1 ON THE APPLICATION OF GENETIC ALGORITHMS TO DIFFERENTIAL EQUATIONS

George Daniel MATEESCU[*]

Abstract

Genetic algorithms can be used in order to solve optimization problems. Such a technique may be used in order to solve differential equations.

Key words: differential equations; genetic algorithms;

JEL Classification: C61, C63

The main issue

There are some fundamental econometric issues which lead to differential equations. For example, taking into account the demand-supply equilibrium, we may consider the variation of the price as being a ratio of the difference between demand and supply.

$$\frac{\Delta P}{\Delta t} = a\left(D(t) - S(t)\right)$$

where *a* is a positive constant. Assuming small values for *t*, we obtain the differential model:

$$P'(t) = a\left(D(t) - S(t)\right)$$

When demand exceeds supply, the right part represents a positive value, which implies that *P* is an increasing function.

Formally, we may consider the general *Cauchy* problem

$$y' = f(x, y), y(x_0) = y_0,$$

where *x* is the independent variable and $y = y(x)$ is the dependent variable.

By using the classical assumption:

$$f : \left[x_0 - X, x_0 + X\right] \times \left[y_0 - Y, y_0 + Y\right] \to \mathsf{R}$$

is continuous and satisfies the *Lipschitz* condition

$$\left|f(x, y_1) - f(x, y_2)\right| \le L\left|y_1 - y_2\right|$$

---

[*] *Romanian Academy, Institute for Economic Forecasting, e-mail: daniel@mateescu.ro.*

it results there exists a single solution *y*.

There are many methods used to find the solution, but, in practice, we always solve the problem by using numerical methods, the *Runge-Kutta* method being the best known. The main goal of this article is to underline the possibility of using a different method, based on genetic algorithms.

Basics of genetic algorithms

Admissible solutions. As in the case of the *Runge-Kutta* method, by using the genetic algorithm we will find the values of the unknown function $y = y(x), y:[a,b] \rightarrow \mathrm{R}$, according to a finite set of equidistant values of the independent variable $x_0 = a < x_1 < ... < x_n = b$, $x_i = a + ih$, $h = \dfrac{b-a}{n}$.

We denote by $y_i = y(x_i), i = 1...n$ the values of the unknown function *y*, in accordance with the given division. Thus, the vector $(y_1, y_2, ..., y_n)$ is an admissible solution.

Population. By using the biological pattern, we will consider the population as being a subset of all chromosomal combinations, i.e. a subset of admissible solutions. Given an instant *t*, we denote the population by $P(t)$. One individual $y = (y_1, y_2, ..., y_n)$ is characterized by its genetic heritage, i.e. the y_i values.

Selection. The individuals in a biological population are, more or less, adapted. Thus, in order to simulate biological (natural) selection, we will select, in each stage, only one subset of individuals, namely those who are best adapted. The surplus of individuals is eliminated, taking into account the decreasing values of the performance function.

In order to evaluate each individual, we will use the following approximate formula for the derivative:

$$y'(x_i) \approx \frac{y_i - y_{i-1}}{h} \qquad \left| y'(x_i) - \frac{y_i - y_{i-1}}{h} \right| \leq const. \cdot h$$

Consequently, the discrete form of the *Cauchy* problem will be:

$$\frac{y_i - y_{i-1}}{h} = f(x_i, y_i), i = 1...n$$

(1)

The above system is, generally, nonlinear.

Finding the vector $(y_1, y_2, ..., y_n)$ which satisfies the above conditions is our goal.

Of course, for an admissible solution, we do not have the equality (1) and, consequently, we have to consider the error formula:

$$\left(\frac{y_i - y_{i-1}}{h} - f(x_i, y_i) \right)^2$$

The performance function, associated to an individual $y = (y_1, y_2, ..., y_n)$ will be:

$$F(y) = \sum_{i=1}^{n} \left(\frac{y_i - y_{i-1}}{h} - f(x_i, y_i) \right)^2$$

An individual from $P(t)$ will be better adapted if its chromosomal heritage implies a smaller value of the function *F*.

Offspring. Two individuals can generate offspring by combining their chromosomal sequences. Thus, by using a *cross-over* operation, $(y_1, y_2, ..., y_n)$ and $(z_1, z_2, ..., z_n)$ will generate:

$$(y_1, y_2, ..., y_k, z_{k+1}, ..., z_n) \text{ and } (z_1, z_2, ..., z_k, y_{k+1}, ..., y_n)$$

Mutation. Each chromosome may suffer some modifications, which may be *hazardous* or may result as *genetic acquisitions*, if we assume that "*education and instruction*" have as result a better performance.

Formally, we will consider that $y_i \pm \varepsilon$ is a mutation for y_i.

Convergence issue

We denote by u_t the best adapted individual in the population, at the instance *t*, i.e. the individual in the population $P(t)$ which has the minimum value of the function *F*. In a previous paper[1] we have already stated that the sequence $(u_t)_{t \geq 0}$ converges, its limit being the solution of the optimization problem $\inf F$.

While the solution is the limit of a convergent sequence, by applying the genetic algorithm, the following assertion is true:

For $\varepsilon > 0$, there is a $y = (y_1, y_2, ..., y_n)$ such that:

$$F(y) = \sum_{i=1}^{n} \left(\frac{y_i - y_{i-1}}{h} - f(x_i, y_i) \right)^2 < \varepsilon$$

It results there is a $y = (y_1, y_2, ..., y_n)$ such that

$$\left| \frac{y_i - y_{i-1}}{h} - f(x_i, y_i) \right| < h$$

Taking into account the approximation of the derivative, we have

$$\left| y'(x_i) - f(x_i, y_i) \right| \leq \left| y'(x_i) - \frac{y_i - y_{i-1}}{h} \right| + \left| \frac{y_i - y_{i-1}}{h} - f(x_i, y_i) \right| < Ch$$

---

[1] *Mateescu G.D., "Optimization by using evolutionary algorithms with genetic acquisitions",* Romanian Journal of Economic Forecasting*, No. 2/2005.*

The last relation shows that the final value $y = (y_1, y_2, ..., y_n)$ is an approximate solution of the *Cauchy* problem, for small values of *h*.

Algorithm

Evolution may be expressed as an algorithm, which uses the basics of life: population, offspring, mutation and selection. We adapted the general algorithm, as described bellow:

Step 1. Generate a population, corresponding to an initial moment *t*, $P(t)$. The initial population consists of a set of *M* vectors, having as chromosomes some uniform perturbation of the initial condition, i.e. $(y_0 \pm \varepsilon_1^m, y_0 \pm \varepsilon_2^m, ..., y_0 \pm \varepsilon_n^m)$, $m = 1..M$, $\varepsilon_i^j > 0$.

Step 2. Let be $t \leftarrow t+1$; generate offspring. Only the best adapted individuals will participate. Given two individuals $(y_1, y_2, ..., y_n)$ and $(z_1, z_2, ..., z_n)$ we will select the closest chromosomes, i.e. the value *k* such that $|y_k - z_k| = \min_{i=1..n} |y_i - z_i|$ and the algorithm will generate two new individuals $(y_1, y_2, ..., y_k, z_{k+1}, ..., z_n)$ and $(z_1, z_2, ..., z_k, y_{k+1}, ..., y_n)$.

Step 3. Applying random mutations.

As we stated in a previous paper[1], the algorithm will converge to an optimal value if the mutation will respect the gradient of the function

$$F(y_1, y_2, ..., y_n) = \sum_{i=1}^{n} \left(\frac{y_i - y_{i-1}}{h} - f(x_i, y_i) \right)^2.$$

Step 4. Evaluate individuals, by using the performance function *F*.

Step 5. Eliminate some individuals, in order to keep the constant population.

Step 6. Continue with ***step 2***, while ending condition has not yet been reached.

The ending condition will be a posteriori:

$$|F(u_t) - F(u_{t-1})| < h ,$$

u_t being the best performance individual in $P(t)$.

Conclusion

Genetic programming may be used in order to find numerical solutions for differential equations. As well as the *Runge Kutta* method, we can find approximate values of the unknown function, with respect to equidistant values for the independent variable.

---

[1] Mateescu G.D., *op cit.*

We used the above algorithm for the equation $y'(t) = 2(t^3 - t^2), y(1) = 0$ in order to find an approximate solution $y : [1,2] \to \mathrm{R}$. We selected a population having 30 individuals, classified by the performance function. At each step, the first 10 individuals will generate 20 new members (offspring). For $h = \dfrac{1}{10}$, the final vector *y* is

$$(0, 0.01, 0.05, 0.13, 0.26, 0.45, 0.71, 1.07, 1.53, 2.11, 2.83)$$

References

Albu L.L, *Macroeconomie non-lineara şi prognoză*, Academia Română, 2002.

Banzhaf W., Nordon P., Keller R.E., Francone F.D., *Genetic Programming – An introduction,* Morgan Kaufmann Publishiers, San Francisco 1998.

Mateescu G.D., "Optimization by using evolutionary algorithms with genetic acquisitions", *Romanian Journal of Economic Forecasting*, No. 2/2005.

Stoer J., Bulirsch R., *Introduction to Numerical Analysis*, Springer-Verlag, 1992.