

Errata for mGA1.0: A Common LISP Implementation of a Messy Genetic Algorithm

Kalyanmoy Deb and David E. Goldberg

Department of General Engineering
University of Illinois at Urbana-Champaign
Urbana, IL 61801

July 9, 1991

1 Introduction

The purpose of this report is to identify some of the errors in messy GA code, mGA1.0 (Goldberg and Kerzic, 1990). The code has also been modified to work in Kyoto Common Lisp (KCL).

The following section indicates errors in the code. The latter section describes the modifications necessary to run in KCL.

2 Errors in mGA1.0

Errors are listed by specifying the page number in the report, the name of the function, and the location of the error in the function. The original statement is then typed followed by the corrected version of the statement.

1. Page 36, function `CUT_AND_STACK`, line 15
original version: `(push (cadr mate2) loc_stack)`
corrected version: `(push (car mate2) loc_stack)`
2. Page 36, function `CUT_AND_STACK`, line 17
original version: `(push (pop mate2) loc_stack)))`
corrected version: `(push (cadr mate2) loc_stack)))`
3. Page 37, function `DET_SELECTION`, line 2
original version: `(if (< loopvar init_select_gen)`
corrected version: `(if (<= loopvar init_select_gen)`
4. Page 39, function `MAKE_NEW_POP`, line 20
original version: `(SET_LIST bit_list 0)`
corrected version: move this statement between lines 22 and 23

3 Extension for Kyoto Common Lisp

The mGA1.0 code would work in Kyoto Common Lisp (KCL) with the above changes and with the following modifications.

1. Page 26, file `aux.lisp`
include the following function

```

;=====SET_ARRAY=====
; sets all elements of an array
(defun SET_ARRAY (array num value)
  (do ((i 0 (1+ i)))
      ((>= i num))
    (setf (aref array i) value)))

```
2. Page 29, function `mGA`, line 3
remove line `(setq w:more-processing-global-enable nil)`
3. Page 30, function `mGA`, line 14
remove line `(setq *random-state* (system:random-create-array 71. 35. seed))`
4. Page 30, function `mGA`, line 31
original version: `(gc-immediately :silent t)`
corrected version: `(gbc t)`
5. Page 30, function `PRIMORDIAL`, line 3
original version: `(setf oldpop (copy newpop))`
corrected version: `(setf oldpop (copy-tree newpop))`
6. Page 31, function `JUXTAPOSITIONAL`, line 3
original version: `(setf oldpop (copy newpop))`
corrected version: `(setf oldpop (copy-tree newpop))`
7. Page 31, function `JUXTAPOSITIONAL`, line 7
original version: `(RESET_POP newpop)`
corrected version: `(setf newpop (make-array popsize))`
8. Page 31, function `JUXTAPOSITIONAL`
add statement `(setf (aref newpop count) (make-population-member))`
between lines 23 and 24
9. Page 46, function `RESET_STAT_INFO`, line 13
original version: `(array-initialize vertnum_spot 0)`
corrected version: `(SET_ARRAY vertnum_spot num_subfunctions 0)`

Acknowledgment

Some of the errors were pointed out by Andrew Horner.

Reference

- Goldberg, D. E., and Kerzic, T. (1990). *mGA1.0: A Common Lisp Implementation of a Messy Genetic Algorithm* (TCGA Report No. 90004). Tuscaloosa: University of Alabama, The Clearinghouse for Genetic Algorithms.