

# Real-time Evolutionary Computation in the Control of Mobile Cyber-Physics System

Gennady Rogachev<sup>1,\*</sup>, Mikhail Patkin<sup>1</sup> and Nikolai Rogachev<sup>1</sup>

<sup>1</sup>Samara State Technical University, 443100, Samara, Russia

**Abstract.** The procedure of the time-optimal controller synthesis in the mobile cyber-physical system control problems in the form of the Dubins car is considered. In the decision quality assessment, the total time spent on executing the task of transferring an object to a given point is taken into account, including the time for the control program calculation and the time of the object movement to the target. Both online and offline variants of the production model numerical determination of the controller as a system of rules “condition” – “action” are considered. As an online variant, it is suggested to use an anytime genetic algorithm. A comparison between online and offline methods of calculating a rule system for the regulator performance is made. The advantages of the online method in solving the problem of transferring the mobile cyber physical system in a homogeneous environment and the advantages of an offline method in a heterogeneous environment are demonstrated.

## 1 Cyber-physical systems as special kind of systems

The world now is at the beginning of a new technological revolution. Its essence is in the iniquitousness of cyber-physical systems (CPS - cyber-physical systems) [1-3]. The essential feature of CPS is an inextricable connection between physical elements they include and elements implementing the calculation. The important category of cyber-physical systems is mobile CPS, the physical part of which has its own mobility. When developing cyber-physical systems one must take into account that they are a single set of interactive computing, communication and physical processes.

Built-in (in the real physical world) computers immediately affect physical processes and, in turn, are influenced by them. The calculations and the physical process are becoming the integral whole; it is impossible to predict and understand the behavior of any of these two parts separately. When parts of such a system (sensors, controllers, actuation devices) exchange information, a third component is added to the specified integral whole: the process of network communication. The robust relationship between discrete computational processes and information transfer with continuous processes in the control objects requires a fundamentally new technology of description, analysis and synthesis of cyber-physical systems.

A certain step in this direction is to use a production model of the controller when modeling a cyber-physical system as a digital control system of continuous object [4]. The dynamic part of the system, which determines the performance of the control object, is described by differential equations, and the control part (controller, sensors, actuation devices and data transmission network from the object to the control part and back) – by the algorithm of its work (a productions system, a system of rules "condition" - "action"). The production (algorithmic) form of controllers description allows unifying the problem of their synthesis. Indeed, the number of elements of the production systems (the pairs of the type "condition" –"action") and its contents are to be specified regardless of the specific problem. At the same time the multi-step synthesis, comprising the controller structure selection, the definition of its parameters, the development of algorithm and program code implementing the controller, is replaced by a direct synthesis of controller action program – the system of rules, i.e. the algorithm of its performance in the form of pseudo-code.

## 2 Cyber-physical systems controller synthesis problem

At present, a general formalized method of solving the problem of controllers synthesis in the production form is absent. One can only indicate some particular methods, which are based on the optimization procedure [5]. One of

\* Corresponding author: [grogachev@mail.ru](mailto:grogachev@mail.ru)

them is the automation of the production systems synthesis procedure through evolutionary computation as a stochastic method of global optimization. This approach will eventually allow to create technical devices that would be able to generate their own software for their control systems and upgrade them if necessary. Evolutionary algorithms are traditionally divided into genetic algorithms (GA) and genetic programming. GA are generally intended to find a solution for optimization problems in the form of a set of parameters.

On the base of this approach, software tools that automate the entire development process of control systems from setting objectives to modeling their work, including software testing, can be created. This approach is especially effective under conditions of extremum no uniqueness of the objective function  $J(x)$ . However, since GA are required to analyze a large number of alternative solutions, the calculation time would be long, which is a significant disadvantage of this method and can be an obstacle for its usage in the controllers performance algorithms synthesis in cyber-physical systems in real-time mode.

In order to overcome this disadvantage, it is suggested to combine the control actions calculating procedure with the control process. In such an application of GA before the calculations are completed every few epochs one should choose the solution that provides the greatest movement of control object towards the target compared to the other available control options and use it at a time interval as a controlling action. In artificial intelligence, this approach corresponds to the online variant of the intelligent agent performance, whereas the previous version of the single computation of controlling action before the beginning of the controlling process is the offline variant. The term "operating in the online mode" is used in relation to the algorithms that are to process data inputs as they become available, rather than waiting until the entire set of data inputs becomes available. Such an intelligent agent performs in accordance with the calculations and actions shift method: firstly, it takes action, and then monitors the environment and computes the next action. In the theory of algorithms, this variant of solving a problem is known as an anytime algorithm - iterative computational algorithm, which is able to produce the best solution for a given moment at any time. When applying the anytime algorithm and the evolutionary method together one should choose the best solution every few epochs before the calculations are completed. This solution is used for some time as the current controlling action.

### 3 Example of program controller for mobile cyber-physical system

As an example of the suggested approach, we consider the problem of synthesis by evolutionary computation of the real time of the rule of mobile cyber-physical system

control. Mobility is achieved with the help of a wheeled robot in the form of the Dubins car. A kinematic model of the wheeled robot is a system of nonlinear ordinary differential equations of the form

$$[\dot{x}, \dot{y}, \dot{\varphi}]^T = [v \cdot \cos(\varphi), v \cdot \sin(\varphi), u]^T \quad (1)$$

where the state variables are two location coordinates  $x$  and  $y$  of the object on the plane and  $\varphi$  the angle of the velocity vector direction. Velocity vector magnitude  $v=1$ . Scalar control  $u$  is limited in magnitude  $|u| \leq 1$  and determines the instantaneous rotation velocity of the velocity vector. Solve the problem of the quickest transfer of the Dubins car from a point  $[-4, 0, -\pi]$  to a small neighborhood of the point with coordinates  $x=0, y=0$ . Since the third coordinate (the angle  $\varphi$ ) at the end point is not set, this is the problem with a movable end of the trajectory. In addition, not only will we take into account the time the control object needs move from the initial point to the final, but also the time during which the computation of the control law is being made. Since the known exact analytical solution for the time-optimal problem [6] involves fixation of the trajectory end and does not take into account the control computation time, we will solve the problem numerically. We will use GA as a method of computing the control law. The software is Global Optimization Toolbox, CPU Intel® Core™ 2 Duo 2.2 GHz. Limit the controlling action calculation time by a preselected value. In the first case, firstly, we will fully define a production system of a software controller and then we will use it for controlling purposes. We assume that if the controlling strategy is chosen correctly, the Dubins car will reach the desired point in eight seconds and that the allowable frequency of the controlling action change equals  $1 \text{ s}^{-1}$ . Then the controller production system takes the form: "if  $t \in [i-1, i)$  then  $u(t) = u_i$ ",  $i = 1, 2, \dots, 8$ .

When carrying out the first series of 15 computational experiments controlling action computation time was limited by 16 sec, which corresponds to 3 working epochs of GA in the specified computing environment. After 16 sec the object motion is under control of the best of the found controller work programs. Thus, the total time from receiving the problem to the end of the movement is 24 sec. As it is shown in Figure 1, in none of the 15 experiments the neighborhood of the point with coordinates  $x=0, y=0$  was not reached in 40 sec in consequence of the poor quality of the derived solutions.

The second series of experiments was carried out with controlling action computation time limited by 64 sec (12 GA working epochs). In this case (Figure 2) one third of the trajectory ended in neighborhood of the given point, but the total time for solving the problem was 72 seconds.

Change the tactic. Initially, under the limit computation time of 1 to define the parameter using the GA so that the rule "if  $t \in [0, 1)$  then  $u(t) = u_1$ " provides the object (1) at a minimum compared to all alternatives resulting deviation from the specified conditions  $x=0, y=0$  the desired final

position. Then, using this rule to control for the first second, we seek out the same arguments parameter  $u_2$  simultaneously. Repeat these steps for the first 7 sec of movement. The total time from the receipt of the task before the end of the movement will be only 9 sec. As in Figure 3, in this embodiment, the search failed to achieve guaranteed a small neighbourhood of a point  $x=0, y=0$ .

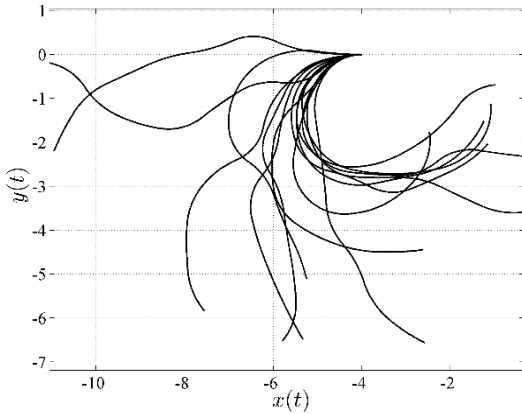


Fig. 1. Dubins car trajectories (computational time 16 sec)

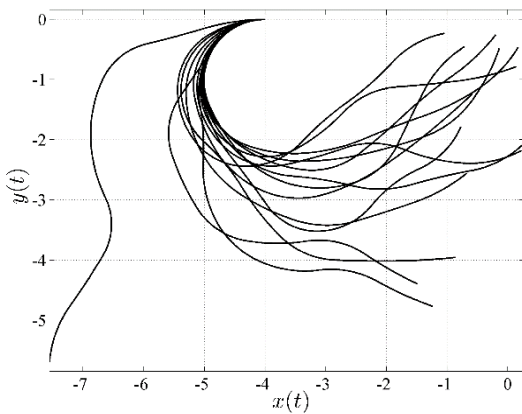


Fig. 2. Dubins car trajectories (computational time 64 sec)

The achieved effect of improving the solution quality while significantly reducing the total solution time can be explained not only by combining in time the processes of determining the control law and movement under the control, but also by decreasing the computational complexity of the problem. Indeed, at each  $i$ step only one parameter  $u_i$  instead of eight is to be determined. Thus a major disadvantage of evolutionary computing, which is the excessive computation time necessary for the analysis of a large number of alternative variants, can be overcome. Therefore, the synthesis problem can be successfully solved in real-time scale by using evolutionary computation and genetic algorithm in particular. Moreover, in a homogeneous environment online variant is undoubtedly more advantageous than the offline variant.

In a heterogeneous environment, the situation is different. Let us assume that on the way of Dubins Car there is an area where movement is not desirable due to any reason (in the following figures this area is indicated as a grey hexagon). Now the online variant of computations on the basis of the plenitude of the knowledge about the environment in which the movement will occur demonstrates the advantage, because it allows the Dubins car to pass the restricted area quite successfully (Figures 4, 5). In contrast, the offline control of the Dubins car copes with simple obstacles, but fails to overcome more difficult ones. (Figures 6, 7).

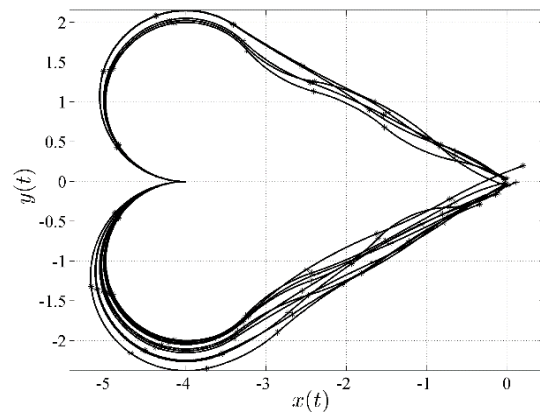


Fig. 3. Dubins car trajectories (anytime algorithm of controlling, full time 9 sec), marks are put through 1 sec

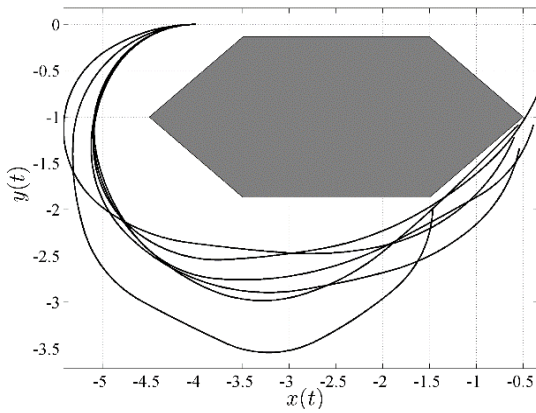
This fact has a rational explanation. Indeed, in the offline variant the rule system of controller performance is fully determined before the movement starts. Plenitude of the information both about the environment and about how all the movement stages impact on the final result is used, long-term planning is carried out. On the contrary, in the online variant of computations there is a short-term planning, each step is determined by the local target to minimize the remaining distance to the desired point. The available information is not about all the environment, but only about its nearest part. In this case, it is difficult to map a route of movement outside the restricted area.

## 4 Conclusions and recommendations

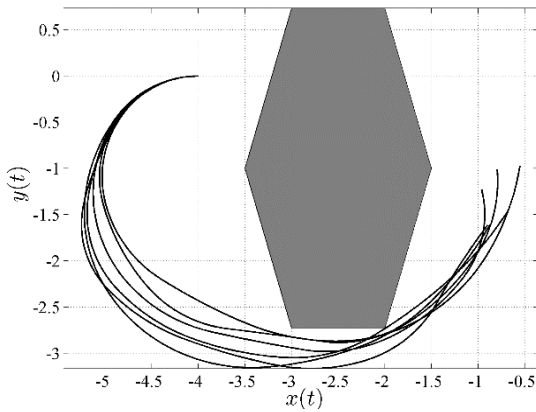
The examples considered demonstrate the possibility of using evolutionary computing in problems of the synthesis of real-time control systems for wheel mobile devices moving in homogeneous and in heterogeneous environments.

It was found that there is a possibility of improving the quality of solving the control problem with both a significant reduction in the total time by accomplishing essential computational procedures for determining the control law while implementing and a reduction of the problem computational complexity due to the use of successive operations of sequential search for each of the

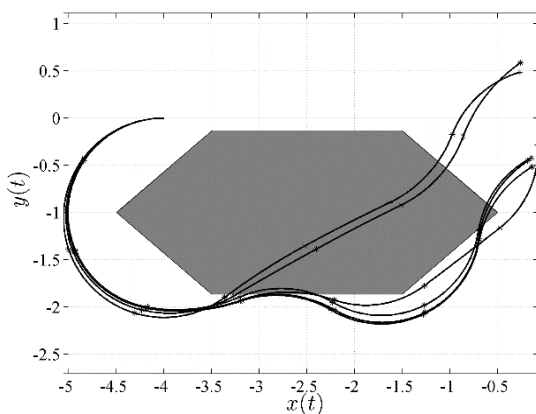
production system instead of the multivariable search algorithm.



**Fig. 4.** Dubins car trajectories in a heterogeneous environment (offline variant, computational time 64 sec, full time 72 sec)



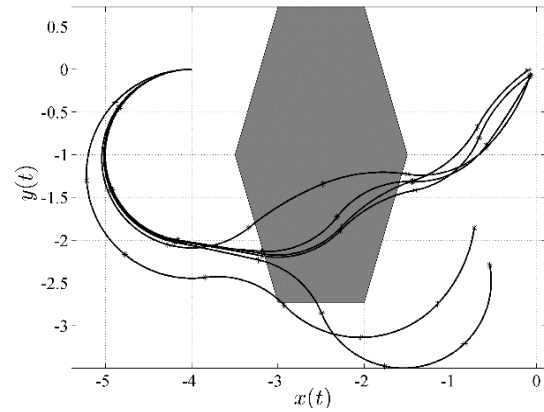
**Fig. 5.** Dubins car trajectories in a heterogeneous environment (offline variant, computational time 64 sec, full time 72 sec)



**Fig. 6.** Dubins car trajectories in a heterogeneous environment (online variant, full time 9 sec)

The main disadvantage of evolutionary computing, which is the excessive computation time necessary for the analysis of a large number of alternative variants, can be overcome in this way. In the absence of obstruction, one

can recommend the online calculation variant combined with the control process. This variant allows significantly reducing the computation time and improving the quality of the solutions found. In a heterogeneous environment, it is advisable to use the offline variant of pre-computation the complete solution.



**Fig. 7.** Dubins car trajectories in a heterogeneous environment (online variant, full time 9 sec)

This work was financially supported by Ministry of Education and Science of Russian Federation (basic part of government contract).

## References

1. *Cyber-Physical System. Driving force for innovation in mobility, healthy, energy and production.* (Springer, 2011)
2. *Living in networked world. Integrate research agenda Cyber-Physical Systems (agenda CPS)* (ACATECH (2015)
3. R. van. Kranenburg, *The Internet of Things. A critique of ambient technology and the all-seeing network of RFID* (Amsterdam, 2008)
4. G.N. Rogachev, Automatic control and computer sciences, **48**, 5 (2014)
5. G.N. Rogachev, Vest. SSTU. Ser: Tech. sci., **41** (2006)
6. L.E. Dubins, American Journal of Mathematics, **79**, 3 (1957)