

核燃料サイクルシステムにおける 再処理プロセスの最適化検討－1

(動力炉・核燃料開発事業団 委託研究成果報告書)

技 術 資 料		
開示区分	レポ ー ト No.	受 領 日
T	J1609 97-002	1998.4.20
この資料は技術管理室保存資料です 閲覧には技術資料閲覧票が必要です 動力炉・核燃料開発事業団 技術協力部技術管理室		

1997年3月

埼玉大学工学部応用化学科

複製又はこの資料の入手については、下記にお問い合わせ下さい。

〒107-0052 東京都港区赤坂1-9-13

動力炉・核燃料開発事業団

技術協力部 技術管理室

Inquiries about copyright and reproduction should be addressed to: Technical Evaluation and Patent Office, Power Reactor and Nuclear Fuel Development Corporation 9-13, 1-chome, Akasaka, Minato-ku, Tokyo 107, Japan

動力炉・核燃料開発事業団 (Power Reactor and Nuclear Fuel Development Corporation) 1997

核燃料サイクルシステムにおける再処理プロセスの最適化検討ー 1

*

松本 史朗

要 旨

長期間にわたる核燃料サイクル技術開発の過程では、適用できる基礎科学技術や社会環境も変化し、核燃料サイクルの満たすべき要件も一定ではない。従来は核燃料サイクル技術の成立性や安全性の実証に重点が置かれていたが、最近では再処理の経済性や廃棄物管理に関心が向けられてきている。

再処理の経済性には廃棄物が極めて大きな影響を及ぼすため、単なる環境への負荷の軽減のためでなく、経済性の向上の目的においても廃棄物発生量をどう減らすか、また廃棄物処分をどのように合理的に行うことに貢献できるかが重要な課題となっている。しかし、再処理廃棄物の発生メカニズムと必然性、発生後の処理方法の妥当性、処分体としての妥当性からみた発生元のプロセスの妥当性等、核燃料サイクルとして全体的総合的な評価はまだ行われていない。

現在の再処理プラントでは固体/液体/気体廃棄物が多量に発生し、これを処理するために多大の労力を要している。また、処分形態も未定な廃棄物が多い。この現実の裏には運転していく際に不可避免的に発生する雑廃棄物が極めて多いという事実がある。しかも発生廃棄物の全体像を把握した上での再処理プロセスも含めた総合評価はいまだになされていない。

よって再処理プラントの運転上発生する非プロセス系廃棄物をも考慮して、再処理プラントから発生する廃棄物量の定量的評価をいかにするかが重要となる。本研究ではプロセスフローシート作成-解析システムをもとに開発された核燃料再処理プロセスフローシート作成-解析システムを廃棄物処理系へ拡張することを試みた。本報告はその基本となるプロセスフローシート作成-解析システムと再処理の主工程について開発された核燃料再処理プロセスフローシート作成-解析システムの概要をまとめるとともに廃棄物処理系へのその拡張を行っているものである。

本報告書は、埼玉大学工学部応用化学科が、動力炉・核燃料事業団の委託により実施した研究の成果である。

事業団担当課：再処理技術開発部 アクチニドプロセス・分析開発室

* 埼玉大学工学部応用化学科

目 次

1. プロセスフローシート作成－解析システム	1
1.1 概 要	1
1.2 プロセスフローシートのグラフィカルな表現	1
1.3 プロセスフローシート計算方法	2
1.4 プログラミング	5
1.5 適 用 例	8
1.6 ま と め	8
1.7 引用文献	9
2. 核燃料再処理プロセスフローシート作成および解析システム、 COMPAS の開発	12
2.1 概 要	12
2.2 はじめに	12
2.3 システムの構成および機能	13
2.4 COMPAS で利用できるコード	15
2.5 適 用 例	17
2.6 ま と め	20
2.7 引用文献	20
3. 廃液処理およびオフガス処理工程のフローシート計算	33

1. プロセスフローシート作成—解析システム

1.1 概要

化学プロセスフローシートは装置とそれらを結ぶフローから構成されている。プロセスの設計あるいは解析はこの化学プロセスフローシートを対象に行われる。プロセスフローシティングに必要な手法およびその手続であるリサイクルの抽出、システムのデコンポジションおよびティアリングの概念は70年代までに提案されている^{1,2)}。また、現状ではプロセスの各種単位操作の計算コードについては整備されている。

プロセスフロー計算を幾何学的なプロセスフローシート上で、直接開発者が行えるならば、プロセスの合成、解析が容易になるであろう。そこで、ストリームおよびユニットを幾何学的に表現したプロセスフローシート上でのプロセスフローシート計算を効率よく行う方法を検討した。具体的には、ストリームおよびユニットの表現とフローシート計算とから構成されたシステムを問題にした。開発したシステムの有効性を検討するために、核燃料再処理プロセスの代表的な工程のフローシート解析を行った。

1.2 プロセスフローシートのグラフィカルな表現

1.2.1 ストリームおよびユニットのフレーム表現

ストリームおよびユニットはそれぞれストリーム変数およびユニット変数を有する。ストリーム変数は対象とするプロセスフローシートでは同一であり、上流側から下流側へ継承されなければならない。また、ユニット変数はそのユニット自身に固有な属性を保持しなければならない。それゆえ、プロセスフローシート計算では個々のストリームおよびユニットは、ストリーム変数およびユニット変数をスロットとするフレームのインスタンスとして表現するのがよい。さらに各ユニットのフレームにはそれぞれにデフォルトの属性をスロット値として保持させ、各インスタンスはそれらを継承することができるようにした。また、ユニットの機能ごとにフレームを作成し、ユニットを階層的に整理することができた。

1.2.2 ストリームおよびユニットのアイコン化

プロセスフローシートをテキスト入力ではなく、グラフィカルに入力するためには、

ストリームのウインドウ上での接続および削除、ならびにユニットの張り付け、移動および削除をマウスで行えることが不可欠である。そこでストリームとユニットをアイコン化し、ストリームやユニットのフレーム情報をそれぞれのアイコンに集約した。

次に、ストリームの接続形式が重要である。ストリームの接続形式とはストリームラインが幾何学的にユニット間でどのような直線で結ばれるかという情報であり、プロセスフローを計算機の画面上に描く際に、ユーザーが見やすいストリームの形を選択することが必要である。ここでは4型式のストリームの形を水平線と垂直線との組み合わせで表現した。具体的には、Fig. 1.1 に示す4種類である。

またストリームの相をあらかじめユーザーが決定し、そのストリームのアイコンを相の種類ごとに分けることで、ストリーム中の物質がユーザーに容易に把握できることが必要である。ストリームの相とはストリーム中を流れている物質の物理化学的な相のことである。本システムでは水相、油相および気相の3種類を用意した。

1.3 プロセスフローシート計算方法

1.3.1 ストリーム-ストリーム行列の生成

Fig. 1.2 に示すプロセスフローの例を用いてプロセスフローシート計算におけるリサイクルループの抽出、デコンポジション、計算順序の決定およびティアリングについて説明を行う。プロセスフローシートは有向グラフとみなせるためリストあるいは行列として表現できる。ここでは、西村らのプロセスフローシートのP行列表現³⁾を採用し、ストリーム-ストリーム行列で表現した。Fig. 1.2 に示すプロセスフローのストリーム-ストリーム行列を以下に示す。行列中のSはストリームを表し、添え字はストリームの番号を表す。ij要素が1であることは入力ストリームiと出力ストリームjとを有するユニットが存在することを表している。また、0はその存在がないことである。

	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7
S_0	0	1	0	0	0	0	0	0
S_1	0	0	1	0	0	0	0	0
S_2	0	0	0	1	1	0	0	0
S_3	0	0	1	0	0	0	0	0
S_4	0	0	0	0	0	1	0	1
S_5	0	0	0	0	0	0	1	0
S_6	0	1	0	0	0	0	0	0
S_7	0	0	0	0	0	0	0	0

1.3.2 リサイクル・ループの抽出

ストリーム-ストリーム行列に基づき以下の手順でリサイクル・ループの抽出を行った。

- (1) システム入力からのストリームはストリーム-ストリーム行列において行の要素が全て0である列の番号に相当するストリームである。全てのシステム入力のストリームを求める。
- (2) システム入力のストリームの1つに着目する。
- (3) システム入力のストリームのリストに、現在、着目しているストリームを付け加える。
- (4) 着目しているストリームからストリーム-ストリーム行列を参照して推移ストリーム列を求め、推移ストリーム列内と同じ要素がリスト中にあるとき、リスト内のその要素から後方のストリーム列をループとみなし、リサイクル・ループの集合に加え、終了する。
- (5) 推移ストリーム列内のそれぞれの要素について(3)に戻る。
- (6) (2)に戻る。

全てのシステム入力からの探索が終了したら、2重に抽出したループを探し、一方を削除する。上記アルゴリズム中の(3)から(5)は一つのサブルーチンとし、再帰的に(5)から(3)へ戻る構造にした。Fig. 1.3 にリサイクル・ループの抽出のアルゴリズムの適用時における処理中のリストの内容を示す。ここでも Fig. 1.2 に示したプロセスフローを例として用いた。図中のアンダーラインはリサイクル・ループを構成するストリームを意味する。

1.3.3 プロセスシステムのデコンポジション

ストリーム-ストリーム行列とリサイクル・ループ集合からサブシステム集合を生成する。抽出したリサイクル・ループ集合の中でループ同士が触れているものは一つのサブシステムとみなし、それらの和をとりサブシステム集合の要素とした。ループを構成していないストリームはそれ自体を一つのサブシステム集合の要素とした。生成したサブシステム集合の要素の順序に従って、ストリームの番号を付けなおした。ここで新たにサブシステム間の関係を表すサブシステム-サブシステム行列を生成した。この行列ではサブシステム i からサブシステム j につながりがあるときに ij 要素は1とし、対角要

素は0とする。この行列を用いて以下の手順によってサブシステム集合を入力から出力へと順番に並べなおした。

- (1) j を 0 にする。
- (2) 行列の j 列に着目する。
- (3) 行列の j 列の要素の和 W を求める。
- (4) もし $W \neq 0$ であれば j をインクリメントして(2)に戻る。
- (5) $W = 0$ であれば、このサブシステム j は入力ストリームである。そこで、行列から j 行目と j 列目を削除し、 j をスタックにプッシュする。もし、行列が 1×1 の行列になったら終了する。
- (6) j を 0 にして、(2)に戻る。

この後、スタックを参照しながらストリームの番号を再び付け替える。このストリームの番号に従って計算を行う。

1.3.4 ループ・ストリーム行列の生成

サブシステム内のリサイクル・ループとストリームとの関係はループ・ストリーム行列で表現する。3.1 においてプロセスフローのリサイクル・ループ集合を求め、3.2 においてサブシステム集合を求めたので、これらの情報からサブシステムごとのループ・ストリーム行列を以下の手順で求める。

- (1) ひとつのサブシステムに注目する。
- (2) このサブシステムに属するリサイクル・ループ i をリサイクル・ループ集合から求める。
- (3) 求めたリサイクル・ループ i と、サブシステムの要素であるストリーム S_j から、 S_j がリサイクル・ループ i に属するとき、 ij 要素を 1 とするようなループ・ストリーム行列を生成する。
- (4) 全てのサブシステムについてループ・ストリーム行列を求めるまで(1)に戻る。

Fig. 1.2 に示したプロセスフローの例からこの操作により生成されたループ・ストリーム行列を以下に示す。行列中の X はリサイクル・ループ 2-3 を表し、 Y はリサイクル・ループ 1-2-4-5-6 を表す。

	S_1	S_2	S_3	S_4	S_5	S_6
X	0	1	1	0	0	0
Y	1	1	0	1	1	1

1.3.5 ティアリング

デコンポジションされたサブシステム内のストリームを計算する際にはループ中のストリームをティアリングしてループのないシステムにし、ティアリングしたストリームのストリーム変数を仮定し、システム内のストリームを順次計算し、収束するまで繰り返し計算を行う。このとき、ティアリングするストリームの数は少ない方が繰り返しの数は少なくなる。そこで、ループストリーム行列に Lee と Rudd のアルゴリズムを適用して、最適なティアリングを行った。²⁾

1.3.6 ユニットのモジュール化

プロセスフローシート計算には、既存の計算コードを用いることが多いが、計算コードのプログラムを書き換えてシステムに組み入れる作業は多大の労力を必要とする。また、計算コードの著作権の問題も考慮して、本システムからユニットの計算コードを本システムのモジュールとして呼び出し、入力ストリームの情報とユニットの属性値をモジュールに送り、計算結果を本システムのストリームに返す形式を用いた。計算コードは、そのプログラム名を個々のアイコン化されたユニットのフレームのロット値として保持させた。これによって、計算コードはシステムと統合されるが、計算コードのプログラムを書き換えたり、本システムとリンクする必要はなくなった。

1.4 プログラミング

1.4.1 ストリームおよびユニットのフレーム表現

本システムはC言語で記述した。ストリームとユニットのフレーム表現はC言語の構造体で実現した。これらの構造体は実行中にストリームやユニットを生成して画面に張り付ける際に動的にメモリに割り当て、ストリームやユニットが破棄される際に割り当てたメモリを解放している。これにより本システムは必要最小限のメモリで動作している。

```

typedef struct Stream {
    short x [20];           /* ストリームラインのX座標列 */
    short y [20];           /* ストリームラインのY座標列 */
    short approach_type;   /* 接続形式を保持する変数 */
    short line_type;       /* ストリームの相を保持する変数 */
    short connectfrom;     /* 接続されているユニットの番号 */
    short from_dx;         /* ストリームの始点X座標 */
    short from_dy;         /* ストリームの始点Y座標 */
    short from_id;         /* 接続されているユニットの種類 */
    short from_unit;       /* ユニットの接続番号 */
    short connectto;       /* 接続されているユニットの番号 */
    short to_dx;           /* ストリームの終点X座標 */
    short to_dy;           /* ストリームの終点Y座標 */
    short to_id;           /* 接続されているユニットの種類 */
    short to_unit;         /* ユニットの接続番号 */
    short meter_x;         /* ストリームアイコンのX座標 */
    short meter_y;         /* ストリームアイコンのY座標 */
    Slot *slot;            /* スロットへのポインタ */
} Stream;

```

```

typedef struct Unit {
    short id;               /* ユニットの種類 */
    short frame;           /* 親フレームの番号 */
    short unit_num;        /* ユニット番号 */
    char unit_name[20];    /* ユニット名 */
    int x;                  /* ユニットのX座標 */
    int y;                  /* ユニットのY座標 */
    Server_image bitmap;   /* ユニットのビットマップイメージ */
    short bit_size_x;      /* ビットマップイメージのサイズ */
    short bit_size_y;      /* ビットマップイメージのサイズ */
    Slot *slot;            /* スロットへのポインタ */
} Unit;

```

1.4.2 フレーム・スロットの表現

フレームのスロットもまた構造体で記述した。これを以下に示す。

```
typedef struct Slot {
    struct Slot *next;      /* 次のスロット構造体へのポインタ */
    char *name;            /* スロット名 */
    char *value;           /* スロット値 */
} Slot;
```

Slot 構造体はそれ自身に次の Slot 構造体へのポインタを保持し、そのポインタが (Slot*) NULL になるまでの鎖状連結としてスロットの列を表現している。そのため各フレームは先頭の Slot 構造体へのポインタを保持するだけでよい。この構造体も実行中に動的にメモリに割り当てており、実行中にスロットの数やスロットの値が簡単に換えられる構造とした。また、割り当てたメモリを解放することにより容易にスロットの破棄ができる構造である。

1.4.3 ユニットの計算コードとの通信

ユニットの計算コードは実行中に本システムの子プロセスとして起動される。本システムと起動された計算コードとの間でのデータの引渡しにはパイプを用いた。すなわち、fork, exec システムコールを用いて本システムの子プロセスとして計算コードが起動し、pipe システムコールを用いて親プロセスである本システムの出力と計算コードの入力をつなぎ、本システムの入力と計算コードの出力をつなぐ。その後、システムからストリームおよびユニットのデータを送り出し、計算コードがそれらを受け取って出力ストリームの値を計算し、親プロセスであるシステムに計算結果を返す。これにより、計算コードのシステムからの分離を実現した。

本システムは UNIX ワークステーション上に実装し、ウィンドウシステムには Open Windows を採用した。また、グラフィカルユーザーインターフェースの構築には X View を用いた。それにより、UNIX ネットワーク上で Open Windows, X11 が起動されているマシンからクライアントとして容易に起動することができる。

1.5 適用例

本システムを溶媒抽出による分離工程のフローシートの解析に適用した。Fig. 1.4 に示すプロセスは溶解工程からの溶液中のウラン、プルトニウムを核分裂生成物質と分離する核燃料の再処理プロセスの共除染工程である。図はフローシートを本システムに入力した画面である。ここでは、プロセス中の4本のパルスカラムの計算コードにはMIXSETを用いた⁴⁾。MIXSETの入力データのフォーマットは本システムのそれと異なるが、データのフォーマットを変換するプログラムを通すことでMIXSETを利用した。

プロセスの収支計算をさせるために各システム入力流量、各成分の濃度等をスロット値として設定する。ウインドウ上のシステム入力のアイコンをマウスでクリックするとスロット値を入力するウインドウを表示させることができる。また、プロセス中の各ユニットの属性値も同様にして設定する。このとき使用する計算コードも各ユニットについて設定できる。この様子を Fig. 1.5 に示す。設定をしなければ各ユニットのデフォルトで実行する。収支計算を開始させると、まずストリーム-ストリーム行列を生成して、ストリームの番号を付け替えて計算順序になるように並べる。生成したストリーム-ストリーム行列を Fig. 1.6 に示す。次にループの摘出を行う。このプロセスでは(5 6 7)と(5 8 9 10 11)である。また(5 6 7 8 9 10 11)はこれらのループを含んだサブシステムとなる。その後ティアリングを行う。この例ではストリームの5をティアリングするとループのないサブシステムになる。このサブシステム中のストリームの計算順序を決める。この例では(6 7 8 9 10 11 5)となる。この順序で収束するまで繰り返し計算を行い、次のサブシステムに移る。こうして全てのサブシステムを計算する。計算結果はシステム出力や各ストリームのアイコンをマウスでクリックするとそのフレームのスロット値として見ることができる。

1.6 まとめ

プロセスの合成、解析が容易に行えるようにプロセスフローシートのストリームおよびユニットをアイコン化し、テキスト入力ではなく、グラフィカルに入力するシステムを提案した。ストリームおよびユニットの情報を階層的に整理するために、そのデータ構造はフレームとし、プロセスフローシート計算を効率よく行うために、ストリーム間の関係はストリーム-ストリーム行列として表現した。その行列に基づいてリサイクルループを摘出し、プロセスシステムのデコンポジションを行い、計算順序を決定した。各リサイクルループの計算においてはティアリング変数の数を最少にするようなティア

リングを行った。

本システムを用いて、溶媒抽出による分離工程のフローシートの解析を行った。その結果、グラフィカルにフローシートが変更され、同時にフローシート計算が行え、本システムの有用性が示された。

1.7 引用文献

- 1) Forder, G. J. and Hutchison, H. P. : *Chem. Eng. Sci.*, 24, 771(1969)
- 2) Lee, W. and Rudd, F. : *A. I. Ch. E. Journal*, 12, 1184(1966)
- 3) Nishimura, H., Y. Hiraizumi, S. Yagi : *Kagaku Kogaku*, 31, 183(1967)
- 4) Gonda, K., K. Oka and S. Fukuda : *PNCT* 841-79-26(1979)

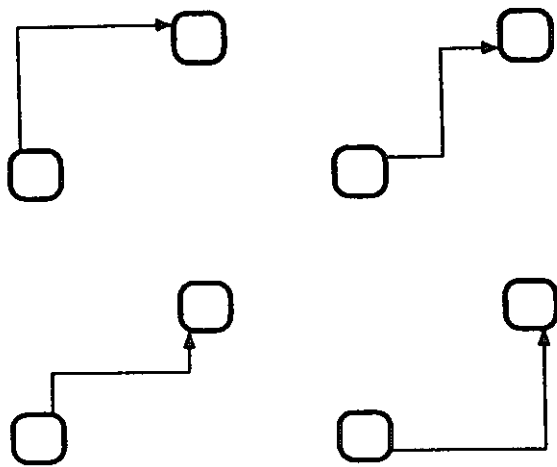


Fig. 1.1 4 kinds of connection of stream between units

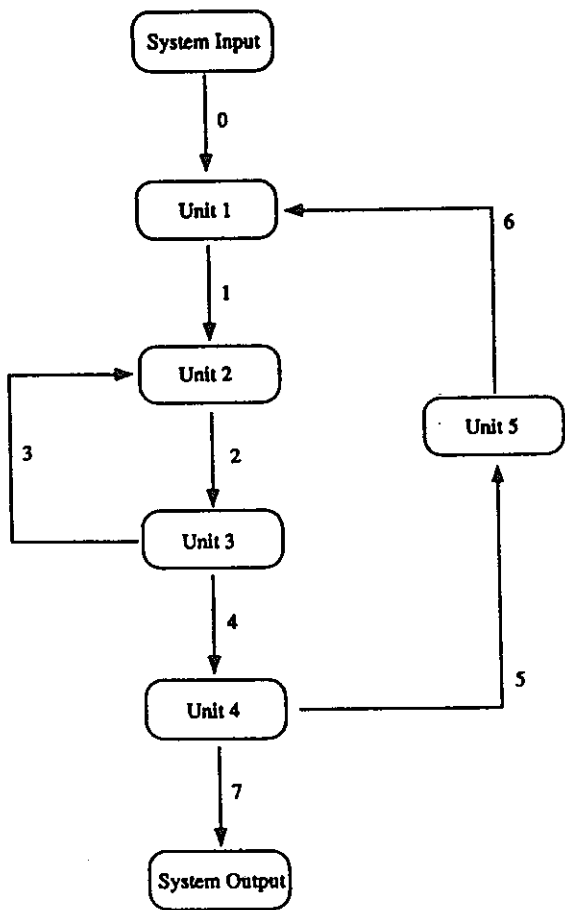


Fig. 1.2 Example of process flow

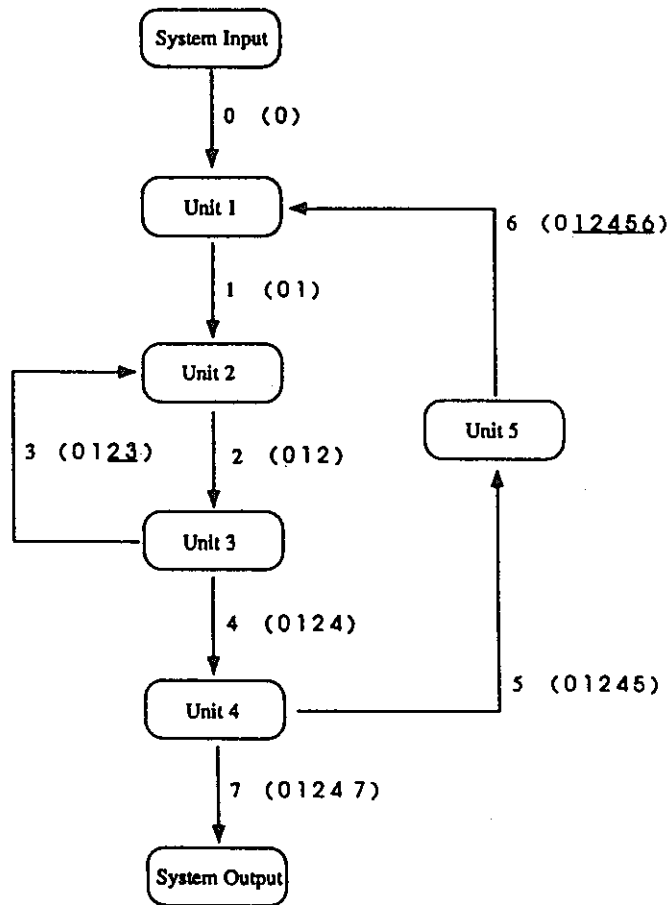


Fig. 1.3 Searching for recycle-loop
(Underline stands for stream of recycle-loop.)

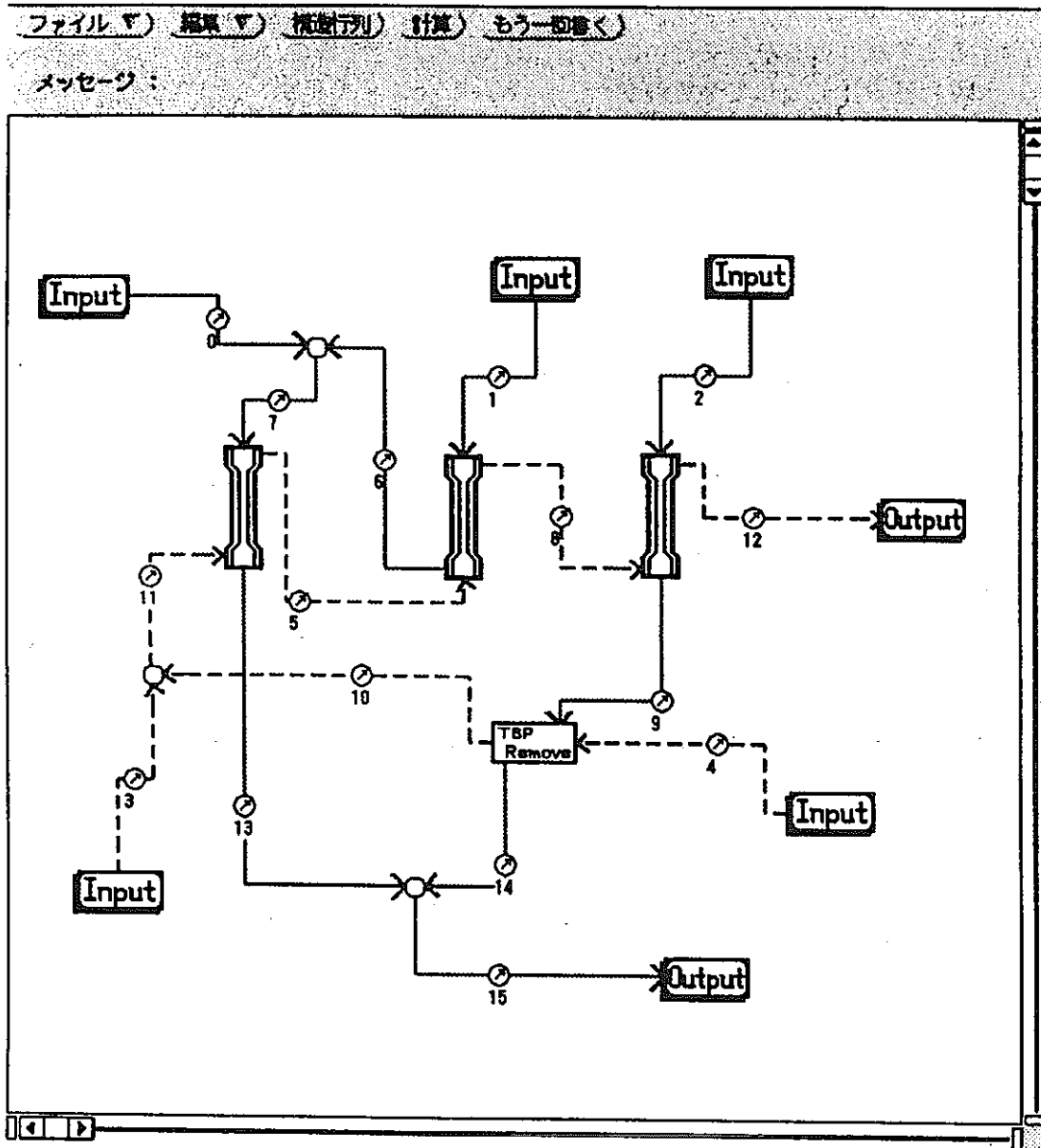


Fig. 1.4 Flowsheet of co-decontamination in reprocessing process

Pulse_Column

適用) 削除) デフォルト) 実行) キャンセル)

スロット CODE

スロット値 mixset

UNIT_IN	2
UNIT_OUT	2
CODE	mixset
CODE_ERR	0.05
BITMAP	data_0.inage
N_Stage	5
Y_Settler	30.0
Y_Mixer	10.0

Fig. 1.5 Example of frame of pulse column

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0									1							
1								1	1							
2										1			1			
3												1				
4											1				1	
5										1	1					
6											1					
7										1					1	
8											1		1			
9												1			1	
10													1			
11															1	
12																1
13																1
14																1
15																1

Fig. 1.6 Stream-stream matrix of flowsheet as shown in Fig. 1.4 (The blank in matrix is zero.)

2. 核燃料再処理プロセスフローシート作成および解析システム、COMPASの開発

2.1 概 要

プロセスフローダイアグラム上で再処理プロセスフローシートの作成および解析が行えるシステム、COMPASを開発した。プロセスフローシート内の溶解槽、ミキサセトラ抽出器などの装置はUNIXワークステーションのディスプレイ上でアイコンとして表現される。フローシートダイアグラムは、マウスの操作によって簡単に作成することができる。装置の計算コードは既存のコードだけでなくユーザーが作成したコードも使用可能である。装置の仕様およびストリーム内の成分濃度は表の形で表示され、ユーザーはその値を書き込むことができる。計算の結果もまたグラフィカルに表示される。二つの使用例によって、COMPASがPurexプロセスの操作条件の決定およびミキサセトラ抽出器内の抽出挙動の解析に利用可能であることを示した。

2.2 はじめに

最近、Purexプロセスに関する研究は、ウランおよびプルトニウムの効率的な回収からプロセス自身の安全性、経済性あるいは信頼性へと関心が移りつつあり、このような視点から改良型のPUREXプロセスに関する研究¹⁾²⁾も進められている。このようなPurexプロセス自身の修正は、廃棄物の組成およびその発生量に影響するはずである。そのため、改良型のピュレックスプロセスを実用プラントへ適用する前に、廃棄物の発生量およびその組成を、プロセスフローシート計算によって慎重に評価しなければならない。一方、使用済燃料の組成は原子炉型、燃焼度、冷却期間などによって変化するので、原子力発電所から再処理施設に運びこまれた燃料の組成は燃料ごとに異なっている。特に、軽水炉においてMOX燃料を用いた場合、その組成は通常のウラン燃料とは大きく異なるはずである。したがって、使用済燃料ごとに再処理工場の運転条件を、あらかじめプロセスフローシート計算により決定する必要がある。

REPROCX³⁾は、英国Hawell研究所で開発された再処理プロセス専用のフローシート解析システムである。これは、過去にHawell研究所で開発された各工程ごとのシミュレーションコードを連結し、一連の再処理プロセスのシミュレーションを可能としたシステムである。しかしながら、REPROCXは、装置の入れ替えなどのプロセスの変更には容易に対応できない欠点がある。すなわち、プロセスの運転計画には有効である

が、新概念に基づく改良型のピュレックスプロセスの評価には適していない。そこで我々は、コンピュータディスプレイに表示されたフローシートダイアグラム上で直接プロセスの編集ができ、なおかつ計算も実行できるシステムCOMPAS (COMputer aided Process flowsheet design and Analysis System for nuclear fuel reprocessing) を開発した。以下に、本システムの構成、機能およびその使用例を示す。

2.3 システムの構成および機能

プロセスフローシート計算の基本的なアルゴリズムは、1970年代後半までに確立しており⁴⁾、それに基づくソフトウェアも市販されてきた⁵⁾。しかしながら当時はカード入力を主体とした汎用機を使用しており、プロセスの接続情報の変更は容易ではなかった。現在の計算機は、GUI(Graphical User Interface)を備えたワークステーションまたはパーソナルコンピュータが主流でありマウスを使用したインタラクティブな入力が可能である。そこで本システムの開発にあたり、(1)マウスによりコンピュータディスプレイ上に直接フローシートダイアグラムを作成できること、(2)装置の仕様やストリーム内の濃度などの情報はフローシートダイアグラム上で直接見ることでき、その値を編集できること、(3)シミュレーションコードやその他のデータをネットワークで共有できること、を目標とした。これらの機能を実現したプロトタイプが1991年に完成し⁶⁾、本システムはそれを使用済核燃料再処理プロセス用に拡張したものである。

Fig. 2.1 に示すようにCOMPASはユーザインターフェース部分とフローシート計算部分の二つのサブシステムにより構成される。COMPASは、UNIXオペレーティングシステムのOpenWindows⁷⁾またはX-window⁸⁾上で動作する。MIXSET⁹⁾やPULCO¹⁰⁾のようなシミュレーションコードは本システムとは独立に存在する。なお、シミュレーションコードの詳細については次章で述べる。

ユーザインターフェース部分の主な機能は、フローシートを描くことおよび流量、濃度、温度などのプロセス変数の入出力である。COMPASを起動するとMAIN WINDOWとSYSTEM REPORT WINDOWが表示される。Fig. 2.2にそのMAIN WINDOWを示す。MAIN WINDOWは、MENU P ANELとFLOWSHEET P ANELから構成される。MENU PANEL内のボタンをマウスでクリックすることによってCOMPASの様々な機能が使用できる。COMPASのユーザインターフェース部分の機能について以下に要約する。

(1) 装置の選択

"Edit"メニュー内の"Select-unit"ボタンを選択するとFig. 2.3に示すようなUNIT

WINDOWとよばれる装置選択画面が現れる。そこで必要な装置のアイコンをマウスでクリックするとFLOWSHEET PANELにその装置のアイコンが表示される。入出力もまた装置のアイコンとして登録されている。

(2) 装置間の接続

"Edit"メニュー内の"Connect"ボタンによって装置間の接続を行う。MENU PANEL内のMessageに表示される指示に従い、装置と装置のアイコンをマウスによって結ぶとストリームが生成される。このストリームもまたアイコンと呼ばれる。

(3) 装置とストリームの削除

"Edit"メニュー内の"Delete"ボタンをクリックすることによって装置およびストリームの削除を行うことができる。

(4) 装置およびストリームへの数値の入力

数値（文字も含む）は装置およびストリームのアイコンへ直接記入することができる。まず、それらのアイコンをマウスでクリックすることによってSETTING VARIABLES WINDOWを開く。Fig. 2.4にミキサセトラのSETTING VARIABLES WINDOWを示す。ウインドウはいくつかのボタンと数値入力用の表で構成されている。表の左の列には変数名を、右の列にはその値をそれぞれ書き込む。この表の変数名およびその値はユーザが自由に変更できるようになっている。ストリームの色はそこを流れる物理的な相を表現している。例えば、赤は有機相、青は水相または気相である。これらは、ストリームアイコンのSETTING VARIABLES WINDOW内の"PHASE"のチェックボックスを指示することによって変更することができる。もし、ユーザがSETTING VARIABLES WINDOW内の変数を記入しない場合、システムが用意するデフォルトの値が使用される。

フローシート計算部分の機能は、入力から出力へ向かって順に変数の値を計算し、途中のすべてのストリームおよび出力アイコンへ計算値を書き込むことである。ユーザが"Calculation"ボタンをクリックするとこれらの実行がはじまる。システム内部ではまず、フローシート内の装置の接続状況が解析され、リサイクルループの抽出、ストリームのティアリングが行われる。次に入力から出力へ向かって順々にシミュレーションコードを呼び出し計算を行う。シミュレーションコードは、装置内のCODE変数に記入してあるものを呼び出す。例えばCODE変数が"MIXSET"であればMIXSETが呼び出される。リサイクルループが存在する場合、システムは自動的に繰り返し計算を行い解を収束させる。計算結果は、ストリームおよび出力アイコンの表の中に書き込まれる。

COMPASシステム自身およびシミュレーションコードはネットワークにより共有さ

れている。ネットワーク機能はUNIXオペレーティングシステムで提供されているものを利用している。

2.4 COMPAS で利用できるコード

前にも述べたように、シミュレーションコードとCOMPASとは独立しているため、どのような言語で書かれていても、UNIXオペレーティングシステム上で実行可能なシミュレーションコードであればすべて利用可能である。これまでに開発されたORIGEN2¹¹⁾やMIXSETのようにFORTRANで記述されたコードは当然利用可能である。しかしながら、それらコードの入出力フォーマットはそれぞれ異なるため、COMPAS入出力フォーマットとの変換プログラムが必要となる。現在のところ、COMPASにはORIGEN2、MIXSETおよびPULCOの変換プログラムが用意されている。

2.4.1 使用済み燃料の組成計算

再処理プロセスのフローシート計算において、発電所から運びこまれる使用済燃料の組成を計算することが必要である。本システムでは使用済燃料の組成計算にORIGEN2を使用することができる。現在のところ、PWRおよびBWRの濃縮ウラン燃料について計算が可能である。Fig. 2.5に示すように、炉形式、ウラン濃縮度、比出力、燃焼度および冷却期間を指定することによって、ウラン1トンあたりの使用済燃料中に含まれる各核種の重量が出力される。

2.4.2 溶解工程

溶解工程では、剪断工程から供給された燃料ピンを硝酸に溶解し、溶解液（液相）、オフガス（気相）およびハル（固相）に分離する。本システムには、連続式およびバッチ式の溶解槽のシミュレーションコードが実装されている。本コードは、硝酸濃度、溶解液の供給量および温度などの操作条件ごとに、各相の総量あたりの重量分率を出力する。ウランの溶解率およびNO_x発生量は古閑ら¹³⁾および本間ら¹⁴⁾の速度式を用いて算出した。他成分の溶解率およびヨウ素などの気体成分の発生量はユーザーが指定することにした。

2.4.3 抽出工程

Purexプロセスにおいて溶媒抽出工程は重要な役割をはたす。これまでに抽出器内のウランおよびプルトニウムの挙動をシミュレートするコードが開発されてきた。SEPHIS、MIXSETなどのコード(15)(16)(17)(18)は例外なくCOMPASに実装することができる。現段階ではそれらのうちのMIXSETおよびPULCOが利用可能である。また、4章で述べるが、ミキサセトラ内のNpの挙動をシミュレートするコードも実装済である。ミキサセトラ型抽出器では、ステージ数、ミキサおよびセトラの体積、使用するシミュレーションコードなどをそのアイコン内の変数に指定し、入口の流量および成分濃度が与えられれば抽出器から排出される水相および有機相の各成分濃度が計算される。Fig. 2.6のようにミキサセトラ抽出器の途中に溶液を供給する場合、装置"mix"を使用することによって2つのミキサセトラとして計算することができる。この装置"mix"は2つのストリームを1つのストリームに合流させる場合に使用する。このほかに、COMPASではパルスカラムや遠心抽出器についても同様に利用可能である。

2.4.4 その他のプロセス

COMPASでは上記の装置のほかに、Fig. 2.3にも示したとおり剪断器および供給液調整装置が用意されている。剪断器のシミュレーションコードでは発生するオフガスの量を計算する。現段階では、これらの発生量はユーザーが指定しなければならない。供給液調整装置は、溶解槽から流れてきた硝酸溶液を次の共除染工程へ供給する際に必要となる。ここでは、硝酸濃度、供給液流量およびウラン濃度を指定する。その他の成分濃度は、重量分率をユーザが指定したウラン濃度 (g/l) を基準とし、自動的に換算して出力する。

上述の装置以外は現在のところCOMPASには実装されてはいない。しかしながら、装置の登録は容易に行うことができる。登録の方法を簡単に述べると、(1)シミュレーションコードをFORTRAN、Cなどの言語で作成する。このとき入力にはUNIXの標準入力（キーボードからの入力、FORTRANでは装置番号5）から受取り、出力は標準出力（画面への表示、FORTRANでは装置番号6）へ送るようにプログラムする。(2)入出力の変数の順序を記述したテキストファイルを用意する。(3)X-Windowなどに添付のプログラム（例えば、Bitmap Editor）によってアイコンの絵を作成する。(4)デフォルト値を指定するため定義ファイルを書き換える。以上で登録が終了し、COMPASをリスタートすればUNITウインドウに登録した装置のアイコンが表示される。

2.5 適用例

2.5.1 Np のマスフロー制御のための操作条件の検討

TRU元素の中でネプツニウムを他の成分からどのように分離するかは非常に興味深い問題である。Purexプロセスでは、次の3つの方法が考えられる。

- (1) 共除染工程でネプツニウムを高レベル廃液側に移し、その後廃棄物から分離する。
- (2) U-Pu分配工程でネプツニウムをウラン製品側へ移しウランと共に燃料としてリサイクルする。
- (3) U-Pu分配工程でネプツニウムをプルトニウム製品側へ移しプルトニウムと共に燃料としてリサイクルする。

このうち(2)と(3)は、(1)と比較して新たなプロセスの追加を必要とせず、さらにネプツニウムを燃料サイクルへ閉じ込めることが可能であることから、ネプツニウムを取り扱う方法としては望ましいと考えられる。しかしながら、ネプツニウムは硝酸溶液中でIV、VおよびVIの原子価をとり、そのうちIV価およびVI価だけがTBPに抽出されることが知られている。また、それらの原子価は、他の成分あるいは共存するイオンの影響でさまざまに変化することから、ネプツニウムの抽出挙動は非常に複雑である。ネプツニウムの流れを制御するためには、その原子価を何らかの方法で制御することが必要である。本研究では、COMPASを用い、フローシート上でネプツニウムの流れを計算し、(2)と(3)の操作を達成できるような操作条件について検討した。

Fig. 2.7にCOMPAS上に実現したPUREXプロセスを示す。フローシート中には、発電炉、剪断機、バッチ式溶解槽、給液調整装置および4つのミキサセトラのアイコンが表示されている。入力アイコンNo.4は発電炉を表現しているが、実際には再処理工場の燃料貯蔵プールである。入力アイコンNo.0および1は、共除染工程へのスクラブ液および抽出溶媒、また、No.2および3はU-Pu分配工程への抽出溶媒およびストリップ液を表している。出力アイコンNo.8および9はともにオフガス、No.15は高レベル廃液、No.20はプルトニウム製品、No.21はウラン製品への流れをそれぞれ表す。ストリーム番号(11, 13, 12)および(17, 18, 16, 19)はリサイクルループである。

ミキサセトラ内のネプツニウムの抽出挙動をシミュレートするために新たに計算コードを作成した。ネプツニウムの分配係数データ(20, 21)および反応速度データ(19)は既往の研究で得られた値を使用した。シミュレーションコードのプロトタイプもCOMPASを使用して作成した。作成の方法は次節で述べる。

(2)および(3)の方法でネプツニウムを取り扱うためには、共除染工程に供給されたネプツニウムを、できるだけ多く次のU-Pu分配工程へ流す必要がある。本研究では、これを実現するために、共除染工程へ供給する溶媒の流量（入力アイコンNo.1）を変化させることを試みた。本研究では、計算にあたり次のようなプロセスを想定した。共除染工程は2バンクのミキサセトラからなり、抽出部は10段、スクラブ部は5段である。1段の容積は、スクラブ部および抽出部がそれぞれ 2.0×10^{-2} および $3.0 \times 10^{-2} \text{m}^3$ である。表1に共除染工程に供給する溶液（ストリームアイコンNo.10）の組成を示す。3.0%濃縮ウラン燃料中のネプツニウムの生成量はORIGEN2によって計算した。炉形式はPWRを想定し、燃焼度20,000、30,000および40,000 MWd/U-tonについてそれぞれ計算を行った。共除染工程に供給するU(VI)濃度は250g/lに固定し、その他のプルトニウムおよびネプツニウムの濃度は、ORIGEN2の計算結果に基づき、U(VI)濃度250g/lに対して比例計算で求めた。なお、ネプツニウムのVI価およびV価の濃度比、 $[\text{Np(VI)}]/[\text{Np(V)}]$ は0.43に固定した。硝酸濃度および亜硝酸濃度は、すべての燃焼度に対してそれぞれ3.0および $2.0 \times 10^{-2} \text{mol/l}$ に固定した²²⁾。表2に共除染工程に供給するスクラブ液（入力アイコンNo.0）およびU-Pu分配工程に供給するストリップ液（入力アイコンNo.3）の組成を示す。

フローシート計算は、それぞれの燃焼度ごとに共除染工程へ供給する溶媒の流量（入力アイコンNo.2）を変化させて行い、ネプツニウムの回収率を求めた。回収率は、共除染工程に供給されたネプツニウムの総量に対してU-Pu分配工程に回収されたネプツニウムの量の比と定義した。すなわち、すべてのネプツニウムがU-Pu分配工程へ流れた場合、回収率は100%である。Fig. 2.8にネプツニウムの回収率と共除染工程へ供給する溶媒の流量との関係を示す。図より、回収率は溶媒の供給量を増大させればそれに従い単調に増加していることがわかる。また、使用済燃料の燃焼度にはそれほど依存していないことがわかった。以上より、共除染工程に供給する溶媒の流量を制御することが、ネプツニウムをU-Pu分配工程側へ回収するひとつの方法であることがわかった。

ここで注意しておかなければならないのは、実際のプロセスにおいて、このような単純な方法でネプツニウムを回収できるとは限らないことである。なぜならば、共除染工程に供給されるネプツニウムのV価とVI価の存在比および亜硝酸濃度は前段階である溶解工程の運転条件に依存し、この挙動は未だに明かとなっていないからである。Purexプロセス内のネプツニウムの挙動については、さらなる研究が必要である。ともあれ、COMPASが、ネプツニウムの流れを制御するためのプロセスの操作条件の検討に有効であることが示された。

2.5.2 ミキサセトラ内の抽出挙動の解析

COMPASはプロセスフローシート計算だけでなくミキサセトラ内の抽出挙動の解析(9)にも使用できることを次に示す。このような利用方法は研究者にとって非常に興味深いものである。Fig. 2.9にCOMPAS上に実現したミキサセトラの1バンクを示す。1バンクは、5ステージで構成されており、各々のステージは下向きの白抜き矢印をもつ正方形のアイコンで表されている。1つのステージに関するシミュレーションコードは、次の単純な物質収支式から作成した。

$$X_{out, j} = \frac{Q_{x, in} X_{in, j} + Q_{y, in} Y_{in, j} + S_j}{Q_{x, out} + Q_{y, out} D_j} \quad (1)$$

$$Y_{out, j} = D_j X_{out, j} \quad (2)$$

ここに、

- D 分配係数
- Q 流量
- S 反応による生成量
- X 水相中の成分濃度
- Y 有機相中の成分濃度

添え字は、

- in 入力
- out 出力
- j 成分
- x 水相
- y 有機相

をそれぞれ表す。考慮した成分はU(VI)、Pu(IV)および硝酸である。各段はすべての成分に対して平衡を仮定した。3章で述べたように、ミキサセトラの中間部からの入力は、装置"mix"を使用して実現した。"calculation"ボタンをクリックすると、収束解を得るまで自動的に繰り返し計算を実行する。計算結果は、通常どおり各ストリームのアイコンをクリックすれば表示することができるが、Fig. 2.10に示すように濃度プロフィールをグラフの形式で出力することも可能である。このように、1ステージのシミュレーションプログラムを作成し、フローシート上でそれらのステージを接続するだけで、1バンクのミキサセトラのシミュレーションコードが完成し計算を自動的に行うことができる。すなわち、COMPASは、繰り返し計算のアルゴリズムと入出力のインターフェースを提供しているため、研究者はそれらのプログラム作成から開放され、抽出や反応の解析に集中できることになる。

以上より、COMPASはシミュレーションコードのプロトタイプ作成、実験室規模の実験の計画にも有効であることがわかる。なお、ミキサセトラ以外のパルスカラム型抽出器あるいは遠心抽出器に対しても数学的なモデルを変更することによってほぼ同様に使用可能である。

2.6 まとめ

COMPASは、再処理プロセスのフローシート計算を行うために開発された。フローシートの作成は、マウスの操作で簡単に行えるため、フローシートの変更にも容易に対応できる。計算結果の表示もマウスで簡単に行え、グラフによる出力も可能である。2つの使用例により、COMPASが、再処理プロセスのフローシート計算だけでなく1つの抽出装置のシミュレーションにも有効であることを示した。COMPASは、使用済燃料再処理の研究および開発のあらゆる面において適用可能である。

2.7 引用文献

- 1) HORWITZ, E.P. et al.: *Solvent Extr. Ion Exch.*, 3, 75(1985).
- 2) PETRICH, G. et al.: *Proc. ISEC90*, 555(1992).
- 3) HARRISON, J.W. et al.: *Proc. ISEC86*, 1, 131(1986).
- 4) WESTERBERG, A.W. et al.: "Process flowsheeting", (1979), Cambridge Univ. Press.
- 5) BIEGLER, L.T.: "Chemical Process Simulation", *Chem. Eng. Prog.*, 85[10], 50 (1989).
- 6) 坂本進、本間俊司、古閑二郎、松本史朗：埼玉大学紀要、工学部、25, 33(1991).
- 7) HELLER, D.: "XView Programming Manual", (1990), O'Reilly & Associates, Inc.
- 8) NYE, A.: "Xlib Programming Manual", (1990), O'Reilly & Associates, Inc.
- 9) GONDA, K., OKA, K., FUKUDA, S.: PNCT-841-79-26, (1979).
- 10) GONDA, K., MATSUDA, T.: PNCT-841-82-19, (1982).
- 11) CROFF, A.G.: *Nucl. Technol.*, 62, 335(1983).
- 12) CROFF, A.G.: ORNL-TM-7175, (1980).
- 13) KOGA, J., HOMMA, S., KANEHIRA, O., MATSUMOTO S., GOTO, M., YASU, T., KAWATA, T.: *RECOD91*, vol.II, 687(1991).
- 14) HOMMA, S., KOGA, J., MATSUMOTO, S., KAWATA, T.: *J. Nucl. Sci. Technol.*, 30[9], 959(1993).
- 15) GROENIER, W.S.: ORNL-4746, (1972).

- 16) RICHARDSON, G.L., SWANSON, J.L.: HEDL-TME75-31, (1975).
- 17) WATSON, S.B., RAINEY, R.H.: ORNL-TM-5123, (1975).
- 18) MITCHELL, A.D.: ORNL-TM-6565, (1979).
- 19) HOMMA, S., KOGA, J., MATSUMOTO S.: Evaluation of Aqueous Reactions in PUREX Extraction Process, *RECOD'94*, vol.III(1994).
- 20) PETRICH, G., KOLARIK, Z.: KfK3080, (1981).
- 21) HOMMA, S., TAKANASHI, M., KOGA, J., MATSUMOTO S.: *Sci. Eng. Reports Saitama Univ.*, 27, 5(1993).
- 22) TSUBOYA, T., TANAKA, T., NEMOTO, S., HOSHINO T.: *Proc.ISEC74*, 1985 (1974).

Table 2.1 Flow rate and concentrations fed into co-decontamination process

Burn-up	[MWd/Ut]	20,000	30,000	40,000
Flow rate	[m ³ /h]	5.0×10^{-2}	5.0×10^{-2}	5.0×10^{-2}
U(VI)	[g/l]	250.0	250.0	250.0
Pu(IV)	[g/l]	1.85	2.38	3.20
Np(V)	[g/l]	5.05×10^{-2}	1.09×10^{-1}	1.92×10^{-1}
Np(VI)	[g/l]	2.16×10^{-2}	4.67×10^{-2}	8.25×10^{-2}
HNO ₃	[mol/l]	3.0	3.0	3.0
HNO ₂	[mol/l]	2.0×10^{-2}	2.0×10^{-2}	2.0×10^{-2}

Table 2.2 Flow rate and concentrations in scrub and strip solution

	Scrub solution	Strip solution
Input icon in Fig.7	No.0	No.3
Flow rate [m ³ /h]	5.0×10^{-2}	3.2×10^{-2}
HNO ₃ [mol/l]	3.0	2.0
U(IV) [g/l]	-	71.4
Hydrazine [mol/l]	-	0.5

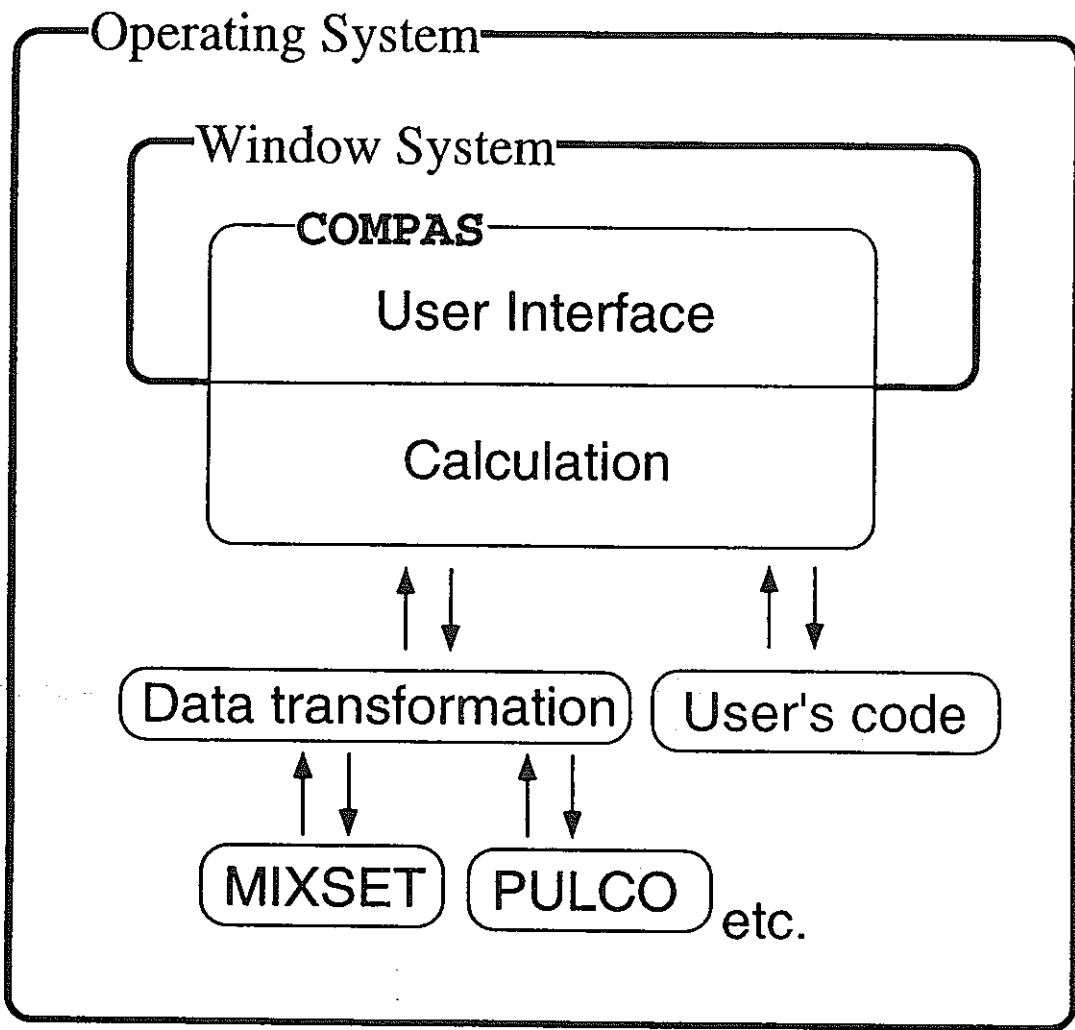


Fig. 2.1 Homma et. al.

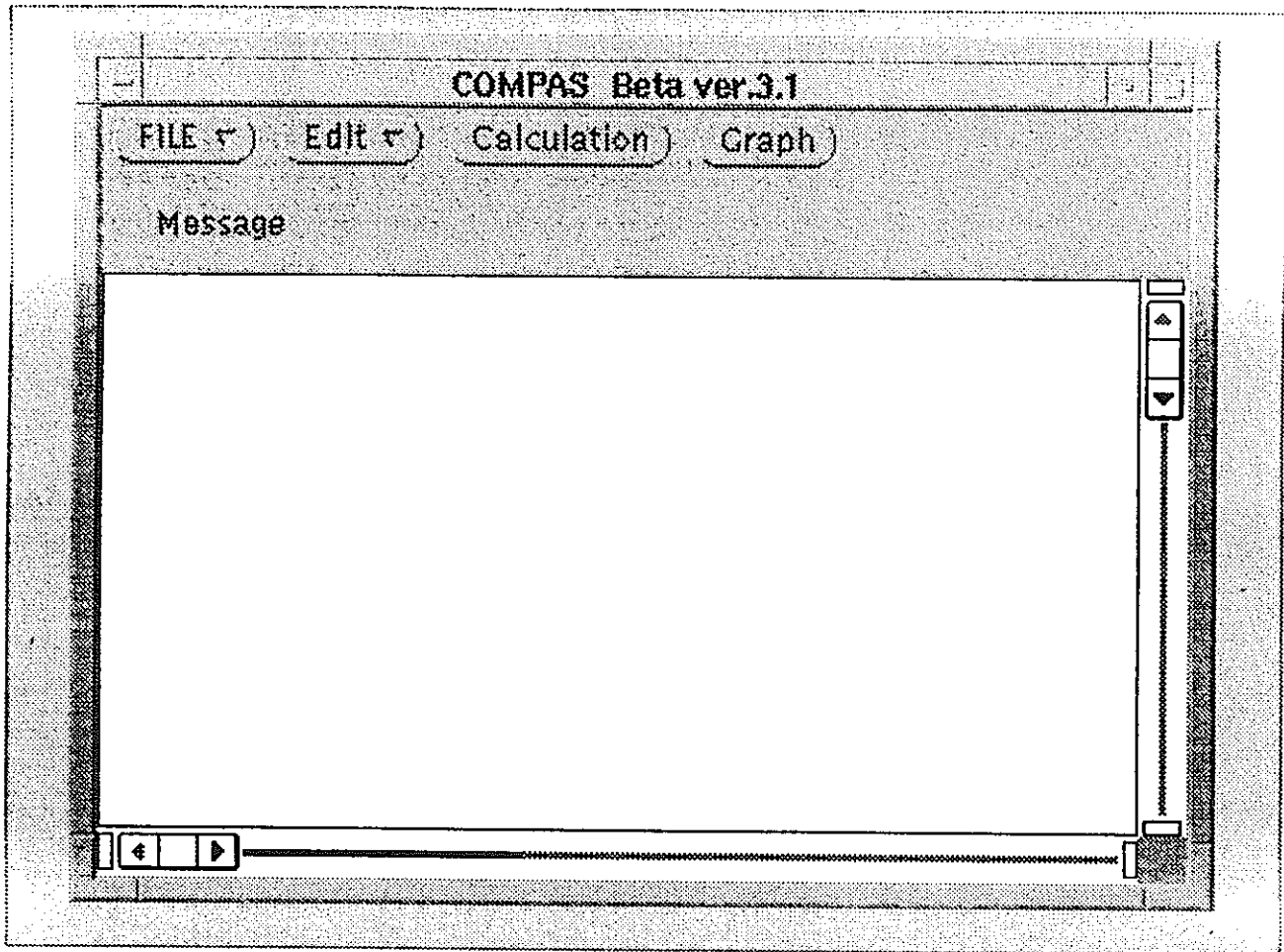


Fig. 2.2 Homma et. al.

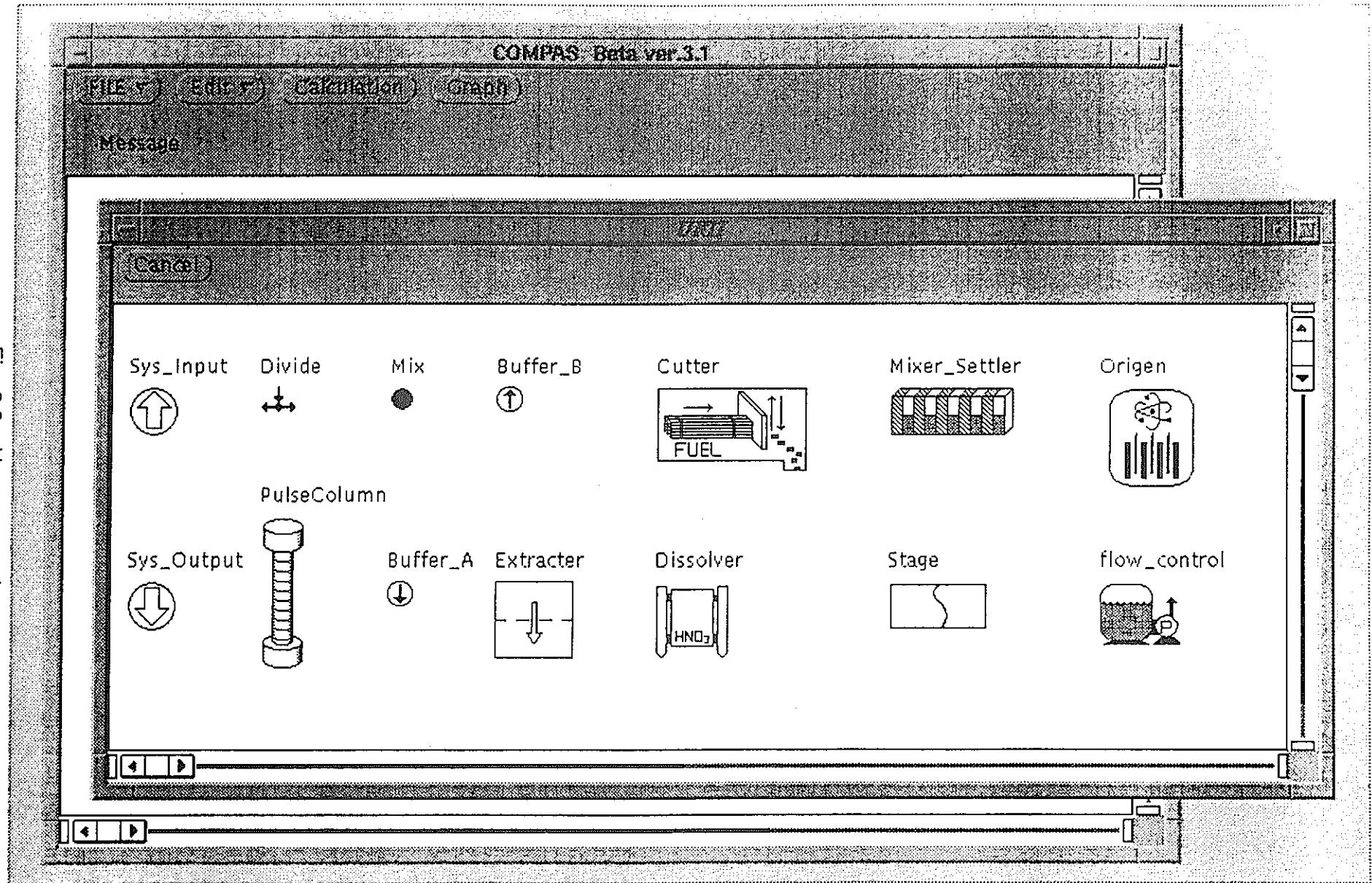


Fig. 2.3 Homma et. al.

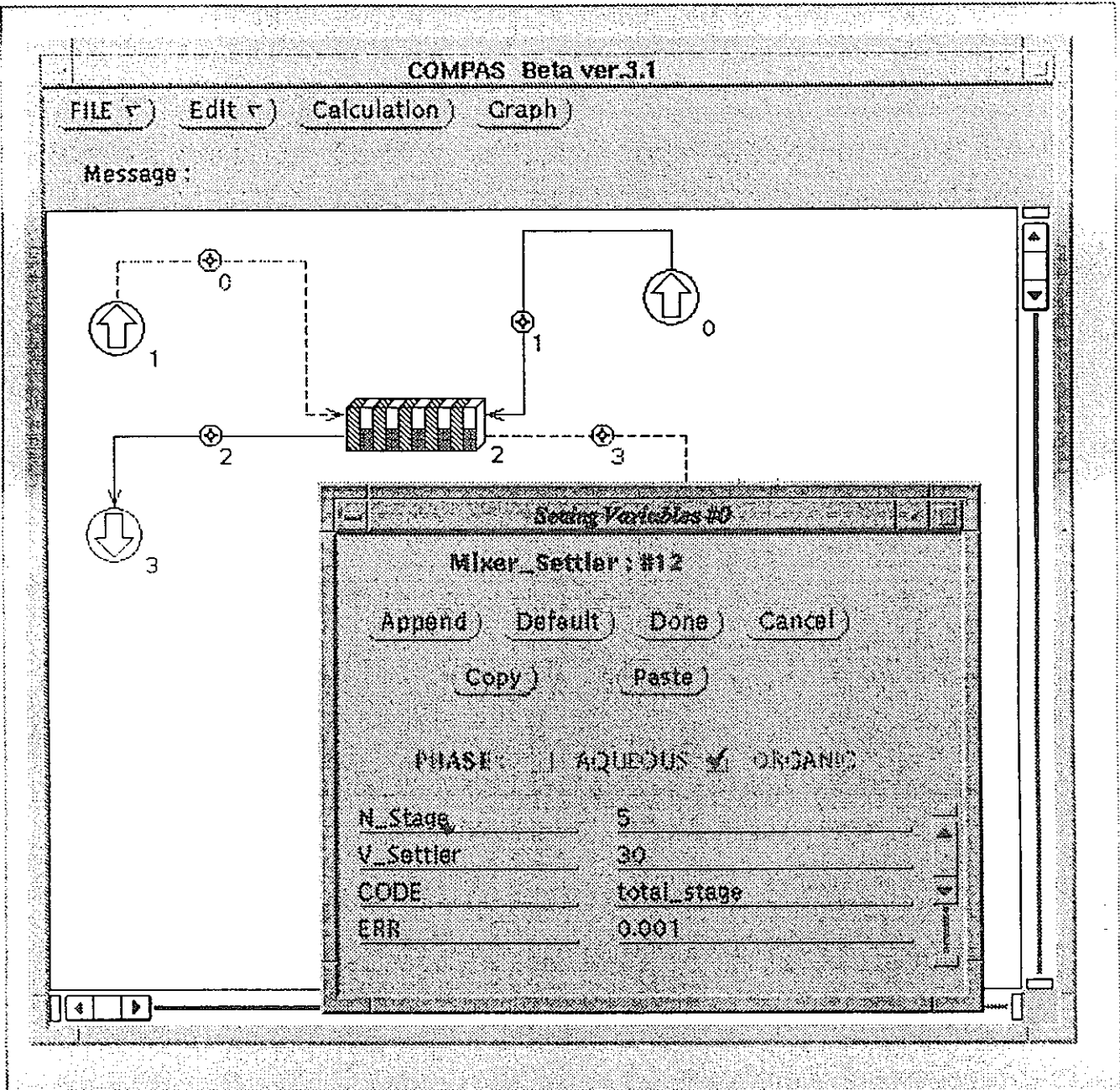
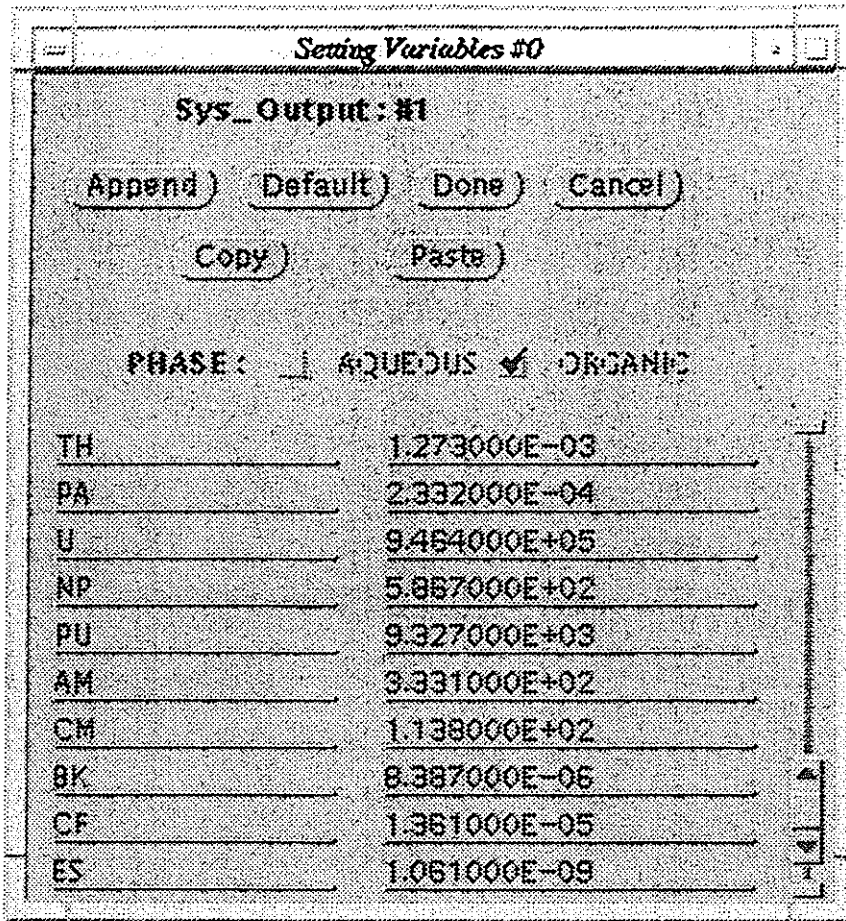
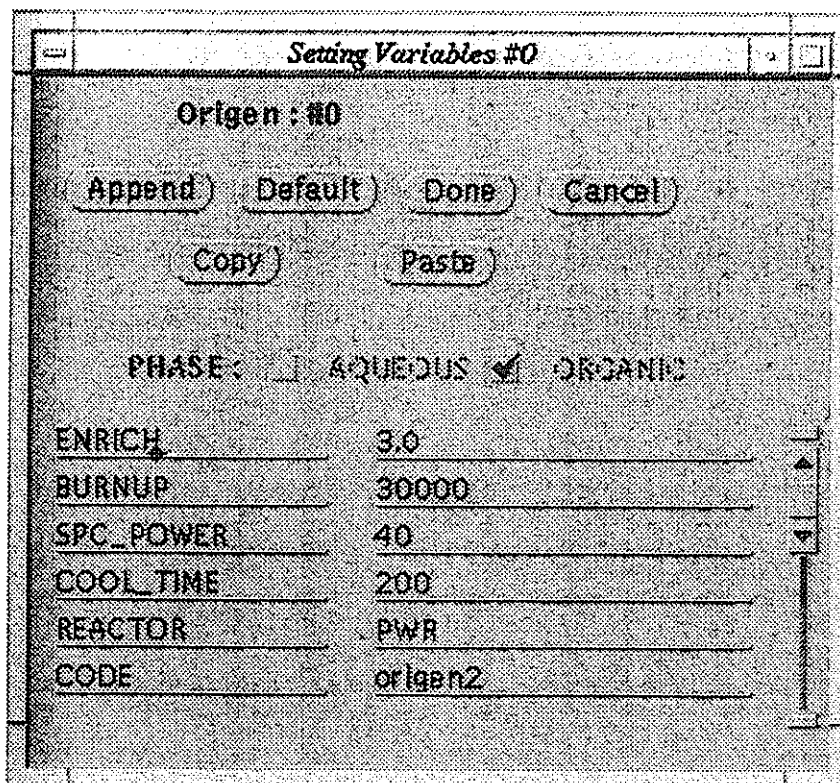


Fig. 2.4 Homma et. al.



Output Window



Input Window

Fig. 2.5 Homma et. al.

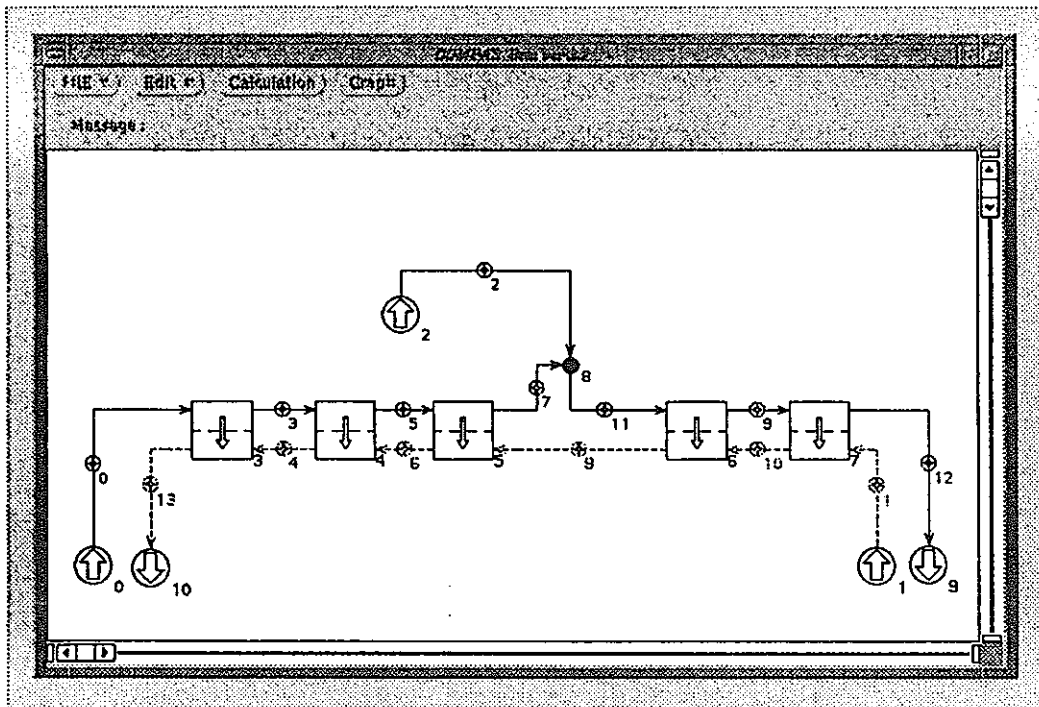
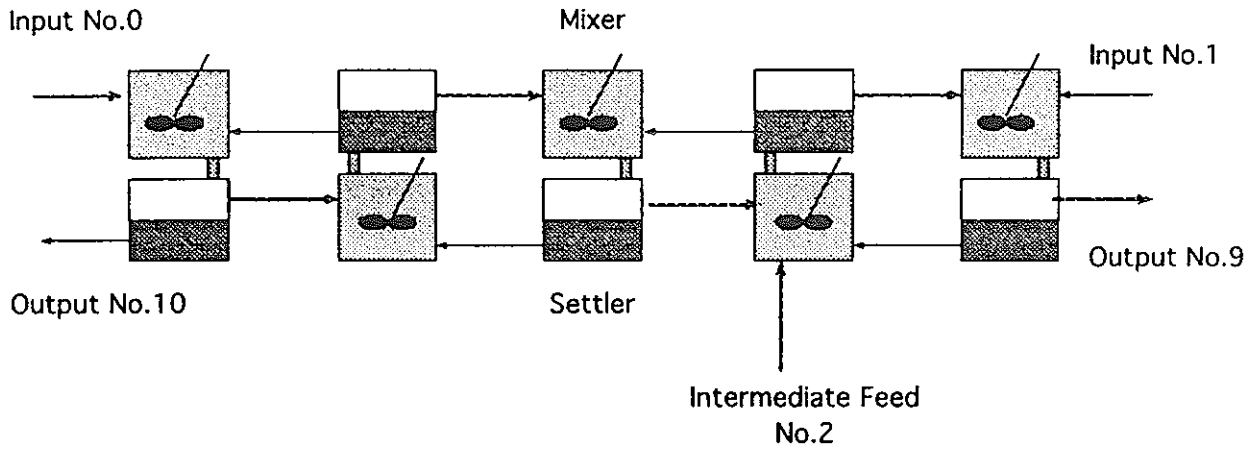


Fig. 2.6 Homma et. al

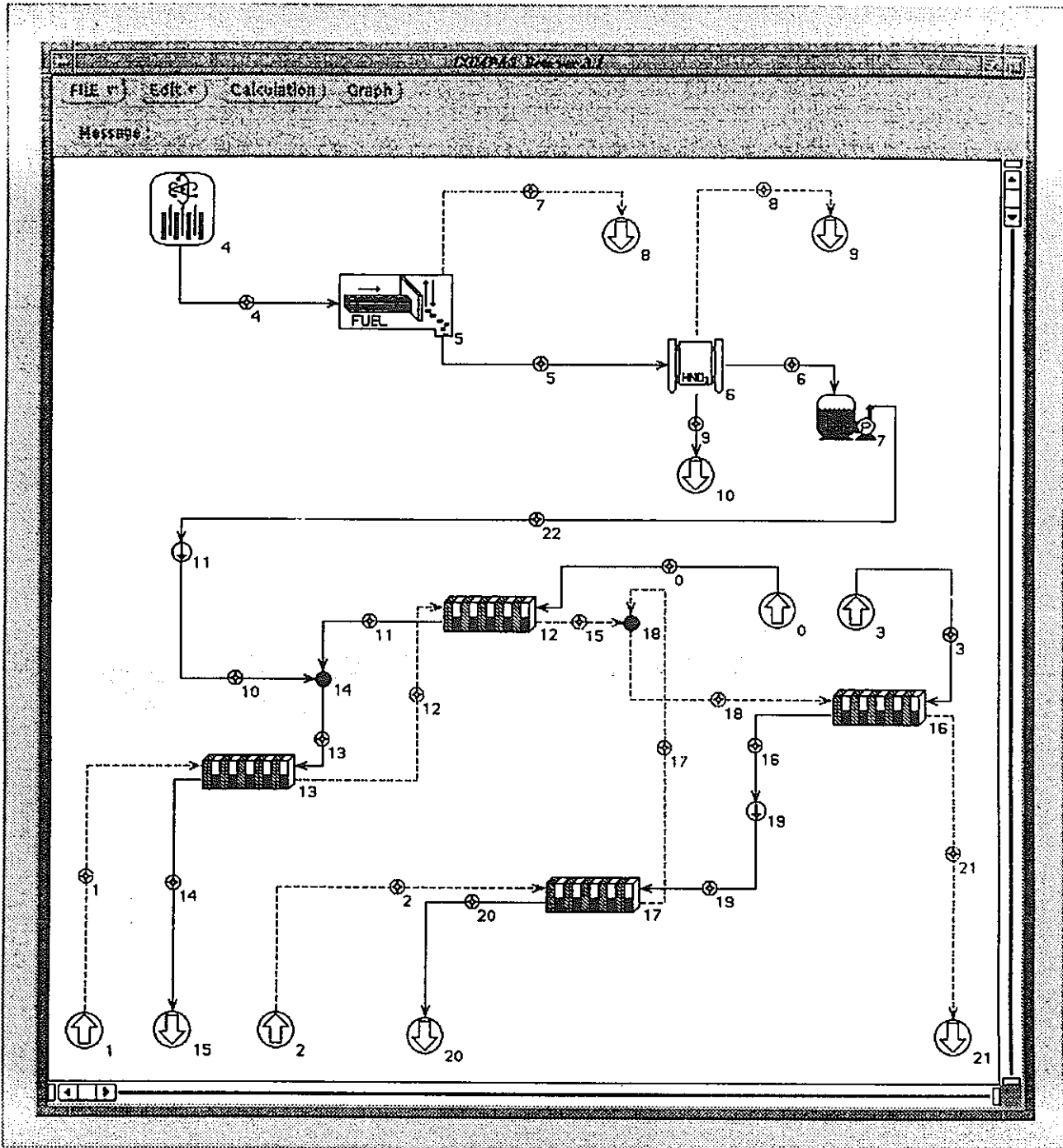


Fig. 2.7 Homma et. al.

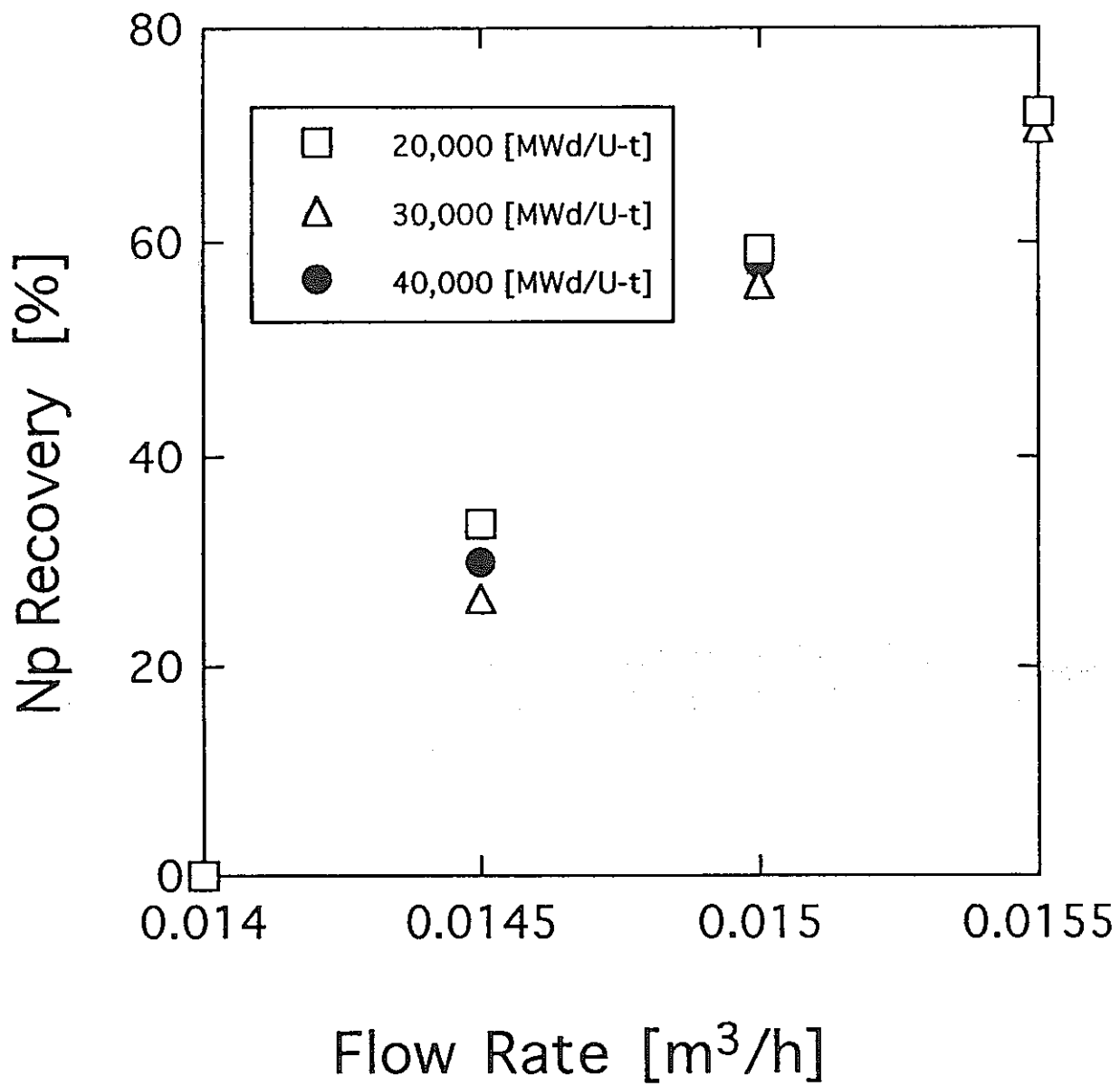


Fig. 2.8 Homma et. al

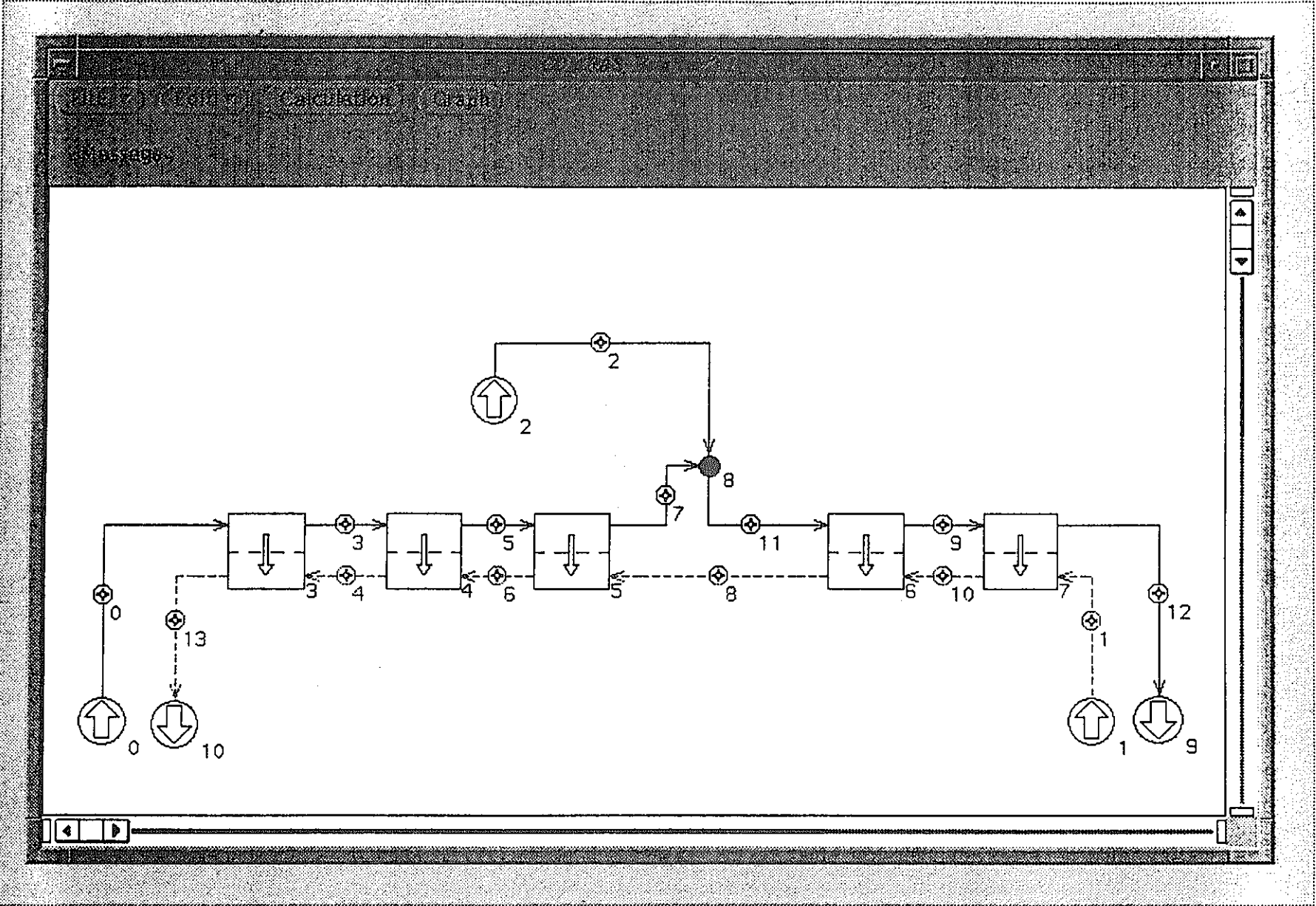


Fig. 2.9 Homma et. al.

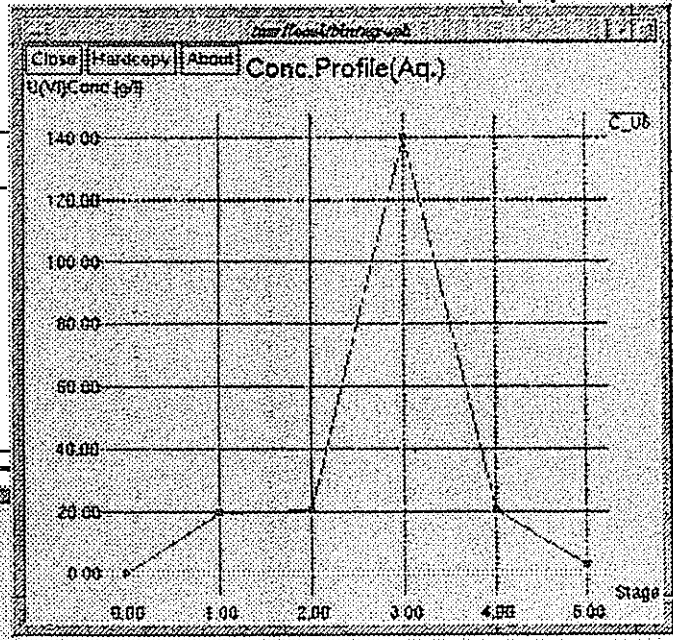
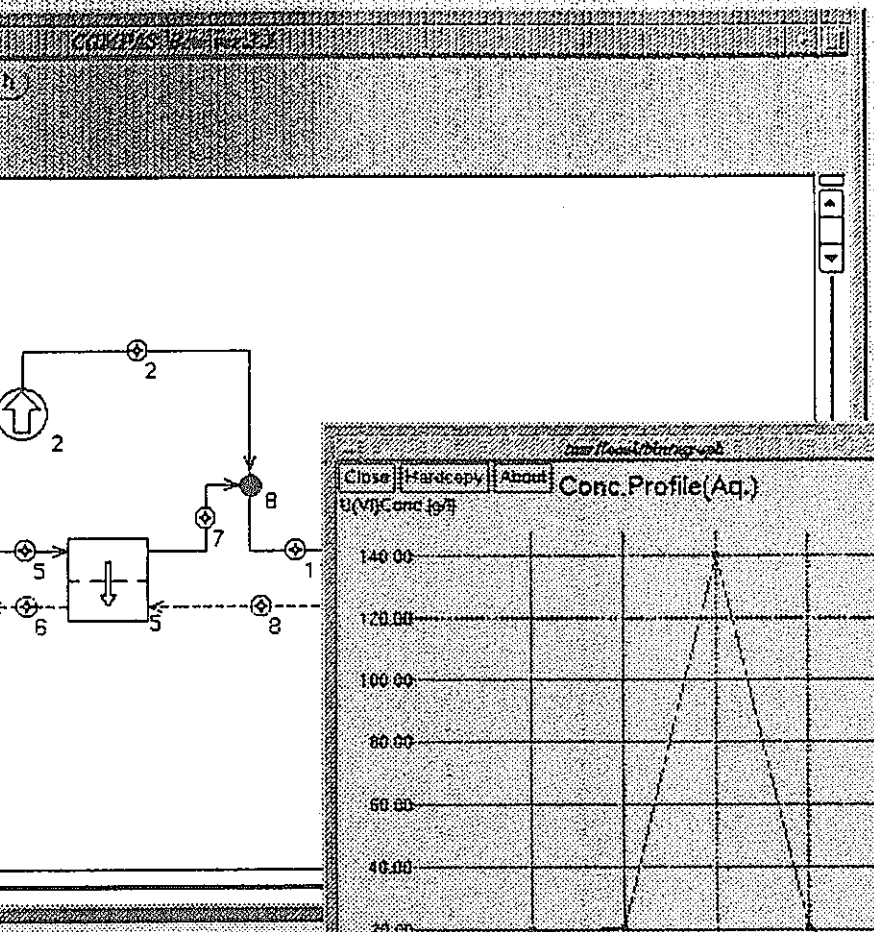
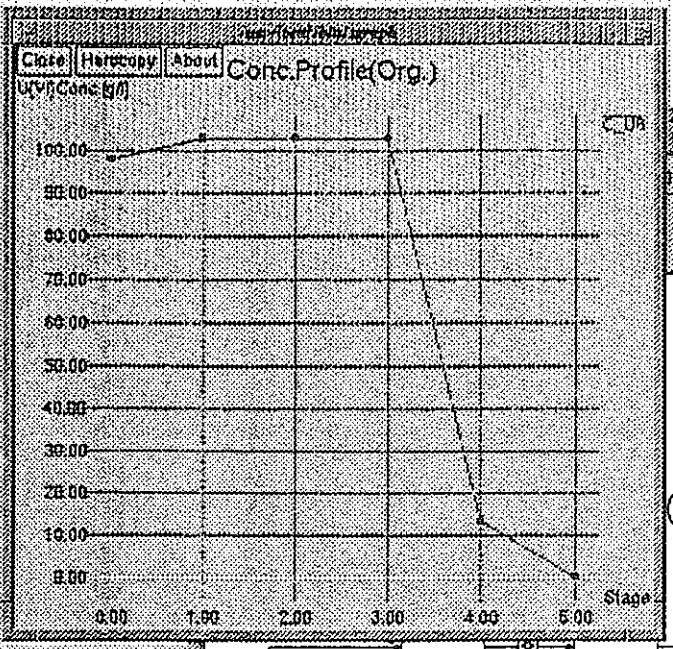


Fig. 2.10 Homma et al.

3. 廃液処理およびオフガス処理工程のフローシート計算

使用済核燃料再処理プロセスにおける廃液処理およびオフガス処理工程のフローシート計算をCOMPAS上で行うために、それぞれのフローシートを計算機上に作成した。これらのフローシートは、動燃の報告に基づいて構成されている。

Fig. 3.1 に示した廃液処理工程では、共除染工程やウランあるいはプルトニウム精製工程から排出される廃液の処理を行っている。処理される廃液は、図中の1および2からこの工程に供給されている。ここで、図中の1は、分離第一サイクルから、2は、分離第二サイクルやウランあるいはプルトニウム精製工程から排出される廃液を表している。図中の3は廃液の濃縮を行う蒸発缶であり、ここで蒸発濃縮された廃液は、図中の4から高放射性廃液貯槽に送られ貯蔵される。図中の5および6は、それぞれ、酸回収のための蒸留塔および蒸発缶である。

また、Fig. 3.2 に示したオフガス処理工程では、使用済燃料の剪断工程および溶解工程から発生する気体廃棄物の処理を行っている。処理される気体廃棄物は、図中の1および2からこの工程に供給されている。ここで、図中の1は、使用済燃料の剪断工程から、2は、溶解工程から排出される気体廃棄物を表している。

ただし、これらの工程のフローシート計算をCOMPAS上で行うためには、これらのプロセスを構成する各装置のシミュレーションコードを整備することが必要である。

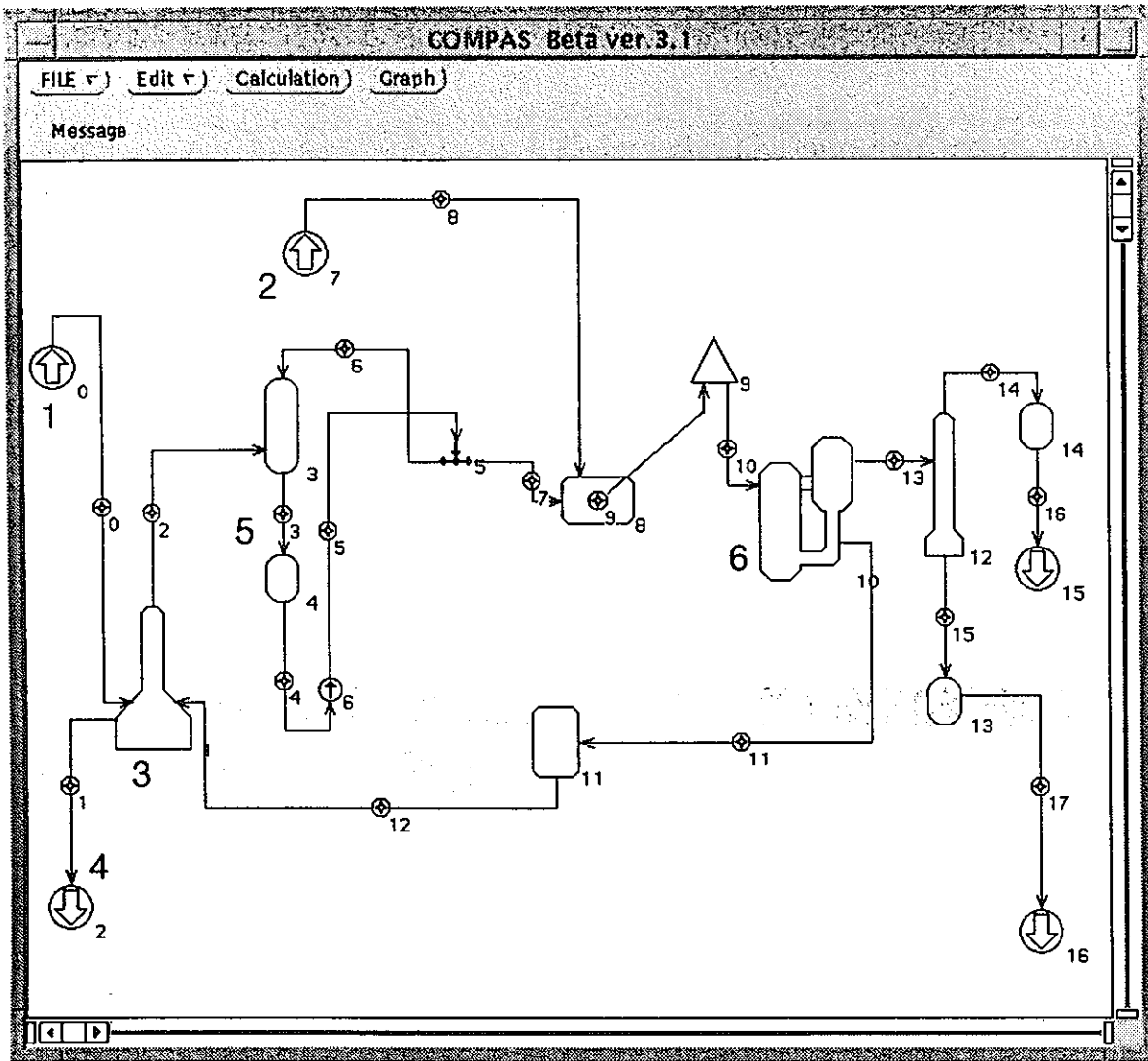


Fig. 3.1 廃液処理フローシート

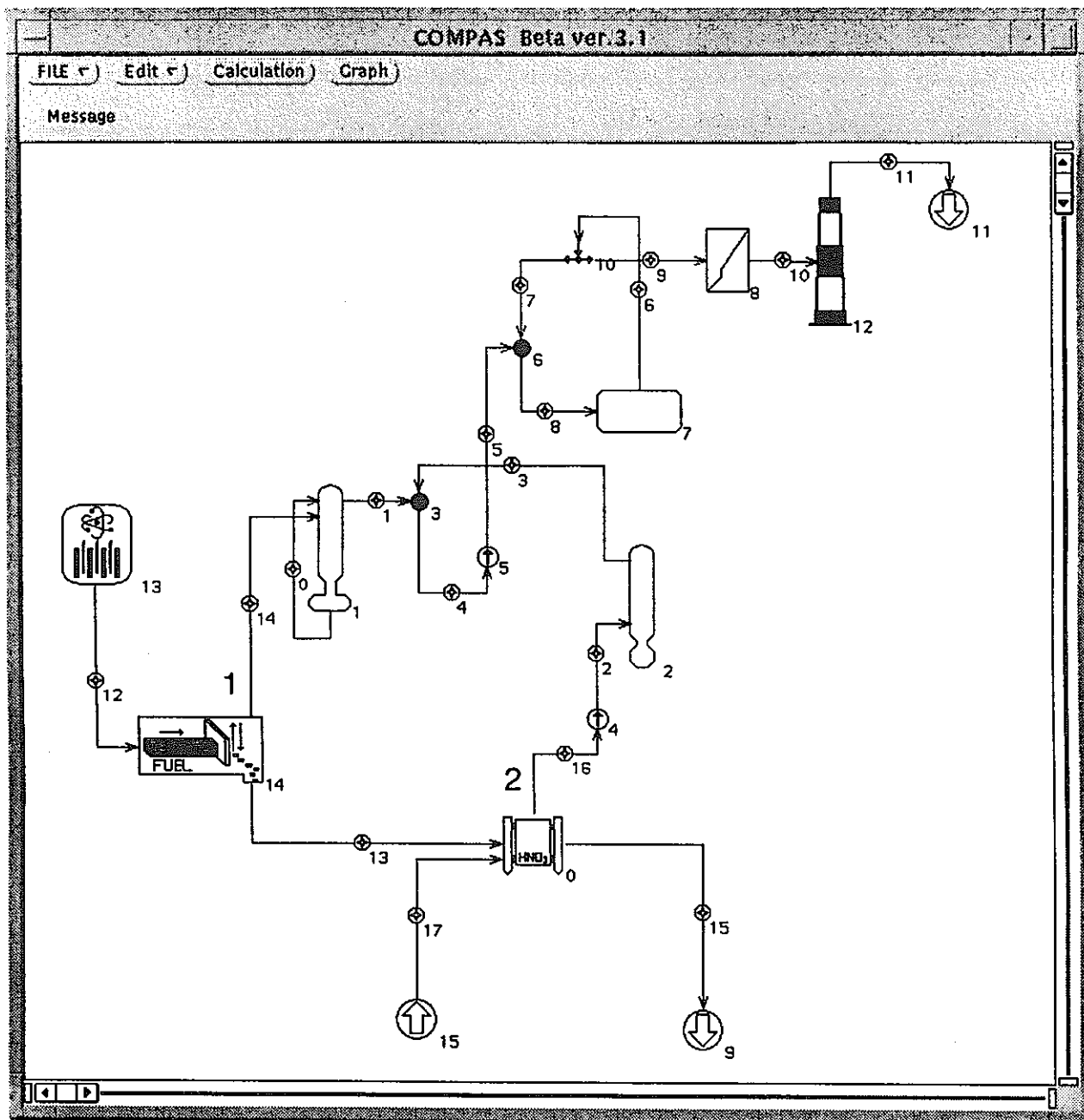


Fig. 3.2 オフガス処理フローシート