NONLINEAR ELECTRONICS (NOEL) PACKAGE 0:

GENERAL DESCRIPTION

by

An-Chang Deng and Leon O. Chua

Memorandum No. UCB/ERL M86/22

13 March 1986

NONLINEAR ELECTRONICS (NOEL) PACKAGE 0:

GENERAL DESCRIPTION

by

An-Chang Deng and Leon O. Chua

Memorandum No. UCB/ERL M86/22

13 March 1986

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

NONLINEAR ELECTRONICS (NOEL) PACKAGE 0:

GENERAL DESCRIPTION

by

An-Chang Deng and Leon O. Chua

Memorandum No. UCB/ERL M86/22

13 March 1986

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

NOEL PACKAGE 0 : GENERAL DESCRIPTION†

An-Chang Deng and Leon O. Chua

Department of Electrical Engineering and Computer Sciences
and the Electronics Research Laboratory
University of California, Berkeley, CA 94720

ABSTRACT

NOEL is a general-purpose circuit simulator for analyzing nonlinear circuits. It consists of several packages; namely,

1. Linear circuit formulations, n-port representations and state equations
2. Computer generation of symbolic transfer functions
3. Nonlinear DC analysis
4. Nonlinear transient analysis
5. Canonical piecewise-linear modeling
6. Canonical piecewise-linear DC analysis
7. Canonical piecewise-linear transient analysis
8. Phase portrait of second order nonlinear circuits

and it will be extended for more packages with different purposes.

The purpose of this report is to define the input format language for describing the circuit to be analyzed by NOEL. Due to the consideration of consistency, the input format of NOEL roughly follows the rules in SPICE, except for those of some circuit elements which are not allowed in SPICE.

March 13, 1986

† Research supported by the Joint Services Electronics Program under Contract F49620-84-C-0057, and the Semiconductor Research Corporation under Grant SRC 82-11-008.

NOEL PACKAGE 0 : GENERAL DESCRIPTION

I Introduction

Circuit simulation is an important technique in circuit design for predicting or verifying the circuit behavior. A number of simulation programs have been developed and are widely used. Each of these programs is capable of analyzing certain classes of nonlinear circuits, and might run into difficulty in case of circuits with general nonlinear characteristics. Moreover, most of these programs require a large amount of memory space which is beyond the capability of today's personal computer.

In view of the above restrictions, we henceforth are motivated to develop a general nonlinear circuit simulator, called NOEL (NOnlinear ELectronics packages), which is written in C language and running under PC-DOS and UNIX operating systems. Since a large class of general nonlinear elements are included in NOEL, it can be used not only as a circuit-level simulator, but also for simulating devices and systems (such as control systems, electric power systems, biological systems, and mechanical systems) as long as they can be modeled by interconnections of those elements allowed in NOEL.

This report is a general description for NOEL which consists of several packages[1-8]; namely,

1. Linear circuit formulations, n-port representations and state equations
2. Computer generation of symbolic transfer functions
3. Nonlinear DC analysis
4. Nonlinear transient analysis
5. Canonical piecewise-linear modeling
6. Canonical piecewise-linear DC analysis
7. Canonical piecewise-linear transient analysis
8. Phase portrait of second order nonlinear circuits

and it will be extended for more packages with different purposes.

II Input Format Language

The format of the input file for describing the circuit roughly follows the rules in SPICE[9] due to the consideration of consistency. Each line in the input file "xx...x.spc", where "xx...x" is any character string with no more than 8 characters, can be classified into one of the following categories : (1) Element Line (2) Model Line (3) Graphic Control Line (4) Comment Line (5) Declaration Line. The last line of the input file should be ".end" or ".END" to denote the end of the file.

1. Element Line

Each element line specifies the element type, connection topology, and characteristic of an element in the circuit. Its format consists of the following sequential fields :

[NAME] [TOPOLOGY] [VALUE/RELATION]

1.1. Name Field

A name field consists of a string of no more than 8 alpha-numerical letters which must begin with a *capital* letter to identify the element type :

'R' : 2-terminal resistor (linear or nonlinear)
'C' : 2-terminal capacitor (linear or nonlinear)
'L' : 2-terminal inductor (linear or nonlinear)
'V' : independent voltage source (time-invariant or time-varying)
'I' : independent current source (time-invariant or time-varying)
'E' : linear voltage-controlled voltage source
'F' : linear current-controlled current source
'G' : linear voltage-controlled current source
'H' : linear current-controlled voltage source
'K' : nonlinear controlled source
'N' : 2-port or 3-terminal resistor (linear or nonlinear)

1.2. Topological Field

The topological field is used to specify the connection of the element in the circuit. It consists of two nonnegative integers for 2-terminal elements (e.g., R, V, I, C, L), four nonnegative integers for 2-port or 3-terminal † elements (e.g., E, F, G, H, N), and 2k+2 nonnegative integers for the nonlinear controlled source with k controlling variables. Each integer number denotes a node number with the convention that the current direction is chosen to flow from the first node to the second node, and the voltage polarity is positive in the first node and negative in the second node. In the case of linear controlled sources (E, F, G, H), the first two nodes correspond to the controlled port, and the last two nodes denote the controlling port. For 2-port or 3-terminal resistor, the first two nodes denote the node numbers for port 1 and the other two represent those of port 2. For a nonlinear controlled source with k controlling ports, the first two nodes represent the node numbers of the controlled port, and each sequential pair of the remaining 2k node numbers characterizes one of the controlling ports.

1.3. Value Field

Value field only appears in the element line for *linear elements* or *time-invariant independent sources* to represent the linear characteristic (resistance for R, capacitance for C, inductance for L, voltage for V, and current for I). It may be an integer number (e.g., 125 or -37), a floating-point number (e.g., 3.14159, or -0.0342), a number followed by an integer exponent (1.0E-14 or 2.43e6), or a number followed by one of the following unit characters to represent a particular scaling factor :

'f' : 1E-15
'p' : 1E-12
'n' : 1E-9
'u' : 1E-6
'm' : 1E-3
'K' : 1E3
'M' : 1E6
'G' : 1E9
'T' : 1E12

e.g., $3.54m = 3.54\times10^{-3}$ and $7.91G = 7.91\times10^{9}$.

1.4. Relation Field

The relation field of each element line is included in a pair of brackets "{.......}" and is defined in the following way for each type of element. The functions f() and g() can be any arithmetical expression accepted by the C compiler, and may contain any mathematical function (e.g., sin(), exp(), fabs()) in the C library. It can also be a character string with no more than 8

† A 3-terminal resistor is modeled as a 2-port element.

characters, which denotes the model name of the element whose element characteristic is defined in a separate model line.

1.4.1. 2-terminal nonlinear resistor

(a) General Characteristic

(i) voltage-controlled relation

$$\{i = f\,(v\,)\}$$

(ii) current-controlled relation

$$\{v = f\,(i\,)\}$$

(iii) implicit relation

$$\{g\,(v\;,i\,) = 0\}$$

(b) Piecewise-Linear Characteristic

(i) voltage-controlled relation

$$\{i = (x_0\,,\,y_0)(x_1\,,\,y_1)\,\ldots\,(x_n\;,\,y_n\,)(x_{n+1}\,,\,y_{n+1})\}$$

(ii) current-controlled relation

$$\{v = (x_0\,,\,y_0)(x_1\,,\,y_1)\,\ldots\,(x_n\;,\,y_n\,)(x_{n+1}\,,\,y_{n+1})\}$$

where the resistor is assumed to be characterized by the 1-dimensional piecewise-linear function, as shown in Fig.1, with n breakpoints $(x_1\,,\,y_1)\,,(x_2\,,\,y_2)\,,\,\ldots\,(x_n\;,\,y_n\,)$. Point $(x_0\,,\,y_0)$ (resp.; $(x_{n+1}\,,\,y_{n+1})$) is an arbitrary point of the ending segment in the left side (resp.; right side). x (resp.; y) denotes the coordinate for the controlling variable (resp.; controlled variable); e.g., x represents the voltage (resp.; current) of a voltage-controlled (resp.; current-controlled) resistor, and y is the corresponding complement variable.

1.4.2. 2-terminal nonlinear capacitor

(a) General Characteristic

(i) voltage-controlled relation

$$\{q = f\,(v\,)\}$$

(ii) charge-controlled relation

$$\{v = f\,(q\,)\}$$

(iii) implicit relation

$$\{g\,(v\;,q\,) = 0\}$$

(iv) capacitance expression

$$\{C(v\,) = f\,(v\,)\}$$

(b) Piecewise-Linear Characteristic

(i) voltage-controlled relation

$$\{q = (x_0\,,\,y_0)(x_1\,,\,y_1)\,\ldots\,(x_n\;,\,y_n\,)(x_{n+1}\,,\,y_{n+1})\}$$

1.4.3. 2-terminal nonlinear inductor

(a) General Characteristic

(i) current-controlled relation

$$\{phi = f\ (i\ )\}$$

(ii) flux-controlled relation

$$\{i\ = f\ (phi\ )\}$$

(iii) implicit relation

$$\{g\ (i\ ,\ phi\ ) = 0\}$$

(iv) inductance expression

$$\{L\ (i\ ) = f\ (i\ )\}$$

(b) Piecewise-Linear Characteristic

(i) current-controlled relation

$$\{phi = (x_0\ ,\ y_0)(x_1\ ,\ y_1)\ .\ .\ .\ .\ (x_n\ ,\ y_n\ )(x_{n+1}\ ,\ y_{n+1})\}$$

1.4.4. time-varying voltage or current source

$$\{f\ (t\ )\}$$

1.4.5. nonlinear controlled source

$$\{y\ = f\ (x_1\ ,\ x_2\ ,\ .\ .\ .\ x_k\ )\}$$

where k is either 1 or 2, and each x_i for i=1,..,k is the i-th controlling variable; e.g., x_i is the open-circuit voltage (resp.; short-circuit current) if the i-th port is voltage-controlled (resp.; current-controlled). y is the controlled voltage (resp.; controlled current) if the output port is a controlled voltage source (resp.; controlled current source).

1.4.6. 2-port or 3-terminal resistor

(a) General Characteristic

(i) voltage-voltage controlled relation

$$\{i\,1 = f\ (v\,1\ ,\ v\,2);\ i\,2 = g\ (v\,1\ ,\ v\,2)\}$$

(ii) current-current controlled relation

$$\{v\,1 = f\ (i\,1\ ,\ i\,2);\ v\,2 = g\ (i\,1\ ,\ i\,2)\}$$

(iii) voltage-current controlled relation

$$\{i\,1 = f\ (v\,1\ ,\ i\,2);\ v\,2 = g\ (v\,1\ ,\ i\,2)\}$$

(iv) current-voltage controlled relation

$$\{v\,1 = f\ (i\,1\ ,\ v\,2);\ i\,2 = g\ (i\,1\ ,\ v\,2)\}$$

(b) Piecewise-Linear Characteristic

Assume the element characteristic is described by a 2-dimensional canonical piecewise-linear function[10]

$$y_1 = a_1 + b_{11}x_1 + b_{12}x_2 + \sum_{j=1}^{\sigma} c_{1j}\ |\ \alpha_{j1}x_1 + \alpha_{j2}x_2 - \gamma_j\ |$$

$$y_2 = a_2 + b_{21}x_1 + b_{22}x_2 + \sum_{j=1}^{\sigma} c_{2j} \mid \alpha_{j1}x_1 + \alpha_{j2}x_2 - \gamma_j \mid$$

where for j=1,2, x_j denotes the voltage (resp.; current) of port j if it is a voltage-controlled port (resp.; current-controlled port); and y_j is the corresponding complement variable in port j. Then the relation field for this piecewise-linear 2-port resistor is given as

$\{(y_1, y_2){:}P1{=}(a_1, b_{11}, b_{12}, c_{11}, c_{12}, \ldots c_{1\sigma});P2{=}(a_2,$

$\$b_{21}, b_{22}, c_{21}, c_{22}, \ldots c_{2\sigma});Bd=(\alpha_{11}, \alpha_{12}, \gamma_1, \alpha_{21}, \alpha_{22},$

$\$\ \gamma_2, \ldots \alpha_{\sigma 1}, \alpha_{\sigma 2}, \gamma_\sigma)\}$

where

(1) $(y_1, y_2) = (i, i)$

(2) $(y_1, y_2) = (v, v)$

(3) $(y_1, y_2) = (i, v)$

(4) $(y_1, y_2) = (v, i)$

if and only if it is a

(1) voltage-voltage controlled 2-port resistor

(2) current-current controlled 2-port resistor

(3) voltage-current controlled 2-port resistor

(4) current-voltage controlled 2-port resistor

Remark : The '$' character represents a continuation for the previous line and should start from the first position immediately following the line to be continued.

2. Model Line

The use of model lines serves as an easy reference when more than one element in the circuit have the same element characteristic, or the model description is too lengthy to fit into the relation field of an element line. The format for the model line is

.model [MODEL_NAME] [{MODEL DESCRIPTION}]

where MODEL_NAME is a character string with no more than 8 characters, and it should match the model name of the relation field of an element line whose element behavior is defined in this separate model line. The format of MODEL DESCRIPTION within the bracket "{......}" should follow the rules of the relation field for each type of element.

3. Graphic Control Line

These lines are used to specify the graphic parameters or titles for the graphic solution when the simulation result is to output from the color monitor.

(i) x_axis title

.x_name xx...x

(ii) y_axis title

.y_name xx...x

(iii) graphic title

.title xx...x

where "xx...x" is a character string with no more than 30 characters (including spacings).

(iv) x_axis range

.x_axis xmin xmax

(v) y_axis range

.y_axis ymin ymax

where xmin, xmax, ymin, and ymax are some numbers such that the graphic box covers the area xmin $\leq$ x $\leq$ xmax and ymin $\leq$ y $\leq$ ymax.

4. Comment Line

The use of comment lines is to make the circuit description in the input file easier to understand. It starts with the character '*' and will be ignored by NOEL.

Example

```
* the graphic title is "Schmitt-Trigger Circuit"
.title Schmitt-Trigger Circuit
* the x_axis title name is "Vin(Volt)"
.x_name Vin(Volt)
* the y_axis title name is "Vout(Volt)"
.y_name Vout(Volt)
* x is between 0 and 10
.x_axis 0 10
* y is between 0 and 10
.y_axis 0 10
```

5. Declaration Line

Declaration lines in the input file will be bypassed to appear in the output file in the case of circuit formulation for generating the C source code.

(i) include line

.include "xx...x.h"

(ii) external declaration line

.extern [TYPE] [VARIABLE]

where TYPE denotes the variable type, e.g., *int* or *double* and VARIABLE is the variable name which may appear in the relation field of some element lines.

Example

```
* include the file "math.h"
.include "math.h"
* declare the external variable kx, ky and x_dot
.extern int kx,ky
.extern double x_dot
```

6. Element Format

1 2-terminal linear resistor

General Format

Rxx...x n1 n2 VALUE

Examples

Rload 4 7 1000

RL23 3 0 1.2K

R1 72 100 2.5E3

2 2-terminal nonlinear resistor

General Format

Rxx...x n1 n2 RELATION

Examples

RNon 2 5 {i=v*v+2*v+1}

Rix 7 0 {v=sin(i)*i+exp(i/2)}

Rinp 1 4 {cos(v+1)*i-v=0}

Rpn 6 9 pnmod

.model pnmod {i=1.0e-14*exp(v/0.026)}

3 2-terminal piecewise-linear resistor

General Format

Rxx...x n1 n2 RELATION

Examples

Rpwl 1 3 {i=(-1,-1)(0,2)(1,3)(2,0)}

Ry 4 0 {v=(-2,5)(1,2)(3,2)(4,-1)}

4 2-terminal linear capacitor

General Format

Cxx...x n1 n2 VALUE

Examples

Cp 2 5 3.2p

Cx 1 3 2.5e-3

Cya 62 17 1u

5 2-terminal nonlinear capacitor

General Format

Cxx...x n1 n2 VALUE

Examples

Cx 1 0 {q=v*v*v+v}

C20 6 7 {v=sin(q*q)+2*q}

Ca 9 3 {v*q-1=0}

C11 7 1 {C(v)=sin(v)}

6 2-terminal piecewise-linear capacitor

General Format

Cxx...x n1 n2 RELATION

Examples

Ca 7 9 {q=(-3,1)(-1,-1)(2,4)}

Cp 6 4 modc

```
.model modc {q=(-4.2,1.2e-3)(-2.7,9.1e-4)
$(0,1.7e-3)(6.4,1.5e-3)}
```

7 2-terminal linear inductor

General Format

```
Lxx...x n1 n2 VALUE
```

Examples

```
Lcoil 7 11 2.7m
Lx 9 3 1.25e-3
L12 4 0 6.7u
```

8 2-terminal nonlinear inductor

General Format

```
Lxx...x n1 n2 VALUE
```

Examples

```
L1 4 5 {phi=i*exp(-i)+2}
Lout 7 2 {i=phi*phi-2*phi}
Lx 4 0 ind
 .model ind {i*i/phi-2*cos(i+phi)=0}
Lcx 4 3 {L(i)=sin(i)}
```

9 2-terminal piecewise-linear inductor

General Format

```
Lxx...x n1 n2 RELATION
```

Examples

```
Lpwl 1 5 {phi=(-4,2)(-3,-1)(0,0)(1,7)}
Lcoup 7 2 pwlind
 .model pwlind {phi=(-2.5,1.4e-2)(-2,0)(0,1e-2)
$(1.6,3.2e-3)}
```

10 time-invariant independent voltage source

General Format

```
Vxx...x n1 n2 VALUE
```

Examples

```
Vcc 1 0 5
Vs 7 2 -3.7
Vin 11 22 0.22K
```

11 time-varying independent voltage source

General Format

```
Vxx...x n1 n2 RELATION
```

Examples

```
Vin 1 0 {sin(1000*t)}
Vx 2 5 {fabs(t-1)+2*fabs(t)}
```

12 time-invariant independent current source

General Format

Ixx...x n1 n2 VALUE

Examples

Is 2 0 3.2m

Ix1 7 2 -1.9

I2 1 3 6.4u

13 time-varying independent current source

General Format

Ixx...x n1 n2 RELATION

Examples

I25 4 2 {cos(t+2)}

Ia7 1 0 {t*t-2}

14 linear voltage-controlled voltage source

General Format

Exx...x n1 n2 n3 n4 VALUE

Examples

Econ 3 6 5 3 10

E12 1 2 3 4 1K

15 linear current-controlled current source

General Format

Fxx...x n1 n2 n3 n4 VALUE

Examples

Fr 2 1 1 4 0.98

F1 1 2 3 4 -2m

16 linear voltage-controlled current source

General Format

Gxx...x n1 n2 n3 n4 VALUE

Examples

G25 2 0 1 7 1E-3

Gb 12 7 3 9 -3.8m

17 linear current-controlled voltage source

General Format

Hxx...x n1 n2 n3 n4 VALUE

Examples

H23 4 2 4 1 1.2E-3

Hratio 4 5 3 9 -3m

18 nonlinear controlled source

General Format

Kxx...x n1 n2 ..... n(2k-1) n2k n(2k+1) n(2k+2) RELATION

Examples

K2 1 2 3 4 5 6 {i=v1*v2-exp(v1+v2)}

Ka 2 5 7 3 {v=3*v1*sin(v1)}

Ky 7 2 5 4 3 8 1 6 nctr

.model nctr {v=i1*v2+sin(i3)}

19 nonlinear 2-port or 3-terminal resistor

General Format

Nxx...x n1 n2 n3 n4 RELATION

Examples

Nx 1 2 3 4 {i1=v1*v2;i2=v1-v2}

N12 1 5 9 10 {v1=i1/i2;v2=sin(i1+i2)}

Na 2 1 3 7 {i1=v1-i2;v2=v1*i2}

N6 2 0 1 0 nmod

.model nmod {v1=i1/v2;i2=i1*i1-v2+2}

20 piecewise-linear 2-port or 3-terminal resistor

General Format

Nxx...x n1 n2 n3 n4 RELATION

Examples

N21 1 2 3 4 pwl

.model pwl {(i,i):P1=(2.1,3,-1.5,1.7,-3.5);P2=(-2.5,7,

$4,-2.6,6.3);Bd=(1,2,1,-2,3,-6)}

7. Limitations

1. The circuit should have no more than 50 nodes.
2. The length of each input line should not exceed 80 characters. (a line longer than 80 characters can continue in the next line starting with '$')
3. The model description for each element should not exceed 500 characters.

In canonical piecewise-linear analysis[6,7]

4. The number of piecewise-linear 1-port (or 2-terminal) resistors in the circuit should not exceed 50.
5. The number of piecewise-linear 2-port (or 3-terminal) resistors in the circuit should not exceed 50.
6. The number of breakpoints in the piecewise-linear characteristic of each piecewise-linear 1-port (or 2-terminal) resistor should not exceed 10.
7. The total number of partition boundaries in the piecewise-linear characteristic of piecewise-linear 2-port (or 3-terminal) resistors should not exceed 500.
8. Each numerical data in the model description of piecewise-linear resistors should have no more than 20 digits.

Remark :

Each upper bound in the above limitations can be modified easily by changing the definitions of the constant variables. Arbitrarily large number can be used for each of the size limit as long as the request is allowed by the hardware capability.

References

[1] A.C.Deng and L.O.Chua, "NOnlinear ELectronics package 1 : linear circuit formulations, n-port representations and state equations," ERL Memo., UCB/ERL M86, University of California, Berkeley, 1986.

[2] A.C.Deng and L.O.Chua, "NOnlinear ELectronics package 2 : computer generation of symbolic transfer functions," ERL Memo., UCB/ERL M86, University of California, Berkeley, 1986.

[3] A.C.Deng and L.O.Chua, "NOnlinear ELectronics package 3 : nonlinear DC analysis," ERL Memo., UCB/ERL M86, University of California, Berkeley, 1986.

[4] A.C.Deng and L.O.Chua, "NOnlinear ELectronics package 4 : nonlinear transient analysis," ERL Memo., UCB/ERL M86, University of California, Berkeley, 1986.

[5] A.C.Deng and L.O.Chua, "NOnlinear ELectronics package 5 : canonical piecewise-linear modeling," ERL Memo., UCB/ERL M86, University of California, Berkeley, 1986.

[6] A.C.Deng and L.O.Chua, "NOnlinear ELectronics package 6 : canonical piecewise-linear DC analysis," ERL Memo., UCB/ERL M86, University of California, Berkeley, 1986.

[7] A.C.Deng and L.O.Chua, "NOnlinear ELectronics package 7 : canonical piecewise-linear transient analysis," ERL Memo., UCB/ERL M86, University of California, Berkeley, 1986.

[8] A.C.Deng and L.O.Chua, "NOnlinear ELectronics package 8 : phase portrait of second order nonlinear circuits," ERL Memo., UCB/ERL M86, University of California, Berkeley, 1986.

[9] A.Vladimirescu, K.Zhang, A.R.Newton, D.O.Pederson, and A.Sangiovanni-Vincentelli, *SPICE Version 2G User's Guide*, ERL Memo., University of California, Berkeley, August 10, 1981.

[10] S.M.Kang and L.O.Chua, "A global representation of multi-dimensional piecewise-linear functions with linear partitions," *IEEE Trans. Circuits and Systems* , Vol. CAS-25, pp.938-940, Nov. 1978.

Figure Caption

Fig.1 A piecewise-linear function characterized by n breakpoints, and two arbitrary points on the ending segments.

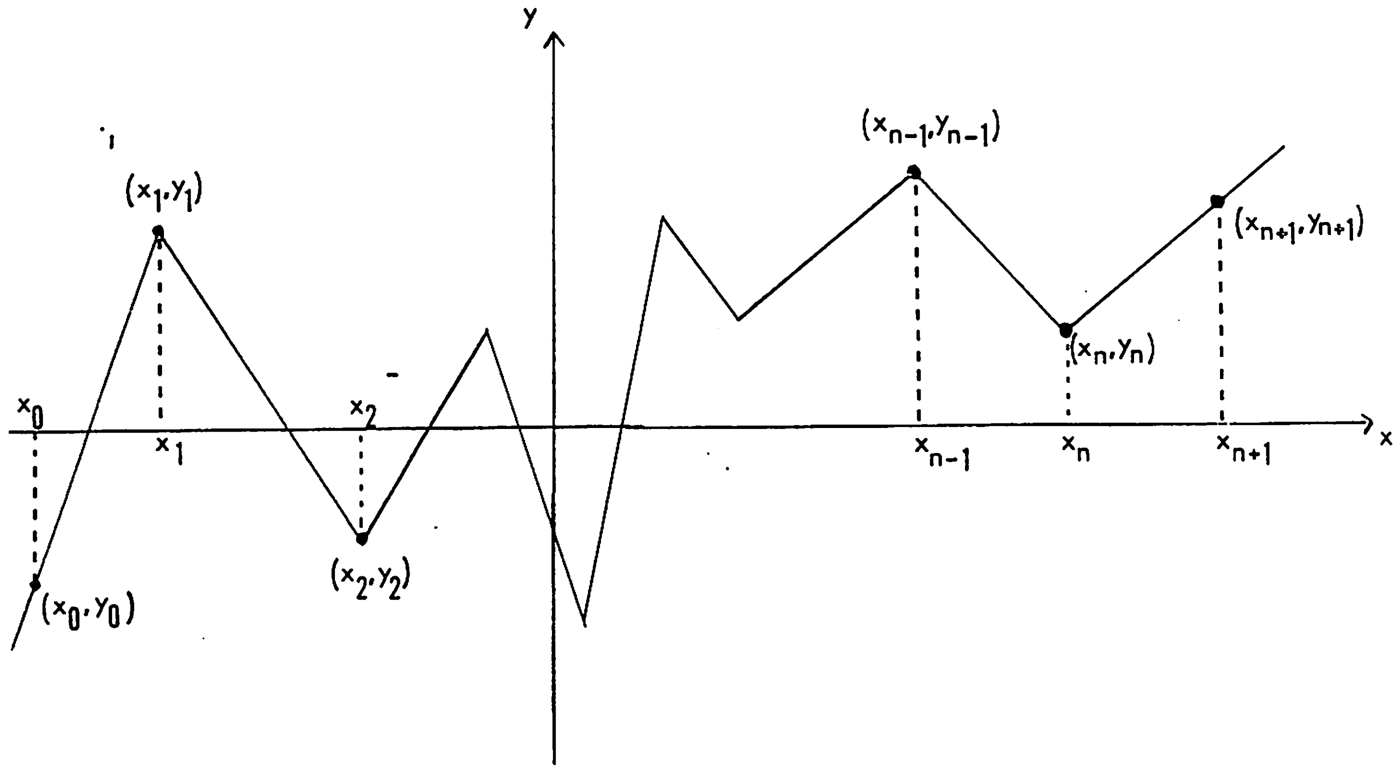

Fig. 1