IFIP Transactions A: Computer Science And Technology, v. A-52, p. 162-167, 1994.

Modeling motion simulation with DEDS

J. T. F. de Camargo, L. P. Magalhães and A. B. Raposo

Department of Computer Engineering and Industrial Automation, Faculty of Electrical Engineering, UNICAMP, CP 6101, CEP 13081-970, Campinas, SP, Brazil
e-mail: tarcisio; leopini; alberto@dca.fee.unicamp.br

Abstract

The computer simulation control problem can be splitted in two parts, namely a local control problem and a global control problem. The local control defines the "behavior" of each object, while the global control is responsible for the interactions among the objects. In this paper we propose an event-oriented scheme to solve the global control problem using concepts related to DEDS *(Discrete Event Dynamic System)* and ESM *(Extended State Machines).*

Keyword Codes: I.3.2; I.3.4; I.2.9
Keywords: Graphics Systems; Graphics Utilities; Robotics

1. INTRODUCTION

Computer simulation/animation is still a recent research field in computing science. Nevertheless, almost all work found in computer simulation literature treats the movement control problem only as a classical control problem. Indeed, without the use of mathematical tools such as differential calculus it would be hard to model the systems around us. On the other hand, many systems will require complex differential equations systems, what is a common fact. In addition, since there is a large number of systems (of all kind) there is no universal solution for any problem. In other words, a suitable approach for one type of system may not be the best for another one.

In order to solve motion simulation problems we split the simulation control problem into two parts: a local control block and a global control block. Suppose a system to be simulated with many objects. For each object we can assign a local control block that contains the "behavorial rules" for this object. The local control block suggests a mathematical structure (such as differential equations) that models the respective body and describes its behavior. For example, for a ball in a free fall, what is the influence of air resistance? Is the ball impact elastic when it collides with the ground?

In order to control the interactions among many objects the global control block suggests a logical approach. When an event occurs, how must the objects respond? Which objects are affected by the event?

This paper introduces the use of DEDS [1] to model interactions among objects in computer simulation. An object can react in different ways according to different extern stimuli. In this way, the occurrence of one event may cause the reaction of an object in a way that could be different if some other event had occurred.

An adequate approach for modeling DEDS is the Extended State Machine (ESM). ESM features are suitable to the development of DEDS for the purpose of simulation or animation. The next sections will discuss an example of a system that was modeled according to this philosophy.

2. MODELING A BIPEDAL STRUCTURE

Consider the bipedal "actor" shown in Figure 1. The figure presents an articulated human-like body composed by four articulated structures. When those structures are joined we have the bipedal figure. In order to simulate this articulated character we can consider both control levels, local and global. The local control is responsible for the effective displacement of each leg, that is, a controller of this kind must control the motion of a single leg from one place to another. There are several classical kinematic or dynamic techniques that can be applied to solve this local control problem. The solution we found for the local control problem is much like those presented in [2,3,4,5], and it can be extended to the solutions presented in [6,7,8] for a dynamic model.

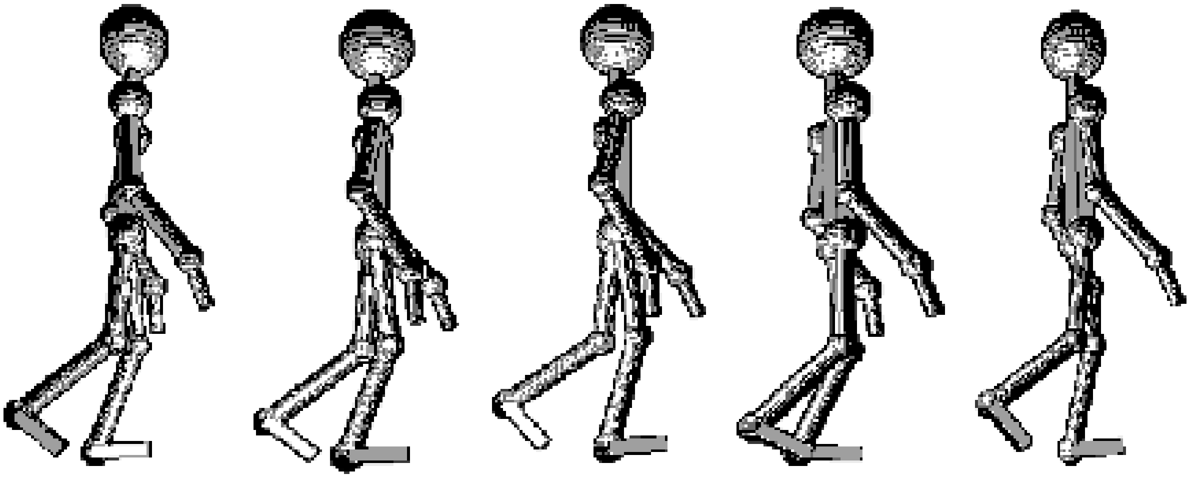

Figure 1: Frames generated for a bipedal structure simulation.

On the other hand, it is not enough to control only the motion of each leg individualy, since, for a bipedal structure, there must be coordination between the legs, otherwise we can not move the entire body. For example, both legs are not allowed to be out of the ground at the same time when the actor is walking. In this way, when dealing with the interaction between legs, we are facing a global control problem.

The solution for the global control problem can be found in DEDS formulation. As a way to represent discrete event systems we have the ESM syntax, which is a suitable tool for modeling this kind of simulation problem. Our global control approach is partially based on the model presented in [9], and the final solution that we propose uses the same tools as those presented in [1,10,11].

For this structure, a *gait pattern* describes a sequence of lifting and placing of the feet. The pattern repeats itself as the figure moves: each repetition of the sequence is called the gait cycle. The time (or number of frames) taken to complete a single gait cycle is the period P of the cycle.

During each gait cycle period any given leg will spend a percentage of that time on the ground. This fraction is called the duty factor of the leg. In our example, the duty factor may be used to distinguish between the walking and running gaits of bipeds. In addition, we can define the stroke as the distance traveled by the body while the leg is still on the ground.

We can see each leg as a basic ESM where, associated with each leg i, we have the following states:

$S_i \in$ set-type(x_i), the *support state.* We will call the time a leg spends on the ground its *support duration.* Its respective phase is the support state.

$T_i \in$ set-type(x_i), the *transfer state.* We will call the time a leg spends in the air its *transfer duration.* Its respective state is the transfer state.

We still have associated with each leg i the following events:

α_i, the leg is lifting;

β_i, the leg is grounding.

The flow of information between the controller and each leg is done through the communication channels:

m_l, send information from the left leg to the controller;

c_l, send information from the controller to the left leg;

m_r, send information from the right leg to the controller;

c_r, send information from the controller to the right leg.

As a basic control rule we have:

It is not allowed to both legs be in the transfer state at the same time when the body is walking.

For the left leg, L_l, we can build the following ESM:

ESM L_l:
$L_l = [(x_l), \emptyset, (m_l, c_l), (\alpha_l, \beta_l), A_l]$
set-type$(x_l) = (S_l, T_l)$

$$
\begin{aligned}
A_l = [\ & [(x_l, (T_l, \beta_l, \emptyset, S_l))]\ , \\
& [(x_l, (S_l, c_l?\alpha_l, \emptyset, T_l))]\ , \\
& [(x_l, (S_l, TRUE, m_l!x_l, S_l))]\ , \\
& [(x_l, (T_l, TRUE, m_l!x_l, T_l))]\ ]
\end{aligned}
$$

A similar ESM can be built for the right leg, L_r.

Once the plant composed by both legs have been designed[1], we will treat the global control problem under a simple point of view. In this case, the actor will move in a straight line indefinitely, and the controller may be represented as follows:

ESM Controller:
Controller = $[(x_c), (y_l, y_r), (m_l, m_r, c_l, c_r), (\alpha_l, \alpha_r), A_c]$
set-type(x_c) = (L_1, L_2, L_3, L_4)

$$
\begin{aligned}
A_c = [\ & [(x_c, (L_1, m_l?y_l, \emptyset, L_2))]\ , \\
& [(x_c, (L_2, y_l = T_l, \emptyset, L_1))]\ , \\
& [(x_c, (L_2, y_l = S_l, c_r!\alpha_r, L_3))]\ , \\
& [(x_c, (L_3, m_r?y_r, \emptyset, L_4))]\ , \\
& [(x_c, (L_4, y_r = T_r, \emptyset, L_3))]\ , \\
& [(x_c, (L_4, y_r = S_r, c_l!\alpha_l, L_1))]\ ]
\end{aligned}
$$

A pictorial representation for the set plant-controller is shown in Figure 2.

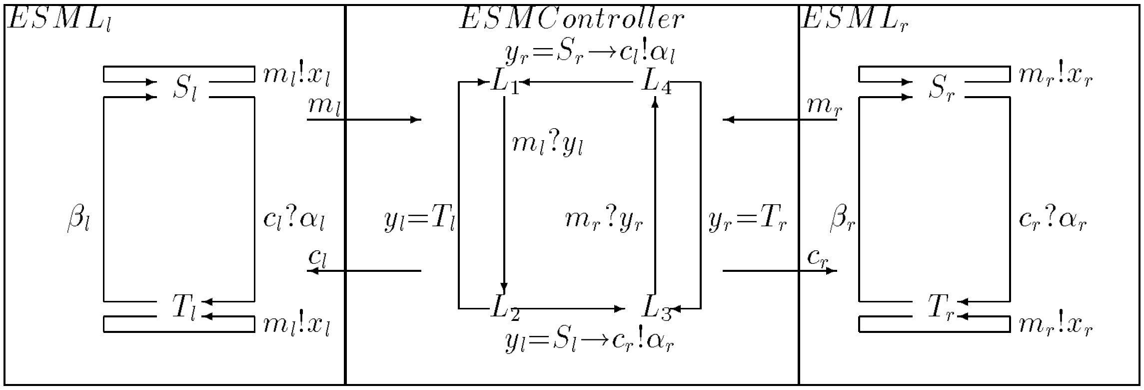

Figure 2: The “closed-loop” system plant || controller.

---

[1] The complete system must also consider the arms and the torso of the body. However, for the sake of simplicity, we will only treat the legs in this paper.

3. IMPLEMENTATION

Based on concepts presented in [12] we have implemented a system called TOOKIMA (*a TOOl KIt for scripting computer Modeled Animation*), that defines a set of tools for the algorithmic description of animations or simulations. This system is able to describe the kinematic/dynamic motion of computer modeled objects. Further details about the TOOKIMA system can be found in [13].

According to the modularity of ANSI-C Programming Language, we have built the ESMs of the bipedal object as "sub-routines" in the script of the animation. The "Global Control ESM" is responsible for the "management" of both legs and the timing control of the simulation sending information such as "lift the left leg".

This system is currently running in a Sun Sparc 2 Station. Figure 1 presents some images generated with the TOOKIMA system for the visual representation of the walking simulation.

4. CONCLUSIONS

We have presented in this paper the application of ESM in computer simulation. We have discussed an application, the bipedal structure, as a practical example. This system represents a relatively complex application and the method is general enough for this kind of simulation. We claim that the use of local and global controllers are suitable for the implementation of a wide range of simulation models. Modeling the global and local controllers as ESMs allow us to split an initial complex problem into several smaller problems.

In addition, the global control structure let us free to think about the local control blocks. That is, no matter which local control technique we adopt, the global control block does not change. In this way, when solving the local control problem we can choose among several techniques such as a kinematic or a dynamic model.

REFERENCES

1. Yu-Chi Ho, *Dynamics of Discrete Event Systems*, Proceedings of the IEEE, v. 77, n. 1, pp. 3-6, January 1989.

2. J. J. Craig, *Introduction to Robotics: Mechanics and Control*, 2^{nd} ed., Addison-Wesley Publishing Company, 1989.

3. A. A. Goldenberg, B. Benhabib and R. G. Fenton, *A Complete Generalized Solution to the Inverse Kinematics of Robots*, IEEE Journal of Robotics and Automation, v. RA-1, n. 2, pp. 14-20, March 1985.

4. J. U. Korein and N. I. Badler, *Techniques for Generating the Goal-Directed Motion of Articulated Structures*, IEEE Computer Graphics and Applications, pp. 71-81, November 1982.

5. B. C. McInnis and C. F. Liu, *Kinematics and Dynamics in Robotics: A Tutorial*

Based Upon Classical Concepts of Vectorial Mechanics, IEEE Journal of Robotics and Automation, v. RA-2, n. 4, pp. 181-187, December 1986.

6. W. W. Armstrong and M. W. Green, *The dynamics of articulated rigid bodies for purposes of animation*, The Visual Computer, v. 1, n. 4, pp. 231-240, December 1985.

7. R. Barzel and A. H. Barr, *A Modeling System Based On Dynamic Constraints*, Computer Graphics, v. 22, v. 4, pp. 179-188, August 1988.

8. R. G. Morgan, *A Descentralizated Variable Structure Control Algorithm for Robotic Manipulators*, IEEE Journal of Robotics and Automation, v. RA-1, n. 1, pp. 57-65, March 1985.

9. M. Girard and A. A. Maciejewski, *Computational Modeling for the Computer Animation of Legged Figures*, Computer Graphics, v. 19, n. 3, pp. 263-270, July 1985.

10. G. Cohen, P. Moller, J. Quadrat and M. Viot, *Algebraic Tools for the Performance Evaluation of Discrete Event Systems*, Proceedings of the IEEE, v. 77, n. 1, pp. 39-57, January 1989.

11. J. S. Ostroff, *Temporal Logic for Real-Time Systems*, Research Studies Press Ltd., John Wiley and Sons Inc., 1989.

12. N. M. Thalmann and D. Thalmann, *Computer Animation - Theory and Practice*; Springer-Verlag; 1985.

13. M. S. Hounsell and L. P. Magalhães, *TOOKIMA - Animação Cinemática (TOOKIMA - Kinematic Animation)*, SIBGRAPI 92 (Brazilian Computer Graphics and Image Processing Conference), November 1992.