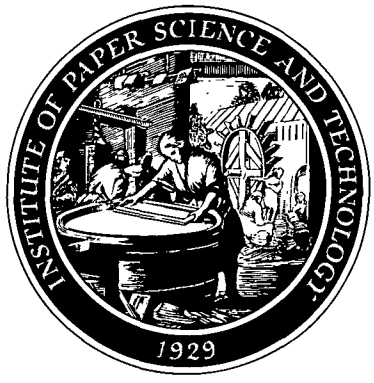
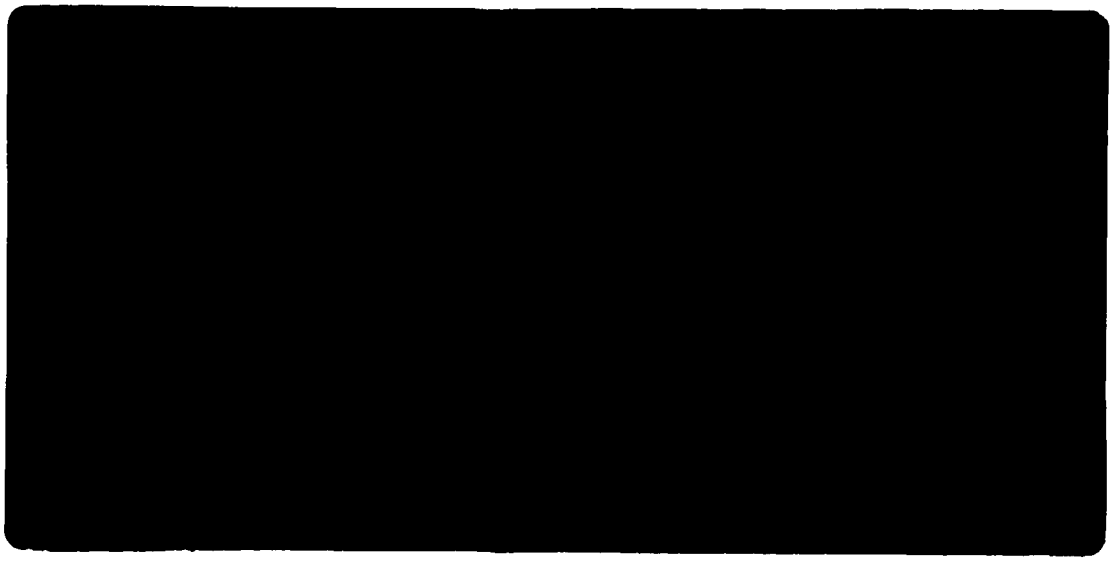


Institute of Paper Science and Technology
Atlanta, Georgia

IPST TECHNICAL PAPER SERIES



NUMBER 457

STRATEGIES FOR CONVERGENCE OF BLOCK SEQUENTIAL SIMULATORS

G.L. JONES

NOVEMBER 1992

Strategies for Convergence of Block Sequential Simulators

G.L. Jones

Submitted to
TAPPI Pulping Conference
Boston, Massachusetts
November 1992

TABLE OF CONTENTS **PAGE**

LIST OF TABLES **PAGE**

LIST OF FIGURES

ABSTRACT

The traditional method of converging the mass and energy flows in a sequential modular block simulation program is based on tear stream testing. Once tested the tear stream is directly substituted or is "accelerated" using secant or second order extrapolation methods. Tear stream testing is inconvenient for large flowsheets with many tear streams. Unless all tear streams are tested, tear stream testing does not assure a global mass or energy balance. A new approach based on the flowsheet topology or connectivity matrix and nodal mass balances has proved very effective for many situations. This alternative method called nodal-balance convergence is easier to use and is often more efficient than tear stream testing. If used properly, it can also assure global convergence of material and energy balances to a desired tolerance.

The various methods are compared and contrasted. Cases where each approach is preferred are discussed. Examples are shown comparing the efficiency of each method in difficult convergence situations.

INTRODUCTION

Mass and energy balance simulation programs are an important tool used by the engineer for a variety of tasks including process design, equipment sizing, process debottlenecking, product troubleshooting, process optimization and environmental abatement. At the heart of M&E balance programs are sets of algebraic and differential equations representing process and product models. The most important part of each unit operations model, the mass and energy balance, assures the closure of the mass and energy over that operation. However, the solution for a realistic process requires iteration over sets of simultaneous equations or the "inversion" of a matrix. The reason for this complication is that the simultaneous equations are coupled by the intermediate flows of mass and energy between the unit operations as shown in Fig. 1

The equations describing each unit operation or flowsheet block can vary widely depending on the function of the unit and are usually nonlinear for realistic process models. Ordinary differential equations resulting from transient effects, reaction kinetics and transport variations add further complexity. While conservation obviously still applies for dynamic systems, convergence at any given time is generally not strictly enforced. The reason for this is that process holdups are usually not monitored during the process of closing the balance. However, a true balance is achieved at the new steady state at which point the holdups become constant and the differential system reduces to an algebraic system. This more complex situation is discussed in more detail later.

DISCUSSION

Closure

Closure refers to the process of solving the system of equations or converging on the steady state balance. Despite the wide variety of process models, simulation architectures and other features, the techniques for converging balances have not progressed beyond two or perhaps three basic techniques. The reason for this is that there are only so many ways to solve a system of nonlinear simultaneous algebraic or differential equations (ADE's).

The approach used to converge the system usually differentiates the structure and philosophy of the various process simulation programs. The most common architecture is referred to as block sequential or sequential modular (SEM) while at the opposite end of the spectrum is the simultaneous equation (SEQ) approach. Some advantages and disadvantages of each are summarized in Table 1.

The SEM approach treats each unit operation model or module as a stand-alone entity represented by its own set of equations. The module is "given" input data including streams and module or equipment parameters and determines or computes "output" or "calculated" streams and equipment parameters. In some cases these calculated streams may actually be inlet streams as in the case of inlet air to a furnace determined from a specified level of excess air and other data.

Table 1. Systems of Equations and Convergence Approaches

Convergence technique	Advantages	Disadvantages
Simultaneous Equation or Modular SEQ	Generates a Jacobian matrix Natural when momentum balances are important	Requires sparse matrix techniques Not user-friendly
Sequential Modular SEM	Easily implemented Easily debugged	Slower to converge for constrained or optimization problems
Hybrid-Sequential Simultaneous Systems	Inherit advantages of SEM methods - Jacobian cannot be reused on the next iteration	Use sparse matrix techniques "locally" - inherit disadvantages of simultaneous approaches

The general form of the equations is given by

$$x_j = g_j(x_i) \quad i = 1,n \quad (i \neq j) \tag{1}$$

This is a special type of system (explicit) in which the dependent variable x does not appear on the right hand side. Those equations which are implicit such as the following,

$$x_j = g_j(x_i) \quad (i = 1,n) \tag{2}$$

are usually converged locally within a unit model using an iterative method such as Newton-Raphson. The convergence properties of the implicit equations are usually much more sensitive to the derivatives of the function with respect to the independent variables. The convergence of the explicit functions is much less restrictive.

Hybrid systems which include local simultaneous equation systems within sequential modular systems can occur quite frequently. For example, local chemical and phase equilibrium may involve solving sets of coupled algebraic equations simultaneously and iteratively. However, the overall system of flowsheet equations may be solved sequentially and iteratively. Such systems are converged "locally" at each process node each time the node is evaluated.

Sparsity

The SEM approach takes advantage of the fact that individual operations depend only on a fixed and usually small number of process streams. This tends to make each operation coupled to the mainly to only a few flows from other modules. The result is that the system of equations is usually sparse. Thus each block of equations can be approximately solved separately without having to resort to solving the entire system. The properties of sparsity and the explicit form of most of the equations give the system a high degree of stability in the convergence process.

The convergence process involves an iterative solution. The most common approach, tear stream convergence, is similar to the Gauss-Seidel method of converging implicit systems such as Eq. 2. The Gauss-Seidel method starts with a set of initial guesses for x which are substituted into g(x) to generate an updated solution set x_u. As each new value of x is obtained it can be used to update the set of initial guesses. This approach is very similar to the "direct-substitution" used in tear stream convergence.

Mixer-Splitter System

The equations for a simple mixer-splitter system consisting of a single component is shown in Fig. 2.

The system of equations for total flow of mass, F, and enthalpy, E, are given by

$$\textit{MIXER:} \quad \begin{aligned} F_1 &= F_7 + F_6 \\ E_1 &= E_7 + E_6 \end{aligned} \tag{3}$$

$$\textit{SPLITTER:} \quad \begin{aligned} F_2 &= f_1 F_1 \\ F_3 &= (1-f_1) F_1 \\ E_2 &= f_1 E_1 \\ E_3 &= (1-f_1) E_1 \end{aligned} \tag{4}$$

$$\textit{MIXER:} \quad \begin{aligned} F_4 &= F_2 + F_5 \\ E_4 &= E_2 + E_5 \end{aligned} \tag{5}$$

$$\textit{SPLITTER:} \quad \begin{aligned} F_5 &= f_2 F_4 \\ E_5 &= f_2 E_4 \\ F_6 &= (1-f_2) F_4 \\ E_6 &= (1-f_2) E_4 \end{aligned} \tag{6}$$

For multi-component systems additional mass balance equations would be required as well as component enthalpy relationships. The total enthalpy flows, E's, are actually coupled to the total mass flows, F's, by the usual enthalpy definition,

$$E = \left(E_f + \int_{T_{ref}}^{T} C_p dT \right) F \tag{7}$$

where subscript "ref" refers to the reference temperature and f refers to enthalpy of formation. For an isothermal system, the temperature-dependent term is constant, and the enthalpy is a constant multiplied by the total mass flow. Thus the enthalpy equations can be dropped from the system. For an isothermal system (also adiabatic and no heat inputs), the sparse system for F is then given in matrix form by Eq. 8 below.

$$
\begin{array}{|ccccccc|}
F_1 & F_2 & F_3 & F_4 & F_5 & F_6 & \\
1 & 0 & 0 & 0 & 0 & -1 & \\
-f1 & 1 & 0 & 0 & 0 & 0 & \\
-(1-f1) & 0 & 1 & 0 & 0 & 0 & = \\
0 & -1 & 0 & 1 & -1 & 0 & \\
0 & 0 & 0 & -f2 & 1 & 0 & \\
0 & 0 & 0 & -(1-f2) & 0 & 1 &
\end{array}
\quad
\begin{array}{|c|}
1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0
\end{array}
\qquad (8)
$$

Note the sparsity indicated by the zero elements. For large process flowsheets sparsity is even more pronounced. Any realistic SEQ approach must store only nonzero elements of the arrays and rely on pointers to these values in solving the system. For the more realistic case of a multi-component system with energy inputs,the process would become non-isothermal, non-adiabatic. This would make the system nonlinear and coupled and the size of the system of equations would increase in proportion to the number of components. In contrast, the sparsity of the system is utilized automatically by the SEM approach. The data storage requirements for the SEM approach are significantly lower than those of the SEQ approach.

In the above flowsheet stream 7 enters the process and stream 3 leaves the process. Thus we intuitively know that $F_3 = F_7$ and $E_3 = E_7$. However, for obvious reasons this fact cannot be used for either a general SEM or SEQ system. Only the input or more precisely the "given" streams are known. For the moment we will assume that we are not constraining the outputs to the process while trying to determine the inputs.

The SEM approach is started with an assumed sequence for the calculation of each block or set of equations. In the above case, the sequence begins with block 1 and proceeds to block 4, e.g. block 1,2,3,4. Now the simplest approach is to assume zero values for the unknowns in the above equations on the first iteration and simply substitute new values as they are obtained with each iteration - a simple direct substitution approach. Accelaration approaches may also be used. However, these require storage of previous iterations and may suffer from stability problems. The Wegstein (secant method) is reasonably stable and can be adjusted through an under-relaxation parameter to minimize instability.

Note that the tear streams represent the minimum set of streams which need to be checked for convergence. Checking additional streams only increases computational and storage requirements.

Once new guesses for F_6 and E_6 are known, a second iteration usually leads to a more accurate set of values of F_6 and E_6. Given the values for stream 6, streams 1, 2 and 3 can be computed directly. When the mixer is reached, a guess for stream 5 is required in order to proceed. Once stream 5 is guessed, new values for stream

5 and 6 are determined and the next iteration can be continued. Streams 5 and 6 are referred to as "tear" streams. The concept is that the stream is "torn" by storing new and old values of the flows for each stream. New values are substituted for old values on each iteration through the loop as shown in Fig. 3.

Tear Stream Convergence

Convergence is measured by the approach to constant values for the flows in the tear streams (5 and 6 in this case). The relative change of each stream component across the tear, i.e. the relative difference between new and old values of F and E, is compared. When the absolute relative difference is less than a specified relative tolerance, ϵ, the loop is said to be converged as noted in Eq. 9. The two loops in the process are represented by tear streams 5 and 6.

$$\textit{Relative Tolerance:} \quad \left| 1 - \frac{F_{new}}{F_{old}} \right| \leq \epsilon \qquad (9)$$

The tolerance check must be applied to all component flows or composition as well as independent thermodynamic variables such as temperature and pressure.

In other cases, it may be necessary to constrain a flow below a specified level by manipulating a second flow. In this case an absolute tolerance, A, is used,

$$\textit{Absolute Tolerance:} \quad A \geq |F_{new}| \qquad (10)$$

This type of tolerance checking is not really part of the solution technique. However, it is a very convenient extension of the method of tear stream testing.

Relation to Flowsheet Topology

Even though each module is designed to achieve local mass and energy balance, the process is not balanced globally until the tear streams are converged. Generally, the user of the program must specify the stream to test for convergence. However, this may not always be obvious. For example, we can easily show that the tear streams are not fixed for this flowsheet, they depend on the specified sequence or calculation order. For the following sequence,

$$Mod_2 \quad Mod_3 \quad Mod_4 \quad Mod_1 \qquad (11)$$

Stream 1 must be specified initially in order to calculate streams 2 and 3 from Mod_2. Again stream 5 is the second tear stream required to continue through Mod_4. Finally Mod_1 generates a new value of stream 1 given stream 6 and the inlet stream. This shows that the tear stream set is a function of the sequence and the topology of the flowsheet. For the sequence shown in Eq. 11, the convergence controller blocks 5 and 6 are added with loops back to blocks 3 and 1 showing that Mod_5 passes the calculation pointer back to Mod_3 and Mod_6 passes the calculation pointer back to Mod_1.

The topology or connectivity of the flow sheet has a fundamental effect on which streams are tear streams and the speed and likelihood of convergence. For example for sequence I, stream 7 can be classified as an inlet stream, stream 3 as an outlet stream, streams 5 and 6 as tear streams and streams 1, 2 and 3 as connect streams. For sequence 2, streams 1 and 6 are tear streams and the streams 2, 3, 4, and 6 are connect streams. As noted above, it is a waste of time to check connect streams for convergence.

The analysis of the topology is perhaps the most reliable way to choose tear streams and thus to converge the flowsheet. Complex flowsheets containing upwards of a thousand process modules could contain dozens of tear streams.

The sequence must be considered to avoid interactions or overlap between loops. Otherwise, loop control could pass back and forth between loops thus slowing convergence. For large flowsheets the sequence can affect the speed and behavior of the flowsheet convergence with the tear stream method.

Non-conservative Tear Stream Testing

While tear stream convergence is usually associated with mass and energy flows, it is sufficiently general that it can be applied to stream structures or variables which do not necessarily obey conservation laws. For example, in simulators which pass stream structures containing substream, component or performance attributes, these streams can also be checked by tear stream testing. Such a method determines whether the equations are consistent and have a solution.

The greatest problem with tear stream methods is the need to specify all tear streams to assure a global balance. Associated with this is the inconvenience of identifying loops, adding loop control modules and specifying loop control and convergence data. The relative error and iteration parameters often require trial and error or experience to determine.

Simultaneous Convergence

Simultaneous approaches do not require recognition of tear streams since stream 3 could be specified and stream i calculated as easily

Nodal-Balance Convergence

A new approach is now used for a wide variety of demanding simulation problems. The method is based on performing a nodal balance around each module. By minimizing the individual nodal errors, the global system is automatically converged. Rather than checking tear streams this method computes over the flowsheet nodes. There is a strong connection, however, between the nodal balance and the tear stream error. Provided there are no systematic errors in the mass and energy balance within a given module, any error which may occur must be due to an error in an entering stream which by definition must be a tear stream. The relative magnitude of the error will be a measure of the relative error of the tear stream.

Since the total mass and total energy flows are known for each stream entering and leaving each module, the convergence algorithm need only check one or both of these values for relative convergence on each iteration. The module with the largest relative error is indicative of the loop with the largest tear stream error. Thus, the algorithm can proceed as follows.

For each node in the flowsheet proceeding through the calculation order, the **total** mass or energy balance error, ϵ_{mod} is determined,

$$\left| \frac{\sum_{I=1}^{n} F_I^i}{\sum_{I=1}^{m} F_I^o} - 1 \right| = \epsilon_{mod} \qquad (12)$$

where the summation is over the streams entering and leaving each module.

Convergence can be tested on either mass or energy (or both). Usually, energy converges at the same rate as mass and only one test is required. In some systems, such as steam and power, energy may be more critical than mass balance. In some systems, mass balance may the only thing of interest and it alone should be checked.

The relative errors are sorted in descending order of magnitude.

$$\max | \epsilon_1 \; \epsilon_2 \; . \; . \; . \; \epsilon_n | \qquad (13)$$

The calculation is restarted at the node with the largest relative error. There is no need to check for intermediate loops or tear

streams. A single convergence block is placed outside the outermost loop or at the end of the calculation order. For the mixer-splitter example, the convergence controller, Mod_5 is placed last and control passes back to either Mod_1 or Mod_3 depending on the relative error at those nodes. When control is passed to Mod_1 (the outer loop), the inner loop is automatically recomputed as would be the case with tear stream convergence. However, the inner loop is not reconverged in the normal sense unless the relative error again becomes larger than the specified tolerance.For the simple example flowsheet, the sequence becomes that shown in Fig. 5.

Using this type of convergence does not preclude using tear stream convergence for selected loops. However, since the nodal-balance convergence controller may pass control back to any loop, care must again be used to avoid defining a loop improperly.

In some cases, it may be necessary to combine tear stream and nodal convergence control. For example, if it is necessary to drive an excess stream flow to zero, a combination of a controller and a tear stream convergence block based on an absolute tolerance criteria can be used to manipulate an input so as to reduce the flow of an output to any desired low value. However, the microbalance approach cannot be used for such a situation because it never actually tests a stream. However, the combination enables the constraint to be satisfied and the system to be globally balanced.

In the nodal balance method as the system becomes more converged, the relative errors decrease and the algorithm selects different modules with successively smaller errors until the system is globally converged. Thus, the relative tolerance assures that the entire system achieves at least this level of convergence. Also the relative tolerance which is specified is consistent with the total mass or energy balance errors as usually defined. For tear stream convergence, it is not always clear how the relative tolerance on two successive iterations of a tear stream relate to the overall mass balance closure.

For moderately sized flowsheets, the nodal-balance approach is very efficient. First of all only one simple calculation is performed for each module. While for a 50 component stream, a similar calculation is required for each component in each tear stream or as a minimum the outermost tear stream. Thus for situations in which the streams contain many components and many tear streams, the microbalance convergence approach requires fewer functional evaluations and less CPU time. This approach is also simpler to define since it requires less information to specify the convergence block.

Global Balances

The microbalance technique assures a true global balance even on

trace components to a tolerance level limited only by machine accuracy. This is possible despite the fact that individual component flows are not checked as with tear stream testing. The reason is that component flows or composition are highly correlated in block simulation routines and each node or module is (or should be) designed to conserve components at the atomic or molecular species level in a consistent manner even when nonconventional components such as lignin, hemicellulose or fibers are present.

This closure issue was tested as part of an analysis of a complex tissue machine flowsheet model. The model included blocks which performed local phase and chemical equilibrium calculations using simultaneous equation solving techniques (Newton-Raphson and matrix solution). These blocks were converged before passing on to the next block in the flowsheet. These blocks offered the opportunity to "lose" material through numerical roundoff. However, they consistently showed 5 or 6 place accuracy in the atomic and molecular balance therefore satisfying the criteria of local or nodal convergence.

Although each streams contained up to 166 components, the analysis focused mainly on the flows of fiber fines, trace ions in the liquid phase such as Ca^{++}, Na^{+}, Mg^{++}, Fe^{+++}, Cl^{-} and SO_{4--} and the same ions bound to the fibers. An overall balance was determined by summing the inlet and outlet flows of each trace component and computing a relative difference. Each of the species was balanced globally to within 3 % compared to a maximum nodal convergence error of .03 %. This indicates that the relative error of a given component increases in inverse proportion to its concentration. The typical trace component flows are on the order of 50 lb/hr compared to $1x10^{6}$ lb/hr for the total stream flowrate. Thus the typical concentration is 50 ppm and the global trace component error is 1.8 ppm. This is certainly satisfactory for this type of analysis.

Measures of Performance

The performance of the two approaches (tear stream and nodal-balance) can be compared in several ways: ease of use, generality, speed and robustness. Some comparisons are summarized in Table 2. For this comparison example the components consisted of water and 10 fiber components. The simplified flowsheet above was used. The separation efficiencies, f_1, specified in Module 2 were fixed at 0.5 for all components. The separation efficiencies for Module 4 were the same for each component and set at 0.99 to the rejects (recycle) for case 1 and 0.999 to the recycle for case 2. These cases were set up to create situations of very high recycle which would lead to very high recycle flows and generally slow convergence by direct substitution methods.

Table 2. Comparison of Convergence Methods

	Tear Stream Testing	Nodal-balance
Number of Convergence blocks	2	1
Number of Parameters	8	3
Split Fraction 0.99	Rel tear stream tol. $1x10^{-5}$	absolute mass bal error = $1x10^{-5}$
time seconds	102.9	29.64
converged mass and energy balance error	$< 1x10^{-5}$	$0.95x10^{-5}$
Iterations	**	1247
Split Fraction 0.999		
time seconds	458	274

Table 2 shows that the microbalance convergence technique requires fewer parameters and blocks and executes 300 % faster in Case 1 and 50% faster in Case 2. The tear stream method produces a tighter micro-balance (smaller maximum balance error) because the criteria for convergence is based on the change in the tear streams. The convergence of the microbalance is exactly as specified. It is possible that the execution time for the two methods would be similar if they could be specified in such a way that they achieved equal levels of overall balance.

A comparison of the tear streams at convergence showed that the two methods gave similar results. However, the tear stream method would be expected to be somewhat more accurate in this case because of the tighter overall tolerance achieved.

Comparisons for More Realistic Cases

The algorithms were compared for several more realistic and complex flowsheets. For example, a thermo-mechanical pulping system consisting of 31 process blocks made up of detailed refiners, screens, cleaners, mixers and consistency controllers had six tear streams. Using a single nodal balance controller and zeroing out

all tear streams the system was converged to a maximum balance error of 0.2×10^{-6} in 50 seconds on a VAX 3100 work station. The same system using six tear stream convergence blocks set to a tear stream tolerance of 1.0×10^{-5} required 150 seconds to converge. The resulting maximum local balance error was slightly higher than obtained with nodal convergence. The loop control with six tear stream convergence blocks also required restarting the system several times to complete the simulation. This is probably a function of loop control and is another drawback with tear stream convergence.

Other more complex flowsheets simulating two-ply linerboard papermachines, tissue papermachines and corrugating medium papermachines all showed a similar performance difference.

The advantages and disadvantages of the two methods are summarized in Table 3.

Table 3. Advantages and Disadvantages of Convergence Methods

Feature	Tear Stream	Nodal-balance
Ease of Use		X
Global Balance		X
Generality - not confined to mass and energy - absolute convergence	X	
Speed		X
Robustness		X
Detects Flowsheet Errors		X
Convergence Acceleration	X	

CONCLUSIONS

This new technique is simple to implement and works well for many complex flowsheets. It also works with tear stream convergence and with flowsheet constraints. Thus there are few situations in which this type of convergence could not be used effectively for sequential modular or hybrid flowsheeting programs.

REFERENCES

Cavett, R. H., "Applications of Numerical Methods to the Convergence of Simulated Processes Involving Recycle Loops," A.P.I Div. Ref. 43, 57 (1963).

Christensen, J. H., Rudd, D. F., "Structuring Design Computations," AIChEJ, 15(1):94-100 (1969).

Evans, L. B., Boston, J. F., Britt, H. I., Gallier, P. W., Gupta, Joseph, B., Mahalec, V., Ng, E., Seider, W. D., Yagi, H., "Aspen: An Advanced System for Process Engineering," Comp. & Chem. Engr., 3:319-327 (1979).

Forder, G. J., Hutchison, H. P., "The Analysis of Chemical Plant Flowsheets," Chem. Engr. Sci., 24: 771-785 (1969).

Lee, W., Rudd, D. F., "Ordering of Recycle Calculations," AIChEJ, 12, 1184 (1966).

MAPPS, "Modular Analysis of Pulp and Paper Systems", Module Technical Documentation, The Institute of Paper Science and Technology, (1992).

Norman, R. L., "A Matrix Method for Location of Cycles of a Directed Group," ibid., 11, 450 (1965).

Upadhye, R. S., Grens II, E. A., "Selection of Decompositions for Chemical Process Simulation," AIChEJ, 21(1):136-143 (1975).

Wegstein, J. H., Communication of the ACM 1(6):9-13 (1956.

Westerberg, A. W. et al. Process Flowsheeting. Cambridge University Press, p. 63-73 (1979).

NOMENCLATURE

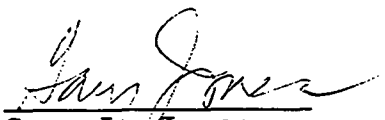
C_p	Heat capacity
E	Enthalpy flowrate
f	Split fraction
F	Mass flowrate
T	Temperature
x	Stream variable
ϵ	error

subscripts

i	component or stream index
f	formation
ref	reference temperature

superscripts

i	inlet
o	outlet



Gary L. Jones

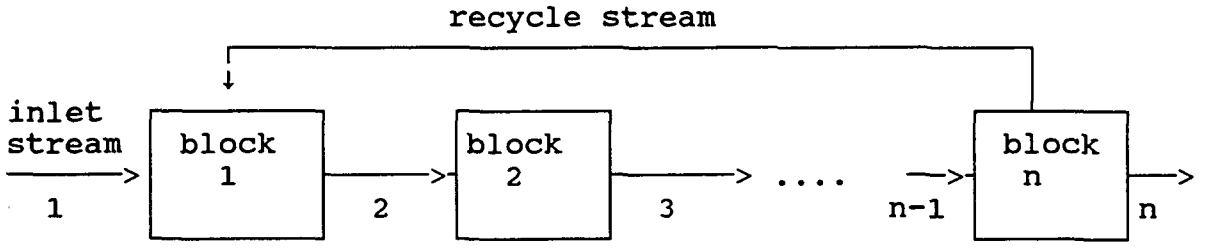


Fig. 1. Conceptual Block Diagram With Recycle

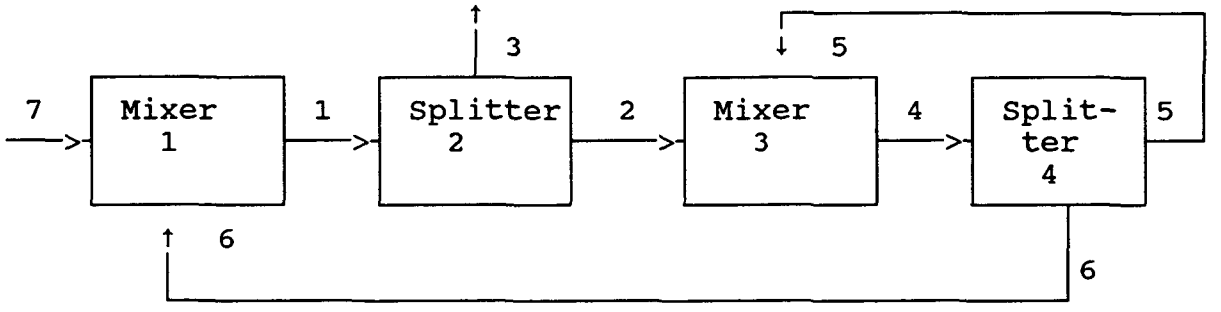


Fig. 2 Simple Mixer-Splitter System

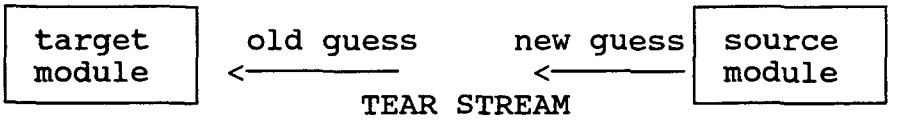


Fig. 3 Concept of a Tear Stream

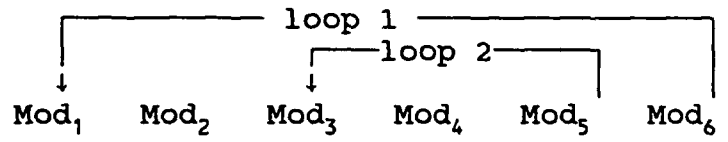


Fig.4 Calculation Sequence with Controller Loops Added Tear Stream Testing

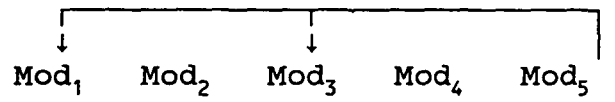


Fig. 5 Calculation Sequence with Convergence Controller Loops Added - Nodal-balance Technique