ANNALS of the ORADEA UNIVERSITY.
Fascicle of Management and Technological Engineering, Volume VII (XVII), 2008

# SIZING IN DYNAMIC RUNNING, BY SIMULATION OF THE LOGISTIC AND CONTROL SUB-SYSTEM

**Gavrilă CALEFARIU[1], Victoria TOMA[1], Flavius SÂRBU[1]**
[1]University Transilvania of Braşov, Department of Economic Engineering and Production Systems
e-mail: gcalefariu@unitbv.ro, vtoma@unitbv.ro, sflavius@unitbv.ro

**Keywords:** simulation, dynamic running, sizing

**Abstract:** In the article the next state equations of the main subsystems composing the manufacturing system, are presented. It is find out that with this regard modeling with the aim of simulation of the manufacturing subsystem leads to achieve of procedural models.

Models for the processing, logistic and controlling subsystem, by making precise the according state sizes and their evolution are presented.

## 1. GENERALS

The most important difference between a models used in simulating, towards an analytical model used in designing, is the fact that the first has to be more precise as the second one. Precision results either from the type of formalism used, or by the level of detailing the states. This observation is valid also for analogical simulation models [1]. An other major difference between a designing model and a simulating one consists in the fact that the second one is easier used, not being formulated as an optimization issue, but only as objectives and constraints are included in its own structure.

The simulating model may be used for validating a designing model, for finding out some parameters of the system (situation as it becomes in a way also designing model) or for both of the above aims.

The scheme used for validating a designing model is represented in figure 1.

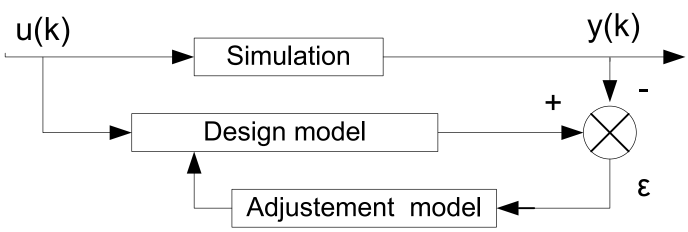


***Fig.1. Validating scheme of the designing model***

The use of simulating model in parameter determination may be achieved in two manners:

a. by using a model building-up technique, so that after simulation all needed parameter shall be directly obtained;

b. by a repeated simulating - correcting sequence, according to the scheme in figure 2.

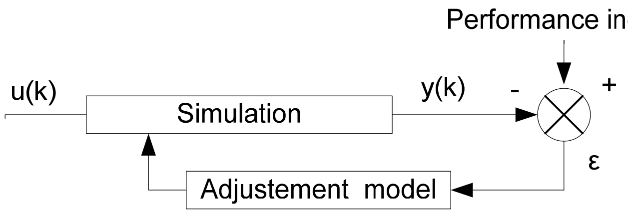


***Fig.2 The use of simulating model by repeated sequence***

Further on the creation of a simulating model is followed, to be used for all the three aims mentioned above.

Because sizing of the sub-system is carried out in stationary regime, being seen as an ideal (wanted) objective for the sub-system to be processed, it would be completely unreal to formulate such a requisite also for the other subsystems. The price paid often for providing stationary regime of the sub-system to be processed is that of an intensive dynamic regime of the other subsystems.

For being credible, from here out, two basic requirements for the simulating model are resulting: to be integral, including all essential subsystems and to be dynamic.

The next state equations being on the grounds of the general dynamic model construction are the dynamic equations of the general model [2]

$$x(k+1) = Ax(k) + Bu(k) + f(x(k), u(k)) \quad (1)$$

and of the subsystem model [2]:

$$x_i(k+1) = A_i x_i(k) + B_i u_i(k) + v_i(k) + f_i(x_i, u_i, v_i) \quad (2)$$

Things are simplified by the absence of the objective functions and of constraints, but even this fact allows detailing.

## 2. SETTING PARAMETERS AND COORDINATING INTERACTIONS

For setting parameters, the case of a system consisting of *J* technological devices, *r* individual transport systems, a store for enter half-products and a store for outcome final products. Each device includes an entering stock number *s* and the same stock number for the outcome of final products. These represent compensating elements of the system’s lack of synchronism and, meanwhile, a chance of obtaining a steady-state regime for the subsystem to be processed.

Among the above mentioned components, in figure 3, only two machine-tools *A* and *B*, two conveying-transferring systems $x_1$ and $x_2$ , (*A*, *B*, $x_1$, $x_2$, being characteristic parameters) an enter store *DJ*, consisting of *J* subsystems (stocking units), $i_j$, $j=1, 2, \ldots, J$ and an outcome store *DE*, consisting of *E* subsystems $e_j$ , $j=1, 2, \ldots, E$, have been selected.

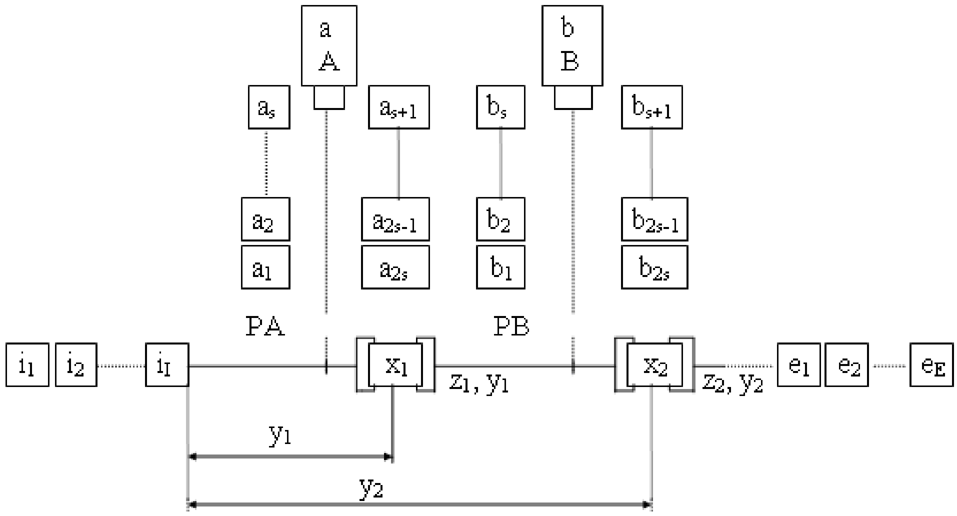


*Fig.3. Processing subsystem*

## 2.1. Logistic subsystem

The logistic subsystem consists in this case of the individual transport subsystems and the local stocking units of each device. The most important role of transport systems is to convey in space and those of the stocking units, is to convey in time. From the technical-economical point of view appears the need of separating the two components: conveying in space is expensive, because of the complexity of the transport systems, conveying in time (waiting) is reasonable from opposite reasons. But, there is yet an other difference also from the point of view of featuring this two aspects, as parameters are conferred to them, i.e.:

- transport (convey) in space has to be featured as an integrating process;
- transport (convey) in time is enough to be featured as a proportional process.

Having on its bas this requisite, the number of parameters conferred to each subsystem (or element) shall be equal to the number of conveying movements in space and in time, it is carrying out and which are presenting differences on a systemic scale.

For the above reasons it is considered that the spatial convey which deserves to be modeled (detailed) executes only the individual transport systems form a machine to the other and between the stores and the taking in delivery process of half-products is a secondary one and may be modeled also by a proportional type parameter.

Parameters assigned to the first transport system are as follows [2]:

- $x_1(k)$ – of proportional type features the type of the item being on the individual transport system. If this hasn't item, $x_1(k)=0$. If the system conveys two (or more) items, two (or more) parameters shall be assigned to it, out of which one for each conveying station.
- $z_1(k)$ – of proportional type, features the virtual event of taking over an item by the individual transport system;
- $y_1(k)$ – of integral type, features the position of the individual transport system towards a reference point of the conveying track.

Possible values for these sizes are:

$$x_1(k)=\begin{cases}\text{code of the conveyed item, if it conveys an item;}\\ 0\text{, contrary;}\end{cases}$$

$$z_1(k)=\begin{cases}1\text{, if the need of taking over of an item appeared:}\\ 0\text{, contrary;}\end{cases}$$

$y_1(k)$ – pertains to the multitude of positions where track of the individual transport system was digitized.

The dynamic evolution of these sizes is shown by the relations:

$$x_1(k+1)=u_1(k)+f_1(i_l,\ a_1,\ \ldots,\ a_{2s},\ b_1,\ \ldots,\ b_{2s},\ \ldots,\ e_1)$$
$$z_1(k+1)=u_2(k9+f_2(x_1,\ i_l,\ a_1,\ \ldots,\ a_s,\ a,\ a_{2s},\ b_1,\ \ldots,\ b_s,\ b,\ b_{2s},\ \ldots,\ e_1) \qquad (3)$$
$$y_1(k+1)=y_1(k)+Tu_3(k)+f_3(x_1(k),\ z_1(k),\ y_1(k))$$

where:

- $u_1(k)$ – represents the order of taking over or ceding a half-product from one of the positions: $i_l,\ a_1,\ \ldots,\ e_1$;
- $u_2(k)$ – order of shifting to one of the positions having accomplished the taking over conditions;
- $u_3(k)$ – average shifting speed of the individual transport system. It may be positive, negative or zero, according to $x_1(k)$, $z_1(k)$ and $y_1(k)$;
- $T$ – time increment.

The expressions $f_1,\ f_2,\ f_3$, are procedural expressions made out of logical expressions, order relations and algebraic expressions and thus, the model expressed by them are of a procedural type:

*Stocker discharging* $a_{2s}$ :"If $x_1(k)=0$ and $z_1(k)=0$ and $a_{2s}\neq 0$, than,

- if $y_1(k)<PA$ it results $u_3(k+1)>0$
- if $y_1(k)>PA$ it results $u_3(k+1)<0$
- if $y_1(k)=PA$, it results: $u_3(k+1)=0$ and $u_1(k+1)=a_{2s}$ and $u_2(k+1)$; End."

*Stocker charging* $a_1$: "If $x_1(k)=17$ and half-product no. 17 is processable on device A or B or C or … , but the item number from $a_1 \ldots a_s$ is minim related to $b_1,\ \ldots,\ b_s,\ c_1,\ \ldots,\ c_s,\ \ldots$ , than:

- if $y_1(k)<PA$ it results $u_3(k+1)>0$
- if $y_1(k)>PA$ it results $u_3(k+1)<0$
- if $y_1(k)=PA$ it results $u_3(k+1)=0$ and $u_1(k+1)=0$ and $u_{a1}(k+1)017$ ; End."

Any similar situation is similar solved.

The writing of these expressions has been preferred to be done in this way, instead of the traditional logarithms, because presently there are computer programs and languages very well adapted for expressions of this type.

For stockers, the basic rule is that of advancing the item possessed towards the work station, as soon as possible.

The parameter assigned to the stocker $a_j$ , $j=1,\ 2,\ \ldots,\ s-1$, is of a proportional type and it will be:

$$a_j(k+1)=u_j(k)+f_j(a_j(k)\ ,\ a_{j+1}(k)) \qquad (4)$$

By procedures, the expression shall be: "If $a_j(k)\neq 0$ and $a_{j+1}(k)=0$ than $u_j(k+1)=0$ and $u_{j+1}(k+1)=a_j(k)$ ; End."

For $a_s$, the procedure is the same, but interaction is made to a :"If $a_s(k)\neq 0$ and $a=0$, than $u_s(k+1)=0$ and $a(k+1)=a_s(k)$ ; End."

Procedures for the outlet stores $(a_{s+1},\ \ldots,\ a_{2s})$ are the same.

## 2.2. Processing subsystem

To any component part of the processing subsystem (production equipment), two parameters are assigned: a proportional one, due to the half-product type (a, b, c …) and an integral one, due to the position within the operation. This detailing level is enough on the system scale. The rule of fixing the type and number of parameters is almost the same with that of the logistical subsystem, the only difference is given by the fact that it exists a slightly difference between the significance of the integral type parameters: at the machine-tool, the significance of this size is that of: “position within the operation”.

The timely evolution of the parameters of each machine will be of the type:

$$a(k+1)=u(k)+f(A(k), a_s(k), a_{s+1}(k)$$
$$A(k+1)=A(k)+TU(k)+F(A(k), a(k)) \quad (5)$$

Procedural models due to the machine-tool are:

- Half-product supply: “If a(k)=0 and $a_s(k)\neq0$, than: u(k+1)=$a_s(k)$ and U(k+1)=$V_{op}$ ; End.”

$V_{op}$ – represents the operation’s carrying out speed. It may be the same for all half-products or different;

- The machine-tool processes : “If $a(k)\neq0$ and

if A(k) < LA , it results U(k+1)=$V_{op}$

if A(k) = LA , it results U(k+1)=0; End.”

- Delivering half-product (or item) : “If $a(k)\neq0$ and U(k)=0 and $a_{s+1}(k)=0$, than a(k)=0 and A(k)=0 and $U_{s+1}(k+1)$=a(k) ; End.”

The simulating question by input and output storing procedures is not different to that of local stockers.

## 2.3. Controlling subsystem

The structure of the controlling subsystem depends on the general concept of the control organization. This may be carried out directly on the machine-tool, or out of the machine-tool, on special controlling machines. If on machine-tool, control may be active (while processing) or passive (at the end of phases and operation). The controlling manner is usually decided once with the processing technology variant, but it may be decided also by systemic considerations, giving to the control an equal weight to any of the technological processing operation, starting from the sizing stage in steady-state of the processing subsystem.

Further on, the case of the control carried out on a separate machine, than the processing ones.

The controlling machine’s state C, having at inputs stockers $c_1, c_2, \ldots, c_s$, and outputs $c_{s+1}, \ldots, c_{2s}$, may be featured in a similar way as those of any technological device, by two parameters:

$$c(k+1)=u_c(k)+f_c(C(k), c_s(k), c_{s+1}(k))$$
$$C(k+1)=C(k)+Tu_c(k)+F_c(C(k), a(k)) \quad (6)$$

The decision taken as a result of the control, consists in framing each of the *k* controlled items *k=1, 2, …, K* within one of the three classes:

- the item corresponds and is to be shifted towards $e_1$ (output);
- the item is a waste and shifts also towards $e_1$;
- the item is a redeemable waste, needing to be repeated one of any operation, for example on one of the devices *A* or *B*.

The procedural model which is solving the question of the decision on the end of the control is:

“If $c(k)\neq0$ and C(k)=$C_k$ , than:

- If the item corresponds, it results: $u_c(k+1)=0$, C(k+1)=0, $U_c(k+1)=0$, $u_{s+1}(k+1)$=label “good”

- if the item is a waste, it results: $u_c$(k+1)=0, C(k+1)=0, $U_c$(k+1)=0, $u_{s+1}$8k+1)=label "waste"
- if the item is redeemable waste, it results: $u_c$(k+1)=0, C(k+1)=0, $U_c$(k+1)=0, $u_{s+1}$(k+1)=code of the operation to be repeated; End".

$C_k$ – represents the average number of controlling operations on the *k* –type item, and $U_c(k)$ – controlling speed at the moment *k.* They may be operated also with differentiated controlling speeds on items. Here, a differentiation by $C_k$ has been considered as enough. (If *k* is an index, it marks the item type and if it is a variable, it marks the timely moment.)

Directing items towards $e_1$ (the item is a good one or a redeemable waste) or towards $a_1$ or $b_1$ (the item needs an operation to be repeated), is achieved when they "arrive" in $e_{2s}$ and are taken over by the conveying system. This "knows" what it has to done with each item class.

## 3. ACHIEVING SIMULATION MODEL

Simulation model is obtained by "assembling" procedural sub-models of the above presented type.

If the initial simulation model contains also other subsystems executing space and time convey (for example assembling/mounting subsystems) or during repeated simulations, corrections are imposing also other subsystems, than the according parameters and specific procedural models will be attached to them.

Simulation needs all initial conditions (parameter value at the moment *0*) and specific constants: position within the system of all points (stockers) where the individual transport systems carries out loading – unloading operations, average speed of the transport system, chip removing level within each operation and average speed of carrying out the respective operation, to be fixed. Also the time increment is to be fixed.

As any larger procedural model, this proposed one needs for its execution a computer language or program and also a high calculating speed and a large memory space.

## 4. CONCLUSIONS

The model presented hereby represents the theoretical base of achieving algorithms for procedural simulating models.

The simulating model proposed hereby, being a procedural model, needs for its execution a computer language or program and also a high computing speed and memory space.

### REFERENCES

1. Boncoi, Gh., Calefariu, G., et al.: Sis*teme de producție,* Vol. I, *Concepte, Automatizări*, Editura Universității Transilvania Braşov, 2000
2. Calefariu, G.: *Optimizarea sistemelor de fabricație*, Editura Universității Transilvania din Braşov, 2002
3 Catrina, D., et al.: *Sisteme flexibile de prelucrare prin aşchiere*, Editura MatrixROM, Bucureşti, 2006
4 Slack, N., et al.: *Operations and process management*, FT Prentice Hall, New York, 2007