Subscripts

a	= index for arrangement
c	= condenser
C,Cs	= heat sink stream
D	= top product
F	= feed
H,Hs	= heat source stream
i,i_D,i_k,i_W	= component number
j	= sequence number
k,k'	= column number
r	= reboiler
W	= the bottoms
o	= surroundings

Superscripts

max	= maximum
min	= minimum
UB	= upper bound

LITERATURE CITED

Fitzmorris, R. E., and R. S. H. Mah, "Improving Distillation Column Design Using Thermodynamic Availability Analysis," *AIChE J.*, **26**(2), 265 (1980).

Freshwater, D. C., "Thermal Economy in Distillation," *Trans. Inst. Chem. Eng.*, **29**, 149 (1951).

Mah, R. S. H., J. J. Nicholas, Jr., and R. B. Wodnik, "Distillation with Secondary Reflux and Vaporization: A Comparative Evaluation," *AIChE J.*, **23**(5), 651 (1977).

Mah, R. S. H., and R. B. Wodnik, "On Binary Distillations and Their Idealizations," *Chem. Eng. Comm.*, 3(2), 59 (1979).

Naka, Y., M. Terashita, S. Hayashiguchi, and T. Takamatsu, "An Intermediate Heating and Cooling Method for a Distillation Column," *J. Chem. Eng. Japan*, **13**(2), 123 (1980).

Naka, Y., and T. Takamatsu, "Design Method of Heat Exchanger Systems with Recovery and Reuse of Steam Condensate," Proceedings of Meeting of Technology and Chemical Engineering for Saving Resource and Energy, Ohita (1979).

Null, H. R., "Heat Pumps in Distillation," *Chem. Eng. Prog.*, **72**(7), 58 (1976).

Rathore, R. N. S., K. A. Van Wormer, and G. J. Powers, "Synthesis Strategies for Multicomponent Separation Systems with Energy Integration," *AIChE J.*, **20**(3), 491 (1974a).

Rathore, R. N. S., K. A. Van Wormer, and G. J. Powers, "Synthesis of Distillation System with Energy Integration," *AIChE J.*, **20**(5), 940 (1974b).

Umeda, T., J. Itoh, and K. Shirako, "Heat Exchange System Synthesis by Thermodynamic Approach," 2nd PAChEC (Pachec '77), Denver, 28 (Aug., 1977).

Umeda, T., K. Niida, and K. Shiroko, "A Thermodynamic Approach to Heat Integration in Distillation Systems," *AIChE J.*, **25**(3), 423 (1979).

Manuscript received March 9, 1981; revision received November 24, and accepted January 13, 1982.

Solution of Recycle Problems in a Sequential Modular Approach

Recycle problems in a sequential modular approach are solved by adopting evolutionary models. When describing a process scheme by means of evolutionary models, a nonlinear algebraic system of equations is obtained. The resulting solution is utilized for updating the values of torn variables in the iterative solution of the scheme.

S. J. PIERUCCI, E. M. RANZI, and G. E. BIARDI

Chemical Engineering Department
Politecnico
Milan, Italy

SCOPE

The combination of a modular sequential approach, as a basic technique for the solution of process schemes, with an equation-oriented approach as convergence promoter for the solution of the recycle problems, has been successfully experimented by the authors. The basic idea is to define approximated, but evolutionary models associated to the rigorous ones adopted inside the process scheme. The evolutionary aspect is given by a set of internal, adaptive parameters, which are redefined after each iteration in order to fit the performance of the rigorous models.

The introduction of these new models adds a parallel simulation problem to the original one, that is the simulation of the same process scheme, where each unit is described by an evolutionary model. The solution is carried out according to an equation-oriented approach. The system of equations is properly decomposed and, then, solved by means of existing algorithms for the solution of nonlinear equations systems. As the evolutionary models give an approximated image, yet coincident with the corrected one at the convergence, the solution of the approximated scheme is seen as the prediction of the iteration variables in the original scheme.

As a consequence of the equation-oriented nature of the convergence promoter, all benefits intrinsic to the approach are present in the method: efficiency, stability and possibility of treating both simulation and design problems in the same manner.

CONCLUSIONS AND SIGNIFICANCE

The introduction of evolutionary models transforms the problem of promoting the convergence in a cyclical process scheme into the solution of an algebraic, nonlinear system of equations. The system is of a large scale but shows a sparse structure. Therefore, it may be easily decomposed following the results of the process scheme decomposition. A system with a

Correspondence concerning this paper should be addressed to S. J. Pierucci.

0001-1541-82-6217-0820-$2.00.

reduced number of unknowns is obtained. As a consequence, all the consolidated, general methods and algorithms available to the simulator for the solution of nonlinear systems may be adopted.

The proposed approach proves satisfactorily reliable and efficient so that the user can face process schemes, even when the presence of detailed models discourages the adoption of the other convergence promoters.

INTRODUCTION

Flowsheeting programs, during the last decade, have grown so much and their use has become so wide that one can fairly affirm that nowadays the overall problem of chemical process calculations must be regarded and met with the view of flowsheeting programs.

The above statement may be true also from the didactic viewpoint and corresponds to one of the most relevant revolution in thinking and operating in the chemical engineering area of the last half of this century.

Among the various aspects of the problem, one draws our special attention, which relates to convergence procedures and, hence, to numerical algorithms adopted within such simulation programs. As all of us know, a problem arises when a cyclical net occurs within the process flowsheet, which happens to be very common in the chemical processes. In fact, most chemical processes are characterized by material and energy recycles; this aspect requires the equation system representing the process units belonging to the recycles to be treated simultaneously. Since the outputs of the units of the cycle are logically, by a feedback mechanism, the input of all units of the sequence, it is not possible to proceed with the calculation of units in any order.

A commonly adopted approach for solving the obtained system of equations, is the so-called "sequential modular approach."

Accordingly, the process is decomposed by assuming the values for certain stream variables (torn stream variables), in such a manner that a sequential calculation becomes possible. After each complete calculation of the units contained inside the sequence is made, a new guess of the values for the assumed variables must be provided. This iterative procedure is repeated until a satisfactory agreement is reached among the values of the system variables in two successive iterations. The definition of an efficient, stable, quick method of promoting the convergence in the above synthetized procedure is commonly named "the solution of recycle problems." An anthitetic approach is called "equation-oriented" approach in the sense that it puts together all equations pertaining to the models of the units encountered within the loop. The system is solved simultaneously by means of a Newton-Raphson or similar numerical procedure.

The two so defined basic approaches are generally quite different in their premises, even if they can be considered complementary to each other in many situations. Both of them have their advantages and drawbacks. But, before entering into a deeper analysis we shall restrict our attention to what we consider the core of the problem: the computing time to be devoted to the simulation of complex chemical units.

Whenever in a cyclical net there is one or more of such units, let's say chemical reactors, complex fractionating units, one faces a problem which is largely dominated by the time needed to solve those units. In fact, there are almost rigid bounds on the number of times the model of a unit can be solved, regardless of the method adopted to reach convergence. Yet, a numerical procedure for the solution of recycle problems must be conceived and set up so that it may be compatible with the previously quoted computing time bounds. Moreover such compatibility can, or must, concern also the simulation models of the complex units, their mathematical structure and their description capability.

Nowadays, it is no longer a matter of linear or nonlinear models. Generally, we know how to solve nonlinear systems of equations and we have reliable algorithms and subroutines to perform such calculations. The qualitative distinction between linear and nonlinear models still applies, but it is losing its original importance. As a matter of fact, the qualitative differences in computing times constitutes the most relevant feature of the problem.

Also the numerical strategy to be adopted for the solution of recycles must be chosen having in mind this particular viewpoint.

Coming back to the sequential modular approach, the convergence is promoted by means of algorithms of *a local predictive type* (Johnson et al., 1960; Vašek et al., 1974; Futterer and Shlosser, 1974; Franks, 1972).

Thus, the iteration variables are updated on the basis of discrepancies located on the torn streams which they belong to. The reliability or "robustness" of such methods diminishes, as well known, with the increasing of the complexity of the scheme to be solved.

Experience shows that these methods are simple, moderately sure, rather flexible but slow. Consequently, they result very often in a number of iterations and in computing time amounts one cannot tolerate for all but the simplest problems. Among the advantages of the simplicity of such methods it is worth quoting the close control of the numerical procedure. Should an error or an ill-conditionment be met within the iterative loop, one is able to locate it very easily without any problem of troubleshooting.

Some attempts to utilize *global predictive methods* have been made in the direction of referring to linear systems which are able to update contemporaneously all the iteration variables (Navigev, 1957; Rosen, 1964; De Leeuw den Bouter, 1974; Dente et al., 1974).

The approach is undoubtedly more convenient than the previous one in respect to the ability of describing the entire process behavior. On the contrary, as far as the reability of the method is concerned, the assumptions made for the linearization of the original system (global split factors are often utilized for this purpose) do not contribute in lowering the risk of wasting large amounts of computing time when trying to solve schemes with highly sophisticated and detailed models.

In order to overcome the gap caused by the sequential modular approach, an antithetic approach, namely an *"equation-oriented"* one, is often tried. The whole system of equation related to the recycle problem is solved simultaneously by means of appropriate techniques, most of them of Newton type (Kubiček et al., 1975; Ketchum, 1975; Umeda and Niida, 1975; Pierucci et al., 1975). A fast convergence is warranted in all cases where the initial guess of all the unknown variables is close to the solution, provided that it exists. A new problem, however, arises from the rigidity of the approach in terms of an *a priori* choice of the set of equations to be solved. This aspect very often causes thermodynamic inconsistencies precluding the existence of a solution of the chosen system.

Moreover, it is practically impossible to have a reasonable initial guess to start the numerical procedure without recurring to a classical sequential approach. Also in this case one has to meet with the inconveniencies due to having errors concentrated on the torn variables. Furthermore, if the numerical procedure fails, it is rather difficult to locate and diagnose the error source. A classic and common situation may be mentioned to illustrate such a sentence: it is generally uneasy to fix *a priori* the equilibrium situation of a process stream that is to predict whether a single-phase or a vapor-liquid equilibrium exists. Nevertheless, the sequential solution of the units within a cyclical net permits to choose contin-

uously the correct situation based on the local values of thermodynamic state variables. Such a possibility is practically forbidden for an equation-oriented method governing simultaneously all variables in a fixed scheme of equations. The procedure stops and it is still difficult to locate the thermodynamic inconsistency, which demands a change in the structure of the system of equations.

In order to cope with such problem with good results, taking advantage of the benefits of both methods and respecting the computing time bounds, it becomes necessary to gain degrees of freedom which can be found in the structure of mathematical models of the process units only. Therefore, we look at two classes of models one beside the other. The first, or original, is related to the most detailed models adopted for simulation purposes and constitutes a fixed reference for the second. The second, which will be referred to as evolutionary, is made up of simplified models, still nonlinear because they are established on physical grounds, under a few limited assumptions. This class of models is employed as convergence promoter and contains effective and adjustable parameters whose determination is left to the simulation runs with the original models.

The global solution procedure consists of two interrelated phases.

The first deals with an iteration with the original models according to a sequential modular approach. During this phase, the internal parameters of the evolutionary models are to be estimated.

The second phase consists in solving the algebraic system of equations obtained by assembling all evolutionary models of the units in the cyclical net. Such a solution can be reached by means of standard subroutines for nonlinear systems after a proper decomposition has been applied. In principle, the obtained results update all stream variables in the process scheme, and consequently also those variables which are the iteration variables in the sequential modular approach.

The combination of the two phases allows convergence to be reached rather effectively. It is worth remarking that as convergence is obtained by the solution of a system of equations describing the process scheme, design problems may be treated in the same way, as shown later on.

CONVERGENCE PROMOTER

We will refer to a scheme of a plant which is constituted by a single maximum cyclical net according to Steward (1965). The presence of more than one maximum cyclical net would lead to a repeated application of the convergence promoter to each cycle.

The streams of the scheme will be called "internal streams" when connecting two of the units inside the scheme, or "external (feed or output) streams" when connecting one of the units with the outside part of the scheme.

Each stream is characterized by a set of variables which completely defines the component flowrates, temperature and pressure.

As it usually happens with such problems, pressures are assigned in each point of the scheme as problem data.

Temperatures and flowrates are the main unknowns of the problem. With the scope of the convergence promoter here proposed, we will only deal with components flowrates, according to a common practice. However, future work will consider the possibility of converging on temperature.

For sake of a better understanding, the following notation will be adopted hereafter:

$$\overline{X} \div \{X_{j,i}\}\ (i = 1, ncp;\ j = 1, ni)$$

represents the set of the molar flowrates of all components i in the internal streams j,

$$\overline{X}^o \div \{X^o_{j,i}\}\ (i = 1, ncp;\ j = 1, nf)$$

is the set of the molar flowrates of all components i in the feed streams j to the scheme.

The model of the unit K is thought as a system of equations m^K both in the component flowrates related to input/output streams, and in a set of internal parameters, which, from a general point of view, may be divided into:

- Internal unknown parameters (or variables) $\overline{Y}_k$
- Internal assigned parameters (or design parameters) $\overline{P}_k$

Therefore, the model may be symbolically represented by:

$$m^K(\overline{X}_e, \overline{X}_o, \overline{X}^o_f, \overline{P}_k, \overline{Y}_k) = 0$$

where: $\overline{X}_e \div \{X_{e,i}\}$, $\overline{X}_o \div \{X_{o,i}\}$, $\overline{X}^o_f \div \{X^o_{f,i}\}$
While the subscripts indicate:

e = all input internal streams to the unit
o = all output streams from the unit
f = all input feed streams to the unit
$i = 1, \ldots, ncp$ all the components
$\overline{P}_k \div \{P_{k,g}\}$ = set of internal design parameters $(g = 1, kn_k)$
$\overline{Y}_k \div \{Y_{k,r}\}$ = set of internal unknown variables $(r = 1 \ldots iv_k)_K$

Provided the values $\overline{X}_e$, $\overline{X}^o_f$, $\overline{P}_k$ are known, then the system m^K may be solved for the unknowns $\overline{X}_o$ and $\overline{Y}_k$.

According to the sequential modular approach, the simulation of the whole scheme is run after assuming the values $\overline{X}^o$ and $\overline{P}_k$, $(k = 1 \ldots nu)$.

The present way of assigning data, which corresponds to the solution of a "pure simulation" problem, may be called as "standard type."

The units of the scheme are solved iteratively according to a sequence of solution which implies the definition of a suitable set of torn streams.

Let be

$$\overline{Z} \div \{Z_i\}\ i = 1, \ldots, ne$$

the set of iteration component flowrates associated to the torn streams, and

$$S\ \{m^1, m^2, m^3, \ldots, m^K, \ldots m^{nu}\}$$

the corresponding sequence of solution of the units.

An iteration begins by assigning numerical values to the set $\overline{Z}$: this permits a sequential solution of the units according to S. When all the sequence S is completed (end of the iteration) then a calculated value is attributed to each flowrate $\overline{X}$ of the internal streams, and, in particular also to those flowrates belonging to the set $\overline{Z}$. Let be $T(\overline{Z})$ their values, then they are equal to $\overline{Z}$, namely

$$\overline{Z} - T(\overline{Z}) = 0$$

when $\overline{Z}$ is the solution of the problem.

The residuals of the foregoing equations may be considered as an error measure of the iterative procedure, even if, in practice, more efficient measures may be found, as for instance: the sum of the squares of weighted differences between the values $\overline{X}$ in two successive iterations. Nevertheless, the error measure plays the role of an index whose value indicates that either a further iteration has to be restarted, or the calculation is concluded.

Therefore, the sequential modular approach to the solution of the scheme may be summarized as follows:

i) Assign the standard type data of the problem; namely, values to the sets $\overline{X}^o$ and $\overline{P}_k$ $(k = 1, nu)$ Guess the values $\overline{Z}$ as first trial.
ii) Solve the sequence S.
iii) Define an error measure at the end of the calculation.
iiii) Restart the sequence from ii) with a new guest of $\overline{Z}$ if a nonacceptable error measure is found. Otherwise, the calculation is concluded.

The phase of updating the values of the torn variables $\overline{Z}$ is undoubtedly the determinant one in defining the economy of the whole calculation. The way proposed by this paper is based on the assumption that each model m^K may be represented by an ap-

proximated, yet evolutionary, one ev^K according to the following definition:

$$ev^K = \begin{cases} X_{o,1} = W^K_{o,1}(\bar{X}_e, \bar{X}^o_f, \bar{\pi}_k, \bar{\delta}_k) \\ X_{o,2} = W^K_{o,2}(\bar{X}_e, \bar{X}^o_f, \bar{\pi}_k, \bar{\delta}_k) \\ \quad \vdots \\ X_{o,ncp} = W^K_{o,ncp}(\bar{X}_e, \bar{X}^o_f, \bar{\pi}_k, \bar{\delta}_k)\ (o = 1, no_k) \\ \bar{C}^K(\bar{X}_o, \bar{X}_e, \bar{X}^o_f, \bar{\pi}_k, \bar{\delta}_k) = 0 \end{cases}$$

or more concisely:

$$ev^K = \begin{cases} \bar{X}_o = \bar{W}^K_o \\ \bar{C}^K = 0 \end{cases}$$

where:

$$\bar{W}^K_o \div \{W^K_{o,i}\},\ \bar{C}^K \div \{C^K_l\}$$
$$\bar{\pi}_k \div \{\pi_{k,l}\},\ \bar{\delta}_k \div \{\delta_{k,n}\}$$

where

$\pi_{k,l}$ = internal unknown variables $(l = 1, \ldots, un_k)$

$\delta_{k,n}$ = internal evolutionary design parameters $(n = 1, \ldots, ed_k)$

$W^K_{o,i}$ and C^K_l whichever algebraic functions

$(i = 1, \ldots, ncp;\ o = 1, \ldots, no_k;$ and $l = 1, \ldots, un_k)$

Variables $\bar{\pi}_k$ may be of whatever nature. They, very often, coincide with some or all of the corresponding $\bar{Y}_k$ belonging to the original model m^K. As regards the parameters $\bar{\delta}_k$, they give the evolutionary aspect to the model ev^K as their values in principle, depend on the performance of the original models. In other words, they are adaptive parameters by which an evolutionary model fits the behavior of the original one. Therefore, these parameters, in general, are updated during each iteration on the basis of the results obtained by the solution of the original models.

In summary, the model ev^K consists of a system of $(no_k \cdot ncp + un_k)$ equations in the unknown X_o and $\bar{\pi}_k$. Provided the values $\bar{X}_e$, $\bar{X}^o_f$ and $\bar{\delta}_k$ are assigned, the system may be solved to give the values of the unknowns.

Of course, the model ev^K depends, both on the type of the unit, or model m^K, which refers to and on the accuracy of the prediction required by updating the values of the torn variables. In fact, the closer the evolutionary models are to the original ones m^K, the more accurate is the prediction of the iteration variables.

Two points, however, are worthwhile mentioning: first, the structure of the evolutionary models renders the output variables $\bar{X}_o$ explicit in the remaining variables of the model; second, no *a priori* limitation exists to the structure of the functions $W^K_{o,i}$ and O^K_f.

The introduction of the models ev^K associates a parallel simulation problem to the original one; that is, the simulation of the same scheme where the equipment is described by evolutionary models. The solution of this problem is an approximated image (but coincident at the convergence) of the solution of the original problem, so that the obtained results may be utilized in promoting the convergence of the original simulation problem.

The system of equations to be solved has a structure that affords using current algorithms for the solution of nonlinear equations systems.

In fact, the following decomposition of the system may be carried out:

- The set $\bar{X}_o$ within each evolutionary model ev^K, is an "admissible output set" (Steward, 1965) of equations $\bar{W}^K_o$
- A tearing variable set is defined containing $\bar{Z}$ and $\bar{\delta}_k$ $(k = 1, \ldots, nu)$
- The convergence equations form the following system:

$$\begin{cases} \bar{Z} - T(\bar{Z}) = 0 \\ \bar{C}^K = 0 \end{cases} \quad (K = 1, \ldots, nu) \qquad (1)$$

$T(\bar{Z})$ indicate the calculated values of $\bar{Z}$. They are obtained by the solution of all the subsystems $\bar{W}^K_o$ according to the sequence $S(\bar{W}^1_o, \bar{W}^2_o, \bar{W}^3_o, \ldots, \bar{W}^{nu}_o)$. S is the sequence adopted when solving the original models. Now the models are substituted by the subsystems $\bar{W}^K_o$.

Practically the dimension of the problem is limited to acceptable figures since both the set of unknown parameters is small (un_k is very often ϕ, or few units) and the decomposition of the original scheme generally is carried out under the assumption of minimizing the number of torn streams.

As a consequence, neither further decomposition of the system nor special techniques for finding the solution are required; in other words, the problem is viewed in terms of the solution of a nonlinear system by means of existing and consolidated algorithms.

In summary, the sequential modular approach to the simulation of a scheme when the convergence is promoted by means of evolutionary models, contains the following steps:

i) Assign standard type data of the problem; namely, values to sets $\bar{X}^o$ and $\bar{P}_k$ $(k = 1, nu)$. Guess as first trial the values for $\bar{Z}$.

ii) Solve sequence $S(m^1, m^2, m^3, \ldots, m^{nu})$ and calculate parameters $\bar{\delta}_k$ of each evolutionary model.

iii) Define an error measure at the end of the calculation.

iiii) If the error is lower than acceptable value, the calculation is concluded; otherwise, update $\bar{Z}$ as follows:
Solve the nonlinear system

$$\begin{cases} \bar{Z} - T(\bar{Z}) = 0 \\ \bar{C}^K = 0 \end{cases}$$

The unknowns are $\bar{Z}$ and $\bar{\pi}_k$ $(k = 1, nu)$; the values $T(\bar{Z})$ are obtained by the sequence $S(\bar{W}^1_o, \bar{W}^2_o, \bar{W}^3_o, \ldots, \bar{W}^K_o, \ldots, \bar{W}^{nu}_o)$. Restart sequence from point ii).

EXTENSION OF EVOLUTIONARY MODELS TO DESIGN PROBLEMS

A design problem occurs when some degrees of freedom of the scheme are saturated by means of "nonstandard" data. This means that some or all the quantities belonging to $\bar{X}^o$ and $\bar{P}_k$ are considered as unknowns of the problem. Their values have to be found by assigning an equal number of specification equations. They concern performances of equipments, product purities, flowrates, etc. Now the solution of the problem shows besides the original material recycles, also a set of information recycles. They are originated by the presence of unknowns and related specification equations, located in different points of the scheme.

A decomposition of such a scheme would assign as iteration variables those related to the pure simulation problem and, additionally, those proper of the design problem.

As a consequence, the design problem may be approached by the same sequence of steps defined for pure simulation problem, provided the convergence promoter is capable of updating the values of all iteration variables.

The present convergence promoter may be applied to this type of problem, without introducing any substantial modification. This happens if the additional unknowns and specification equations introduced by the design problem refer to the set of variables and evolutionary parameters treated by the evolutionary models. Therefore, being $\bar{\pi}$ and $\bar{\delta}$ the whole sets of internal variables and parameters associated to the evolutionary models, the following assumptions have to be satisfied in order to apply the proposed method:

- The set $\bar{s} \div \{s_p\}$ $(p = 1, \ldots, ns)$ of additional variables belongs either to the set $\bar{X}^o$ or to $\bar{\delta}$
- The ns additional specification equations are of the type:

$$\bar{d} \div d^p\ (\bar{X}^o, \bar{X}, \bar{\pi}, \bar{\delta}) = 0\ (p = 1, \ldots, ns)$$

where d^p is any algebraic function.

The convergence promoter is then applied by considering $\bar{Z}$, $\bar{\pi}$ and $\bar{s}$ as tearing variables, while the corresponding convergence equations define the following system:

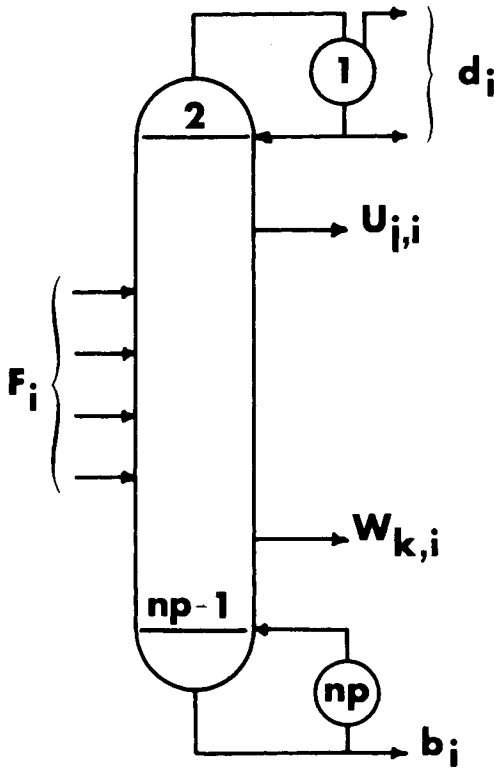

Figure 1. Distillation column.

$$\begin{cases} \overline{Z} - \overline{T}(\overline{Z}) = 0 \\ \overline{C}^K = 0 \\ \overline{d} = 0 \end{cases} \tag{2}$$

The solution of the above system is carried out by the same techniques and procedures described when dealing with the pure simulation problem. The obtained solution is then utilized in promoting the convergence of the problem.

EVOLUTIONARY MODELS

We are concerned here with the description of models commonly used to solve simulation problems. The units which are usually involved in such problems deal with mixers or splitters of streams, heat exchangers, reactors, one step liquid-vapor equilibrium separators (flash drums), countercurrent two-phases unit (distillation or absorber columns).

The evolutionary models associated with them are not in principle unique, due to the fact that they have so general and flexible structure to allow different possibilities of description.

Therefore, the next evolutionary models have to be considered in terms of results of the experience gained by the authors in this field.

Mixer (two input streams 1 and 2 give the output stream 3)

$$ev: X_{3,i} = X_{1,i} + X_{2,i} \quad (i = 1, ncp)$$

Splitter (one input stream 1 is splitted into streams 2 and 3)

$$ev: \begin{matrix} X_{2,i} = \lambda X_{1,i} \\ X_{3,i} = (1 - \lambda) X_{1,i} \end{matrix} \quad (i = 1, ncp)$$

Where λ is an evolutionary parameter

Heat exchanger [one stream (1,3) exchanges with stream (2,4)]

$$ev: \begin{matrix} X_{3,i} = X_{1,i} \\ X_{4,i} = X_{2,i} \end{matrix} \quad (i = 1, ncp)$$

Reactor (one input stream 1 gives output stream 2)

$$ev: X_{2,i} = X_{1,i} + \sum_{1=j}^{NR} \nu_{ij} R_j \quad (i = 1, ncp)$$

where R_j is an evolutionary parameter

Flash drum (one input stream 1 is flashed in the vapor stream 2 and the liquid stream 3)

$$ev: \begin{cases} X_{2,i} = K_i X_{1,i}(1 - \alpha)/(\alpha + K_i(1 - \alpha)) (i = 1, ncp) \\ X_{3,i} = X_{1,i}\, \alpha/(\alpha + K_i(1 - \alpha)) \\ \alpha = \begin{cases} \alpha^* \text{ for } 0 \le \alpha^* \le 1 \\ 0 \text{ for } \alpha^* < 0 \\ 1 \text{ for } \alpha^* > 1 \end{cases} \\ C = \sum_{1=i}^{NCP} \dfrac{X_{1,i}(1 - K_i)}{\alpha^* + K_i(1 - \alpha^*)} = 0 \end{cases}$$

where

K_i = evolutionary parameters
α^* = an internal parameter associated to the convergence equation C

The possibility for α^* of having nonphysical values (<0 or >1) allows the model to be solved even when a single phase is present.

DISTILLATION COLUMN

The equations of the theta convergence promoters (Holland, 1963) proved able to describe the evolutionary model. According to Figure 1, the subscript m refers to the values calculated by the original model. Thus, the evolutionary model ev is defined as follows:

$$\begin{aligned} d_i &= R_i(d_i)_m \\ b_i &= \theta R_i(b_i)_m \\ U_{j,i} &= \theta_j^U R_i(U_{j,i})_m \quad j = 2, \ldots, np - 1 \\ W_{k,i} &= \theta_k^W R_i(W_{k,i})_m \quad k = 2, \ldots, np - 1 \end{aligned}$$

where:

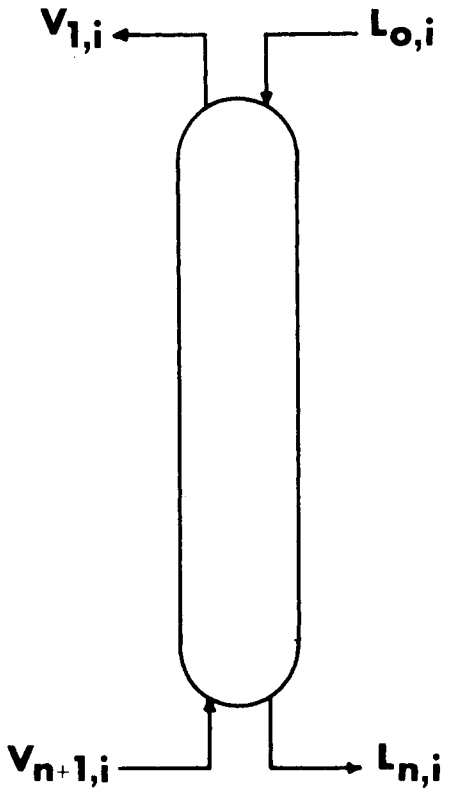

Figure 2. Absorber.

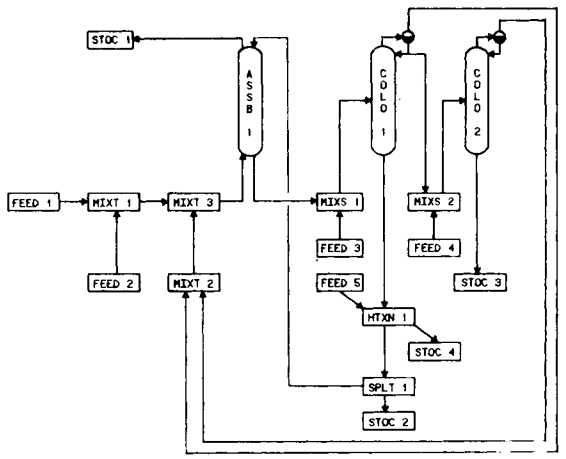

Figure 3. LPG scheme.

$$R_i = \frac{F_i}{(d_i)_m + \theta(b_i)_m + \theta_j^U(U_{j,i})_m + \theta_k^W(W_{k,i})_m}$$

$(d_i)_m$, $(b_i)_m$, $(U_{j,i})_m$, $(W_{k,i})_m$ are evolutionary parameters while θ, θ_j^U, θ_k^W are internal variables which are searched by means of an equivalent number of equations dealing with the performance of the column; i.e., flowrates of the products, their purities, and so on. The most common ones are related to the total flow of each output stream, namely:

$$C = \begin{cases} \Sigma d_i - D = 0 \\ \Sigma U_{j,i} - U_j = 0 \\ \Sigma W_{k,i} - W_k = 0 \end{cases}$$

where:

$$D, U_j, W_k = \text{assigned values}$$

ABSORBER

With reference to Figure 2, the evolutionary model is:

$$ev: \begin{cases} V_{1,i} = A_i V_{n+1,i} + B_i L_{0,i} \\ L_{n,i} = (1 - A_i)V_{n+1,i} + (1 - B_i)L_{0,i} \end{cases}$$

A_i and B_i evolutionary parameters defined as follows:

$$A_i = 1/(1 + S_{1,i}S_{2,i}S_{3,i} \ldots S_{n,i} + S_{2,i}S_{3,i} \ldots S_{n,i} + \ldots S_{n,i})$$

$$B_i = A_i(1 + S_{2,i}S_{3,i}S_{4,i} \ldots S_{n,i} + S_{3,i}S_{4,i} \ldots S_{n,i} + \ldots S_{n,i})$$

where

$$S_{j,i} = \frac{L_j}{K_{j,i}V_j}$$

EXAMPLES

The convergence promoter has been successfully implemented into a revised version of the simulation program SFINGE (Pierucci et al., 1976). To this end, the algebraic system of equations, generated by the evolutionary models, has been represented, inside the program, by means of a technique commonly used by "equation oriented" flowsheeting programs (for instance, Pierucci et al., 1975). Basically the algorithm adopted for the numerical solution of the resulting system is a combination of different algorithms which are entered in turn according to the strategy described by Buzzi and Casapollo (1972).

Several case studies of industrial interest were run in order to verify the reliability of the proposed method; the results confirmed the benefits of adopting such an approach in terms of flexibility, stability and convergence rate.

A typical "LPG scheme" is reported, as an example, in Figure 3.

Each unit is identified by a name and an index which differentiates units having the same name. In fact the single name is able to classify the performance of the unit as:

FEED: dummy unit indicating an input stream to the scheme
STOC: dummy unit indicating and output stream from the scheme
MIXT: adiabatic mixer
MIXS: non adiabatic mixer
HTXN: heat exchanger
SPLT: isothermal splitter
ASSB: absorber with theoretical trays
COLO: distillation column with theoretical trays

Table 1 reports the values of component flowrates, temperatures and pressures of each input stream to the scheme.

Table 2 reports the design parameters and further specifications for each unit.

The pressure of each stream, except those where it has been properly specified, is equal to 18.1E + 5 Pa.

Physical properties have been calculated by means of a Chao-Seader model for hydrocarbon mixtures; polynomial expressions were used in estimating water properties. All data have been collected from the internal data bank of SFINGE program.

In order to find a minimum set of torn streams, a decomposition of the scheme was carried out automatically according to the algorithm proposed by Pierucci et al. (1973); the stream entering COLO1 was elected as unique stream of the searched tearing set.

Table 3 reports the convergence behavior for four iterations: the figures under the column Z indicate the first guess of the flowrates in the torn stream; while $T(Z)$ indicates the calculated values in the same iteration. The figures under "predicted" are those predicted by the present convergence promoter: these values are the new guess for the next iteration.

Table 3 also reports an error measure, defined automatically by the program on the basis of the values of all the variables in two successive iterations.

The amount of computing time for the whole calculation was about 680 second on an UNIVAC 1100/60 computer; of that amount, only 1.5 s was spent by the convergence promoter.

The same example has been solved in terms of design problem by considering the split factor of unit SPLT1 as unknown. The related specification equation was related to the purity of normal pentane in the oil leaving the absorber, namely the request was of a molar fraction of 0.3.

In order to put into evidence the information recycle generated by the nonstandard data, the scheme of Figure 3 was modified in Figure 4.

The unit RILE1 is a dummy unit whose task is both to calculate the discrepancy of the specification equations, and to introduce

TABLE 1. FEED STREAMS TO LPG SCHEME

Flowrates (mol/time)	Feed 1	Feed 2	Feed 3	Feed 4
CO_2	.00000	.00000	.00000	.61000
H_2	11.190	2.2400	.00000	.00000
CH_4	7.1700	5.4000	.00000	.13000
C_2H_6	11.440	38.650	4.8900	5.8300
H_2S	.00000	.00000	.00000	.82000
C_3H_3	8.5500	28.320	7.7900	40.050
C_4H_{10}	3.1900	2.5500	41.080	69.500
C_5H_{12}	.00000	.00000	259.79	.00000
2-Met. Pentane	.00000	.00000	64.770	.00000
3-Met. Pentane	.00000	.00000	27.410	.00000
C_6H_{14}	.00000	.00000	45.200	.00000
M. CY. Pentane	.00000	.00000	4.1600	.00000
Benzene	.00000	.00000	2.2200	.00000
Cycle-Hexane	.00000	.00000	1.1300	.00000
Total	41.540	77.160	458.44	116.94
Temperature (K)	311.	311.	403.	338.
Pres. CPa*E-5)	21.13	29.17	18.10	29.21

TABLE 2. DESIGN PARAMETERS AND INFORMATION FOR LPG SCHEME

MIXT 1	Adiabatic mixer
MIXT 2	Adiabatic mixer
MIXT 3	Adiabatic mixer
MIXS 1	Nonadiabatic mixer. Output temperature assigned is 403 K.
MIXS 2	Nonadiabatic mixer. Output temperature assigned is 338 K.
HTXN 1	Heat Exchanger. Cooling water at 293 K and 4.13E + 5 Pa. A maximum temperature increment of 10 K in the output stream. Find the total water flowrate in order that the output temperature of hot stream is 308.3 K.
SPLT 1	Isothermal Splitter. A split of 25% for the oil entering the absorber.
ASSB 1	Absorber: 18 theoretical Trays. Constant pressure of 18.1 E + 5 Pa.
COLO 1	Distillation Column: 24 Theroretical Trays (Condenser and reboiler included). Condenser at 18.E + 5 Pa, constant pressure drop per tray of 1,723 Pa. Feed tray is the 12 from the top. The condenser provides a distillate of 14.413 mol/time in the vapor phase and 127.70 mol/time in the liquid phase. The liquid reflux to the column is 589.42 mol/times.
COLO 2	Distillation column: 14 Theoretical Trays (Condenser and Reboiler Included). Condenser at 29.17E + 5 Pa, constant pressure drop per tray of 1,516 Pa. Feed tray is the 4 from the top. The condenser provides a distillate of 57.572 (mol/time) in the vapor phase and a liquid reflux to the column of 69.574 mol/time.

TABLE 3. CONVERGENCE BEHAVIOR FOR SIMULATION PROBLEM

	ITER.1 E = 7.93		ITER.2 E = 0.040		ITER.3 E = 0.001		ITER.4 E = 0.0003	
	Z	T(Z)	Predicted	T(Z)	Predicted	T(Z)	Predicted	T(Z)
BIOS	0.0	0.0737	0.1657	0.1715	0.1804	0.1801	0.1790	0.1799
IDRO	0.0	0.2097	0.4207	0.3716	0.3858	0.3821	0.3797	0.3835
META	0.0	0.8424	1.782	1.714	1.777	1.768	1.756	1.769
ETAN	10.0	19.80	39.79	41.38	43.30	42.96	42.60	42.78
IDRS	0.0	0.1977	0.5052	0.5289	0.5514	0.5407	0.5317	0.5344
PROP	30.0	49.17	53.77	59.54	59.04	60.22	60.68	60.73
NBUT	40.0	59.17	56.46	52.56	52.97	52.56	52.66	52.66
NPEN	200.0	282.58	318.77	327.27	330.23	330.69	330.76	330.85
2M5A	60.0	76.20	82.57	83.77	84.20	84.25	84.26	84.27
SM5A	20.0	31.31	35.06	35.52	35.69	35.71	35.72	35.72
NESA	30.0	51.24	58.11	58.77	59.02	59.05	59.05	59.06
MCIC	0.0	4.16	5.344	5.406	5.429	5.432	5.432	5.432
BENZ	0.0	2.220	2.818	2.865	2.882	2.883	2.884	2.884
CICE	0.0	1.130	1.458	1.473	1.478	1.479	1.479	1.479

TABLE 4. CONVERGENCE BEHAVIOR FOR DESIGN PROBLEM

	ITER.1 E = 180.0		ITER.2 E = 41.0		ITER.3 E = 0.637		ITER.4 E = 0.014		ITER.5 E = 0.0002	
	Z	T(Z)	Predict	T(Z)	Predict	T(Z)	Predict	T(Z)	Predict	T(Z)
BIOS	0.0	0.07376	0.0844	0.16545	0.19073	0.16962	0.16404	0.15923	0.15793	0.15711
IDRO	0.0	0.2097	0.21462	0.45211	0.46025	0.33608	0.33164	0.33164	0.33165	0.33165
META	0.0	0.8424	0.90853	1.8369	1.9818	1.6235	1.5789	1.5523	1.5491	1.5403
ETAN	10.0	19.80	22.766	38.256	48.252	41.312	38.962	38.647	38.545	38.441
IDRS	0.0	0.1977	0.26126	0.5009	0.66446	0.52116	0.47047	0.4770	0.48284	0.48244
PROP	30.0	49.17	48.251	70,876	59.116	56.554	60.141	58.562	58.538	58.296
NBUT	40.0	59.17	56.918	55,936	53.739	52.003	51.631	51.902	51.873	51.887
NPEN	200.0	282.58	311.99	341.46	323.71	318.85	318.41	317.80	317.54	317.36
2M5A	60.0	76.20	84.798	89,397	81.887	81.225	81.271	81.198	81.154	81.136
3M5A	20.0	31.31	36.128	37.946	34.690	34.431	34.456	34.427	34.409	34.402
NESA	30.0	51.24	60.186	62.864	57.300	56.925	56.981	56.939	56.910	56.900
MCIC	0.0	4.16	5.5302	5.7815	5.2722	5.2369	5.2417	5.2380	5.2353	5.2343
BENZ	0.0	2.22	2.8817	3.0537	2.8043	2.7796	2.7808	2.7783	2.7767	2.7761
CICE	0.0	1.13	1.5155	1.5767	1.4341	1.4256	1.4272	1.4263	1.4256	1.4254
SPLIT	0.25		0.30545		0.21976		0.22204		0.22176	

an information recycle stream to the unit SPLT1.

Table 4 summarizes the results and predictions for five iterations: the reported figures have the same meaning of the corresponding ones in Table 3.

The amount of computing time for the whole calculation was about 700 while only about 2 s was spent by the convergence promoter.

LITERATURE CITED

Buzzi Ferraris, G., and A. Casapollo, "Risoluzione di sistemi di equazioni algebriche non lineari e loro applicazione a problemi di Ingegneria Chimica," *Ing. Chim. Ital.*, 8, 261 (1972).

De Leeuw den Bouter, J. A., and A. G. Swenker, "Tisflow, a flowsheet simulation program based on new principles," Proceedings of Int. Symp. on Computer Application in Process Development, *Erlangen (D)* (April, 1974).

Dente, M., G. Biardi, E. Ranzi, and S. Pierucci, "An executive program for the logical decomposition and the numerical convergence of process flowsheets," *Ibid.*

Futterer, E., and H. Schlosser, "Konvergenz probleme bei Anwendung modularer Simulations-programme," *Ibid.*

Holland, C. D., "Multicomponent distillation," Prentice Hall, Inc., Englewood Cliffs, NJ (1963).

Johnson, A. I., et al., "GEMCS manual and application studies," McMaster University, Ontario, Canada (July, 1960).

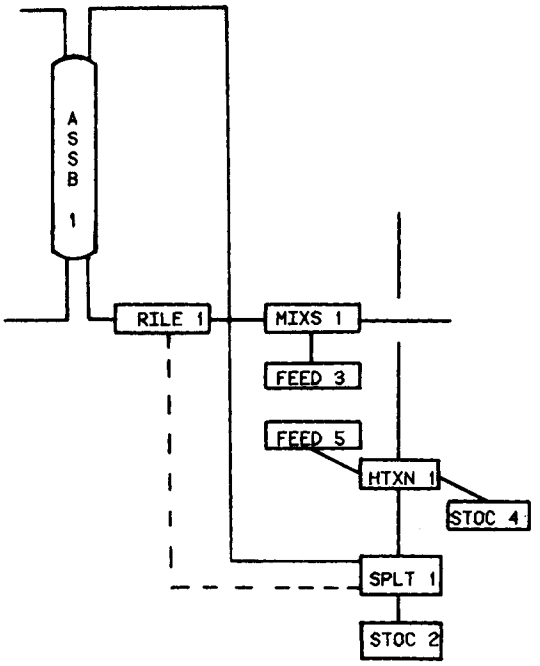

Figure 4. Information recycle in the LPG scheme of Figure 3.

Ketchum, R. G., "Computation of interlinked separation Columns for non ideal Mixtures," Proceedings of Int. Symp. on Computer 75, *Karlovy Vary (CS)* (Sept., 1975).

Kubiček, M., V. Hlavaček, and F. Pročhasca, "Global Modular Newton-Raphson technique for the simulation of an Interconnected plant Applied to Complex Rectification Columns," *Ibid.*

Nagiev, N. F., "Material balance in complex and multistage recycle Chemical Process," *Chem. Eng. Prog.*, **53**, 6, 297 (1975).

Pierucci, S., E. Ranzi, G. Biardi, and M. Dente, "Studi di analisi dei Sistemi dell'Ingegneria Chimica. Nota II: descrizione di un nuovo algoritmo per la risoluzione del problema del tearing," *Ing. Chim. Ital.*, **9**, 5 (1973).

Pierucci, S., G. Biardi, E. Gabrielli, and E. Ranzi," DIGMA: a new program for process flow-sheet calculations," Proceedings of Int. Symp. on Computers 75, *Karlovy Vary (CS)* (Sept., 1975).

Pierucci, S., G. Biardi, E. Ranzi, and M. Dente, "SFINGE: a chemical plants simulation programme," *Ing. Chim. Ital.*, **12**, 163 (1976).

Rosen, E. N., "A machine Computation method for performing material balances," *Chem. Eng. Prog.*, **58**, 10, 69 (1962).

Steward, D. V., "Partitioning and Tearing Systems of Equation," *J. Siam Numer. Anal.*, B, **2**, 2 (1965).

Umeda, T., and K. Niida, "Integration of process Design Systems," Proceedings of Int. Symp. on Computer 75, *Karlovy Vary (CS)* (Sept., 1975).

Vašek, V., M. Dohnal, J. Klemeš, and J. Lučha, "Methods of chemical Plant Balance Simulation Problem Solving," Proceedings of Int. Symp. on Computer Application in Process Development, *Erlangen (D)* (April, 1974).

NOTATION

C = whichever algebraic function
d = whichever algebraic function
ed = number of design parameters inside an evolutionary model
ev = evolutionary model
K = vapor liquid equilibrium constant
K_n = number of internal design parameters
iv = number of internal unknown variables to an unit
L = liquid component flowrate
m = system of equations representing the model of an unit
ncp = number of components
ne = number of iteration variables
nf = number of feed streams
ni = number of internal streams
no = number of output streams from an unit
ns = number of s-type variables
nu = number of units of the scheme
P = internal assigned parameters to an unit
R_j = extent of jth reaction
s = additional unknown defining a design problem
S = logical sequence of solution of the units inside the cyclica net
$T(\bar{Z})$ = calculated value of $\bar{Z}$ at the end of one iteration
un = number of unknown variables inside an evolutionary model
V = vapor component flowrates
W = whichever algebraic function
X = molar flowrate
Y = internal unknown variables of an unit
Z = component flowrate in a torn stream

Subscripts

e = input internal streams to an unit
f = input feed streams to an unit
g = internal assigned parameters
i = component
j = stream
k = number characterizing an unit
l = internal unknown variable in an evolutionary model
m = calculated value by the original model
n = internal evolutionary parameters
o = output streams from an unit
p = additional specification defining the design problem
r = internal unknown variable
s = iteration component flowrate

Superscripts

K = number characterizing an unit
o = feed stream to the scheme
p = additional specification equation defining a design problem

Greek Letters

α = liquid fraction
δ = evolutionary parameter
θ = theta parameter according to theta method
λ = split factor
$\nu_{i,j}$ = stoichiometric coefficient of component i in the jth reaction internal unknown variable in an evolutionary parameter

Manuscript received May 18, 1981; revision received October 6, 1981