Memetic Algorithms for Combinatorial Optimization Problems in the Calibration of Modern Combustion Engines

K. Knödler, J. Poland, and A. Zell
University of Tübingen,
Department of Computer Science
Sand 1, D - 72076 Tübingen, Germany
knoedler@informatik.uni-tuebingen.de

A. Mitterer
BMW Group Munich,
80788 München, Germany
alexander.mitterer@bmw.de

This work focuses on the high relevance of *memetic algorithms* (MAs, see e.g. [1] or [2]) in the calibration of modern combustion engines[1]. The MA used in this work is given by the following pseudo code:

```
begin
    for j := 1 to μ do
        i := Local-Search(generateSolution());
        add individual i to P;
    endfor;
    repeat
        for i := 1 to n_cross · μ
            select two parents i_a, i_b ∈ P randomly;
            i_c := Local-Search(Recombine(i_a, i_b));
            add individual i_c to P;
        endfor;
        for i := 1 to n_mut · μ
            select an individual i ∈ P randomly;
            i_m := Local-Search(Mutate(i));
            add individual i_m to P;
        end repeat;
        P := select(P);
        if P converged then P := mutateAndLS(P);
    until terminate=true;
end;
```

A random starting population is tackled by a local search operation, e.g. a *Monte Carlo Algorithm* (MC) in order to receive local optimum solutions. The numbers of individuals taken for crossover and for mutation are given by $n_{cross} \cdot \mu$ and $n_{mut} \cdot \mu$, respectively. After the convergence of the algorithm, all individuals but the best one are mutated and the MC is applied.

We studied three combinatorial optimization problems in the field of engine calibration. For transparency reasons, only the *final smoothing of maps defined by look-up tables* that was introduced in [4] is discussed here. Figure 1 visualizes a simple problem instance: there are three sets of candidates labeled with circles, diamonds, and squares. Every set itself defines a possible look-up table and hence a possible map. The idea is to mix up the sets in order to receive one final well defined look-up table with best smoothness properties. The mixing is noncritical, since the candidates at one grid point yield only slightly different engine behavior, e.g. fuel consumption or exhaust emission. We use

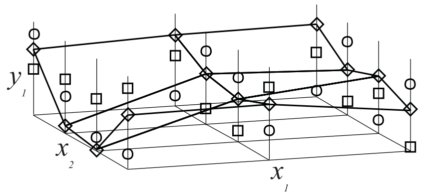

Figure 1: Candidates for a look-up table.

the variable alphabet encoding, i.e. each grid point j corresponds to one position of a chromosome that takes n_j values: $v = (v_j)_{j=1}^{N} \in \bigotimes_{j=1}^{N}\{1 \ldots n_j\}$. Here, n_j is the number of candidates available at (x_1^j, x_2^j). This encoding allows standard mutation and crossover operations. We define the objective function $\phi(M) = \sum_{1 \le i < j \le n} neigh(i,j) \cdot |y^{(i)} - y^{(j)}|$ as smoothness of a map M where $neigh(i,j)$ is 1 for neighboring grid points, otherwise 0.

The consequent application of the MC (after each recombination and mutation) given by

```
repeat N times
    Choose j ∈ 1, . . . , M randomly;
    Find all k ∈ 1, . . . , n_j such that φ becomes minimal;
    Choose randomly one of these k;
end repeat;
```

significantly improves the regular genetic algorithm (GA). Beyond it, the MA outperforms a hybrid GA that uses the MC as mutation operation by up to 25%. Since the MC works locally it does not need complete fitness evaluations. However, to compensate higher computation times of the MA, the GAs were run with increased generation numbers and population sizes. In addition island parallel populations were used.

References

[1] P. Moscato, Memetic Algorithms: A Short Introduction, New Ideas in Optimization, D. Corne, M. Dorigo and F. Glover, McGrawHill, London, 1999.

[2] P. Merz, Memetic Algorithms for Combinatorial Optimization Problems, PhD thesis, University of Siegen, 2000 (provided by the library)

[4] J. Poland et al., Evolutionary Search for Smooth Maps in Motor Control Unit Calibration, Stochastic Algorithms: Foundations and Applications (SAGA), 2001

[1] This research has been supported by the BMBF (grant no. 01 IB 805 A/1).