Yugoslav Journal of Operations Research
6 (1996), Number 2, 257-265

PRODUCTION PROCESS SIMULATION MODEL MONITORING USING COMPUTER ANIMATION

Božidar RADENKOVIĆ, Aleksandar MARKOVIĆ, Zoran RADOJIČIĆ

Laboratory for Simulation,
Faculty of Organizational Sciences, University of Belgrade,
Jove Ilića 154, 11000 Belgrade, Yugoslavia

Abstract: The simulation process can be applied to behaviour analysis of complex systems, whose physical modelling in a laboratory environment would be unattainable, expensive or time consuming. In addition to the classical approach to simulation model monitoring using numerical, graphical and statistical results, experience that has shown the addition of a visual presentation using animation significantly improves the simulation process as a whole. Special attention in this paper is given to the modelling of production processes as parts of modern production systems. This paper also presents an example of production process simulation model animation, using commercial animation software (Proof Animation). Experience based on the animated simulation models shows that simulation model development significantly speeds up, and model verification becomes much easier. Also, animation of the simulation is very useful in explaining the simulation to managers who may not be as technically skilled as those who have performed the analysis. This approach enables easier development, organisation and planning of the production processes under investigation.

Keywords: Computer simulation, computer animation, production processes

1. INTRODUCTION

Computer software to model dynamic systems has been existing for more than three decades. During this period, computer modelling and simulation have been enhanced by powerful simulation languages, statistical techniques, and graphical modelling techniques. As the field of simulation language design has matured, the level of sophistication and performance expected by simulation analysts has increased. Simulation users are demanding sophisticated graphical, statistical, and other tools to assist in the analysis of complex systems.

Until the early 1980's, computer graphics technology was expensive and available only on large mainframe computers. Nowadays, it is possible to buy small, powerful and somewhat low-priced personal computers that have excellent graphics and animation facilities. Thus, computer graphics nowadays are replacing the more traditional printed report as a presentation medium. Since 1976, when Hurrion published his thesis on Visual Interactive Simulation (VIS), the use of visual iconic pictures in simulation, or animation, as it is called in the United States, has dominated simulation. The use of animated computer graphics now forms an integral part of nearly every simulation project, and most simulation software provide some animation component or interface towards existing animation tools [2].

Computer graphics and animation can assist in all phases of a simulation project, from its initial problem specification to the final project of implementation. A very important aspect of using animation is the possibility of watching simulation dynamics in a simulation run phase. Instead of relying on printed results at the end of a simulation, it is now possible to observe the dynamics of a model unfold in the form of a video film [3].

2. GRAPHICS AND ANIMATION IN DISCRETE SIMULATION

Discrete-event simulation (DES) deals with the modelling of systems that can be represented by a series of events. The simulation describes each discrete event, moving from one to the next as time progresses. The activity, as the second important DES concept, can be defined as a set of operations which transform the conditions of one or more system entities. Condition changes are shown in initial and final stages of the activity. The process as the third important DES concept represents a series of simultaneous, logically connected events describing these phenomena from their initialisation all the way to the end (Fig. 1). The modelled systems are dynamic and, almost invariably, stochastic [5].

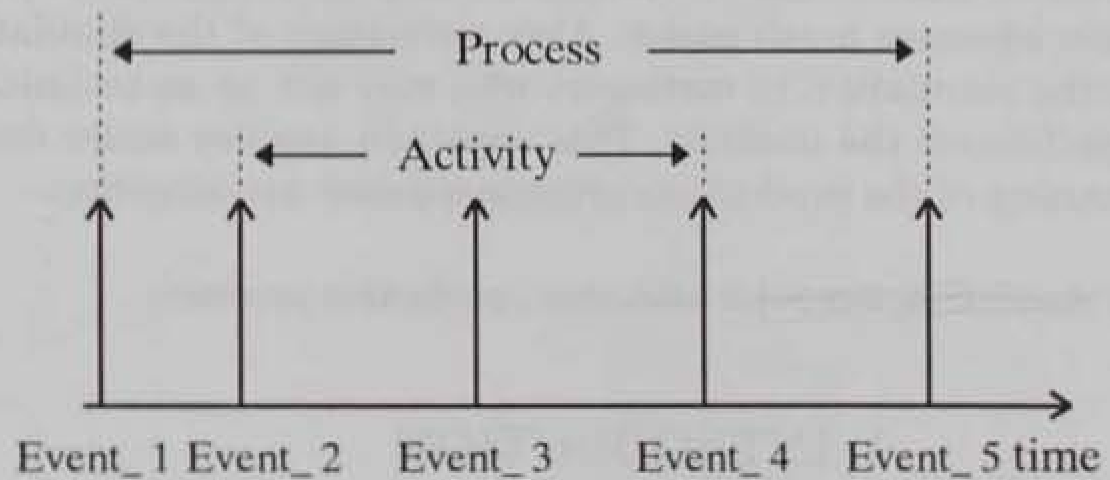

Figure 1. Relation between event, activity and process

For the animation of systems defined in this way, the creation of a direct relation between event concepts, activities and processes, on the one hand, and corresponding concepts of computer animation on the other, is significantly important.

The concept of linking graphics with discrete simulation began in the mid-1970s and became a standard practice by the mid-1980s. Different methodologies in the use of simulation graphics have emerged from the United States and the UK. The technique adopted in North America is represented by the usage of high-resolution colour graphics terminals in order to animate the dynamics of a simulation, but not let the user interact with the model while it is running. This approach is often called post-processing graphics. The method developed in the UK uses graphics to animate a model as it is running, but also allows the user to interact during the execution of the model, and is known as visual interactive simulation [4].

There are basically two types of graphics animation used for discrete event simulation. Both methods are obliged by current graphics hardware technology. The two basic representational methods can be summarised as [3]:

1. *Character graphics*. Most common micro-computer screens are text-based. Text can be placed on the screen at locations defined by a grid of 24 rows by 80 columns. The repeated drawing and erasing of text at slightly different locations will give the impression of elements moving. This simple technique can give a surprisingly realistic view of a simulation model, especially if colour is used to code elements in the model.

2. *High resolution bit-mapped graphics*. The technology for high-resolution bit-mapped graphics has been improving dramatically, and becoming increasingly cost-effective. The display area of the screen is divided into a grid which usually consists of 640 rows by 480 columns. The intersection of a row and a column is known as a pixel, and it can be displayed by one of many different colours.

Lately, vector graphics have become a standard graphics type of animation in the simulation of discrete events and as graphics hardware becomes less costly and more cost-effective it is certain that animation systems for simulation will begin to use three-dimensional models of representation on a routine basis.

3. POST-PROCESSING GRAPHICS

Post-processing animation is one of the most-used techniques in animating discrete-event models. This approach means that the animator is driven by output from the simulation logic. As Fig. 2 shows, simulation model development and picture development represent two separate exercises [1]. The developer or analyst uses a software tool to design the picture, usually employing a high quality graphics terminal. In designing the static background, he must select icons to represent the entities (perhaps designing new ones as required) and position the queues or activities (objects) in the picture. The simulation program is written as if there were no visual outputs.

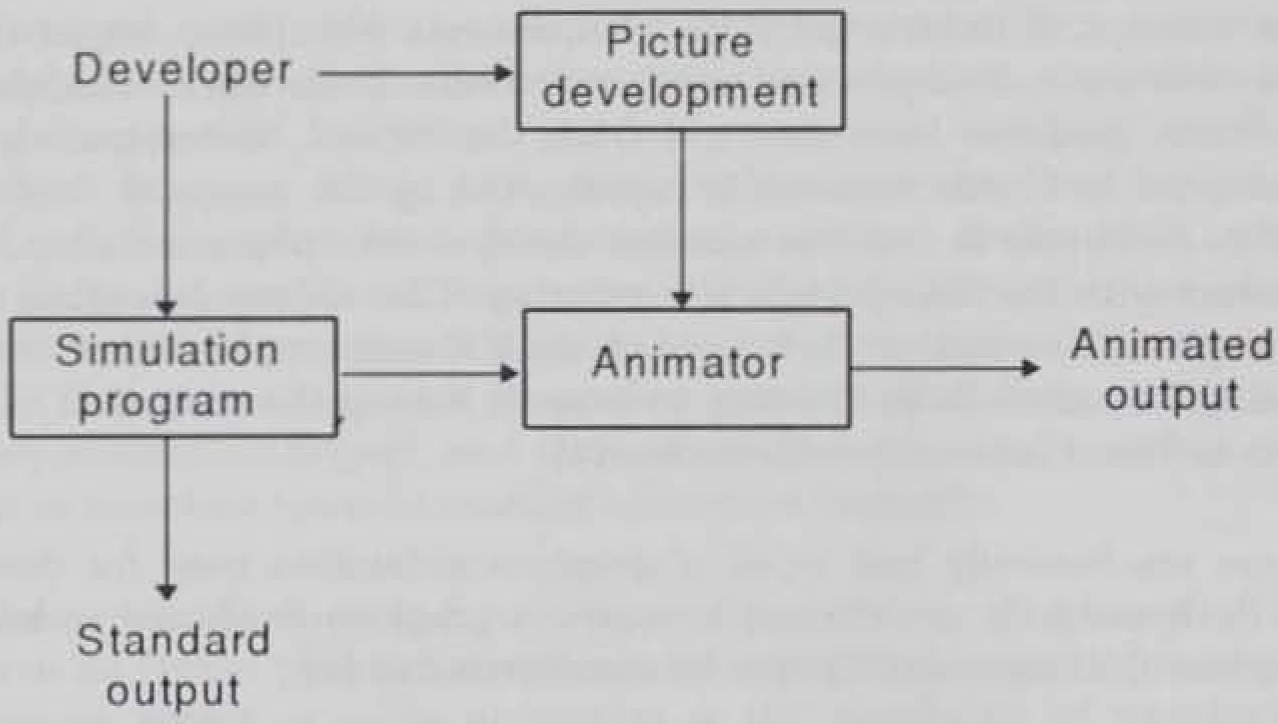

Figure 2. Interaction between developer and animator in post-processing animation

When the simulation runs, it produces formatted output which is decoded by the animator. The output produced by the running simulation, coupled with the information for the separately designed display, enables the animator to identify the position of the object on the screen and move the icon representing the entity from one place to the other.

This approach has some advantages that can be summarised as:

- Post-processing allows full exploitation of the processing power and graphics hardware, maximising the performance.
- It allows animations to run on a computer other than the one used to run simulation.
- Post-processing allows more detailed study of long-term system behaviour through rapid animation execution, and fast-forwarding.
- Finally, and perhaps more importantly, animations can be based on models developed using a wide variety of simulation software - or even using a non-simulation programming language or other software environment.

4. EXAMPLE

In discrete-event simulation, a computer animation is mostly applied to observe the dynamics of the following systems:

- Transportation means in industries
- Massive transportation means in cities
- Highway transportation
- Flexible manufacturing system
- Distribution centre
- Just-in-time manufacturing
- Computer operating system performance
- Large scale on-line computerised transaction processing etc.

An example shall be discussed here of using animation in the simulation of production processes in wood industries concerning to the production of laminated wood rafters which are products in modern building.

4.1 Description of a real system

The laminated wood production process consists of a series of operations related technologically and in time which create a final product (laminated wood rafter) out of the initial raw material. The basic technology creating the simulation model includes the following operations [6]:

1. delivering of raw materials
2. cutting, machine processing
3. ripping, machine processing
4. nibbling, machine processing
5. creating the laminated wood rafter, craftsmen hand-work

4.2 Implementation on simulation language GPSS/H

For the above described system, a simulation model was developed and implemented in GPSS/H simulation language which was consequently enlarged by corresponding commands - macro definitions (Fig. 3.a) and macro calls (Fig. 3.b) thus enabling the creation of a special data file which is necessary for model animation.

When the simulation is run, GPSS/H generates a data file with an ATF extension used by the animation software, in this case the PROOF Animation.

4.3 Animation with PROOF Animation software

Proof Animation is a family of general purpose systems animation software for IBM PC AT or compatible computers. It is a vector-based, file-driven, and post-processing animation system [7] which consists of:

- A CAD-like drawing capability for creating layouts and defining shapes and paths,
- A set of animation commands and the ability to process programme-generated sequences of these commands.
- A set of presentation commands and the ability to process hand-tailored sequences of these commands.

Proof animation does not attempt to create photorealistic sequences or frame-by-frame effects. It is used to create a precisely described, smoothly moving description of a complex system whose states are constantly changing.

Proof Animation is independent of any specific simulation or programming language and its open command set allows a wide variety of software to be used.

```
          :
*
* Proof Animation Macro Definitions
*
PUTTIME STARTMACRO
     TEST NE AC1,&ATFTIME,*+3
     BPUTPIC FILE=ATF,(AC1)
TIME *.*
     BLET &ATFTIME=AC1
     ENDMACRO
*
CREATE  STARTMACRO
PUTTIME MACRO
     BPUTPIC FILE=ATF,(#B)
Create #A *
     ENDMACRO
*
DESTROY STARTMACRO
PUTTIME MACRO
     BPUTPIC FILE=ATF,(#A)
Destroy *
     ENDMACRO
*
PLACEON STARTMACRO
PUTTIME MACRO
     BPUTPIC FILE=ATF,(#A)
Place * On #B
     ENDMACRO
          :
```

(a)

```
          :
CREATE MACRO   daska1,XID1
  PLACEON MACRO   XID1,CP5/
     BLET    &MFS2=&MFS2-1
  WRITE   MACRO   Mfs2,&MFS2
     ADVANCE  3,1
     QUEUE    RPARAC
     SEIZE    PARAC
  PLACEON MACRO   XID1,CP6
  SETCOLOR MACRO   XID1,5
     DEPART   RPARAC
     ADVANCE  PH(DD)*2/17+3
     RELEASE  PARAC
     BLET    &DPAR=&DPAR+1
  PLACEON MACRO   XID1,CP7
     ADVANCE  4,1
     BLET    &MFS3=&MFS3+1
  WRITE   MACRO   Mfs3,&MFS3
  DESTROY MACRO   XID1
          :
          :
```

(b)

Figure 3. Part of the GPSS/H code with corresponding macro definitions (a) and macro calls (b)

Animation preparation consists of at least two stages:

1. Production of layout for the animation
2. Creation of the run animation file

Except the drawing of the layout itself (fixed objects), *the production of a layout* where the animation is run consists of the creation of all object classes participating in animation (Fig. 4), as well as the definition of paths used for the object movements (Fig. 5). The description of the layout is saved in the *layout* file.

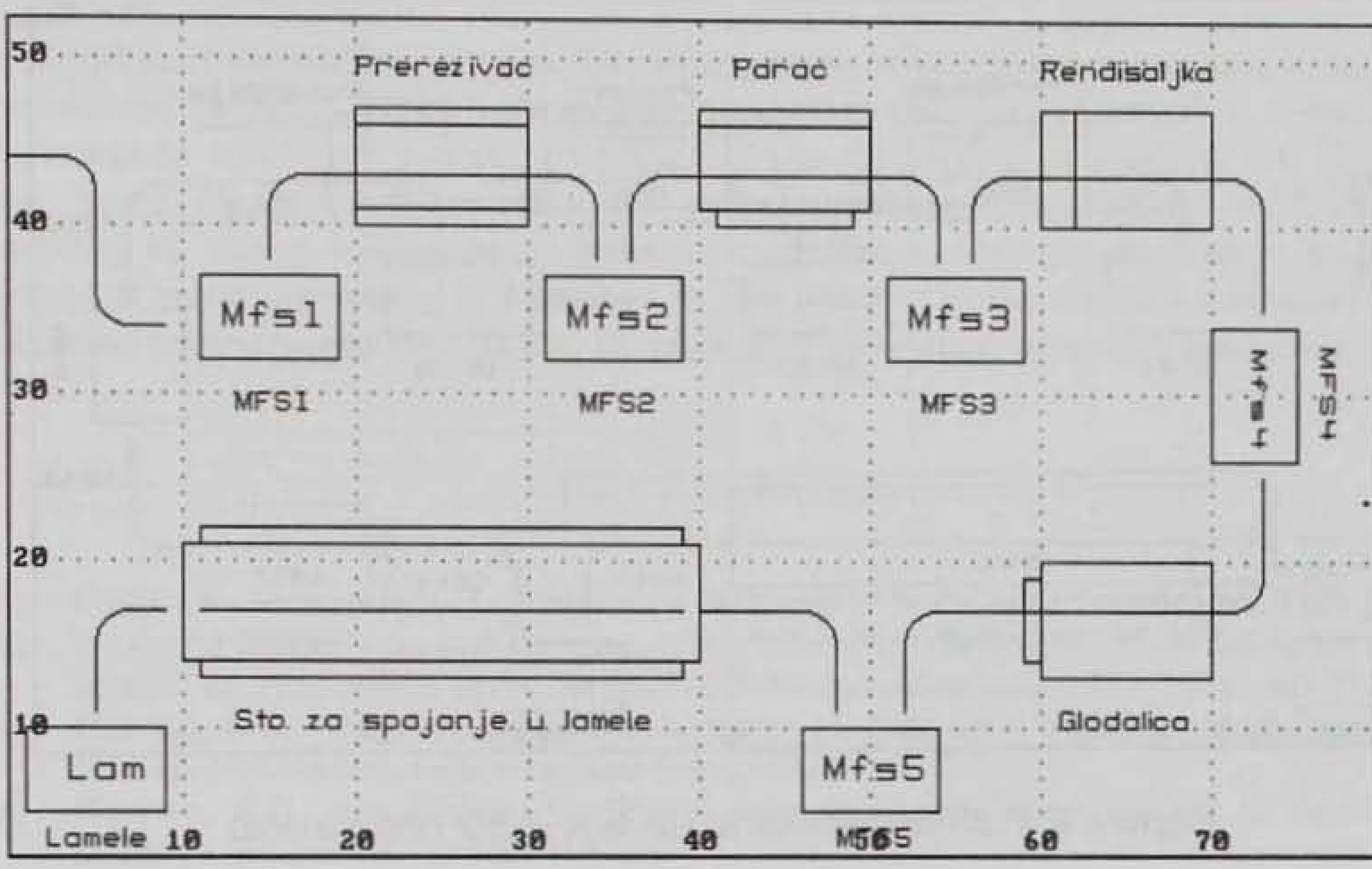

Figure 4. Animation layout with fixed objects and paths (Draw mode)

The creation of a data file with a description of the animation run is in most cases the output from one of the simulation languages (in this case GPSS/H) and includes commands to create and deleting objects, change the places of objects, place objects on a desired path, etc.

An animation detail from the production process of laminated wood rafters is presented on the Fig. 6. The present condition of this process can be noted from the Figure, including the utilisation of existing machinery, the conditions of inter-operational storage, etc. Observing the animation as a whole it is possible to follow the dynamics of the production process flow and control the quality of the operation itself. It is also possible to spot critical points in the process as well as values which represent crucial parameters in the system's operation.

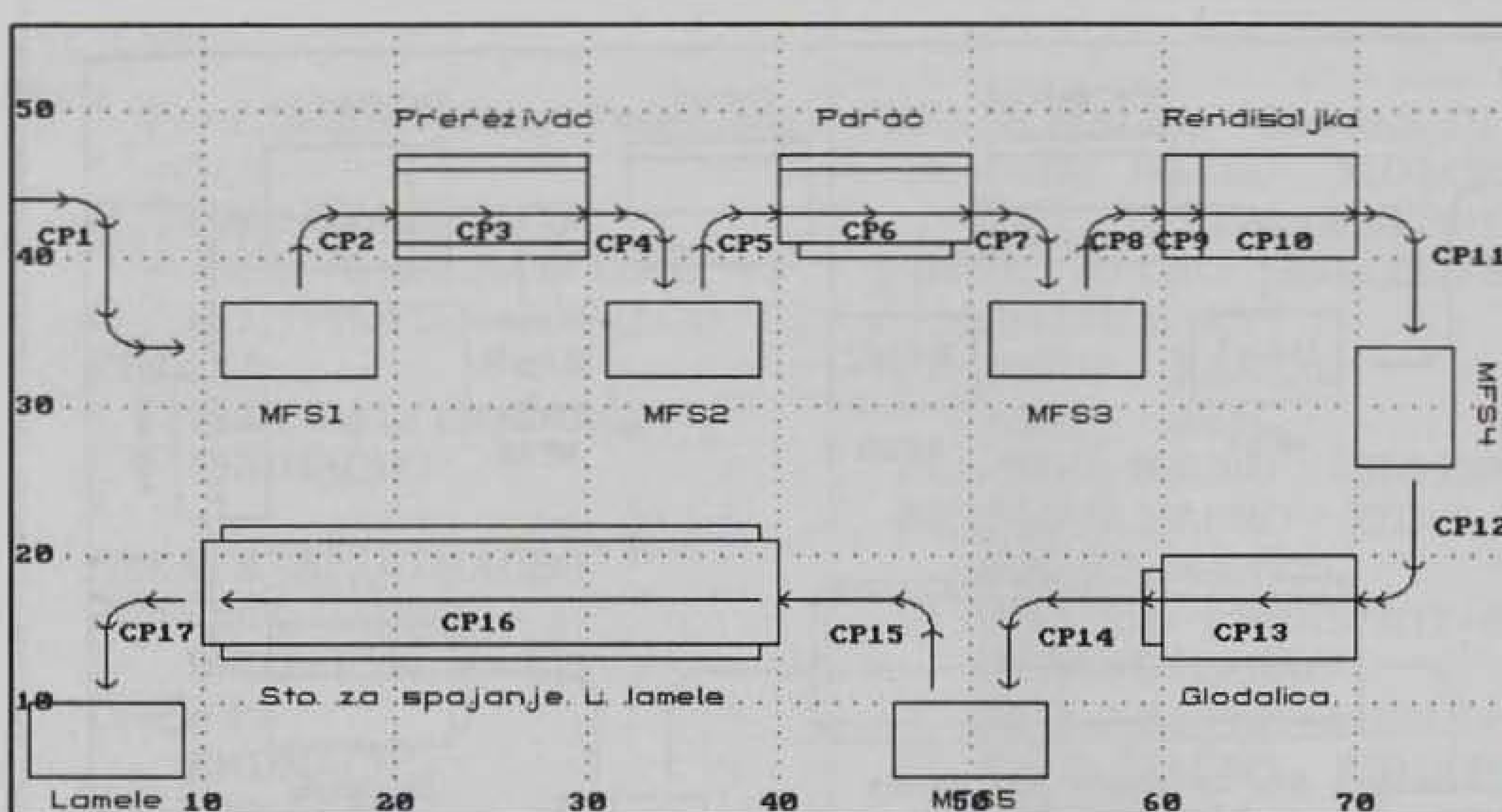

Figure 5. Path definitions in the layout file (Path mode)

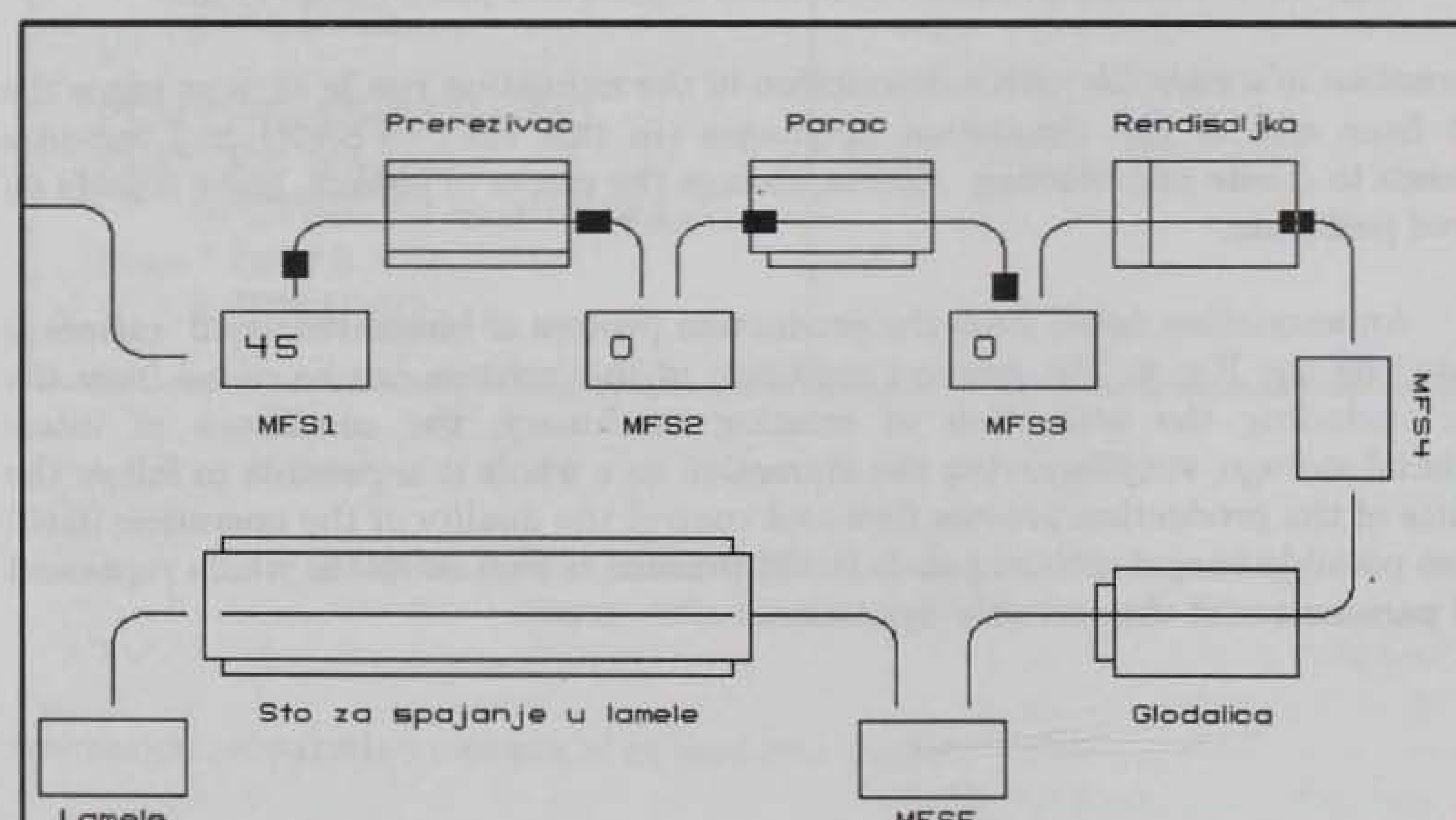

Figure 6. An animation detail

5. CONCLUSION

Graphics animation is one of the most efficient ways to present and monit simulation dynamics. A simulation model observed not only by means of standard simulation results, but also visually, becomes more clear, comprehensive and approachable to eventual users. For analysts it enables a more simplified analysis of the process and avoiding error. Thus, this method has become a necessary component of any significant simulation project. The importance of computer animation in DES can be measured by the fact that most available software today is supported by a certain set of commands which are a direct function of the simulation animation. On the other hand, some of this software has its own incorporated animation modules. The possibility of using animation in behaviour control in a single production process in the field of wood processing is described in this paper. For the implementation PROOF Animation software was chosen and its main characteristics were also described.

REFERENCES

[1] Davies, R., and O'Keefe, R., *Simulation Modeling with Pascal,* Prentice Hall, 1989.

[2] Fiddy, E., Bright, J.G., and Hurrion, R.D., "SEE-WHY: Intreractive simulation on the screen", in: *Proceedings of the Institute of Mechanical Engineers,* 293 (1981) 167-172.

[3] Hurrion, R.D., "Graphics and interaction", in:, M. Pidd (ed.), *Computer Modelling for Discrete Simulation,* John Wiley and Sons, 1989.

[4] Hurrion, R.D., and Secker, R., "Visual interactive simulation, an aid to decision making". *Omega*, 6(5), (1978) 419-426.

[5] Radenković, B., and Marković, A., *Computer Simulation and Simulation Languages*, Faculty of Organizational Sciences, Belgrade, 1992, (in Serbian).

[6] Radenković, B., Marković, A., and Vulović, M., "Simulation study of laminated wood rifter production process", in: *Proceedings of YU INFO '95,* (in Serbian).

[7] Wolverine Software Corporation, Using Proof Animation, Tutorial and Reference Guide, USA, 1992.