8.3.3 Integration with AI planners and schedulers

In addition to more sophisticated marketing-planning, there is potential for linking the marketing-level planning tools with classical AI planners and schedulers. The reason for doing so is that while many land managers can appreciate the soundness of a marketing plan they may, due to their circumstances, be unable to make the transitional steps required to achieve the goal state. The phenomenon of land-management *lock-in* is well recognised.

Figure 8.1 illustrates a system where the GA-based land-use planning tools interface with an AI planner/scheduler. The marketing land-use planner supplies the goal states and the AI planner evaluates the feasibility of the transitions required to achieve the goal. The feasibility may be limited by the constraints within which the land-manager

is forced to operate, for example limits on funds to invest. Temporal conflicts in the scheduling of resources may also be limiting, for example patterns of crop rotations. The feasibility of the marketing plan would be fed back to the GA as either a supplementary- or primary-fitness evaluation. Such a system would have significant potential in identifying how public bodies can best intervene, such as by targeting grant-aid, to achieve positive land use changes.

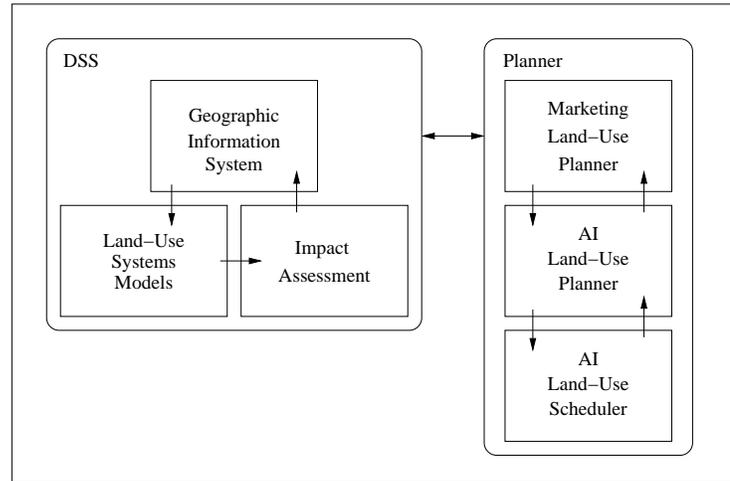


Figure 8.1: Future development of the land-use planning tools

8.4 Summary of outcomes

The GA-based land-use planning tools developed in this thesis are useful in providing marketing-level plans, defining the mix and spatial pattern of land uses, to meet one or more objectives. They have been successfully tested using problems typical, in scale, of expected real-world applications. The performance of the GAs is sufficiently robust and efficient that they may be employed for practical planning tasks. The GAs are flexible enough that they can support an iterative investigation of a land-use planning problem by the land manager rather than operating as a black box providing a single answer. The GAs form a key part of the DSS, being used to evaluate both alternative land-use strategies proposed by scientific research and the impact of proposed policy changes.

The hybridisation of the GAs with the DSS greedy-algorithm has reinforced the view that hybridisation results in powerful optimisation methods. To fully exploit the power of the hybridisation it was necessary to employ online-optimisation of the operator appli-

cation probabilities. The online-optimisation approach is well suited to investigating the effectiveness of alternative or representation-specific operators.

The mGAs developed are a significant advance in the methods available to investigate trade-offs between non-commensurable and conflicting objectives. The solutions found for the test application were as good as those proposed by a group of experienced land-managers. The mGA results also stimulated discussion of alternative land-management strategies between the land-managers. It is anticipated that the mGAs will be used as a basis for negotiation, in situations where there is disagreement between interest groups over proposed patterns of land management. The mGAs provide a quantitative means of evaluating the structure of the trade-off between objectives against which choices of particular solutions can be justified.

Bibliography

- Antonisse, J. (1989). A new interpretation of schema notation that overturns the binary encoding constraint, *in* J. Schaffer (ed.), *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA, pp. 86–91.
- Arthur, J. L. (1997). Clarification on the use of linear programming and GIS for land-use modelling, *International Journal of Geographical Information Science* **11**(4).
- Bäck, T. & Schwefel, H. (1993). An overview of evolutionary algorithms for parameter optimisation, *Evolutionary Computation* **1**(1): 1–23.
- Baker, J. E. (1989). Reducing bias and inefficiency in the selection algorithm, *in* J. Schaffer (ed.), *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA, pp. 14–21.
- Barrett, T. M. (1997). Voronoi tessellation methods to delineate harvest units for spatial forest planning, *Canadian Journal of Forestry Research* **27**: 903–910.
- Barták, R. (1999). On the boundary of planning and scheduling, *in* G. Petley, A. Codrington & R. Aylet (eds), *Proceedings of the 18th Workshop of the UK Planning and Scheduling Special Interest Group*, UK Planning and Scheduling Special Interest Group, University of Salford, pp. 28–39.
- Beedasy, J. & Whyatt, D. (1999). Diverting the tourists: a spatial decision-support system for tourism planning on a developing island, *JAG* **1**(3–4): 163–174.
- Bettinger, P., Sessions, J. & Boston, K. (1997). Using tabu search to schedule timber harvests subject to spatial wildlife goals for big game, *Ecological Modelling* **94**: 111–123.

- Beurden, U. C. J. V. & Padding, P. (1994). Areal units for environmental decision support: Theory and practice, *EGIS* pp. 352–362.
- Biethahn, J. & Nissen, V. (1995). *Evolutionary Algorithms in Management Applications*, Springer, Berlin.
- Boehlje, M. & Eidman, V. (1984). *Farm Management*, Wiley and Sons, New York.
- Bramlette, M. & Bouchard, E. (1991). *Handbook of Genetic Algorithms*, VNR Computer Library, Von Nostrad Reinhold, New York, chapter 10. Genetic Algorithms in Parametric Design of Aircraft, pp. 109–123.
- Brookes, C. (1997a). A parameterized region-growing programme for site allocation on raster suitability maps, *International Journal of Geographical Information Science* **11**(4): 375–396.
- Brookes, C. J. (1997b). Designing optimal patch configurations using a genetic algorithm, *GIS/LIS '97 Annual Conference and Exposition Proceedings*, pp. 393–399.
- Cartwright, H. & Harris, S. (1993). The application of the genetic algorithm to two-dimensional strings: The source apportionment problem, in S. Forrest (ed.), *Proceedings of the Fifth International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA, pp. 416–423.
- Carver, S. J. (1991). Integrating multi-criteria evaluation with geographical information systems, *International Journal of Geographical Information Systems* **5**(3): 361–367.
- Chuvieco, E. (1993). Integration of linear programming and GIS for land-use modelling, *International Journal of Geographical Information Systems* **7**(1): 71–83.
- Cox, L., Davis, L. & Qiu, Y. (1991). *Handbook of Genetic Algorithms*, VNR Computer Library, Von Nostrad Reinhold, New York, chapter 11. Dynamic Anticipatory Routing in Circuit-Switched Telecoms Networks, pp. 124–143.
- Davis, L. (1989). Adapting operator probabilities in genetic algorithms, in J. Schaffer (ed.), *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA, pp. 61–69.

- Davis, L. (1991). *Handbook of Genetic Algorithms*, VNR Computer Library, Von Nostrand Reinhold, New York, chapter 1. A Genetic Algorithms Tutorial, pp. 1–101.
- de Jong, K. (1990). Introduction to the second special issue on genetic algorithms, *Machine Learning* **5**(4): 351–353.
- Deb, K., Agrawal, S., Pratap, A. & Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimisation: NSGA-II, *KanGAL Report 200001*, Kanpur Genetic Algorithms Laboratory (KanGAL), Indian Institute of Technology, Kanpur, India.
- Densham, P. J. (1991). *Geographical Information Systems*, J. Wiley and Sons, chapter Spatial Decision Support Systems, pp. 403–412.
- Dibble, C. & Densham, P. J. (1997). Generating interesting alternatives in GIS and SDSS using genetic algorithms, *GIS/LIS '93 Annual Conference and Exposition Proceedings*, Minneapolis, Minnesota.
- Diplock, G. & Openshaw, S. (1996). Using simple genetic algorithms to calibrate spatial interaction models, *Geographical Analysis* **28**: 262–279.
- Eastman, J. R., Weigen, J., Kyem, P. A. K. & Toledano, J. (1995). Raster procedures for multi-criteria/multi-objective decisions, *Photogrammetric Engineering and Remote Sensing* **61**(5): 539–547.
- El-Swaify, S. A. & Yakowitz, D. S. (eds) (1998). *Multiple Objective Decision Making for Land, Water and Environmental Management*, CRC Press Ltd., USA.
- Eshelman, L., Caruana, R. & Schaffer, J. (1989). Biases in the crossover landscape, in J. Schaffer (ed.), *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA, pp. 10–19.
- Feng, C. & Lin, J. (1999). Using a genetic algorithm to generate alternative sketch maps for urban planning, *Computers Environment and Urban Systems* **23**: 91–108.
- Fonseca, C. M. & Fleming, P. J. (1994). An overview of evolutionary algorithms in multiobjective optimization, *Technical report*, Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, U. K.

- Fonseca, C. M. & Fleming, P. J. (1995a). Multiobjective optimization and multiple constraint handling with evolutionary algorithms i: A unified formulation, *Technical Report 564*, University of Sheffield, Sheffield, UK.
- Fonseca, C. M. & Fleming, P. J. (1995b). An overview of evolutionary algorithms in multiobjective optimization, *Evolutionary Computation* **3**(1): 1–16.
- Fonseca, C. M. & Fleming, P. J. (1998). Multi-objective optimisation and multiple constraint handling with evolutionary algorithms -part 1: A unified formulation., *IEEE Transactions on systems Man and Cybernetics-Part A: Systems and Humans* **28**(1): 26–37.
- Forman, R. & Godron, M. (1986). *Landscape Ecology*, Wiley and Sons, New York.
- Freisleben, B. (1997). *Handbook of Evolutionary Computation*, Institute of Physics, chapter Metaevolutionary approaches, pp. C7.2:1–C7.2:8.
- Goldberg, D. (1989). *Genetic Algorithms in Search, Optimisation and Machine Learning*, Addison-Wesley, Reading, MA.
- Goldberg, D., Deb, K., Kargupta, H. & Harik, G. (1993). Rapid, accurate optimisation of difficult problems using fast messy genetic algorithms, in S. Forrest (ed.), *Proceedings of the Fifth International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA, pp. 56–64.
- Goldberg, D., Korb, B. & Deb, K. (1989). Messy genetic algorithms: Motivations, analysis and first results, *Complex Systems* **3**: 493–530.
- Goldberg, D., Korb, B. & Deb, K. (1991). Don't worry be messy, in R. Belew & L. Booker (eds), *Proceedings of the Fourth International Conference of Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA, pp. 24–30.
- Haupt, R. L. & Haupt, S. E. (1998). *Practical Genetic Algorithms*, Wiley-Interscience, John-Wiley and Sons, New York.
- Holland, J. (1975). *Adaptation in natural and artificial systems*, second edn, The University of Michigan Press, Ann Arbor. Second edition, 1992, MIT Press.

- Horn, J., Nafpliotis, N. & Goldberg, D. (1994). A niched pareto genetic algorithm for multiobjective optimisation, in Z. Michalewicz (ed.), *Proceedings of the First IEEE Conference on Evolutionary Computation*, IEEE Service Centre, Piscataway, New Jersey, pp. 82–87.
- Hunter, W. G. & Kittrell, J. R. (1966). Evolutionary operation: A review, *Technometrics* **8**: 389–396.
- Ibaraki, T. (1997). *Handbook of Evolutionary Computation*, Institute of Physics, chapter Simulated annealing and tabu search, pp. D3.5:1–D3.5:2.
- Jankowski, P. (1995). Integrating geographical information systems and multiple criteria decision-making methods, *International Journal of Geographical Information Systems* **9**(3): 251–273.
- Jones, A., El-Swaify, S., Graham, R., Stonehouse, D. & Whitehouse, I. (1996). A synthesis of the state-of-the-art on multiple objective decision making for managing land, water and the environment, in S. A. El-Swaify & D. S. Yakowitz (eds), *Multiple Objective Decision Making for Land, Water and Environmental Management*, CRC Press LLC, USA, pp. 719–729.
- Kirley, M. & Green, D. S. (2000). An empirical investigation of optimisation in dynamic environments using the cellular genetic algorithm, in D. Whitley et al. (eds), *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2000)*, Morgan Kaufmann, San Francisco, Las Vegas, USA, pp. 11–18.
- Knowles, J. D. & Corne, D. W. (1999). Approximating the nondominated front using the pareto archived evolution strategy, *Evolutionary Computation* **7**(3): 1–26.
- Koza, J. R. (1993). *Genetic Programming*, MIT Press.
- Lochnar, R. H. & Matar, J. E. (1990). *Designing for Quality*, Chapman and Hall, London.
- Michalewicz, Z. (1992). *Genetic Algorithms + Data Structures = Evolution Programs*, Springer, Berlin.
- Moon, D., Jeck, S. & Selby, C. (1996). Elements of a decision support system: Information, model and user management, in S. A. El-Swaify & D. S. Yakowitz (eds), *Multiple*

-
- Objective Decision Making for Land, Water and Environmental Management*, CRC Press LLC, USA, pp. 323–334.
- Noreen, E. (1989). *Computer-intensive methods for testing hypotheses: an introduction.*, John Wiley and Sons, London.
- Openshaw, S. & Schmidt, J. (1996). Parallel simulated annealing and genetic algorithms for re-engineering zoning systems, *Geographical Systems* **3**(201-220).
- Roise, J. (1990). Multicriteria nonlinear programming for optimal spatial allocation of stands, **36**(3): 487–501.
- Samet, H. (1989). *The design and analysis of spatial data structures*, Addison Wesley, Reading, MA.
- Sequeira, R. A., Olson, R. L., Willers, J. L. & McKinion, J. M. (1994). Automating the parameterization of models using genetic algorithms, *Computers and Electronics in Agriculture* **11**: 265–290.
- Shaffer, J. (1985). Multiple objective optimisation with vector evaluated genetic algorithms, in J. Grefenstette (ed.), *Proceedings of the First International Conference on Genetic Algorithms*, Lawrence Earlbaum, Hillsdale, NJ, pp. 93–100.
- Spears, W. & de Jong, K. (1999). Dining with GAs: operator lunch theorems, in W. Banzhaf & C. Reeves (eds), *Foundations of Genetic Algorithms 5*, International Society for Genetic Algorithms, Morgan Kaufmann, p. 316.
- Srinivas, N. & Deb, K. (1995). Multiobjective optimisation using nondominated sorting in genetic algorithms, *Evolutionary Computation* **2**(3): 221–248.
- Syswerda, G. (1989). Uniform crossover in genetic algorithms, in J. Schaffer (ed.), *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA, pp. 2–9.
- Tarp, P. & Helles, F. (1997). Spatial optimization by simulated annealing and linear programming, *Scandinavian Journal of Forest Research* **12**: 390–402.

- Valenzuela-Rendón, M. & Uresti-Charre, E. (1997). A non-generational genetic algorithm for multiobjective optimisation, *in* Thomas Bäck et al. (eds), *Proceedings of the Seventh International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA., pp. 658–665.
- van Beek, P. (1995). A review of decision support tools (dsts), *Technical report*, Macaulay Land Use Research Institute.
- van Beek, P. & Nunn, J. (1995). *Introduction to Rapid Appraisals*, SyTREC Pty Ltd.
- van Dijk, S., Thierens, D. & de Berg, M. (1999). On the design of genetic algorithms for geographical applications, *in* W. Banzhaf et al. (eds), *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'99)*, Morgan Kaufmann, San Francisco, Orlando, Florida, USA., pp. 188–195.
- van Veldhuizen, D. A. & Lamont, G. B. (2000). Multiobjective evolutionary algorithms: Analysing the state of the art, *Evolutionary Computation* **8**(2): 125–147.
- Whitley, D. (1988). GENITOR: a different genetic algorithm, *Proceedings of the Rocky Mountain Conference of Artificial Intelligence*, A Publisher, Denver, CO, pp. 118–130.
- Whitley, D. (1989). The GENITOR algorithm and selection pressure - why rank-based allocation of reproduction trials is best, *in* S. J.D (ed.), *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA, pp. 133–140.
- Wolpert, D. & Macready, W. (1997). No free lunch theorems for optimisation, *IEEE Transactions on Evolutionary Computation* **1**(1): 67–82.
- Zhu, X., Aspinall, R. J. & Healey, R. G. (1996). ILUDSS: A knowledge-based spatial decision support system for strategic land-use planning, *Computer and Electronics in Agriculture* **15**: 279–301.
- Zitzler, E., Deb, K. & Thiele, L. (1999). Comparison of multiobjective evolutionary algorithms: Empirical results, *TIK-Report 70*, Institute für Technische Informatik and Kommunikationsnetze, Zürich.

- Zitzler, E. & Thiele, L. (1998). An evolutionary algorithm for multiobjective optimisation: the strength pareto approach, *TIK-Report 43*, Institute für Technische Informatik and Kommunikationsnetze, Zürich.

Appendix A

GA Class Hierarchies

This appendix presents the class hierarchies for the GAs implemented in this thesis. Three types of class may be identified: *GA classes* are those applied to the land-use planning problems; the *super classes* contain generic functionality and the *mixin classes* contain the methods used to override generic functionality.

A.1 GA

The GA classes hold many of the control parameters for the GAs, such as the fitness function(s) to be used, event counters and visualisation settings. The main division in the hierarchy is between the Land-Block and P&P representations. These classes are combined, using multiple inheritance from the SOGA and mGA mixins, to create the four GA classes in the middle of Figure A.1. The GAs used to investigate online-parameterisations are further specialised with the Online-parameterisation and EVOP mixins.

A.2 Population

The population classes are simply list-based structures each limited to a particular genotype class as its element type. The population class is where the methods controlling the process of evolution are implemented. These methods include selection, replacement and the calculation of selection-fitness values. There are only two mixin classes for populations, with the mGA and EVOP populations having specialised forms of evolution, see Figure A.2.

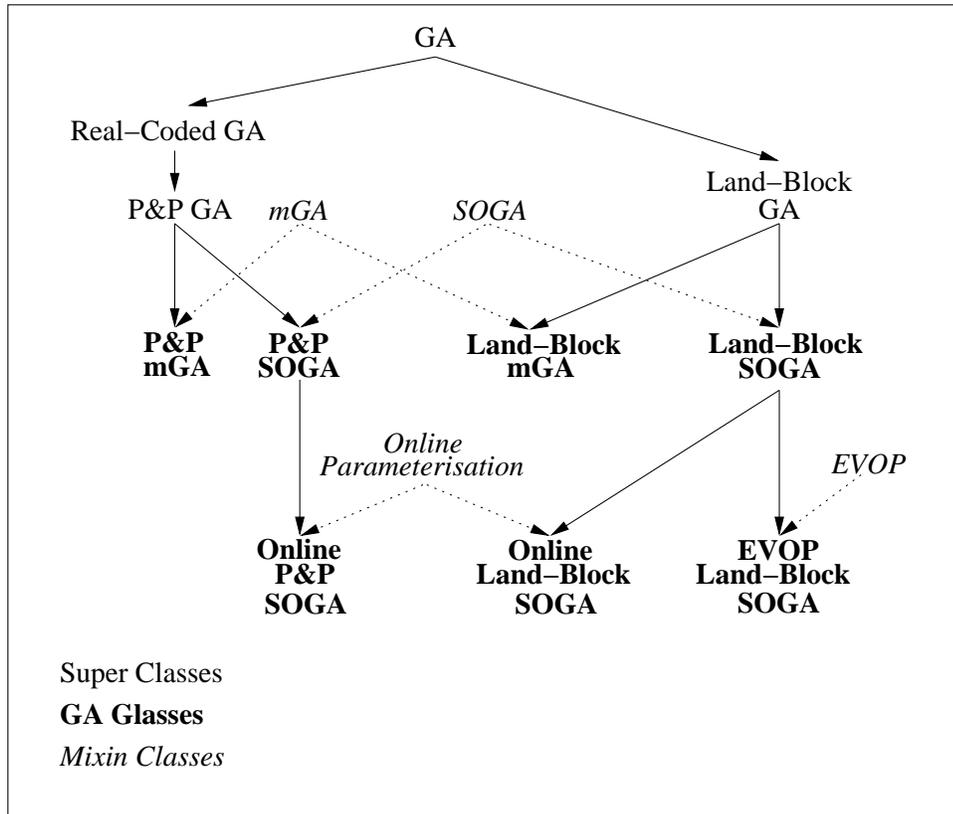


Figure A.1: GA class hierarchy

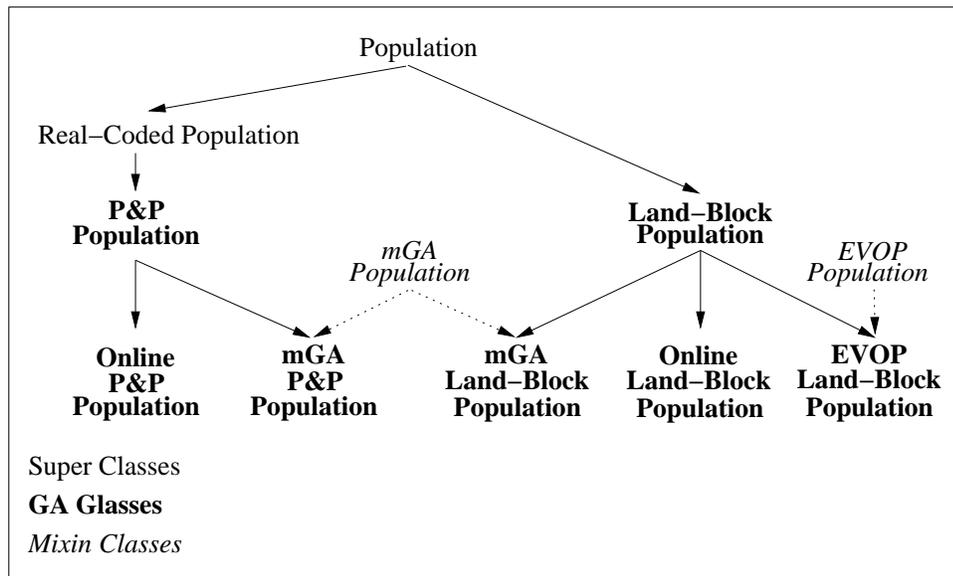


Figure A.2: Population class hierarchy

A.3 Genotype

Again the primary division is between the Land-Block and P&P classes with two mixin classes (online-parameterisation and mGA), see Figure A.3. The genotype is an array-based class with individual elements of specific gene classes. It is at the genotype level that fitness-function evaluation takes place, with genotypes passed to the DSS. All operators are partially implemented at the generic genotype level with representation-specific operations carried out at the gene level.

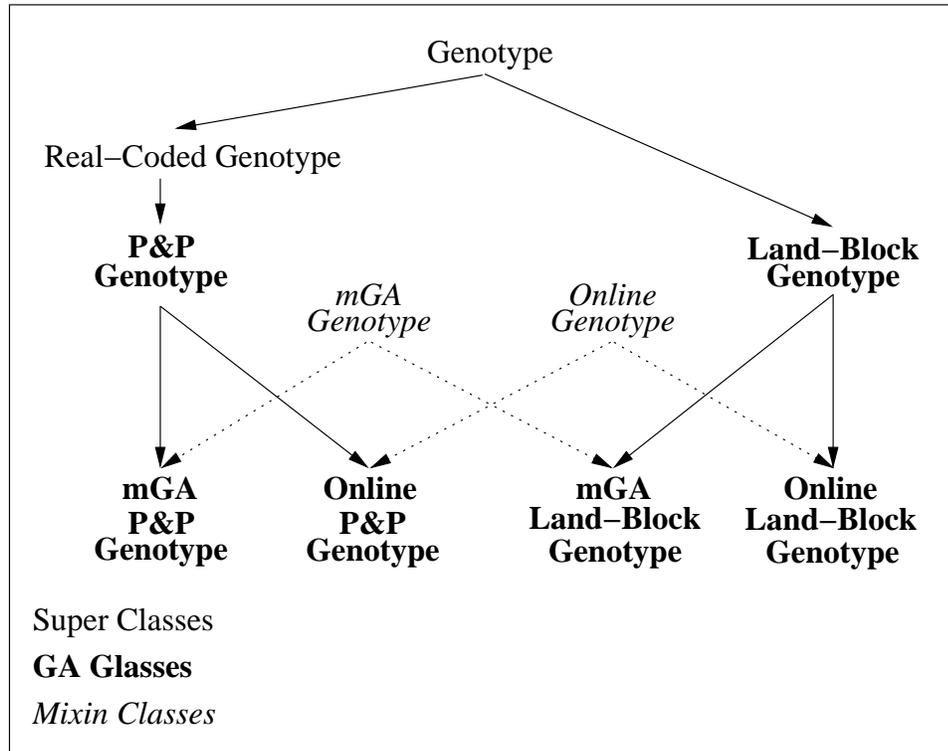


Figure A.3: Genotype class hierarchy

A.4 Gene

For genes the class hierarchy is much simpler with no mixin classes. Figure A.4 shows the main division between Land-Block and P&P genes. The sub-classes of P&P gene are used by the DSS to control the land uses allocated by the DSS greedy-algorithm.

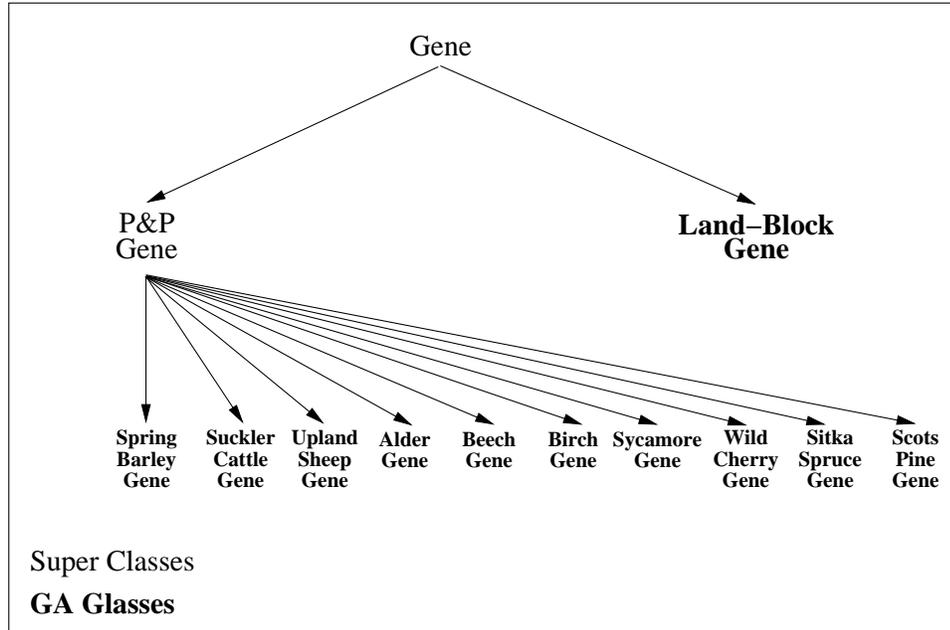


Figure A.4: Gene class hierarchy

A.5 Operator

The operator class is used to control the process of reproduction. The primary division is between the binary (two parent) and the unary (one parent) operators, see Figure A.5. These control the process of parent selection. Standard operators are those used for the initial testing of the GAs, with the simpler form of adaptation for operator probabilities. The online-operator mixin adds the parameters and methods needed to track the fitness gained by individual operators over the course of the GA run. The online-operator mixin also provides the methods used to record the ancestry of offspring genotypes.

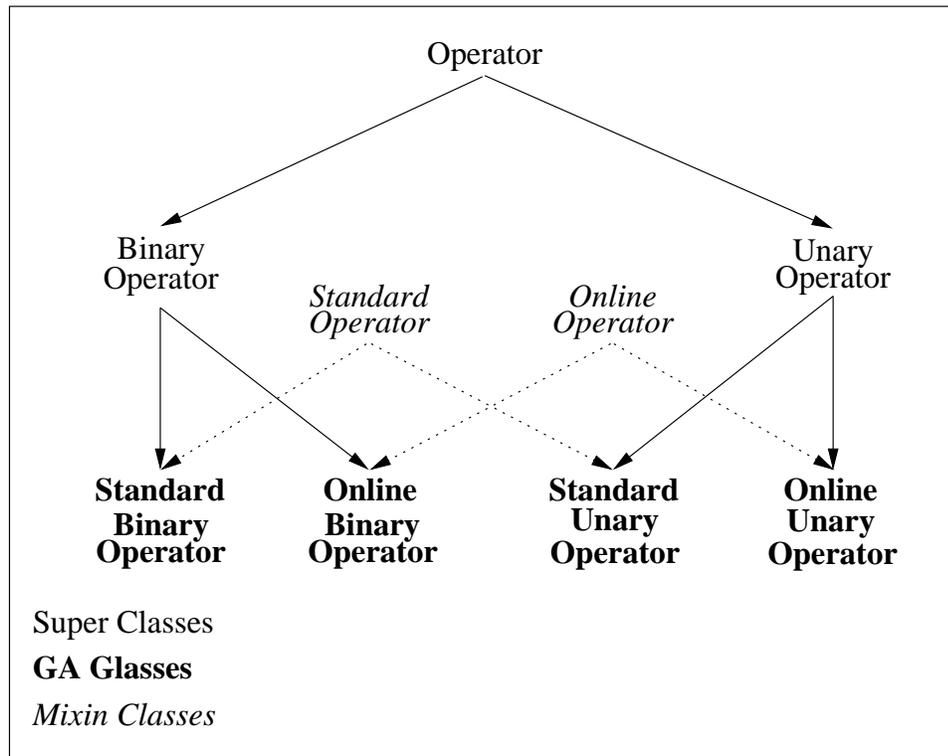


Figure A.5: Operator class hierarchy

Appendix B

Workshop Materials

This appendix contains the information used by the delegates at the soft systems workshop to produce their individual “workable-compromise” allocations balancing financial returns against diversity of land-use. The delegates were provided with:

- Introduction to the planning task.
- Description of the Hartwood Research Station.
- Agro-climatic data.
- Oblique aerial views.
- Hartwood Research Station and surrounding areas as evaluated by the Land Capability for Agriculture.
- Hartwood Research Station and surrounding areas as mapped by the Soil Survey of Scotland.
- Hartwood aerial photo-map including field names.
- Field-by-field biophysical description (*sample*)
- Blank field map
- Allocation sheet with delegate details

Appendix C

Published Papers

- Matthews, K.B., Craw, S., Elder, S., Sibbald, A.R. and MacKenzie, I. (2000) Applying Genetic Algorithms to Multiobjective Land Use Planning, in D. Whitley et al.(eds.) *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2000)*, Morgan Kaufmann, San Francisco, pp613-620.
- Matthews, K.B., Craw, S., Elder, S. and Sibbald, A.R. (2000) Evaluating Multi-Objective Land Use Planning Tools Using Soft Systems Methods, in M. Garagnani (ed.) *Proceedings of the 19th Workshop of the UK Planning and Scheduling Special Interest Group*, Open University, Milton Keynes, pp109-120.
- Matthews, K.B., Craw, S., MacKenzie, I., Sibbald, A.R. and Elder, S. (1999) Applying Genetic Algorithms to Land Use Planning, in G. Petley et al. *Proceedings of the 19th Workshop of the UK Planning and Scheduling Special Interest Group*, University of Salford, Manchester, pp109-115.
- Matthews, K.B., Sibbald, A.R. and Craw, S. (1999) Implementation of a spatial decision support system for rural land use planning: integrating GIS and environmental models with search and optimisation algorithms, *Computers and Electronics in Agriculture*, **23**:9-26.
- Sibbald, A.R., Milne, J. A., Matthews, K. B. and Birch, C.P.D. (2000) Decision support for hill and upland farms in the UK, in D. Gagneaux and J. R. Poffet (eds.), *Livestock Farming Systems: Integrating animal science advances into the search for sustainability*, EAAP publication no.97, Wageningen Pers: 284-288.