

# Parental Population Sizing in Evolutionary Strategies

Tao-Yuan Huang

Institute of Electrical Engineering,  
National Taiwan University,  
Taipei, Taiwan, ROC, 106  
hubert@ipmc.ee.ntu.edu.tw

Yung-Yaw Chen

Institute of Electrical Engineering,  
National Taiwan University,  
Taipei, Taiwan, ROC, 106  
yuchen@ipmc.ee.ntu.edu.tw

**Abstract** – One of the major differences of Evolution Strategies (ES) from other Evolutionary Computation (EC) is self-awareness of the distinction between parental and offspring populations. This feature is all the more prominent when it comes to the availability of two ES variants: COMMA and PLUS ES (i.e.  $(\mu, \lambda)$  and  $(\mu+\lambda)$  ES). Even though these two schemes capture, respectively, the antipodal emphases on exploration and exploitation, conventional practices of ES still suffer from the problem of premature convergence. By the proposition of an adaptive parental population sizing approach, the authors wish to buttress the original ES scheme. This population sizing approach differs from previous ones in that only size of the parental population is adjusted. Simulation on several benchmark problems vindicates the virtue of this modification. Moreover, the modified algorithm also sheds light on convergence status of the population for a given optimization task, hence allowing further improvements on ES toward premature convergence checking and parallel search mechanisms.

## 1. Introduction

Evolution Strategies (ES) were developed by Rechenberg [1] and Schwefel [2] to solve difficult parameter optimization problems. Like *Genetic Algorithms* (GA) [3] and *Evolutionary Programming* (EP) [4], this methodology falls into the generic discipline of *Evolutionary Computation* (EC). EC are biology-inspired techniques most often employed for optimization and search problems. The rationale behind these techniques is fitness-driven random walk after a (global) optimum by an evolving population of possible solutions. Over generations, the population improves in fitness values with search directions and convergence velocity determined by evolution operators. A typical evolution cycle can be summarized into the loop shown in Figure 1 (after the figure given in [5]). As can be observed, recombination (crossover), mutation, and selection are all common evolution operators among different EC branches.

The performance of these population-based algorithms depends greatly on interaction between evolution operators and the optimization problem at hand. The operators' inclination shapes the virtual fitness landscape as seen by the population, while this sampled terrain leads the direction of subsequent search. This complication often results in a

prolonged search process or even premature convergence to some local optimum. Moreover, the interactive nature of these operators also makes it very difficult to hand-tune EC for the wide array of problems. In recognition of this fact, adaptive evolutionary operators are studied and implemented.

Among these researches, the self-adaptive mutation mechanism in ES presents itself as a promising contrivance. It is assumed that strategy parameters for surviving individuals are better than those perish. Under this assumption, the self-adaptive mechanism should be able to moderate the degree of mutation for each individual according to its proximity to the (global) optimum. However, real-world implementations of ES still suffer from slow convergence or even premature convergence due to a lack of coordination among its evolutionary operators. This difficulty results mainly from: (1) blind belief in the spontaneous correspondence from self-adapted mutation step sizes to local fitness terrain, and (2) greediness of the ranking-based selection which erase the trace of interaction between mutation and problem surface. This paper tries to address the second shortcoming by adjusting dynamically the size of the parental population from generation to generation. Not only do simulation results vindicate the virtue of this modified ES, the evolution behavior also demonstrates interesting links between population diversity measures and solution convergence.

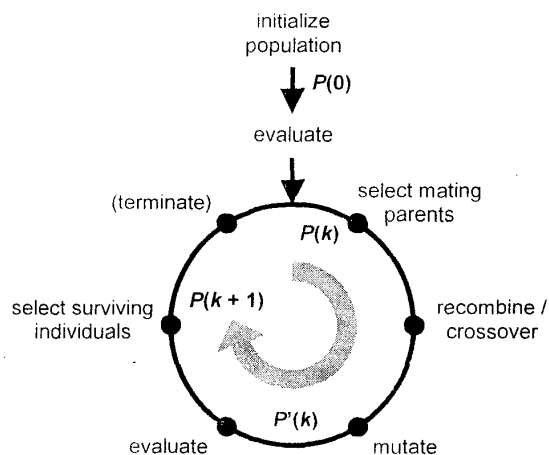


Figure 1. A Generic EC Evolution Cycle (after [5])

## 2. Evolution Strategies

The Evolution Strategies used in this paper are as those described in [6][7]. Each run of ES computation starts with a randomly generated initial population  $P(0) = \{\bar{a}_1(0), \dots, \bar{a}_\mu(0)\}$  of  $\mu$  individuals.

**Definition 1.** *Individual in an ES Population* ( $\bar{a}_j(k)$ )

For standard ES employing  $n$ -dimensional uncorrelated self-adaptive mutation over an  $n$ -dimensional optimization problem, the ES population at the  $k^{\text{th}}$  generation is denoted as

$$P(k) = \{\bar{a}_1(k), \dots, \bar{a}_\mu(k)\} \quad (1)$$

in which the  $j^{\text{th}}$  individual is of the form

$$\bar{a}_j(k) = (\bar{X}_j(k), \bar{\sigma}_j(k)) \quad (2)$$

where

$$\bar{X}_j(k) \in R^n \quad (3)$$

is the vector of object variables in the  $n$ -dimensional search space, and

$$\bar{\sigma}_j(k) \in R_i^n \quad (4)$$

is the vector of strategy parameters.

The object variable portion of an individual signifies the actual position in the real-valued search space for that individual. On the other hand, strategy parameters of an individual are responsible for mutation, and are often referred to as "step sizes".

At the  $k^{\text{th}}$  generation, these  $\mu$  individuals act as parents to produce an intermediate population ( $P'(k)$ ) of  $\lambda$  individuals ( $1 \leq \mu \leq \lambda$ ) through recombination. Usually, the ratio between  $\mu$  and  $\lambda$  is set to 1:7 [24]. When mating parents are chosen to generate offspring, they not only recombine their object variables but also their strategy parameters.

**Definition 2.** *Recombination* ( $r_{\text{RD}}$ )

There are many recombination mechanisms used in ES. The most common ones are *discrete recombination* and *intermediate recombination*. This paper employs *binary discrete recombination*, through which  $\lambda$  offspring are created

$$\bar{a}'_j(k) = r_{\text{RD}}(\bar{a}_s(k), \bar{a}_t(k)); \quad \forall j=1, \dots, \lambda \quad (5)$$

where  $s$  and  $t$  are the indices of two randomly drawn parents. Each component in  $\bar{a}'_j(k)$  (objective variables as well as strategy parameters) is designated as the corresponding component of a chosen parent. This contributing parent is drawn randomly for each component of the offspring individual with replacement.

As mentioned earlier, there are many recombination operators employed by researchers. The algorithm proposed by this paper is applicable with any of these operators.

Once the intermediate offspring population of size  $\lambda$  is obtained, it itself undergoes self-adaptive mutation. A standard implementation of  $n$ -dimensional self-adaptive mutation looks like the following:

**Definition 3.** *Self-adaptive Mutation* ( $m_{\text{SA}}$ )

For each child in the offspring population, self-adaptive mutation is performed so that

$$\bar{a}''_j(k) = m_{\text{SA}}(\bar{a}'_j(k)); \quad \forall j=1, \dots, \lambda \quad (6)$$

where strategy parameters are varied first according to

$$\sigma''_{ij}(k) = \sigma'_{ij}(k) \cdot \exp[\tau \cdot N_j(0, 1) + \tau \cdot N_{ij}(0, 1)]; \quad \forall j=1, \dots, \lambda; i=1, \dots, n \quad (7)$$

which are then used to mutate object variables

$$x''_{ij}(k) = x'_{ij}(k) + \sigma''_{ij}(k) \cdot N_{ij}(0, 1); \quad \forall j=1, \dots, \lambda; i=1, \dots, n \quad (8)$$

In this paper,  $\tau' = (\sqrt{2n})^{-1}$  and  $\tau = (\sqrt{2\sqrt{n}})^{-1}$ , and  $N(0, 1)$  represents a normally distributed random variable with zero expectation and unity variance.

After mutation, the intermediate population  $P'(k)$  will enter the selection pool ( $P''(k)$ ) where a new population of  $\mu$  individuals will be chosen as parents for the next generation ( $P(k+1)$ ). Conventionally, the selection pool can be either the entirety of offspring alone or the union of offspring and parents together [10], i.e.

**Definition 4.** *COMMA ES and PLUS ES* (i.e.  $(\mu, \lambda)$  and  $(\mu + \lambda)$  ES)

In COMMA ES, the selection pool is made up of the entirety of  $\lambda$  offspring alone

$$P''(k) = \{\bar{a}''_1(k), \dots, \bar{a}''_\lambda(k)\} \quad (9)$$

whereas in PLUS ES, the intermediate population consists of the union of  $\mu$  parents and  $\lambda$  offspring

$$P''(k) = \{\bar{a}_1(k), \dots, \bar{a}_\mu(k), \bar{a}''_1(k), \dots, \bar{a}''_\lambda(k)\} \quad (10)$$

The COMMA and PLUS ES are special cases of the more general strategy of  $(\mu, \kappa, \lambda)$  ES, where  $\kappa$  denotes the maximal lifespan of an individual. With  $\kappa = 1$ , the strategy resembles COMMA ES, while with  $\kappa = \infty$ , it resembles PLUS ES [25].

Once pooled, these candidates are subject to the selection operator ( $s_j$ ) so that

$$P(k+1) = s_j(P''(k)) \quad (11)$$

Usually, truncation selection operator based on fitness ranking is used, although tournament selection and other selection operators could be used.

With these definitions, algorithm of the standard ES used in this paper are given as follows:

**Algorithm 1.** *Standard COMMA/PLUS ES*

```

k = 0;
initializing P(0);
evaluating P(0);
while (termination criteria is not fulfilled) do
    recombination:  $\bar{a}'_j(k) = r_{\text{RD}}(\bar{a}_s(k), \bar{a}_t(k)); \quad \forall j=1, \dots, \lambda$ 
    mutation:  $\bar{a}''_j(k) = m_{\text{SA}}(\bar{a}'_j(k)); \quad \forall j=1, \dots, \lambda$ 
    evaluating  $P''(k) = \{\bar{a}'_1(k), \dots, \bar{a}'_\lambda(k)\}$ 
    selection:
    if COMMA ES
         $P''(k) = \{\bar{a}''_1(k), \dots, \bar{a}''_\lambda(k)\};$ 
    else if PLUS ES
         $P''(k) = \{\bar{a}_1(k), \dots, \bar{a}_\mu(k), \bar{a}''_1(k), \dots, \bar{a}''_\lambda(k)\};$ 
    fi

```

```

    P(k+1) = sr(Pn(k));
    k = k + 1;
od

```

### 3. Improvements to Evolution Strategies

#### 3.1 Previous Researches

Because standard ES use deterministic truncation selection, whose selective pressure is high [13], the population tends to search “locally” after few generations [15]. Despite its simplicity and the high convergence rate it encourages, the truncation selection operator ironically undermines ES’ capability to overcome local optima. Once the population is trapped, self-adaptation in mutation step sizes does nothing more than to aggravate the situation. Especially when the ratio between  $\mu$  and  $\lambda$  is very small, there exists such a crowding effect that only top-ranking individuals survive onto the next generation, whose strategy parameters are limited in size and often have great similarity. This phenomenon contributes to the fast convergence rate of COMMA ES.

There have been innumerable attempts to improve global performance of ES. Most of these improvements focus on explicitly replacing conventional crossover, mutation, and selection operators with new ones. In general, these efforts can be classified into two major categories. One attempts to deploy the population over search space with sufficient diversity. The other aims to enhance the efficacy of search guidance. These two approaches correspond to the two main concerns of all EC: “it can be argued that there are only two primary factors in genetic search ... *population diversity* and *selective pressure* ... In a sense, this is just another variation on the idea of *exploration* versus *exploitation* ...” [11]. However, due to the antipodal natures between exploitation and exploration, or between convergence and diversity, few of these improvements can fulfill both criteria simultaneously [12][17][18].

There are two possible workarounds. The first is those algorithms that employ local competition on sub-population level instead of a global competition [16][19], because search diversity in these cases is upheld by spatially distinct sub-populations each evolving to local minima nearby. These modified algorithms often require a large population that will inevitably increase computation load. Moreover, the connection between global fitness landscape seen by the whole population and its collective mutation step sizes has been cut off. In fact, it can be asserted that these modified algorithms are equivalent to parallel search exercised with these sub-populations [20]. The other solution, which the authors try to exemplify in this paper, lies in altering the behavior of selection by making it more intelligent, more informed about the *status quo* of search process.

#### 3.2 Parental Population Sizing

As mentioned earlier, one of the weaknesses in ES lies with the high selective pressure. Taken the fact that knowledge about fitness surface is encoded implicitly within spatial and fitness distribution of the population as well as distribution

of mutation step sizes, much of these hints will definitely be eliminated under a high selective pressure. In this aspect, sustenance of population diversity is much more important than the creation of it through mutation. With the idea in mind, many studies have gone straight ahead to increase the size of the entire search population in order to host more diversity in one generation. However, sustenance of diversity lies not with sheer number of individuals. Having a large amount of similar individuals due to high selective pressures is worse than having only one in the population, because in the former case futile recombination and mutation are prone to happen.

Selective pressure in ES is determined by  $\mu$  and  $\lambda$ , because it is a function of the ratio between size of the to-be-selected population (parents for the next generation) and size of the whole selection pool (for truncation selection operator) [13]. This ratio  $T$  is inversely and nonlinearly proportional to selective pressure. In COMMA ES,  $T$  is  $(\mu / \lambda)$ ; in PLUS ES,  $T$  is  $(\mu / (\mu + \lambda))$ . For many researchers, it has been a conventional practice to use the recommendatory  $\mu$ - $\lambda$  ratio of 1:7 [6]. According to this suggestion, the selective pressure is high and premature convergence is almost inevitable.

The manipulation of selective pressure can be achieved through varying  $\mu$  and  $\lambda$ . Nevertheless, absolute value of  $\lambda$  also determines the computation load in terms of function evaluation required. Consequently, changing the value of  $\mu$  seems to be a reasonable and efficient approach. Therefore, this paper proposes a modified ES algorithm that dynamically adjusts selective pressure through *parental population sizing*, in accordance with population diversity measures at each generation.

### 4. Effects of Parental Population Sizes on ES

To show the effects of  $\mu$ - $\lambda$  ratio has on ES, especially when  $\mu$  varies over a fixed  $\lambda$ , several simulations are conducted over six benchmark functions.

#### 4.1 ES to be Compared

The ES sets used in simulation are denoted by  $\mu$ - $\lambda$  COMMA/PLUS ES according to the actual number of individuals involved and the pooling scheme used. The sets employed are:

- Set 1.** 30-200 COMMA ES (30-200-CES), and  
100-200 COMMA ES (100-200-CES), and  
180-200 COMMA ES (180-200-CES).
- Set 2.** 30-200 PLUS ES (30-200-PES), and  
100-200 PLUS ES (100-200-PES), and  
180-200 PLUS ES (180-200-PES).

Other algorithm parameters are listed in Table 1.

#### 4.2 Benchmark Functions

Task objectives in the simulation are function optimization, with fitness values obtained directly from the objective

**Table 1. Parameters Used for Simulation**

$\mu$	= as denoted.	(number of parents)
$\lambda$	= as denoted.	(number of offspring)
$\sigma^j_i(0)$	= 3.	(initial step size)
$k_{max}$	= 2000.	(number of generations, except 1000 for $f_6$ ).

functions. The benchmark functions are listed in Table 2 and 3 [14][12][21]. A few comments are worth mentioning beforehand. As can be observed in the tables,  $f_1$  is the unimodal sphere minimization problem, while  $f_2$  and  $f_3$  are two different versions of the multimodal Ackley optimization problem with different local minimum basin widths and depths. In fact, these problems represent varying degrees of multimodality. For instance,  $f_4$  is the Griewank function with the largest search space among the six benchmark problems. Finally,  $f_5$  represents multimodal functions with many large local minima, whereas  $f_6$  from the Shekel's family of functions has only ten local minima.

**4.3 Observation on Simulation Results**

Simulation results for Set 1 & 2 are tabulated in Table 4 & 5 respectively. Each simulation case has been conducted 50 times independently. After ranking individual results, the two top-most and two worst cases are discarded. The tabulated values are thus obtained from the central 46 cases. Best performing ES in each case is encircled with bold borders.

As can be seen from the results, optimal ratio between  $\mu$  and  $\lambda$  is indeed problem dependent. Nevertheless, both

COMMA and PLUS ES (Table 4 and 5) show a general preference for larger  $\mu$ - $\lambda$  ratios as the complexity (multimodality) of problems increases. Also noteworthy is the divergent inclination of "180-200-CES" in Table 4. It is so because its large  $\mu$ - $\lambda$  ratio produces such a low selective pressure that the population refuses to converge.

**5. Dynamic Parental Population Sizing**

From the previous results, it can be concluded that conventional  $\mu$ - $\lambda$  ratio is not optimal for all problems. However, optimality of a given ratio is problem dependent. It seems that  $\mu$ - $\lambda$  ratio is indeed an important factor determining the performance of ES.

With these affirming results, the authors wish to pursue the idea of tuning  $\mu$ - $\lambda$  ratio dynamically in the belief that (1) a excessively large  $\mu$  value may still increase computation overhead, and (2) an intelligent selection operator can reach a reasonable  $\mu$ - $\lambda$  ratio to maintain the balance between performance and algorithm resources.

The basic idea of dynamic parental population sizing is: when diversity of the population is high, lessen the selective pressure so that this diversity can be passed intact onto the next generation; on the other hand, when the diversity is low, choose only the topmost parent to save resources, and let self-adaptive mutation does its job. Here, diversity of the population is based on fitness only, since knowledge of the fitness terrain possessed by the ES is reflected in the fitness distribution of its population only. Therefore, several fitness

**Table 2. Benchmark Functions Used in Simulation** (with  $n$  indicating dimension of the functions: i.e.  $1 \leq i \leq n$ )

Function	n	Range
$f_1(X) = \sum_i x_i^2$	30	[-15, 15] <sup>n</sup>
$f_2(X) = (-20) \exp((-0.2) \sqrt{\frac{1}{n} \sum_i x_i^2}) - \exp(\frac{1}{n} \sum_i \cos(2\pi x_i)) + offset_2$	30	[-15, 15] <sup>n</sup>
$f_3(X) = (-10) \exp((-0.05) \sqrt{\frac{1}{n} \sum_i x_i^2}) - \exp(\frac{1}{n} \sum_i 1.3 \cos(0.5\pi x_i)) + 20 + offset_3$	15	[-15, 15] <sup>n</sup>
$f_4(X) = \frac{1}{4000} \sum_i x_i^2 - \prod_i \cos(x_i/4) + offset_4$	30	[-600, 600] <sup>n</sup>
$f_5(X) = 0.02 \times \sum_i (x_i - 2)^2 + 3.5 \sin((\sum_i x_i) - 5) + 2.5 \sin(25 \cdot (\sum_i x_i - 2.3 \cdot (\sum_i x_i) + 2.7)) + 1.5 \cos(10 \cdot (\sum_i x_i) - 7.5) + offset_5$	15	[-15, 15] <sup>n</sup>
$f_6(X) = -\sum_i [(X - \bar{a}_i)(X - \bar{a}_i)^c + c_i]^{-1}$ ( $1 \leq j \leq 10$ , values of constants are given in Table 3)	4	[0, 10] <sup>n</sup>

**Table 3. Coefficient Constants Used in  $f_6(X)$ :**

j	1	2	3	4	5	6	7	8	9	10	
$\bar{a}_i$	$i=1$	4	1	8	6	3	2	5	8	6	7
	$i=2$	4	1	8	6	7	9	5	1	2	3.6
	$i=3$	4	1	8	6	3	2	3	8	6	7
	$i=4$	4	1	8	6	7	9	3	1	2	3.6
$c_i$	0.1	0.2	0.2	0.4	0.4	0.6	0.3	0.7	0.5	0.5	

**Table 4. Comparison among Results (COMMA ES)**

(averaged over the central 46 out of 50 runs; "Mean Best" indicating mean best fitness value in the last generation; "STD of Best" denoting standard deviation among the best fitness values; top-ranking cases encircled by bold borders)

Function	30-200-CES		100-200-CES		180-200-CES	
	Mean Best	STD of Best	Mean Best	STD of Best	Mean Best	STD of Best
$f_1$	<b>1.597×10<sup>-27</sup></b>	<b>1.110×10<sup>-26</sup></b>	3.486×10 <sup>-18</sup>	2.330×10 <sup>-17</sup>	1053.026	1135.161
$f_2$	0.076	0.254	1.756×10 <sup>-7</sup>	2.182×10 <sup>-8</sup>	20.219	0.116
$f_3$	0.260	0.264	3.324×10 <sup>-7</sup>	4.347×10 <sup>-11</sup>	4.114	4.120
$f_4$	0.026	0.033	0.001	0.004	1.998	0.944
$f_5$	0.473	0.158	0.315	0.101	1.060	0.452
$f_6$	-9.258	2.800	-8.903	2.920	-6.252	3.464

diversity measures have to be defined first before going onto the modified algorithm.

**Definition 5.** *Best, Average, Worst Population Fitness* ( $F_B, F_A, F_W$ ):

Given a population  $P(k) = \{\bar{a}_1(k), \dots, \bar{a}_\mu(k)\}$  of  $\mu$  individuals with known fitness values  $\{\phi_1(k), \dots, \phi_\mu(k)\} = \{\Phi(\bar{X}_1(k)), \dots, \Phi(\bar{X}_\mu(k))\}$  at the  $k^{\text{th}}$  generation over a *minimization* task, where  $\Phi(\cdot)$  is the fitness function, the *Average Population Fitness* is computed by

$$F_A(k) = \frac{1}{\mu} \sum_i \phi_i(k) \quad (12)$$

and the *Best* and *Worst Population Fitness* are defined respectively as

$$F_B(k) = \min_j \phi_j(k) \quad (13)$$

and

$$F_W(k) = \max_j \phi_j(k) \quad (14)$$

**Definition 6.** *Fitness Concentration ( $F_C$ ):*

With the same population given above, the measure of *Fitness Concentration* is defined to be

$$F_C(k) = \frac{F_W(k) - F_A(k)}{F_W(k) - F_B(k)} \quad (15)$$

however, if  $F_W(k)$  equals  $F_B(k)$ ,  $F_C(k)$  is defined as unity. Physical meaning of this measure is self-evident. When fitness distribution of the population is indistinctive, its average fitness (i.e.  $F_A(k)$ ) should be close to the worst population fitness ( $F_W(k)$ ), and a low fitness concentration will be obtained. Contrarily, when most individuals of the population possess good fitness, a high fitness concentration is obtained. Moreover, *given a unimodal landscape*, a high fitness concentration value signifies that most individuals are all very close to the global optimum, whereas a low fitness concentration value means that only a few individuals make it to the optimum.

Based on this measure, a relation between the actual fitness landscape and fitness distribution of the population could be established. That is, a high  $F_C$  suggests that fitness

**Table 5. Comparison among Results (PLUS ES)**

(averaged over the central 46 out of 50 runs; "Mean Best" indicating mean best fitness value in the last generation; "STD of Best" denoting standard deviation among the best fitness values; top-ranking cases encircled by bold borders)

Function	30-200-PES		100-200-PES		180-200-PES	
	Mean Best	STD of Best	Mean Best	STD of Best	Mean Best	STD of Best
$f_1$	1.664×10 <sup>-18</sup>	1.399×10 <sup>-17</sup>	3.069×10 <sup>-18</sup>	1.490×10 <sup>-17</sup>	3.413×10 <sup>-12</sup>	1.411×10 <sup>-11</sup>
$f_2$	0.457	0.623	4.229×10 <sup>-7</sup>	9.984×10 <sup>-7</sup>	6.117×10 <sup>-7</sup>	1.323×10 <sup>-6</sup>
$f_3$	0.538	0.368	3.324×10 <sup>-7</sup>	0.000	3.324×10 <sup>-7</sup>	0.000
$f_4$	0.079	0.152	0.003	0.005	0.000	0.000
$f_5$	0.128	0.047	0.161	0.033	0.199	0.041
$f_6$	-10.099	1.655	-10.536	7.998×10 <sup>-10</sup>	-10.536	7.577×10 <sup>-10</sup>

distribution can somehow reflect the actual landscape, since fitness values of most individuals are concentrated. On the other hand, when  $F_C$  is low, fitness landscape as seen by the population may be very different from the actual one, since representative tendency could not be established.

It is known from the previous simulations that when selective pressure of this scheme is lowered, selective variance increases and divergence in COMMA ES will occur. This is because the truncation selection fraction  $T$  is always smaller or equal to 0.5 in PLUS ES, while for COMMA ES, inappropriate ratio between  $\mu$  and  $\lambda$  will raise the fraction above 0.5 and bring about the inclination to diverge.

Therefore, an extra lower bound ( $p_m(k)$ ) for the number of selection-pooling parents is devised. By this lower bound, there should be at least such many parents entering the selection pool to compete with the offspring. The modification means to make the truncation fraction  $T$  less or equal to 0.5, so that a reasonable degree of diversity can be remained without the threat of divergence [13]. Hence this modified COMMA ES now turns into somewhat PLUS-like. This minimal number,  $p_m$ , is defined as

**Definition 7.** *Minimal Pooling Parent Number ( $p_m(k)$ )*

For a given parental population size  $\mu$ ,

$$p_m(k) = \max((2\lfloor \mu(k) - \lambda \rfloor), 1) \quad (16)$$

$$\text{so that } T = \frac{\mu(k)}{(p_m(k) + \lambda)} \leq 0.5.$$

### 5.1 Modified ES Algorithm

According to the aforementioned rationale, the modified ES algorithm in terms of measures introduced is as follows:

**Algorithm 11.** *Modified ES with Dynamic Parental Population Sizing* (for minimization)

$k = 0$ ;

initializing  $P(0)$ ;

evaluating  $P(0)$ ;

**while** (termination criteria is not fulfilled) **do**

*recombination:*  $\bar{a}'_j(k) = r_{\text{bw}}(\bar{a}_i(k), \bar{a}_r(k))$ ;

$\forall j = 1, \dots, \lambda$

**Table 6. Comparison among Results (Modified ES)**

(averaged over the central 46 out of 50 runs; "Mean Best" indicating mean best fitness value in the last generation; "STD of Best" denoting standard deviation among the best fitness values; top-ranking cases are encircled by bold borders)

Function	Best of Set 1		Modified COMMA ES		Best of Set 2		Modified PLUS ES	
	Mean Best	STD of Best	Mean Best	STD of Best	Mean Best	STD of Best	Mean Best	STD of Best
$f_1$	1.597×10 <sup>-27</sup>	1.140×10 <sup>-26</sup>	2.525×10 <sup>-47</sup>	1.597×10 <sup>-46</sup>	1.664×10 <sup>-18</sup>	1.399×10 <sup>-17</sup>	9.201×10 <sup>-21</sup>	3.153×10 <sup>-20</sup>
$f_2$	1.756×10 <sup>-7</sup>	2.182×10 <sup>-8</sup>	1.715×10 <sup>-7</sup>	6.286×10 <sup>-14</sup>	4.229×10 <sup>-7</sup>	9.984×10 <sup>-7</sup>	1.907×10 <sup>-7</sup>	1.053×10 <sup>-7</sup>
$f_3$	3.324×10 <sup>-7</sup>	4.347×10 <sup>-13</sup>	3.324×10 <sup>-7</sup>	1.407×10 <sup>-15</sup>	3.324×10 <sup>-7</sup>	0.000	3.324×10 <sup>-7</sup>	1.692×10 <sup>-15</sup>
$f_4$	0.001	0.004	0.000	0.002	0.000	0.000	0.000	0.002
$f_5$	0.315	0.101	0.063	0.033	0.128	0.047	0.114	0.032
$f_6$	-9.258	2.800	-10.536	8.460×10 <sup>-16</sup>	-10.536	7.577×10 <sup>-16</sup>	-10.536	5.005×10 <sup>-16</sup>

mutation:  $\bar{a}''_j(k) = m_{s_A}(\bar{a}'_j(k)); \forall j = 1, \dots, \lambda$

evaluating  $P''(k) = \{\bar{a}''_1(k), \dots, \bar{a}''_z(k)\}$

calculating Fitness Concentration ( $F_c(k)$ ) of the parental population  $P(k)$

calculating the new parental population size  $\mu(k+1)$  by

$$\mu(k+1) = (\lambda - 1)(1 - F_c(k)) + 1 \quad (17)$$

selection:

if COMMA ES

calculating  $p_m(k+1)$  from Eq. (16)

$$P''(k) = \{\bar{a}_1(k), \dots, \bar{a}_{p_m(k+1)}(k), \bar{a}''_1(k), \dots, \bar{a}''_z(k)\}$$

else if PLUS ES

$$P''(k) = \{\bar{a}_1(k), \dots, \bar{a}_\mu(k), \bar{a}''_1(k), \dots, \bar{a}''_z(k)\}$$

fi

$$P(k+1) = s_r(P''(k));$$

$k = k + 1;$

od

### 5.2 Performance of the Modified ES

Performance of the modified ES in the same suite of benchmark functions is tabulated in Table 6. If compared separately, it is obvious that the modified COMMA ES performs better than the original Set 1 ES in all problems. Also, the modified PLUS ES performs better than Set 2 ES in four out of six problems. Finally, when modified COMMA and PLUS ES are compared, it is found out that the former still performs better. It seems that due to the partial elitism of the modified COMMA ES and the parental population sizing scheme, the downsides of standard COMMA ES are all removed. Examples of the evolution of parental population size across the generations are given in Figure 2.

## 6. Conclusions

In this paper, the authors propose a modified algorithm to calculate parental population size for ES. The idea comes

from the observation that conventional  $\mu/\lambda$  ratio used in ES produces an excessively high selective pressure in the truncation selection operator for population diversity to sustain. Therefore, premature convergence occurs under the disguise of a fast convergence velocity. This is also why ES have been called by many as a local optimization technique. Some reasoning on the tension between diversity and selective pressure in ES can be found in [22][23].

It is concluded by the authors that high selective pressure of ES ranking-based selection operator hinders free evolution of self-adaptation by admitting only a small portion of the best of offspring to survive as parents in the next generation. These new parents often have similar strategy parameters in addition to similar object variables. Moreover, since unsupervised self-adaptation of strategy parameters fails to readjust mutation steps in accordance with current search status, mutated individuals have difficulties in overcoming local optima. Consequently, when these two deficiencies add up, a vicious circle emerges: lack of diversity in mutation steps leads to homogeneity in object variables, which in turn aggravates the irrelevancy of mutation operator.

To solve this deficiency, the mechanism of parental population sizing is proposed. The mechanism feeds on diversity measures of the parental population and computes the number of parents to be selected for the next generation, adjusting selective pressure dynamically. By varying the number of parents over a fixed offspring population size, diversity in mutation steps is preserved and deficiency repaired. What is more, since only the size of parental population is altered instead of that of offspring population, there should be less computation overhead and memory requirement than conventional offspring population sizing techniques. To sum up, the modification proposed in this paper implies more efficient and relevant usage of population resources.

Moreover, a phenomenon is found that parental population sizes in some cases fall down to the lowest level of one and stays there for the rest of simulation. This interesting point suggests (1) a convergent state is reached and external verification mechanism should step in, or (2)

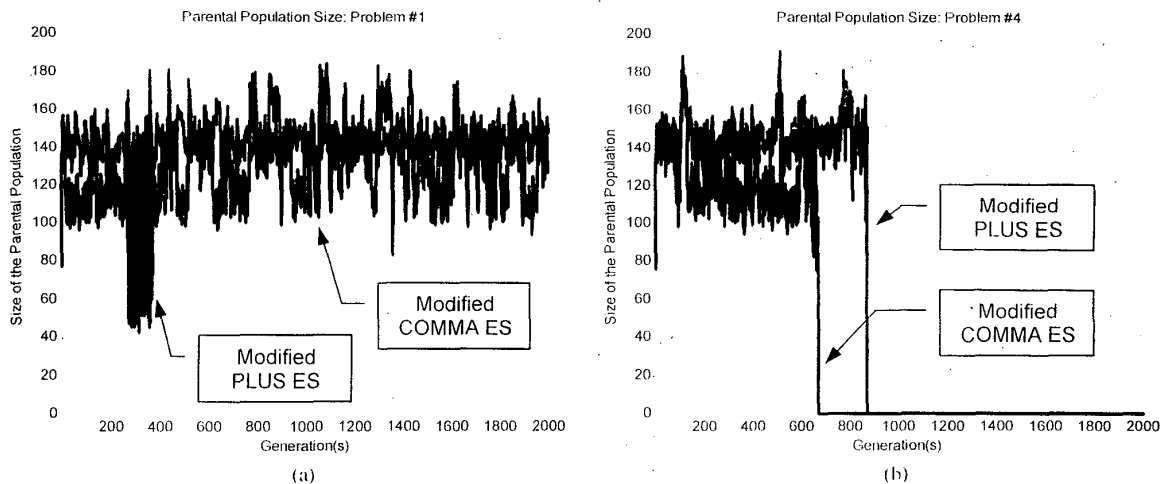


Figure 2. Exemplary Figure of Parental Population Size (averaged over the central 46 runs: 2.a-- for  $f_1$  and 2.b-- for  $f_4$ )

further modification of the parental population sizing approach (for example, the lowest level of parental population number should be two rather than one). At any rate, simulation on several benchmark problems vindicates the virtue of this population sizing approach.

## 7. References

- [1] I. Rechenberg, *Evolutionstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Stuttgart, Germany: Frommann-Holzboog, 1973.
- [2] H.-P. Schwefel, *Evolutionstrategie und numerische Optimierung*. Dissertation, Technische Universität Berlin, Germany, May 1975.
- [3] J. H. Holland, "Outline for a Logical Theory of Adaptive Systems", *J. Assoc. Comput. Mach.*, vol. 3, pp. 297-314, 1962.
- [4] L. J. Fogel, "Autonomous Automata", *Ind. Res.*, vol. 4, pp. 14-19, 1962.
- [5] H.-P. Schwefel and F. Kursawe, "On Natural Life's Tricks to Survive and Evolve", *Proceedings of The 1998 IEEE International Conference on Evolutionary Computation*, pp. 1-8, 1998.
- [6] T. Bäck, U. Hammel, and H.-P. Schwefel, "Evolutionary Computation: Comments on the History and Current State", *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 3-17, Apr. 1997.
- [7] D. B. Fogel, "An Introduction to Simulated Evolutionary Optimization", *IEEE Trans. on Neural Networks*, Vol. 5, No. 1, pp. 3-14, 1994.
- [8] T. Bäck, "On the Behavior of Evolutionary Computation in Dynamic Environments", *Proc. 1998 IEEE World Congress on Computational Intelligence*, pp. 446-451, 1998.
- [9] J. Born, *Evolutionstrategien zur Numerischen Lösung von Adaptationsaufgaben*. Dissertation A, Humboldt-Universität, Berlin, 1978.
- [10] Jörg Heitkötter and David Beasley, eds., "The Hitch-Hiker's Guide to Evolutionary Computation: A list of Frequently Asked Questions (FAQ)", USENET: comp.ai.genetic. Available via anonymous FTP from rtfm.mit.edu/pub/usenet/news.answers/ai-faq/genetic/, 1999.
- [11] D. Whitley, "The GENITOR Algorithm and Selection Pressure: Why Rank-Based Allocation of Reproductive Trials is Best", *Proceedings of the Third International Conference on Genetic Algorithms*, San Mateo: Morgan Kaufmann, pp.116-121, 1989.
- [12] X. Yao and Y. Liu, "Evolutionary Programming Made Faster", *IEEE Trans. Evolutionary Computation*, vol. 3, no. 2, July 1999.
- [13] T. Blicke and L. Thiele, "A Comparison of Selection Scheme Used in Evolutionary Computation", *Evolutionary Computation*, vol. 4, no. 4, p.p. 361-394, 1997.
- [14] T. Bäck and H.-P. Schwefel, "An Overview of Evolutionary Computation for Parameter Optimization", *Evolutionary Computation*, vol. 1, no. 1, pp. 1-23, 1993.
- [15] H.-G. Beyer, "Toward a Theory of Evolution Strategies: Self-Adaptation", *Evolutionary Computation*, vol. 3, no. 3, pp. 311-347, 1996.
- [16] M. Gorges-Schleuter, "A Comparative Study of Global and Local Selection in Evolution Strategies", *PPSN V*, pp. 367-377, 1998.
- [17] I. Rechenberg, *Evolutionstrategie '94*, Stuttgart: Frommann-Holzboog, 1994.
- [18] A. Ostermedier, A. Gawelczyk, and N. Hansen, "A Derandomized Approach to Self-Adaptation of

- Evolution Strategies", *Evolutionary Computation*, vol. 2, no. 4, pp. 369-380, 1995.
- [19] J.-M. Yang, Y.-P. Chen, J.-T. Horng, and C.-Y. Kao, "Applying Family Competition to Evolution Strategies for Constrained Optimization", *Lecture Notes In Computer Science*, Issue 1213, pp. 201-211, 1997.
- [20] H. Mühlenbein, "Asynchronous Parallel Search by the Parallel Genetic Algorithm", *Proceedings of the Third IEEE Symposium on Parallel and Distributed Processing*, pp. 526-533, 1991.
- [21] Olivier François, "An Evolutionary Strategy for Global Minimization and Its Markov Chain Analysis", *IEEE Transactions on Evolutionary Computation*, Vol. 2, No. 3, pp. 77-90, 1998.
- [22] Hans-Georg Beyer, "Toward a Theory of Evolution Strategies: On the Benefits of Sex -- the  $(\mu/\mu, \lambda)$  Theory", *Evolutionary Computation*, Vol.3 No. 1, pp. 81-111, 1996.
- [23] Hans-Georg Beyer, "Mutate Large, But Inherit Small! On the Analysis of Rescaled Mutations in  $(1, \lambda)$ -ES with Noisy Fitness Data", *PPSN V, LNCS no. 1498*, pp. 109-118, 1998.
- [24] H.-P. Schwefel, "Collective Phenomena in Evolutionary Systems", *Preprints of the 31st Annual Meeting of the International Society for General System Research*, Budapest, 2, pp. 1025-1033, 1987.
- [25] H.-P. Schwefel and Günter Rudolph, "Contemporary Evolution Strategies", *Advances in Artificial Life, LNAI 929*, pp. 893-907, 1995.