

NEW VARIANTS OF THE DIFFERENTIAL EVOLUTION ALGORITHM: APPLICATION FOR NEUROSCIENTISTS

L. Buhry, A. Giremus, E. Grivel, S. Saïghi and S. Renaud

IMS Laboratory, UMR 5218 CNRS
University of Bordeaux – ENSEIRB
351, Cours de la Libération 33405 TALENCE – FRANCE
Email: laure.buhry, audrey.giremus, eric.grivel, sylvain.saïghi, sylvie.renaud@ims-bordeaux.fr

ABSTRACT

When dealing with non-linear estimation issues, metaheuristics are often used. In addition to genetic algorithms (GAs), simulating annealing (SA), etc., a great deal of interest has been paid to differential evolution (DE). Although this algorithm requires less iterations than GAs or SA to solve optimization issues, its computational cost can still be reduced. Variants have been proposed but they do not necessarily converge to the global minimum. In this paper, our contribution is twofold: 1) we present new variants of DE. They have the advantage of converging faster than the standard DE algorithm while being robust to local minima. 2) To confirm the efficiency of our variants, we test them with a benchmark of functions often considered when studying metaheuristic performance. Then, we use them in the field of neurosciences to estimate the parameters of the Hodgkin–Huxley neuronal activity model.

1. INTRODUCTION

In the field of signal processing, several approaches operate in two steps: system modelling and model parameter estimation. The estimation step consists in searching for the set of parameters which minimizes a given error criterion such as the least square error, the maximum likelihood or the minimum variance. Depending on the constraints of the application (real-time and storage capacity), different off-line/on-line or iterative algorithms can be considered. Thus, one can use subspace methods, adaptive or optimal filtering, or the expectation-maximisation (EM) algorithm, etc. Their optimization steps may be based on classical techniques such as the steepest descent or the Newton Raphson methods. Nevertheless, when dealing with highly non-linear estimation issues, alternative solutions, such as metaheuristics, must be considered. They may be potentially useful, especially when locating the global optimum is a difficult task. Although their computational costs are quite high, they are well suited to optimization problems.

These optimization techniques are inspired by natural systems like metallurgy for the simulated annealing (SA), biology of evolution for the genetic algorithms (GAs) or ethology for the ant colony algorithms or the particle swarm optimization. Thus, they have already been used in a large number of application areas, such as biomedical applications to estimate intensity distributions for brain magnetic resonance images [11], Rayleigh fading channel simulation [5], in the field of multimedia [1], etc.

In this paper, we focus our attention on the differential evolution (DE) algorithm. Invented by Price and Storn in 1995 [12], DE belongs to the class of evolutionary algorithms. Like GAs, DE consists of a population of individuals which evolves towards a parameter vector that minimizes a beforehand defined fitness function F_{fit} . The purpose of DE is then to explore and to evaluate new regions of the solution space, by building new candidate solutions from existing ones. It uses mechanisms inspired by biological evolution, namely the reproduction, the recombination, and the selection. However, unlike GAs, the exploration is automatically regulated since new individuals result only from the recombination of individuals of the initial population. In addition, instead of keeping

the best individuals of the k^{th} generation¹, DE consists in comparing pairwise one individual of a generation with its mutant. Thus, in DE, every candidate of the new generation has a fitness function F_{fit} that is lower (or equal) to the fitness function of the previous one.

Our contribution is twofold: firstly, we suggest a new variant of the DE in order to improve its convergence speed, hence its computational cost, while avoiding local minima. Secondly, we evaluate the resulting approach in the field of neurosciences. Indeed, in this area, signal processing plays a role that is more and more active. Thus, we have already studied the relevance of GAs to estimate the parameters of the neuronal activity model proposed by Hodgkin–Huxley [6] and widely used in neurosciences [2].

The remainder of this paper is organized as follows. In the second part, we present the classical DE strategies. Then, in the third part, we describe the proposed variants and compare the performance of all the strategies on the benchmark of functions introduced by De Jong in [7] and widely used in the optimization field. Finally, in the fourth part, we apply and test our variant on a neuro-scientific problem: the parameter estimation of the Hodgkin–Huxley model.

2. CLASSICAL STRATEGIES OF THE DIFFERENTIAL EVOLUTION ALGORITHM

2.1 Classical DE

DE consists in generating a population of NP^2 individuals which are composed of D parameters, also called “genes”. The population is initialized by randomly choosing individuals in a uniform manner within the boundary constraints of the model. Then, at each time step, new trial individuals are built by means of two operations: the so-called “differentiation” and the “recombination”. In the following, we define $X_k^r(i)$ as the i^{th} gene of the r^{th} individual of the k^{th} generation.

Differentiation: the r^{th} new parameter vector, $X_{k,trial}^r$, is generated by adding to an individual $X_k^{r_1}$ randomly chosen among the k^{th} generation in a uniform manner, the weighted difference between two other population members, $X_k^{r_2}$ and $X_k^{r_3}$, with $r_1 \neq r_2 \neq r_3$:

$$\forall r = 1, \dots, NP, \quad X_{k,trial}^r = X_k^{r_1} + F \cdot (X_k^{r_2} - X_k^{r_3}), \quad (1)$$

where F is usually set to 0.5.

Recombination: the r^{th} mutant individual, $X_{k,mu}^r$, inherits genes of $X_{k,trial}^r$ with a probability CR , where $CR \in [0, 1]$ is usually set to 0.9. By generating u according to a uniform distribution over $[0, 1]$ one has:

$$\forall i = 1, \dots, D, \forall r = 1, \dots, NP, \quad X_{k,mu}^r(i) = \begin{cases} X_{k,trial}^r(i) & \text{if } u < CR \\ X_k^r(i) & \text{otherwise.} \end{cases}$$

¹ k^{th} “generation” denotes here the population at the iteration k .

²For the sake of clarity, we use the same notations as in the seminal paper [12].

Finally, a selection is carried out by comparing the fitness function values of $X_{k,mult}^r$ and of X_k^r , respectively, as follows:

$$X_{k+1}^r = \begin{cases} X_{k,mult}^r & \text{if } F_{fit}(X_{k,mult}^r) \leq F_{fit}(X_k^r) \\ X_k^r & \text{otherwise} \end{cases}$$

As for terminal conditions, one can either set the number of iterations N_{iter} or define an upper bound on the fitness function value.

2.2 Other existing strategies

Other strategies have been proposed for the last years [4]. They aim at speeding up the convergence by better choosing the differentiation direction $X_k^{r2} - X_k^{r3}$ and the search region. Among them, an approach known as the *RAND/BEST* has been shown to improve the convergence speed. It operates like the classical DE, except that the trial individual is generated as follows:

$$\forall r = 1, \dots, NP, \quad X_{k,trial}^r = X_k^{\min} + F \cdot (X_k^{r2} - X_k^{r3}) \quad (2)$$

where X_k^{r2} and X_k^{r3} are drawn randomly and uniformly among the population and X_k^{\min} denotes the individual of the k^{th} generation that minimizes the fitness function F_{fit} . According to [4], "it looks like a chaotic local search around the current best solution".

Another strategy consists in using the best individual and combining it with 3 other randomly chosen individuals – instead of using only 2. Here, this method is called the *RAND/MIN*. Thus, the r^{th} trial individual, $X_{k,trial}^r$, is generated as follows:

$$\forall r = 1, \dots, NP, \quad X_{k,trial}^r = X_k^{r1} + F \cdot (X_k^{r1} - X_k^{\min}) + F \cdot (X_k^{r2} - X_k^{r3}) \quad (3)$$

In this case, the differentiation direction is $X_k^{r1} - X_k^{\min} + X_k^{r2} - X_k^{r3}$. Although the computational costs of these methods are not greater than the complexity of the standard DE, the candidate that can be found may correspond to a local minimum. Indeed, by always using the best individual of the current population in the differentiation step, the region of the global minimum may never be explored.

3. NEW VARIANT OF THE DIFFERENTIAL EVOLUTION

3.1 Description

Although the above-mentioned variants of the DE improve convergence, they increase the chance to be trapped in a local minimum. Indeed, they tend to limit the exploration nearby the best individual of each generation. To overcome this difficulty, we propose to favor all the individuals yielding a high value of the fitness function as opposed to only considering the best one. For that purpose, at the differentiation step, we suggest drawing the individuals with a probability that is inversely proportional to the value of the fitness function for this individual. In this way, no computational load is devoted to exploring "uninteresting" regions of the solution space while diversity among the population is maintained. Let p_k^r denote the probability for individual X_k^r to be selected. We first consider the following definition for this probability:

$$p_k^r = K \cdot \exp(-F_{fit}(X_k^r)), \quad (4)$$

where K is a normalization constant, which is adjusted so that the sum of the elementary probabilities equals 1 and that these probabilities define a multinomial distribution over the set of individuals. However, if the fitness function takes high values, all the probabilities p_k^r fall in the tail of the exponential function and have similar values. Therefore, the proposed approach is not discriminant enough. As an alternative to definition (4), we propose to assign the following probability to individual X_k^r :

$$p_k^r = K' \cdot \exp\left(\frac{-\alpha \cdot (F_{fit}(X_k^r) - F_{fit}(X_k^{\min}))}{F_{fit}(X_k^{\max}) - F_{fit}(X_k^{\min})}\right). \quad (5)$$

where K' is the normalisation constant and X_k^{\max} is the individual of the k^{th} generation that yields the maximal value of the fitness function. This formula deserves some commentaries.

- 1) By subtracting $F_{fit}(X_k^{\min})$ and dividing by $F_{fit}(X_k^{\max}) - F_{fit}(X_k^{\min})$, we make sure that the argument of the exponential function lies in the interval $[-1, 0]$. This choice leads to probability values evenly spread in the interval $[0, 1]$.
- 2) The factor α can be adjusted by the practitioner. For small values of α , the resulting multinomial distribution tends to become uniform. On the contrary, if a too high value of α is considered, the distribution tends to the dirac delta measure centered in X_k^{\min} . In this case, only the best individual has a chance to be selected.
- 3) It should be noted that this definition bears similarity to the acceptance probability of the SA algorithm.

Our approach is expected to speed up convergence in the presence of local minima. More precisely, the *RAND/MIN* and *RAND/BEST* variants tend to explore only the region nearby the best individual which may be a local minimum. They still have a chance to escape thanks to the random generation of the search direction. However, it may take a lot of iterations. On the contrary, our technique can be seen as a parallel exploration of the regions which are the most likely to comprise the solution to the optimization problem. In this way, the global minimum is reached in less iterations.

In the next section, we compare 6 DE strategies: the classical DE, the *RAND/BEST*, the *RAND/MIN* and 3 proposed variants. 1) The *DEvariant* consists in simulating individuals X_k^{ri} , for $i = 1, 2, 3$, in equation (1), according to the multinomial distribution described above. 2) In the variant of the *RAND/BEST* denoted as *RAND/BESTvariant*, we replace X_k^{\min} , in equation (2), by an individual drawn according to the multinomial distribution whereas the two other individuals are uniformly drawn. 3) Finally, in the *RAND/MIN* variant referred to as *RAND/MINvariant*, we replace X_k^{\min} , in equation (3), by an individual generated according to the multinomial distribution whereas the three other individuals are uniformly drawn.

3.2 Results: comparative study on a benchmark of functions

To evaluate the relevance of our variants, we suggest comparing the different strategies and variants of the DE on the set of functions belonging to the De Jong benchmark [7], also used in [4]:

$$f_1(X) = \sum_{i=1}^3 X(i)^2, \quad -5.12 \leq X(i) \leq 5.12 \quad (6)$$

$$f_2(X) = 100(X(1)^2 - X(2))^2 + (1 - X(1))^2, \quad -2.048 \leq X(i) \leq 2.048 \quad (7)$$

$$f_3(X) = \sum_{i=1}^5 [X(i)], \quad -5.12 \leq X(i) \leq 5.12 \quad (8)$$

$$f_4(X) = \sum_{i=1}^{10} (i \cdot X(i)^4 + \varepsilon_i), \quad -5.12 \leq X(i) \leq 5.12 \quad (9)$$

where ε_i is a white gaussian noise

$$f_5(X) = \frac{1}{0.002 + \sum_{j=1}^2 1/(c_j + \sum_{i=1}^2 (X(i) - a_{ij})^6)} - \gamma, \quad -65.536 \leq X(i) \leq 65.536 \quad (10)$$

where $[x]$ denotes the whole number portion of number x . The parameters are defined as follows: $\gamma = 500$, $c_j = 1 + j$, $a_{1j} = -16 \pmod{(16, 5)} - 2$ and $a_{2j} = -16 \lfloor i/5 \rfloor - 2$ where $\pmod{(x, y)}$ means x modulus y . Note that f_1 and f_2 have a global minimum whereas f_3 , f_4 and f_5 have several local minima.

The simulation results presented hereafter are mean values computed from 50 runs of each DE algorithm. All the variants have been initialized identically to make a fair comparison. The 1st generation is uniformly distributed within the boundary constraints of each test function and the parameter α has been set to 5.

First of all, as all the algorithms often yield approximately the same final values, we do not report those values in this paper.

To illustrate the behavior of the different algorithms, we have represented in figure 1 the average evolution of the fitness function over the iterations for functions $f_1(X)$ and $f_3(X)$.

The functions $f_1(X)$ and $f_2(X)$ have a global minimum. In this case, the optimization algorithms have no chance to fall in local minima. Therefore, the variants *RAND/MIN* and *RAND/BEST* outperform the DE and the proposed variants, as expected. However, it should be noted that our variants converge faster than the classical DE because instead of blindly exploring the solution space, they favor the regions nearby the best individuals of each generation. To illustrate this remark, figure 2 represents different generations of individuals for the classical DE and its variant, the $DE_{variant}$. After 20 iterations, we can observe that, with the variant, the candidate parameter vectors are very close to each other and to the true optimum of $f_2(X)$. On the contrary, when considering the standard DE, the parameter vectors are more scattered.

As for the function $f_3(X)$, it has several local minima. For this difficult optimization issue, the *RAND/MIN* and *RAND/BEST* do not converge to the optimal solution since they are trapped in local minima. On the contrary, the classical DE as well as the variants $DE_{variant}$ and *RAND/BEST*_{variant} reach the global minimum. It should be noted that it takes far less iterations for the $DE_{variant}$ to converge than for the classical DE.

To test the robustness of the proposed approaches, we have evaluated their convergence speed for all the functions of the De Jong benchmark. Table 1 summarizes the number of iterations required for $\{(f_i(X_k^{min}) - \min(f_i)) / (f_i(X_1^{min}) - \min(f_i))\}_{i=1,2,3,4}$ to reach the value 10^{-4} , where $\min(f_i)$ is the minimum of f_i and $f_i(X_1^{min})$ is the minimal value of the fitness function for the 1st generation. We can see in table 1 that our variants are on average better than the standard DE in terms of convergence speed. For the functions that have a global optimum like f_1 and f_2 , they are slower than the *RAND/BEST* and *RAND/MIN* strategies. However, when the functions have local optima, the classical strategies tend to fall in local minima (see fig 1) and fail to retrieve the global minimum. For instance, for the function f_3 , the *RAND/BEST* and the *RAND/MIN* do not converge 44 out of 50 runs and 45 out of 50 runs, respectively. On the contrary, our variants manage to find the global minimum. To conclude, on the benchmark of functions, our variants converge faster than the classical DE and are more robust to local minima than the *RAND/MIN* and *RAND/BEST* implementations. Indeed, in the presence of local minima, the *RAND/BEST* and *RAND/MIN* may never reach the optimal solution because their explorations are centered around the best individual of each generation which may be far from the optimal solution, especially when dealing with local minima. On the contrary, the proposed variants are more efficient by searching in parallel in different “interesting” regions of the solution space. In the next section, we use these algorithms to retrieve the parameters of Hodgkin–Huxley biophysical model from electrical recordings carried out on a neuromimetic circuit.

	f_1	f_2	f_3	f_4	f_5
<i>DE</i>	24	25	82	36	43
$DE_{variant}$	15	19	35	19	25
<i>RAND/MIN</i>	12	13	–	9	–
$RAND/MIN_{variant}$	14	17	–	11	32
<i>RAND/BEST</i>	9	9	–	9	–
$RAND/BEST_{variant}$	17	20	123	22	71

Table 1: Number of iterations required to reach the value 10^{-4} . Mean value over 50 simulations. – : the strategy does not converge.

4. APPLICATION TO A NEUROSCIENTIFIC PROBLEM

4.1 Context

For the past few years, a great deal of interest has been paid to neurosciences, especially neuromorphic engineering. This topic is at the intersection of various fields such as biology, physics and signal processing. Its purpose is to design autonomous robots, artificial sensory and neural systems, etc. It plays a key role in biomedical

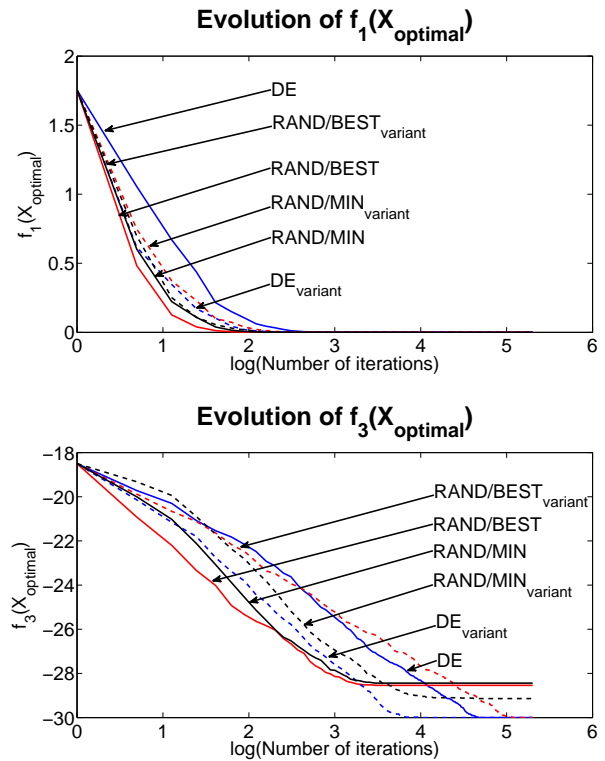


Figure 1: Evolution of f_1 and f_3 versus neperian logarithm of the number of iterations. Plain lines: classical DE strategies, dashed lines: associated variants.

and robotics. Indeed, neural implants have been made to alleviate the undesirable effects of the Parkinson disease and the epilepsy. Another example is the use of bionic arms for amputees. They are connected and controlled by the remaining nerves.

In this section, we focus our attention on approaches that make it possible to characterize and to simulate the neuronal electrical activity. The electrical activity of a neuron is the consequence of the diffusion of ionic species, such as potassium and sodium, through its membrane, see figure 3B. As an example, figure 3A provides eight neuronal spikes recorded thanks to a micro-electrode.

In 1952, Hodgkin and Huxley introduced a model establishing an analogy between electronical circuits and biological phenomena. See figure 4. This formalism leads to a set of differential equations describing the ionic flows with unknown parameters corresponding to ionic conductances for instance. At the same time, Hodgkin and Huxley proposed a method to estimate the parameters of their model from the activity of biological neurons. This method, known as voltage-clamp [6], is still widely used to estimate the parameters of ionic channel models in a neuron. However, this method is based on several approximations which allow for the model parameters to be estimated. Nevertheless, when dealing with the time constants of the set of differential equations, the estimation accuracy has to be improved [10]. In particular, it is difficult to “accurately” estimate the time constants of the model. As an alternative, one can jointly estimate all the parameters of a single ionic channel by finding the minimum of a fitness function F_{fit} . Thus, studies have been carried out to estimate the Hodgkin–Huxley (HH) model parameters from biological recordings. J. L. Madden et al. have used the Levenberg–Marquardt method [9] or a gradient–descent approach [3]. However, the estimations obtained with those methods can correspond to a local extremum of the cost function. More recently, in [2], we proposed to use metaheuristics to estimate the HH model parameters.

Hodgkin–Huxley neuron model

When dealing with the HH model, leak current and two ionic species such as potassium and sodium can be considered. The

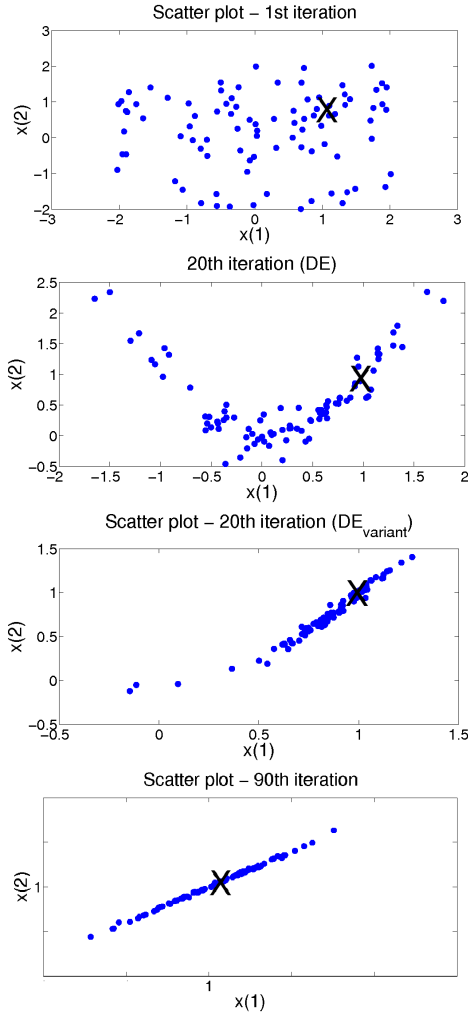


Figure 2: Evolution of scatter plots for f_2 with standard DE and the $DE_{variant}$. The initialization is the same for both algorithms; the final distribution is almost the same for both. Only the final distribution obtained with the classical DE appears. The cross \times represents the optimal set of parameters.

equivalent electrical circuit is represented in figure. 4.

The current flowing across the membrane is integrated on the membrane capacitance, C_{mem} , as follows:

$$C_{mem} \frac{dV_{mem}}{dt} = I_K + I_{Na} + I_{leak} + I_S \quad (11)$$

where V_{mem} denotes the membrane potential, I_S an eventual stimulation or synaptic current and I_K , I_{Na} , I_{leak} the potassium, the sodium and the leak currents respectively. These latter satisfy the following equations:

$$I_K = g_K n^4 (V_{mem} - E_K) \quad (12)$$

$$I_{Na} = g_{Na} m^3 h (V_{mem} - E_{Na}) \quad (13)$$

$$I_{leak} = g_{leak} (V_{mem} - E_{leak}) \quad (14)$$

where $\{g_i\}_{i=K,Na,leak}$ is the maximal conductance value, $\{E_i\}_{i=K,Na,leak}$ is the ion-specific reversal potential, and n , m and h represent the so-called activation term of the potassium channel, the activation term and the inactivation term of the sodium channel respectively. Those terms are dynamic functions describing the permeability of the membrane channels to the ion considered. In addition, they all satisfy the following differential equation:

$$\tau_x \frac{dx}{dt} = x_\infty - x \quad \text{where } x = n, m, h \quad (15)$$

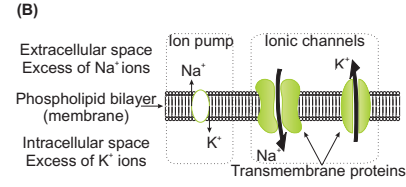
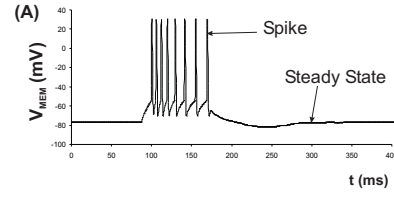


Figure 3: A: recording of membrane voltage V_{MEM} . B: scheme of a neuron membrane.

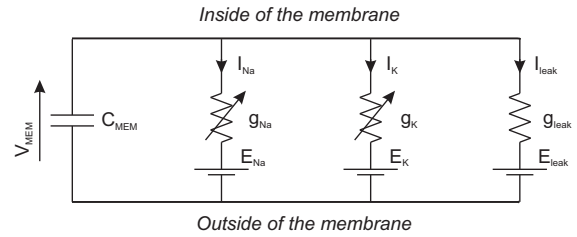


Figure 4: Scheme of equivalent electrical circuit of a neuron.

where τ_x denotes the time constant for the convergence. It should be noted that, when the time t increases to $+\infty$, x converges towards x_∞ which is a sigmoid function of V_{mem} .

$$x_\infty = \frac{1}{1 + \exp\left(\frac{\mp(V_{mem} - V_{offset_x})}{V_{slope_x}}\right)} \quad (16)$$

where V_{offset_x} denotes the sigmoid offset and V_{slope_x} the sigmoid slope. The sign before $(V_{mem} - V_{offset_x})$ is $-$ for the activation term and $+$ for the inactivation.

Considering the set of differential equations (1–6), 15 parameters have to be estimated in order to define the HH model. Thus, we define three vectors storing the unknown parameters:

$$X_K = [g_K \tau_n E_K V_{offset_n} V_{slope_n}], X_{leak} = [g_{leak} E_{leak}],$$

$$X_{Na} = [g_{Na} \tau_m \tau_h E_{Na} V_{offset_m} V_{offset_h} V_{slope_m} V_{slope_h}].$$

X_K , X_{Na} or X_{leak} represent the X vector used in the section 2. **System** : the integrated circuit (IC) designed by our group [10] reproduces in real-time the electrical activity of a neuron following the HH formalism. For a given set of parameters, we record individually each ionic channel response by applying different step values, denoted $stim$, on the membrane voltage.

To find the parameters of the HH model, the estimation method can be applied separately on each ionic channel. Here, we will focus our attention on the potassium channel. Since the leak current I_{leak} can be obtained by solving the affine equation (4), g_{leak} and E_{leak} can be estimated thanks to a linear regression. The parameter estimation of potassium requires more complex techniques such as DE because of the strong non-linearities of its equations.

During 50ms (biological real-time, with a sampling period of 0.01ms) we record independently the currents, denoted $I_{reference,K}$ and $I_{reference,Na}$, after applying successive steps values on the membrane voltage. Then, we apply a discretization³ to the HH model equations with the estimated parameters to obtain an analytical expression of the currents, denoted $I_{software}$.

³Approximation of partial difference by : $\tau_x \frac{x_n - x_{n-1}}{\Delta t} = x_\infty - x_n$

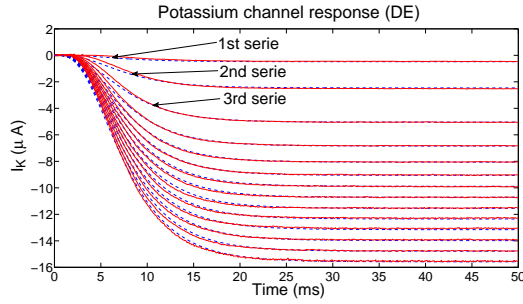


Figure 5: Potassium channel Response. Continuous line : potassium channel hardware response to different steps of stimulations. Dashed line : software response obtained with extracted parameters ($DE_{variant}$).

The fitness function F_{fit} that we suggest minimizing is defined by :

$$F_{fit}(X_K) = \sum_t \sum_{stim} (\beta \times (I_{reference,K}(t, stim) - I_{software}(X_K, t, stim)))^2 \quad (17)$$

where t corresponds to the time and $stim$ corresponds to the different values applied to the membrane voltage. Note that a factor $\beta = 10^6$ is considered to avoid numerical limitations.

4.2 Comparative study on the potassium ionic channel

The responses of the potassium channel, $I_{reference,K}$ and $I_{software}$ are shown in figure 5 for the $RAND/MIN_{variant}$. We can see that the curves are almost overlaid for each stimulation current. The small discrepancy during the 15 first milliseconds is due to chip defects. Indeed to compute n^4 in equation (12) multipliers involving a translinear loop are used. However, these multipliers are not really linear with weak currents [10]. Moreover, when performing random multiple starts of this algorithm in the same conditions, we obtain the same extracted parameters with a precision of six significative digits.

In figure 6, we have represented the evolution of the fitness function for only 3 strategies for the sake of clarity: the classical DE, the $RAND/MIN$ which exhibits the smallest time of convergence, and its variant, the $RAND/MIN_{variant}$. The $RAND/MIN_{variant}$ performs better than the classical DE, but the $RAND/MIN$ converges in less iterations than our variant. The most likely explanation is that the fitness function for the potassium channel has few local minima despite of the complexity and non-linearity of the equations. We are currently working on the joint estimation of the sodium and potassium channels where the proposed approach is expected to yield good results.

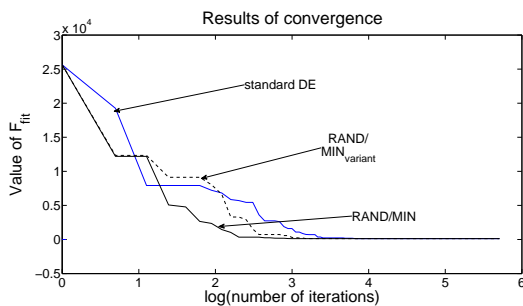


Figure 6: Evolution of fitness function with the DE and their variants. Dashed line represents the evolution of the variant corresponding to the continuous line classical algorithm.

5. CONCLUSION

In this paper, we propose new variants of DE intended to speed up the convergence while avoiding local minima. We have first validated the performance of the proposed approach on the De Jong

benchmark of functions. It takes less iterations for our variant than for the classical DE to converge. In addition, when dealing with local minima, our approach efficiently explores in parallel different regions of the solution space to find the global minimum unlike the $RAND/BEST$ and $RAND/MIN$ strategies. Then, we have applied it on a problem originally adressed by neuroscientists. This study illustrates the relevance of the proposed variant of the DE on complex and noisy systems involving differential equations. In further works, it would be interesting to compare the performance of DE with another new simple and efficient optimization algorithm, the artificial bee colony (ABC) [8].

Acknowledgements

Research was partly funded by the E.U. Grant FACETS (FP6-IST-FETPI-2004-15879).

REFERENCES

- [1] P. Besson, V. Popovici, J.-M. Vesin, J.-P. Thiran, and M. Kunt. Extraction of audio features specific to speech production for multimodal speaker detection. *IEEE Transactions on Multimedia*, 10:63–73, Januar 2008.
- [2] L. Buhry, S. Saïghi, A. Giremus, E. Grivel, and S. Renaud. Parameter estimation of the Hodgkin–Huxley model using metaheuristics: application to neuromimetic analog integrated circuits. In *IEEE–BioCAS08*, 2008.
- [3] S. Doi, Y. Onoda, and S. Kumagai. Parameter estimation of various Hodgkin-Huxley-type neuronal models using a gradient-descent learning method. In *IEEE–SICE 2002*, pages 1685–1688, August 2002.
- [4] V. Feoktistov and S. Janaqi. Generalization of the strategies in differential evolution. In *IEEE, 18th International Parallel and Distributed Processing Symposium*, April 2004.
- [5] J. Grolleau, E. Grivel, and M. Najim. Two ways to simulate rayleigh fading based on the sinusoidal stochastic model. *IEEE Signal Processing Letters*, 15:107–110, 2008.
- [6] A. Hodgkin and A. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *Jour. of Physiology*, 117:500–544, 1952.
- [7] K. D. Jong. *An analysis of the behaviour of a class of genetic adaptive systems*. PhD thesis, University of Michigan., Michigan, 1975.
- [8] D. Karaboga and B. Basturk. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm. *Jour. of Global Optimization*, 39:459–471, 2007.
- [9] J. L. Madden, Z. B. Miled, R. C. Y. Chin, and J. Schild. On parameter estimation for neuron models. In *BIBE '00: Proceedings of the 1st IEEE International Symposium on Bioinformatics and Biomedical Engineering*, page 253, 2000.
- [10] S. Saïghi, L. Buhry, Y. Bornat, G. N’Kaoua, J. Tomas, and S. Renaud. Adjusting the neuron models in neuromimetic ICs using the voltage–clamp technique. In *IEEE–ISCAS08*, pages 1564–1567, May 2008.
- [11] P. Shroeter, T. L. J. M. Vesin, and R. Meuli. Robust parameter estimation of intensity distributions for brain magnetic resonance images. *IEEE Trans. on Medical Imaging*, 17(2):172–186, 1998.
- [12] R. Storn and K. Price. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11:341–359, 1997.