# Arnold Cat Map and Sinai as Chaotic Numbers Generators in Evolutionary Algorithms

Lenka Skanderova and Ivan Zelinka

Department of Computer Science, VSB - Technical University of Ostrava, 17. listopadu 15, 708 33 Ostrava-Poruba Czech Republic
{lenka.skanderova,ivan.zelinka}@vsb.cz

**Abstract.** It is commonly known that evolutionary algorithms use pseudorandom numbers generators. They need them for example to generate the first population, they are necessary in crossing or perturbation process etc. In this paper chaos attractors Arnold Cat Map and Sinai are used as chaotic numbers generators. The main goal was to investigate if they are usable as chaotic numbers generators and their influence on the cost functions convergence's speed. Next goal was to compare reached values of Arnold Cat Map and Sinai and assess which attractor is better from the view of cost function convergence's speed.

## 1 Introduction

Differential evolution (DE) and Self migrating aglorithm (SOMA) belong to the great family of evolutionary algorithms. Evolutionary algorithms are based on three basic principles - natural selection, crossing and mutation. They work with population of individuals, which is developing during generation cycles. DE is typical example of the evoutionary algorithm, because in its principle we can find each aspect mentioned above. There is the population of individuals, individuals are crossed and mutated. Best survive while worse die [12]. On the other hand the principle of SOMA is different. There is the population of individuals too but individuals migrate in space of possible solutions. There is no offspring. The individuals just change their positions. Generation cycles are replaced by migration cycles and crossing is replaced by perturbation.

For DE and SOMA, generator of pseudorandom numbers is necessary in many steps of the algorithms. The first population of individuals is created randomly. Each individual is created by parameters and fitness value. Each parameter has its own lower and upper bound. The parameters of individuals in the first population in these algorithms are generated randomly in that bounds. It is not possible to cross over these bounds. Pseudorandom numbers generators are necessary for example in crossing, perturbation vector creating etc. In this paper Sinai and Arnold Cat Map have been chosen as the chaotic numbers generators. As the cost functions 1st de Jong's, Schwefel's and Ranna's function have beed chosen.

In 2013 papers [1]-[5] have been written in connection with DE. In [6]-[9] DE and SOMA are connected with chaos. In [10] we can find a pseudorandom number generator of q-Gaussian random variables based on chaotic map dynamics.

I. Zelinka et al. (eds.), *AETA 2013: Recent Advances in Electrical Engineering and Related Sciences*, Lecture Notes in Electrical Engineering 282,
DOI: 10.1007/978-3-642-41968-3_39, 

## 2 Evolutionary Algorithms

As it was mentioned above, evolutionary algorithms are based on three basic principles according to Darwin's and Mendel's theory. The family of these algorithms is very robust. In this paper, DE and SOMA have been chosen because their main principles are different.

### 2.1 Differential Evolution

Differential evolution works with population of individuals. These individuals are crossed and mutated and best survive while worse die. At beginning the first population is generated randomly. The number of individuals is given by parameter *NP*. As it was mentioned above individuals are created from parameters and fitness value. Fitness value says how good is this individual in population. When the first population is generated, reproduction cycle can begin:

- For each individual three next random individuals are chosen.
- Mutation is realize according to the Eq.1, where $v_j$ denotes noise vector, $r1$, $r2$, $r3$ are three randomly chosen individuals from the population and $F$ denotes mutation constant, its values can move in interval [0,2].
- New individual creating. For each parameter random number $r$ from interval [0,1] is generated. If $r$ is smaller than crossing probability $CR$ the value from the noise vector is chosen as a parameter of new individual, else parameter from the actual individual is chosen.
- The fitness value of new individual is computed. If it is smaller than fitness of actual individual, new individual will replace actual individual, else, new individual will be forgotten and actual individual will stay in the population.

There are many types of DE. These types ussually differ in noise vector computation. In this paper DE/rand/1/exp is used [11].

$$v_j = x^G_{r3,j} + F(x^G_{r1,j} - x^G_{r2,j}) \tag{1}$$

### 2.2 Self Organizing Migrating Algorithm (SOMA)

The principle of SOMA differs from DE. There is no offspring, individuals just migrate in the space of possible solutions. The begin of the algorithm is same like in DE. First popualtion is generated randomly. Next steps differs according to the type of SOMA. In this paper SOMA AllToOne is used. Except this, AllToAll, AllToOne Random, AllToAll Adaptive and AllToOne Adaptive exist. In the case of AllToOne next steps are realize [13]:

- *Leader*, the individual with best fitness, is chosen.
- Each individual migrate to the *Leader* – for each individual perturbation vector $\alpha$ is created. Perturbation vector consists of 0 and 1, and says in which direction the individual will migrate to the *Leader*.

- Individual migrate to the *Leader* in steps. The value of parameter $t$ which denotes step is ussually 0.11. Migration of individual is realize according to the Eq.2 where $r$ is a new candidate solution, $r_0$ denotes actual individual, $m$ is a difference between *Leader* and start position of individual and $t$ is the step, $t \in [0, PathLength]$.

$$r = r_0 + mt\alpha \tag{2}$$

## 2.3 Chaos

There is no precise definition of chaotic system, we can find just sme conditions to clasify the system as chaotic:

- The system must be sensitive to initial conditions. This condition is very well known as a "butterfly effect" [14].
- The system must be topologically mixing. The typical example of chaotic system is mixing of colored dyes or fluids[15].
- The system's periodic orbits must be dense. Every point in the space is approached arbitrary closely by periodic orbits.

Chaotic map is define such that:

**Definition 1.** *A map: $f : X \to X$ of a metric space is said to exhibit sensitive dependence on initial conditions if there is a $\Delta > 0$, called a sensitivity constant, such that for every $x \in X$ and $\varepsilon > 0$ there exists a point $Y \in X$with $d(x,y) < \varepsilon$ and $d(f^N(x), f^N(y)) \geq \Delta$ for $N \in \mathbb{N}$ [14].*

According to [14] this means that "*the slightest error $(\varepsilon)$ in any initial condition $(x)$ can lead to a macroscopic discrepancy $(\Delta)$ in the evolution of the dynamics. $\Delta$ does not depend on x, nor on $\varepsilon$ but only on the system.*"

**Theorem 1.** *Chaotic maps exhibit sensitive dependence on initial conditions, except when the entire space consists of a single periodic orbit [14].*

To visualize the chaotic movement phase diagram of movement is ussually used. Time is implicit in this diagram and each axis represents one dimension of the state. If the graph creates the closed curve, this curve will be called orbit. It can be very often seen that the system ends in the same movement for all begining states in the area around this movement - it seems that this system is attracted to this movement. This movement is called attractor of the system [16]. In [17] author says about attractor: "*If oneconsiders a system and its phase space, then the initial conditions may be attracted tosome subset of the phase space (attractor) at time $t \to \infty$.*" Next author says about strange attractor: "*For many other attractors the attracting set can be much more irregular (some woud say pathological) and, in fact, can have a dimension that is not an integer. Such sets have been clled fractal and, when they are attractors, they are called strange attractors.*"

In this paper Sinai attractor (Eq. 3) and Arnold Cat Map (Eq. 4) have been chosen. Sinai billiard is mentioned in [21] and [22]. In [18], [19] and [20] we will find Arnold Cat Map's using.

$$\begin{aligned} x_{n+1} &= (x_n + y_n + \sigma \cos(2\pi y_n)) \bmod 1, \\ y_{n+1} &= (x_n + 2y_n) \bmod 1 \end{aligned} \tag{3}$$

$$\begin{aligned} x_{n+1} &= (x_n + y_n) \bmod 1, \\ y_{n+1} &= (x_n + ky_n) \bmod 1 \end{aligned} \tag{4}$$

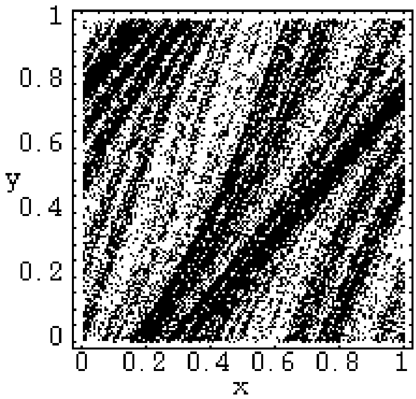


**Fig. 1.** Sinai map

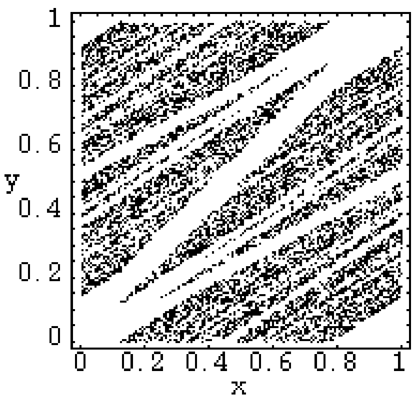


**Fig. 2.** Arnold Cat Map

## 3 Motivation

The main goal was to find out if Sinai and Arnold Cat map are useful as a generators of chaotic numbers and comparison of these chaotic equations each other. These equations have been compared from the view of the evolutionary algorithms (DE and SOMA) cost functions convergence's speed.

## 4 Experiments Design

For experiments HP Pavilion dv7-6050 with processor Intel Core i7 with frequency 2 GHz, 4 GB RAM and graphic card AMD Radeon HD 6770M and Microsoft Visual Studio 2010 have been used. The experiments have beed processed by Mathematica 8 and Gnuplot 4.6. Precise setting of DE is mentioned in Table 1 and precise setting of SOMA in Table 2. Schwefel's function, 1st de Jong's function and Ranna's function have been chosen as the testing functions. Schwefel's function has many local extremes while 1st de Jong is not so jagged. And Ranna's function has been chosen because there is no global extreme described in the literature. For each function and chaos equation special experimental set has been created. For each equation, Sinai and Arnold Cat Map, one set of experiments has been created. In each set each experiment has been repeated one hundred times.

**Table 1.** DE setting

| Parameter | Value |
|---|---|
| $NP$ | 100 |
| $D$ | 20 |
| $Migrations$ | 300 |
| $PathLength$ | 3 |
| $Step$ | 0.11 |
| $\alpha$ | 0.1 |

**Table 2.** SOMA setting

| Parameter | Value |
|---|---|
| $NP$ | 50 |
| $D$ | 20 |
| $Generations$ | 1800 |
| $F$ | 0.9 |
| $CR$ | 0.4 |

## 5 Results

Results are mentioned in Tab. 3, where we can see minimum, maximum and average reached values for DE and SOMA, where 1st de Jong's, Ranna's and Schwefel's functions have been used as testing functions. In Fig. 3 comparison of Arnold Cat Map and Sinai used as chaotic numbers generators in DE, where 1st de Jong's function has been used as testing function is depicted. Fig. 4 shows the comparison, where SOMA has been used as the evolutionary algorithm and Ranna's function has been used as the cost function and in Fig. 5 comparison of both chaotic numbers generators used in DE, where Schwefel's function has been chosen as the cost function is depicted.

**Table 3.** Comparison of chaotic numbers generators Sinai and Arnold Cat Map, from the view of evolutionary algorithm convergence's speed. For better illustration there is smaller numbers in the rows with 1st de Jong's function. Settings of evolutionary algorithms are mentioned in Tables 1 and 2.

| Evolutionary algorithm | Pseudorandom numbers generator | Function | Min | Max | Average |
|---|---|---|---|---|---|
| DE | Sinai | 1st de Jong | $6.963 \times 10^{-9}$ | $7.015 \times 10^{-8}$ | $2.530 \times 10^{-8}$ |
| | | Ranna | -6220.839 | -5240.327 | -5641.828 |
| | | Schwefel | -8379.658 | -8379.658 | -8379.658 |
| | Arnold Cat Map | 1st de Jong | $9.343 \times 10^{-9}$ | $9.634 \times 10^{-8}$ | $3.859 \times 10^{-8}$ |
| | | Ranna | -6359.965 | -5048.777 | -5460.802 |
| | | Schwefel | -8379.658 | -8379.658 | -8379.658 |
| SOMA | Sinai | 1st de Jong | $3.021 \times 10^{-33}$ | $1.020 \times 10^{-31}$ | $2.199 \times 10^{-32}$ |
| | | Ranna | -9065.627 | -8477.0319 | -8789.540 |
| | | Schwefel | -8379.658 | -8261.219 | -8377.289 |
| | Arnold Cat Map | 1st de Jong | $4.569 \times 10^{-33}$ | $1.185 \times 10^{-31}$ | $3.509 \times 10^{-32}$ |
| | | Ranna | -8967.828 | -8402.458 | -8669.560 |
| | | Schwefel | -8379.658 | -8261.219 | -8376.105 |

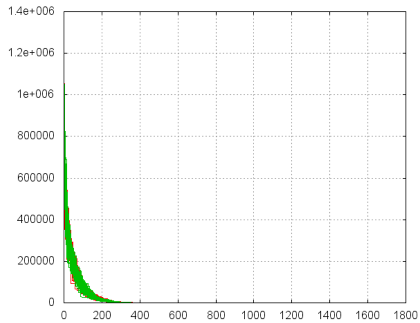


**Fig. 3.** DE, 1st de Jong's function, comparison of Arnold Cat Map (red) and Sinai (green)

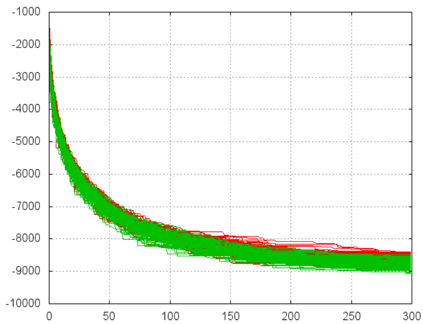


**Fig. 4.** SOMA, Ranna's function, comparison of Arnold Cat Map (red) and Sinai (green)

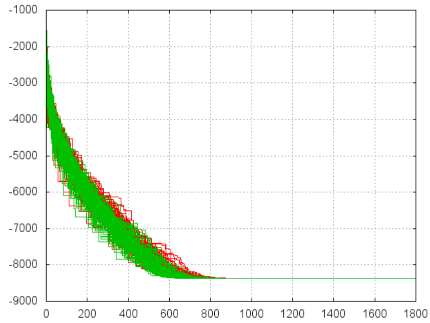


**Fig. 5.** DE, Schwefel's function, comparison of Arnold Cat Map (red) and Sinai (green)

## 6 Conclusion

In Tab. 3 there are described minimum, maximum and average values reached during the evolution for DE and SOMA, where 1st de Jong's, Ranna's and Schwefel's functions has been used as testing function and Arnold Cat Mapand Sinai have beed used as psedorandom numbers generators. From these results we can make some conclusions:

- We can say that both algorithms are usable as the generators of chaotic numbers. When we look at the Table 3 we will see that their results are comparable.
- DE, 1st de Jong's function: We can say that when Arnold Cat Map was used, 1st de Jong's function convergence has been faster. But the difference between Arnold Cat Map and Sinai is very small and we can consider it insignificant, see Fig. 3.
- DE, Ranna's function: There is no global minimum mentioned in literature by Ranna's function. As we can see both algorithms reached values around -6300. Arnols Cat Map reached smaller values than Sinai and the situation is the same like in the previous case, see Fig. 5.
- DE, Schwefel's function: Both algorithms reached the global minimum of the Schwefel's function. Their values mentioned in Table 3 are the same.
- SOMA, 1st de Jong's function: Reached values are significantly smaller than in DE. From the view of chaotic numbers generators both algorithms reached comparable values. Arnold Cat Map reached smaller values than Sinai. But the difference is insignificant.
- SOMA, Ranna's function: The same situation happened when Ranna's function has been used as the cost function. The results of both chaotic numbers generators are comparable, see Fig. 4.
- SOMA, Schwefel's function: As well as in previous cases both algorithms reached comparable values. In Schwfel's function they reached the global minimum of the cost function.
- The results of both attractors are very similar. When we look at Figs. 1 and 2 it is not surprising phenomenon, because these attractors are relatively similar.

**Acknowledgements.** The following grants are acknowledged for the financial support provided for this research: Grant Agency of the Czech Republic - GACR P103/13/08195S, by the Development of human resources in research and development of latest soft computing methods and their application in practice project, reg. no. CZ.1.07/2.3.00/20.0072 funded by Operational Programme Education for Competitiveness, co-financed by ESF and state budget of the Czech Republic, and by Grant of SGS No. SP2013/114, VŁB - Technical University of Ostrava, Czech Republic.

## References

1. Koloseni, D., et al.: Differential evolution based nearest prototype classifier with optimized distance measures for the features in the data sets. Expert Systems with Applications 40, 4075–4082 (2013), doi:10.1016/j.eswa.2013.01.040
2. Tasgetiren, M.F., et al.: A variable iterated greedy algorithm with differential evolution for the no-idle permutation flowshop scheduling problem. Computers & Operations Research 40, 1729–1743 (2013), doi:10.1016/j.cor.2013.01.005
3. De Melo, V.V., Carosio, G.L.C.: Investigating Multi-View Differential Evolution for solving constrained engineering design problems. Expert Systems with Applications 40, 3370–3377 (2013), doi:10.1016/j.eswa.2012.12.045
4. Depolli, M., et al.: Asynchronous Master-Slave Parallelization of Differential Evolution for Multi-Objective Optimization. Evolutionary Computation 21, 261–291 (2013)
5. Maione, G., Punzi, A.: Combining differential evolution and particle swarm optimization to tune and realize fractional-order controllers. Mathematical and Computer Modelling of Dynamical Systems 19, 277–299 (2013), doi:10.1080/13873954.2012.745006
6. Senkerik, R.: On the Evolutionary Optimization of Chaos Control - A Brief Survey. In: Zelinka, I., Snasel, V., Rössler, O.E., Abraham, A., Corchado, E.S. (eds.) Nostradamus: Mod. Meth. of Prediction, Modeling. AISC, vol. 192, pp. 35–48. Springer, Heidelberg (2013)
7. Senkerik, R., Oplatkova, Z., Zelinka, I.: Evolutionary Synthesis of Control Rules by Means of Analytic Programming for the Purpose of High Order Oscillations Stabilization of Evolutionary Synthesized Chaotic System. In: Zelinka, I., Snasel, V., Rössler, O.E., Abraham, A., Corchado, E.S. (eds.) Nostradamus: Mod. Meth. of Prediction, Modeling. AISC, vol. 192, pp. 191–201. Springer, Heidelberg (2013)
8. Zelinka, I., et al.: An Ivestigation on Evolutionary Reconstruction of Continuous Chaotic Systems. Mathematical and Computer Modelling 57, 2–15 (2013), doi:10.1016/j.mcm.2011.06.034
9. Nguyen, T.D., Zelinka, I.: Using Method of Artificial Intelligence to Estimate Parameters of Chaotic Synchronization Systems. In: Mendel 2011-17th International Conference on Soft Computing, Mendel, pp. 22–29 (2011)
10. Umeno, K., Sato, A.H.: Chaotic Method for Generating q-Gaussian Random Variables. IEEE Transactions on Information Theory 59, 3199–3209 (2013), doi:10.1109/TIT.2013.2241174
11. Wang, H., et al.: Gaussian Bare-Bones Differential Evolution. IEEE Transactions on Cybernetics 43, 634–647 (2013), doi:10.1109/TSMCB.2012.2213808
12. Caamano, P., et al.: Evolutionary algorithm characterization in real parameter optimization problems. Applied Soft Computing 13, 1902–1921 (2013), doi:10.1016/j.asoc.2013.01.002
13. Senkerik, R., et al.: Synthesis of feedback controller for three selected chaotic systems by means of evolutionary techniques: Analytic programming. Mathematical and Computer Modelling 57, 57–67 (2013), doi:10.1016/j.mcm.2011.05.030
14. Hasselblatt, B., Katok, A.: A First Course in Dynamics: With a Panorama of Recent Developments. Cambridge University Press (2003) ISBN 0-521-58750-6

15. Vellekoop, M., Berlund, R.: Om Intervals, Transitivity = Chaos, vol. 101, pp. 353–355. The American Mathematical Monthly, JSTOR (1994)
16. Liu, C.: A novel chaotic attractor. Chaos, Solitons & Fractals 39, 1037–1045 (2009)
17. Grebogi, C., et al.: Chaos, Strange Attractors, and Fractal Basin Boundaries in Nonlinear Dynamics. In: Non-Linear Physics for Begginers: Fractals, Chaos, Pattern Formation, Solutions, Cellular Automata and Complex Systems, pp. 111–117. Word Scientific Publishing Co. Pte. Ltd. (1998)
18. Chen, F., et al.: Period Distribution of the Generalized Discrete Arnold Cat Map for N=2(e). IEEE Transactions on Information Theory 59, 3249–3255 (2013), doi:10.1109/TIT.2012.2235907
19. Bao, J.H., Yang, Q.G.: Period of the discrete Arnold cat map and general cat map. Nonlinear Dynamics 70, 1365–1375 (2012), doi:10.1007/s11071-012-0539-3
20. Kanso, A., Ghebleh, M.: A novel image encryption algorithm based on a 3D chaotic map. Communications in Nonlinear Science and Numerical Simulation 17, 2943–2959 (2012), doi:10.1016/j.cnsns.2011.11.030
21. Yang, R., et al.: Harnessing quantum transport by transient chaos. CHAOS 23 (March 2013), doi:10.1063/1.4790863
22. Pecora, L.M., et al.: Regularization of Tunneling Rates with Quantum Chaos. International Journal of Bifurcation and Chaos 22 (October 2012), doi:10.1142/S0218127412502471