DEVELOPMENT OF FLOWGRAPH-MAKING PROGRAM BY PROLOG

ISAMI YOSHIFUKU AND RYOICHI FUKUMURA

Department of Chemical Engineering, Faculty of Engineering,
Kagoshima University, Kagoshima 890

Key Words: Chemical Process Problem, Design Problem, Flowgraph Representation, AI Language, Prolog

Introduction

A chemical process problem solution generally has many difficult and complex components. To simplify the picture of the problem solution, it is convenient to use flowgraph representation^{2,3}, where an edge and a node correspond to a variable and an equation (or calculation technique) respectively. These flowgraphs are composed of flowgraph units which are converted individually from fundamental equations in the n -th cell or stage.

Hitherto, problems of multiple-effect evaporator processes, multicomponent distillation processes, multistage extraction processes, cooling tower processes and gas absorption processes have been treated. The procedure used is as follows:

- (1) formulation of fundamental equations in the n -th stage or n -th cell model manually.
- (2) construction of flowgraph by combination of flowgraph units manually.
- (3) programming with the procedural language BASIC or FORTRAN manually.
- (4) calculation by computer.

In the above procedure, flowgraph construction, apart from plain problems, consumes much of our time in selection and combination of flowgraph units. In this study, the flowgraph-making program FG-MAKE, written by AI language Prolog, is developed and applied to a crosscurrent multistage extraction process problem.

1. Flowgraph

Two examples are shown.

(Ex. 1) Construct a flowgraph which gives Reynolds number Re from density ρ and viscosity μ of fluid, pipe diameter D , and mass flow rate w .

The fundamental equations are as follows.

$$A = \pi D^2/4 \quad (1-1)$$

$$Q = w/\rho \quad (1-2)$$

$$u = Q/A \quad (1-3)$$

$$Re = Dup/\mu \quad (1-4)$$

Figure 1 shows a flowgraph for this problem. Quantities D , w , ρ , μ are called known variables, and Re is a goal variable.

(Ex. 2) Construct a flowgraph which gives friction factor f from Re , based on the following equation:

$$1/\sqrt{f} = 3.2 \log(Re\sqrt{f}) + 1.2$$

Fundamental equations are as follows:

$$y = Re\sqrt{f} \quad (2-1)$$

$$f = 1/(3.2 \log(y) + 1.2)^2 \quad (2-2)$$

A flowgraph for this problem is shown in Fig. 2, where a trial-and-error method with variable f will be used. Such a variable is called a tearing variable.

2. Algorithm and Data Base of the FG-MAKE Program

2.1 Algorithm

The purpose of this study is to construct a flowgraph which gives a goal variable from known variables when the fundamental equations are given. The number of flowgraphs constructed, tearing variables and unused equations are also to be searched.

To construct the FG-MAKE program by Prolog language there are two methods: one that follows the way from a goal variable to known variables, and one that goes in the opposite direction. We have adopted the former method because of its facility for

Fig. 1. Flowgraph of Ex. 1

Fig. 2. Flowgraph of Ex. 2

* Received April 5, 1991. Correspondence concerning this article should be addressed to I. Yoshifuku.

programming. Generally, representation of an equation is divided into two modes, explicit mode $z=f(x, y, \dots)$ and implicit mode $f(x, y, z, \dots)=0$. Preliminarily, all implicit equations are converted to explicit mode, where a list composed of an unknown variable, other variables and equation number is formed. An unknown variable is said to be listed up if the above list is formed.

First, an equation which contains a goal variable is searched, and the goal variable is listed up. For example, the goal variable Re is listed up as: $[Re, [D, u, \rho, \mu], (1-4)]$ in Ex. 1. This list-up operation must be carried out about a goal variable and unknown variables, and known variables must not be listed up. Second, all unknown variables attached to this equation are listed up. In Ex. 1, we have $[u, [Q, A], (1-3)]$, where variable u is attached to Eq. (1-4). Then these two lists are combined, where plural lists will be created according to the number of variables.

Thus this procedure will be continued until unknown variables do not appear. When a goal variable or an unknown variable appears twice in the procedure, this variable, termed a tearing variable, is treated as a known variable and must not be listed up.

A flowgraph thus obtained is called a coarse flowgraph, which looks like an overgrown tree with many leaves (corresponding to variables) and branches (corresponding to equation numbers). By cutting off unnecessary leaves and branches in a coarse flowgraph, we can get a refined flowgraph.

The FG-MAKE program written in Prolog language has twenty-four predicates composed of operations such as converting, searching, listing, cutting off and printing.

2.2 Data base of the FG-MAKE program

From the above discussion, the data base to be inserted into the FG-MAKE program should consist of implicit mode equations, known variables and a goal variable.

In the case of Ex. 1, we have the following data base:

equation $([A, D], (1-1))$,
equation $([Q, w, \rho, \mu], (1-2))$,
equation $([u, Q, A], (1-3))$,
equation $([Re, D, u, \rho, \mu], (1-4))$,
known-variable $([D, w, \rho, \mu])$,
goal $([Re])$

where 'equation', 'known-variable' and 'goal' are predicates in Prolog language, and an argument such as $([A, D], (1-1))$ shows a list of variable and equation number.

The FG-MAKE must be programmed so as to construct several flowgraphs by inputting this data

Fig. 3. n -th stage model of crosscurrent multistage extraction process

base. It is desirable, in addition, to obtain the number of flowgraphs (NF) attached to the class, which is based on the number of tearing variables (NT) and unused equations (NU).

3. Application Examples

3.1 Ex. 1

First, the FG-MAKE program is applied to Ex. 1, whose problem and data base are shown above. Results are as follows:

class 1 ($NT=0, NU=0$): $NF=1$

One class and one flowgraph attached to this class have been obtained, where no tearing variable exists and all equations are used to construct the flowgraph. Naturally, this flowgraph is the same as shown in Fig. 1.

3.2 Ex. 2

Second, the program is applied to Ex. 2, where the data base is

equation $([y, Re, f], (2-1))$,
equation $([f, y], (2-2))$,
known-variable $([Re])$,
goal $([f])$.

The result is as follows:

class 1 ($NT=1, NU=0$): $NF=2$

Two flowgraphs are constructed, where the tearing variable is f . One of the flowgraphs constructed is found to be the same as shown in Fig. 2.

3.3 Ex. 3

It is desired to develop a numerical method for the problem of the crosscurrent multistage extraction process system shown in Fig. 3, which is usually solved by a tedious graphical procedure. The feed stream R_{n-1} containing the solute A to be extracted and the solvent stream S_n enters at n -th stage. The solvent and raffinate streams are assumed to be in liquid-liquid equilibrium and the problem is to determine the composition of raffinate leaving at n -th stage.

The fundamental equations in the n -th stage are as follows:

$$R_{n-1} + S_n = E_n + R_n \quad (3-1)$$

Fig. 4. Flowgraph of Ex. 3

$$x_{An-1}R_{n-1} = y_{An}E_n + x_{An}R_n \quad (3-2)$$

$$x_{Cn-1}R_{n-1} + S_n = y_{Cn}E_n + x_{Cn}R_n \quad (3-3)$$

$$y_{An} = DC(x_{An}) \quad (3-4)$$

$$y_{Cn} = DC(x_{Cn}) \quad (3-5)$$

$$x_{An} = SCR(x_{Cn}) \quad (3-6)$$

$$y_{An} = SCE(y_{Cn}) \quad (3-7)$$

Eqs. (3-1) to (3-3), Eqs. (3-4) to (3-5), and Eqs. (3-6) to (3-7) represent mass balance equations, liquid-liquid equilibrium relations, and solubility curve¹⁾ respectively, where SCR and SCE denote the raffinate and extract sides from plait point in the solubility curve. We use the following data base:

equation () $\rightarrow$ Eqs. (3-1)–(3-7),
 known-variable $\rightarrow ([R_{n-1}, S_n, x_{An-1}, x_{Cn-1}])$,
 goal $\rightarrow ([x_{An}])$,

The application of the FG-MAKE program to this problem has established the following four classes.

class 1 ($NT=2, NU=1$): $NF=24$

class 2 ($NT=3, NU=1$): $NF=24$

class 3 ($NT=4, NU=1$): $NF=16$

class 4 ($NT=1, NU=3$): $NF=3$

By considering the facts that the increase in difficulty of calculation will be strikingly accelerated with increase in NT and that the increase in NU gives an incomplete calculation, we select class 1. A typical flowgraph is shown in Fig. 4, where tearing variables are x_{An} and E_{An} and the unused equation is Eq. (3-5).

Calculation is commenced with a left-hand side variable composed of $x_{An}, R_{n-1}, S_n, E_n, x_{Cn-1}, x_{An-1}$

Fig. 5. Sequential stage flowgraph of Ex. 3

and must be continued until the deviation between assumed and calculated values becomes smaller than the desired tolerance.

This is the n -th stage calculation, and by continuing this procedure from the first stage to the final stage we can construct manually the sequential stage flowgraph shown in Fig. 5. This flowgraph represents the solution of the problem of finding x_{An} if data R_0, x_{A0}, x_{C0} and the number of stage N are given.

Nomenclature

A	= area of pipe	[m ²]
D	= diameter of pipe	[m]
DC	= distribution curve	[—]
E	= extract stream	[kg/s]
f	= friction factor	[—]
NF	= number of flowgraphs	[—]
NT	= number of tearing variables	[—]
NU	= number of unused equations	[—]
u	= velocity of fluid	[m/s]
R	= raffinate stream	[kg/s]
Re	= Reynolds number	[—]
S	= solvent stream	[kg/s]
SCR	= solubility curve in raffinate side	[—]
SCE	= solubility curve in extract side	[—]
w	= mass flow rate	[kg/s]
x	= composition of raffinate	[kg/kg]
y	= composition of extract	[kg/kg]
y	= defined in Eq. (1-6)	[—]

<Subscripts>

A	= solute
B	= diluent
C	= solvent
n	= stage number

Literature Cited

- 1) Rao, C. Raja and C. Venkata Rao: *J. Appl. Chem.*, **6**, 269 (1956).
- 2) Yoshifuku I.: *J. Chem. Eng. Japan*, **17**, 309 (1984).
- 3) Yoshifuku I.: *J. Chem. Eng. Japan*, **21**, 538 (1988).