Case Study — Multiobjective Optimization of PWR Reload Cores

In the preceding sections the focus has primarily been on the describing the fundamental features of various stochastic optimization methods. When it comes to applying these methods to real-world engineering problems, it is often necessary to depart a long way from the "text book" implementation in order to obtain an algorithm that performs effectively. Clearly problem-specific adaptations are inevitably problem-specific, so the purpose of this case study is not to provide a blueprint for a real GA, but to illustrate by example the sort of issues which need to be addressed when tackling real-world problems.

PWR Reload Design

The Pressurised Water Reactor (PWR) reload core designer's task is to identify the configuration of fresh and partially burnt fuel and burnable poisons (BPs) (control material) which optimizes the performance of the reactor over the ensuing cycle, while ensuring that various operational constraints are always satisfied.

A typical PWR core contains 193 fuel assemblies arranged with quarter-core symmetry, as shown in Figure 1. At each refueling one third or one quarter of these may be replaced. It is common practice for fresh fuel assemblies to carry a number of BP pins. It is also usual to rearrange old fuel in order to improve the characteristics of the new core. This shuffling can entail the exchange of corresponding assemblies between core quadrants, which is equivalent

Figure 1: A Typical PWR Core.

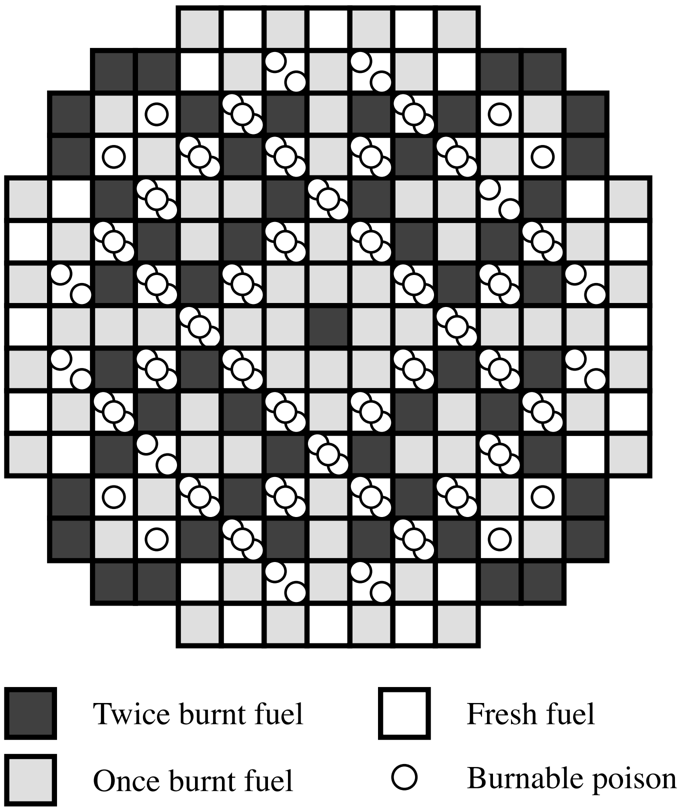
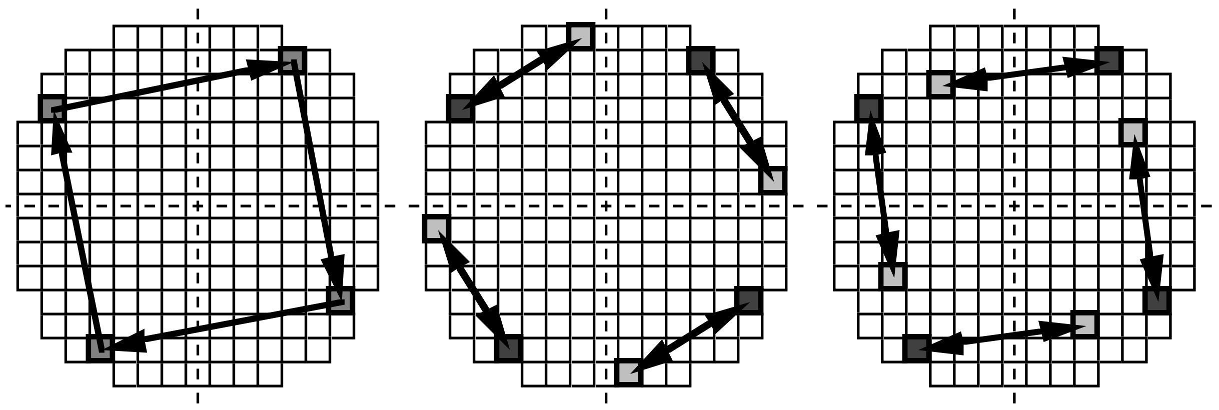


Figure 2: Typical Fuel Assembly Exchanges.



Orientation changing — Location changing — Location and orientation changing

to changing the assembly ‘orientations’, or the exchange of different assemblies, which changes their locations and possibly their orientations also. Examples of each exchange are shown in Figure 2.

Thus, a candidate core loading pattern (LP) of predetermined symmetry must specify:

- the fuel assembly to be loaded in each core location,
- the BP loading with each fresh fuel assembly, and
- the orientation of each burnt assembly (fresh fuel is totally symmetric).

Each old fuel assembly in a core quadrant will have unique neutronic characteristics, while fresh fuel of the same design will be identical.

It is readily apparent that the search for the best LP is a combinatorial optimization problem of tremendous magnitude — there are approximately 10^{43} possible LPs even if no BPs are used.

Nuclear power reactors inherently have some highly nonlinear characteristics. This means that whatever objective functions and constraints are used to define and quantify acceptable LPs, some of these parameters will inevitably be nonlinear functions of the problem’s control variables. One consequence of this nonlinear nature is that numerous local minima (or maxima) are created at constraint boundaries.

The unavoidable necessity of using computationally expensive reactor physics models of the core means that derivative information is not directly available.

In combination these attributes:

- high combinatorial dimensionality,
- nonlinear objectives and constraints,
- multimodality,
- computationally expensive objective and constraint evaluations, and
- lack of direct derivative information,

describe an extremely difficult optimization problem.

Two stochastic optimization methods which have been found to be particularly effective for searching the multimodal, nonlinear, combinatorial solution space of such problems are simulated Annealing [Kropaczek and Turinsky, 1991; Parks and Kropaczek, 1995; Stevens *et al.*, 1995] and GAs [Poon and Parks, 1993; DeChaine and Feltus, 1995; Bäck *et al.*, 1996].

The PWR reload core design problem has been tackled in many different ways [see Downar and Sesonske, 1988; Turinsky and Parks, 1999] and one interesting point to emerge from a review of past work is the diversity in objective functions chosen. It is apparent that the PWR reload core design problem is in reality a multiobjective optimization problem, where an improvement in one objective is often only gained at the cost of deteriorations in other objectives — trade-offs are necessary.

MOGA Implementation

The following sections describe a GA-based search method which is designed to perform true multiobjective optimization on PWR reload core design problems. This allows the trade-off surface between the objectives to be identified in a single run and offers the designer a family of LPs lying on this surface, from which to choose those worthy of further consideration.

There are six important problem-specific implementation features:

- The solution representation
- The crossover operator
- The mutation operator
- The archiving scheme
- The selection procedure
- The population assessment procedure

Figure 3: Representation of a Loading Pattern (with Quadrant Rotational Symmetry).

1	32	6	19	21	20	29	32
32	17	23	10	32	12	32	32
6	24	22	32	13	32	28	32
19	9	32	16	25	31	32	8
21	32	14	26	15	32	4	.
20	11	32	30	32	18	3	.
29	32	27	32	5	2	.	.
32	32	32	7	.	.	.	.

Fuel Assemblies

0	1	0	0	0	0	0	0
1	0	0	0	1	0	3	0
0	0	0	2	0	3	0	0
0	0	2	0	0	0	0	0
0	1	0	0	0	0	0	.
0	0	3	0	0	0	0	.
0	3	0	0	0	0	.	.
0	0	0	0	.	.	.	.

Burnable Poisons

0	0	3	1	1	1	2	0
1	3	3	3	0	3	0	0
0	1	2	0	3	0	3	0
2	1	0	2	2	1	0	0
2	0	1	2	2	0	2	.
2	1	0	3	0	0	1	.
3	0	1	0	2	3	.	.
1	0	0	0	.	.	.	.

Orientations

Solution Representation

Although traditional GAs map problems to strings of binary bits and manipulate these encodings, it is more natural, and therefore preferable [Goldberg, 1989], to represent each solution by three two-dimensional arrays (chromosomes), corresponding to the physical layout of the fuel assemblies, their BP loadings and their orientations respectively, as shown in Figure 3. This preserves important neighbourhood information, which would inevitably be lost if a one-dimensional string representation was used.

In the fuel assembly array, in Figure 3, each old (retained) fuel assembly has a unique identifier and there is one type of fresh fuel (#32). The assemblies on the interior quadrant boundaries are duplicated because of the LP's rotational symmetry. In the burnable poison chromosome, 0 indicates that no BPs are loaded in that location and other integers identify particular BP designs from the range available. In the orientation chromosome, the integers identify the quadrant from which that assembly comes (e.g. 0 indicates the bottom right quadrant, 1 the top right etc.).

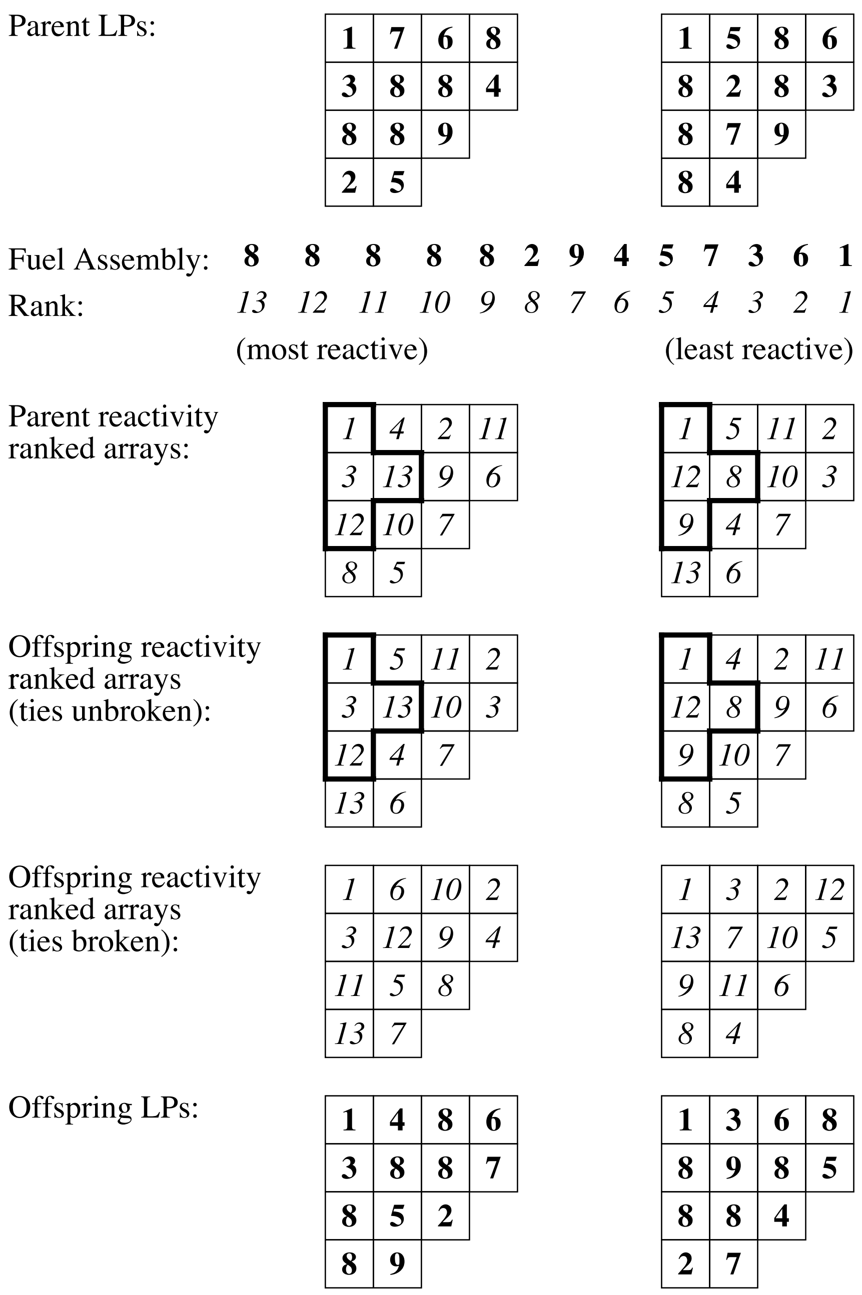
Crossover Operator

The role of the crossover operator is to combine information from parent solutions to create offspring solutions with, it is hoped, better objective function values. For combinatorial problems application-specific crossover operators are required to guarantee that valid offspring are produced — in this case to ensure that the fuel assembly inventory is maintained.

For this application Poon's Heuristic Tie-Breaking Crossover (HTBX) operator [Poon and Parks, 1993] is used. HTBX maps the parent fuel assembly arrays to ranked arrays based on the assemblies' reactivities.[1] It then combines randomly selected complementary parts of these arrays through a 'cut and paste' operation, and uses a simple tie-breaking algorithm to produce valid offspring reactivity-ranked arrays. Finally the assembly-ranking mapping is reversed to produce the offspring assembly LPs. The BP loadings and assembly orientations

1. Reactivity is a parameter related to the amount of fissile material in the fuel and thus a convenient indicator of other attributes of interest.

Figure 4: An Example of the Heuristic Tie-Breaking Crossover Operator.


Parent LPs:
Fuel Assembly:
Rank:
(most reactive)
(least reactive)
Parent reactivity ranked arrays:
Offspring reactivity ranked arrays (ties unbroken):
Offspring reactivity ranked arrays (ties broken):
Offspring LPs:

Parent LPs:

1	**7**	**6**	**8**
3	**8**	**8**	**4**
8	**8**	**9**	
2	**5**		

1	**5**	**8**	**6**
8	**2**	**8**	**3**
8	**7**	**9**	
8	**4**		

Fuel Assembly:	**8**	**8**	**8**	**8**	**8**	**2**	**9**	**4**	**5**	**7**	**3**	**6**	**1**
Rank:	*13*	*12*	*11*	*10*	*9*	*8*	*7*	*6*	*5*	*4*	*3*	*2*	*1*

(most reactive) (least reactive)

Parent reactivity ranked arrays:

1	*4*	*2*	*11*
3	*13*	*9*	*6*
12	*10*	*7*	
8	*5*		

1	*5*	*11*	*2*
12	*8*	*10*	*3*
9	*4*	*7*	
13	*6*		

Offspring reactivity ranked arrays (ties unbroken):

1	*5*	*11*	*2*
3	*13*	*10*	*3*
12	*4*	*7*	
13	*6*		

1	*4*	*2*	*11*
12	*8*	*9*	*6*
9	*10*	*7*	
8	*5*		

Offspring reactivity ranked arrays (ties broken):

1	*6*	*10*	*2*
3	*12*	*9*	*4*
11	*5*	*8*	
13	*7*		

1	*3*	*2*	*12*
13	*7*	*10*	*5*
9	*11*	*6*	
8	*4*		

Offspring LPs:

1	**4**	**8**	**6**
3	**8**	**8**	**7**
8	**5**	**2**	
8	**9**		

1	**3**	**6**	**8**
8	**9**	**8**	**5**
8	**8**	**4**	
2	**7**		

are all inherited from one or other parent. Thus, the reactivity distribution (and it is hoped, in consequence, other attributes) of an offspring LP resembles, but is not necessarily identical to, parts of both parents. This procedure is illustrated by an example in Figure 4.

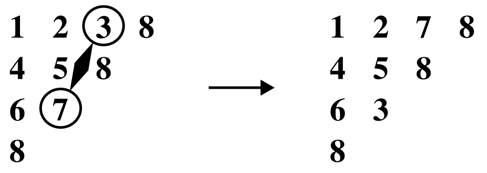
Mutation Operator

The mutation operator makes small changes to trial LPs, and is necessary to ensure that the entire search space is accessible. For this application the mutation operator performs one or two fuel assembly shuffles, randomly allocating allowed BP loadings and orientations to the assemblies affected, as illustrated by the examples in Figure 5. It is used as an alternative to

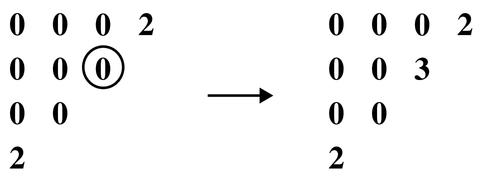
Figure 5: Examples of the Mutation Operator.

Do any or all of:

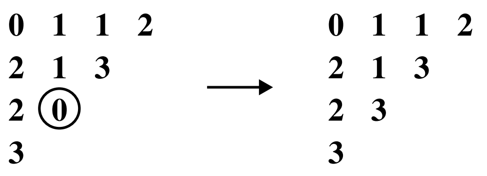
- Perform a binary fuel assembly exchange



- Make an allowed change to the BP chromosome



- Make an allowed change to the orientation chromosome



crossover, i.e. offspring are produced using either mutation or crossover but not both, the choice between operators being made randomly. The relative frequencies with which these operators are chosen are approximately 25% and 75% respectively, this ratio having been determined (by extensive testing) to give good performance on PWR reload design problems.

Archiving

This GA implementation uses the multiobjective archiving scheme detailed in the previous handout, i.e. nondominated solutions are recorded. The only variation from the scheme described there is that, because a large amount of information is recorded about each LP which is archived, the archive is restricted in size. An element of dissimilarity archiving is therefore used to limit the number of LPs recorded. The overall archiving logic is as follows:

- After each trial solution has been evaluated it is compared with existing members of the archive.
- If it dominates any members of the archive, those are removed and the new solution is added.
- If the new solution is dominated by any members of the archive, it is not archived.
- If it neither dominates nor is dominated by any members of the archive, it is archived if it is sufficiently dissimilar to existing archive members.[2] Once the archive is full, any new non-dominated solution to be archived in these circumstances replaces the most similar one in

the archive. This means that nondominated solutions are discarded from the archive as the search progresses, but such discarding is unavoidable unless the archive size is set so large that all nondominated solutions encountered can be stored.

Selection

In the example that follows, up to 25% of the parents for each new generation are chosen from the archive, thus introducing multiobjective elitism to the selection process. Parents are chosen from the current population using the *Pareto-based ranking* scheme described in the previous handout to assign selection probabilities and *stochastic remainder selection without replacement* (see the handout on Genetic Algorithms) to do the selecting.

Population Assessment

In the example that follows the Generalized Perturbation Theory based reactor model employed in FORMOSA-P [Kropaczek *et al.*, 1994] was used to evaluate LPs, but, in principle, the evaluation of objectives and constraints can be performed using any appropriate reactor physics code.

Code Performance

Figures 6 and 7 illustrate the performance of this multiobjective GA by plotting projections in two-objective space of the initial and final populations and the final archive contents for an optimization run on a PWR core in which there were three objectives:

- feed enrichment minimization (minimization of the amount of fissile material in the fresh fuel),
- discharge burnup maximization (maximization of the energy generated by the fuel to be discharged at the end of the cycle under consideration), and
- radial form factor (RFF) minimization (minimization of the ratio of the peak to average assembly power throughout the cycle, a safety related objective).

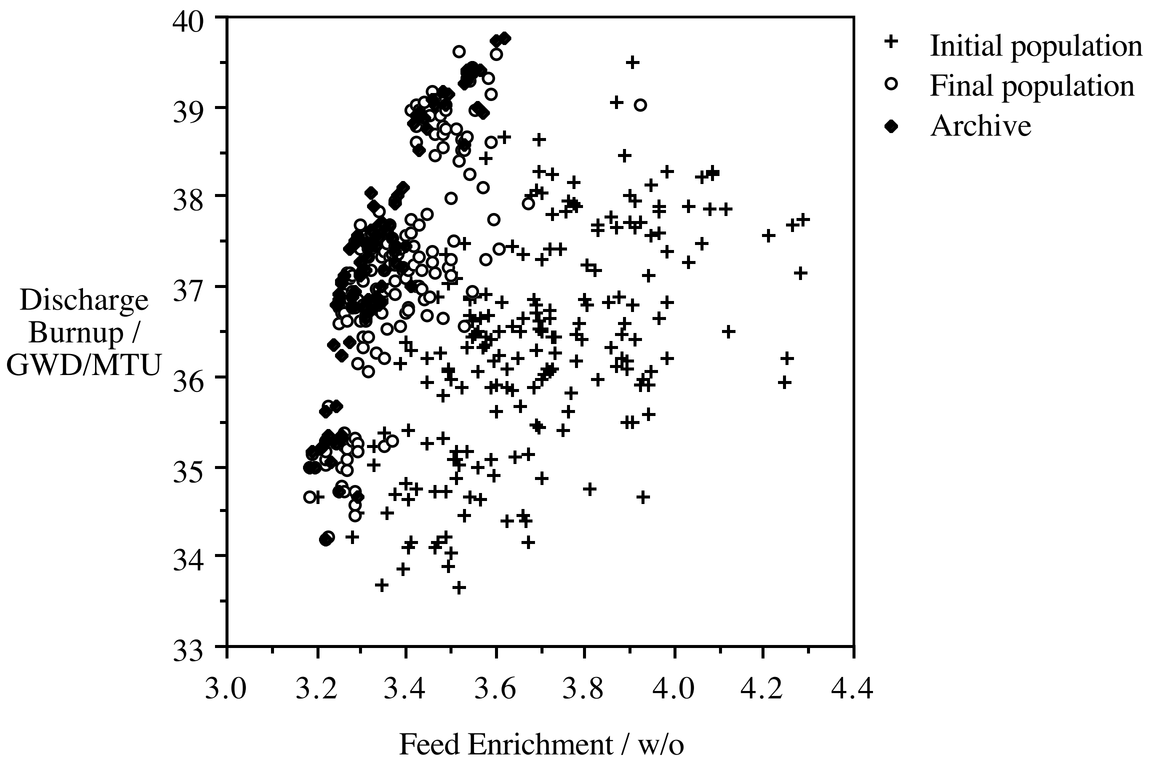
In this run a population of 204 was evolved for 51 generations, so that 10 404 LPs were examined in all.

It can be seen that the final population is on average much better than the initial population with respect to all three objectives. During the run 1186 LPs were archived, which is a good indication of the effectiveness of the search method. (Note that only solutions with RFF values below 1.50 were archived.)

The archive of 100 LPs, and indeed the final population, gives the designer a clear picture of the trade-off surface between the competing objectives. The (expected) strong trade-off

2. For these purposes the degree of dissimilarity between two LPs is defined in terms of the difference in their reactivity distributions.

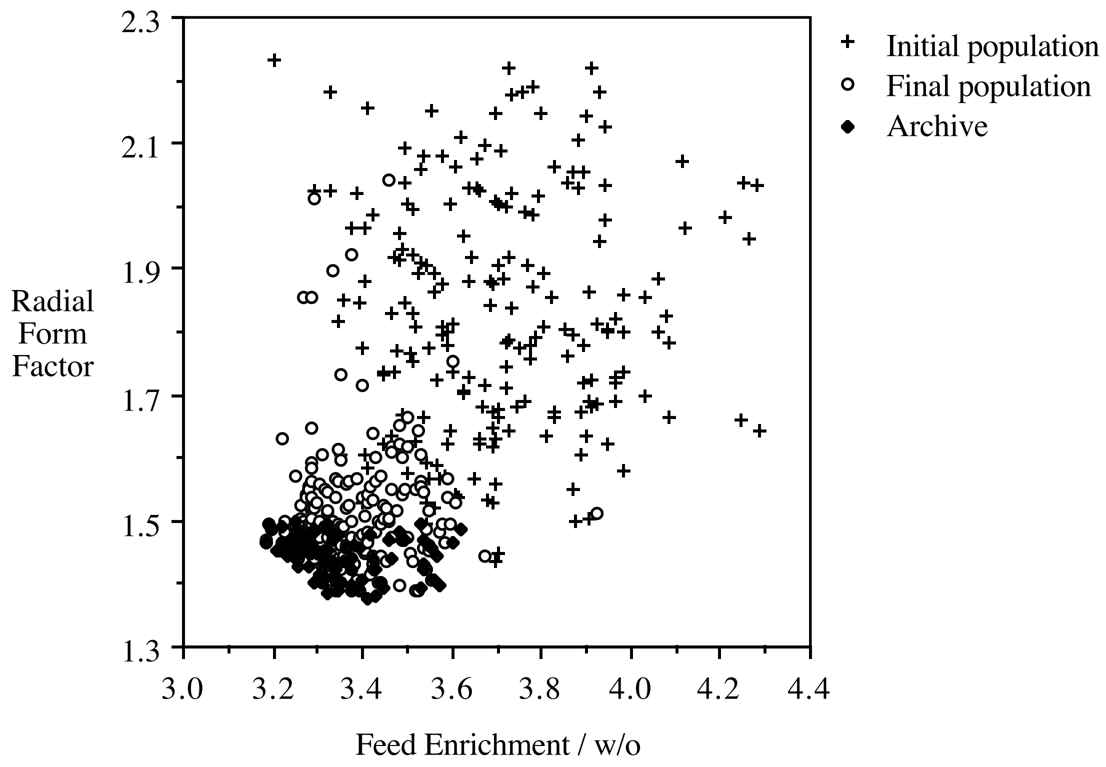
Figure 6: MOGA Performance — Discharge Burnup vs Feed Enrichment.



between feed enrichment and discharge burnup is clearly shown. A trade-off between feed enrichment and RFF is also indicated.

Figure 8 shows the evolution in two-objective space of the contents of the archive. It is

Figure 7: MOGA Performance — Discharge Burnup vs Radial Form Factor.



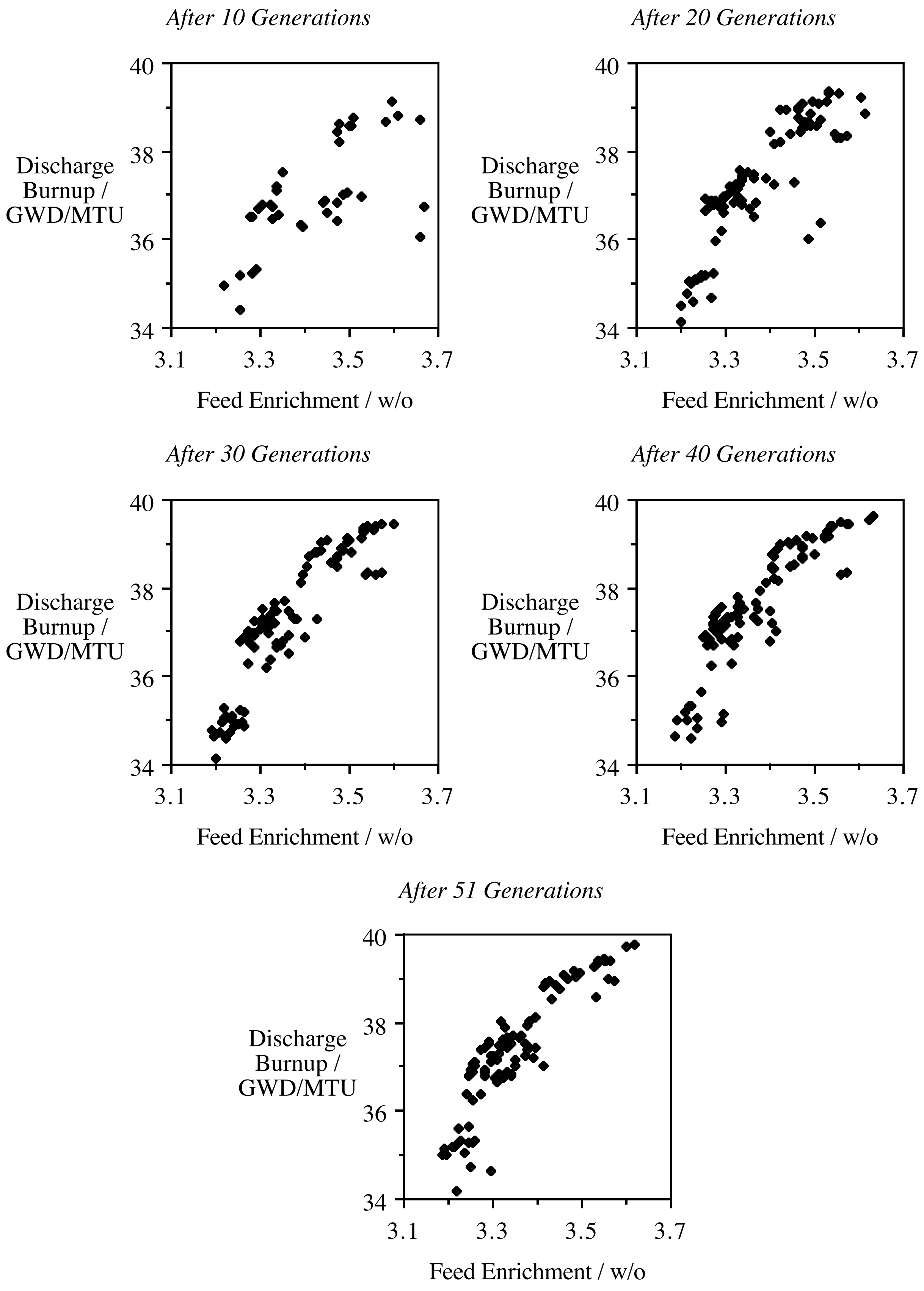


Figure 8: MOGA Archive Evolution — Discharge Burnup vs Feed Enrichment.

apparent that the general shape of the trade-off surface is established quite early, and, therefore, if the purpose of performing multiobjective optimization is just to identify a part of the search space worthy of closer examination, the length of the MOGA run could be made significantly shorter, with an associated reduction in execution time.

These results show that a single MOGA run can present the reload core designer with a wealth of information about the range of achievable values of different (conflicting) objectives and the necessary trade-offs between them. Using this information the designer is able to make a much better informed decision about the areas of the search space worthy of closer examination.

References

Bäck, T., J. Heistermann, C. Kappler and M. Zamparelli (1996) “Evolutionary Algorithms Support Refueling of Pressurized Water Reactors” *Proc. Third IEEE Conference on Evolutionary Computation*, Piscataway, NJ, IEEE Press, 104-108.

DeChaine, M.D., and M.A. Feltus (1995) “Nuclear Fuel Management Optimization using Genetic Algorithms” *Nucl. Technol.* **111**, 109-114.

Downar, T.J., and A. Sesonske (1988) “Light Water Reactor Fuel Cycle Optimization: Theory Versus Practice” *Adv. Nucl. Sci. Tech.* **20**, 71-126.

Goldberg, D.E., (1989) *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison Wesley, Reading, MA.

Kropaczek, D.J., and P.J. Turinsky (1991) “In-core Nuclear Fuel Management Optimization for Pressurized Water Reactors Utilizing Simulated Annealing” *Nucl. Technol.* **95**, 9-31.

Kropaczek, D.J., P.J. Turinsky, G.T. Parks and G.I. Maldonado (1994) “The Efficiency and Fidelity of the In-Core Nuclear Fuel Management Code FORMOSA-P” *Reactor Physics and Reactor Computations* (Edited by Y. Ronen and E. Elias), Ben Gurion University of the Negev Press, 572-579.

Parks, G.T., and D.J. Kropaczek (1995). “Nuclear Fuel Management” *Adaption of Simulated Annealing to Chemical Optimization Problems* (Edited by J.H. Kalivas), Elsevier Science, Amsterdam, 205-222.

Poon, P.W., and G.T. Parks (1993) “Application of Genetic Algorithms to In-Core Nuclear Fuel Management Optimization” *Proc. Joint Int. Conf. Mathematical Methods and Supercomputing in Nuclear Applications*, Karlsruhe, **1**, 777-786.

Stevens, J.G., K.S. Smith, K.R. Rempe and T.J. Downar (1995) “Optimization of Pressurized Water Reactor Shuffling by Simulated Annealing with Heuristics” *Nucl. Sci. Eng.* **121**, 67-88.

Turinsky, P.J., and G.T. Parks (1999) “Advances in Nuclear Fuel Management for Light Water Reactors” *Adv. Nucl. Sci. Tech.* **26**, 137-165.