LSC: A PORTABLE CONTINUOUS SIMULATION SOFTWARE

M.Gaci , A Babaamer
Instrumentation and Control laboratory
2 ,Bd Frantz Fanon -ALGIERS-

ABSTRACT

This paper presents a Continuous Simulation Software (LSC) developped in our laboratory.
The first version of this software treats dynamical systems , that are described by a set of explicit linear or non linear algebro-differential equations .
The software has an interface (High Level Language) , that permits an auto-guided dialogue with the user .
The internal architecture of the system is structured in two subsystems : a control subsystem and an operating subsystem
The package contains a library of algorithms and external functions that permits simulation models from different fields (socio-economic , technological ...)

INTRODUCTION

LSC is a software , that permits the simulation of dynamical models described by state equations with or without coupled equations .

LSC is initialy implemented on the VAX 750 VMS environment , and is interactive , extensible , portable with graphic capabilities Gaci (1985) .

This software is written with the support of two languages associated to two traitements levels :
- a compilation - interpretation and derivation level , with PASCAL language
- a numerical traitement level ,with FORTRAN 77 language.

The transportability of LSC , can be assured by many machines having the following technical characteristics :
- the PASCAL and FORTRAN 77 compilers accepted by ANSI and ISO normes
- Sufficent memory

EXTERNAL ARCHITECTURE

The goal of the simulation system is to assist the user with computer facilities in simulation development. The external architecture of the system LSC is mainly constituted by an interface (High level language) which permits the auto guided dialogue with the user, and it follows the main steps used in the process of modelisation by systems approach Ǵaci (1985) see figure 1 .

The way to help the user is to regroup the commands associated with specific operations : (Initializations,Declarations,...)

This regroupment of commands was defined with respect to two points:

1°) The large steps for analysis and modeling process

2°) To avoid the errors due to the mode of appearance of commande sequences (exemple : run the execution before verify the model coherency)

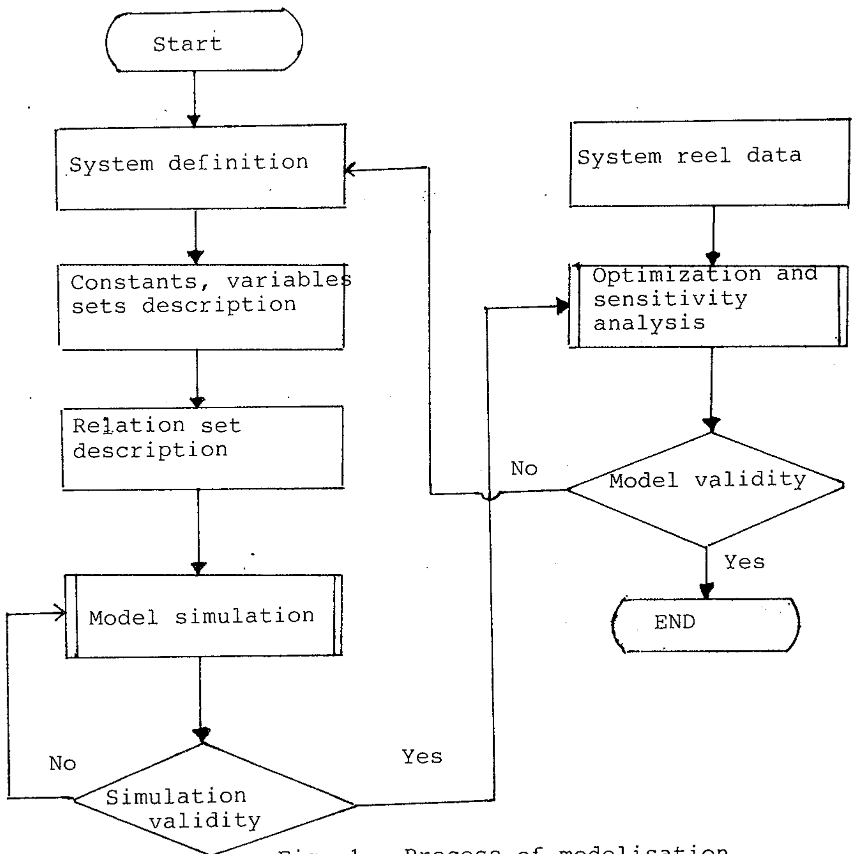

Fig 1 . Process of modelisation

The main environments retained in the structure of the high level language of LSC are the following:

- -Starting environment : permits to define the model and type of simulation
- -Formal declaration environment : regroups different commands that permit the description of set of variables (sates ,inputs outputs , auxiliary ...), set of constants and that makes the memory allocation
- -Initialization environment : permits the user to affect the initial values of state variables , constants and arrays
- -Integration environment : permits to define the system simulation parameters .
- -Relations environment : permits the description of set of equations quantifying the model (state , coupled , output equations ...)
- -Editionenvironment :permits the user to edit the results of simulation by printing the table or ploting the curves
- -Control environment : permits the control of coherency of the model

The figure 2 , illustrates the principle of environment sequences in the system

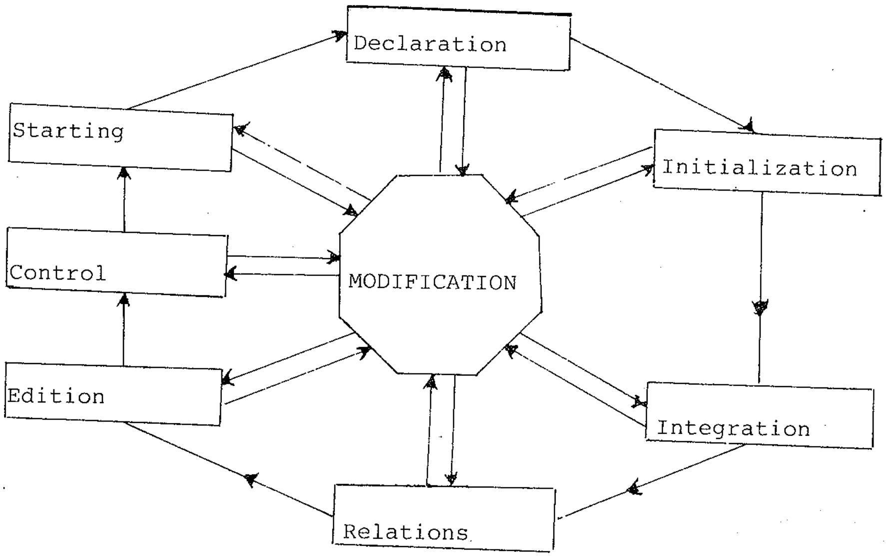

Fig 2 . Environment sequences in the system

FUNCTIONALITY OF LSC :

Simulation

In continuous simulation context, LSC permits to resolve the explicit stiff or non stiff algebro-differential system.

$\dot{\underline{x}} = f(\underline{x}, \underline{u}, t)$ state equation

$\underline{z} = g(\underline{x}, \underline{u}, t)$ algebric equation (generally coupled or auxilliary equation)

In stiff models, the system is integrated by Gear algorithm, Gear (1971) and the jacobian matrix is computed formally by the derivative module.

In non stiff models, the user has the performing packages containing the modified **Hamming and Runge Kutta Gill** algorithms.
The euler algorithm can be used when ever a precise simulation is not required.

There is also a special procedure for the simulationof recurrent dynamical systems, in the form :

$x_{t+1} = f(x_t, u_t)$

Different fields of models can be treated by this software.The user has many possibilities to quantify relations by external functions :

Delay, switch, step, ramp, tabl, tabhl, noise, randu, compr, sin, cos, log, alog,sqrt,exp,lim ...

Sensitivity Analysis

We plan to developp the environment of sensitivity analysis in parameter or decision context.
At dynamical model :

$\dot{\underline{x}} = f(\underline{x}, \underline{u}, \underline{p}, t)$ (1)

Where :

$\underline{x}$: state variable vector

$\underline{u}$: decision vector

$\underline{p}$: parameter vector

We define the sensitivity coefficient P.M Frank (1978) by :

$$v_i = \frac{\partial x_i}{\partial p} \quad (2)$$

$$\frac{\partial v_i}{\partial t} = \frac{\partial}{\partial t}\left(\frac{\partial x_i}{\partial p}\right) = \frac{\partial f_i}{\partial p} + \sum_{j=1}^{N} \frac{\partial f_i}{\partial x_j} \frac{\partial x_j}{\partial p} \quad (3)$$

$$\dot{\underline{v}} = f_p + J\underline{v} \quad (4)$$ state sensitivity equation

Where:
J:Jacobian of the right hand side expression in (1)
f_p:Vector of sensitivity functions
The LSC , computes automatcly J and f_p and resolve the linear system (4) .

INTERNAL ARCHITECTURE

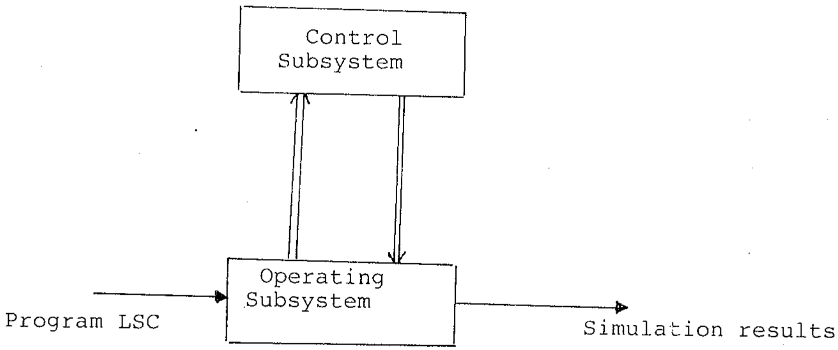

Fig 3. Subsystem decomposition of LSC

From the point of the view of the system approach , LSC is defined as a structure of two interacting subsystems Mesarovic (1970) Fig .3 :

- -A control subsystem , having the following functions :
  - .analysis and interpretation of commands
  - .data management
- -Operating subsystem ,executing the tasks provided from the directives of the control subsystem.

The internal architecture is not visible for the user ,but he can have access , when an entention is necessary for particular traitment ; for this raison the internal architecture is decomposed into modules Fig .4 .

The modules retained are the following :

- -Compilation interpretation module Babaamer ,(1985) : having the following functions:
  - .Lexical and syntaxic analysis of commands and expressions
  - .Internal code generation for dynamical models

-Formal derivative module , Derradji (1986), Sitbon (1975) : give the formal derivative expression for the right hand side expression of the differential system and conditions the jacobian matrix .

-Data structure module assures the internal representation of data models and stocks the results of the simulation model .

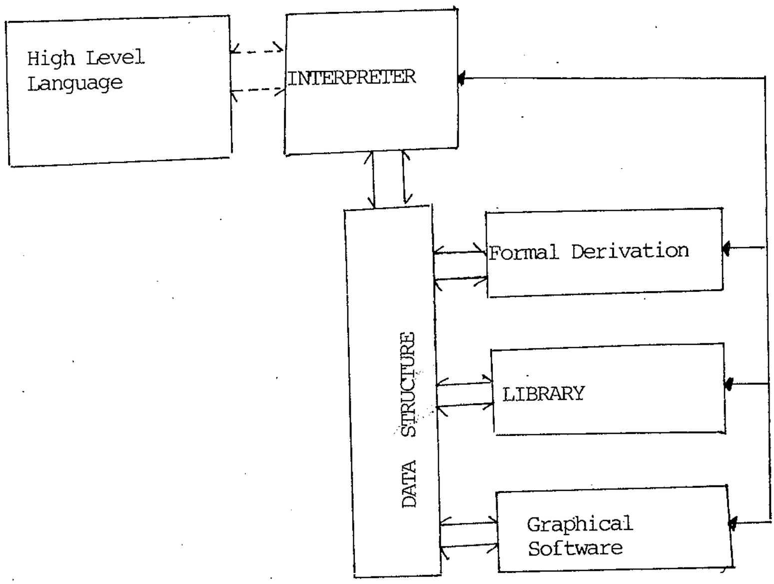

Fig 4. Modular structure of LSC

-Library module : regroupes the different integration packages and subroutines of external functions .

-Graphical Software module :
Actually the graphical software contains only one procedure which is the display of curves in discrete form .
However we are developping a software in order to use graphical screen and ploter .

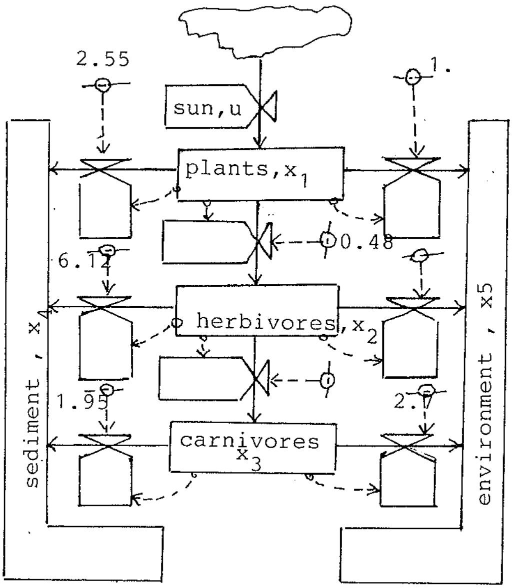

Fig 5 . Flow diagram of Cedar Bog Lake model

```
$R LSC
SIMUL CEDAR
TYPE DIFF
FINI
COM DECLARATION FORMELLE
ETAT PLANT,HERBIV,CARNIV,SEDIM,ENVIR
ENTR SUN
FINI
COM INITIALISATION
PLANT=0.83
HERBIV=0.003
CARNIV=0.0001
SEDIM=0
ENVIR=0
FINI
COM PARAMETRES DE SIMULATION
PARA T=0,TMAX=2,DT=0.04
MODE EUL
FINI
COM DYNAMIQUE DU MODELE
SUN=95.9*(1.0+0.635*SIN(6.28*T))
PLANT.=SUN-(1.0 +2.55+0.48)*PLANT
```

```
HERBIV.=0.48*PLANT -(6.0+6.12+4.85)*HERBIV
CARNIV.=4.85*HERBIV-(2.7+1.95)*CARNIV
SEDIM.=2.55*PLANT+6.12*HERBIV+1.95*CARNIV
ENVIR.=1.0*PLANT+6.9*HERBIV+2.7*CARNIV
FINI
COM EDITION DES VARIABLES
NUPR=4
PRINT SUN,PLANT,HERBIV,SEDIM
GRAPH SUN(35,157,*),PLANT(0,40),HERBIV(0,1),SEDIM(0,130)
10
TERM
```

Fig 6.LSC **source** listing program

TIME	SUN	PLANT	HERBIV
0.00000E+00	9.59000E+01	8.30000E-01	3.00000E-03
1.60000E-01	1.47300E+02	1.55040E+01	2.89299E-01
3.20000E-01	1.51027E+02	2.69666E+01	6.60746E-01
4.80000E-01	1.03625E+02	3.00919E+01	8.44861E-01
6.40000E-01	4.90577E+01	2.47066E+01	7.70569E-01
8.00000E-01	3.79362E+01	1.71546E+01	5.56806E-01
9.60000E-01	8.05754E+01	1.51081E+01	4.26607E-01
1.12000E+00	1.37428E+02	2.07960E+01	5.12995E-01
1.28000E+00	1.55764E+02	2.91063E+01	7.41917E-01
1.44000E+00	1.18577E+02	3.24125E+01	9.04056E-01
1.60000E+00	6.03576E+01	2.76882E+01	8.50519E-01
1.76000E+00	3.51032E+01	1.93349E+01	6.31697E-01
1.92000E+00	6.62369E+01	1.51101E+01	4.50908E-01

Fig 7. Simulation results

LSC USE EXAMPLE

We simulated Cedar Bog Lake model Burnes ,(1980) with LSC language .
The model consists of five states :
1- Plants: x_1 2- Herbivores : x_2 3 - Carnivore : x_3
4- Sediment : x_4 5- Environment : x_5
and u which is the source or driving input of the sun's energy.
These entities are modeled in terms of their energy content (cal/cm^2)and the energy transfers as shown in the flow diagram that appears in figure 5 . The LSC source listing program is shown in figure 6 the simulation results appear in figure 7.

CONCLUSION

LSC is actually used to simulate dynamic systems , however it is expected to integrate in the future some modules ; such as :
- sensitivity analysis module
-interactive and graphical module
In order to use it as a computer aided modelisation system .

REFERENCES

Burns ,J.R (1980) . From flow diagram to simulation model IEEE SMC Conference
Gaci,M (1985) . Un outil informatique de simulation de systèmes dynamiques . Séminaire interne
Babaamer , A (1985) . Réalisation d'un interpreteur du module Simul . Mémoire USTHB
Derradji , N (1986) . Réalisation d'un logiciel de dérivation formelle . Mémoire USTHB
Frank, P.M (1978) . Introduction to System Sensitivity Theory Academic Press
Gear , C.W (1971) . Numerical Initial Value Problems in ordinary differential equations . Prentice Hall
Sitbon , A (1975) . Les structures de listes et leurs applications . Masson
Mesarovic ,M.D , and al ,(1970) . Theory of Hierarchical Multilevel Systems Academic Press