# Solving Problems with Overlapping Building Blocks

**Buthainah Al-Kazemi** **Chilukuri K. Mohan**
2-177 CST, Dept. of EECS
Syracuse University, Syracuse, NY 13244-4100, U.S.A.

Let $|B|$ be the number of 1s in vector $B$, $B[i:j]$ the substring of $B$ from position $i$ to $j$ (inclusive of $i, j$), and $B[i:]$ the substring of $B$ beginning from position $i$ to the end of $B$, and $g_k$ a function on $k$-bit strings (where $k > 0$).

For $k > \theta \geq 0$, a "$\theta$-overlap function" is defined as:

$$f_{k,\theta}(B) = \begin{cases} |B| \text{ if the length of } B < k, \\ g_k(B[0:k-1]) + f_{k,\theta}(B[k-\theta:]) \text{ else.} \end{cases}$$

Note that an overlap problem can be equivalently represented as a non-overlap problem on longer chromosomes with equality constraints among some components, e.g., $f_{3,2}(abcde) \equiv f_{3,0}(\underline{abc}\ \underline{bcd}\ \underline{cde})$.

This paper reports experimental results obtained on problems of this class [1], using a deceptive function

$$g_k(A) = \begin{cases} 1 \text{ if } |A| = k, \\ 0.9 - |A|/2k \text{ otherwise.} \end{cases}$$

We have experimented with traditional crossover operators, as well as *Gradient Selective Crossover* (SX) [2, 3], in which the $j$-th bit of an offspring of $\boldsymbol{x}$ and $\boldsymbol{y}$ is

$$\begin{cases} x_j \text{ if } f(\boldsymbol{x}[\bar{j}]) - f(\boldsymbol{x}) \ < \ f(\boldsymbol{y}[\bar{j}]) - f(\boldsymbol{y}) \\ y_j \text{ otherwise.} \end{cases}$$

where $\boldsymbol{x}[\bar{j}]$ denotes the result of reversing the $j$th bit in $\boldsymbol{x}$, i.e., $(x[\bar{j}])_j = 1 - x_j$, and $(x[\bar{j}])_i = x_i$ for $i \neq j$.

Our experiments used steady-state GAs with roulette wheel reproduction selection, replacing parents by offspring, but preserving the current best solution. In experiments on 200-bit problems with $6 \leq k \leq 9$, and $0 \leq \theta \leq (k-1)$, the number of fitness evaluations required by SX to find the global optimum ranged from 0.5M to 1.9M; increasing the amount of overlap did not increase the computational effort required to find the global optimum

The following table presents some results comparing SX with the Bayesian Optimization Algorithm (BOA) [5] as well as GAs using 1PTX (2PTX and UX exhibit almost identical performance). Results are averages over ten randomly initialized trials; 'M' denotes millions (of fitness evaluations). For the problems corresponding to the last three rows in the table, optimal fitness values are 195, 65, and 48, respectively, reached by SX but not the others; best fitness values obtained are shown in square brackets.

| Problem | Size | 1PTX | BOA | SX |
|---|---|---|---|---|
| $f(3,1)$ | 31 | 0.01M | 0.02M | 0.07M |
| $f(3,1)$ | 61 | 0.04M | 0.05M | 0.2M |
| $f(3,1)$ | 181 | 1.0M | 0.25M | 0.8M |
| $f(6,5)$ | 200 | [143] | [175.5] | 1.4M |
| $f(8,5)$ | 200 | [47.5] | [58.5] | 0.8M |
| $f(9,5)$ | 200 | [34.9] | [43.2] | 1.6M |

[1] Please contact the authors at bsnak@twcny.rr.com or at ckmohan@syr.edu, for details omitted here.

## References

[1] G. Harik *et al.*, *Ev. Comp.* 7(3):231-253, 1999.

[2] C.K. Mohan, Proc. ACM SAC, 374-378, 1998.

[3] C.K. Mohan, Proc. CEC, 1542-1549, 1999.

[4] H. Muhlenbein *et al.*, *Ev. Comp.* 7(4):353-376, 1999.

[5] M. Pelikan *et al. Ev. Comp.*, 2000.