

限定資料

ニアフィールド現象の変遷に関する 動的解析手法の開発

(平成5年度)

平成6年3月

技術資料		
開示区分	レポートNo.	受領日
8	J1281 94-007	1995.2.3
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		

日揮株式会社

この資料は、動燃事業団の開発業務を進めるため、特に限られた関係者だけに開示するものです。ついては、複製、転載、引用等を行わないよう、また第三者への開示又は内容漏洩がないよう管理してください。また今回の開示目的以外のことに使用しないよう特に注意してください。

本資料についての問い合わせは下記に願います。注)

〒107 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

ニアフィールド現象の変遷に関する
動的解析手法の開発

(平成5年度)

平成6年3月

日揮株式会社

要 旨

高瀬博康*

動的解析の手法開発として、人工バリア内が拡散支配である限り様々な条件変動やコロイド形成等の新たな現象を考えた場合にも共通して適用可能な広義拡散方程式を導出し、具体的なFEPsに対応するものとして、

- (1) 瞬時平衡、線型吸着、溶解度限界モデル
- (2) 真性コロイドモデル
- (3) 擬似コロイドモデル
- (4) 真性コロイド、擬似コロイド共存モデル
- (5) 非線型吸着モデル
- (6) 水理・化学条件の時間・空間的变化を考慮したモデル
- (7) 沈殿・溶解反応の速度論的表現を加えた真性コロイド、擬似コロイド共存モデル

を作成した。

また、この解析手法を実用に供するために、汎用DAEソーパーSPADEとグラフィカルユーザーインターフェイスSAMGUIを改良した。主たる改良点は、

- (1) SPADEの自動差分機能の追加、方程式入力フォーマットの簡単化、リスタート機能、及び多数回の計算を一括処理するためのユーティリティプログラムの付加
- (2) SAMGUIへのデータ移送と作画の自動化

である。

*日揮株式会社 原子力・環境・エネルギー事業本部

更に、改良されたソフトウェアを用いて、Tc-99を例として決定論的及び統計的感度解析を予備的に実施した。

決定論的解析としては、瞬時平衡・線型吸着・溶解度限界モデルによる計算を行ない、これを標準ケースとして以下のモデルの変動を考えた場合の結果の変化を合計約400ケースについて解析した。

- (1) 真性コロイドモデル
- (2) 擬似コロイドモデル
- (3) 真性コロイド、擬似コロイド共存モデル
- (4) 水理条件の経時変化を考慮したモデル
- (5) 化学条件の（空間的に一様な）経時変化を考慮したモデル
- (6) 化学条件の（空間的に不均質な）経時変化を考慮したモデル

結果として、真性コロイドが存在する場合、大きな核種放出率の増加が生じ得ることがわかった。

統計的解析としては、標準ケースからのモデル及びパラメータの変動を統計的に作成した感度解析を行った。この結果に基づき、決定論的感度解析の網羅性を補完するものとしての位置付けが明らかとなった。特に、人工バリア周辺の地下水条件が一定期間より好ましいものとなりその後元の状態に戻った場合、ゆるみ域が核種についての化学的シンクとなり、一旦ここに沈殿した核種が条件復帰後に一気に放出されることにより、元の化学条件では見られない過渡的なピーク放出率が現れることが示された。このような効果は、従来の決定論的な思考では見落されていたものであり、二つの手法を併用することの利点を示す例とすることができる。

以上の結果から、人工バリアの長期的挙動に関して予想される幅広い前提条件と多岐にわたる現象を共通の手法で網羅するものとして本研究で開発した動的解法手法が有効であることがわかった。

この成果を受け、次年度には、平成三年度報告書で対象とした全ての核種について、多くのFEPsを考えた解析を行うことが可能となった。この様な解析の実施にあたっては、感度解析をより体系的かつ効率的に行える様にするために、SPADEのフロントエンドを若干改良することが必要である。

SUMMARY

Hiroyasu Takase*

The generalized diffusion model, which can be applied to variety of phenomena provided mass transport is diffusion dominated, was developed. Utilizing this general configuration, following specific models were formulated;

- (1) usual instantaneous solubility limited, linear sorption model,
- (2) true colloid model,
- (3) pseudo colloid model,
- (4) co-existing true/pseudo colloid model,
- (5) nonlinear sorption model,
- (6) time/space dependent chemical/hydrological condition model,
- (7) co-existing true/pseudo colloid model with kinetic precipitation/dissolution reaction.

In order to make calculations using the above methodology, SPADE (general DAE solver) and SAM GUI (graphical user interface) were modified. Major modifications are;

- (1) addition of following functionalities to SPADE;
 - automatic discretization
 - user friendly format of equation input
 - restart
 - front end to process large number of calculations,
- (2) automation of SPADE results transport to SAM GUI and of subsequent visualizations.

Furthermore, a set of preliminary sensitivity analyses were carried out for Tc-99 as an example.

The deterministic analyses were performed for a number of different modelling assumptions. Central case was defined as the usual solubility limited model, and following conceptual perturbations were also considered;

- (1) true colloid model,
- (2) pseudo colloid model,
- (3) co-existing true/pseudo colloid model,
- (4) evolving (spatially uniform) model,
- (5) evolving (spatially heterogeneous) model,

* Nuclear & Advanced Technology Division, JGC Corporation

Also stochastic sensitivity analyses were executed sampling the conceptual/parameter variations from the central case stochastically. As the result, it was clarified that this approach can improve the completeness of the formation of a chemical sink in the decompressed zone, and thus, in the significant increase of release rate due to the rapid dissolution of the precipitation formed in the perturbed period. This example indicated the possible improvement of the completeness by using the both methods iteratively.

Based upon the above results, now it is clear that dynamical system analysis method developed in this study can be applied to the variety of future conditions and a set of phenomena being expected for the evolution of the engineered barriers.

In the next phiscal year, it can be recommended to perform thorough analyses for all the H-3 nuclides and wide range of FEPs. To do this ,it is necessary to make a minor modification to the front end of SPADE so that the sensitivity analyses can be done in a more systematic and efficient way.

目次

はじめに	1
1. 動的解析の方法論	5
1.1 解析の目的及びその必要性	5
1.2 ニアフィールド現象の変遷に関する FEPs の抽出	7
1.3 FEPs の分類	12
1.4 広義拡散方程式の導入	15
1.4.1 広義の拡散性とその分布形	17
1.4.2 広義拡散方程式の数学的定式化	22
1.4.3 ニアフィールド核種移行問題への適用	
2. ソフトウェアの改良	42
2.1 汎用 DAE ソーバー SPADE	42
2.1.1 SPADE の基本機能	42
2.1.2 今年度の改良点	62
2.1.3 今後の課題	67
2.2 グラフィカルユーザーインターフェイス SAM GUI	68
2.2.1 SAMGUI の概要	68
2.2.2 今年度の改良点	70
3. 動的解析の試行 1 (決定論的感度解析)	73
3.1 概要	73
3.2 標準ケース	74
3.2.1 支配方程式	74
3.2.2 解析体系	75
3.2.3 境界条件	75
3.2.4 移行パラメータ	75
3.2.5 SPADE への入力	79

3.2.6	結果	81
3.3	擬似コロイドを考慮したモデル	87
3.3.1	支配方程式	87
3.3.2	移行パラメータ	87
3.3.3	SPADEへの入力	88
3.3.4	結果	88
3.4	真性コロイドを考慮したモデル	94
3.4.1	支配方程式	94
3.4.2	移行パラメータ	94
3.4.3	SPADEへの入力	95
3.4.4	結果	95
3.5	擬似コロイドと真性コロイドの共存を考慮したモデル	100
3.5.1	支配方程式	100
3.5.2	移行パラメータ	100
3.5.3	SPADEへの入力	100
3.5.4	結果	101
3.6	人工バリア周辺の水理条件の経時変化を考慮したモデル	105
3.6.1	変化のパターン	105
3.6.2	ケース設定	107
3.6.3	結果	107
3.7	人工バリア内の化学条件の時間・空間的变化	111
3.7.1	空間的に一様な変化	111
3.7.2	空間的に不均質な変化	117
4.	動的解析の試行2 (統計的感度解析)	124
4.1	概要	124
4.2	統計的感度解析の方法論	125
4.3	予備的解析	126
4.3.1	$\alpha(x, t)$ の統計的発生	126
4.3.2	解析結果	129

4.4 今後の課題.....	140
5.まとめ.....	150
付録-1 抽出されたFEPsの内容.....	153
付録-2 人工バリアが移流支配となる可能性について.....	166
付録-3 溶解度限界モデルについての広義拡散方程式の解の検証	170
付録-4 ビルトイン関数のリスト.....	171
付録-5 各ケースに対するSPFファイル.....	172

表 リ ス ト

表1.2 —1 ニアフィールド現象の変遷に関するFBPs

表3.2.4 —1 標準ケースでのパラメータ値

図 リ ス ト

- 図1.4.1 —1 複数の移行形態の分布形としての表現(概念図)
- 図1.4.1 —2 相転移としての化学挙動の表現(概念図)
- 図1.4.1 —3 速度論的反応の表現(概念図)
- 図1.4.3 —1 通常の溶解度限界モデルに対応した拡散係数の分布形
- 図1.4.3 —2 全濃度 C の関数としての α の表現(通常の溶解度限界モデル)
- 図1.4.3 —3 真性コロイドを考慮した場合の拡散係数の分布形
- 図1.4.3 —4 全濃度 C の関数としての α の表現(真性コロイドモデル)
- 図1.4.3 —5 疑似コロイドを考慮した場合の拡散係数の分布形
- 図1.4.3 —6 全濃度 C の関数としての α の表現(疑似コロイドモデル)
- 図1.4.3 —7 真性コロイド及び疑似コロイドを考慮した場合の拡散係数の分布形
- 図1.4.3 —8 全濃度 C の関数としての α の表現
(真性コロイド、疑似コロイド共存モデル)
- 図1.4.3 —9 全濃度 C の関数としての α の表現(非線型吸着モデル)
- 図1.4.3 —10 時間—空間的条件変化の影響の表現例
(放射線分解による酸化還元フロントの生成・移動)

- 図3.2.2 —1 解析体系
- 図3.2.6 —1 標準ケースにおける濃度プロファイルの例
- 図3.2.6 —2 標準ケースにおける核種放出率の経時変化
- 図3.2.6 —3 標準ケースについての感度解析の結果(溶解度)
- 図3.2.6 —4 標準ケースについての感度解析の結果(遅延係数)
- 図3.3.4 —1 疑似コロイドモデルによる濃度プロファイル算出例(R=2.8)
- 図3.3.4 —2 疑似コロイドモデルによる核種放出率算出例(R=2.8)
- 図3.3.4 —3 疑似コロイドモデルによる感度解析結果(R=2.8)
- 図3.3.4 —4 疑似コロイドモデルによる濃度プロファイル算出例(R=280)
- 図3.3.4 —5 疑似コロイドモデルによる核種放出率算出例(R=280)
- 図3.4.4 —1 真性コロイドモデルによる濃度プロファイル算出例
- 図3.4.4 —2 真性コロイドモデルによる核種放出率算出例
- 図3.4.4 —3 真性コロイドモデルによる感度解析結果

- 図3.5.4 —1 コロイド共存モデルによる濃度プロファイル算出例
- 図3.5.4 —2 コロイド共存モデルによる核種放出率算出例
- 図3.5.4 —3 コロイド共存モデルによる感度解析結果
- 図3.6.1 —1 水理条件の時間的变化パターン
- 図3.6.3 —1 経時的な水理条件変化モデルによる核種放出率算出例($q_{w1}=10^4$)
- 図3.6.3 —2 経時的な水理条件変化モデルによる核種放出率算出例($q_{w1}=10^5$)
- 図3.6.3 —3 経時的な水理条件変化モデルによる感度解析結果
- 図3.7.1 —1 化学条件変化による溶解度時間変化のパターン
- 図3.7.1 —2 経時的な化学条件変化モデルによる濃度プロファイル算出例($q_{w1}=10^4$)
- 図3.7.1 —3 経時的な化学条件変化モデルによる核種放出率算出例($q_{w1}=10^4$)
- 図3.7.1 —4 経時的な化学条件変化モデルによる濃度プロファイル算出例($q_{w1}=10^5$)
- 図3.7.1 —5 経時的な化学条件変化モデルによる核種放出率算出例($q_{w1}=10^5$)
- 図3.7.1 —6 経時的な化学条件変化モデルによる感度解析結果
- 図3.7.2 —1 放射線分解を考えた場合の腐食生成物層中溶解度変化の例
- 図3.7.2 —2 化学条件の時間・空間的变化を考慮したモデルによる濃度分布算出例
- 図3.7.2 —3 化学条件の時間・空間的变化を考慮したモデルによる核種放出率算出例
-
- 図4.3.1 —1 標準ケースでの広義拡散関数 $\alpha_0(x, t)$
- 図4.3.1 —2 統計的に作成した摂動 $\alpha \sim(x, t)$ の例
- 図4.3.1 —3 δ を種々に変えた場合の例(ベントナイト最外側)
- 図4.3.2 —1 統計的感度解析の結果
- 図4.3.2 —2 条件変動に対応した過渡的核種挙動の例(概念図)
- 図4.4 —1 決定論的手法と統計的手法とを相互保管的に用いた感度解析の概念
- 図4.4 —2 ゆるみ域における化学的シンク形成を考慮したモデルにより算出された濃度プロファイル
- 図4.4 —3 ゆるみ域内地下水流量が小さい(標準ケースと同じ)場合の核種放出率
- 図4.4 —4 ゆるみ域内地下水流量が小さい(標準ケースの 10^3 倍)場合の核種放出率

はじめに

高レベル放射性廃棄物地層処分の性能評価には、多様な不確実性が含まれる。この不確実性の種類を、性能評価の手順に則して分類すれば、以下の様なものがそれぞれ挙げられる。

- (1) 処分された廃棄物中の放射性物質が人間に影響を及ぼす過程、あるいはこの影響の程度を規定する環境条件の将来の返遷に関する F E P s (Features, Events and Processes)の選定に関する不確実性
- (2) 選定された個々の F E P s 及び F E P s 間の相関についての理解とその数学的定式化に関する不確実性
- (3) 定式化された数学的モデルを用いて評価を行う際の入力データの選定に関する不確実性

これらの不確実性が生じる原因は、上記のいずれに於いても（F E P s の選定、数学モデルの作成、及び入力データ値の設定）選択の可能性は複数存在しており、しかも、評価対象となる時間・空間スケールの大きさに起因する観察結果や経験の不足のために、いずれの選択肢をとることが適切かを考える際に主観的な判断を完全に排除することは極めて困難であり。上述した種々の不確実性の影響が評価結果に含まれることが不可避となる。

このような種々の不確実性の要因が性能評価に含まれざる得ないことは、性能評価研究を実施する各段階を通じて極めて重要な意味を持つこととなる。例えば、性能評価の実施に先立ち設計、シナリオ解析、モデル開発及びデータ収集を行う準備段階においては、これに引き続いて行われる評価に含まれる不確実性をできるだけ小さくすることが課題となり、このために、それぞれの不確実性の要因について感度解析を行うことによって、各要因による不確実性を減らすための課題（設計変更によって該当する不確実性の影響を低減する、保守的なモデルを用いることによって該当する不確実性要因が顕在化する

事を防ぐ、現象理解・データ収集を行うことによって本質的な不確実性そのものを低減する等)の重要分類を行うことが必要となる。性能評価研究が全体として研究領域にまたがるものであることを考えれば、この点は、プロジェクト全体の収束性を保つ上で極めて重要である。

また、実際に評価を行ない、地層処分の安全性についてのある(暫定的なものであったとしても)結論を導く際には、その判断の根拠としての評価結果に含まれる不確実性の種類と大きさを付記することが必須である。これは、統計的な評価手法を用いる場合には、勿論、明示的に為されることとなり、また(保守的な)決定論的評価においても、評価結果が被ばく線量の上限を与えることの確実(信頼性レベル)を示すことによって為されなくてはならない。

評価の各段階において必要となる不確実性の取扱いが最も有効に行われてきたのは、はじめに述べた三種類の不確実性要因のうち(3)のものについてである。これはパラメータの不確実性と呼ばれ、通常の性能評価モデルを用い、入力データ値をある統計的分布を持つ母集団からサンプリングすることによって行われる感度解析/不確実解析によって定量化することが可能である。この結果、例えば、人工バリア中の核種挙動についてパラメータの不確実性の影響を低減するための課題として最も重要なものは溶解度を算出するための熱力学データベースの確立であり、インベントリ制限となる $Se-79$ や娘核種の沈殿成長という二次的問題を有する $Am-241$ 、 $Am-243$ といった例外を除けば分配係数取得の重要性はあまり高くないといった重要度分類が得られている。然しながら、この結果はあくまで(1)及び(2)の不確実性を無視したものであり、感度解析に用いるモデル及びその前提条件が変われば結果も当然変わる可能性がある。実際、最近の研究で放射性元素の濃度が溶解度に達すると瞬時に沈殿が形成され過飽和状態は決して生じないとする従来のモデルに代わり、溶解・沈殿に反応速度を現実的にモデル化した例では、評価結果が溶解度に支配されるのは沈殿反応が数週間のタイムスケールで生じる速い反応である場合に限られ、反応がこれよりも遅い場合には溶解度は核種放出率を実効的に制限し得る要因とはならないことが示されている〔1〕。従って、理想的には、他の2つの不確実性要因についても、同様の定量的解析手法を用いる事が望ましい。しかし、現実には、これら二つの不確実性要因において、はじめに述べた選択肢

(FEPs、数学的定式化)を変更することは、即ちモデル自体の変更となってしまうために、これまでこれらの要因(シナリオの不確実性、及びモデルの不確実性とそれぞれ呼ばれる。)についての定量的解析手法は開発されていない。

本研究は、前出の三つの不確実性要因(シナリオ、モデル、パラメータ)の全てについて、定量的な感度解析と不確実性解析とを行う事を可能とする手法の開発を目的としたものである。

具体的には、人工バリア内の物質移動が拡散支配であることのみを前提として、人工バリア内外の化学条件や周囲の水理条件が時間・空間的に可変であり、多相の化学種(陽イオンあるいは陰イオンの溶存形、吸着種あるいは擬似コロイド、沈殿あるいは真性コロイド)間の化学反応による相転移が生じつつ、各自が移動するという最も広い意味での核種移行問題を考える。後述する様に、この問題は、

- ・外部事象(気候変化、海面変動、隆起、浸食、火成活動等)による化学条件の時間・空間的变化

という、これまでの一定環境条件とは異なる代替的なシナリオを含み、また、人工バリア内部事象として、

- ・コロイド生成・移動、放射線分解によるレドックスフロントの移動等

の新たなFEPsを含むと共に、

- ・速度論的化学反应
- ・非線型吸着

といったモデル化のオプションも包含し得る極めて一般性の高いものである。本研究では、これらのFEPsの選定や環境条件の仮定、あるいはモデル化における選択という事項に起因する不確実性を定量的に解析するために、上述した幅広い問題が共通して

持っている特性に着目して広義の拡散方程式を導出する。この広義の拡散方程式は、広義拡散関数と呼ばれるパラメータ関数を内包しており、このパラメータ関数を考慮すべき F E P s に対応して設定することによって単一のモデルで複数の F E P s についての問題を解析することが可能となる。

今年度の研究は、上述した各種の不確実性を定量的に解析するための手法の開発と、これを可能とするソフトウェアの改良、並びにこれらを用いた解析の試行を行ったものである。以下、本報告書ではその成果をとりまとめるものとする。なお、構成は、手法の開発（1章）、ソフトウェアの改良（2章）、解析の試行（3章及び4章）の順であり、最後に結果と今後の課題とをまとめた（5章）。

1. 動的解析の方法論

1.1 解析の目的及びその必要性

従来の性能評価においては、主としてモデルの単純化を図るために、以下の項目を先験的に仮定した人工バリア中核移行モデルが用いられてきた〔2〕、〔3〕、〔4〕。

- ・人工バリア内の物質移動は拡散支配である
- ・全ての化学反応（溶解、沈殿、吸着）は瞬間のうちに平衡に到ること
- ・吸着は線型のもの（Henry 型）であること
- ・コロイド等の溶存種以外の可動な物質が存在しないこと
- ・人工バリア内の化学条件及び周辺の化学、水理条件は空間的に一様であること
- ・人工バリア内の化学条件及び周辺の化学、水理条件は時間的に一定であること

これらの仮定は、いずれも十分に検証された後に用いられているわけではなく、むしろ多分に公理主義的な性格の強いものである。従来の性能評価は、これ等のはじめに定めた「約束事」を前提としたモデルを用いているという意味から、stylize された評価と呼ばれることもある〔5〕。

実際、上記の6つの仮定を置くことによって、人工バリア中の核種移行問題の数学的取扱いは極めて簡単なものとなる。まず第一に、一定の（時間的、空間的に）環境条件を仮定することによって溶解度や遅延係数といった一般的には時間及び関数となるべきパラメータが定数となり、次にコロイド等が存在することにより評価対象となる核種の移動形態は沈殿、吸着種及び溶存種に限定されることとなる。そして更に、化学反応の瞬時平衡性と吸着の線型性とを仮定することによって、本来は個々に独立した方程式によって記述されるべき各移行形態間の関係が簡単な代数的規則によって表現されるため、結果として一種類の方程式のみを考えれば良いこととなる。

この様に、上記の仮定を置くことは、モデルを単純化し計算効率を高める反面、そ

の代償として適用可能な現象を限定してしまうという短所も有している。実際に、この制約が、「はじめに」で述べた様に、パラメータに起因するもの以外の不確実性が定量的には扱えないものとなっている最大の要因である。

既に述べた様に、本研究の目的は、上記の制約をとり除くことによって、前出の5つの仮定と必ずしも一致しないより一般的な前提条件の下で、より広い範囲のモデル化のオプションを考えた解析を可能とすることである。

このような解析の必要性は、以下のことを考えれば明らかである。

現在、我が国の性能評価研究は、平成3年度の研究開発成果報告書〔2〕の結果を踏まえ、より進んだ段階に対応した第二次取りまとめを行うために、種々の研究開発を展開するという位置付けにある。ここで、注意すべき点は、平成3年度報告書が人工バリアを中心として我が国における地層処分成分性の見通しを示したと同時に、多くの課題を明らかにしたということである。これらの課題の中には、従来の評価モデルにおいて入力データとして含まれる溶解度や分配係数等についてのデータ収集と信頼性の高いデータセットの作成といったものに加えて、気候変動や火成活動あるいは断層形成といった代替的な現象についてのモデル開発、さらに非線型吸着モデルや速度論的反応モデルを採用した場合の評価結果の変動程度の明確化といった多様な事項が含まれる。そして、既に述べた様に、従来の評価モデルでは、入力データ値の感度解析及び不確実解析しか実施できないために、変動シナリオの検討、代替的現象のモデル化、代替的モデルを用いることによる結果の変動の調査といった、他の課題については、全体的視野からの重要度分類を行うことが難しい。このことは、まさに本研究で目的としている新たな解析手法の必要性を示すものと言うことができる。

次節では、上述した従来の評価モデルの制約を取り除くために、具体的にどのようなFEPsを考えるべきかを明らかにすることにより、本研究で開発する解析手法の対象とする範囲を定めることとする。

1.2 ニアフィールド現象に関するF E P s の抽出

本節では、従来の性能評価モデルで用いられた限定的な仮定に代り、より一般的な問題に適用可能な解析手法を開発するために、その対象とすべきF E P s を具体化することとする。このための検討の手順として、1.1 に述べた6つの仮定とは相反する様な状況を想定し、次にそれぞれの状況をもたらし得る様なF E P s あるいは結果として問題となる様なF E P s を抽出する。

1.1 に述べた6つの仮定それぞれについて、これと相反する様な状況を含めたより一般的な記述は次のようなものとなる。

- ・人工バリア内の物質移動は、拡散支配あるいは移流分散支配である。
- ・化学反応は十分に速く従って瞬時平衡は仮定できるか、あるいは遅く、速度論的効果が問題となる。
- ・吸着は線型あるいは非線型である。
- ・コロイド等の溶存種以外の可動な物質が存在する可能性がある。
- ・人工バリア内及び周辺の化学、水理条件は空間的に不均質である可能性がある。
- ・人工バリア内及び周辺の化学、水理条件は時間的に可変である。

次に、上記のそれぞれについて、対応するF E P s を抽出した。結果を表1.2-1 に示す。

なお、F E P s の抽出にあたっては、OECD/NEAによって作成されたF E P s リスト〔5〕、及びSKIによって作成されたF E P s リスト〔6〕を参考とした。各F E P s の内容については、付録-1を参照のこと。

表1.2-1 ニアフィールド現象の変遷に関するF E P S(1)

A. 人工バリア内の物質移動機構に関するもの

(拡散支配に結びつくもの)

- ベントナイト冠水・膨潤による止水効果

- 地下深部の動水勾配が小さいものであること

(移流・分散支配に結びつくもの)

- ベントナイトのイライト化

- 周辺母岩の亀裂中へのベントナイトの侵入による緩衝材の密度低下

- 亀裂中での地下水流によるベントナイトの浸食による緩衝材密度低下

- 施工時のベントナイトブロック間の空隙を通じた選択的流路の形成

B. 化学反応速度に関するもの

- 放射性元素濃度の過飽和

- 遅い吸着による実効的遅延効果の低下

C. 吸着の非線型性に関するもの

- 吸着サイトの飽和

- 競合するイオンの影響

D. 多相の移動物質に関するもの

- 沈殿フロントの移動

- 擬似コロイドの生成

- 真性コロイドの生成

- コロイドの移動、ろ過及びベントナイト空隙の閉塞

- 陰イオン排斥

- 表面拡散

- ガラス固化体の一様 (congruent) / 非一様 (incongruent) 溶解

表1.2-1 ニアフィールド現象の変遷に関するF E P S(2)

E. 空間的に不均質な環境条件に関するもの

- 放射線分解による酸化還元フロントの形成
- 酸化性地下水の人工バリア周辺への流入によるE h 勾配
 - ・深層風化によるもの
 - ・新たな断層形成によるもの
 - ・隆起、浸食による処分場のdenudationによるもの
- 人工バリアと地下水との反応によるpH勾配の形成
 - ・コンクリート反応によるもの
 - ・ベントナイト反応によるもの
 - ・ガラス溶解によるもの
 - ・オーバーパックの腐食によるもの
- 酸性地下水の人工バリア周辺への流入によるpH勾配の形成
 - ・微生物によって触媒された黄鉄鉱の酸化(acid mine drainage)によるもの
 - ・火成活動によるもの
- 人工バリアと地下水との反応によるイオン強度勾配の形成
 - ・コンクリート反応によるもの
 - ・ベントナイト反応によるもの
 - ・ガラス溶解によるもの
 - ・オーバーパック腐食によるもの
- 高イオン強度水の人工バリア周辺への流入によるイオン強度勾配の形成
 - ・acid mine drainageによるもの
 - ・火成活動によるもの
 - ・海面変動によるもの
- シリカと鉄の共沈による「ポンプ効果」
- 人工バリア材間の吸着特性の相異

表1.2-1 ニアフィールド現象の変遷に関するF E P S(3)

F. 環境条件の時間変化に関するもの

- 放射線分解によるEh上昇
- 酸化性地下水の人工バリア周辺への流入によるEh上昇
 - ・深層風化によるもの
 - ・新たな断層形成によるもの
 - ・隆起、浸食による処分場のdenudationによるもの
- 人工バリアと地下水との反応によるpHの上昇
 - ・コンクリート反応によるもの
 - ・ベントナイト反応によるもの
 - ・ガラス溶解によるもの
 - ・オーバーパック腐食によるもの
- 酸性地下水の人工バリア周辺への流入によるpH低下
 - ・acid mine drainageによるもの
 - ・火成活動によるもの
- 人工バリアと地下水との反応によるpHの上昇
 - ・コンクリート反応によるもの
 - ・ベントナイト反応によるもの
 - ・ガラス溶解によるもの
 - ・オーバーパック腐食によるもの
- 高イオン強度水の人工バリア周辺への流入によるイオン強度の上昇
 - ・acid mine drainageによるもの
 - ・火成活動によるもの
 - ・海面変動によるもの

表1.2-1 ニアフィールド現象の変遷に関するFEP S(4)

F. 環境条件の時間変化に関するもの(続き)

-人工バリア周辺の水理条件の経時変化

- ・気候変化によるもの
- ・地形変化によるもの
- ・風化作用によるもの
- ・隆起、浸食によるもの
- ・沈降、堆積によるもの
- ・海面変動によるもの
- ・地震によるもの
- ・断層形成によるもの
- ・火成活動によるもの
- ・人工バリアからの熱、及び化学種の供給による二次鉱物の溶解、沈殿

1.3 FEPsの分類

表1.2-1に示した種々のFEPsによりもたらされる人工バリア内及びその周辺の状況のうち、数学的定式化を考える上で最も基本的な相異は、人工バリア内が拡散支配となる場合と移流分散支配となる場合の間の区分である。実際、これらをモデル化するには、それぞれ異なる基本方程式を用いなければならない。

人工バリア内の地下水流速は、高圧縮ベントナイトが極めて高い止水性を有することにより、かなり保守的な動水勾配の流速場の中に人工バリアが置かれた場合でも事実上静水状態(pe 数 $\ll 1$)となることが示されている〔2〕,〔3〕。

他方、表1.2-1Aに示されている移流・分散支配となる状況をもたらす得るFEPsについて文献調査と簡単なモデルを用いた試算を行った結果、次の様なことが明らかとなった(検討内容の詳細については、付録-2を参照されたい。)

(1) ベントナイトのイライト化〔2〕〔3〕

ベントナイトのイライト化は、ベントナイトの主成分であるNa型モンモリロナイトが地下水中のカリウムと反応する事により別種の粘土鉱物であるイライトに変換されることであるが、この反応は通常の温度領域では極めて遅く、人工バリア設計に於いて坑道離間距離や廃棄物の定置間隔を十分にとり緩衝材中の最高温度を100℃以下に保つ事により回避することが可能と考えられる。

(2) 周辺母岩の亀裂へのベントナイトの浸入による緩衝材の密度低下〔6〕

水分飽和した高圧縮ベントナイトは高い膨潤圧を有するために、人工バリア周辺の母岩の空隙特に亀裂中に侵入することが予想される。この場合には、緩衝材の密度が低下し、これに従って透水性が上昇する可能性がある。然しながら、この現象についての実験結果に基づく経験的モデルを用いた試算の結果、通常想定される亀裂開口幅では、緩衝材の密度低下は微小であり、移流分散支配につながる様な透水性の上昇の可能性は小さいことが明らかとなった。この現象によって移流支配となる状況がもたらされるのは、処分場の地表での露出や処分場を横切る断層形成といった現象が生じた場合に限定されるものと考えられる。

(3) 亀裂中での地下水流によるベントナイトの浸食による緩衝材密度低下〔7〕

亀裂内に侵入したベントナイトの先端部分は、ゲル化することにより地下水流による浸食を生じる可能性がある。この結果として、(2)と同様に緩衝材の密度低下と、これに伴う透水性の上昇が予想される。しかし、この現象を想定した実験の結果として、地下水流速が 10^{-4} m/sec という極めて大きいものにならない限りは浸食が生じないことが示されており、(2)と同じく通常の条件下ではこの可能性は無視し得るものと考えられる。

(4) 施工時のベントナイトブロック間の空隙を通じた選択的流路の形成〔8〕

水分飽和したベントナイトは高い可塑性を有することから、施工時のブロック間の空隙は冠水に伴って自然に閉鎖されると考えられる。また、小規模の実験ながら、この事を裏付ける様なデータも得られている。緩衝材施工方法の詳細が未確定の現状ではこの可能性を完全に否定することは難しいが、本研究では、今後の技術開発によりこの可能性は十分小さくすることができるものと仮定する。

以上の事から、適切に設計・施工される人工バリアシステムに於いて、通常の条件下で移流・分散支配の状況が出現する可能性は小さいとすることができる。また、例外的にこの様な状況を生じ得る処分場の露出や断層発生は、この他にも多くの破局的な効果をもたらし得ることから、むしろ独立シナリオとしてその発生可能性に重点を置いたシナリオ解析を行うことが適当と考えられる。そこで、本研究で対象とするFEPsは、移流・分散支配につながるFEPsを除いた他の全てのものと設定する。

次節では、以上の検討結果を踏まえて、対象とする全てのFEPsに適用可能な広義の拡散方程式を導出すると共に、この新たなモデルを一般の人工バリア中核種移行問題に適用するための方法についてまとめる。

1.4 広義拡散方程式の導出

本研究で開発する解析手法の用途は、既に述べた様に、性能評価に介在するシナリオ、評価モデル及びパラメータの不確実性の影響を定量化するためのものであり、この手法を用いた解析の目的は1.3に示した諸F E P sがどの様に生じそして進展するかという事の詳細を明らかにすることではなく、各F E P sがもし生じたとすると人工バリア中の核種挙動はどの様にそしてどの程度影響されるかを概括的かつ網羅的に調べることにある。そこで、本研究における各F E P sの取り扱いとしては、その原因ではなく結果に焦点を当てたものとする必要がある。例えば、ベントナイトへの核種吸着は、現象の機構としては表面錯体生成、共沈、デッドエンドポアへの取り込み、物理吸着という様々なものが想定し得るが、結果として問題となるのはその速度論的特性、線型性（あるいは非線型性）、及び吸着の強さという点のみであり、本研究では、核種移行の観点から重要となるこれらの特性のみをモデル化することとする。また、上記の現象機構の分類は、吸着の特性のうちのあるものが重要であるという感度解析の結果が得られた場合に、これを今度の課題として反映する際の現象論的解釈として用いられることとなる。（例えば、非線型性という特性が重要であることが明らかとなった場合に、表面錯体生成という現象理解の立場から錯体生成可能なサイト（S-O-H）の密度とベントナイトの吸着表面積を調べる研究を課題として設定すること等。）

以上の観点から、本研究で作成するモデルをShadow Model〔9〕と呼ぶこともできる。これは種々のF E P s自体をモデル化するのではなく、その結果として核種移行について生じる影響を「影」として写し取るものであることに対応している。

本節では、1.2及び1.3に述べた諸F E P sの核種移行への影響を「影」として写し取ることのできる広義拡散方程式、あるいはShadow Diffusion Modelを導出し、一般的な核種移行問題への適用方法を検討した結果をまとめる。

1.4.1 広義の拡散性とその分布形

1.3に述べた様に、本研究で開発するShadow Modelは、人工バリア内が拡散支配であることを前提として、表1.2-1に示した他の全てのFEPsの核種移行への影響を定量的に表現可能なものであることが要件となる。これらのFEPsの影響を包括的に表現し得るためには、このShadow Modelは以下の様な一般性を有したものでなくてはならない。

- (1) 瞬時平衡としての化学反応及び速度論的効果の両者が表現可能であること
- (2) 線型及び非線型の吸着が表現可能であること
- (3) コロイド等、溶存種以外の可動な物質の生成・移動を表現可能であること
- (4) 化学・水理条件の空間的不物質性の影響を表現可能であること
- (5) 化学・水理条件の経時変化の影響を表現可能であること

ここでは、数学的定式化に先立ち、(1)~(5)を満足する様なモデル化の概念について以下に説明する。

モデル化にあたって、まず上記(1)~(5)のうち(3)に注目する。つまり、人工バリア内で、放射性元素は比較的拡散性の高い溶存種、全く拡散しない沈殿あるいはベントナイトへの吸着種、及び0ではない拡散係数をもつ真性コロイドや擬似コロイドといった複数の形態で存在するものとする。この状況は、図1.4.1-1に示す様に拡散係数を横軸とした確率分布（濃度分布）として最も自然に表現される。つまり、従来の評価モデルに於いて単一の移行形態の濃度を未知数として取り扱っていたことに代わり、複数の移行形態についての拡散係数の分布を考えることによって、(3)を可能とする。

次に、(1)及び(2)に対応する溶解・沈殿、吸着、コロイド生成といった化学的挙動を考える。これらの挙動は、各移行形態の間での相転移として表現することが可能である。例えば、溶解・沈殿といった挙動は溶存種のピークと沈殿のピークとの間での相転移であり、吸着は溶存種のピークと吸着種との間の相転移、コロイド生成

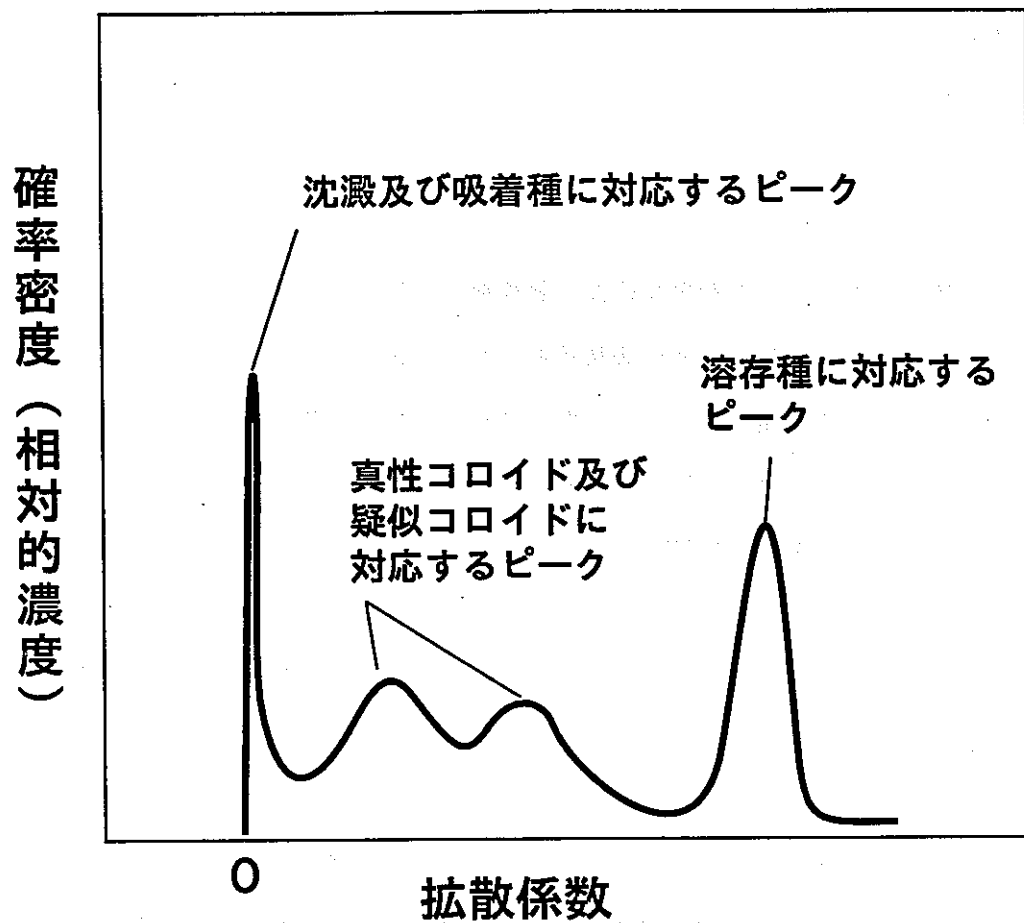


図 1.4.1-1 複数の移行形態の分布形としての表現 (概念図)

は溶存種とコロイドのピークの間での相転移である(図1.4.1-2)。

そして、これらの化学反応について瞬時平衡が成立する場合には、各ピークの高さについて簡単な代数的関係が成立するものと仮定することができる。また、特に吸着については、これが線型である場合には代数的関係式は最も簡単な比例式(線型式)となり、非線型の場合にはラングミュア型のisothermの様に非線型な代数式を考えることが必要となる。また、更に、速度論的な効果を取り入れることは、分布形が上述した瞬時平衡の理想的な形かられることを許容し、常に分布形がこのずれを解消して平衡状態に戻ろうとする力を受けるものと考えることによって為される。つまり、この力が強い程反応は速く、弱い程遅いものとして表現されることとなる(図1.4.1-3)。

最後に(4)、及び(5)については、分布形の挙動を表現するパラメータ(瞬時平衡の場合の代数的関係式中の定数、及び速度論的反應の場合の復元力の強さ)を、時間・空間的に一定ではなく、位置及び時間の関数として表現することによって要件が満たされることとなる。

1.4.2 広義拡散方程式の数学的定式化

ここでは、1.4.1に述べた広義の拡散性とその分布形に基づく概念モデルに定式化する。定式化にあたっては、1.4.1に述べた溶存種、真性あるいは擬似コロイド吸着種及び沈殿という区分は勿論、より広範なFEPs(例えば表面拡散や陰イオン排斥等)に対応できる様、任意の分布形を対象として広義拡散方程式を導出する。

一般に分布形について、拡散係数Dを持つ移行形態としての放射性元素の濃度が、位置 x 、時刻 t において $c(x, t, D)$ であるとする。ここで、この特定の移行形態についての物質収支式は、次の様に表現される。

$$\theta \frac{dc}{dt} = \nabla \cdot (\theta D \nabla c) - \sigma(c) - \lambda \theta c \quad (1)$$

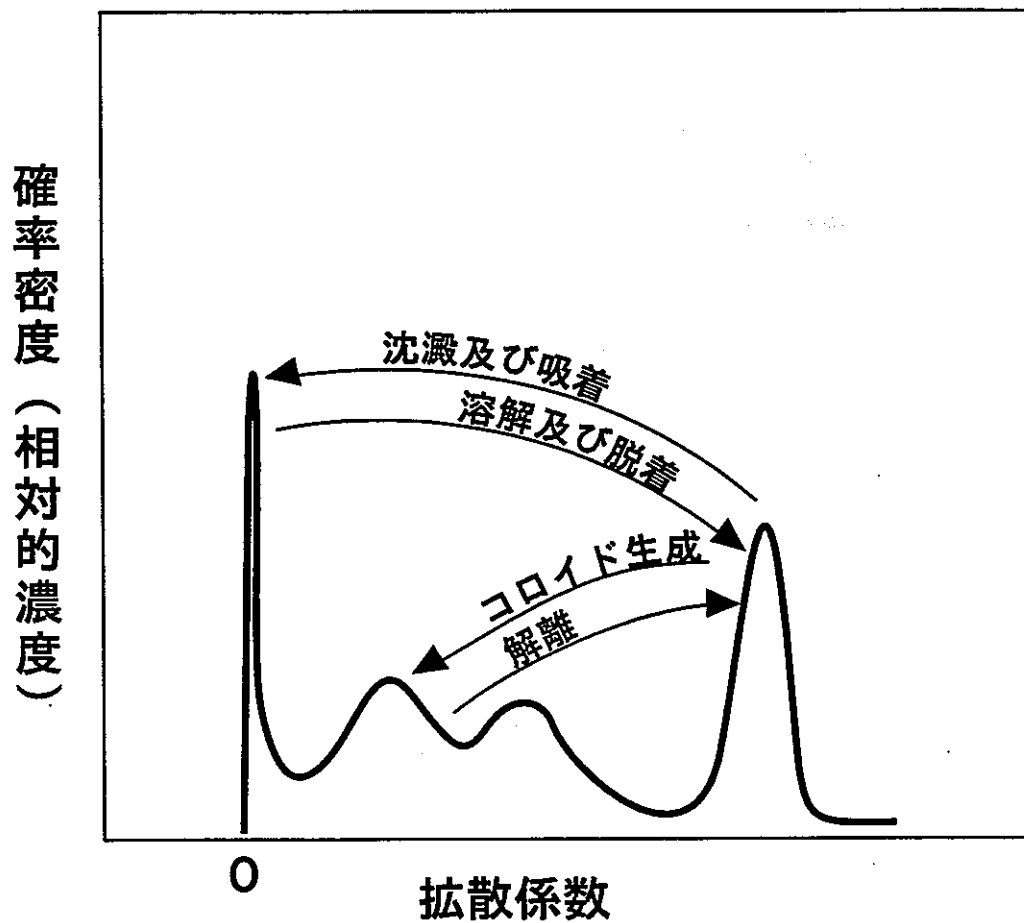
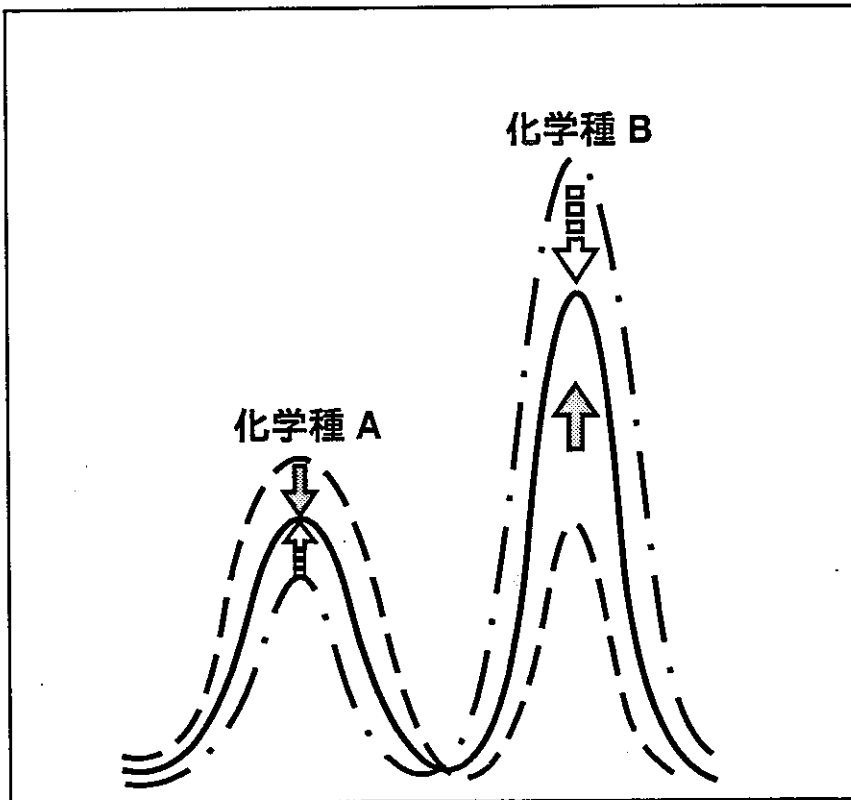


図 1.4.1-2 相転移としての化学挙動の表現 (概念図)








-  平衡状態での分布形
-  化学種 Aについて過飽和な状態での分布形
-  化学種 Bについて過飽和な状態での分布形
-  反応 $A \rightarrow B$ による分布形の変化
-  反応 $B \rightarrow A$ による分布形の変化

図 1.4.1-3 速度論的反應の表現 (概念図)

ここで、添字 t は時間に関する偏微分を、 ∇ は位置に関する偏微分を表し、また、 θ は媒体の空隙率を、 λ は核種の崩壊定数を表す。通常の拡散方程式と異なり、(1)式は $\sigma(c)$ という項を含んでおり、この項が 1.4.1 に述べた化学反応による相転移に伴う生成・消滅の効果を表現するものとなっている。

次に、以下の式によって、全ての移行形態を包含した核種の総濃度 $C(x, t)$ を定義する。

$$C(x, t) = \int c(x, t, D) dD \quad (2)$$

(1)式の両辺を D について積分し、更に(2)式を用いることによって、

$$C_t = \int \Delta D c dD - \lambda C \quad (3)$$

が得られる。ここで、(1)式中の $\sigma(c)$ 項は異なる移行形態の間での交換であり、従って全濃度の増減をもたらすものではないために、 D に関する積分によって消滅する。

(3)に於いて右辺の微分と積分の順序を交換することにより、この式は次の様に変形される。

$$C_t = \Delta \int D c dD - \lambda C \quad (4)$$

但し、 Δ はラプラシアンを表す。

また、位置 x 、時刻 t において、拡散係数 D を持つ移行形態が全濃度 $C(x, t)$ に占める割合を $\beta(D, C, x, t)$ ($0 \leq \beta \leq 1$) と定義することによって、

$$c(x, t, D) = \beta(D, C, x, t) C(x, t) \quad (5)$$

という関係式が得られる。 β の具体的な関数形は、モデル化しようとするFEPsに対応した化学反応式に依って定まるものであり、ここでは不定として取り扱う。

さらに、平均拡散係数 $\alpha(x, t)$ を、

$$\alpha = \int D \beta dD \quad (6)$$

として定義することにより、これを用いて(4)は次の様に書き換えられる。

$$C(x, t) = \Delta[\alpha C] - \lambda C \quad (7)$$

(7)式が、全濃度 $C(x, t)$ についての広義拡散方程式である。この式と通常の拡散方程式との相異は、この式に x, t についての不定の関数である平均拡散係数 $\alpha(x, t)$ が含まれることである(以下、この $\alpha(x, t)$ を広義拡散関数と呼ぶ)。この不定の関数が存在するために、(7)式はこのままでは解くことができない。広義拡散方程式を用いた手法は、この不定状態即ち剰余の自由度を利用して、様々なFEPsの影響をモデル化するものである。つまり、(7)式のモデルは常に不変のまま、 $\alpha(x, t)$ の関数形を各FEPsに対して設定することによって広範なシナリオ、モデル化オプションを表現する。

1.4.3 ニアフィールド核種移行問題への適用

1.4.2に述べた様に、広義拡散方程式(7)は、そこに含まれる不定の広義拡散関数 $\alpha(x, t)$ の具体的な形を定めることによって、個々の問題に対応した物理的モデルとして完成される。ここでは、いくつかの例を用いて、 $\alpha(x, t)$ の設定方法を説明する。

(1) 従来の評価モデルに応じた $\alpha(x, t)$ の設定

一定環境条件の下で、瞬時平衡の溶解制限、線型瞬時吸着を仮定し、コロイドが存在しないことを仮定した従来の評価モデルでは、拡散性についての分布は、図1.4.3-1に示す様に、拡散係数=0の二つのピーク（沈殿と吸着種）と、拡散係数 $D_{s,1}$ を持つピーク（溶存種）の三群から成る離散的な分布として表現される。ここで、 $\alpha(x, t)$ は平均の拡散係数であることから次の様に設定することができる。

① 液相濃度が溶解度以下である場合 ($C < c^*R$ の場合)

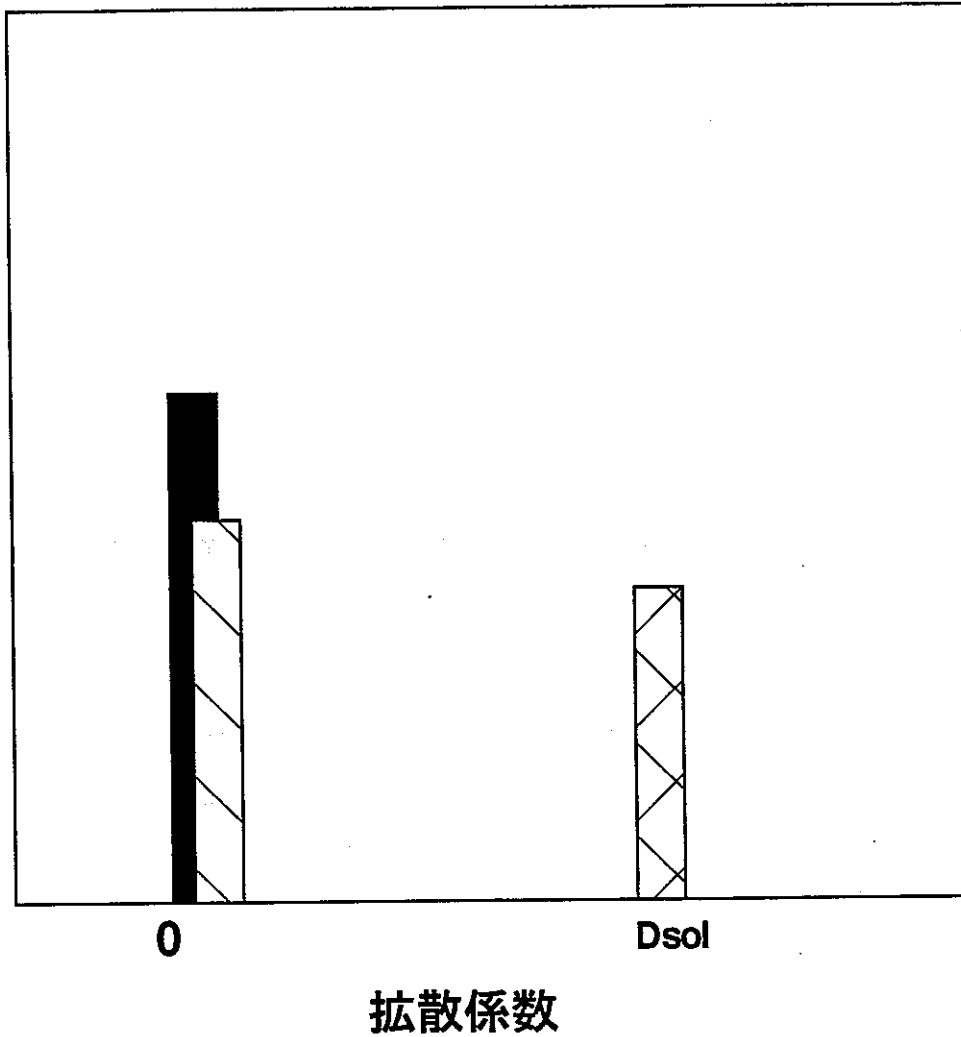
この場合は、沈殿は存在せず、全濃度 C は溶存種と吸着種との間に $1 : R - 1$ に配分されることから、それぞれの濃度は C/R 及び $(1 - R)C/R$ となる（但し、 R : 遅延係数）。そこで、

$$\alpha = \frac{\int D c dD}{\int c dD} = \frac{D_{s,1} C/R}{C} = D_{s,1}/R \quad (8)$$

が導かれる。

② 液相濃度が溶解度以上の場合 ($C > c^*R$ の場合)

確率密度
(相対濃度)



- 沈澱物
- ▨ 吸着種
- ▩ 溶存種

図 1.4.3 - 1 通常の溶解度限界モデルに対応した
拡散係数の分布形

この場合には、溶解度を超過したものが沈殿するため、沈殿物の濃度は $C - c^*R$ となる。そして、溶存種と吸着種との間で残りの c^*R が $1 : R - 1$ に配分されることから、これらの濃度はそれぞれ c^* 及び $(R - 1)c^*$ となる。

この事から、①と同様に、

$$\alpha = \frac{\int D c \, dD}{\int c \, dD} = \frac{D_{s,1} c^*}{C} \quad (9)$$

が導かれる。この場合の α を、全濃度 C の関数として図 1.4.3 - 2 に模式的に示す。

こうして(7)式の広義拡散方程式と、代数的関係式(8)、(9)式と連立することによって、従来の評価モデルが再現されることとなる。実際、この手法による解と、従来の評価モデルの解（解析解）とは一致することが検証された（付録 - 3 参照）。

(2) 真性コロイドの生成・移行を考えた場合の $\alpha(x, t)$ の設定

この場合には、拡散係数の分布は、図 1.4.3 - 3 に示す様に拡散係数 = 0 の二つのピーク（沈殿と吸着種）と拡散係数 $D_{s,1}$ のピーク、及び拡散係数 D_{1c} （真正コロイド）という四群から成る離散的分布となる。更にここで簡単なため、沈殿のうちある一定の割合 K_{1c} が真性コロイドとなることを仮定すると、 $\alpha(x, t)$ は次の様に求められる。（勿論、他のモデル化オプションがあればこれを採用することも可能である。）

① 液相濃度が溶解度以下の場合

上記の仮定から、この場合には真性コロイドは存在せず、 α は(1)の①と同一

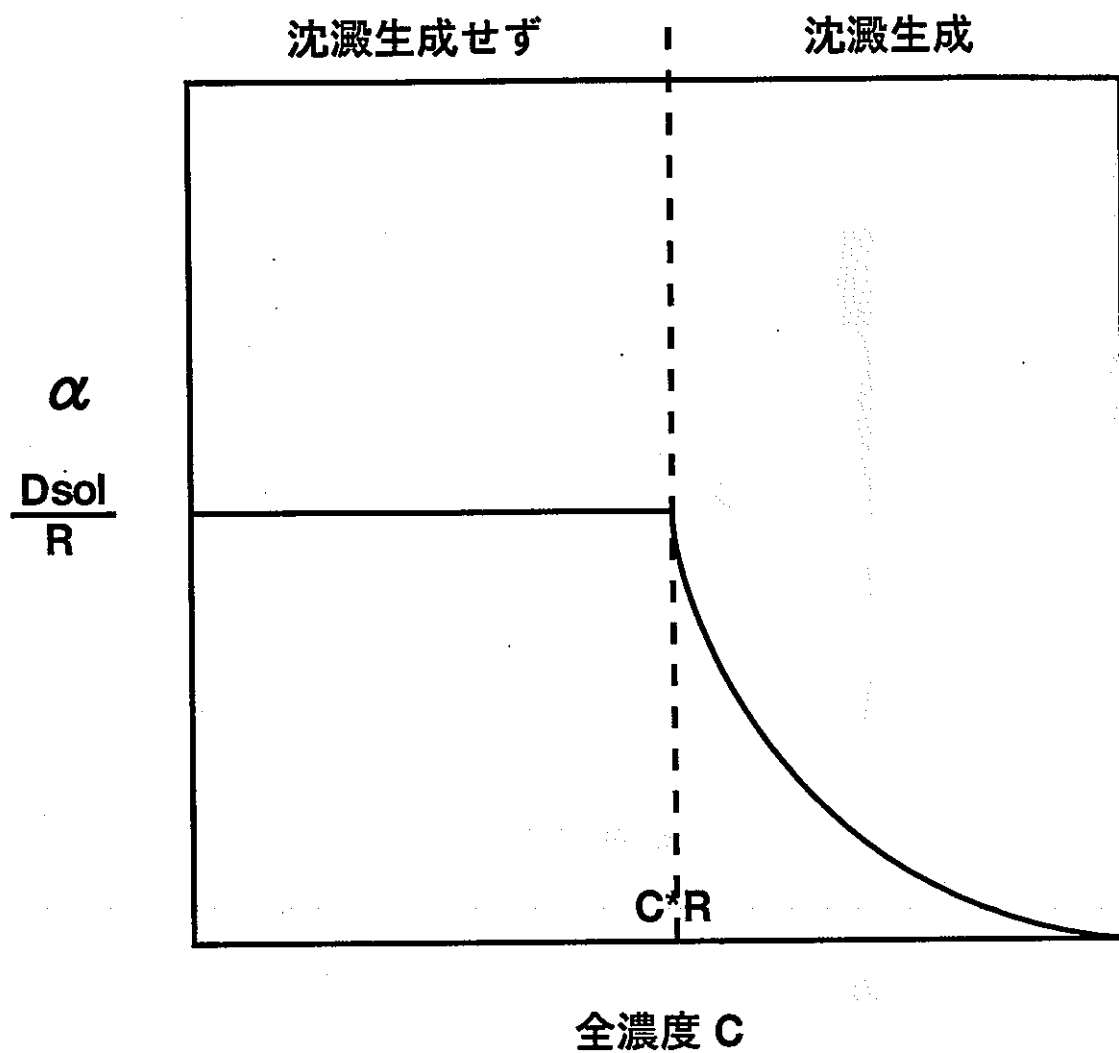
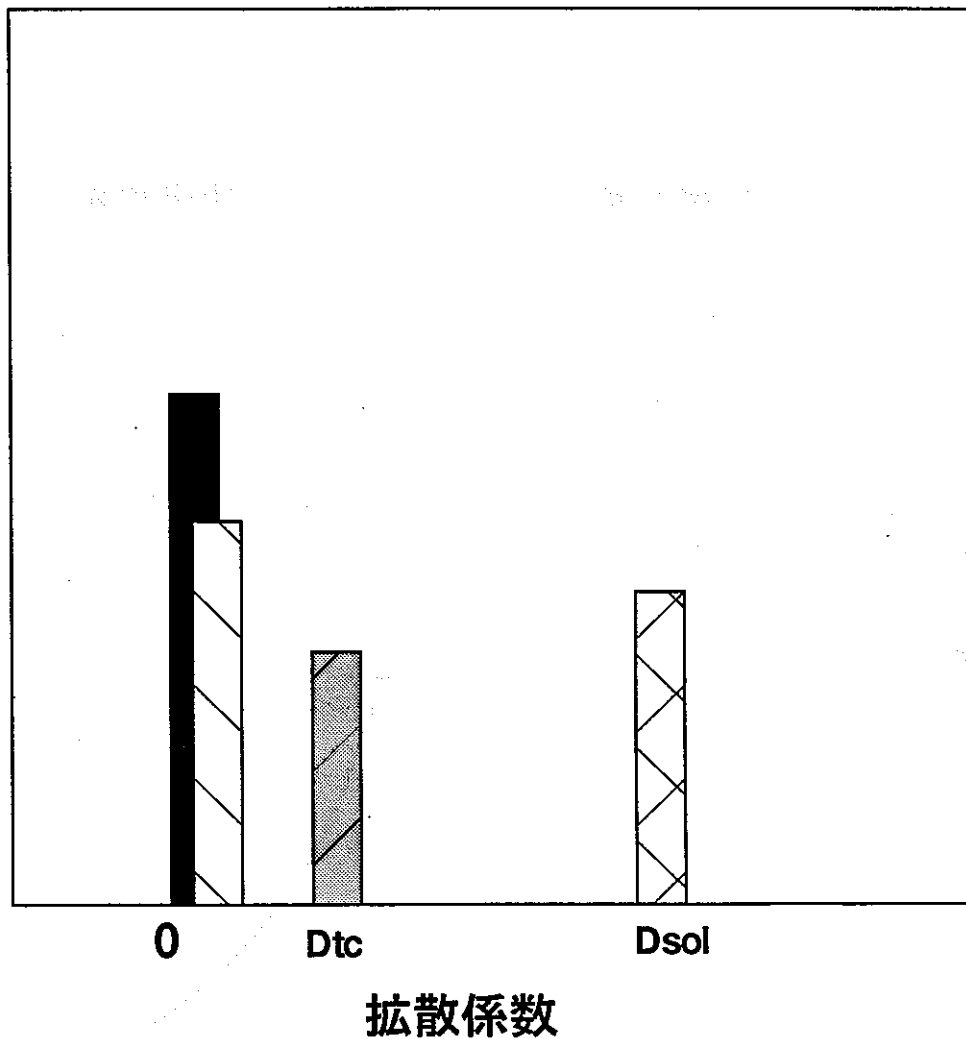


図 1.4.3 - 2 全濃度Cの関数としての α の表現
(通常の溶解度限界モデル)

確率密度 (相対濃度)



- 沈澱物
- ▨ 吸着種
- ▩ 溶存種
- ▧ 真性コロイド

図 1.4.3 - 3 真性コロイドを考慮した場合の拡散係数の分布形

となる。

$$\alpha = \frac{D_{s,01}}{R} \quad (10)$$

② 液相濃度が溶解度以上の場合

この場合には、全濃度は、次の様に配分されることとなる。

溶存種濃度 = c^*

吸着種濃度 = $c^* (R - 1)$

真性コロイド濃度 = $K_{1c} (C - c^* R)$

沈殿物濃度 = $(1 - K_{1c}) (C - c^* R)$

従って、 $\alpha(x, t)$ は、

$$\alpha = \frac{D_{s,01} c^* + K_{1c} D_{1c} (C - c^* R)}{C} \quad (11)$$

となる。(1)と同様、 C の関数としての α を模式的に図1.4.3-4に示す。

(3) 擬似コロイドの生成・移行を考えた場合の $\alpha(x, t)$ の設定

この場合の拡散係数の分布は、図1.4.3-5に示す様に、拡散係数=0の二つのピーク（沈殿及び吸着種）、拡散係数 $D_{s,01}$ のピーク（溶存種）、及び拡散係数 $D_{p,c}$ のピーク（擬似コロイド）の四群の離散的分布となる。ここで、(2)と同様に吸着種のうちのある一定の割合（ $K_{p,c}$ ）のものが擬似コロイドになると仮定すると、 α は次の様に仮定する。

① 液相濃度が溶解度以下の場合

全濃度は、次の様に配分される。

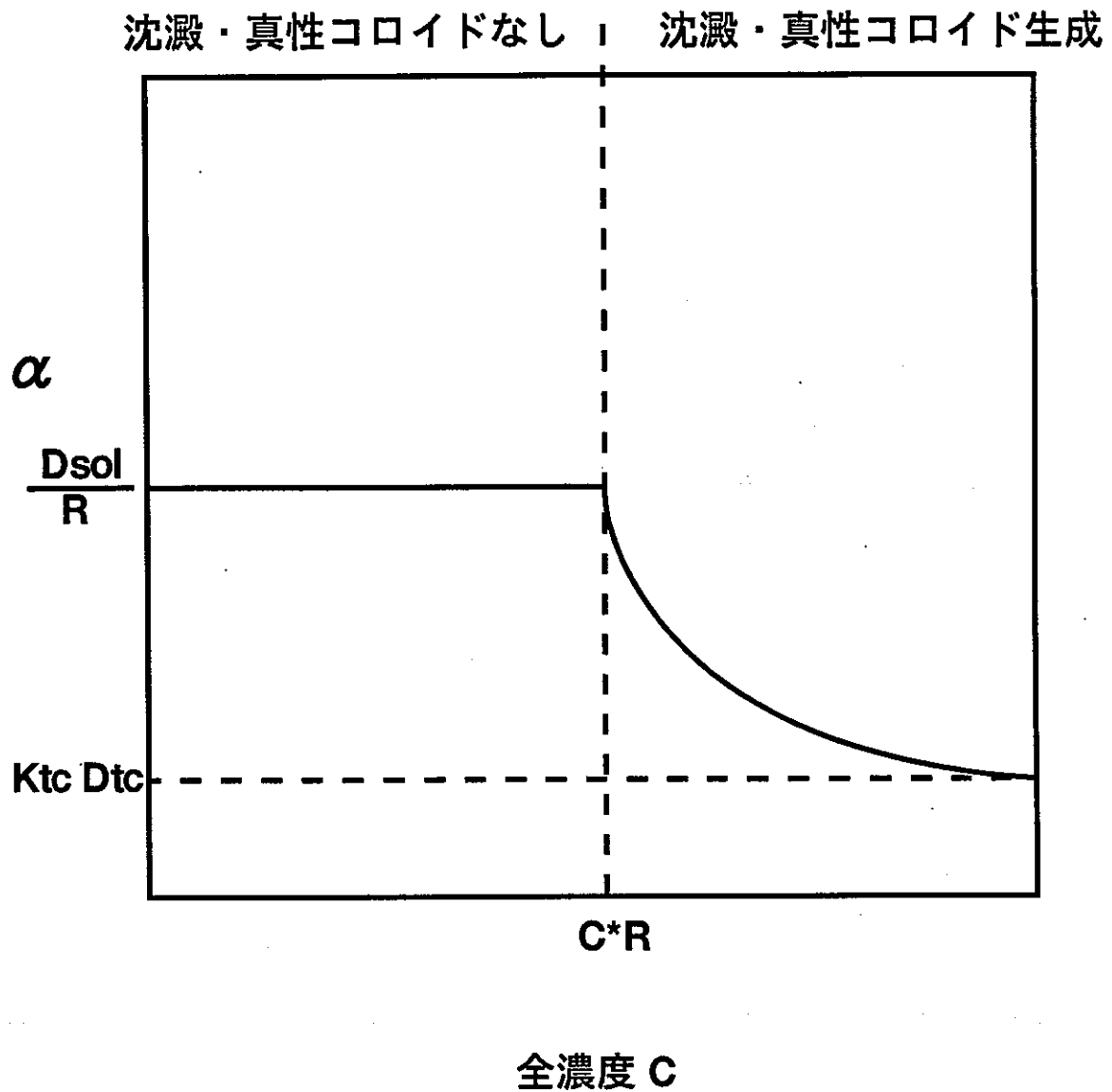
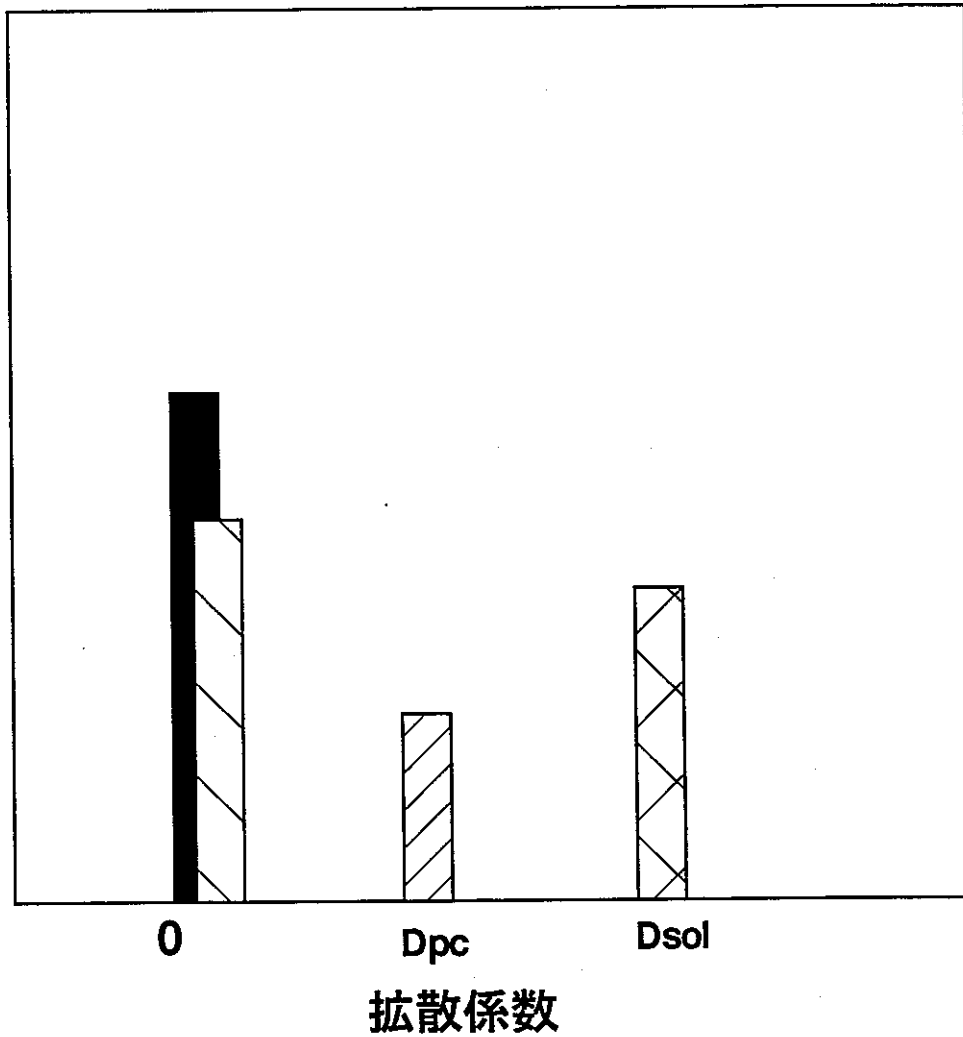


図 1.4.3 - 4 全濃度 C の関数としての α の表現
(真性コロイドモデル)

確率密度
(相対濃度)



- 沈澱物
- ▨ 吸着種
- ⊠ 溶存種
- ▧ 疑似コロイド

図 1.4.3 - 5 疑似コロイドを考慮した場合の拡散係数の分布形

溶存種濃度 = C/R

吸着種濃度 = $(1 - K_{pc}) (R - 1) C/R$

擬似コロイド濃度 = $K_{pc} (R - 1) C/R$

従って、 α は次の様に設定される。

$$\alpha = \frac{D_{s,01} + K_{pc} (R - 1) D_{pc}}{R} \quad (12)$$

② 液相濃度が溶解度以上の場合

全濃度は、次の様に配分される。

溶存種濃度 = c^*

吸着種濃度 = $(1 - K_{pc}) (R - 1) c^*$

擬似コロイド濃度 = $K_{pc} (R - 1) c^*$

沈殿物濃度 = $C - c^* R$

従って、 α は次の様に設定される。

$$\alpha = \frac{D_{s,01} c^* + K_{pc} D_{pc} (R - 1) c^*}{C} \quad (13)$$

なお、 C の関数としての α を模式的に図1.4.3-6に示す。

(4) 真性コロイドと擬似コロイドの共存を考えた場合の $\alpha(x, t)$ の設定

この場合には、拡散係数についての分布は、図1.4.3-7に示す様に五群の離散的分布となる。また、 α の設定は、(10)~(13)式を統合することによって次の様なものとなる。

(疑似コロイドは常に存在)

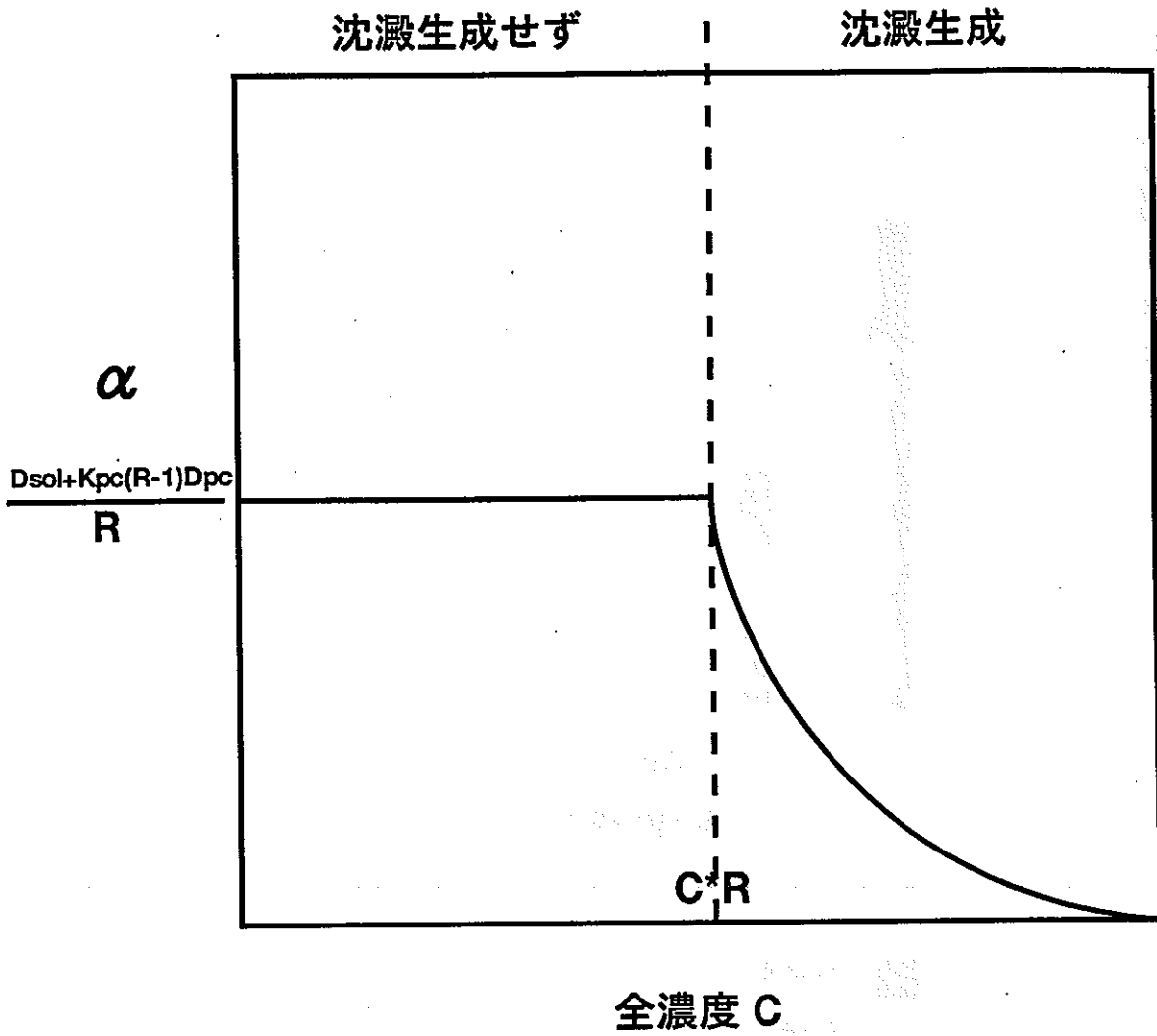
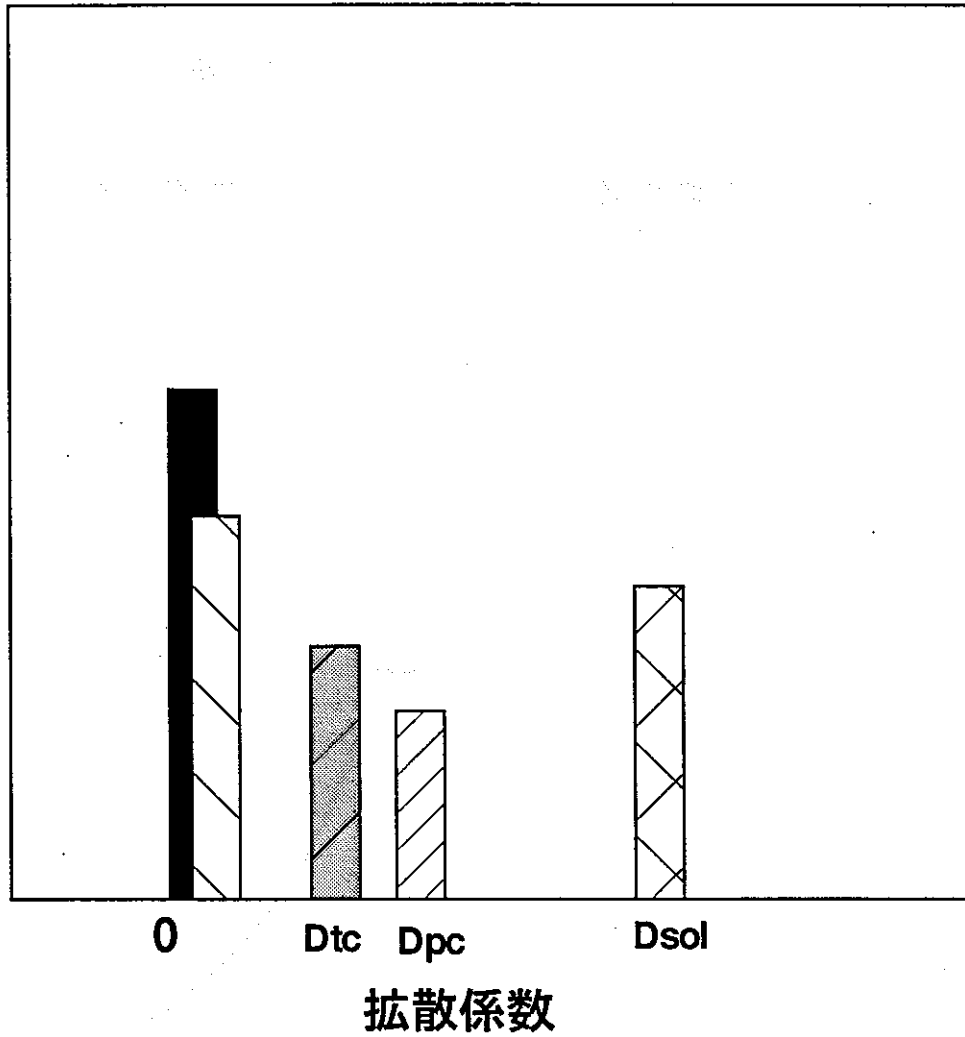


図 1.4.3 - 6 全濃度 C の関数としての α の表現
(疑似コロイドモデル)

確率密度 (相対濃度)



- 沈澱物
- ▧ 吸着種
- ▩ 溶存種
- ▨ 真性コロイド
- ▧ 疑似コロイド

図 1.4.3-7 真性コロイド及び疑似コロイドを考慮した場合の拡散係数の分布形

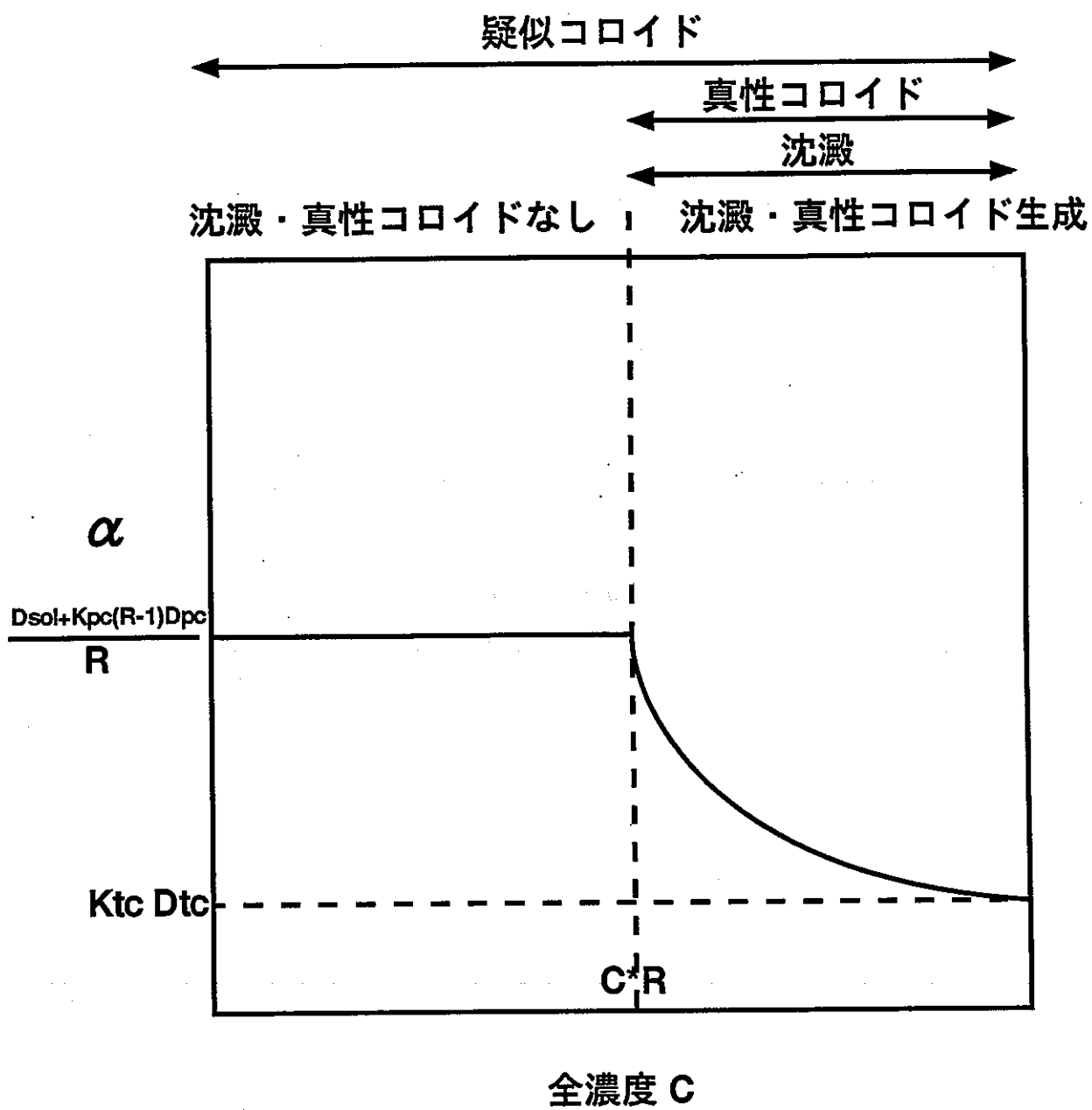


図 1.4.3 - 8 全濃度 C の関数としての α の表現
(真性コロイド、疑似コロイド共存モデル)

① 液相濃度が溶解度以下の場合

$$\alpha = \frac{D_{s,01} + K_{PC} (R-1) D_{PC}}{R} \quad (15)$$

② 液相濃度が溶解度以上の場合

$$\alpha = \frac{D_{s,01} C^* + K_{PC} D_{PC} (R-1) C^* + K_{1c} D_{1c} D_{PC}}{C} \quad (16)$$

(5) 非線型吸着を考えた場合の $\alpha(x, t)$ の設定

次に、ベントナイトへの核種吸着は非線型である場合の $\alpha(x, t)$ について検討する。ここで、簡単のために溶存種と吸着種のみに着目することとし、それぞれの濃度を c 及び q とする。

線型吸着の場合には、 c と q との間に簡単な比例関係が成立したが、非線型吸着の場合には c の非線型の関数 f を用いて、

$$q = f(c) \quad (17)$$

という表現が一般的に成立する。 f の具体的な形として Langmuir 型の吸着等温式

$$f(c) = \frac{\mu c}{1 + \eta c} \quad (18)$$

但し、 μ 、 η は正の定数

を考えると、 c は全濃度 C の関数として次の様に表現される。

$$C = c + q = c + \frac{\mu c}{1 + \eta c} \quad (19)$$

(19)式を c についての二次式、

$$0 = \eta c^2 + (\mu - \eta C + 1) c - C$$

に書き直しその正の根をとることにより、

$$C = \frac{-(\mu - \eta C + 1) + \sqrt{(\mu - \eta C + 1)^2 + 4 \eta C}}{2 \eta} \quad (20)$$

従って、対応する $\alpha(x, t)$ は、

$$\alpha(x, t) = \frac{D_{s,01}}{C} \frac{-(\mu - \eta C + 1) + \sqrt{(\mu - \eta C + 1)^2 + 4 \eta C}}{2 \eta} \quad (21)$$

と設定されることになる (図1.4.3-9)。

(6) 環境条件が時間・空間的に変化する場合の $\alpha(x, t)$ の設定

(1)~(5)においては、全て環境条件が一定であることを仮定しており、このため、対応する c^* 、 R 、 K_{ic} 、 K_{pc} 、 μ 、 η といったパラメータは全て定数として取り扱われていた。広義拡散方程式を用いた手法において、時間・空間的に変化する環境条件を取り扱うことは最も容易であり、即ち、これらのパラメータを x 、及び t の関数として定義することによって定式化を行えば良い。

例えば、人工バリア内の酸化還元状態が時間的に変化する場合、空間的な不均質性をひとまず無視することとして、これに伴う c^* の時間的変化を、

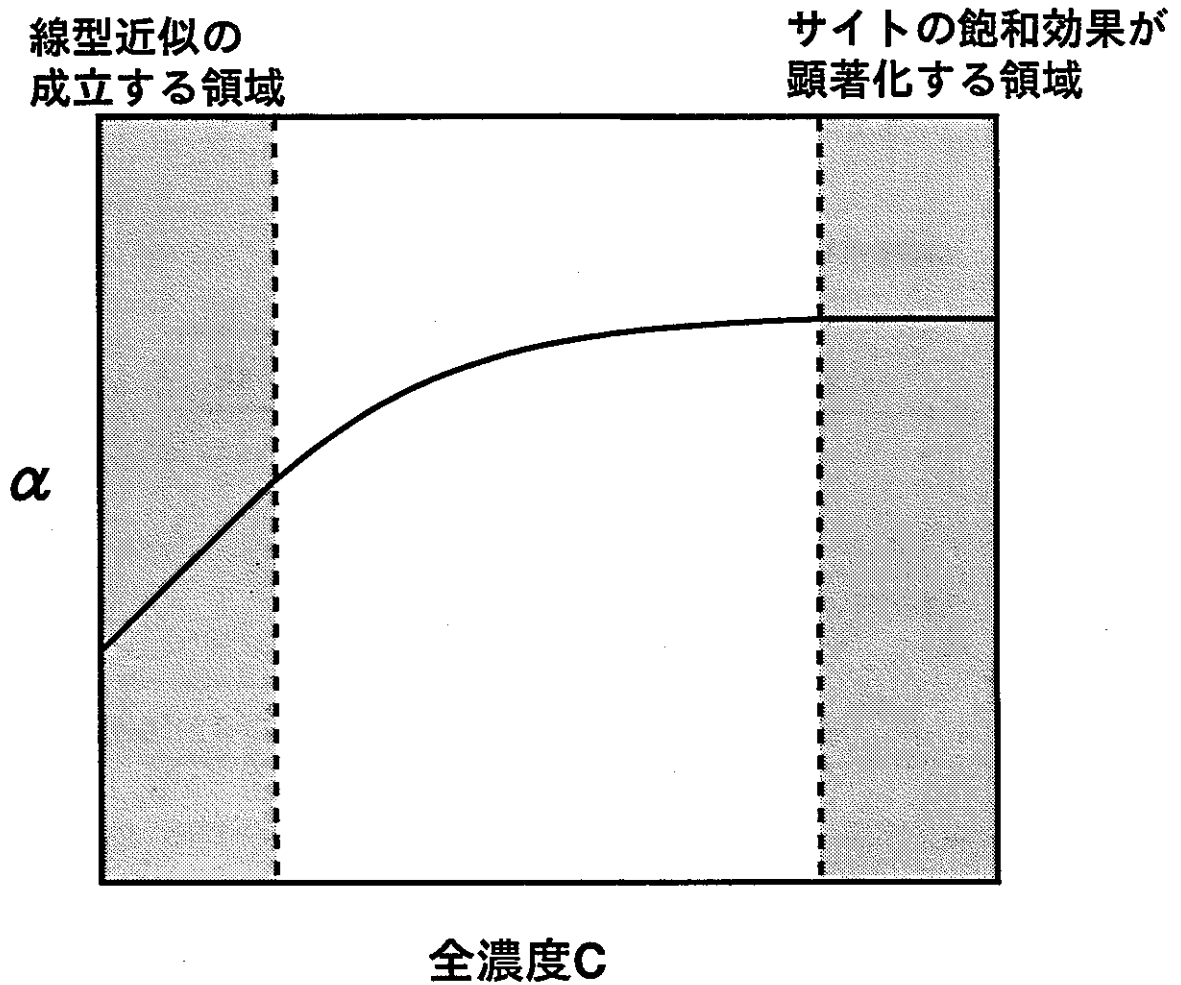


図 1.4.3 - 9 全濃度Cの関数としての α の表現
(非線型吸着モデル)

$$c^*(t) = c^*_o \cdot \phi(t) \quad (22)$$

と表現することができる。ここで、 c^*_o は標準（還元）状態での溶解度であり、 $\phi(t)$ は状態変化の原因となっている具体的なF E P s（例えば酸化性地下水の浸入、あるいは放射線分解）のタイムスケールと時間的挙動（周期的か非周期的か等）の予測に対応して設定される関数である。

また、空間的な不均質性をも考えることとした場合には、 $\phi(t)$ の代わりに $\Phi(x, t)$ を考えることによって、

$$c^*(x, t) = c^*_o \cdot \Phi(x, t) \quad (23)$$

と表現することができる。例えば、放射線分解による酸化還元フロントの生成と移動を考える場合には、 $\Phi(x, t)$ を経時的に移動するステップ関数（あるいはより現実的に誤差関数）として表現することも可能である（図1.4.3-10参照）。

(7) 反応速度を考慮した場合の解法

系が非平衡となる可能性がある場合には、(1)-(6)に示した様な α についての代数的な方程式を導くことはできない。この場合には、以下に示す様に $\alpha(x, t)$ についての常微分方程式を広義拡散方程式と連立することが必要となる。

(5)に述べた真性コロイド及び擬似コロイドの共存モデルに於いて、最も遅い反応は恐らくは沈殿の生成とその溶解であり、この反応のみを速度論的に表現することを考える（つまり、他の全ての反応は瞬時平衡であるとする）。

この時、液相濃度を $C_{o,1}$ とすると、他の移行形態の濃度はそれぞれ次の様になる。

吸着種 $(1 - K_{pc}) (R - 1) C_{s,01}$

擬似コロイド $K_{pc} (R - 1) C_{s,01}$

沈殿 $(1 - K_{tc}) (C - C_{s,01} \cdot R) \text{Sh} (C - C_{s,01} R)$

真性コロイド $K_{tc} (C - C_{s,01} \cdot R) \cdot \text{Sh} (C - C_{s,01} R)$

次に、溶解・沈殿の反応速度式は一般に、

$$C_{s,01} t = g (C_{s,01}, C^*) \quad (24)$$

と書かれる。ここで、最も簡単であり、かつ溶解・沈殿については最も一般的である一次反応速度式

$$C_{s,01} t = -\varepsilon (C_{s,01} - C^*) \quad (25)$$

を考えることとする。

(25)式は以下の関係を用いて $\alpha (x, t)$ についての式に書き直すことが可能である。

$$\alpha = \frac{D_{s,01} C_{s,01} + D_{pc} C_{pc} + D_{tc} C_{tc}}{C} \quad (26)$$

$$= \frac{D_{s,01} C_{s,01} + D_{pc} K_{pc} (R - 1) C_{s,01} + D_{tc} K_{tc} (C - C_{s,01} R) \text{Sh} (C - C_{s,01} R)}{C}$$

$$* C_{s,01} = \frac{\{\alpha - D_{tc} K_{tc} \text{Sh} (C - C_{s,01} R)\} C}{D_{s,01} + D_{pc} K_{pc} (R - 1) - D_{tc} K_{tc} R \text{Sh} (C - C_{s,01} R)} \quad (27)$$

(27)式を(25)式に代入することにより、

$$\frac{\alpha t C + \{\alpha - D_{tc} K_{tc} \text{Sh} (C - C_{s,01} R)\} C t}{D_{s,01} + D_{pc} K_{pc} (R - 1) - D_{tc} K_{tc} R \text{Sh} (C - C_{s,01} R)} \quad (28)$$
$$= -\varepsilon (C_{s,01} - C^*)$$

を得る。この(28)式を広義拡散方程式と連立することによって、溶解・沈殿のKineticsを考慮したモデルが得られることとなる。

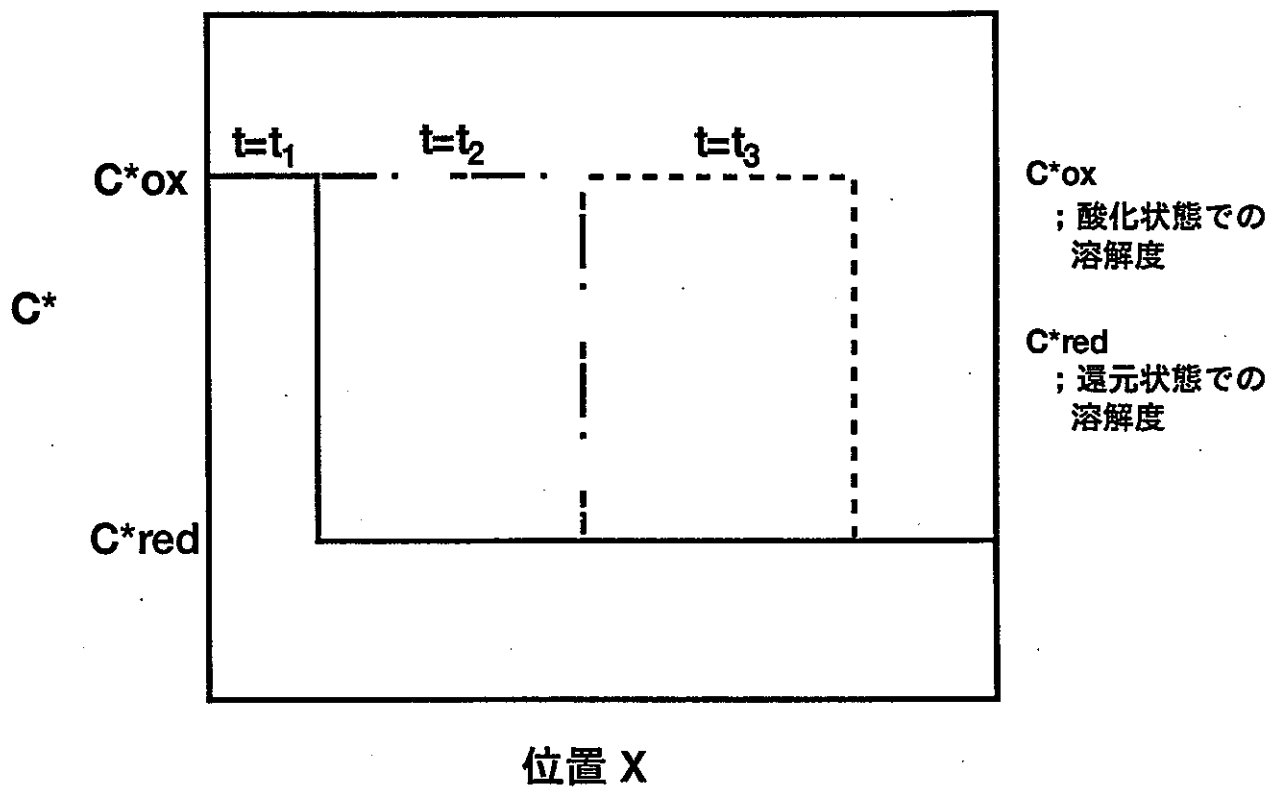


図 1.4.3 - 10 時間—空間的条件変化の影響の表現例
 放射線分解による酸化還元フロントの生成・移動)

1. 参考文献

- [1] H.Takase and P.Grindrod, " Modelling Precipitation of Radioactive Elements and Their Effect upon Redox Conditions ", Proceedings of 5th HLW Management Conference, Las Vegas, to be published
- [2] 動力炉・核燃料開発事業団, "高レベル放射性廃棄物地層処分研究開発の技術報告書", PNC TN 1410 92-081, 1992
- [3] NAGRA, Project Gewähr 85, 1985
- [4] 高瀬博康, MESHNOTE プログラムマニュアル, 日揮株式会社技術レポート, 1992
- [5] D.Hodgkinson and T.Summerling, " A Review of Approaches to Scenario Analysis for Repository Safety Assessment ", Proceedings of International Symposium on Safety Assessment of Radioactive Waste Repositories, CEC, IAEA, OECD/NEA, CEA, 1989
- [6] J.Andersson et al, " The Joint SKI/SKB Scenario Development Project ", SKI Technical Report 89:14, 1989
- [7] R.Pusch et al, " Piping and Erosion Phenomena in Soft Clay Gels, SKB TR 87-09, 1987
- [8] P.Grindrod and H.Takase, " The Saturation Behaviour of Clay Buffers ", Intera Report, 1993
- [9] 高瀬博康, ニアフィールド環境変化の動的シナリオ解析手法の開発(1), PNC ZJ 1281 93-008, 1993

2. ソフトウェアの改良

1.に述べた広義拡散方程式による動的解析を実効するために、昨年度の研究[1] で作成した二つのソフトウェアSPADE とSAM GUI のそれぞれについて改良を行った。SPADE は、広義拡散方程式とそのニアフィールド問題への適用（1.4.2及び1.4.3参照）の様に、多数の微分方程式と代数的制約条件とから成る連立系(DAE:Differential and Algebraic Equations)についての汎用ソーパーであり、SAM GUI はSPADE の計算結果を個々の計算ケースについての詳細な情報及び多数のケース間での概括的情報のそれぞれについて整理・図形化するインターフェイスである。

以下に、これらのソフトウェアの概要と、今年度行った改良の内容について述べる。

2.1 汎用DAE ソーパー SPADE

SPADE(Solver Package for Algebraic and Differential Equations)は、複数の微分方程式と代数的制約条件とから成る連立系であるDAE についてのソーパーであり、特に解の安定性が問題となる「stiff な系」（複数の微分方程式が、それぞれタイムスケールの数桁異なる物理的プロセスに対応しているために、精度良い解を得るためには小さいタイムスケールに対応して小さいタイムステップをとり、長いタイムスケールをカバーするためには莫大な数のステップをとることを必要する様な問題）についても効率的な計算を可能とするという意味で、汎用性を有するものである。このような汎用性を得るために、SPADE は、可変タイムステップ及び可変求積オーダーに対応した最新の計算アルゴリズムを用いている。

本節では、まず2.1.1においてSPADE の機能の概要を述べた後、2.1.1において今年度行った改良（主としてDAE の設定に関するもの）の内容をまとめることとする。また、今後更に考えて行く必要のある諸課題について、2.1.3において検討を加える。

2.1.1 SPADE の基本機能

SPADE は半explicitと呼ばれるクラスのDAE についての汎用ソーパーである。半explicitなDAE は、微分方程式群と代数方程式群とを分離して取り扱うことが可能であるため、

より一般的な全explicitな問題に比べて、比較的容易に解法することが可能である。この意味からは、SPADE の適用性にはある種の制約があることとなるが、半explicitな問題は1.に述べた広義拡散方程式や、より一般的な反応・拡散問題といった性能評価で対象とする問題の大部分を含むものである。また、SPADE は、他の汎用ソルバーと同じくインデックス0ないしは1のDAE を対象とする。以下に、DAE のインデックスという概念から始めて、SPADE の機能の概要をまとめることとする。

(1) DAE についての理論

① DAE の一般形

全explicitな(最も一般的な)形式のDAE は次の様に書かれる。

$$\dot{F}(x, x, t) = 0 \quad (1)$$

但しここでドットは時間に関する微分を表すものとする。(1)式は一階の導関数のみを含むものであるが、常微分方程式論において良く知られている様に、高階の微分方程式は一階の常微分方程式の連立として表現できるため、このことによって一般性が失われることはない。また(1)式には位置に関しての微分は含まれていないが、本来の偏微分方程式を空間について差分化することにより、これを格子点についての複数の常微分方程式に展開することが可能であるため、この点も(1)式の一般性を制限するものとはならない。

これに対して、SPADE が対象とする半explicitなDAE は、次の様なものである。

$$\dot{x} = f(x, y, t) \quad (2)$$

$$0 = g(x, y, t)$$

但し、 f 及び g は x, y, t についての連続関数とする。

ここでは、DAE は微分方程式群 ((2)中上の形式) と代数方程式群 (下の形式) とに区分され、更に変数は、微分変数 x と代数変数 y とに大別される。SPADE が(2)を対象とするのは、(2)式については適切な数値解法が見出されているのに対して、(1)式についての数値解法は多くの場合不安定なものとなることによる。

② DAE のインデックス

一般に線型の常微分方程式のみから成る連立系では、任意の初期条件について滑らかな解が得られるが、代数的制約条件を含むDAE ではこれは必ずしも正ではない。あるDAE では、連立線型常方程式系と同様に常に滑らかな解が得られるが、他の場合では、ある初期条件から出発した解は有限時間のうち無限大に発散（解の爆発と呼ばれる）あるいは振動してしまう。このため、DAE のソルバーを作成する際には、常に滑らかな解が得られる様なDAE のクラスを予め明らかにしておくことが重要である。

この様なDAE の分類を与える概念がインデックスである。DAE のインデックスとは、この方程式系に含まれる全ての変数の一階の導関数が、 x 及び t の連続関数となる様にするための微分操作の回数として定義される。これをもう少し具体的に見るために、(2)式を考える。(2)式に於いて、DAE を構成する方程式が全て前者の形であり、

$$\dot{x} = f(x, t) \quad (3)$$

という形式であった場合、ここでは既に全ての x は x と t の連続関数として表現されており、上記の定義から、そのインデックスは0となる。

また、より一般の(2)式については、後者の式を時間微分することにより、

$$\dot{x} = f(x, y, t) \quad (4)$$

$$g_x(x, y, t) \dot{x} + g_y(x, y, t) \dot{y} = -g_t(x, y, t)$$

と書き換えられる。つまりここで、左辺の g_x が連続関数であれば、本来のDAE

((2)式)で代数変数であった y についてもその導関数が x, y, t の連続関数で表現されることとなり、この場合のインデックスは1である。

これに対して、 g_x が不連続点を有する様な場合には、変数を変換して再度微分を行うことが必要となる。この操作によって新たに得られる関数が連続である場合には、このDAE のインデックスは2となる。以下、同様の操作を考えることによって、より高いインデックスのDAE が想定される。

SPADE は、全ての汎用DAE ソルバーと同様、インデックス0及び1のDAE のみを解法するものである。

この制約のために、ユーザーは、SPADE に入力するDAE を予め外部でインデックス0あるいは1のものに変換しておくことが必要である。これは、原理的には上記の微分・代入操作を繰り返すことによって実施し得るが、適当な代入操作が常に見つかるとは限らない。そこで、これに代わるより現実的な方法が見出されている。この方法について、次に述べることとする。

③ インデックスの降下と制約条件の安定化

②に述べた微分と代入操作の繰り返しによるインデックスを行った結果、本来のDAE である(2)式は、

$$\dot{x} = f(x, t) \quad (5)$$

という常微分方程式系に置き換えられる。然しながら、この過程で(2)式に含まれていた代数的制約条件は失われてしまい、代わりに、(5)式についての可能な初期条件の組合せのうちある一部のもののみが(2)式で対応することとなる。ここで、本来のDAE に含まれていた制約条件が、

$$0 = g(x, t) \quad (6)$$

という形に書き直されているものと仮定して、新たな変数を導入することによって、(5)式と(6)式とが連立された場合の解のみが現れる様にするというのがこの方法の主眼である。

このために、 g_x の階数と同じ大きさを持つ変数ベクトル μ を導入して、新たなDAE として、

$$\dot{x} = f(x, t) + g_x^T \mu \quad (7)$$

$$0 = g(x, t)$$

を考える。ここで、 $x(t)$ が本来のDAE の解となることの必要十分条件は、ある時刻 t における $x = x(t)$ について $g(x, t) = 0$ が成立することである。従って、(7)式の唯一の解は、 x が(5)式の解であり、かつ $\mu = 0$ であるものに限られることとなる。

この方法の具体例については、昨年度の研究報告書を参照されたい。なお、1. に述べた広義拡散方程式のニアフィールド問題の適用に対応するDAE はインデックス1であり、上記の操作は必要とはならない。

(2) stiff な問題のための時間差分の方法

① stiff な常微分方程式及びこれに対する解法の要件

常微分方程式系における stiff な問題とは、物理的には、タイムスケールの大きく異なる複数のプロセスが同時に含まれる様な現象として理解される。これを数学的に表現すると、次の様なものとなる〔2〕。

「DAEにおける stiff な問題とは、(2)式に含まれるのヤコビアン行列

$$J_{ij}(x, y, t) = (\partial F_i / \partial x_j)(x, y, t) \quad (8)$$

が複数の固有値を持ち、それぞれの固有値の実部は負でその大きさが異なる様なものである。」

この様な問題は、それぞれ時間と共に減衰する複数の成分を持つが、固有値の大きさが大きく異なる時には、一つの成分が急速に減衰し、他はそのままとどまるという状況が生じる。

例えば、次の方程式

$$\dot{x}^{(1)} = \begin{bmatrix} 998 & 1998 \\ -999 & -1999 \end{bmatrix} x^{(1)}, \quad x^{(0)} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (9)$$

は良く知られた解析解

$$x^{(1)} = \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \exp(-t) \\ \exp(-1000t) \end{bmatrix} \quad (10)$$

を有する。これを数値解法によって求める際には、対応するヤコビアン行列の固有値が -1000 及び -1 となる。つまり、精度を維持するためには 10^8 倍の速さで急減少する固有値 -1000 に対応した小さなタイムステップのサイズが求められ、かつ、現象のふるまいを見るためには固有値 -1 の解が十分減衰する様な長い期間の計算が必要となる。

このことは、通常の陽解法を用いた場合には、計算するタイムステップ数が莫大なものとなることを意味する。

これに対して、陰解法は何らかの形で解の経時変化を内挿する関数の存在を前提として、過去の計算結果と時間的に整合的となる次回値 (predictor) を求め、これを方程式系の制約を満足する様に修正する (corrector) という predictor-corrector のスキームを用いるものである。このため、陰解法は stiff な問題に対して、より大きなタイムステップを採ることを可能にする。

陰解法を用いた stiff な問題の解析において、より高効率の処理を可能とするためには、可変タイムステップ可変オーダーのアルゴリズムをとることである。まず、可変オーダーのスキームを用いることにより、(2) 式中の f , g が時間に関して非線型性の強いふるまいをした時、これに対応して高次の予測オーダー (内挿関数の次数と考えてよい) が自動的に選択され、合理的な (即ち極端に小さくない) タイムステップで解の精度を維持することができる。更に、可変タイムステップのスキームを用いることにより、解空間上の流線としての解のふるまいに即した処理が為される。即ち、単位時間当りの変化が大きければ小さいタイムステップを選び (可変オーダーであることにより、極端に小さくはない)、系が定常に近づけば大きいタイムステップが選ばれる。

つまり stiff な問題に最も適した数値解法とは、可変タイムステップ可変オーダーのアルゴリズムにおいても、predictor-corrector スキームが安定となる様な陰解法であるようなものであるということが出来る。

② 従来の陰解法

stiff な問題について適用可能な陰解法は Gear 等 [4] によって 1970 年代初頭に開始された。BDF 法の基本的な発想は、あるタイムステップにおける時間微分値をそのタイムステップから M ステップ前までの間の計算値の線型和として与えることである。

$$dx_n/dt = (1/h) \sum_{j=1}^M b_j x_{n-j} \quad (11)$$

(h : タイムステップの大きさ)

(11) 式は、そのまま predictor のスキームとして用いられる。

$$x^p = x_{n-1} + h \quad dx_n/dt \quad (12)$$

次に、corrector としては、上式の x_n^p を初期推定値として、Newton反復を行うことによって修正値 x_n^c が算出される。

このBDF法による解は、固定タイムステップの場合、最大5次まで安定であるということがGear〔3〕によって示されている。しかし、可変タイムステップのアルゴリズムにおいては、不安定性が生ずることも示されている〔4〕。この点は、前項で述べた様に、DAEの汎用的な数値解法としては致命的な短所であると言える。

この点を改良し、可変タイムステップ可変オーダーでも安定となる様にしたものが、Byrne とHindmarsh による改良BDF法である。

③ 改良BDF法〔4〕

a. Predictor 及び Correctorの公式

$t = t_{n-1}$ 迄の解 x_{n-1} ($j = 1, \dots, n$) が得られているとして、 q 次の予測オーダーで x_n^p を推定するとする。この時、 q 次の内挿多項式 (自由度 $q + 1$) で、 g 個の前回値を通り、かつ、 $t = t_{n-1}$ でその解の微係数がこの関数の傾きに一致する様なものが存在する。即ち、多項式を $\pi_{n-1}(t)$ とすると、

$$\begin{cases} \pi_{n-1}(t) (t_{n-1}) = x_{n-1}, & i = 1, 2, \dots, q \\ \dot{\pi}_{n-1}(t_{n-1}) = \dot{x}_{n-1} \equiv f(x_{n-1}, t_{n-1}) \end{cases} \quad (13)$$

Predictor スキームを、ここでは、この様な多項式 π_{n-1} を π_n :

$$\begin{cases} \pi_n(t_{n-1}) = x_{n-1}, & i = 0, 1, \dots, q, \\ \dot{\pi}_n(t_n) = \dot{x}_{n-1} \equiv f(x_n, t) \end{cases} \quad (14)$$

に更新する問題と考える。そこで、多項式 π_n を π_{n-1} の単なる外挿として、

$$\begin{cases} x_n^p = \pi_{n-1}(t_n) \\ \dot{x}_n^p = \dot{\pi}_{n-1}(t_n) \end{cases} \quad (25)$$

を満足する様に設定する。勿論この x_n は元の制約条件を満足するものではないために、Newton反復による correctorスキームを経て x_n^0 に修正される。そして、この両者が近いものであれば、最終解はtime evolutionとして整合的であり、かつ制約条件を満たす真の解として受け入れられる。

以上では、一つの変数 x について説明をしたが、全く同じことが N 個の変数からなるベクトル \mathbf{x} についても成立する。以下、このより一般的な条件の下で説明を続ける。

さて、 q 次の Nordsieck 列 Z_{n-1} を考えてみる。

$$Z_{n-1} = \left(x_{n-1}, h \dot{x}_{n-1}, \frac{h^2}{2} \ddot{x}_{n-1}, \dots, \frac{h^q}{q!} x_{n-1}^{(q)} \right) \quad (16)$$

この Nordsieck 列は、本来は truncate されたテイラー展開の各項を成分とするベクトルであるが、今考えている問題では、2 階以上の導関数は解析的には未知であるため、前回までの計算値を用いて差分的に定義することとする。この時、 q 次の多項式となる Z_{n-1} の成分和は、 q 個の過去の計算値を通り、かつ、 $t = t_{n-1}$ での微係数を与えるという意味で、 π_{n-1} と同値である。

そこで、改良 BDF 法では、 π_{n-1} の代わりに、これと同値の情報を含む Nordsieck 列 Z_{n-1} を更新する。この公式は簡単に、次式で与えられる。

$$Z_n^p = Z_{n-1} A_{(q)} \quad (17)$$

但し、 $A_{(q)}$ は q 次のパスカル三角行列：

$$A_{ij (q)} = \begin{cases} 0, & i < j \\ \binom{i}{j} = i! / j! (i-j)!, & i \geq j \end{cases} \quad (18)$$

である。

(例)

$$A_{(5)} = \begin{pmatrix} 1 & & & & \\ 1 & 1 & & & \\ 1 & 2 & 1 & & \\ & & & & \\ & & & & \end{pmatrix}$$

$$\begin{pmatrix} 1 & 3 & 3 & 1 & & \\ 1 & 4 & 6 & 4 & 1 & \\ 1 & 5 & 10 & 10 & 5 & 1 \end{pmatrix}$$

ここで、真の Nordsieck列と Z_n^p との間の誤差について極めて便利な関係式が成立することを示す。

明らかに、真の Nordsieck列は π_n に対応するものでなくてはならず、

$$\begin{cases} Z_n = (x_n, h \dot{x}_n, \dots, \frac{h^q}{q!} x_n^{(q)}) \\ x_n^{(q)} \equiv \pi_n^{(q)}(t_n) \end{cases} \quad (19)$$

が成り立つ。

他方、 Z_n^p については、

$$\begin{cases} Z_n^p = (x_n^p, h \dot{x}_n^p, \dots, \frac{h^q}{q!} x_n^{p(q)}) \\ x_n^{p(q)} \equiv \pi_{n-1}^{(q)}(t_n) \end{cases} \quad (20)$$

である。そこで、

$$\Delta_n(t) = \pi_n(t) - \pi_{n-1}(t) \quad (21)$$

とし、 x_n の予測誤差を e_n とする：

$$e_n = x_n - x_n^p \quad (22)$$

この時、明らかに $\Delta_n(t)$ について、

$$\begin{aligned} \Delta_n(t_{n-i}) &= 0, \quad i=1, 2, \dots, q \\ \Delta_n(t_n) &= e_n \end{aligned} \quad (23)$$

が成り立つことから、 $\Delta_n(t)$ は一意に、

$$\Delta_n(t) = \prod_{i=1}^q [(t - t_{n-i}) / (t_n - t_{n-i})] e_n \quad (24)$$

と定まる。また更に、

$$S = (t - t_n) / h, \quad \xi_i = (t - t_{n-i}) / e_n \quad (25)$$

として、多項式 $\Lambda_n(s)$ を、

$$\Lambda_n(s) = \prod_{i=1}^q (1 + s / \xi_i) \quad (26)$$

と定義すると、 $\Delta_n(t)$ は、

$$\Delta_n(t) = \Delta_n(t_n + h_n) = \Lambda_n(s) e_n \quad (27)$$

として表される。

以上に基づき Nordsieck列の予測誤差を算出すると、

$$\begin{aligned} Z_n - Z_n^p &= h^j \pi_n^{(j)}(t_n) / j! - h^j \pi_{n-1}^{(j)}(t_n) / j! \\ &= h^j \Delta_n^{(j)}(t_n) / j! \\ &= \Lambda_n^{(j)}(0) e_n / j! \\ &= \iota_j e_n \quad (j=0, 1, \dots, q) \quad (28) \end{aligned}$$

となる。但し、 ι_j は $\Lambda_n(s)$ の j 次の項の係数とする。 $\iota = [\iota_0, \iota_1, \dots, \iota_q]$

とすると、(29) 式は、

$$Z_n = Z_n^p + e_n \iota \quad (29)$$

と書き換えられる。

さて、次に correctorスキームを考える。(29) 次の $j=1$ の例は、

$$h X_n = h X_n^p + (X_n - X_n(0)) \iota_1 \quad (30)$$

となる。 $\iota_1 = \sum_{i=1}^q (1/\xi_i)$ であって既知であり、また $X_n = f(x_n, t_n)$ である

ことから、(30) 式を変形して、未知の(求めるべき) X_n^c を u とすると、

$$G(u) \equiv (u - X_n^p) - (h/\iota_1) [f(u, t_n) - x_n(0)] \quad (31)$$

となり、 $G_{(n)} = 0$ の解が求められる X_n^0 となる。これは、行列N

$$N = I - (h/\tau_1) J \quad (32)$$

を用いたNewton法によって解くことができる。但しJはfのヤコビアン行列。

b. 誤差管理とタイムステップサイズ及び予測オーダーの調整

a. に述べた Predictor-Correctorの公式が、stiff な問題に対して真に効率的な解法となるためには、既に述べた様に、解のふるまいに対応してタイムステップサイズと予測オーダーが適宜自動的に調整されることが必要である。このためには、まず予測の誤差を定量化することが必要となる。

古典的なBDF法では、k次の陰解法における局所予測誤差（あるタイムステップでの誤差） $E_n(k)$ は、

$$E_n(k) = X_n + \frac{1}{b_{n0}} \left(h \dot{x}_n + \sum_{i=1}^k b_{ni} x_{n-i} \right) \quad (33)$$

で与えられる。但しここで、 x_i はcorrection後の最終解としbは(11)式における係数とする。

(33)式を、a.で用いた多項式 π を用いて変形することにより、近似的に以下が成立することがByrneとHindmarsh〔4〕によって確かめられている。

$$E_n(q) = -1/\tau_1 \left[1 + \prod_2^q \left(\frac{t_n - t_{n-1}}{t_{n-1} - t_{n-2}} \right) \right]^{-1} e_n + O(H^{q+2}) \quad (34)$$

$$E_n(q-1) = -(\xi_1, \xi_2 \dots \xi_{q-1} / \tau_1(q-1)) (h^q y_n^{(q)} / q!) + O(H^{q+1}) \quad (35)$$

$$E_n(q+1) = \frac{-\xi_{q+1}}{(q+2)\tau_1(q+1) \left[1 + \prod_2^q \left(\frac{t_n - t_{n-1}}{t_{n-1} - t_{n-2}} \right) \right]} (e_n - Q_n e_{n-1}) + O(H^{q+3}) \quad (36)$$

但し、 $Q_n = (C_n / C_{n-1})(h/h_{n-1})^{q+1}$

$$C_n = \xi_1 \xi_2 \cdots \xi_n \left\{ 1 + \prod_{i=2}^n \left(\frac{t_n - t_{n-i}}{t_{n-1} - t_{n-i}} \right) \right\} / (q+1)!$$

ここで、 H はタイムステップサイズの上限值であり、 $E_{n(q)}$, $E_{n(q-1)}$, $E_{n(q+1)}$ はそれぞれ $t = t_n$ での q , $q-1$, $q+1$ オーダー予測の誤差である。

以上の誤差公式に基づき、タイムステップサイズとオーダーの調整を以下の方法で行う。

まず、ユーザーが誤差許容範囲 ε 、重み係数ベクトル $w = (w_1, \dots, w_n)^T$ 、及び ε に対応する期間 S (ε/s が単位時間での誤差許容範囲となる。) を定めることが必要となる。ここで、次に、以下のテストを行うものとする。

$$\begin{cases} D_{q+i} \leq \bar{\varepsilon} \equiv \varepsilon h / s & (i = -1, 0, 1) \\ D_{q+i} = \max_j \{ |E_{n_j}(q \pm i)| / w_j \} \end{cases} \quad (37)$$

そして、この条件が満たされる場合、次回の計算が十分な精度で実施できるものと仮定する。また、逆に (37) 式の条件を成立させるためには、各オーダーそれぞれ次回のタイムステップを

$$h'_{q+i} = \eta_{q+i} \cdot h \quad (i = -1, 0, 1) \quad (38)$$

と修正することが必要と判断するものとする。この時、 η_{q+i} は次を満たすことが十分条件である。

$$\eta_{q+i} = (\bar{\varepsilon} / D_{q+i})^{1/(q+i+1)} \quad (39)$$

実際の計算においては、次の手順で上記の調整が行われる。

(i) q 次の予測で $t = t_n$ のステップが終了して、満足すべき結果が得られた時、次に、 q , $q-1$, $q+1$ で次の計算を行った場合の誤差 $E_{n(q)}$, $E_{n(q-1)}$, $E_{n(q+1)}$ を (45) ~ (47) 式により算出する。

(ii) (i) の結果に基づき η_{q+i} ($i = -1, 0, 1$) を算出する。

(iii) η_{q+i} の中で最大となる様な i を選定し、次回のタイムステップ h' を

$$h' = \max_i (\eta_{q+i}) \cdot h \quad \text{とする。}$$

以上の手順により、常に計算効率を最大とする様に、予測オーダーとタイムステップサイズが自動的に調整されることとなる。

SPADEコードは、この Byrneと Hindmarshの改良BDF法によりDAEを解法するものである。なお、このコード自体のアルゴリズムについては、次節を参照されたい。

(3) SPADEの解法

本節では、まずSPADEの解法原理について述べ、次いで各ステップの詳しい手順をまとめる。

① 解法の原理

f_i 及び g_j をそれぞれ i 番目の常微分方程式と j 番目の代数方程式とし、 \dot{x}_i 及び y_j をそれぞれ微分変数及び代数変数とした時、(2) 式を次の様に表すことができる。

$$\begin{aligned}\dot{x}_i &= f_i(x, y, t) \\ 0 &= g_j(x, y, t)\end{aligned}\quad (40)$$

SPADEは、上記のDAEの stiffな方程式が含まれる場合、即ちタイムスケールが全く異なる複数のプロセス（前節の例における振り子の運動と半径方向の振動、あるいは速い化学反応と遅い化学反応等）が同時に含まれている場合にも安定で高効率の解法として、可変タイムステップ可変オーダーのBDFを用いている。

SPADEにおける数値解法の手順は、以下の通りである。

- ① (16) 式の形式でDAEを定義する。
- ② 関数 f_i 、 g_j 及び変数 x_i 、 y_j を指定する。
- ③ 全ての変数の初期値を与える。
- ④ 全ての変数の時間に関する微係数の初期値を与える。（代数変数については推定値）
- ⑤ 初期条件③、④が①で定義したDAEを満足することを確認する。
- ⑥ はじめのタイムステップの大きさ $h = h_0$ を指定する。
- ⑦ x_i 及び y_j の現在値 ($t = t$) に基づき、 $t = t + h$ での値 x^p 及び y^p を予測する。（predictor スキーム）
- ⑧ x^p 及び y^p を初期推定値としてNewton Raphson反復を行い、元のDAE

を満足する解 x^c 及び y^c を求める。(corrector スキーム)

- ⑨-a もしNewton Raphson反復が収束した場合、predictor (x^p, y^p) と corrector (x^c, y^c) とを比較する。これら二つが十分近いものであった時には、この比較結果に基づき次回のタイムステップのためのファクター η を算出する。
- ⑨-b もしNewton Raphson反復が収束しない場合、必要な措置をとり計算をやり直す。
- ⑩ オーダーを1つ下げた場合のタイムステップ次回値とオーダーを1つ上げた場合のタイムステップ次回値とを計算し、⑨-aの結果と併せ、タイムステップ最大となる様なオーダーを選択する。
- ⑪ 新たなタイムステップと新たな(もし必要であれば)オーダーを設定し、次々回のステップについての⑦を行う。

各ステップの詳細は次項で述べるものとする。なお、可変タイムステップ可変オーダーのスキームを用いることの利点は以下である。

- (i) 可変オーダーのスキームにより、もし与えられた f_1, g_1 が時間に関して非線型性の強いものであった時、これに対応して高次の予測が自動的に選定され、精度が保たれる。
- (ii) 可変タイムステップのスキームにより、解空間上の流線としての解のふるまいに即した処理が為される。即ち、単位時間当りの変化が大きければ小さいタイムステップを選び、系が定常に近づけば大きいタイムステップが選ばれる。

② 数値解法の概要

SPADEは、Byrne と Hindmarshによって考案された Nordsieck列のアプローチを用いている。即ち、未知変数 $\mathbf{||} = (\mathbf{||}, \mathbf{||})^T$ を直接取り扱うかわりに、 q 次の Nordsieck列

$$Z = \left(u, h \dot{u}, \frac{h^2}{2} \ddot{u}, \dots, \frac{h^q}{q!} u^{(q)} \right)$$

を実質的な未知数として解法を組み立てる。但し、 Z は $n \times q + 1$ の行列であり、 q が当該タイムステップにおける解法のオーダーである。

この Nordsieck列を用いることによって、前節で説明した改良BDF法が以下のような数値解法として具体化される。以下、簡単のため前出の手順のうち⑦からはじめることとする。

〔ステップ⑦〕

t_{n-1} までの結果が得られている時に、次の t_n のpredictionを得るためには、Nordsieck列を用いて次の計算を行う。

$$Z_n^p = A^{(q)} Z_{n-1} \quad (41)$$

ここで、上添字はpredictionに対応するものとし、 $A^{(q)}$ は $q + 1$ 次のパスカル三角行列である。

例えば $q = 2$ の時、

$$\begin{pmatrix} u_n^p \\ h \dot{u}_n^p \\ \frac{h^2}{2} \ddot{u}_n^p \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u_{n-1} \\ h \dot{u}_{n-1} \\ \frac{h^2}{2} \ddot{u}_{n-1} \end{pmatrix}$$

このステップで注意すべき点は、代数変数についても同様の操作によりpredictionが行われるということである。このことについてはすぐ後に説明を加える。

〔ステップ⑧〕

ステップ⑦の結果、即ち u_n^p を初期推定値として、Newton Raphson反復を行うのがこのステップである。この反復解法に用いられる行列 N は、以下の様に定義される。

$$N_n = I - \gamma_n J_n \quad (\text{微分変数について}) \quad (42)$$

$$N_n = \gamma_n J_n \quad (\text{代数変数について}) \quad (43)$$

但し、 J_n は f ないしは g のヤコビアン行列とする。

上式において代数変数については単位行列が含まれていない。これは、元の DAE において代数変数についての時間微分項が含まれていないことから明らかであろう。

さて、 N_n が得られた後には、 u_n^p を初期推定値として Newton Raphson 反復を実施する。現在の SPADE では各タイムステップにおいて 5 回までの反復が許容されている。

Newton Raphson 反復の収束判定は、以下の二つのクライテリアにより行われる。

$$\rho = \left[\frac{\|u_n^{p(k)} - u_n^{p(k-1)}\|}{\|u_n^{p(1)} - u_n^{p(0)}\|} \right]^{1/k} \quad (44)$$

$$\xi = \frac{\rho}{1 - \rho} \|u^k - u^{k-1}\| \quad (45)$$

ここでもし、 $\rho > 0.9$ あるいは $\xi > 0.3$ のうち少なくともどちらか一方が生じた場合には収束判定は“N₀”とされる。

〔ステップ⑨-a〕

Newton Raphson 反復が収束した場合、次には収束結果である u_n^c と初期推定値 u_n^p との比較がなされる。 u_n^c は DAE の制約条件を満足し、 u_n^p は解の経時変化を外挿するものであるが、真の解 u_n はこの両者の性質を有するものでなくてはならない。このため、 u_n^c と u_n^p とは互いに近い値であること

とが求められるためである。

またここで、代数変数 y はこの比較検定の対象からはずされる。というのも、 y についての導関数 \dot{y} は元の DAE の形式から言って厳密な意味を持たないためである。つまり、微分変数 $||$ についての比較検定の結果が “yes” であれば、 y_p の値によらず y_c が真の解 y_* として採用される。

SPADE では、 u_p と u_c との差異にユーザー指定のある上限を設けることによってこの検定を行う。そして、検定結果が “no” であれば、次の ⑨-b ステップと同様の措置がとられる。

〔ステップ⑨-b〕

Newton Raphson 反復が収束しなかった場合、あるいは u_c と u_p との差が大きい場合には、通常、タイムステップのサイズを低減することが試みられる。しかし、SPADE では計算効率を向上するために、以下の多段階の措置を用意している。これは、SPADE が行列 N を毎回計算している訳ではないために、使われている N が古すぎる可能性があるためである。タイムステップの低減に先立って、 N を更新することにより収束を得ることが試みられる。

- (i) N の定義式中にある γ の値が、最後に N を計算した時から大きく変わっているか否かを調べる。もし変わっている場合には、新たな γ の値を用いて N を計算し直す。
- (ii) γ の変化が大きくないか、(i) によっても収束が得られなかった場合、 J の再計算が行われる。
- (iii) (i) 及び (ii) にもかかわらず、未だ収束が得られない場合には、タイムステップの大きさを半分にしてステップ⑦が再び行われる。

〔ステップ⑩〕

Newton Raphson 反復が収束し、 u_c と u_p とが十分近いものであった時、 u_c は真の解 u_* として認められる。そして、次に行うことは、次のタイムステップにおけるタイムステップサイズと、予測のオーダーを定めることである。というのも、現在の予測のオーダーが次のタイムステップにおいても最適のものか否かは不明であるためである。

この操作は、具体的には、現在のオーダー q 、及びこれを低下、増加させた $q-1$ 、 $q+1$ のそれぞれのオーダーについて、次回のタイムステップサイズを得るためのファクター η 、 η_{up} 、 η_{down} を計算し、次回のタイムステップサイズが最大、つまり計算効率が最大となる様なオーダーを見出すことにより為される。

いずれのオーダーが採用されても、対応する次回の計算のためには新たなステップサイズ $\bar{\eta} h$ に対応して Nordsieck 列の更新を行う必要がある。

$$\bar{\eta} h = h \cdot \max \{ \eta, \eta_{up}, \eta_{down} \} \quad (46)$$

また、オーダーの変更を伴う場合には、Nordsieck 列の次数も変える必要がある。

オーダーを増加する際には、テイラー展開の $q+1$ 次の項を 0 と仮定して、Nordsieck 列の末尾に 0 列ベクトルを追加することによってこの変形は為される。しかし、オーダーを低下する場合には、他式の次数の項も含めて見直し新たな $q-1$ 次までのテイラー展開が $q-1$ 個の既計算値を内挿する様に調整するという少し複雑な操作が必要となる。

既に述べた様に、このステップ⑩は非線型の stiff な方程式を高効率で処理する上で最も重要な部分である。

〔出力〕

出力にあたっては、計算結果に対しての若干の調整を必要とする。これは、SPADE の可変タイムステップのスキームのために、ユーザー指定の時刻において計算結果が得られることは必ずしも保証されないためである。そこで、SPADE は、前回の Nordsieck 列が、即ち各成分の和によって表されるテイラー展開が、内挿関数としての条件を満足しているものであることを用い、ユーザー指定の時刻における推定値を求めてこれを出力する。

2.1.2 今年度の改良点

昨年度に開発を行った中核部分に加えて、SPADE を動的解析手法のツールとして効率的に活用するという観点から、今年度は次のような機能を付加することにより改良を行った。

- ・ 初期条件セットの自動設定機能
- ・ リスタート機能
- ・ ビルトイン関数
- ・ 偏微分方程式についての一次元、二次元グリッド作成及び自動差分化機能
- ・ 複数のケースに関する計算を一括処理するためのユーティリティプログラム

以下に、それぞれの改良の内容についてまとめる。

(1) 初期条件セットの自動設定機能

SPADE を用いてDAE の解を求める際には、まず、各微分及び代数変数についての初期条件を設定することが必要となる。ここで、(2)式中の代数制約条件が簡単な形式の場合にはユーザーが外部で計算を行うことにより初期条件セットが全ての制約条件を満足するにすることは比較的容易であるが、これは一般的には必ずしも成り立たない。そこで、ユーザーが適当な初期条件セットを与えた場合、SPADE が自動的にこの近傍にありかつ全ての制約条件を満足する様な正しい初期条件セットに書き直すことができる様に改良を加えた。これは次の様にして行われる。

$$\dot{X} = f(x, y, t) \quad (47)$$

$$0 = g(x, y, t) \quad (48)$$

というDAE について、ユーザーの入力した初期条件セット (x_0, y_0) が(48)式を満足しない場合、SPADE はこれを初期設定値として、代数変動 y_0 を変化させることにより、(48)式を満足する様な組合せを算出する。即ち、

$$(\tilde{x}_0, \tilde{y}_0) = (x_0, y_0) + (0, \delta y_0) \quad (49)$$

として、(48)式を満たす $(\tilde{x}_0, \tilde{y}_0)$ を与えるような δy_0 をNewton法で求める。このために、(48)式をTaylor展開して δy_0 について一次までの項を残すことにより、

$$0 = g(x_0, y_0, t_0) + GY(x_0, y_0, t_0) \delta y. \quad (50)$$

とし、この事から次が導かれる。

$$\delta y_0 = -Gy^{-1}(x_0, y_0, t_0) g(x_0, y_0, t_0) \quad (51)$$

但し、 Gy : ij 成分が、 $g_{ij} = \partial g_i(x_0, y_0, t_0) / \partial y_j$ として表される
ヤコビアン行列

そして、(51)式を反復的に用いることにより、目標とする δy_0 が算出される。この際、本来のDAEのインデックスが1であれば、 Gy は正規であり(51)式による計算が進められることとなる。もし Gy が特異であれば、対応するDAEのインデックスは2以上であることとなり、この時点でSPADEはエラーメッセージを発して停止する。

こうして δy_0 が、つまり $(\tilde{x}_0, \tilde{y}_0)$ が求まった後、SPADEはこれを(47)式に代入して、 \dot{X}_0 を算出する。最後に、 \dot{y}_0 については、(48)式を微分して、

$$0 = Gx(x_0, y_0, t_0) \dot{X}_0 + Gy(\tilde{x}_0, \tilde{y}_0, \tilde{t}_0) + g(\tilde{x}_0, \tilde{y}_0, \tilde{t}_0) \quad (52)$$

を得、これから、

$$y_0 = -gy^{-1}(\tilde{x}_0, \tilde{y}_0, \tilde{t}_0) \{gt(\tilde{x}_0, \tilde{y}_0, \tilde{t}_0) + Gx(\tilde{x}_0, \tilde{y}_0, \tilde{t}_0) \dot{X}_0\} \quad (53)$$

を導くことによって、上と同様にNewton法で \dot{y}_0 を推定する。こうして、初期値及び初期微分値が自動的に設定されることとなる。

(2) リスタート機能

DAEの解に不連続点がある場合、もしくはNewton反復で用いるヤコビアン行列が特異となった場合、SPADEはいくら小さなタイムステップをとっても計算を先に進めることができなくなる。そこで、もし反復計算におけるタイムステップが、計算機精度において全く同一となってしまった場合 ($t_n = t_{n+1}$)、SPADEは上記のうちのいずれかが

生じたものと判断する。そして、この時点での計算結果を(1)のユーザー指定の初期条件（の初期推定値）と同じ様に(48)式を満足するものに自動的にリセットして、計算をリスタートとする。このリスタートにもかかわらず状態が改善されない場合には、SPADE はエラーメッセージを発して計算を停止する。また、このリスタートにより計算が継続された場合には、計算終了時に出力ファイルにリスタート時刻（複数あった場合にはそれらの全てを）を打ち出す。

この操作においては、(1)と同様に、代数変数についてはリスタート前の数値が置き換えられ、SPADE の解は不連続なジャンプを示すこととなる。このジャンプの大きさについては制限がないために、リスタートによって計算結果が他の解に移っている可能性があることに注意を要する。一般的に言ってリスタートは、本来のDAE に何らかの問題があることの兆候であり、多くの場合はDAE の見直しの方がより根本的な解決策を与える。但し、環境条件をステップ関数的に変える場合などは、はじめから不連続性の存在が意図されており、リスタート機能は有効な計算手段となる。

(3) ビルトイン関数

DAE の入力、あるいは特に環境条件を時間・空間的に変化させることによって外部 FEPs の影響を表現するような場合、変化のパターンを種々の初等関数な特殊な関数で表す必要性が生じる。そこで、SPADE にこれらの関数を内蔵する様、改良を行った。

ビルトイン関数のリストを付録-4に示す。

(4) 偏微分方程式についての一次元及び二次元グリッドの作成及び自動差分化機能

SPADE で解くべきDAE は、偏微分方程式の形をとる可能性がある。1.で述べた広義拡散方程式はその例である。この様な場合には既に述べた様に、偏微分方程式を空間差分化することによって各格子点上での常微分方程式群に書き換える操作が必要となるが、これを手動で行う場合には、空間分割数だけ類似の常微分方程式を入力しなければならず、ユーザーの負担は大きなものとなる。そこで、SPADE が自動的に一次元及び二次元のグリッドを作成し、対応する差分化を行うことができる様改良を行った。この改良に

対応して、現在のSPADE 上では以下の操作により偏微分方程式を取り扱うことができる。

① 座標軸の指定

グリッドの作成に先立ち、ある変数（例えば x と y ）が位置変数であることを宣言し（SPATIAL キーワードによってなされる）、それぞれの最大・最小値と格子点とを定める。格子点の設定としては等間隔と不均一間隔のオプションがあり、前者の場合には分割数を、後者の場合は格子点のリストをそれぞれ入力する。

② グリッドの作成

①で設定された格子点の組合せとしてグリッドを作成する。

③ ベクトル変数の定義

偏微分方程式に含まれる物理量は位置の関数であり、これは②のグリッドに対応して各格子点上で定義されることとなる。つまり、この変数は複数の数値の組として表現されるベクトル変数である。ある変数がベクトル変数であることは VECTOR キーワードで宣言される。この宣言は方程式系の入力に先立って行われなくてはならない。

④ ベクトルサブスクリプトの指定

境界条件の指定や、指定された領域での代数的制約条件の定義においては、ベクトル変数の全体ではなく、そのうちの一つの格子点あるいは領域だけ抜き出す必要が生ずる。これを行うためには、予めベクトルサブスクリプトによる指定を行う必要がある。（例えば、 c がベクトル変数であるとき、これにベクトルサブスクリプトを付け加えた $c [x=0, y=1]$ は、点 $(0, 1)$ での c の値を示すこととなる。）

⑤ 方程式の入力

方程式の入力に当たっては、(3)に述べたビルトイン関数に加え、以下の三種類の形式の演算子を用いる。

- $d(\langle \text{変数の代数表現} \rangle, \langle \text{時間変数} \rangle) \dots \dots$ 一階の時間微分演算子
- $d(\langle \text{変数の代数表現} \rangle, \langle \text{位置変数} \rangle) \dots \dots$ 一階の空間微分演算子
- $d(\langle \text{変数の代数表現} \rangle, \langle \text{位置変数} \rangle), \langle \text{位置変数} \rangle) \dots \dots$ 二階の空間微分演算子

従って、例えば、SPADE 方程式

$$d(c, t) = d(c, x, x)$$

は、通常の形式では、

$$\partial c / \partial t = \partial^2 c / \partial x^2$$

を表現することとなる。

代数方程式、微分方程式はいずれも、ベクトルサブスクリプトの指定がない場合には領域全体に適用すべきものとして認識される。これに対して、例えば、

$$c(x=0) = C \text{ STAR}$$

$$d(c(x=0), x) = 0$$

はそれぞれ、

$$c(0, t) = C \text{ STAR}$$

$$\partial x / \partial x |_{x=0} = 0$$

に対応し、環境条件等の特定の位置のみに適用されるものとして認識される。なお、空間微分の演算子は、中心差分に対応した差分演算子であり、上記の方程式を入力した段階でコード内で自動的に差分化が行われる。

(5) 複数のケースに関する計算を一括処理するためのユーティリティプログラム

1.に述べた様に、本研究で開発した解析手法は、感度解析の実施を最大の目的としており、このため必然的に多数回の計算を行うことが求められている。SAPDE では、この様な多数回の解析を一括して処理するためのフロントエンドとして、解析対象とするパラメータ値の組合せを指定することで計算ケース群を定義し、これに対応してSPADE 本体をシークエンシャルに起動するユーティリティプログラムが付加されている。また、このプログラム上で、同時に出力ファイル群の指定を行うことができる。

2.1.3 今後の課題

2.1.2 に述べた改良によって、SPADE は動的解析を実施するための実用的なソフトウェアとなった。今後、この解析を更に体系的、かつ高精度・高効率で実施するためには、以下の様な改良を行うことが望ましい。

(1) 移動/可変グリッド

現在のSPADE は、固定グリッドで偏微分方程式を解法するものであるが、反応フロントの効果が顕著になるような問題を解くためには、このフロントの伝播を検知し各時刻でこれに最も適したグリッドを逐次作成することが必要である。この点についての改良は、2.1.2 に述べたベクトル変数の取扱いを大幅に変更するものとなるが、特に二次元及び三次元の問題を現実的な計算時間内で解くためには必須である。

(2) 有限要素法

既に述べた様に、現在のSPADE は差分法による空間離散化に基づき作成されているが、将来、より複雑な形状を持つ系に適用して行く上では、有限要素法を使用可能なオプションを作成することが望ましい。

(3) 感度解析をより効率的に実施するための改良

現在のSPADE は、2.1.2 に述べた最も初歩的な感度解析を実施するためのフロントエンドのみを有している。今後、より体系的かつ網羅的に感度解析を行うためには、パラメータ値についての統計的サンプリングを行う機能、あるいは、結果の各パラメータについての偏導関数をより明示的に求める機能を付加することが必要である。この点は、(1)及び(2)に比して比較的容易であると共に、実用面での重要性が高いものと考えられる。

2.2 グラフィカルユーザーインターフェイスSAM GUI

グラフィカルユーザーインターフェイスはSPADEの出力を集約することによって以下の二つのレベルでの情報の視覚化を行う。

個々の計算結果の図形化

各変数の空間分布（ある時刻での）や経時変化（ある位置での）をグラフ化する（スナップショットと呼ばれる）と共に、固定されていた変数を動かすことによってアニメーションを作成する。（例えば、ある変数の空間分布が時間と共にどのように変わるかについてのアニメーション）

複数のケース間の比較

複数のケースについての最大核種放出率とパラメータ値との関係等より概括的な情報を集約し、図形化する。

2.2.1 SAM GUIの概要

(1) オブジェクトの種類

SAM GUIでは、その中で取り扱われるオブジェクトを分類し、それぞれの特徴を明確に定義することが必要である。以下の諸概念は、現在のSAM GUIが識別するオブジェクト群である。

① 第一種変数

第一種変数は、ニアフィールドからの核種放出率あるいは濃度分布といった、SPADE計算結果である。

SAM GUIの中では、第一種変数は、サブドメイン（⑤参照）と変数自体の名称を指定することによって定義される。

② 第二種変数

第二種変数は、広義拡散関数等のshadow modelに含まれ、第一種変数の変化

を規定するものである。

③ パラメータ

パラメータは、第一種変数を第二種変数の関数として表現する際の式に含まれるスカラー量の定数である。パラメータはそれ自体の名称によって指定され、その数値は、現在のSAM GUIでは外部ファイルから読み込まれる。

④ 定数

第一種変数を第二種変数の関数として表現する際の式に含まれるスカラー量の定数であり、特定の名称を持たないもの。定数の値は、関数の表現を指定する際に、ユーザーが直接入力する。

⑤ 表現

第一種変数と第二種変数との間の、代数的な関係式。表現は、位置、時間、定数、パラメータ、第一種変数及び第二種変数を含むものである。この表現を指定するに当たっては、SPADEと同じビルトイン関数を用いることができる。

⑥ ザブドメイン

特定の第一種変数群が定義される領域。通常は、母岩、ベントナイト層、オーバーパック等の空間分割がサブドメインとして定義される。

SAM GUIは、上記①～⑥について、ユーザーの指定した情報に基づき、次項に述べる種々の図形出力を作成することができる。

(2) 図形出力

現在のSAM GUIが作成可能な図形出力は、以下の5種類のものである。

① オーバービュー

全体システムのコンポーネント構造を示すと共に、各コンポーネント毎に一つの第一種変数を指定して、系の変化の概要を出力する。この出力はスナップ

ショットとしても、アニメーションとしても可能である。

② 空間分布線グラフ

複数の第一種変数を指定し、これらの空間分布を通常のグラフとして表現する。ある指定時刻のプロファイル（スナップショット）及びプロファイルの経時変化（アニメーション）のいずれも作成することができる。

③ 空間分布面グラフ

複数の第一種変数間の相対的強さを表現する。この表現は、空間分布として表示され、スナップショット及びアニメーションのいずれも作成可能である。

④ 経時変化プロット

ある指定された位置において、第一種変数の経時変化をプロットする。

⑤ 流線プロット

選択された二つの第一種変数の組をそれぞれたて軸、よこ軸とした平面上で、系の状態変化を“流線”として表示する。

2.2.2 今年度の改良点

SAM GUIへの入力ファイルとしては、SPADEの出力が標準的である。これら、第一種変数及び第二種変数共に差分化による格子点の数値であり、従って時間のみの関数となっている。従来のバージョンでは、この様な変数（第一種及び第二種をまとめてモデル変数と呼ぶ）の外部ファイルの読み込みからGUIの使用がはじめられていた。今年度の改良により、SPADE-SAM GUI間のデータ移送が自動化されたことにより、このユーザーの介入したデータのやりとりは必要のないものとなった。具体的には、SPADEを実効する際に<問題名称>.samというファイルを指定することによって、計算結果はSAM GUIによって読み取り可能な形式に変換され、SAM GUIに自動的に転送された後、第一種変数として定義される。そして次に、

- (i) システムの定義用ファイルの作成
- (ii) 出力の関数表現の定義
- (iii) 図形出力の指定

という三つの操作を経て必要なグラフが得られることとなる。

(1) システムの定義

ここでは、全体領域がいくつかのサブドメインに分割され、各サブドメイン間の関係が定義され、更に各サブドメインにおいて第一種変数が同定されたものをシステムと呼ぶ。従って、システムの定義は、各サブドメインの定義、サブドメイン間の関係の定義、第一種変数の同定という三つの操作で為される。

(2) 出力の関数表現の定義

SPADEの出力は、核種濃度等のDAEの解であり、SAM GUIで図形化する情報に一対一対応している訳ではない。従って、SAM GUIに入力する時点で、SPADE出力から図形化したい情報を算出するための関数表現を指定することが必要となる。

(3) 図形出力の指定

図形出力の指定は、前節に述べた5種類の図形出力モードのうち一つを選択し、さらに必要な情報、タイトル、変数名、座標軸のタイプ（リニアか対数か）及びレンジ等、を与えることによって為される。

また、任意の空間分布出力は、簡単な操作で、スナップショットとしてもアニメーションとして表現可能である。

2 参考文献

- [1] 高瀬博康, ニアフィールド環境変化の動的シナリオ解析手法の開発(1),
PNC ZJ 1281 93-008, 1993

- [2] L.Petzold, " A Description of DASSL : A Differential Algebraic System
Solver ", IMACS Transactions on Scientific Computing 1, North Holland,
1983

- [3] C.Gear, " Simultaneous Numerical Solution of Differential Algebraic
Equations ", J.IEEE Transactions on Circuit Theory, 18, pp 89-95, 1971

- [4] G.Byrne and A.Hindmarsh, " A Polyalgorithm for the Numerical Solution of
Differential Algebraic Equations ", ACM Transactions on Mathematical
Software 1, pp 71-96, 1975

3. 動的解析の試行1（決定論的感度解析）

3.1 概要

本章では、1.に述べた広義拡散方程式を用いた解析手法による、感度解析の内容と結果についてまとめる。本章で述べる解析は、1.4.3で作成したモデル群のうち、

- ・従来の評価モデルに対応するもの
- ・擬似コロイドを考慮したモデル
- ・真性コロイドを考慮したモデル
- ・擬似コロイドと真性コロイドの共存を考慮したモデル
- ・人工バリア周辺の水理条件の経時変化を考慮したモデル
- ・人工バリア内の化学条件の不均質性と、経時変化とを考慮したモデル

を用いて行ったものである。また、この試解析では、簡単のため崩壊系列を有さない核分裂生成物を例とした計算を行うこととし、Tc-99を選んだ。

ここでの解析は、広義拡散係数の関数形を具体的FEPsに対応して設定したものであり、個々の解析ケースの物理的意味が明確にされている。この観点から、広義拡散関数の形を統計的にサンプリングした解析と区別するために、以下、上記の解析を決定論的感度解析と呼ぶこととする。

解析の目的は、既に述べた様に、通常のパラメータ値についての感度の定量化に加えて、代替的な前提条件あるいはモデル化オプションを考えた場合の人工バリアからの核種放出率の変動を調べることである。このための手順として、平成三年度報告書〔1〕に用いられていた従来の評価モデルに対応した計算を行い、これを標準ケースとして更に上記の諸事項について個々に変動を与えて多数ケースの計算を行うこととする。この計算結果、標準ケースを中心としてその近傍での諸パラメータ、前提条件変更、モデル変更の感度を分析することが可能となる。なお、より広範な条件での感度解析を行うためには、体系的な感度解析手法（パラメータ値の統計的サンプリング

により応答曲線を明らかにする等)を設定し、これに対応したソフトウェアの改良を行うことが必要となる。この点については、章末に、今後の課題として検討を加えることとする。

3.2 標準ケース

3.2.1 支配方程式

標準ケースは、従来の評価モデルに対応したものであり、以下の仮定に基づき広義拡散関数を設定して解析を行うこととする。

- ・全ての化学反応（溶解／沈殿、吸着／脱着）は瞬時のうちに平衡に到る
- ・吸着は線型のもの（Henry型）である
- ・コロイド等の溶存種以外の可動な物質は存在しない
- ・人工バリア内の化学条件及び周辺の水理条件は空間的に一様である
- ・人工バリア内の化学条件及び周辺の水理条件は時間的に一定である

以上の仮定に対応した広義関数の代替的条件と、広義拡散方程式とを連立する事により、標準ケースについて支配方程式は次のものとなる（1.参照）

$$C_t = \Delta \{ \alpha(\chi, t) C \} - \lambda C \quad (1)$$

$$\alpha(\chi, t) = D_{s,01} / R \quad (C < c^*R \text{ の場合}) \quad (2)$$

$$D_{s,01} c^* / C \quad (C > c^*R \text{ の場合})$$

- 但し、 $\alpha(\chi, t)$: 広義拡散関数
 $D_{s,01}$: 溶存種の拡散係数
 $C(\chi, t)$: Tc-99の全濃度（沈殿、吸着種、及び溶存種）
 R : 遅延係数

c^* : 溶解度

(1)、及び(2)式はオーバーパック腐食生成物層及び緩衝材の全域に適用されるものとする。

3.2.2 解析体系

平成三年度報告書と同様、図3.2.2-1に示す体系を解析対象とする。

この様な解析体系に対応して、支配方程式は円筒座標系で立式することが必要となる。また、後述する人工バリア周辺の水理条件の経時変化と比較可能となる様に、上記報告書とは異なり、人工バリアの周辺には母岩ゆるみ域があることを想定している。

3.2.3 境界条件

(1) 内側境界条件

内側境界条件としては、ガラス固化体領域で溶解限度が成立することから、インベントリが十分な期間にはここでの液相濃度一定とする。

(2) 外側境界条件

上述したゆるみ域の存在と対応して、mixing cell条件を仮定する。

3.2.4 移行パラメータ

平成三年度報告書における解析に含まれるものについては、全て同報告書と同じ数値を用いることとした。他のパラメータについては、以下の仮定を置いた。

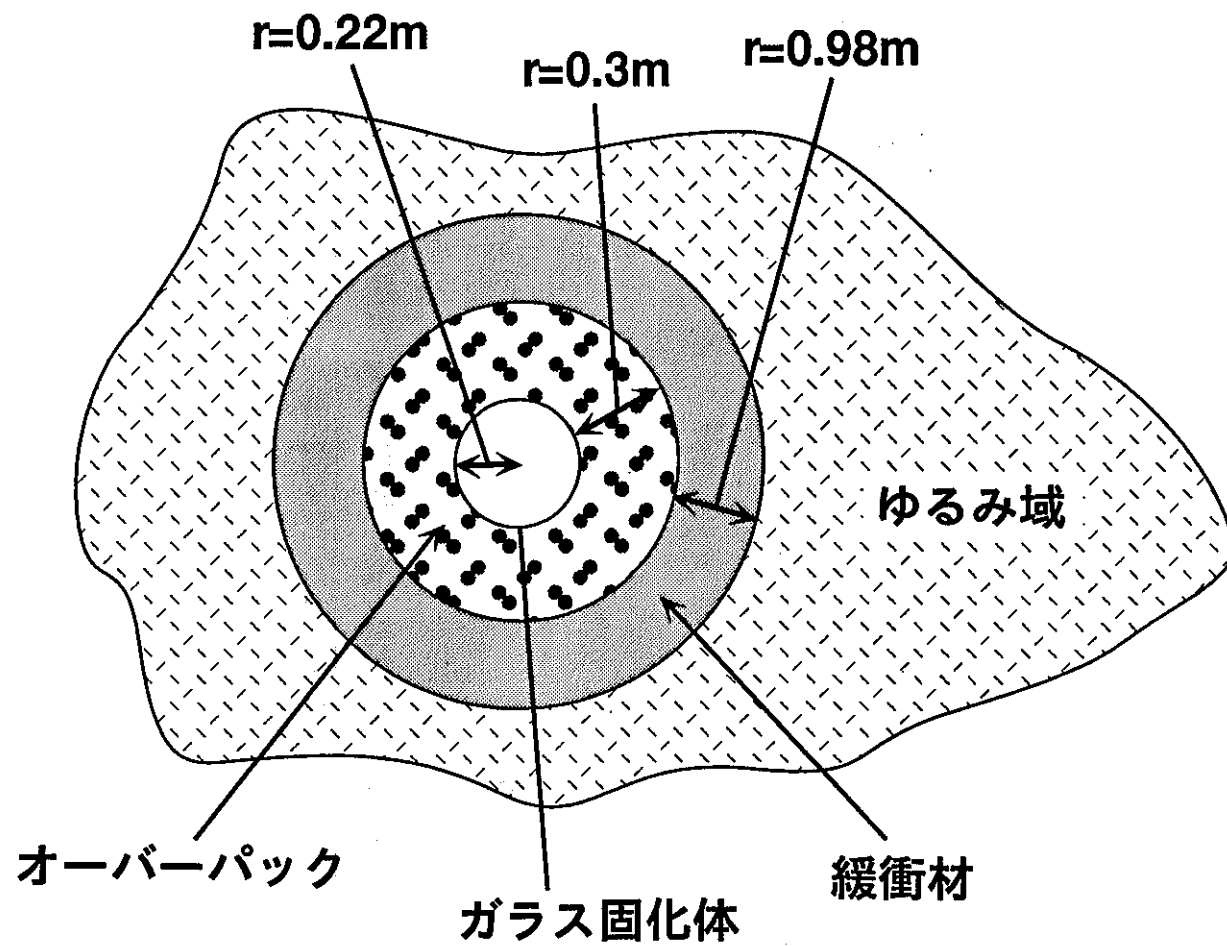


図 3.2.2-1 解析体系

- ・腐食生成物層の吸着は保守的に無視することとした、空隙率及び溶存種の拡散係数としては、緩衝材中と同じ値を用いる。
- ・ゆるみ域の地下水流量としては、スイスの評価例〔2〕で用いられた数値を仮定する。

こうして定められたパラメータ値を表3.2.4-1に示す。

表3.2.4-1 標準ケースでのパラメータ値

パラメータ	設定値
緩衝材空隙率 (-)	0.33
腐食生成物層空隙率 (-)	0.33
緩衝材中遅延係数 (-)	2.8
腐食生成物層中遅延係数 (-)	1
緩衝材中T _c 実効拡散係数 (m ² /s)	3×10 ⁻¹⁰
腐食生成物層中T _c 実効拡散係数 (m ² /s)	3×10 ⁻¹⁰
緩衝材中K _{p c} (-)	0
腐食生成物層中K _{p c} (-)	0
緩衝材中K _{t c} (-)	0
腐食生成物層中K _{t c} (-)	0
ガラス変質層空隙率 (-)	0.33
ガラス変質中T _c 実効拡散係数 (m ² /s)	3×10 ⁻¹⁰
T _c 溶解度 (mol/l)	10 ⁻¹²
ゆるみ域中地下水流量 q (m ³ /yr)	0.71 ×10 ⁻³
T _c 半減期 (年)	2.1 ×10 ⁵

3.2.5 SPADEへの入力

以上述べた標準ケースは、これに対応したSPF入力ファイル (central.SPF) を作成することによってSPADEへと入力される。(付録-5に、標準ケースに対応したSPFファイルを示す)。以下に、その概要を述べる。

SPF入力ファイルは、それぞれ関連する箇所をまとめたいくつかのブロックにより構成されている。以下の説明では、ファイル中の該当部分をブロック番号で参照することとする。

(1) 解析領域及び未知変数 (block 4)

オーバーパック腐食生成物層、及び緩衝材中では広義拡散方程式(1)が偏微分方程式として定義されることから、それぞれの領域についてグリッドを作成する必要がある。また、ガラス固化体、及びゆるみ域では完全混合を仮定するために、これらは単一セルとして表現される。

オーバーパック腐食生成物層と緩衝材領域の差分化により、各セルは円環形状のものとなる。各セルの内、外半径、内、外断面積及びセル内の水容量はそれぞれこの段階で定数として定義されている。各領域での未知数(全濃度 C 、広義拡散関数 α)は格子点毎に定義されたベクトル変数であり、それぞれの領域に対応して C_1 、 C_2 、及び α_1 、 α_2 と指定されている。

(2) 広義拡散関数 (block 6)

広義拡散関数 α_1 、 α_2 についての代数的制約条件(2)式は、それぞれSIDE方程式として定義されている。また、後で述べる代替ケースの計算のために、(2)式はコロイド移動を含めた一般式として表現されているが、標準ケースではコロイド存在比である K_{pc} と K_{tc} とをいずれも0とすることによって、その存在は否定されている。

(3) 広義拡散方程式 (block 7、8)

広義拡散方程式は、各セル間での核種フラックス (ベクトル変数 fluxout1 及び fluxout2) を用いて定義されている。

(4) 内側境界条件 (block 4、7.5)

内側境界条件は、ガラス中の全濃度 Cglass についての S I D E 方程式として block 4 中で定義される。オーバーパック腐食生成物層第一セルへの流入フラックスは、この Cglass に基づき計算される (block 7.5)

(5) 外側境界条件 (block 7.4、3)

外側境界条件は、mixing cell 条件として block 7.4 で定義されている。標準ケースでは、地下水流量 q は定数であるが、block 3 に於ける q に関する表現を用いて経時的变化させることが可能である。

(6) 出力 (block 1、11)

SPADE は自動的に、指定された時刻に於ける全ての変数値をプリントする。出力時刻は、標準ケースにおいては $0 \sim 10^7$ 年間で 10^5 年間隔 (100 点) とされている (block 1)。

さらに、

peak flux . . . 最大放出率

peak time . . . 最大放出率の生ずる時刻

drop time . . . 放出終了時刻

というモニター用の出力がblock11で設定され、これらについても計算終了時にプリントされる。

(7) パラメータ値(block2、3等)

物理定数群についてはblock2及び3において数値が設定され、その他のものは関連する表現が与えられるところ(例えば K_{pc} 、 K_{tc} はblock6に於ける α の表現)でそれぞれ設定される。

3.2.6 結果

標準ケースについての計算結果を、図3.2.6-1(濃度プロファイル)及び図3.2.6-2(核種放出率経時変化)に示す。この結果から、標準ケースにおける解は、従来のモデルによる解と同様、次の様な挙動を示すことがわかる。

(1) 初期の挙動

最初期には各セルでの濃度は溶解度よりも著しく小さく、その後数百年の間に、内側のセルから次第に溶解度近くの定常値へと漸近していく。また、これに対応して、人工バリア外側でのフラックスも立ち上がり始める。

(2) 定常期の挙動

約500年経過時点で系内の濃度プロファイルは定常状態となり、この状態が約 5.5×10^6 年まで継続する。ここでは、系外への流失と系内での崩壊分を補充するだけのフラックスが各セル間を移動していることとなる。なお、この時期でのインベントリの低下は崩壊に支配されたものである。

(3) 最終期の挙動

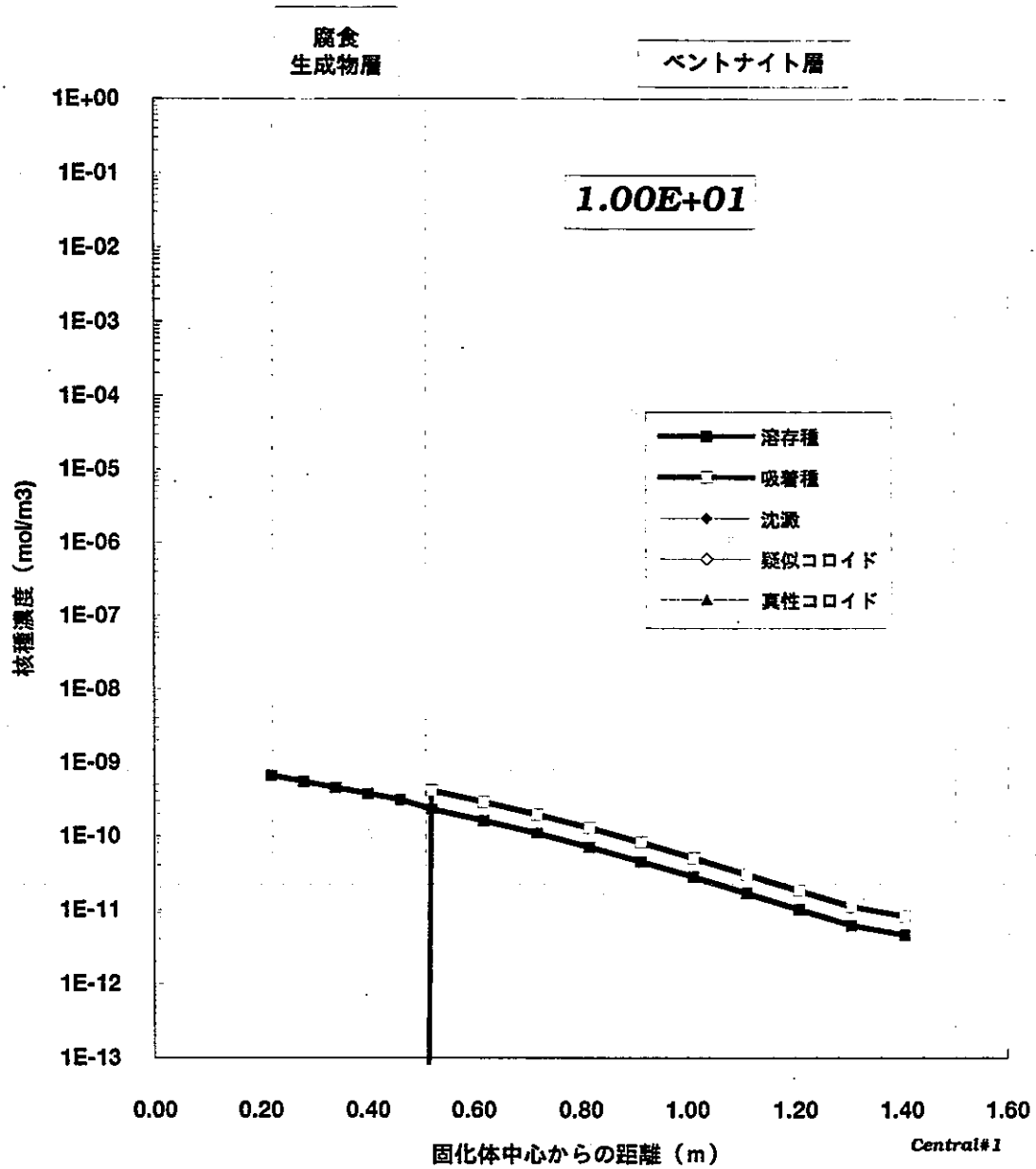


図3.2.6-1 標準ケースにおける濃度プロファイルの例

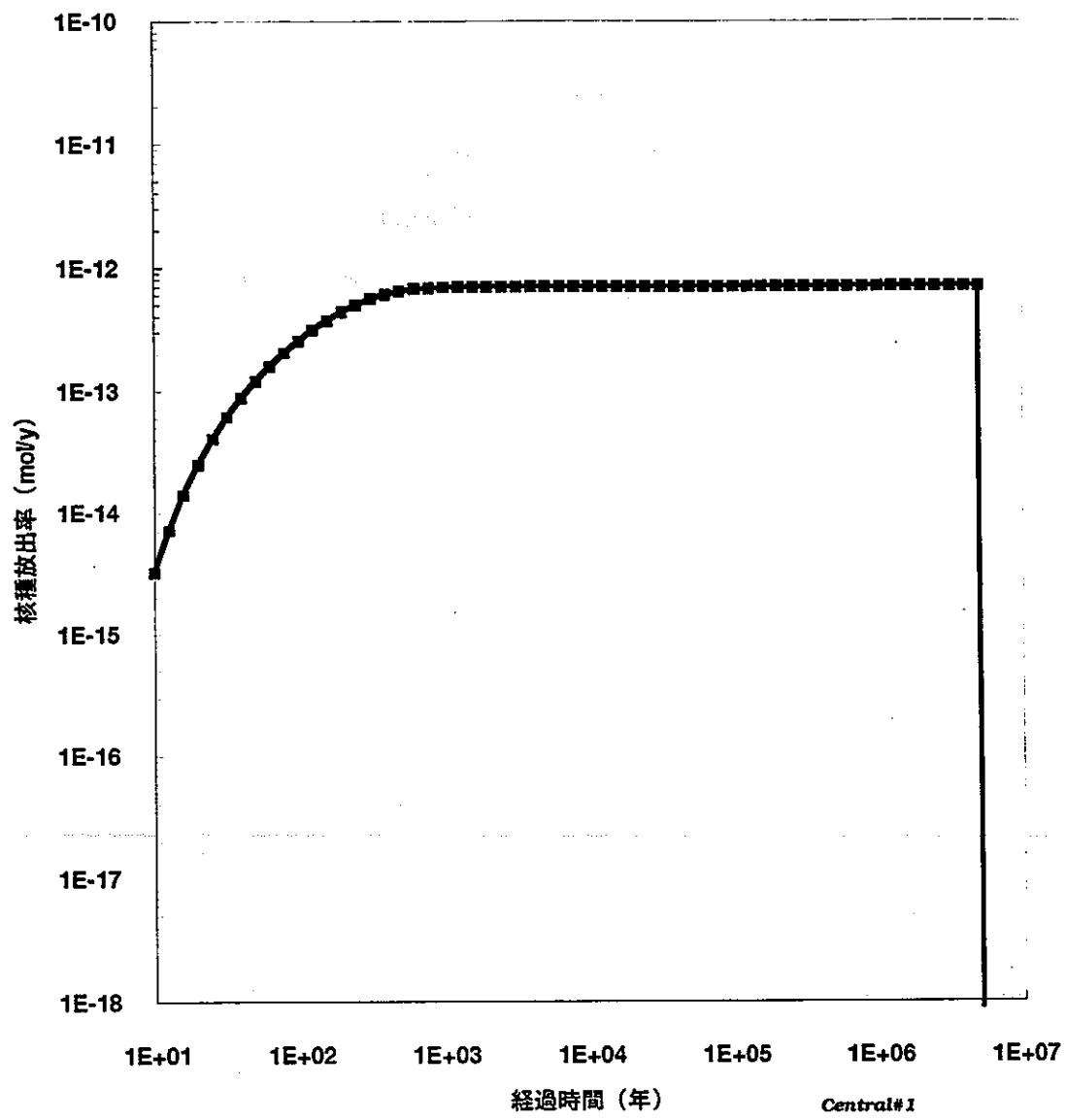


図3.2.6-2 標準ケースにおける核種放出率の経時変化

約5. 5×10^6 年経過時点で、ガラス固化体内に残留するTc-99の総量は、液相中濃度を溶解度に保持できない程度にまで減衰する。そして、これ以降はTc-99の消費は崩壊ではなく、系外への流出によって支配されることとなる。この時期には、各セルでの液相中濃度の低下に伴い、脱着による吸着濃度の低下も生じる。そして、比較的短期間のうちに、核種放出率は無視小へと低下する。(2)の期間を通じての最大放出率は 6.9×10^{-3} mol/年であった。また、標準ケースの計算に要した時間は、Silicon Graphics社のIndigo R4000ワークステーションで15秒(CPUタイム)であり、全タイムステップ数は726である。なお、この計算ではerror toleranceは 10^{-8} と設定した。

また、後述する各モデルについて行う感度解析では、計算ケースがかなり多くなることが予想される。そこで、複数ケースの計算結果を集約して可視化するための手法として、各ケースの計算結果から、

- ・ Peak Flux (最大核種放出率)
- ・ Drop Time (放出終了時刻)
- ・ Peak Outside (人工バリア外に存在する核種量の最大値)

という三つの情報を抽出してケース毎にプロットすることとした。

標準ケースについて行った通常のパラメータ値(溶解度、及び遅延係数)についての感度解析の結果を、上述した手法で整理したものをそれぞれ図3.2.6-3及び図3.2.6-4に示す。

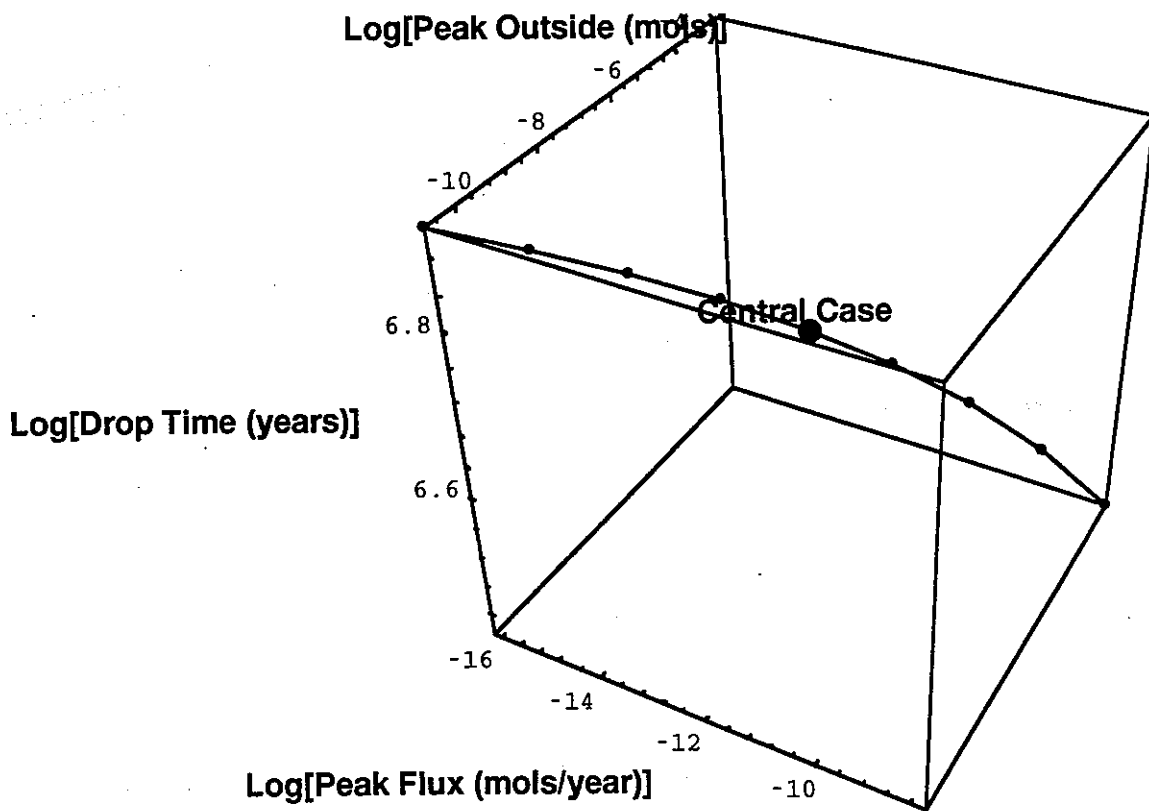


図3.2.6-3 標準ケースについての感度解析の結果（溶解度）

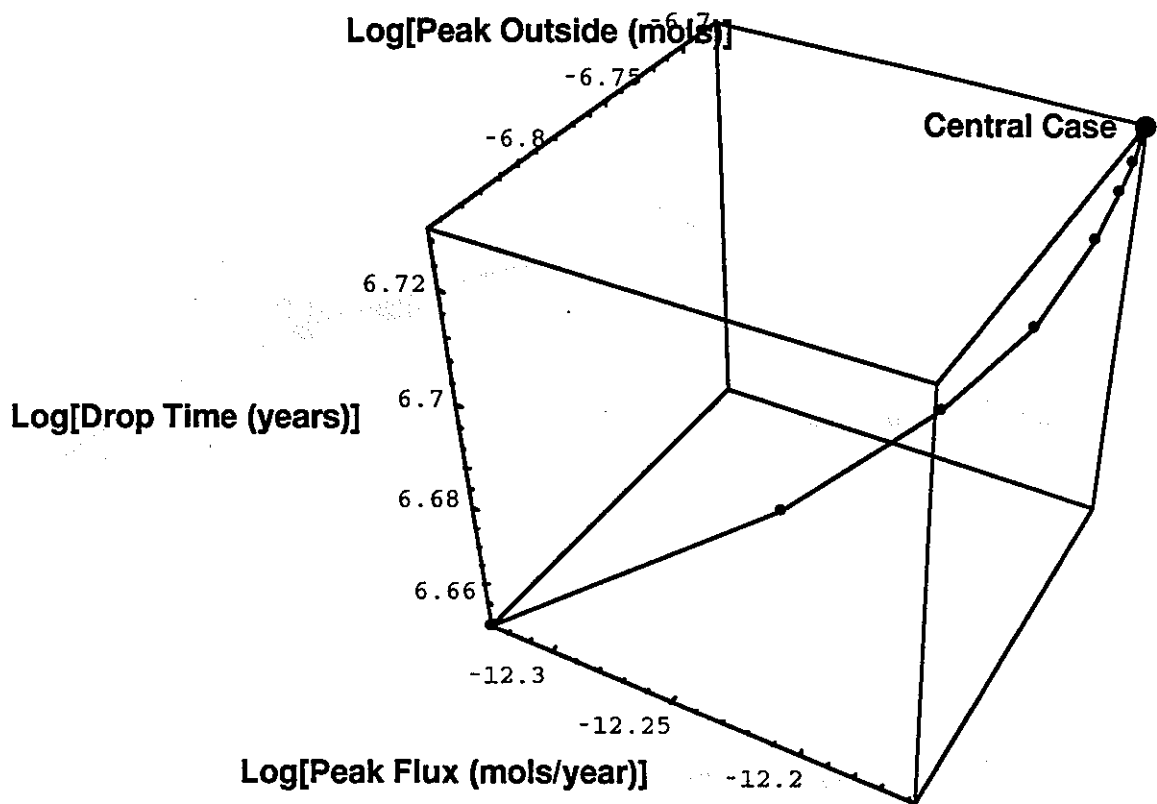


図3.2.6-4 標準ケースについての感度解析の結果 (遅延係数)

3.3 擬似コロイドを考慮したモデル

3.3.1 支配方程式

標準ケースの仮定のうち、核種の移行形態についての仮定を、

- ・核種は、溶存形のものに加え、有機酸等の可動なコロイドに吸着された形態（擬似コロイド）でも移行し得る。

と変更し、かつ1.に述べた様なベントナイトへの吸着種濃度に対して一定の割合 K_{pc} で擬似コロイドが存在すると仮定することにより、本モデルについての支配方程式

$$C_t = \Delta [\alpha(x, t) C] - \lambda C \quad (3)$$

$$\alpha(x, t) = \frac{D_{s01} + K_{pc} (R-1) D_{pc}}{R} \quad (C < c^* R) \quad (4)$$

$$\frac{D_{s01} c^* + K_{pc} D_{pc} (R-1) c^*}{C} \quad (C > c^* R)$$

が導出される。

3.3.2 移行パラメータ

移行パラメータのうち溶存形のものに関しては標準ケースと同一のものを用いることとし、擬似コロイドについては、拡散係数 D_{pc} を $D_{s01}/10$ と設定して試算を行った。

また、 K_{pc} については、

{0.0, 10^{-8} , 10^{-7} , 10^{-6} , 10^{-5} , 10^{-4} , 10^{-3} , 10^{-2} , 10^{-1} , 0.2, 0.5,

1.0 }

という12種類の数値を用いて感度解析を行うこととした。

3.3.3 SPADEへの入力

標準ケースについてのSPADE入力ファイルcentral.spfについて、上記の二つのパラメータ $D_{p,c}$ 及び $K_{p,c}$ の値を変動させる（標準ケースではいずれも0となっている）ことによって本モデルについての入力ファイルを作成した。

3.3.4 結果

擬似コロイドを考慮した場合の計算結果の例（ $K_{p,c}=0.1$ ）を、図3.3.4-1（濃厚プロファイル）及び図3.3.4-2（核種放出率経時変化）に示す。ここでは、 $Tc-99$ が低吸着性であることから擬似コロイドの寄与も小さく、実際、感度解析の結果（図3.3.4-3）を見ても最大核種放出率の標準ケースからの増加は極めて小さい事がわかる。

しかし、吸着性がより大きい元素については、擬似コロイドの影響がより大きくなることが考えられる。図3.3.4-4及び図3.3.4-5は遅延係数を標準ケースの100倍とした場合の計算結果である。ここでは、擬似コロイド濃度が上昇し、核種放出への寄与も増加していることがわかる。

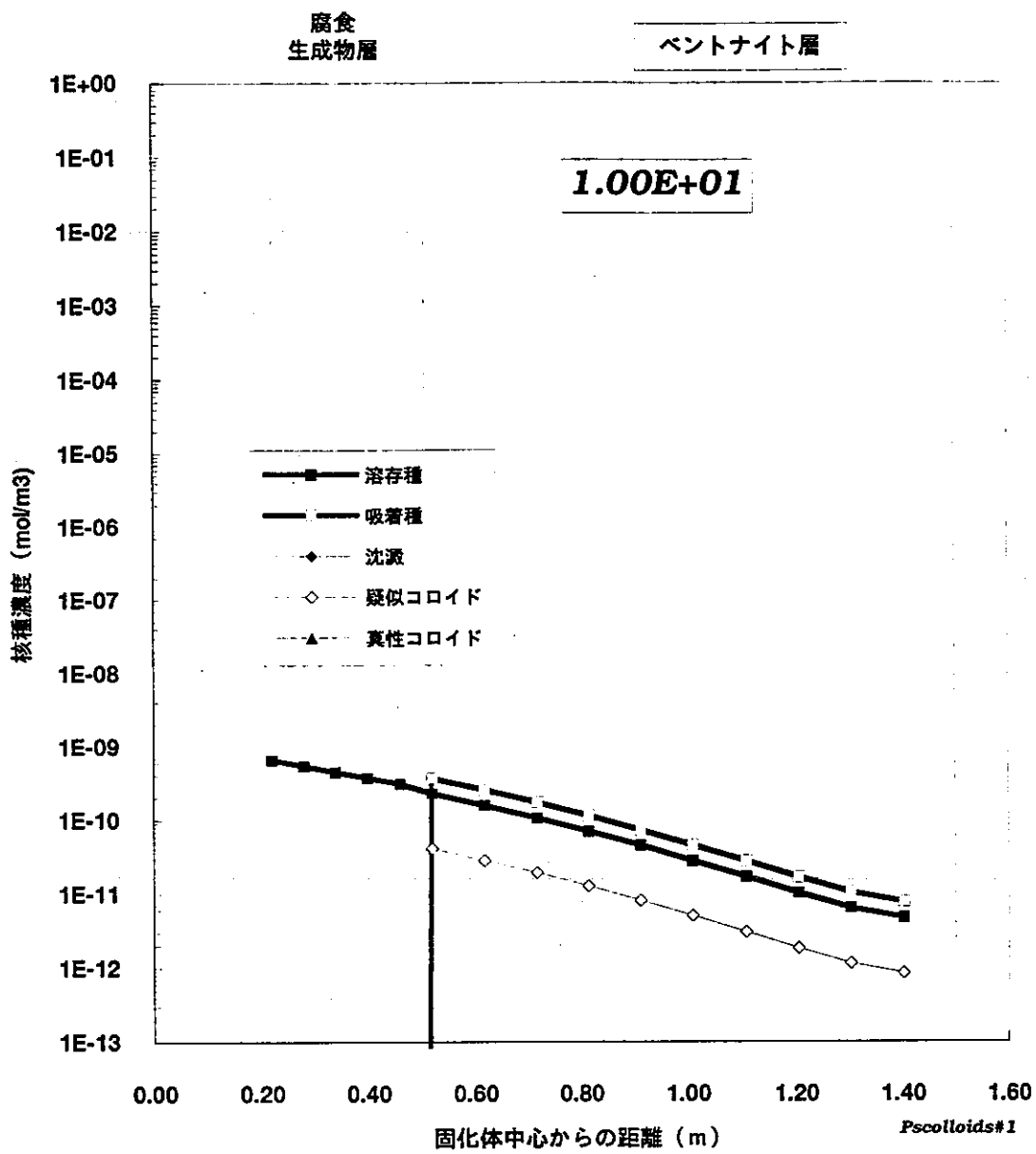


図 3.3.4 - 1 疑似コロイドモデルによる濃度プロファイル算出例
(R=2.8)

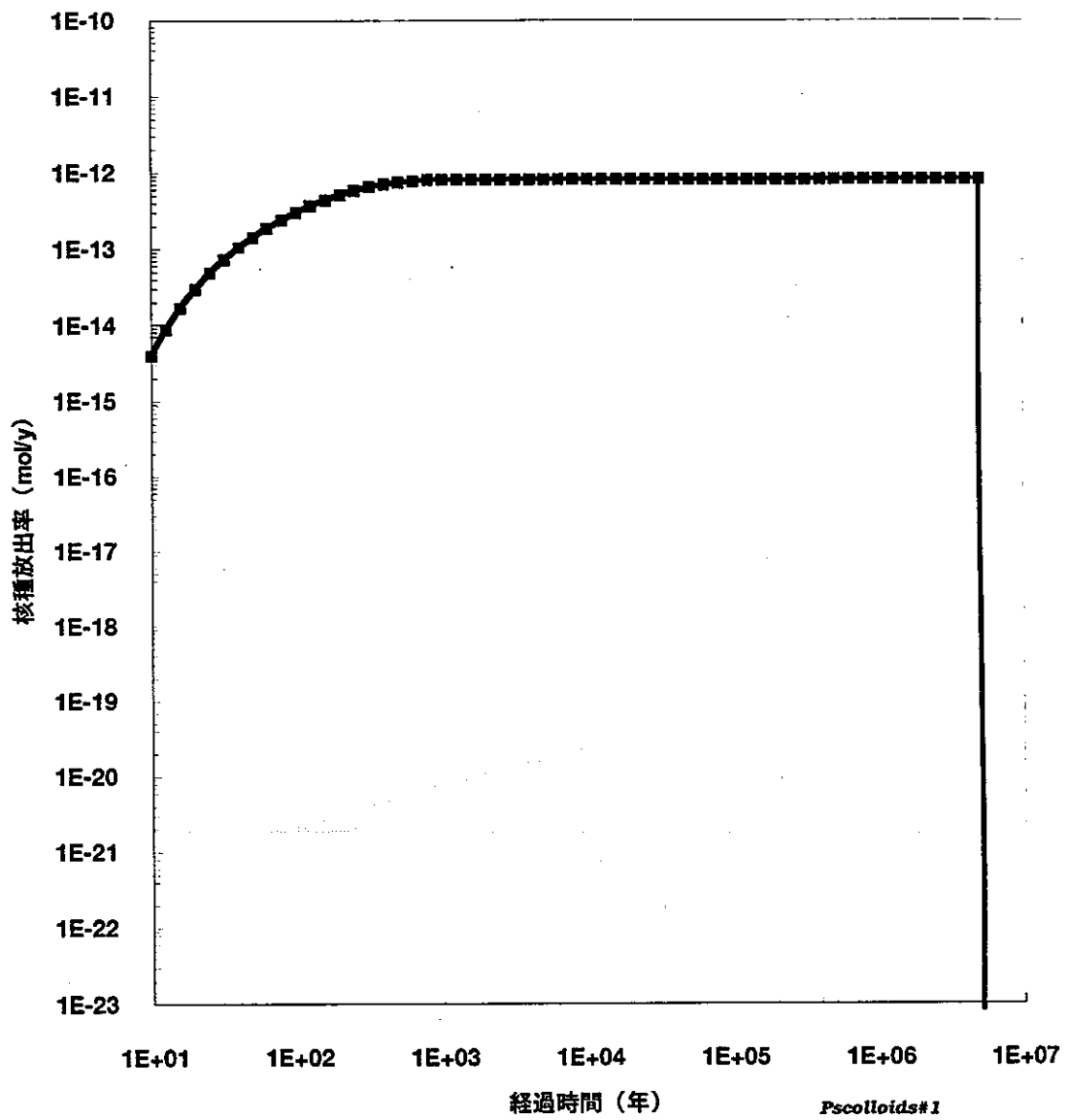


図3.3.4-2 擬似コロイドモデルによる核種放出率算出例
($R=2.8$)

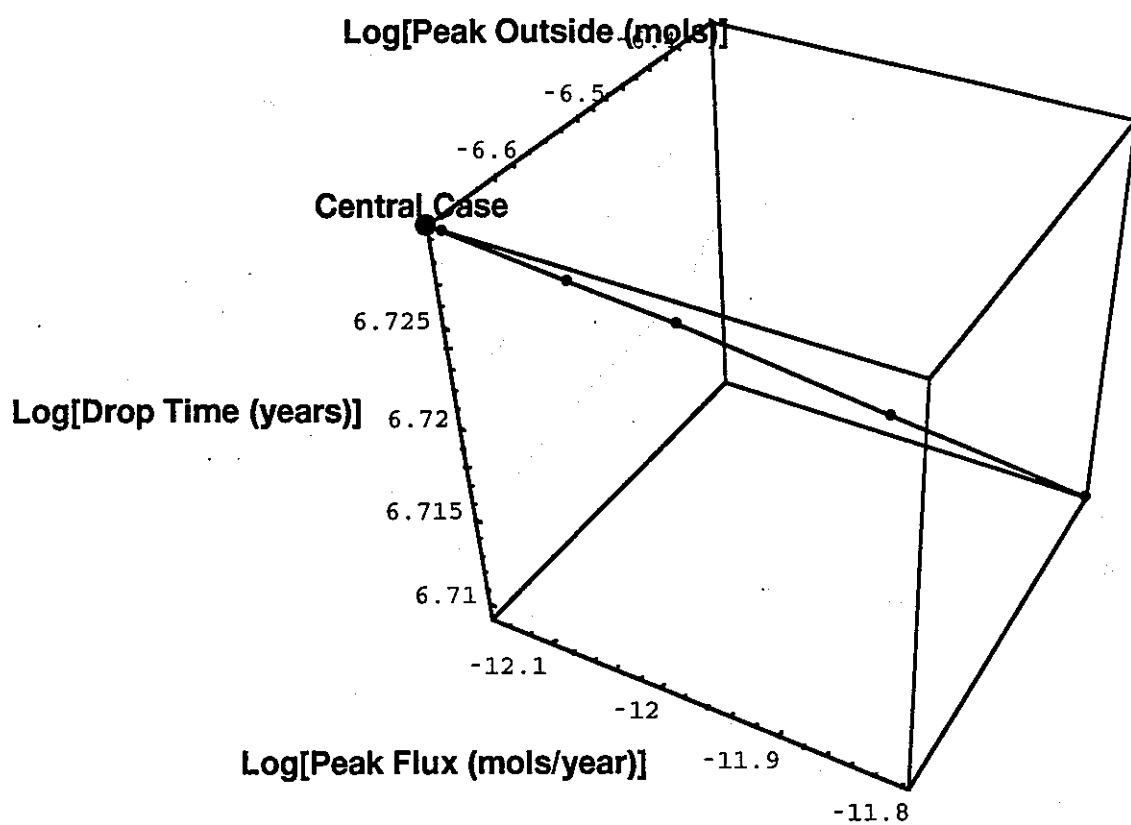


図3.3.4-3 擬似コロイドモデルによる感度解析結果

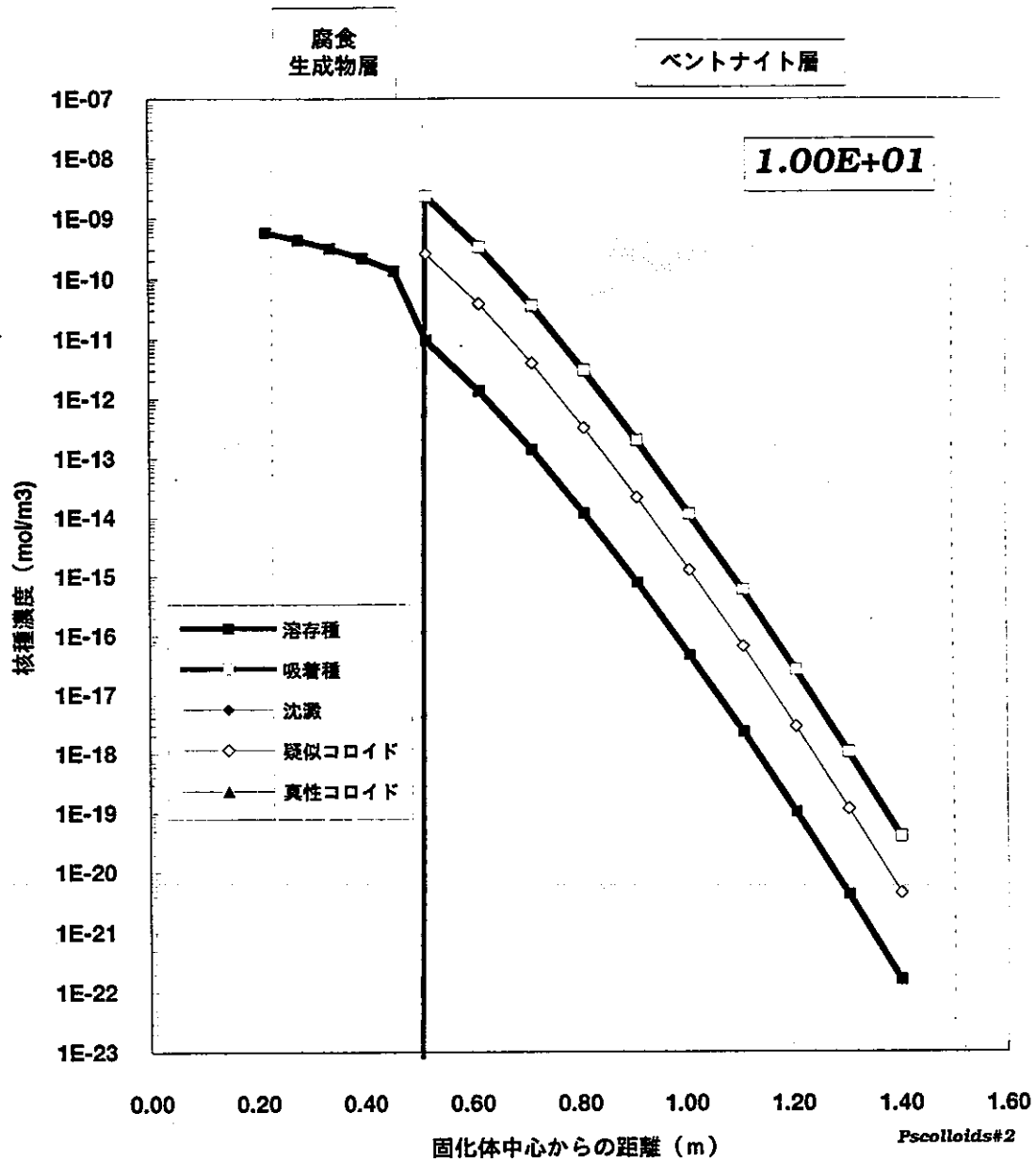


図 3.3.4-4 擬似コロイドモデルによる濃度プロファイル算出例 (R=280)

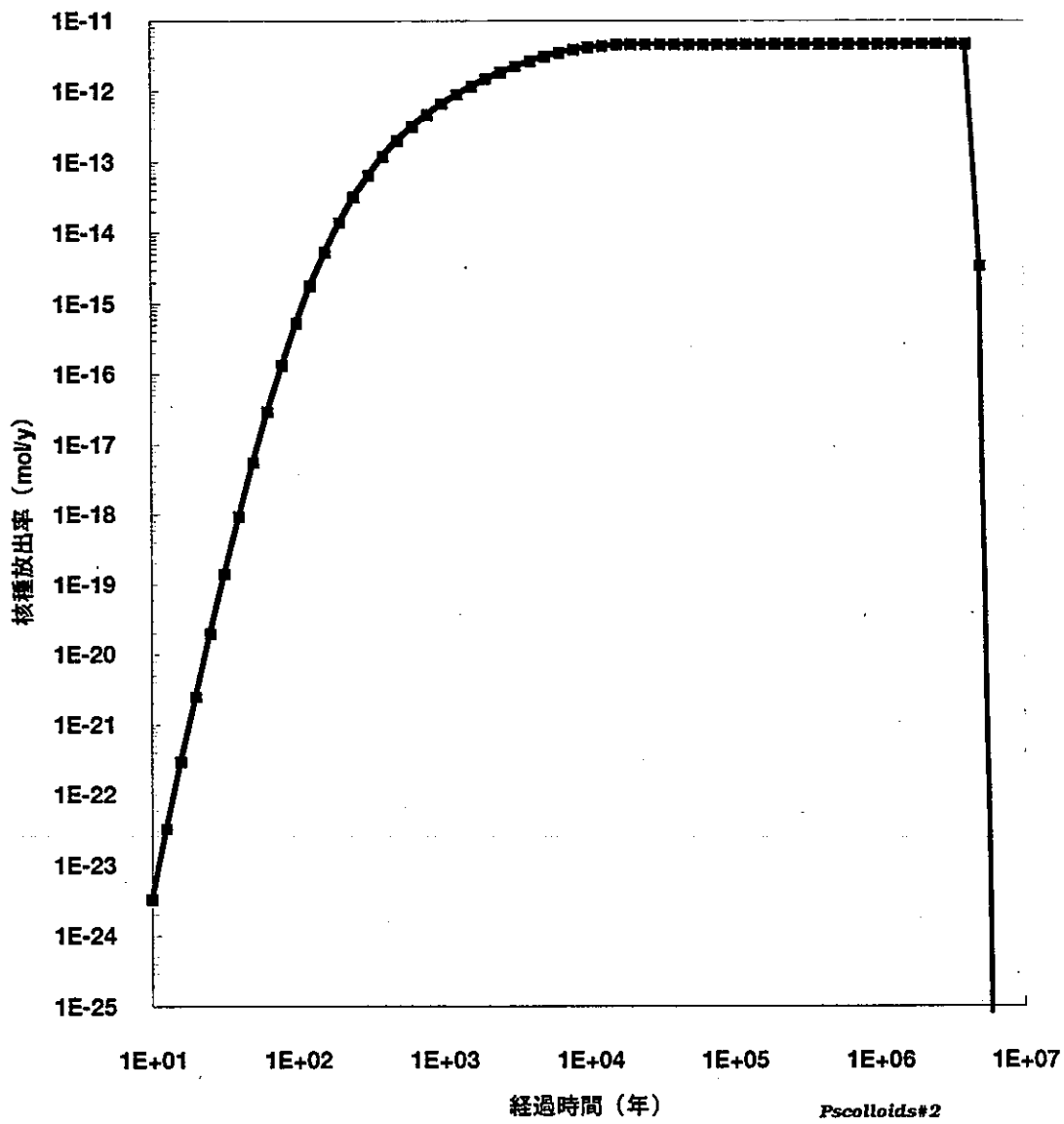


図 3.3.4 - 5 擬似コロイドモデルによる核種放出率算出例 (R=280)

3.4 真性コロイドを考慮したモデル

3.4.1 支配方程式

真性コロイドを考慮したモデルは、1.に述べた様に、沈殿のうちある一定の割合 K_{tc} がコロイドと可動となることを仮定したものであり、この仮定の下に支配方程式は、

$$C_t = \Delta [\alpha(x, t) C] - \lambda C \quad (5)$$

$$\alpha(x, t) = \frac{D_{s,01}}{R} \quad (C < c^*R) \quad (6)$$
$$= \frac{D_{s,01} c^* + K_{tc} D_{tc} (C - c^*R)}{C} \quad (C > c^*R)$$

となる。

3.4.2 移行パラメータ

3.3.2と同じく、真性コロイドの拡散係数 D_{tc} は $D_{s,01}/10$ 一定と仮定し、この条件の下で腐食生成物層中及緩衝材中での K_{tc} をそれぞれ以下の様に変動させて感度解析を行った。

$$K_{tc,1} = \{0.0, 10^{-8}, 10^{-7}, 10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 0.2, 0.5, 0.0, 1.0\} \quad (\text{腐食生成物層中})$$

$$K_{tc,2} = \{0.0, 10^{-8}, 10^{-7}, 10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 0.2, 0.5, 0.0, 1.0\} \quad (\text{緩衝材中})$$

$K_{i,c.1}$ 及び $K_{i,c.2}$ がそれぞれ 12 種類の数値を採ることから、全体としては $12 \times 12 = 144$ の解析ケース数となる。

3.4.3 SPADEへの入力

SPADEへの入力にあたっては、多数ケースの一括処理を必要とするため、2. に述べたフロントエンドのユーティリティプログラムを用いて3.4.2において設定した $K_{i,c.1}$ 及び $K_{i,c.2}$ の 144 通りの組み合わせを用いてシーケンシャルに SPADE を起動するという手法を用いた。

3.4.4 結果

真性コロイドを考慮した場合の計算結果の例 ($K_{i,c.1} = K_{i,c.2} = 0.1$) を、図 3.4.4-1 (濃度プロファイル) 及び図 3.4.4-2 (核種放出率経時変化) に示す。図 3.4.4-2 から明らかな様に、この様な $K_{i,c}$ 値の組み合わせでは最大核種放出率は標準ケースと比較して大きく上昇している。この理由は、図 3.4.4-1 に示されている様に、初期にはガラス固化体中のみ存在している沈殿のうちある割合 (この例では 1/10) が真性コロイドとして外側に拡散し、そこで解離することによって液相濃度は上昇し過飽和となる。そして、溶解度を超過したものの一部は再び真性コロイドとなり、更に外側へと拡散することとなる。この結果として沈殿フロントが徐々に外側へと移動してついにはゆるみ域に到る。この状態では、ゆるみ域中にも真性コロイドが存在することとなり、ここでは、これらは移流によって持ち去られることによって結果として図 3.4.4-2 に示した様な極めて高い核種放出率をもたらされる。

感度解析として実施した 144 ケースの結果を図 3.4.4-3 にまとめる。核種最大放出率は $K_{i,c}$ の設定について極めて大きな感度を有し、最大では標準ケースよりも 9 桁高いものとなっている。また、 $K_{i,c}$ が 10^{-5} より小さい場合には最大放出率は標準ケースから有意な変化を示さないが、 10^{-5} 以上となった場合には 2 桁以上の増大につながる。この事は、仮に真性コロイドがベントナイトによって 99.999% 超過

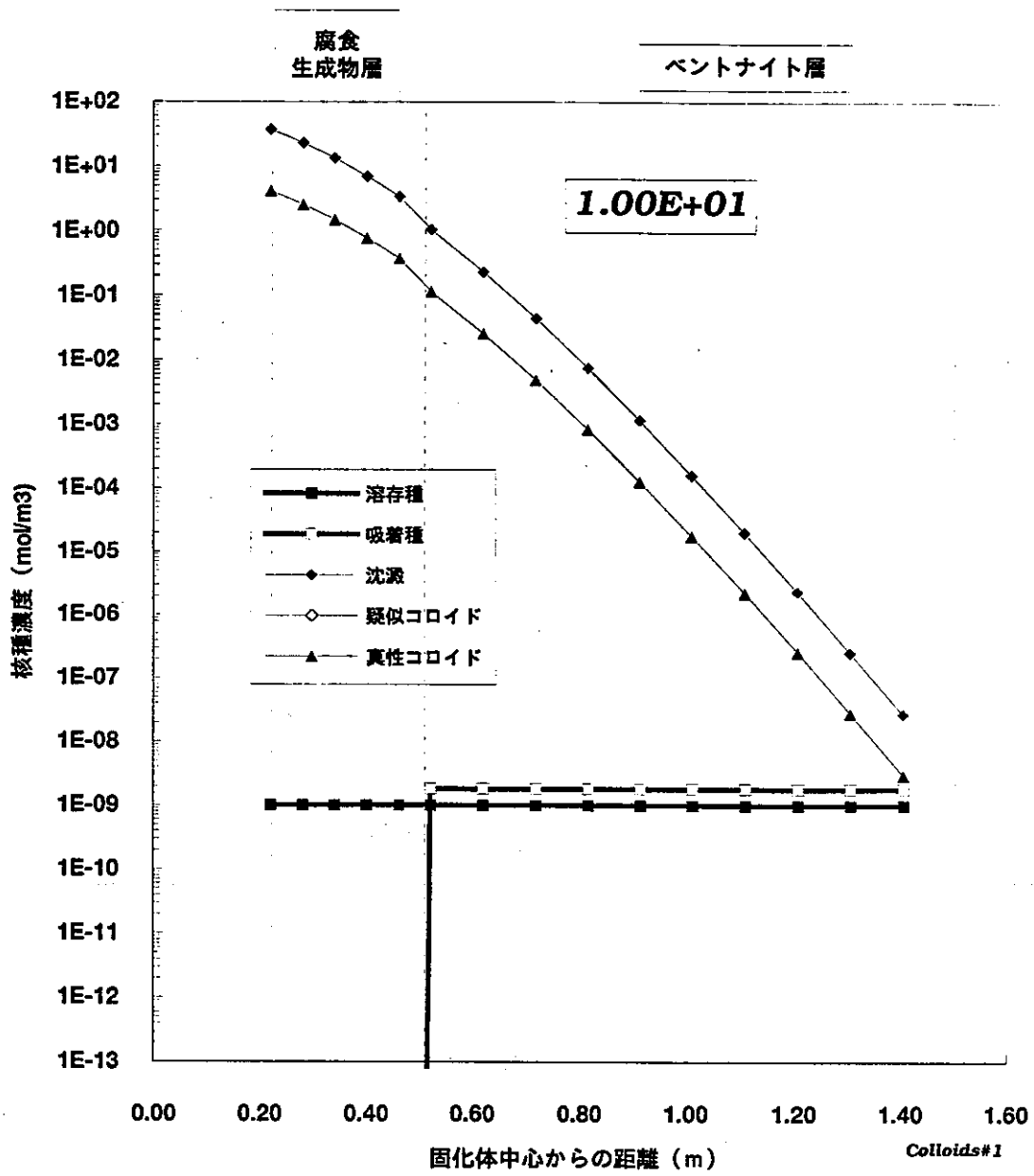


図 3.4.4-1 真性コロイドモデルによる濃度プロファイル算出例

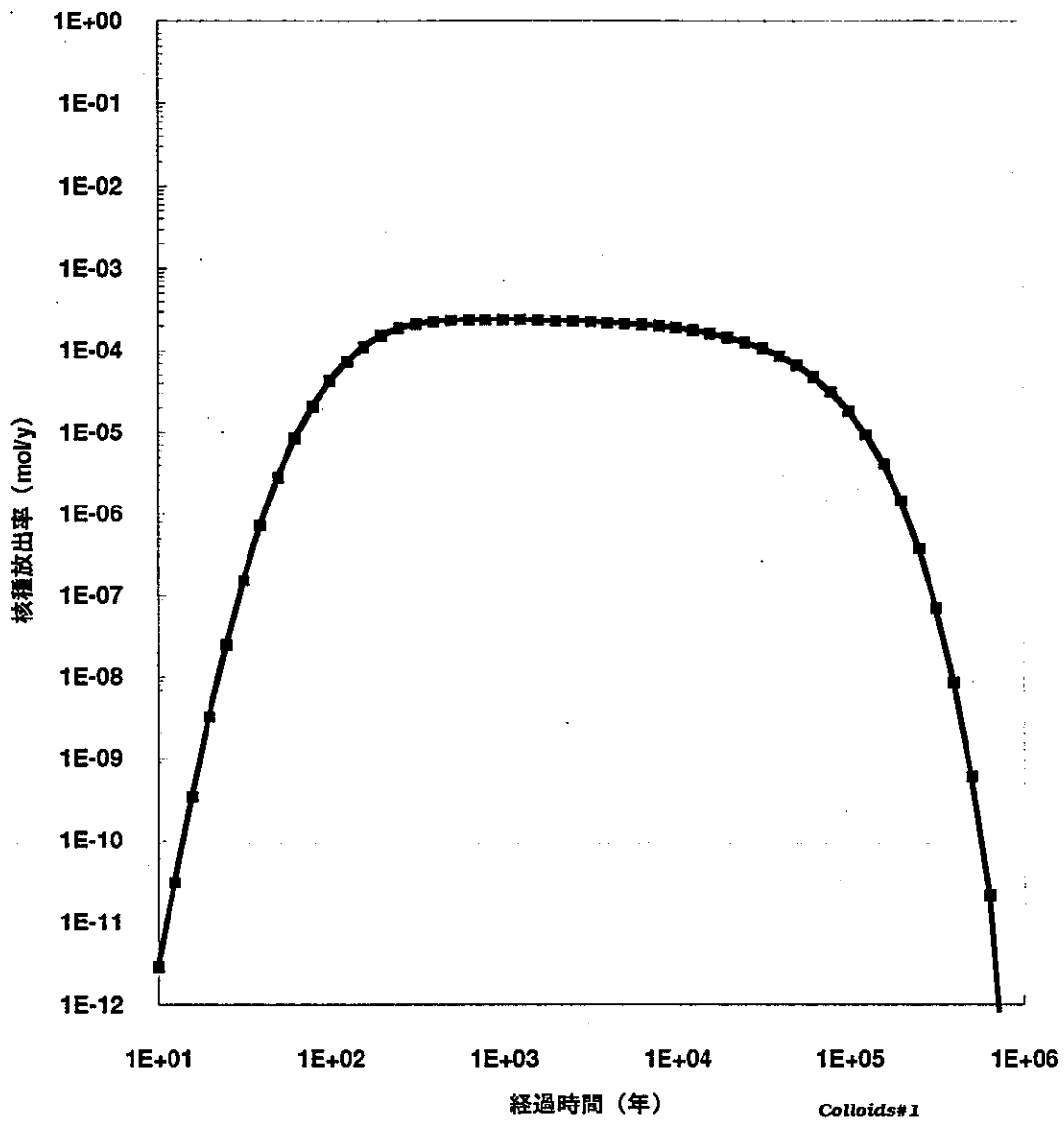


図3.4.4-2 真性コロイドモデルによる核種放出率算出例

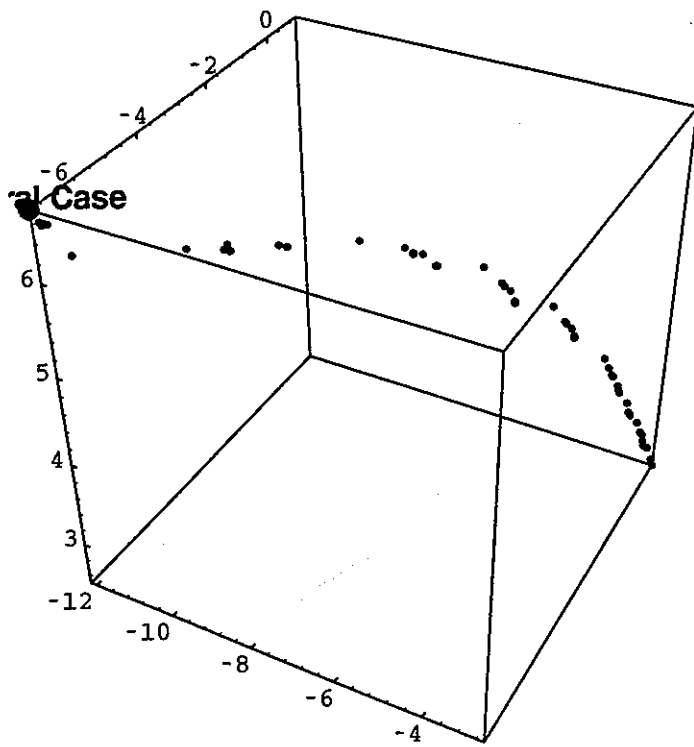


図3.4.4-3 真性コロイドモデルによる感度解析結果

されることが実験的に確かめられたとしても核種移行挙動への影響は否定できないことを意味している。

3.5 擬似コロイドと真性コロイドの共存を考慮したモデル

3.5.1 支配方程式

3.3.1 及び 3.4.1 に述べた支配方程式を併せることにより、これら二種類の共存を仮定した以下の支配方程式が得られる。

$$C_t = \Delta [\alpha(x, t) C] - \lambda C \quad (7)$$

$$\alpha(x, t) = \frac{D_{s,1} + K_{p,c} (R-1) D_{p,c}}{R}, \quad (C < c^*R) \quad (8)$$

$$= \frac{D_{s,1} c^* + K_{p,c} D_{p,c} (R-1) c^* + K_{t,c} D_{t,c} (C - c^*R)}{C}, \quad (C > c^*R)$$

3.5.2 移行パラメータ

ここでは、簡単のため $K_{t,c,1} = K_{t,c,2} = K_{t,c}$ と仮定し、 $K_{t,c}$ 及び $K_{p,c}$ それぞれについて、 $\{0.0, 10^{-8}, 10^{-7}, 10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 0.2, 0.5, 0.0, 1.0\}$ という12種類の数値を設定した合計144ケースの感度解析を行った。

3.5.3 SPADEへの入力

真性コロイド及び擬似コロイドの場合と同様に、フロントエンドのユーティリティプログラムを用いることによって144ケースの計算を一括処理した。

3.5.4 結果

擬似コロイドと真性コロイドが共存する場合の計算結果の例 ($K_{i,c} = K_{p,c} = 0.1$) を図3.5.4-1 (濃度プロファイル) 及び図3.5.4-2 (核種放出率) に示す。

濃度プロファイルは、真性コロイドのみを考えた場合と同様に、沈殿フロントの形成と真性コロイドの拡散を介したその移動が生じていることを示している。また、このことを反映して、核種放出率も標準ケースと比較してかなり大きな数値を示す。

144 ケース全体についての感度解析の結果を図3.5.4-3 にまとめる。この結果は図3.4.4-3 とほぼ同様であり、Tc-99 の様に低吸着性の核種については擬似コロイドの感度は真性コロイドとの比較において相対的に無視し得るものであることがわかる。

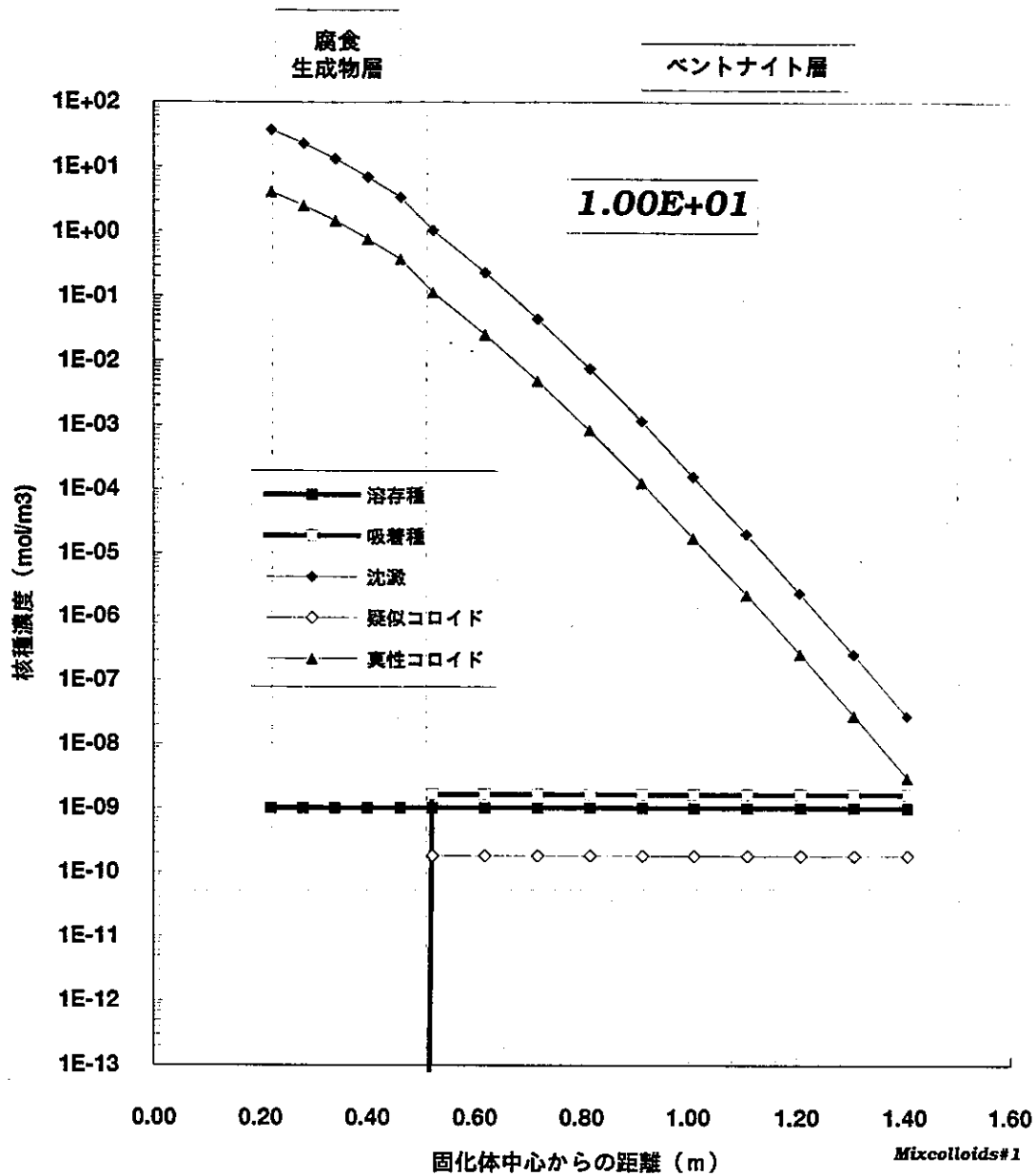


図3.5.4-1 コロイド共存モデルによる濃度プロファイル算出例

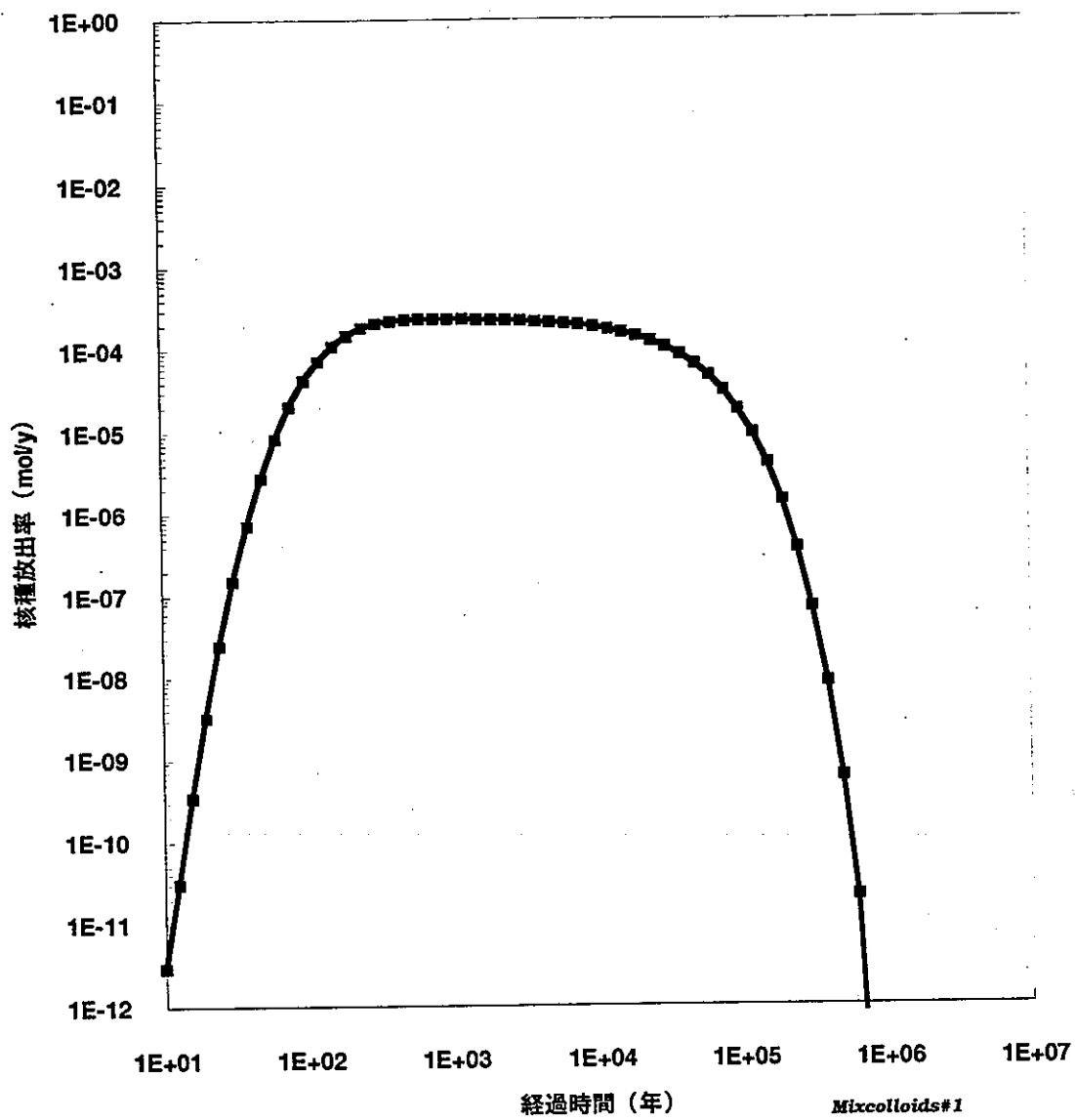


図3.5.4-2 コロイド共存モデルによる核種放出率算出例

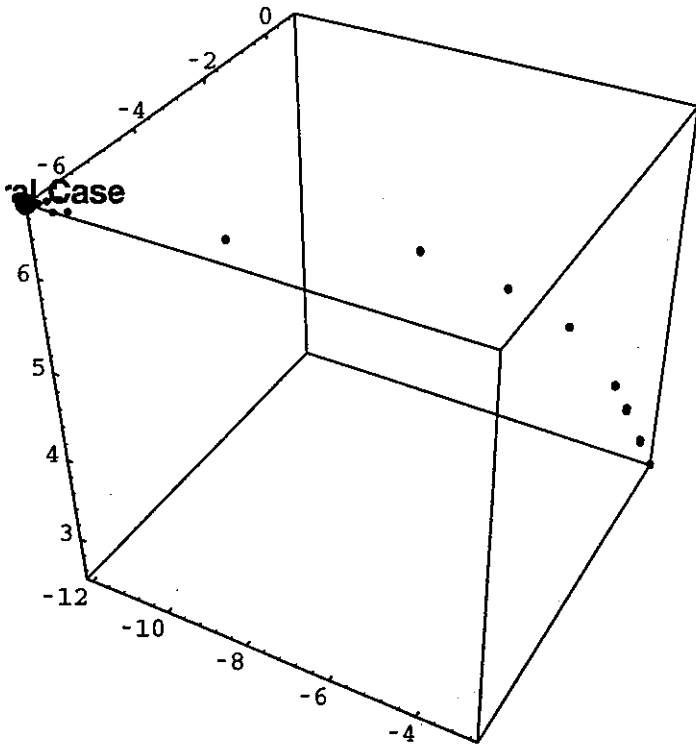


図 3.5.4 - 3 コロイド共存モデルによる感度解析結果

3.6 人工バリア周辺の水理条件の経時変化を考慮したモデル

3.6.1 変化のパターン

本モデルでは、支配方程式については標準ケースと同一のものを用いる。ただし、外側境界条件を規定するゆるみ域中の地下水流量 q については、人工バリア周辺の水理条件が経時的に変化することに対応して時間の関数として表現することが必要となる。ここでは、 q について標準スケールを指定できる関数表現として、次式で与えられる正規分布形の摂動を用いる。

波形関数 $Q(t)$

$$\begin{aligned} Q(t) &= 2t/T, & (0 \leq t \leq T/2) \\ &= -2t/T, & (-T/2 < t < 0) \end{aligned} \quad (9)$$

摂動を考慮したゆるみ域流量 $q(t)$

$$q(t) = q_0 \left(1 + q_1 \exp\left(\frac{-Q(t)^2}{q_w^2}\right) \right) \quad (10)$$

但し q_w : 摂動の時間スケール $q_w = q_1/T$
 q_1 : 摂動の強さ

図3.6.1-1に、周期 $T=30000$ 年、摂動の強さ $q_1=10^4$ 、つまり30000年周期でゆるみ域流量が 10^4 倍となる様な摂動を与えた場合の、種々の q_1 値に対する $q(t)$ の時間変化の例を示す。 q_1 値が小さい時には摂動の時間スケールは小さく、スパイク状の変化が現れている。これに対して q_w を除々に増加すると変化の幅は大きくなり、ついには前回の摂動とつながる様になっている。

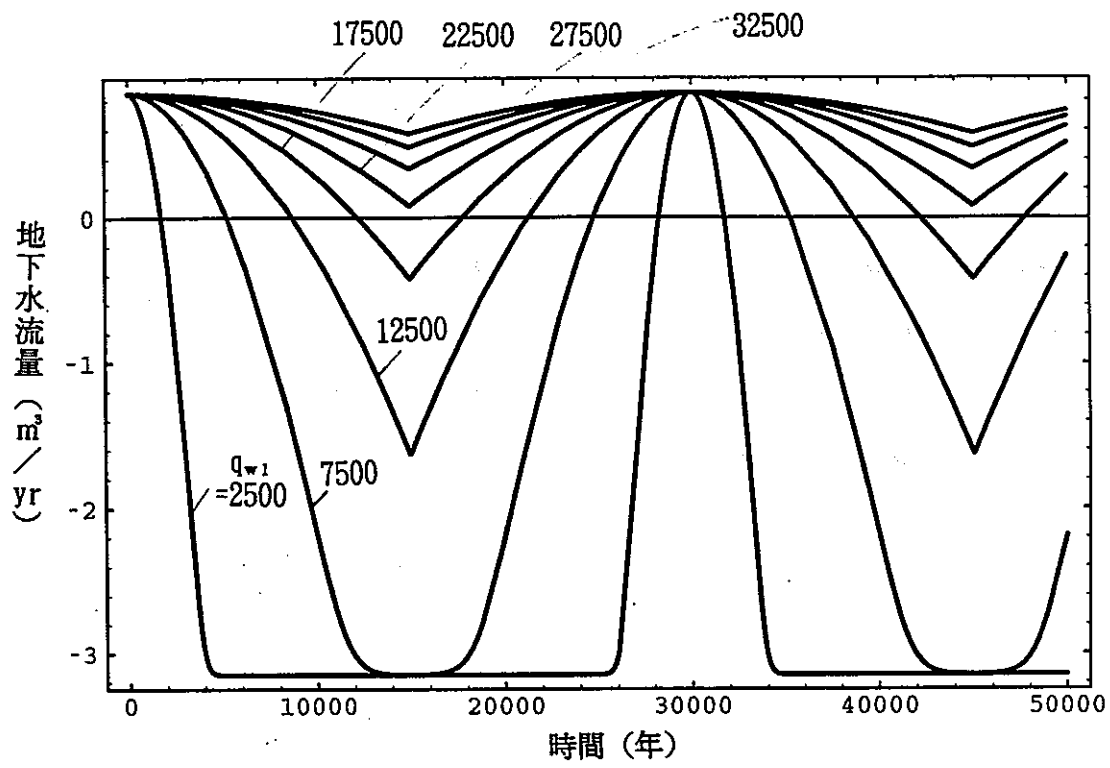


図 3.6.1 - 1 水理条件の時間的変化パターン

3.6.2 ケース設定

ここでは、(10)式の $q(t)$ に含まれるパラメータをそれぞれ以下の様に設定した。

$$q_0 = 0.71 \times 10^{-3} \quad (\text{m}^3/\text{年}) \quad (\text{標準ケースと同じ})$$

$$q_1 = \{1.0, 10.0, 10^2, 10^3, 10^4\}$$

$$q_{w1} = \{2500, 5000, 7500, 10000, 15000, 30000, 50000, 100000\}$$

また、摂動の周期 T については、1. で抽出した F E P s のうち気候変化に対応して 30,000 年とした。この様な設定に基づき、 $5 \times 8 = 40$ ケースについての感度解析を実施した。

3.6.3 結果

計算結果である核種放出率の経時変化の例 ($q_1 = 10^4$ 、 $q_{w1} = 10^4, 10^5$) をそれぞれ、図 3.6.3-1 及び図 3.6.3-2 に示す。摂動の時間スケールが小さい場合には核種放出率に周期的な変動が現れるが、スケールが大きい場合には複数の摂動が重複することに伴って放出率の変動も小さくなっている。

40 ケースの感度解析の結果を図 3.6.3-3 にまとめる。ここでは、3.6.2 に示したパラメータ値の大きな変動にもかかわらず結果として生ずる核種放出率の変化は 1.5 桁程と小さく、水理条件が変化する効果は、それ自体としては余り大きな感度を有さないことがわかる。

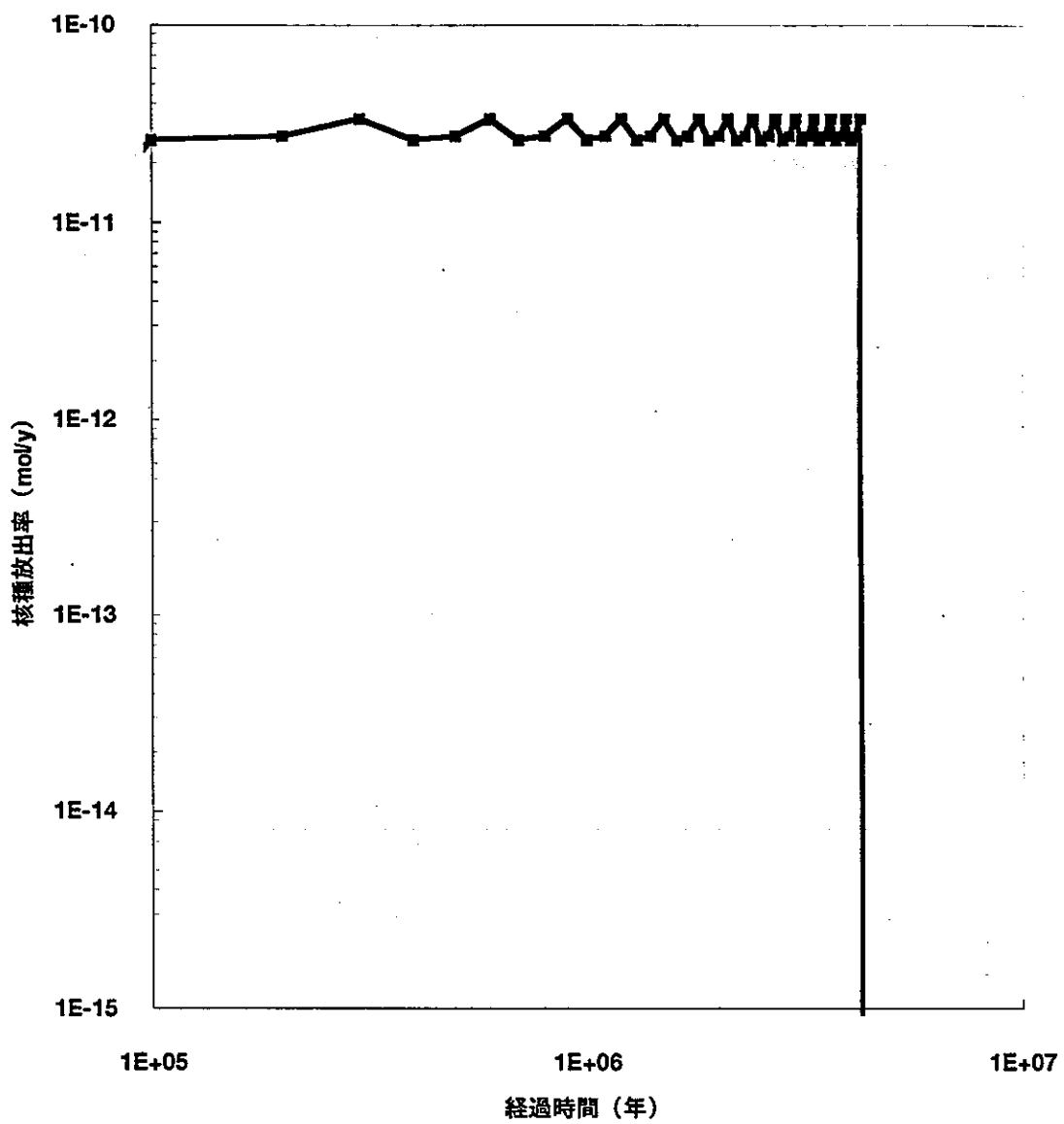


図3.6.3-1 経時的な水理条件変化モデルによる核種放出率算出例
 ($q_{w1} = 10^4$)

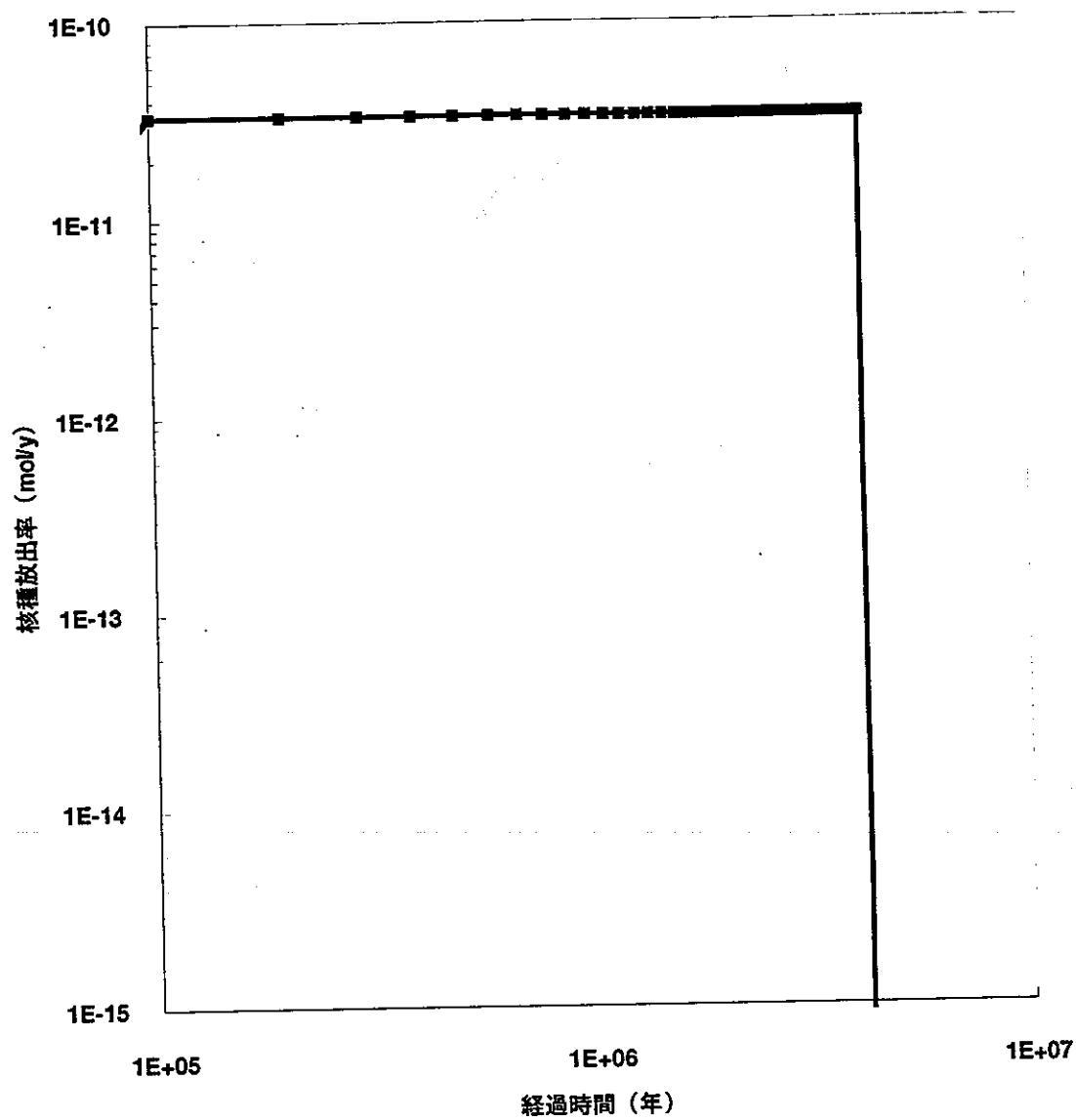


図 3. 6. 3 - 2 経時的な水理条件変化モデルによる核種放出率算出例
 ($q_{w1} = 10^5$)

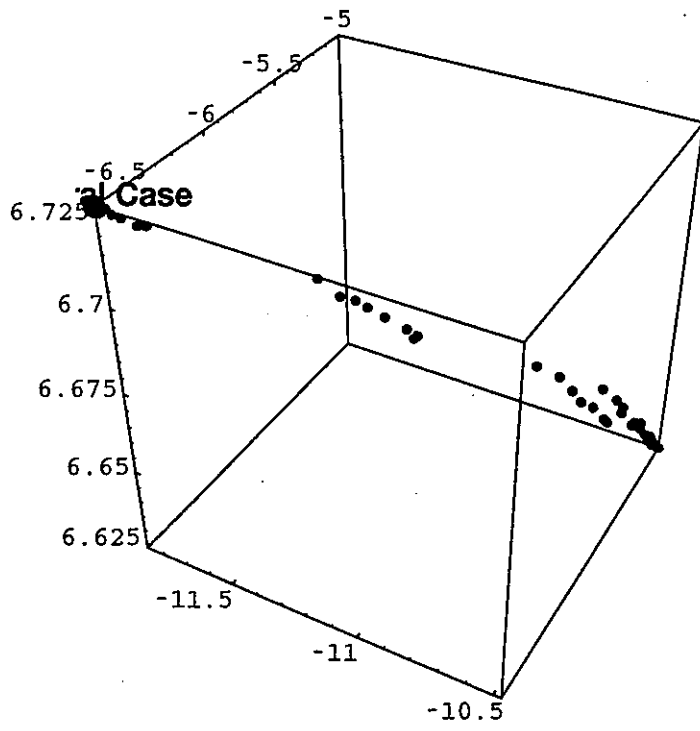


図3.6.3-3 経時的水理条件変化モデルによる感度解析結果

3.7. 人工バリア内の化学条件の時間・空間的变化

3.7.1 空間的に一様な変化

(1) 変化のパターン

3.6 に述べたものと同じく正規分布形の摂動を考慮して、化学条件の変化に対応した溶解度 c^* を時間の関数として次の様に表現する。

$$c^*(t) = c^*_0 \left(1 + 10^8 \exp \left(-\frac{Q(t)^2}{q_w^2} \right) \right) \quad (11)$$

この場合の溶解度の時間変化のパターンは、図3.7.1-1に示すものとなる。

(2) ケース設定

ここでは、1.に述べたFEPsのうち海面変動に着目することとし、Milankovitchのサイクルに合わせて $T=10^5$ 年と設定した。 q_w については、

$$q_w = \{1000, 2500, 5000, 7500, 10000, 15000, 30000, 50000, 100000\}$$

という9種類の数値を考えることとする。

(3) 結果

解析結果である濃度プロファイル及び核種放出率経時変化の例 ($q_w = 10^4, 10^5$) を図3.7.1-2～図3.7.1-5にそれぞれ示す。水理条件の場合と同じく、 q_w が小さい場合にのみ核種放出率の周期的変化が見られる。なお、ここでは溶解度が低下して行く時期に当たっているため、全域で沈殿が生じている。

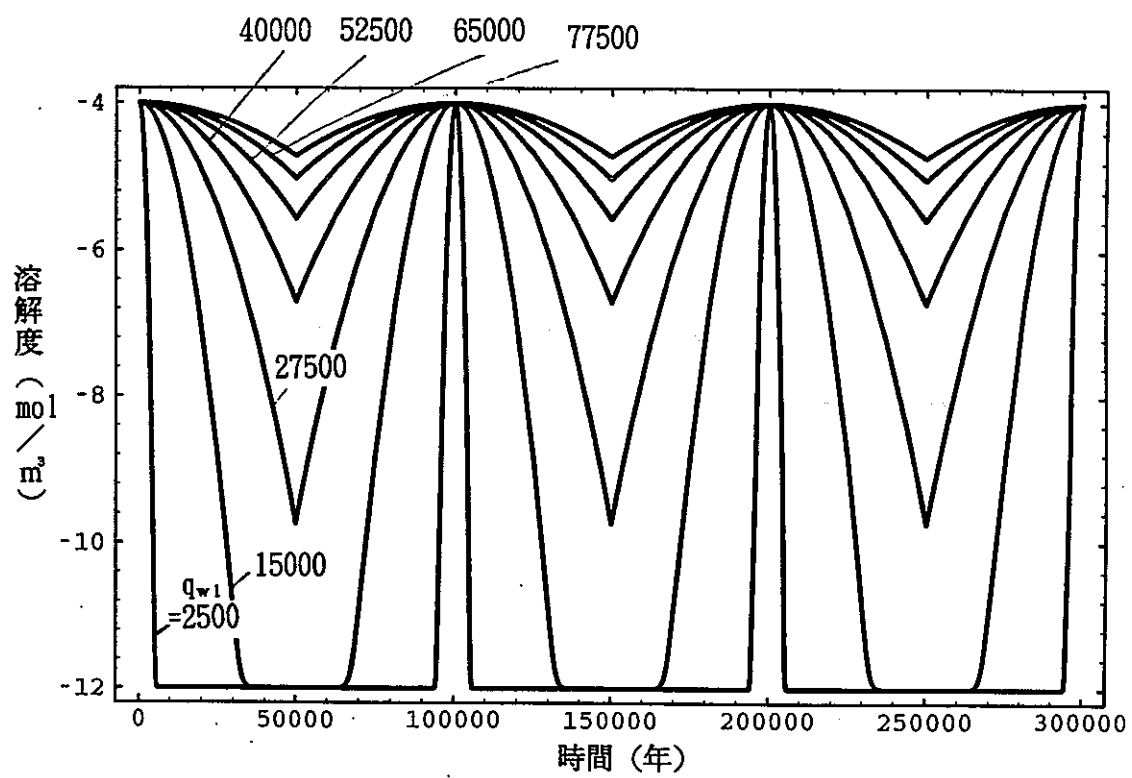


図3.7.1-1 化学条件変化による溶解度時間変化のパターン

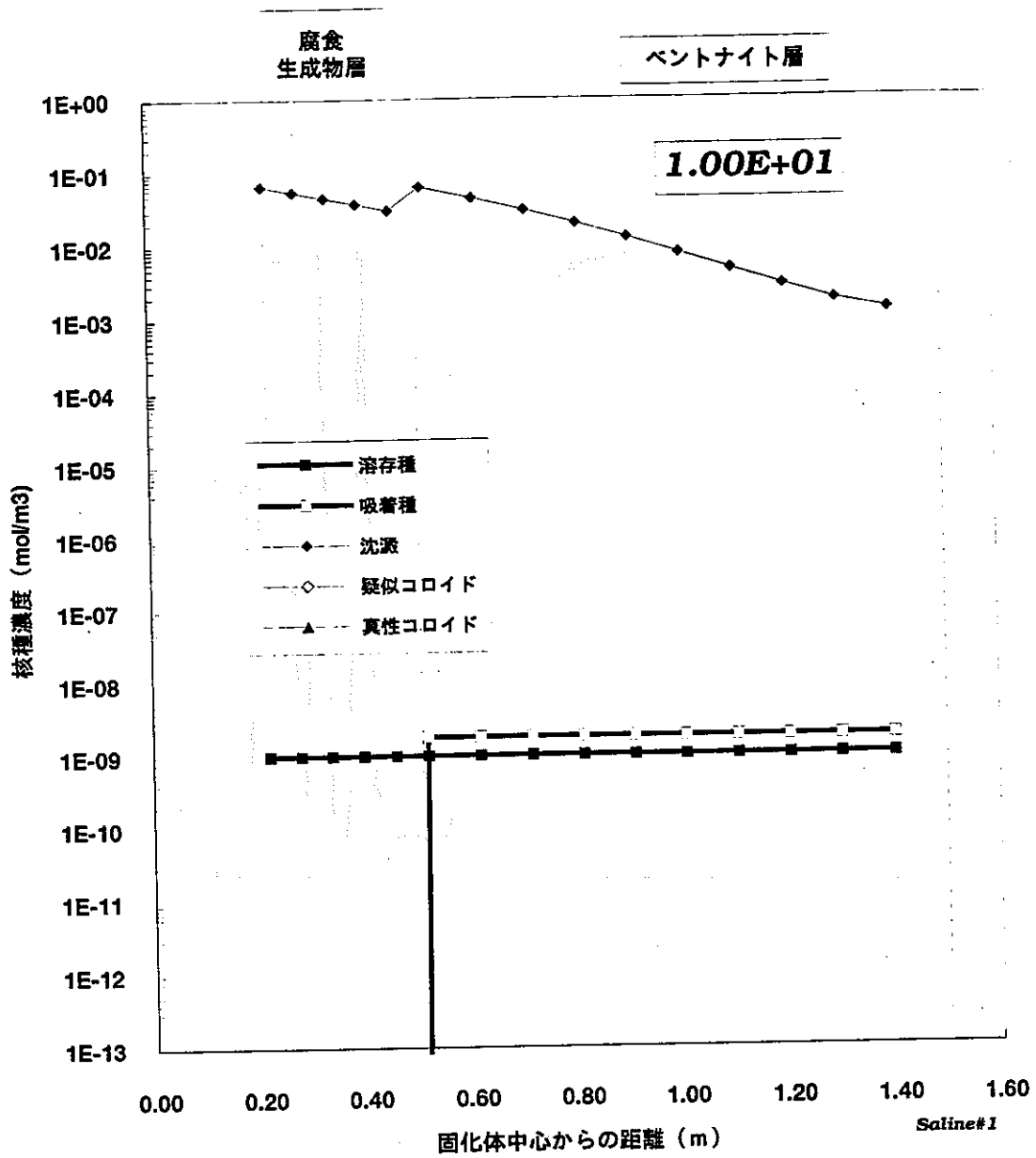


図 3.7.1 - 2 経時的化学条件変化モデルによる濃度プロファイル算出例
($q_{w1} = 10^4$)

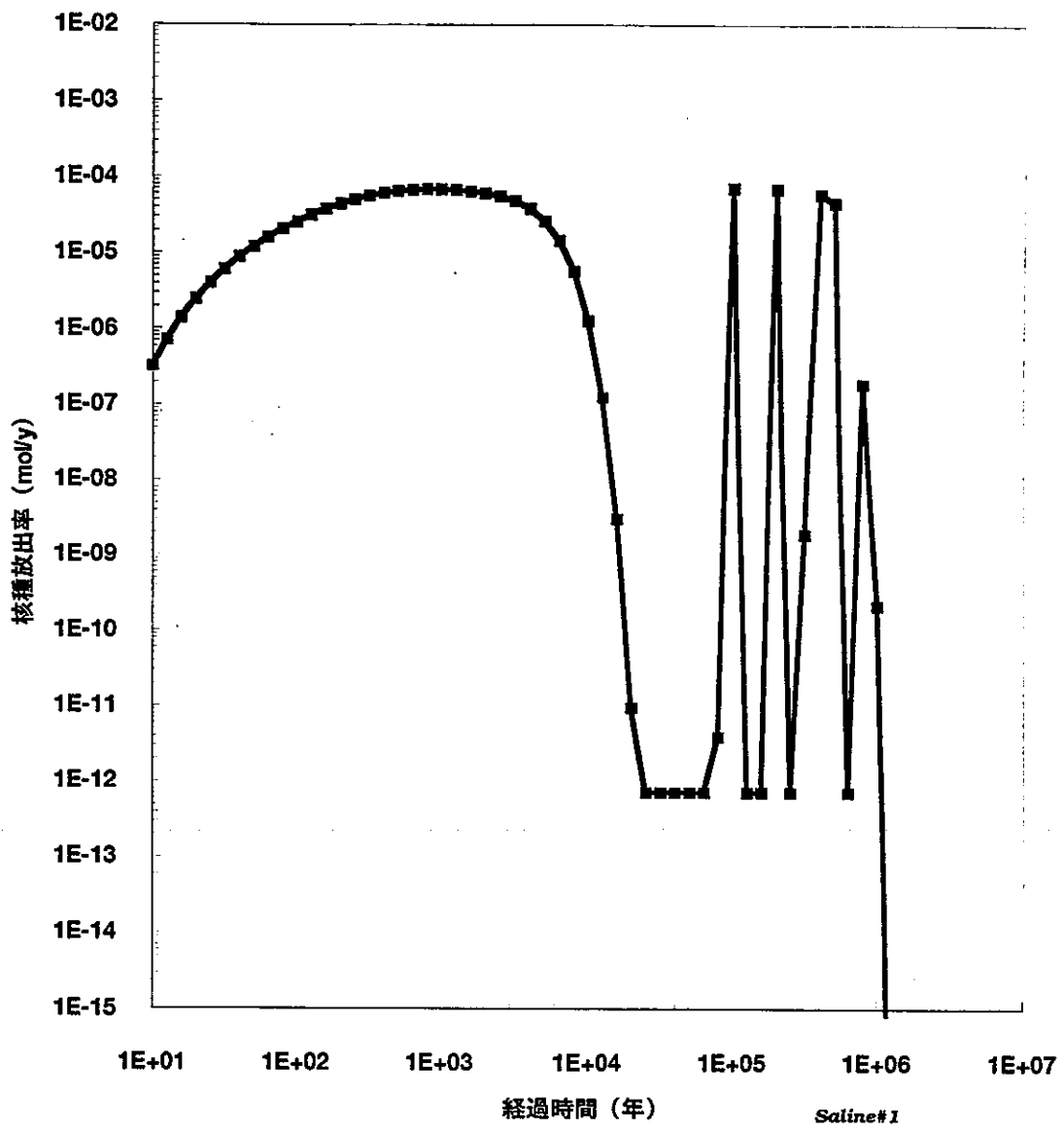


図3.7.1-3 経時的化学条件変化モデルによる核種放出率算出例
($q_{wi}=10^4$)

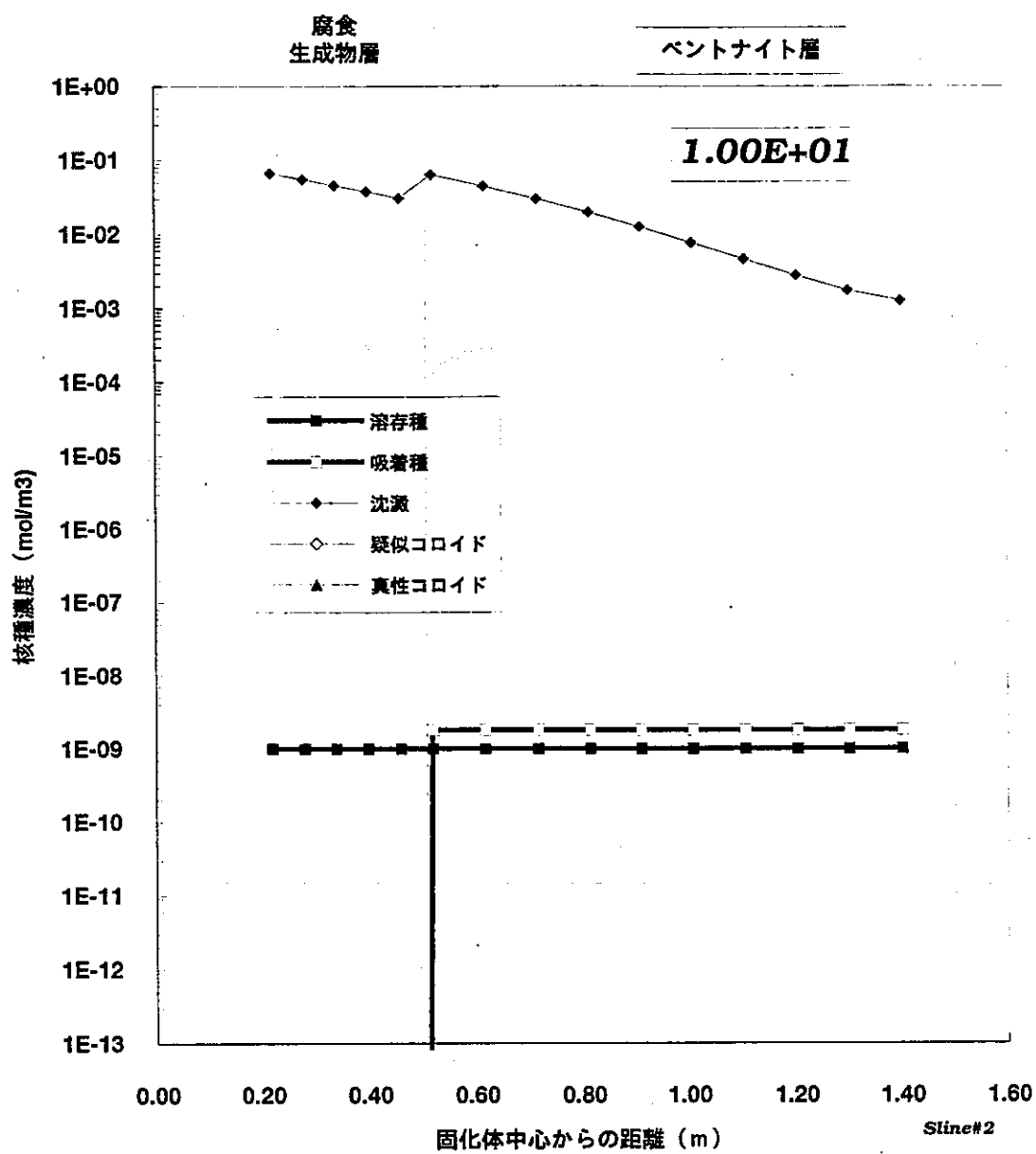


図 3.7.1-4 経時的化学条件変化モデルによる濃度プロファイル算出例
($q_{wi}=10^5$)

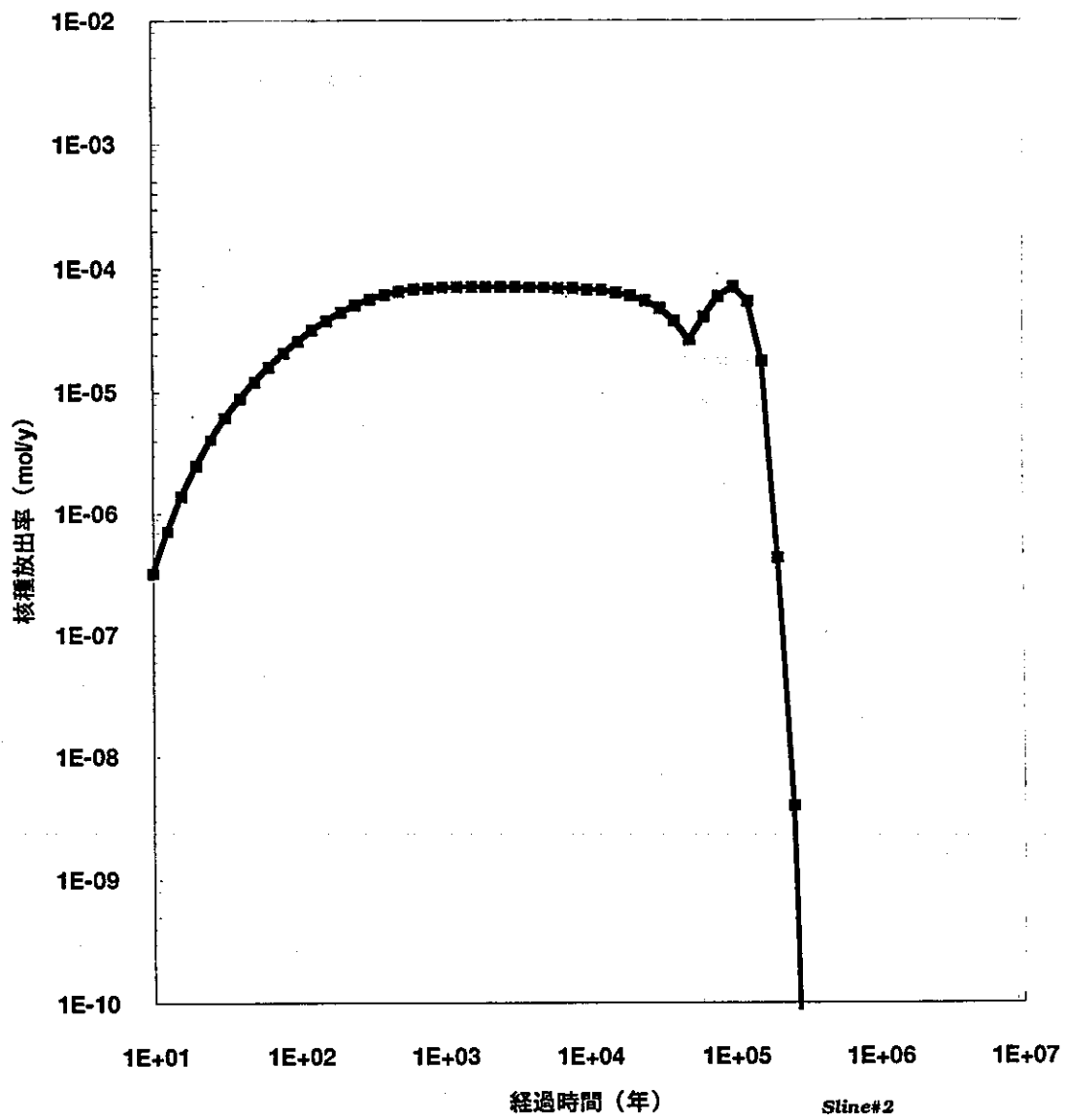


図 3.7.1 - 5 経時的化学条件変化モデルによる核種放出率算出例
 ($q_{w1} = 10^5$)

図3.7.1-6は、9ケースの計算結果をまとめたものである。溶解度の時間変化のパターンは大きく異なっているにもかかわらず、核種放出率の最大値の変動はわずかである。これは、Tc-99の濃度分布が条件変化に速やかに対応し刻々と変化する条件毎に定常状態を達成しており、最大放出率は最も高い溶解度に対応しているためと考えられる。

以上の事から、化学条件の変化は、それが空間的に一様なものである場合には最も高い溶解度を採用することで通常の評価モデルのパラメータ値変更として表現し得るものであることがわかる。換言すれば、この点のみについては、時間的変動の詳細を知る必要はなく、最悪の条件が推定されさえすれば良いということになる。

3.7.2 空間的に不均質な変化

(1) 変化のパターン

空間的に不均質な変化をもたらし得るFEPsとして、ここでは放射線分解による地下水の酸化還元電位の上昇を考える。このFEPsに対する溶解度の時間・空間的な変化のパターンとして、以下の $c^*(x, t)$ を仮定する。

$$c^*(x, t) = c^*_0 \left(1 + c^*_1 \left(1 - \frac{x - L_g}{L_0 - L_g} \right) \right) \cdot \left(1 - \exp\left(-\frac{t}{T_1}\right) - \left(1 - \exp\left(-\frac{t}{T_2}\right) \right) \right) \quad (12)$$

但し c^*_0 : 標準ケースでの溶解度

c^*_1 : 酸化性条件での溶解度を標準ケースのそれで規格化したもの

L_g : ガラス固化体半径

L_0 : オーバーパック半径

T_1, T_2 : 時間変化に関する定数

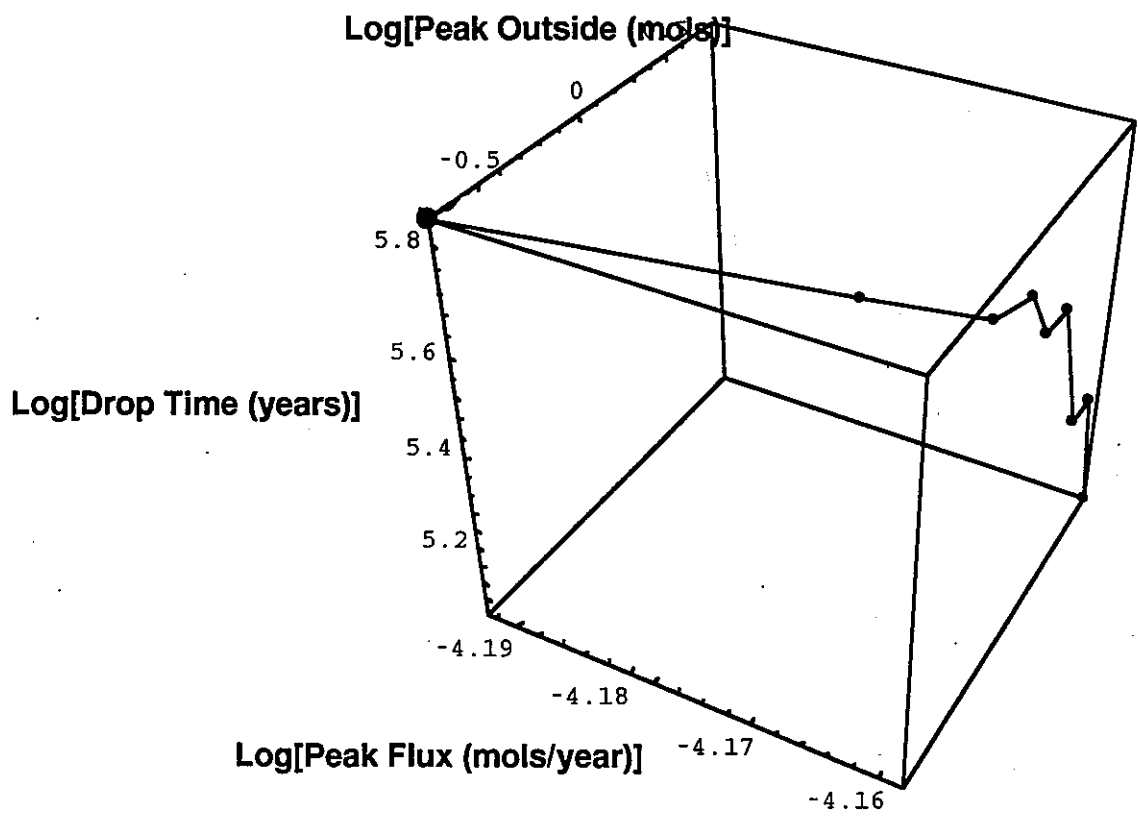


図3.7.1-6 経時的化学条件変化モデルによる感度解析結果

(12)式は、ガラス固化体近傍の α 放射性核種の影響により生成する H_2O_2 等の酸化剤が還元剤である腐食生成物層中で消費されることに対応し、各時点でのガラス固化体内の溶解度と緩衝材中での溶解度（還元性で固定）とを線型に内挿し、かつ、 α 放射線強度の減衰に応じてガラス固化体の溶解度が時間的に増減することを表現したものである。

放射線分解のソースとなる α 放射性核種としては最も寄与の大きいAm-241を考えることとし、これに対応して $T_1=200$ 年、 $T_2=1000$ 年と設定した。腐食生成物層内の溶解度は、この二つの時間の間でピークを示すこととなる。この設定の下に、 $c^*_i=10^6$ として計算した $c^*(x, t)$ を図3.7.2-1に示す。

(2) 結果

解析結果を図3.7.2-2（濃度プロファイル）及び3.7.2-3（核種放出率）に示す。

腐食生成物層中では、ガラス固化体から遠ざかるにつれて溶解度が低下することに伴い、沈殿が生じている。しかし、沈殿のフロントは緩衝材中にまでは達しておらず、このため、核種放出率については標準ケースからの変化は見られない。

但し、ここで仮定した $c^*(x, t)$ は一例に過ぎず、他の関数形を指定した場合には沈殿フロントの成長が腐食生成物層中にとどまらないケースも考え得る。

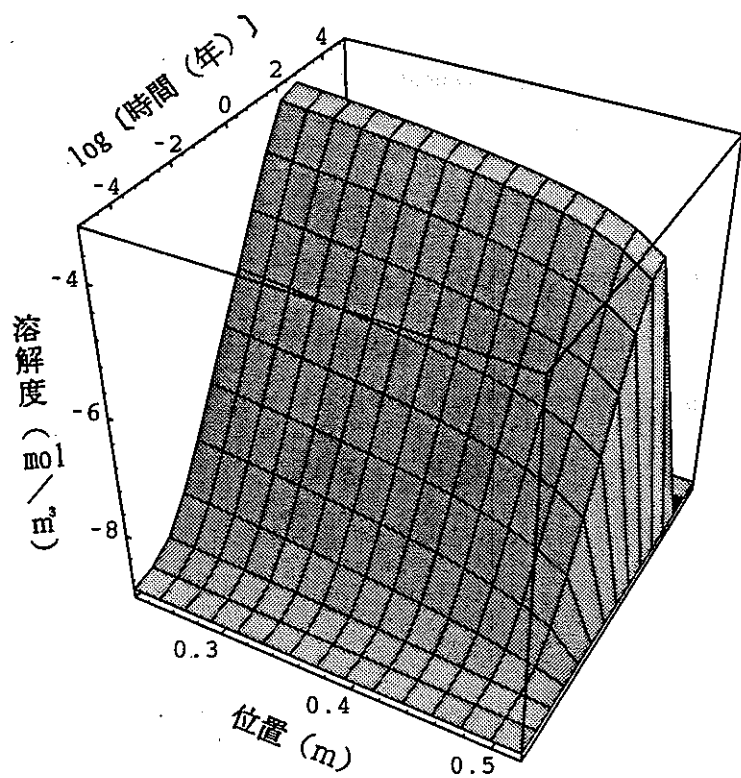


図3.7.2-1 放射線分解を考えた場合の腐食生成物層中溶解度変化の例

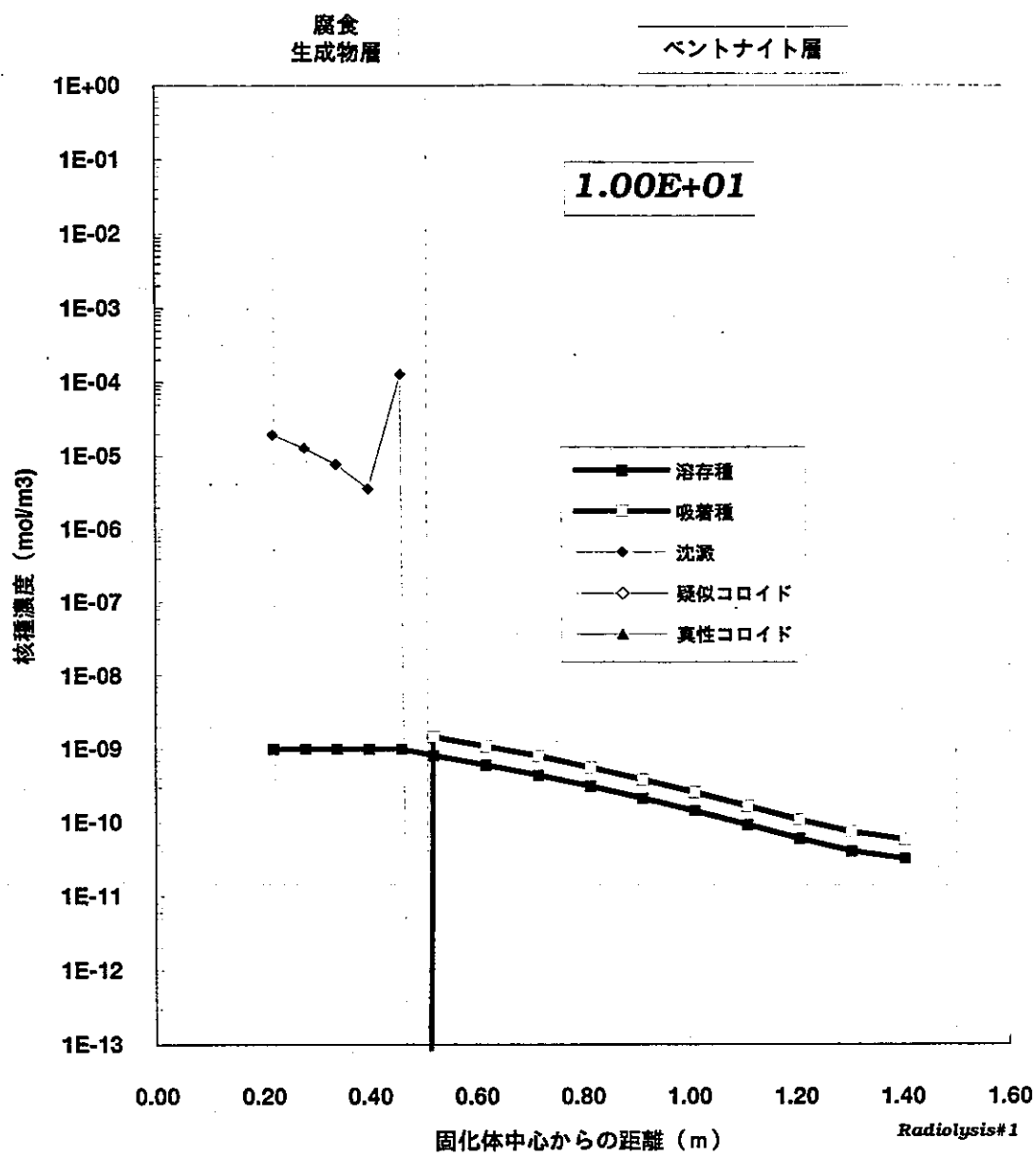


図 3.7.2 - 2 化学条件の時間・空間的变化を考慮したモデルによる濃度分布算出例

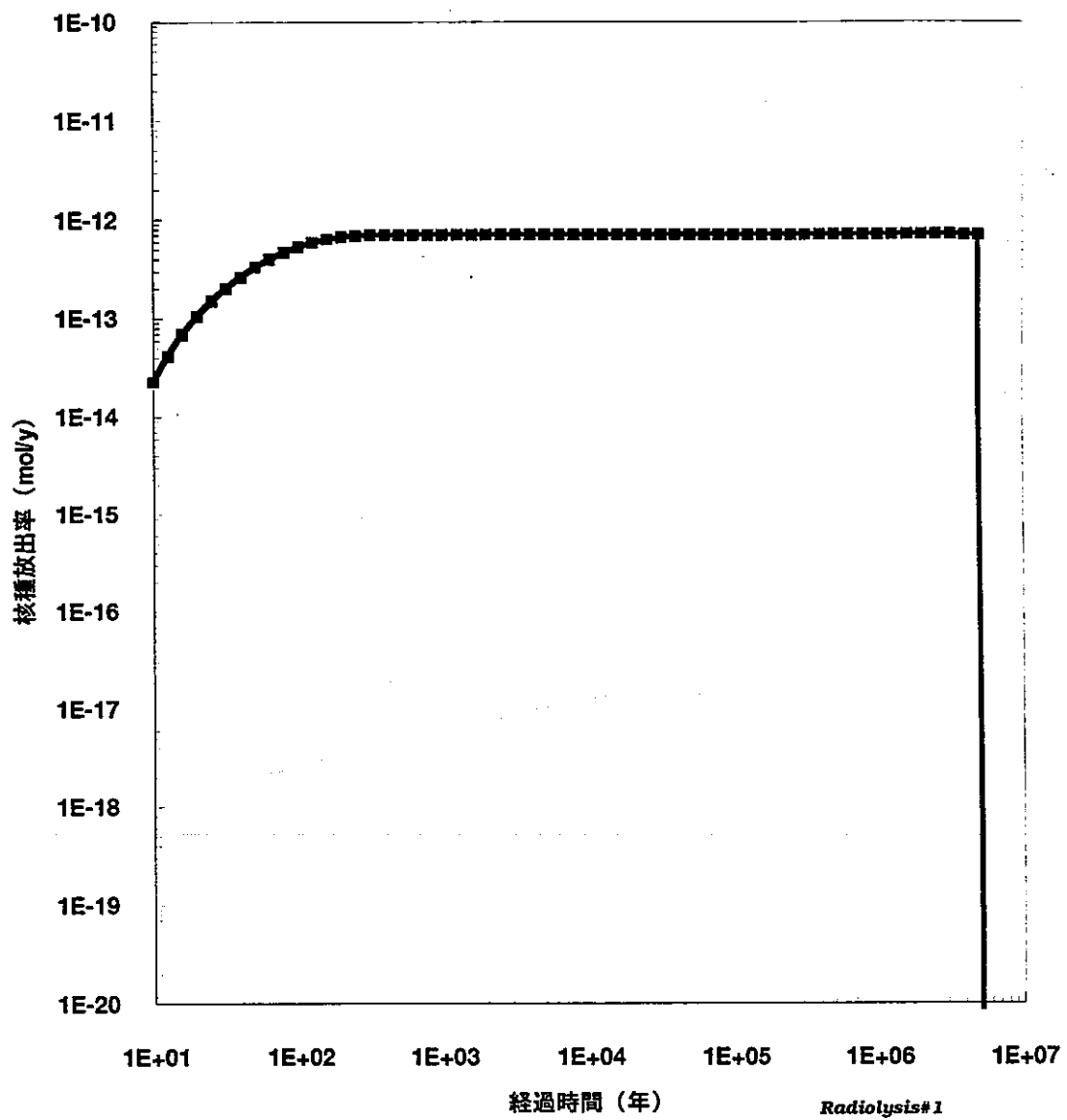


図 3.7.2 - 3 化学条件の時間・空間的变化を考慮したモデルによる核種放出率算出例

3. 参考文献

- [1] H.Takase and P.Grindrod, " Modelling Precipitation of Radioactive Elements and Their Effect upon Redox Conditions ", Proceedings of 5th HLW Management Conference, Las Vegas, to be published
- [2] NAGRA, Project Gewähr 85, 1985

4. 動的解析の試行2 (統計的感度解析)

4.1 概要

3.に述べた決定論的な感度解析は、いずれも1.で抽出されたFEPsに対応して広義拡散関数 $\alpha(x, t)$ の代数的制約条件を定式化するという方法で行われたものである。この方法は、モデルの物理的意味が明確になるという長所を有する一方で、その感度解析としての網羅性には疑問の余地がある。

一般に、性能評価の網羅性には次の二つのものがあると考えられる。

・第一種の網羅性

FEPsリストの抽出等ボトムアップの方法[1]で可能な網羅性であり、抽出を行う専門家グループのどのメンバーも、これ以上は新しい項目を思いつかないという状況。この網羅性は、母体となる専門家グループの構成や各人の主観的判断に依存するものであり、例えば性能評価を行う側から見てFEPsリストが網羅的なものであっても、規制側が別途招集した専門家グループ(例えば安全審査における顧問会)で新たな意見が出されればくつがえされる性質のものである。

・第二種の網羅性

トップダウンの方法[1]で達成され得る網羅性であり、予め全領域が明確にされており、これが分類された状況に対応する。例えば、あるパラメータが正の値を採ることが物理的に明らかである場合、しきい値を設けてそこから下を「小」上を「大」に区分することによって、失われるものは何もない。この意味での網羅性は、評価を行う組織、個人等に依存しない、本質的なものである。

以上のうち、明らかに第二種の網羅性が達成されることが好ましいが、ボトムアップ法の一つである決定論的感度解析ではこれは原理的に不可能である。そこで、本研究では、決定論的感度解析を網羅性の観点から補うものとして、統計的感度解析の方法を検討することとした。

1.で述べた広義拡散方程式に含まれるパラメータは、広義拡散関数 $\alpha(x, t)$ のみであり、

これについては、上の例における正の実数全体、と同様に可能な関数形全てを含んだ関数空間[2] を考えることによって予め全領域を明確にすることが可能である。つまり、決定論的感度解析において、 $\alpha(x, t)$ を代数的制約条件としてimplicitに表現していたことに代わり、この x 及び y の関数としての関数形をexplicitに考えて上記の関数空間から具体的な関数形を統計的にサンプリングすることによって、理論的には第二種の網羅性を達成することが可能となる。

本章では、この様な統計的感度解析の手法について検討を加え(4.2)た後、この手法を用いて試解析を実施する(4.3)。更に、この結果を踏まえ、今後の研究において決定論的感度解析を補うものとしてこの統計的方法をどの様に利用して行くべきかについて、例を含めて説明を加える(4.4)。

4.2 統計的感度解析の方法論

既に述べた様に、広義拡散方程式

$$C_t = \Delta [\alpha(x, t) C] - \lambda C \quad (1)$$

を用いた手法では、 $\alpha(x, t)$ の関数形が設定されれば、計算に必要な全ての条件は定まることとなる。一般には $\alpha(x, t)$ の形は未知であるため、3.で行った様にこれを C の関数として代数的に表現することによって計算が行われる。しかし、これに代わって、 α の x 及び t の関数としての関数形を入力することによっても計算が可能である。そして、この関数のとり得る全ての形を調べることができれば、(1)式の解が示す可能性のある全ての挙動を知ることができるはずである。これが、統計的感度解析の基本的な考え方である。

本研究では、 $\alpha(x, t)$ の母集団として、ルベグ二乗可積分な関数全体の集合つまりヒルベルト空間[2] を考えることとする。但し、ここでは、ヒルベルト空間についての細かな知識は全く前提としない。以下に述べる解析手法を理解するためには、

- (1) ヒルベルト空間は、連続関数の全てと、ステップ関数やデルタ関数といった一部の不連続関数を含むものであり、性能評価に用いられる可能性のあるほとんど全ての関

数形を含んだ十分広いものである。

- (2) ヒルベルト空間には、通常の（有限次元の）線型空間と同じく正規直交基底（二次元ユークリッド空間におけるベクトル（1, 0）及び（0, 1）の様に）が存在し、任意の関数は正規直交基底の線型和として表現される。

という二点を知ることによって十分である。上記の(1)は、ヒルベルト空間を想定することを正当化するものであり、(2)はより実用的な意味を持つものである。つまり、ヒルベルト空間の正規直交基底の一つであるFourier級数を用いて、任意の関数は近似可能であり、近似の精度は必要に応じていくらかでも向上させることが可能である。

4.3 予備的解析

4.3.1 $\alpha(x, t)$ の統計的発生

3.の標準ケース（但し、ここでは簡単のため $q \sim \infty$ とすることにより、外側境界で $c = 0$ としている）で、溶解度限界モデルに対応してimplicitに設定された α は、 x 及び t の関数としては、図4.3.1-1に示す形となっている。図中の各曲線は、空間差分に対応する格子点場での α の経時変化を示している。 $x = 0.1$ 及び 0.2 の格子点は、ガラス固化体内にあり、初期にはこれらの点には余剰のTc-99が存在しているために沈澱物を含めた平均の拡散係数である α は極めて小さい。そして、 10^8 年以上経過した時点でインベントリが消費されることに対応して、これらの点に於ける α は急激に上昇する。他方、その他の点では全期間を通じて沈澱が存在しないために、 α は溶存種と吸着種との平均拡散係数、つまり見かけの拡散係数として時間的に一定である。

本解析では、3.と同様に、上記の標準ケースを中心とした感度解析を行うために、 α を

$$\log [\alpha(x, t)] = \log [\alpha_0(x, t) \cdot (1 - \delta \tilde{\alpha}(x, t))] \quad (2)$$

但し、 $\alpha_0(x, t)$: 標準ケースでの α

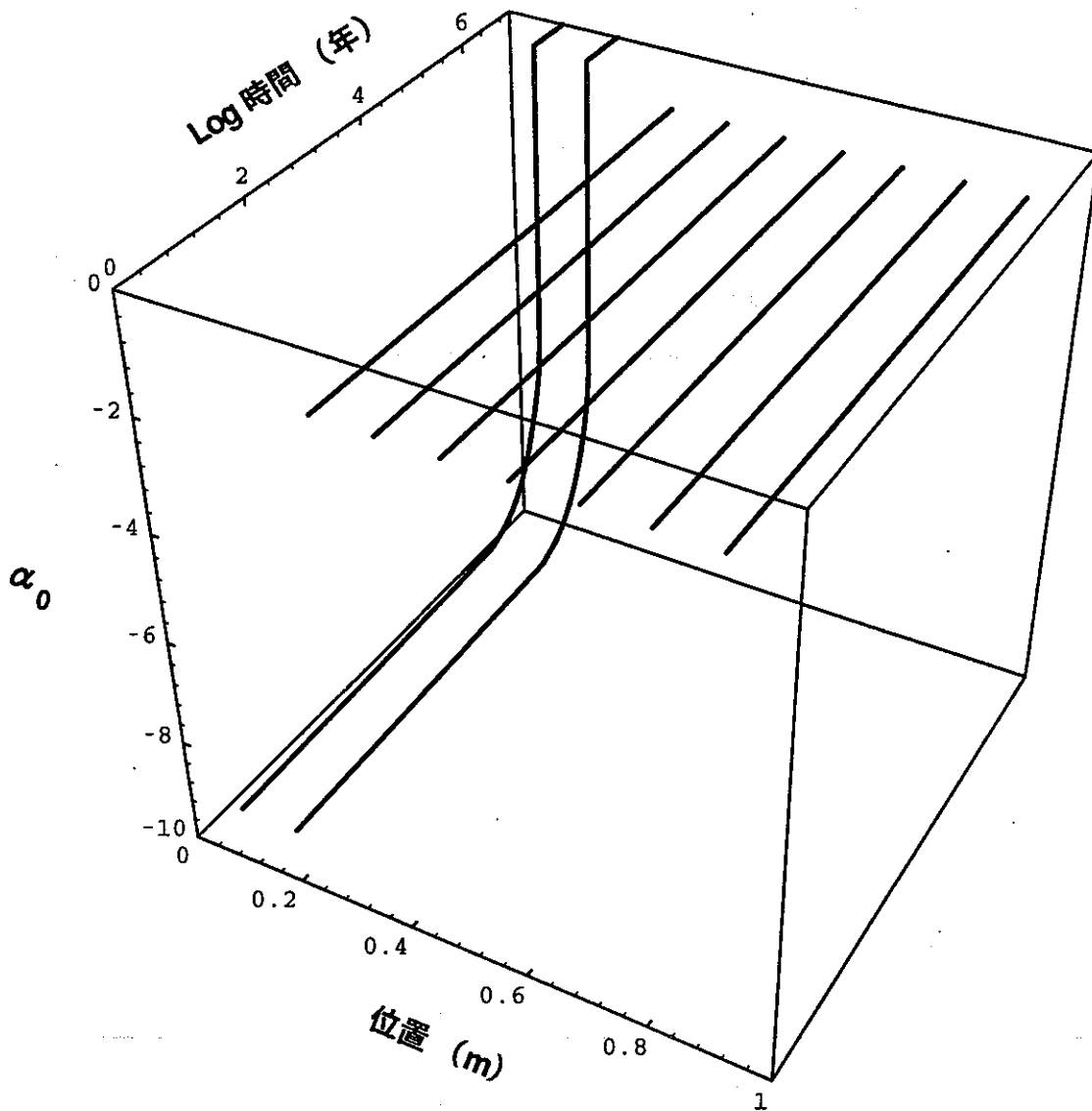


図 4.3.1-1 標準ケースでの広義拡散関数 $\alpha_0(x, t)$

として表現する。そして $\tilde{\alpha}(x, t)$ を4.2に述べた統計的サンプリングによって、

$$\tilde{\alpha}(x, t) = \sum_{m, n} a_{mn} \cos[2\pi m(x/L) + \varepsilon] \cos[2\pi m(t/T) + \rho] \quad (3)$$

として定義する。つまり、感度解析で用いる α は、標準ケースに於ける α_0 に統計的にサンプリングされた $\delta\tilde{\alpha}$ という摂動を与えたものとなる。

解析に際しては、この事から、一般性を失うことなく α を、例えば、

$$\alpha(x, t) = \sum_{m, n} a_{mn} \cos[2\pi m(x/L) + \varepsilon] \cdot \cos s[2\pi m(t/T) + \rho] \quad (4)$$

- 但し、 m, n : 任意の自然数
 a_{mn} : Fourier 級数の各項の係数
 L : 解析領域の長さ
 T : 評価期間
 ε, ρ : 初期位相

として表現することが可能となる。

そこで、(4)式における a_{mn} 、 ε 及び ρ を統計的にサンプリングすることにより、原理的には(パラメータ値の場合のモンテカルロ法と同じく) $\alpha(x, t)$ の全ての形式をサンプリングすることが可能となる。そして、こうしてサンプリングされた多数の $\alpha(x, t)$ を(1)に入力して解析を行うことによって、(1)式の全ての解の挙動を調べることができる。

なお、(4)式は、本来は無限級数であるが、実際の解析にあたっては有限個の m, n で打ち切ったものを用いることとなる。具体的には、

- m, n : 0~10の自然数
 L : 系の長さ (1.0m)
 T : 評価期間 (10⁷ 年)

と設定した。

摂動 $\tilde{\alpha}(x, t)$ の例、及び種々の δ についての $\alpha(x, t)$ の変化を、それぞれ図4.3.1-2及び4.3.1-3に示す。

4.3.2 解析結果

統計的に作成した6種類の摂動 $\tilde{\alpha}(x, t)$ のそれぞれについて{0, 0.2, 0.4, 0.6, 0.8, 1.0}という6つの δ の値を考慮した計36ケースの計算を行った。結果を図4.3.2-1に示す。

これらの結果は、次の様な点を示唆している。

- (1) いずれのケースにおいても、放出期間は 2×10^6 年 $\sim 2.2 \times 10^6$ 年の間にあり、つまり、平均核種放出率はほぼ一定である。これは(2)及び(3)式で定義される $\alpha(x, t)$ が、 $\alpha_0(x, t)$ を中心とした振動であり、長期的には α の増加(標準ケースよりも核種が総じて動きやすい状態)と α 減少(動きにくい状態)とが相殺されるためと考えられる。
- (2) 最大核種放出率の観点からは、いずれのケースにおいても標準ケースを上回るものとなっている。これは、たとえ短期的であっても、 α が増加している状態を考えることの効果が顕著となることを示している。

以上の二点から、長期的な平均値としての環境条件を固定したケース(標準ケース)は、同じ平均値を持ちながら時間的に変動する条件を考えたケースよりも小さい核種放出率に帰結するものと考えられる。つまり、最大核種放出率は、短い期間であっても好ましくない環境条件が現れる状態に支配されることとなる。

更に、各ケースでの $\alpha(x, t)$ の摂動パターンと計算結果とを比較すると、次の様な点が推論できる。

- (3) 環境条件がAという状態からBという状態に変化した場合、核種放出率のピークはむしろAからBへの変化の過度状態において達成される可能性がある。

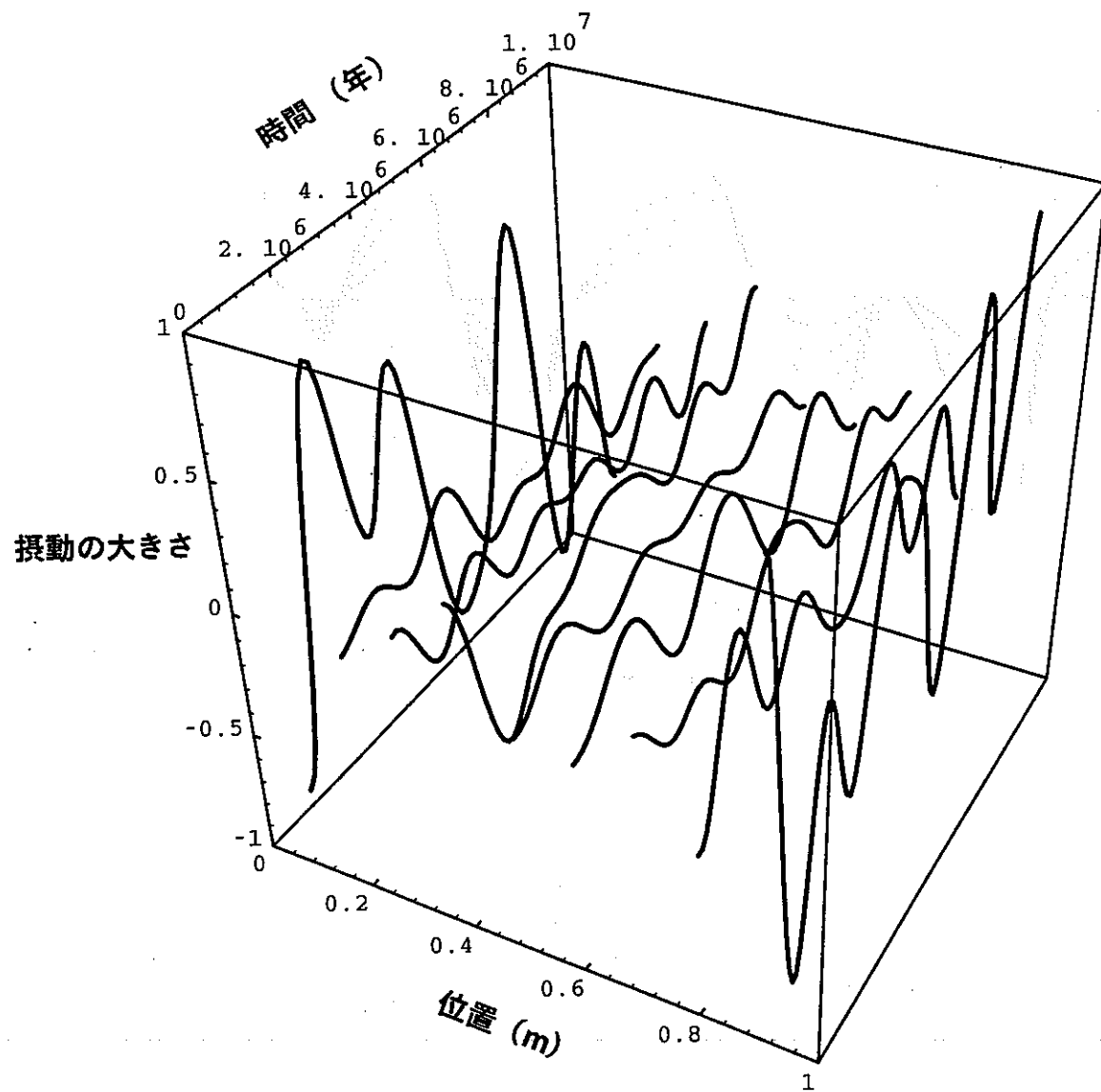


図 4.3.1-2 統計的に作成した摂動 $\tilde{\alpha}(x, t)$ の例

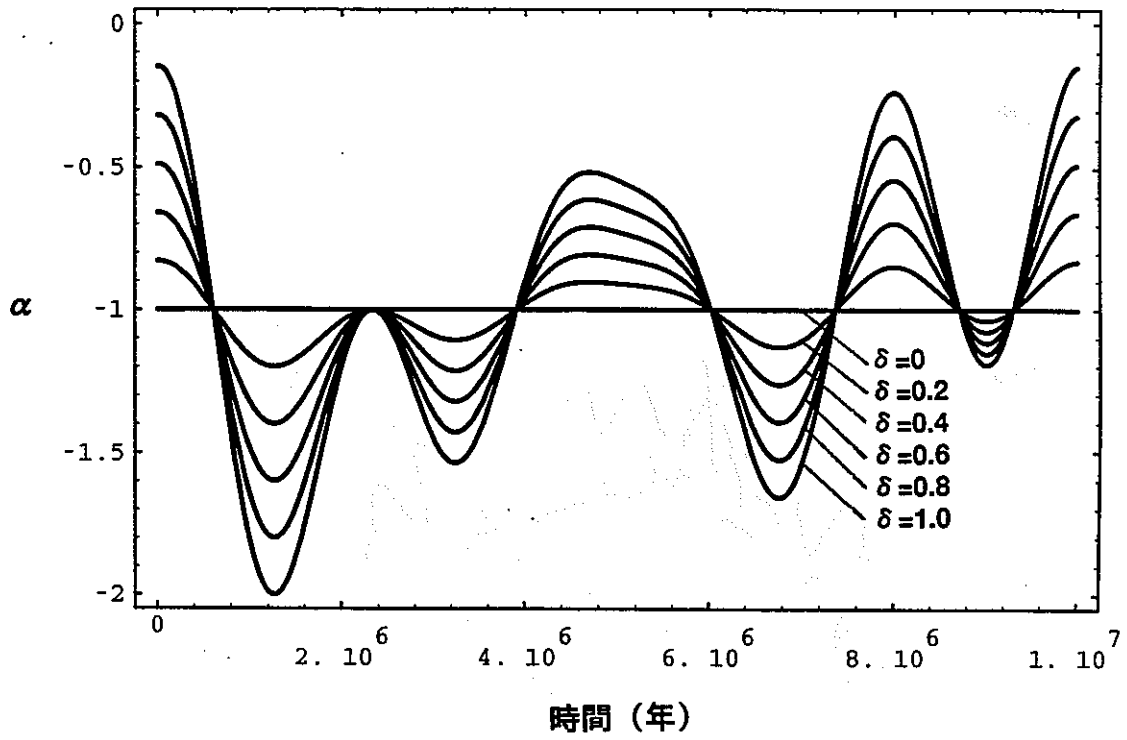


図 4.3.1-3 δ を種々に変えた場合 α の例
(ベントナイト最外側)

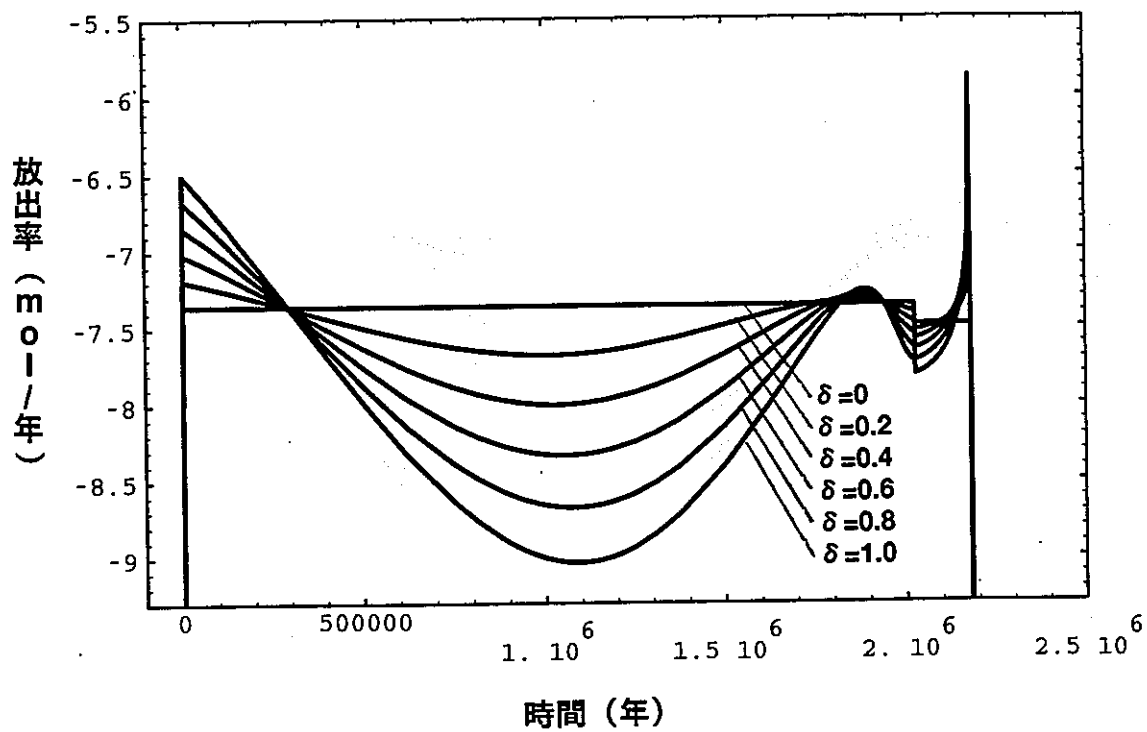


図 4.3.2 - 1 (1) 統計的感度解析の結果

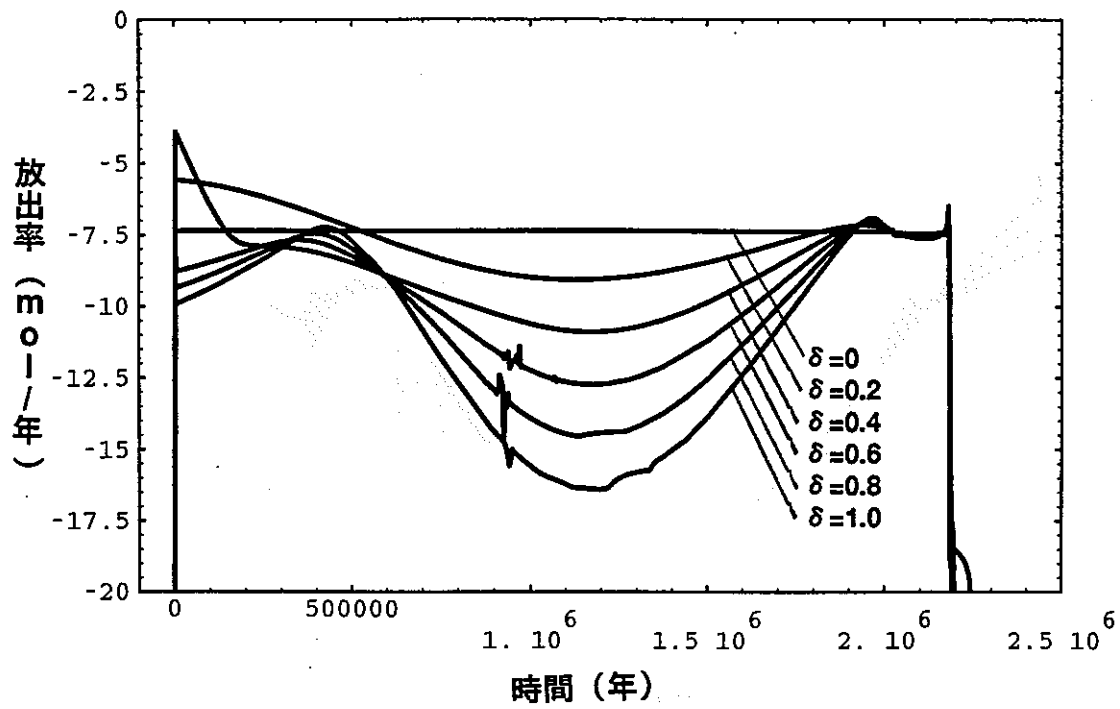


図 4.3.2-1 (2) 統計的感度解析の結果

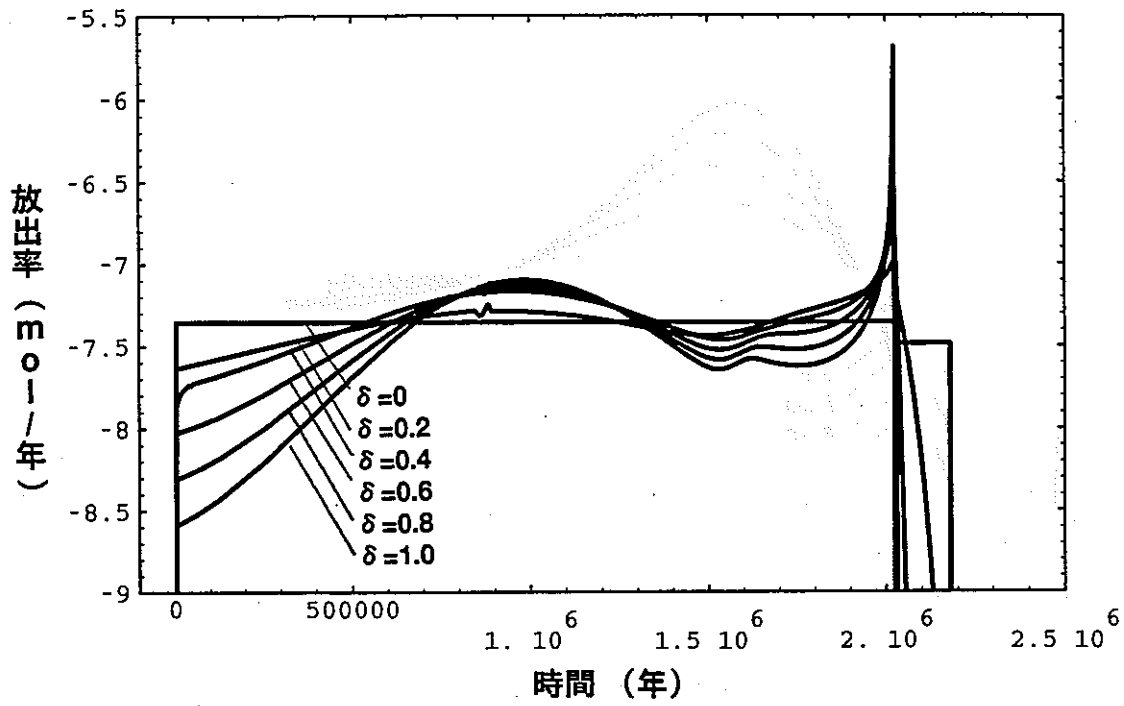


図 4.3.2-1 (3) 統計的感度解析の結果

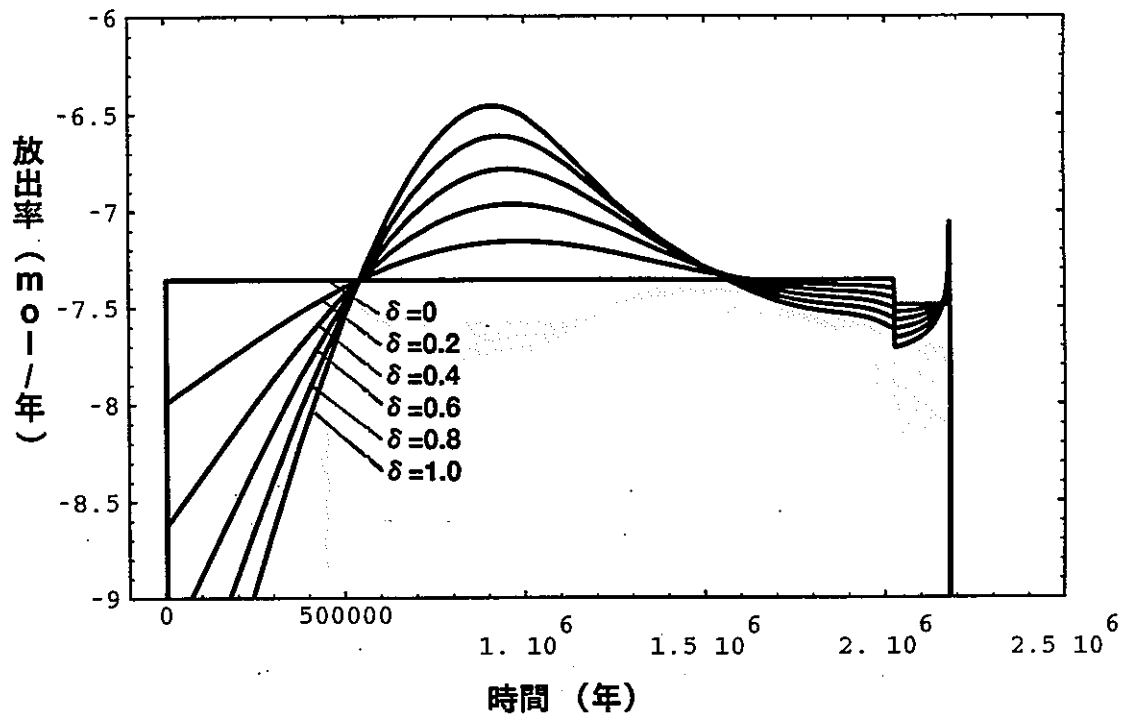


図 4.3.2-1 (4) 統計的感度解析の結果

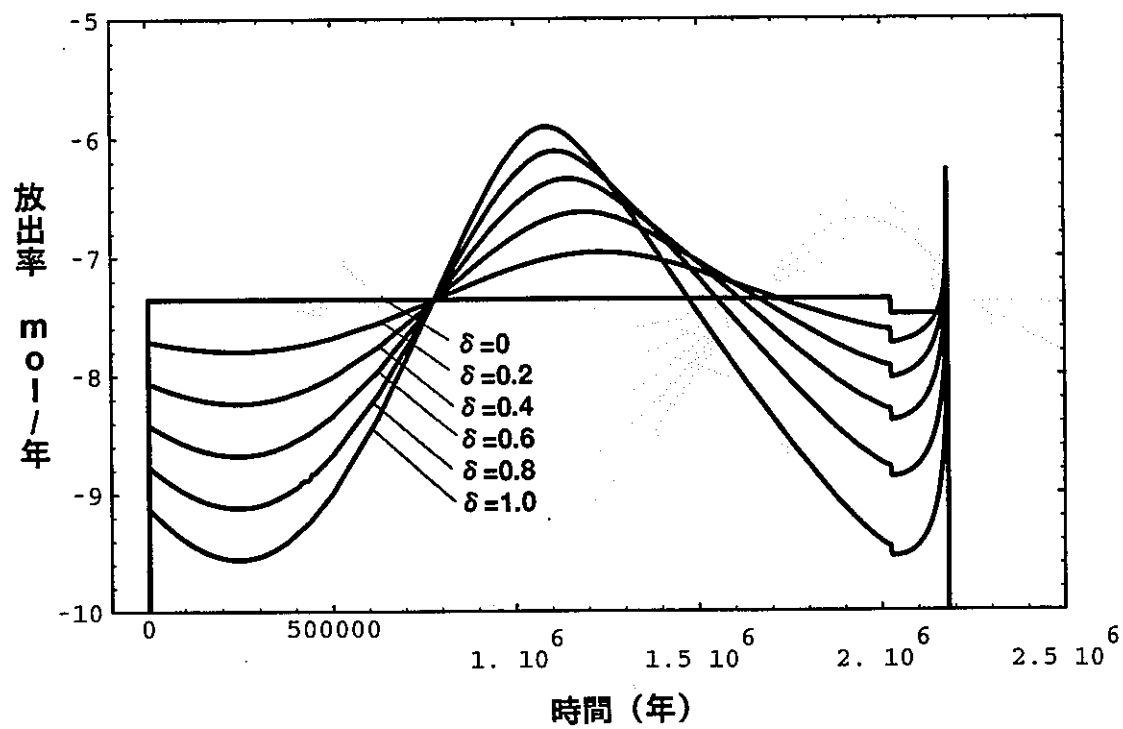


図 4.3.2-1 (5) 統計的感度解析の結果

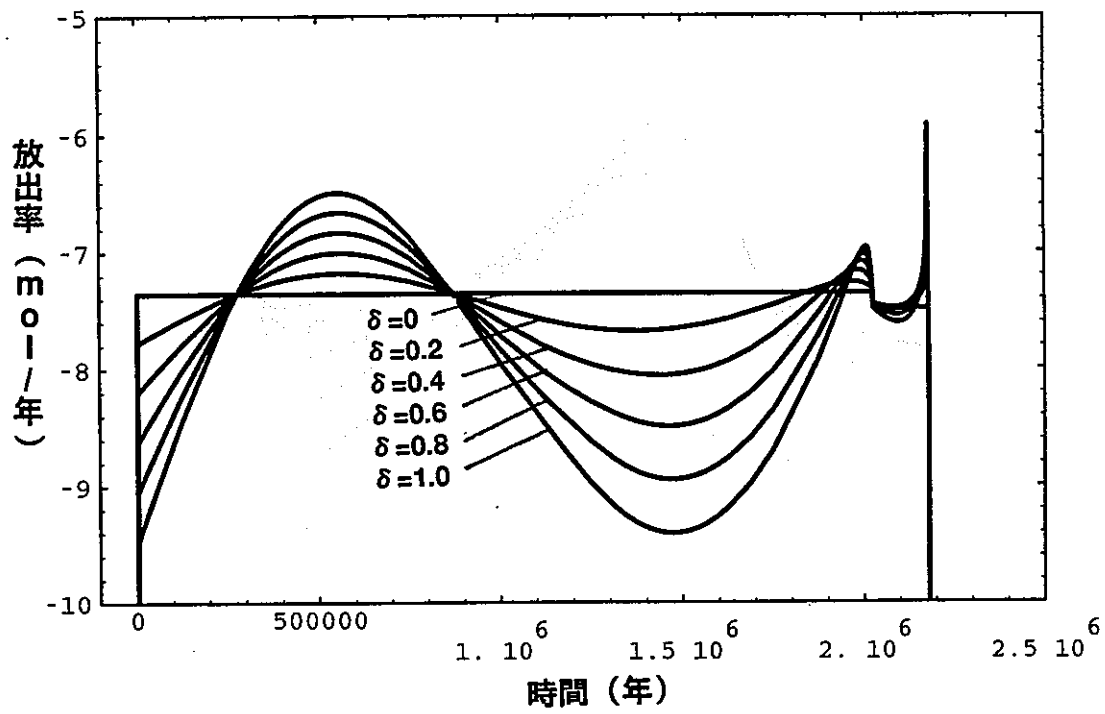


図 4.3.2-1 (6) 統計的感度解析の結果

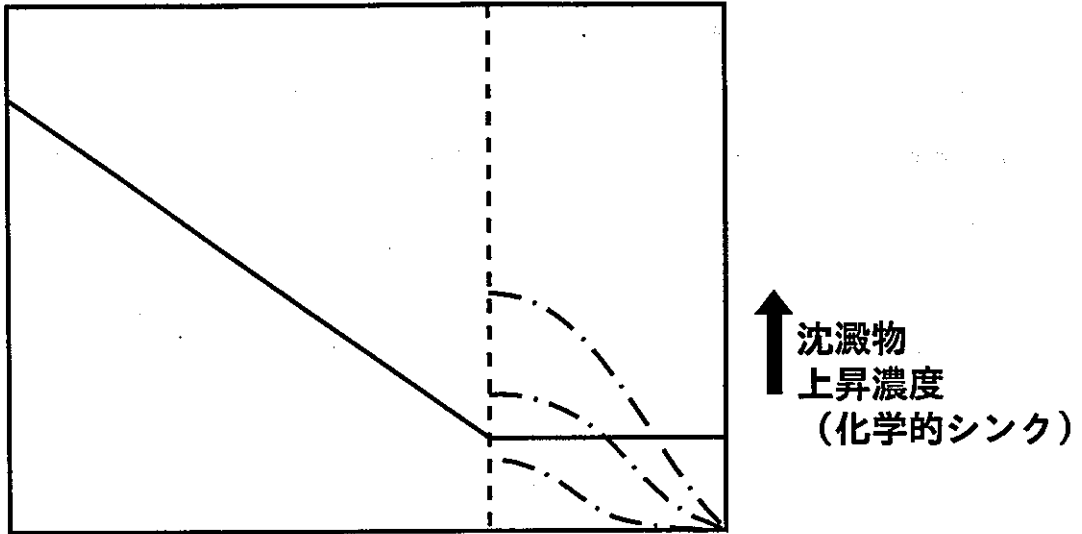
この様な可能性の例として、より具体的には次の様な場合が考えられる（図4.3.2-2）通常の溶解度限界、線型吸着の核種移行挙動において、ある時期周囲の地下水条件がより小さな溶解度を与える好ましいものになり、一定期間後、もとの地下水条件に戻る場合、周囲の地下水が好ましい条件にある期間は人工バリア内から放出される核種がゆるみ域内で沈澱するという化学的なシンクが形成され、その後地下水が元の条件に戻った時点でこの沈澱が急速に溶解することによって大きな核種放出率が出現することとなる。

元の地下水条件、及びより好ましい地下水条件をそれぞれ固定して考えた場合には、この様な現象は生じ得ない。つまり、上述した核種放出率のピークは、二つの条件の間の過度挙動を考えた場合にはじめて現れるものである。

この点については、次節でもう少し詳しく説明することとする。

—— 液相濃度
 - - - 沈澱物濃度

(a) より好ましい地下水条件での核種挙動



(b) 条件が復旧する際の過度的核種挙動

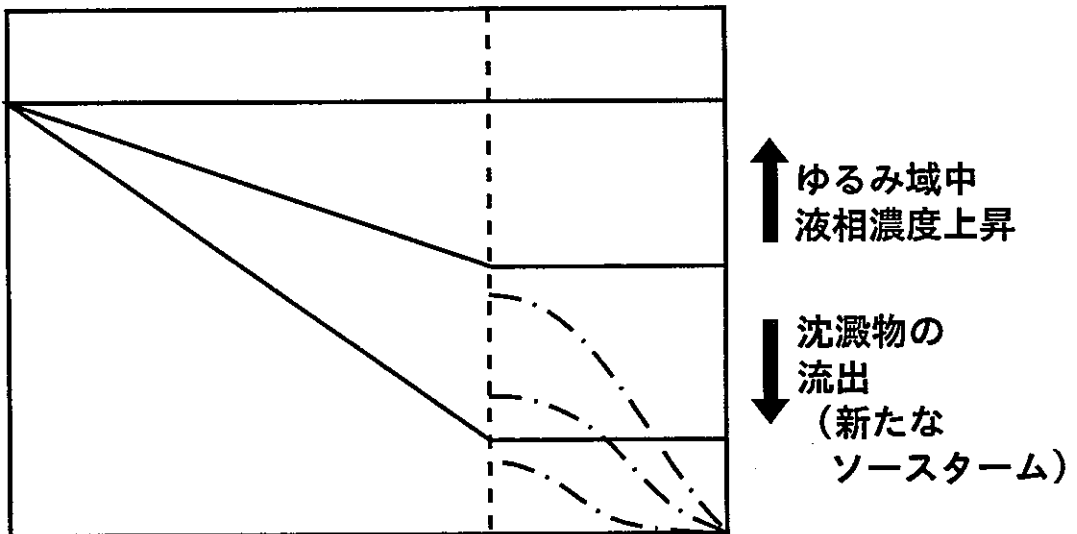


図 4.3.2-2 条件変動に対応した過度的核種挙動の例 (概念図)

4.4 今後の課題

既に述べた様に、統計的感度解析は決定論的感度解析を補完して、性能評価をより網羅的なものとする位置付けにある。今年度の研究ではこれら二つの方法は独立した形で適用されたが、将来的には以下に示す様に相互に補完的なものとして用いる事が必要と思われる。

(1) 決定論的感度解析の結果を中心とした局所的感度の定量化

一般に、理論的なモデルが実際の現象を完全には説明しきれない様に、決定論的モデルは現実の一つの近似として捉えるべきものである。さらに、本研究の決定論的感度解析で用いる諸モデルが不明点の残る現象についての第一次近似としての意味合いが強いことを考えれば、個々の決定論的解析モデルの周辺には、これと類似しているが少しずつ“ずれ”を持ったモデル化オプションが本来は存在しているものと考えられる。そこで、これらのモデル化のバリエーションを、決定論的感度解析で対象とした評価ケースを中心としつつ、統計的な摂動を与えることにより拾うという方法が重要となる。

つまり、今年度の研究で標準ケースを中心として統計的な感度解析を行った様に、他のケース（コロイド、条件の経時変化等）を中心とした統計的感度解析を行うことによって、極めて幅広い可能性の中で点にすぎないかも知れない決定論的解析の諸ケースをそれぞれ拡張することが可能である。この方法の概念を図4.4-1に示す。

(2) 統計的感度解析の結果に基づく新たなFEPsの抽出

シナリオ解析におけるFEPsの抽出は、多分に各専門家の経験と想像力に負う所が大きい。しかし、人工バリアの長期挙動については全体としての経験は限られているために、FEPs抽出のソースとなる情報量はそもそも限られているというのが現状である。

この方法は、上の様な状況を踏まえて、人工バリアの長期挙動についての経験の不

● 決定論的解析による
網羅性の向上

→ 統計的解析による
網羅性の向上

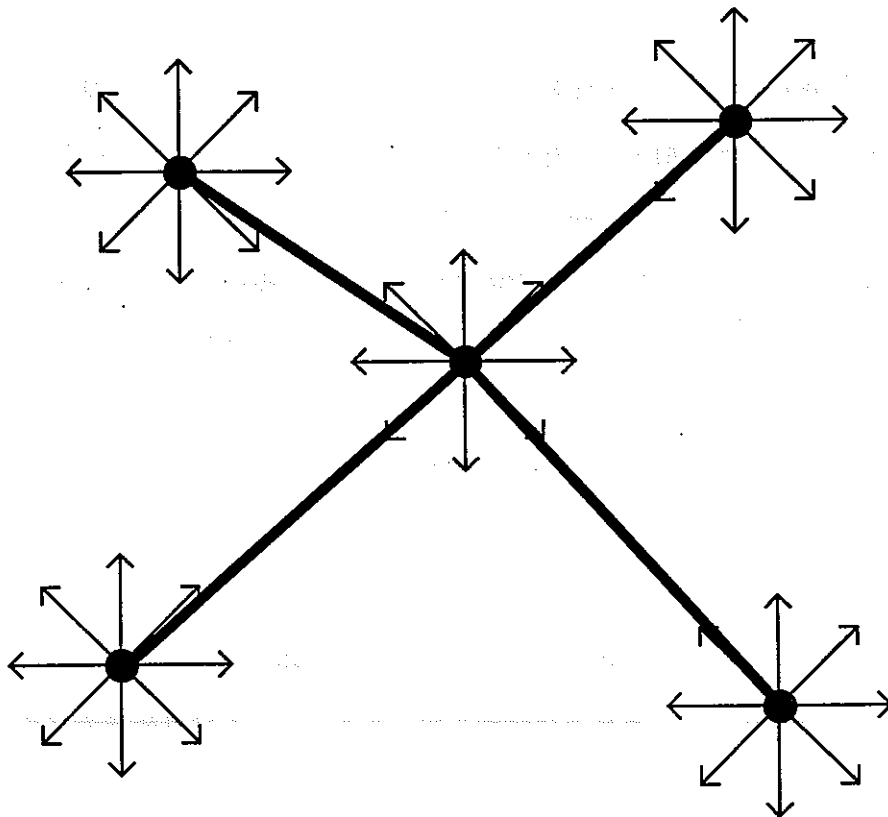


図 4.4-1 決定論的手法と統計的手法とを相互補完的に用いた
感度解析の概念

足を統計的感度解析の結果で補うというものである。つまり、純粹に統計的に作成された $\alpha(x, t)$ は、我々がこれまで見落としていた物理的な現象と何らかの関連があるかも知れない。そこで、この α からさかのぼって見落とされていた現象を拾い出して見るという行為が意味を持つこととなる。

この具体的な例としては、4.3に述べた人工バリア外側での化学的シンクの形成がある。実際、周囲の地下水化学条件が一定期間好ましいものになり再び元の状態に戻ることによって核種放出率が増加する可能性は明らかにこれまで無視されていた。然しながら、全節に述べた様に、少なくとも原理的にはこの様な可能性は否定できないことになる。

この考察からさかのぼって考えると、人工バリア外側に化学的シンクが形成される原因となり得るPEPsとして次の二つが考えられる。

- ・ 周囲の地下水環境が一定期間変化し、ゆるみ域での核種溶解度が一時的に低下する。
- ・ 抗道支保として鉄材を用いた場合、一定期間この部分が強還元性となり、その後腐食生成物の消費によって元の地下水条件に戻る。

ここでは、このようなPEPsが生じた時に、実際に核種放出率は増加するのだろうか。この点を見るために、広義拡散方程式を用いて、ゆるみ域での溶解度を 10^6 年間低下させて、その後再び標準ケースの数値に戻すという計算を行った。

図4.4-2にこの場合の濃度分布を示す。核種濃度は外側において急激に低下しており、ゆるみ域において沈殿が形成されていることを示唆している。しかし、標準ケースではゆるみ域内の流量が極めて小さく、この沈殿がゆっくりとしか溶解しないために核種放出率(図4.4-3)に大きな変化はない。

これに対して、標準ケースの1000倍に当たる $0.71\text{m}^3/\text{yr}$ の地下水がゆるみ域中を流

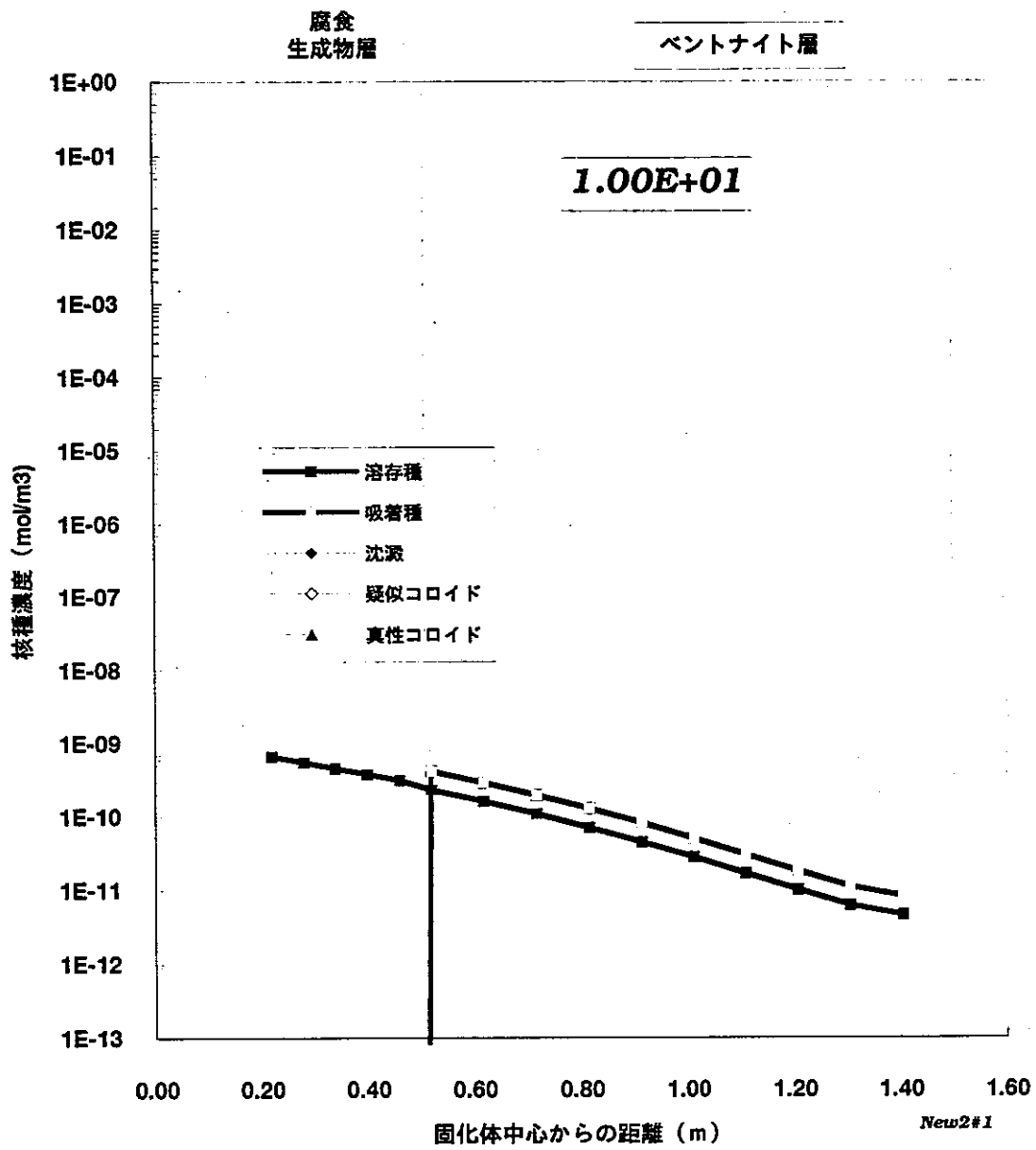


図4.4 —2 (1) ゆるみ域における化学的シンク形成を考慮したモデルにより算出された濃度プロファイル

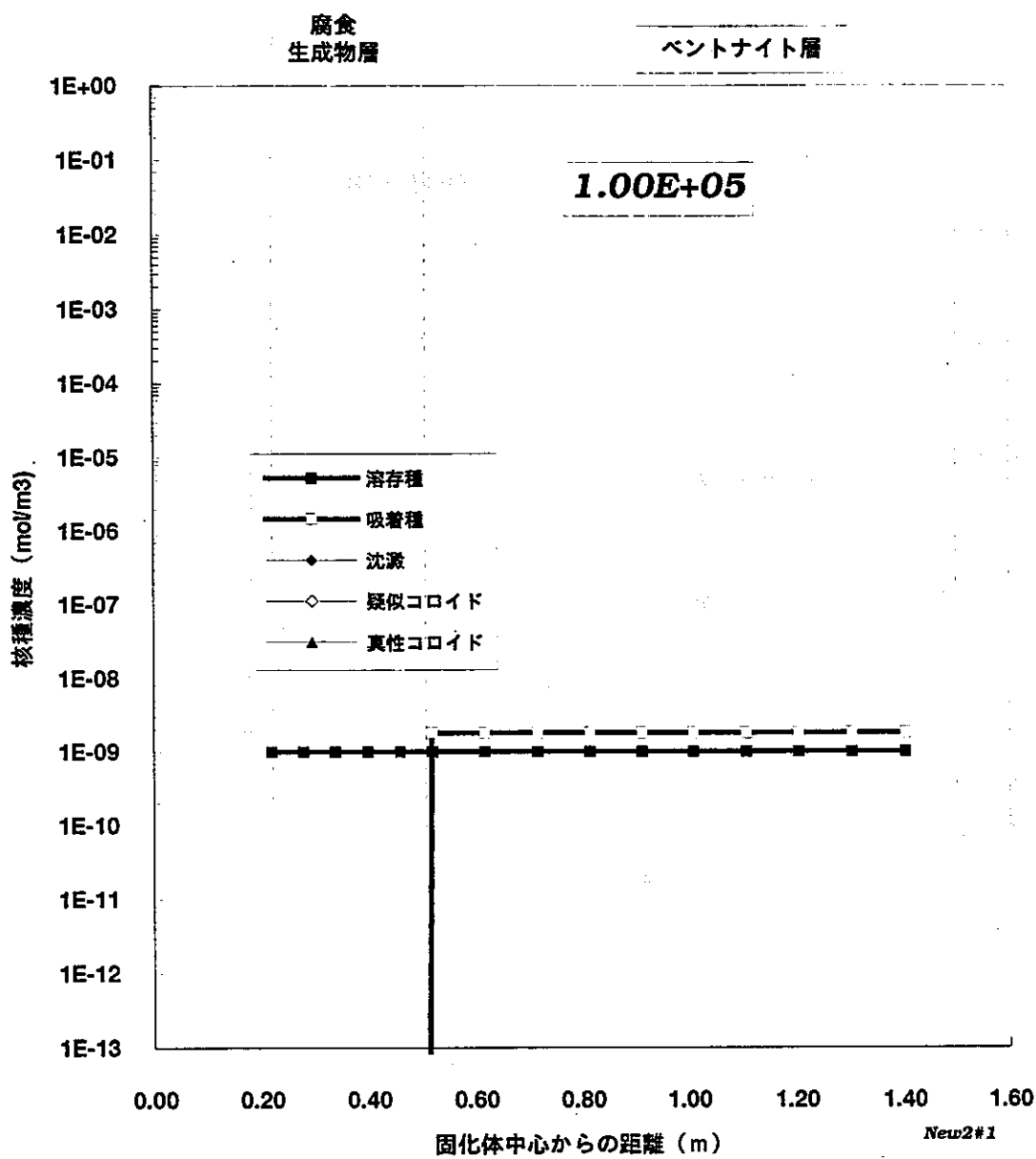


図4.4 —2 (2) ゆるみ域における化学的シンク形成を考慮したモデルにより算出された濃度プロファイル

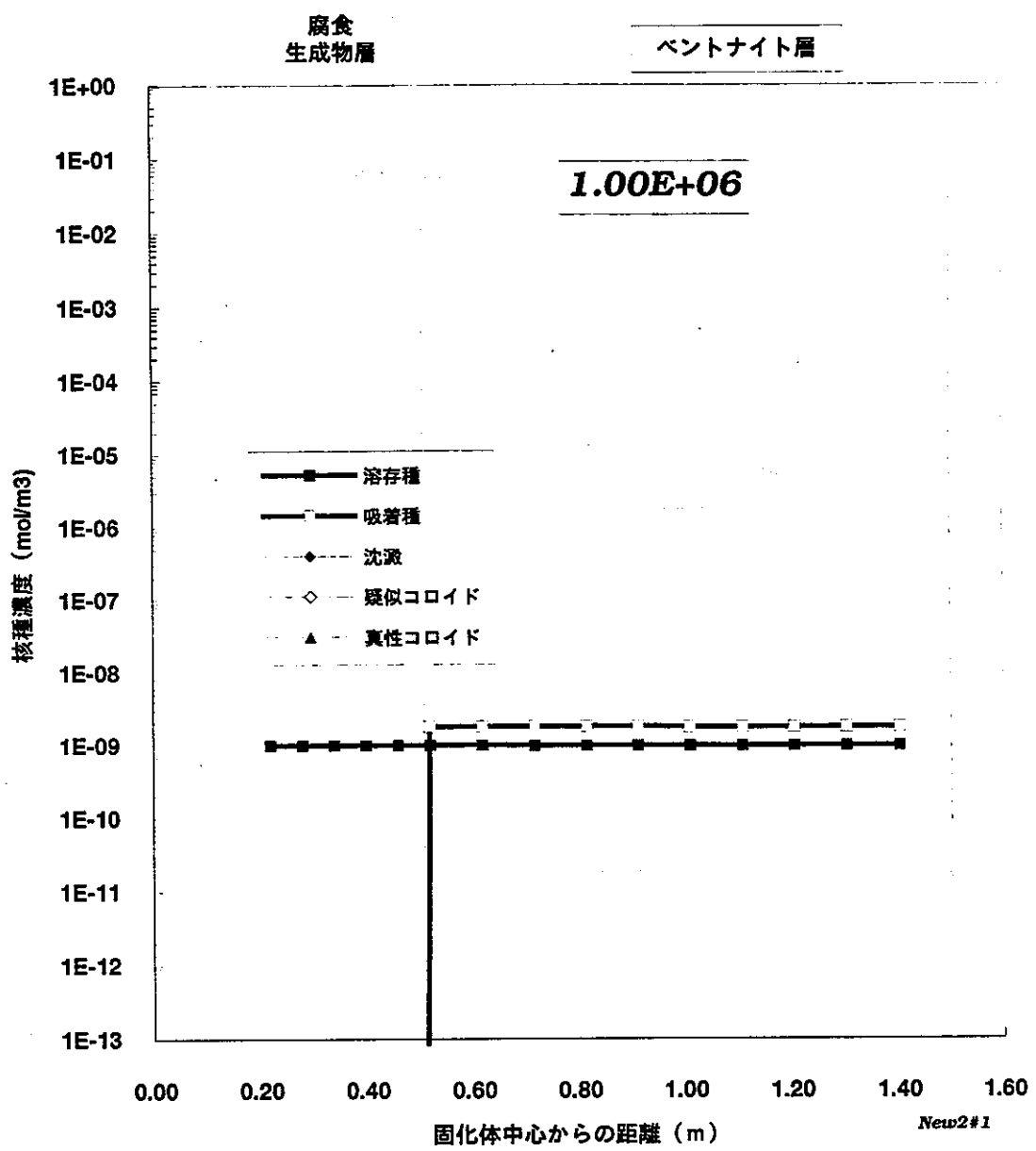


図4.4 —2 (3) ゆるみ域における化学的シンク形成を考慮したモデルにより算出された濃度プロファイル

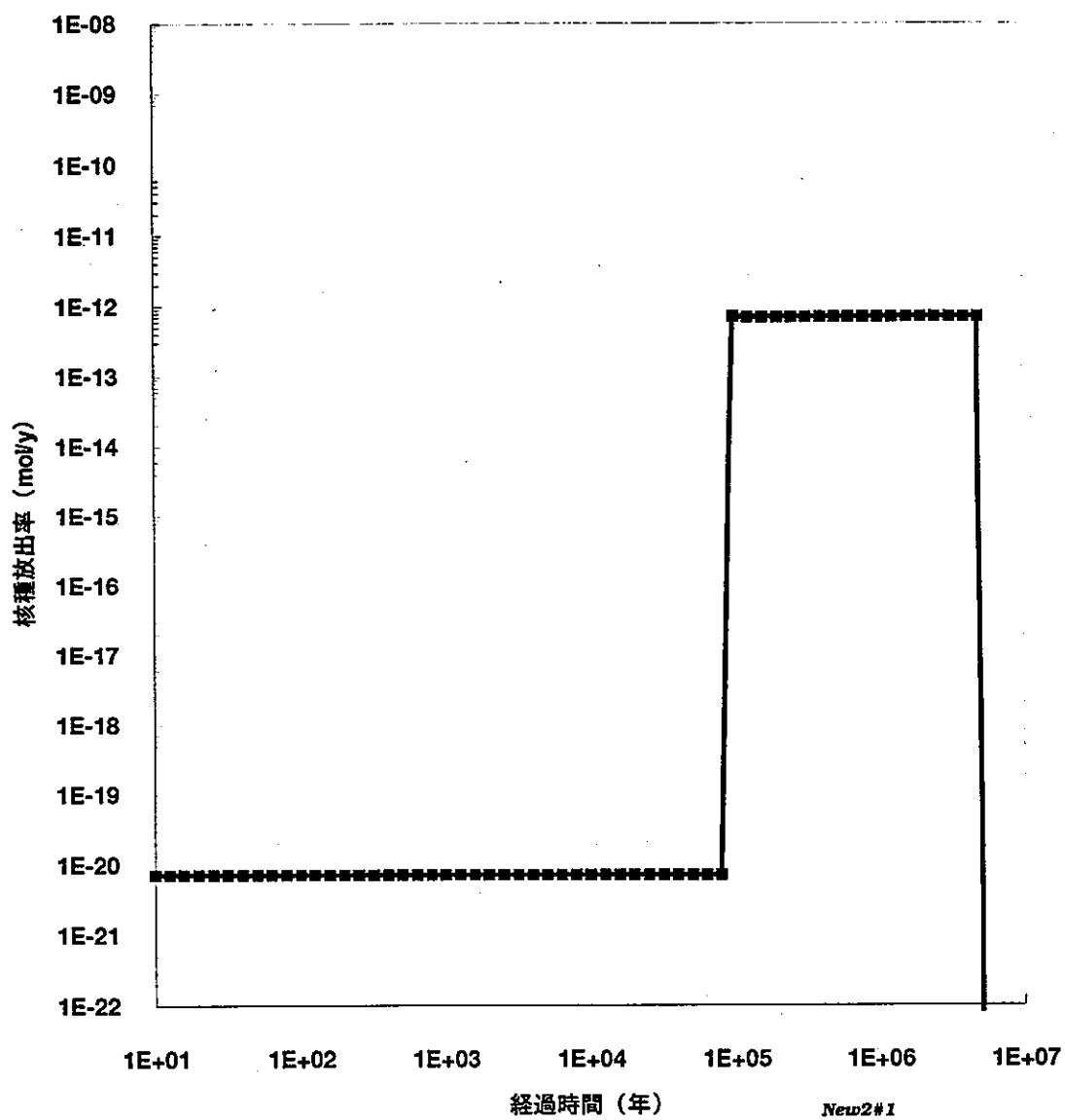


図4.4 —3 ゆるみ域内地下水流量が小さい(標準ケースと同じ)場合の核種放出率

れる場合の核種放出率を計算した結果が図4.4-4である。ここでは、ゆるみ域内に沈澱として蓄えられた核種が、豊富な量の地下水によって一気に溶解・放出されることによって、スパイク状の放出率の上昇が認められる。

以上の例題は、FEPsの抽出 → 決定論的感度解析 → 統計的感度解析 → FEPsの再抽出 → 決定論的解析による確認 という手順を反復的に用いることにより、性能評価の網羅性能評価が向上する可能性を示すものと言うことができる。

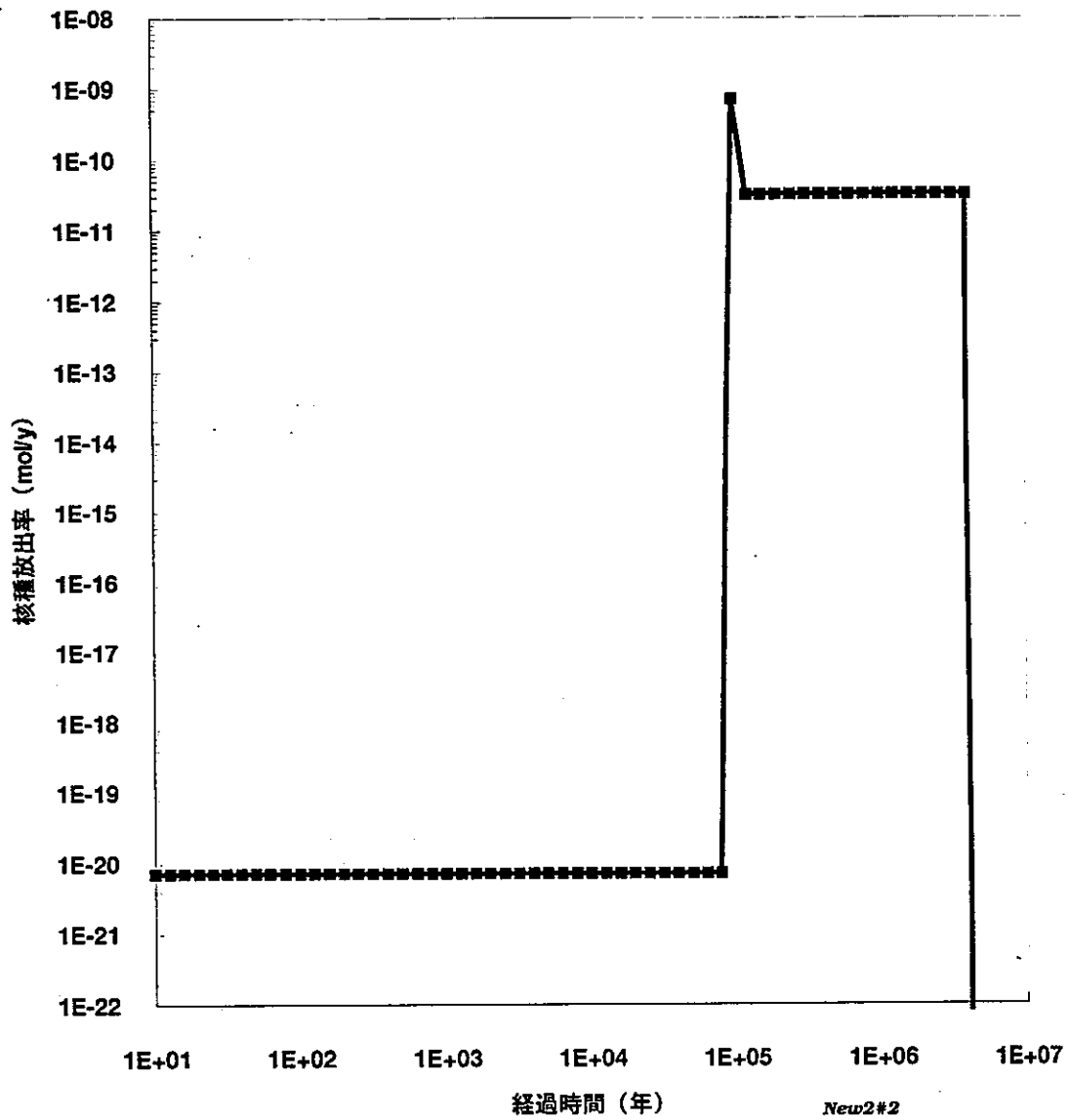


図4.4 ー4 ゆるみ域内地下水流量が大きい(標準ケースの 10^3 倍)場合の核種放出率

4. 参考文献

- [1] D.Hodgkinson and T.Summerling, " A Review of Approaches to Scenario Analysis for Repository Safety Assessment ", Proceedings of International Symposium on Safety Assessment of Radioactive Waste Repositories, CEC, IAEA, OECD/NEA, CEA, 1989

5. まとめ

動的解析の手法開発として、人工バリア内が拡散支配である限り様々な条件変動やコロイド形成等の新たな現象を考えた場合にも共通して適用可能な広義拡散方程式を導出し、具体的なFEPsに対応するものとして、

- (1) 瞬時平衡、線型吸着、溶解度限界モデル
- (2) 真性コロイドモデル
- (3) 擬似コロイドモデル
- (4) 真性コロイド、擬似コロイド共存モデル
- (5) 非線型吸着モデル
- (6) 水理・化学条件の時間・空間的变化を考慮したモデル
- (7) 沈殿・溶解反応の速度論的表現を加えた真性コロイド、擬似コロイド共存モデル

を作成した。

また、この解析手法を実用に供するために、汎用DAEソルバーSPADEとグラフィカルユーザーインターフェイスSAMGUIを改良した。主たる改良点は、

- (1) SPADEの自動差分機能の追加、方程式入力フォーマットの簡単化、リスタート機能、及び多数回の計算を一括処理するためのユーティリティプログラムの付加
- (2) SAMGUIへのデータ移送と抽画の自動化

である。

更に、改良されたソフトウェアを用いて、Tc-99を例として決定論的及び統計的感度解析を予備的に実施した。

決定論的解析としては、瞬時平衡・線型吸着・溶解度限界モデルによる計算を行ない、これを標準ケースとして以下のモデルの変動を考えた場合の結果の変化を合計約400 ケースについて解析した。

- (1) 真性コロイドモデル
- (2) 擬似コロイドモデル
- (3) 真性コロイド、擬似コロイド共存モデル
- (4) 水理条件の経時変化を考慮したモデル
- (5) 化学条件の（空間的に一様な）経時変化を考慮したモデル
- (6) 化学条件の（空間的に不均質な）経時変化を考慮したモデル

結果として、真性コロイドが存在する場合、大きな核種放出率の増加が生じ得ることがわかった。

統計的解析としては、標準ケースからのモデル及びパラメータの変動を統計的に作成した感度解析を行った。この結果に基づき、決定論的感度解析の網羅性を補完するものとしての位置付けが明らかとなった。特に、人工バリア周辺の地下水条件が一定期間より好ましいものとなりその後元の状態に戻った場合、ゆるみ域が核種についての化学的シンクとなり、一旦ここに沈殿した核種が条件復帰後に一気に放出されることにより、元の化学条件では見られない過渡的なピーク放出率が現れることが示された。このような効果は、従来の決定論的な思考では見落されていたものであり、二つの手法を併用することの利点を示す例とすることができる。

以上の結果から、人工バリアの長期的挙動に関して予想される幅広い前提条件と多岐にわたる現象を共通の手法で網羅するものとして本研究で開発した動的解法手法が有効であることがわかった。

この成果を受け、次年度には、平成三年度報告書で対象とした全ての核種について、多くのFEPsを考えた解析を行うことが可能となった。このような解析の実施にあたっては、感度解析をより体系的かつ効率的に行える様にするために、SPADEのフ

ロントエンドを若干改良することが必要である。

本文中のFEPsリスト(表1.2-1)において抽出されたFEPsのうち、主としてモデル作成あるいは予備的解析の対象としたものについて内容を以下にまとめる。

ベントナイトのイライト化

ベントナイトの主成分であるモンモリロナイトは、空隙水中のアルミニウム及びカリウムと反応することによって、イライトと呼ばれる別種の粘土鉱物と石英の混合物へと変化する可能性がある。この現象については、これまで幅広いレビューがなされており[Grauer, 1986; Grauer, 1988; Pusch and Karnland, 1988]、以下の点が明らかとなっている。

- ・イライト化が生じるためには、臨界温度(約 130 °C)が存在し、この温度以下ではイライト化速度は無視し得るほど小さい。
- ・イライト化速度は、主として拡散による緩衝材外部からのカリウムの供給速度によっても制限される。

以上、及び豊富に存在するナチュラルアナログの事例から考えて、長期間を考えた場合、イライト化がベントナイトに要求される諸機能を著しく低下させる可能性は低いものと考えられている。

Grauer, R. (1986), Bentonite as a Backfill Material in the High-Level Radioactive Waste Repository: Chemical Aspects, NAGRA Technical Report 86-12E, Baden, Switzerland

Grauer, R. (1988), The Chemical Behaviour of Montmorillonite in a Repository Backfill: Selected Aspects, NAGRA Technical Report 88-24E, Baden, Switzerland

Pusch, R. and O. Karnland, (1988), Hydrothermal Effects on Montmorillonite: A Preliminary Study, SKB Technical Report 88-15, Stockholm, Sweden

周辺母岩の亀裂中へのベントナイトの侵入

ベントナイトは高い可塑性を有するために、それ自体のもつ膨潤圧によって周辺母岩中の亀裂へと侵入する可能性が指摘されている。この様な現象が生じた場合には、長期間にわたって徐々に緩衝材密度が低下する事に伴い、人工バリア中を通過する地下水流量が増加する可能性がある。

しかしながら、ベントナイトの亀裂中への侵入を模擬した実験によれば、侵入速度は極めて遅くまた経時的に低下するため、通常の亀裂開口幅では問題とはならないことが示されている。なお、この点については、同実験結果を現在の我が国の人工バリア設計に適用した計算を本研究でも実施している。結果については付録-2 を参照のこと。

亀裂中での地下水流によるベントナイトの浸食

亀裂中に侵入したベントナイトの先端部分はゲル化しており、地下水流速が一定以上となった場合にはこの部分での浸食が生じ得る。しかしながら、この様な状況を模擬して行った実験の結果から、地下水流速が 10^{-4} m/s 以下である場合には浸食が生じないことが明らかとなっている。実際の深部地下環境で、この様な流速が生じる可能性は極めて小さく、従って浸食によるベントナイト透水係数の増加はほぼ無視し得るものと考えられている。詳細については、付録-2 を参照のこと。

放射生元素の過飽和

一般に、難溶解生元素については、人工バリア中の核種移行問題をガラス浸出率律速で解いた場合と溶解度限界の内側境界条件で解いた場合とでは、溶解限度の場合の方が核種放出率が小さくなることが知られている。つまり、このフラックスの差は元素の沈澱生成による化学的なシンクタームによって相殺されていることとなる。従って、この沈澱生成反応の速度が物質移行との比較において相対的に遅い場合には、ガラス固化体内あるいは緩衝材中の内側領域において放射性元素濃度の過飽和状態が出現し得る。

この点については、沈澱生成の反応速度を考慮したモデル化が行われており [Takase and Grindrod, 1994]、このモデルによれば反応の時定数が数週間以上である場合には過飽和によって核種放出率が有為に増加し得る。

Takase and Grindrod, (1994), Precipitation Kinetics of Radioactive Elements and Their Effect upon Redox Conditions of the Nearfield, Proceedings of the Fifth Annual International High Level Nuclear Waste Repository, Source Term Development, to be published

吸着サイトの飽和及び競合するイオンの影響

表面錯体生成モデルあるいはイオン交換モデル等の有限個の吸着サイトを考えたモデルでは、液相側の核種濃度が高い領域では吸着サイトが飽和に近づくにつれて吸着挙動の非線型性が現れることが知られている。また、この様な吸着挙動が生じた場合には、競合するイオン（必ずしも放射性のものとは限らない）が吸着サイトを専有してしまうことによって、注目する核種の吸着量が低下する可能性がある。

沈澱フロントの移動

崩壊系列を考えた場合、親核種の溶解度が娘核種のそれよりも一定以上大きい時には、崩壊によって親核種が娘核種に変わることにより緩衝材中においても娘核種の沈澱が形成され得る。またこの沈澱領域は、親核種の移行・崩壊に伴って徐々に拡大することが示されている[動力炉・核燃料開発事業団, 1992]。この場合には、娘核種に対する実効的な拡散長さが低下することとなり、極端な場合には沈澱が人工バリア外側に到る可能性も考えられる。なお、このような沈澱フロントの移動速度は、主として親核種の遅延特性に支配されることとなる。

動力炉・核燃料開発事業団, (1992), 高レベル放射性廃棄物地層処分研究開発の技術報告書—平成3 年度—

コロイド生成及び移動

ガラス固化体が溶解する際、核種は完全に溶解した形態の他に、 $10^{-9}\sim 10^{-6}\text{m}$ 程度の半径を持つ粒子(コロイド)の形態をもとり得る。これらの中には、核種自体から成る均質なもの(真性コロイド)と、非放射性的コロイドに核種が吸着した非均質なもの(疑似コロイド)とが含まれる。

一般に、ガラス固化体溶解に際しては、コロイドとしての核種溶出は不可避であると考えられており[Apted et al., 1986;Bjorner et al., 1989]、むしろ研究の焦点は、これらのコロイドがベントナイトの微小空隙中を濾過されることなく移動し得るかという点にある。

コロイドは、移行に際して、物理的濾過、吸着、解離といった種々の挙動を示し得る。ベントナイトの空隙が微小であり、かつ高い屈曲率を示すものであることから、物理的な濾過がコロイドとしての核種移行を強く制限することが推定されており、この推論は類似系として有機コロイドを用いた実験によって支持されている(コロイドとして移行する核種濃度は検出限界以下であった)が、実際の高圧縮ベントナイトにおける実験は未だ行われていない。

また、コロイドの半径が 10^{-8}m 以上である場合には、重力による沈降の速さが拡散による移動の速さを上回ることが計算されており、それよりも小さいコロイドについてはベントナイトの平均的な空隙径を沈降するのに約1年程度を要することが知られている[Apps et al., 1983]。

Apted et al., (1986), Release of Actinides from Defense Waste Glass under Simulated Repository Conditions, Nucl. Tech. Vol.73, pp.165-178

Bjorner et al., (1989), Corrosion of Radioactive, Crushed Waste Glass, in Scientific Basis for Nuclear Waste Management XII, ed. W.Lutze and R.Ewing, Ma

terials Research Society, Pittsburgh, USA, pp113-120

Apps, J. et al., (1983), Status of Geochemical Problems Relating to the Burial of High-Level Radioactive Waste, NUREG/ CR-3062 and LBL-15103, Lawrence Berkeley Laboratory, Berkeley, USA

表面拡散及び陰イオン排斥

高圧縮ベントナイト中の空隙径が極めて小さいものであり、かつ、ベントナイト粒子の表面は通常負に帯電していることから、空隙水のうち有意な割合は水分子の極性とベントナイト粒子の電荷との間の電気的力により配列したもの（表面水）となっている。表面水は自由水よりも粘性が大きく、特徴的な水理・物質移動特性を有するものと考えられている。そして、この表面水層中に存在する核種の拡散挙動は、表面拡散と呼ばれている。

一方、核種が陰イオンとして存在している場合には、これらがベントナイト粒子表面の負の電荷と反発しあうために、表面水層中から核種が排除されるという現象が生じ得る。この現象は、陰イオン排斥と呼ばれる。陰イオン排斥が生じた場合には、核種移行に実効的な空隙率は全空隙のそれよりも小さいものとなる。

放射性分解による酸化還元フロントの形成

ガラス固化体中の α 放射性核種の影響によって、地下水の放射線分解が生じた場合、生成される過酸化水素やその他のラディカルのために固化体近傍の地下水が酸化性となることが想定される。これらの酸化剤は、固化体から外側に向けて移行することとなるが、 α 線の飛程が小さく分解される地下水量に限られること及び現在の我が国の設計案ではこの過程で還元剤である腐食生成物層を通過することから、オーバーパックスの内側で消費されるものと考えられている[動力炉・核燃料開発事業団, 1992]。しかしながら、前出の沈澱反応の速度論的效果による核種濃度の過飽和が生じた場合には、オーバーパックスの外側に有意な量の α 核種が沈澱、吸着している可能性があり、ここでは空隙水のほぼ全てが分解に関与し得ることから、酸化還元フロントが形成される可能性のあることが示唆されている[Takase and Grindrod, 1994]。

動力炉・核燃料開発事業団, (1992), 高レベル放射性廃棄物地層処分研究開発の技術報告書—平成3年度—

Takase and Grindrod, (1994), Precipitation Kinetics of Radioactive Elements and Their Effect upon Redox Conditions of the Nearfield, Proceedings of the Fifth Annual International High Level Nuclear Waste Repository, Source Term Development, to be published

気候変化

気候変化のうち、温度変化自体が深部地下環境の条件に有意に影響を及ぼすことは考えにくい[McEwen and de Marsily, 1991]。また、降雨量の変化についても、処分場周辺を流れる地下水の流束を若干変化させる可能性はあるものの、その影響程度はさして大きいものとは考えにくい[McEwen and de Marsily, 1991]。

気候変化の影響として我が国において最も大きな影響が生じ得るものは、永久凍土層の形成である。我が国は、かつて大規模な氷河作用の影響を受けてはいないものと考えられているが、他方、北部日本については氷河の周辺部に位置していた可能性も指摘されており[Chorely et al., 1984]、永久凍土についてはその可能性を否定することは難しいものと考えられる。永久凍土層が形成された場合には、地下水流動パターンが大規模に変更される一方、地球化学条件についても高イオン強度の水への変化等が予想される。

我が国では大規模の(全世界的な)氷河期の影響のみが有意になると考えた場合、気候変化に関して考慮すべきタイムスケールは10万年程度となり、より小規模の氷河期を考えた場合には3万年程度を想定する必要がある。

McEwen and de Marsily, (1991), The Potential Significance of Permafrost to the Behaviour of a Deep Radioactive Repository, SKI Technical Report 91:8

Chorly et al., 1984, Geomorphology, Methuen, London

海面変動

過去70万年間の地球規模での気候変動とこれに伴う海面変動は、Milankovitchサイクルと呼ばれる10万年の周期に従って生じている。また、第四紀の我が国での海面変動の調査によって、この期間で最低の海水準は現在よりも約140m低く（約20万年前）、また、最高位置は現在よりもわずかに10~25m 高いに過ぎないことが明らかとなっている[Japan Association for the Quarternary Research, 1987]。また、以上に加えて、人間活動の影響による地球温暖化によっても今後数百年の間に海面の上昇が生じ得るものと推定されている[Warrick and Farmer, 1989]。

この様な天然及び人為を介した海面変動の結果として、現在の海水域に建設された地層処分場が淡水域に到ること、及び淡水域に建設された処分場が海水域に到ることの両者の可能性が考えられる。前者が生じた場合の主要な変化は、処分場周辺の動水勾配が増加することと、地下水の性質が海水起源の高イオン強度のものから淡水起源のものに変化することである。また、後者については、これと逆方向の変化が生じるものと推定される。

The Japan Association for Quarternary Research, 1987, Quarternary Maps of Japan II, Prehistoric Remains and Paleogeography, University of Tokyo Press

Warrick and Farmer, 1989, The Greenhouse effect and rising sea level. Manuscript, IUNCN. Release of the Sahel Status Report, Helsinki

付録-2

長期間にわたる核種移行バリアとしてのベントナイトの性能は、処分場の形状変化や、ベントナイト自体の持つ物性値の変化に依存する。ここでは、ベントナイトの岩盤亀裂からの流出に起因するベントナイトの透水性の変化、および、オーバパックの腐食膨張によるベントナイトの実効厚さの変化と言うプロセスをモデル化することによって、この依存性を定量化可能とした。

オーバパックを取り巻くベントナイトの密度の変化は、主に周辺岩盤の亀裂へのベントナイトの流出により起こるものと予想される。処分ピットでの亀裂分布の推定は現実的には処分地層に依存するが、本検討では、亀裂を含む岩盤を巨視的に多孔質媒体とみなした場合に推定できる地下水流量が同等となる様な亀裂開口幅及び亀裂頻度を推定し、検討を行なった。その際、亀裂の頻度を処分ピット間隔に1本と仮定した。廃棄体の処分において、亀裂間の幅がピット間隔より遥かに大きい場合（この場合には、上記の等価透水係数を保つため、亀裂の開口幅も必然的に大きくなる）、そのような亀裂はピット位置をずらすことにより容易に回避し得るものと考えられるからである。無限に続く平板亀裂と地下水流速の関係は、

$$V_{Fiss} = \frac{\rho g}{\eta} \frac{1}{2(S+b)} \frac{(2b)^3}{12} i$$

で表わされる[SNOW 1980]。ここで、

- V_{Fiss} : 亀裂内部地下水流速 [m/sec]
- ρ : 水密度 [=1000 kg/m³]
- g : 重力加速度 [=9.807 m/s²]
- η : 水粘性係数 [=0.001Ns/m²]
- $2S$: 亀裂間隔 [m]
- $2b$: 亀裂開口幅 [m]
- i : 動水勾配[-]

である。また、地層を多孔質媒体とみなした場合の地下水のダルシー流速 (V_{Darcy} [m/s]) は、地層の透水係数を K_{Pore} [m/s]、とした場合に、

$$V_{Darcy} = K_{Pore} i$$

であり、地下水流量を等価とすると、

$$V_{Darcy} = V_{Fiss} \frac{b}{S+b}$$

と表わせる。ここで、前述の様に、亀裂間隔をピット間隔と仮定することで、亀裂の開口幅を推定した。また、ここでの周辺岩盤の透水係数の設定は、ゆるみ領域（健全な岩の透水係数の10倍）と同等とした。更に、これで求められた亀裂にベントナイトが流出するものと仮定し、ベントナイト侵入の実験式[PUSCH1]、

$$X_{mig} = a b^2 \text{Log}(t+1)$$

ここで、

X_{mig} : 亀裂への侵入距離 [m]

a : 比例係数 [$5 \times 10^4 \sim 1 \times 10^6$, 1×10^6 を使用]

t : 時間 [y]

を使用して、膨脹体積を求めベントナイトの実効密度に変換した。更に、ベントナイトの乾燥密度と透水係数を求めた実験のデータを基に、ベントナイトの透水係数に変換した。その際、亀裂内部の地下水流速が 10^{-4} m/yを超える様な場合には、ベントナイト粒子の剥離が起こるものとして[PUSCH2]、ベントナイトの流出速度を地下水流速の変化する時点での流出速度に固定した。ベントナイト空隙率(Eps_{Bent} [-])と透水係数(K_{Bent} [m/s])の関係は以下の近似式を使用した。

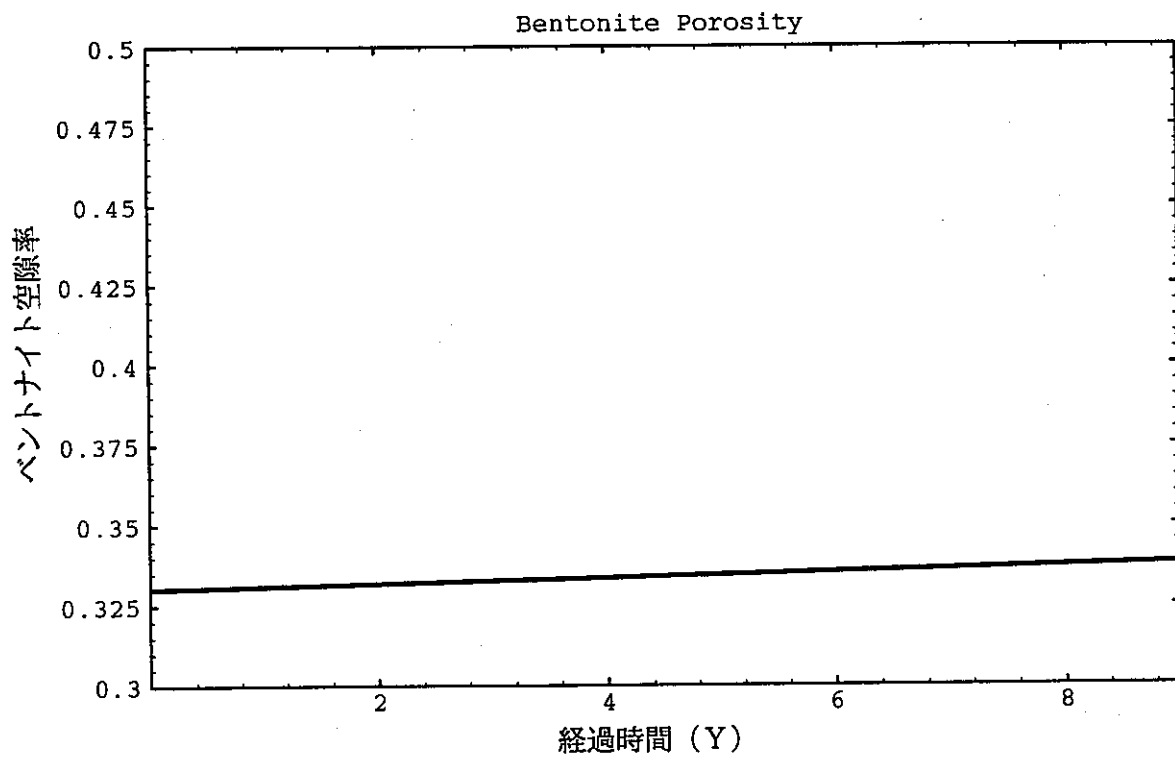
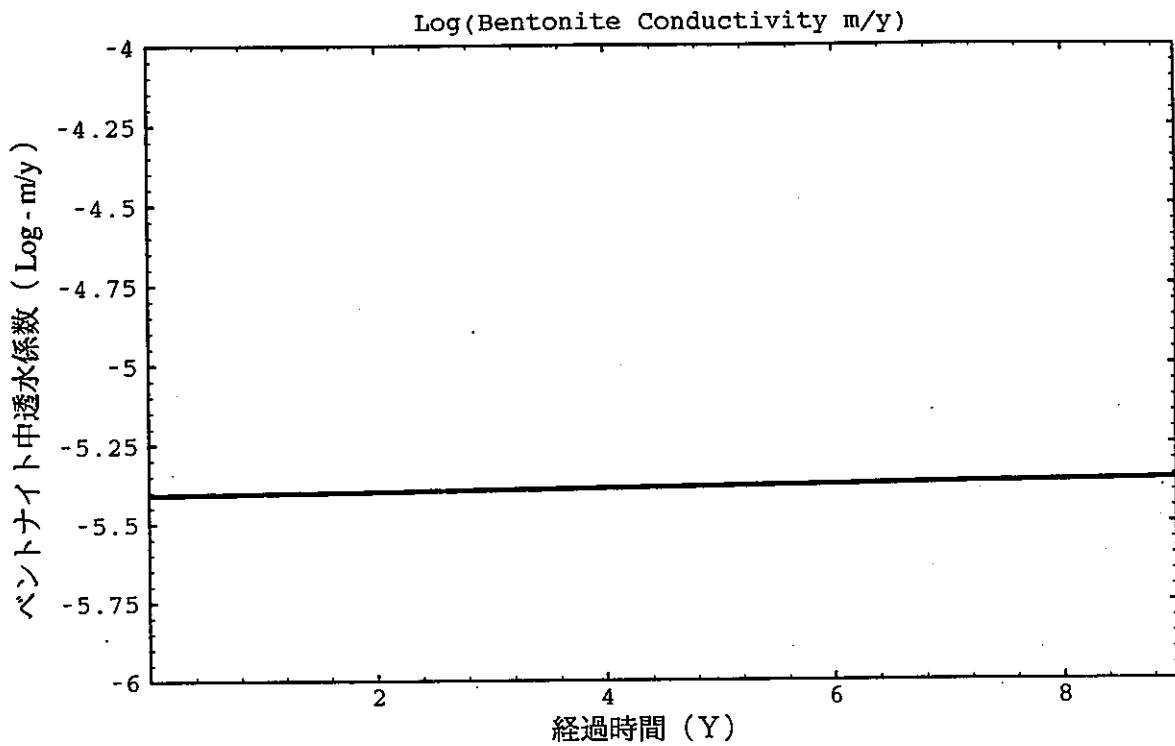
$$K_{Bent} = 10^{(-17.247 + \frac{8.280}{0.1+2.7(1-Eps_{Bent})})}$$

計算結果を次項移行に示す。通常予想される開口幅0.88mmは勿論、かなり保守的な数値である3.3mmの場合でも、緩衝材の透水係数上昇は移流支配につながるものではない。

周辺母岩透水係数 0.315 m/y

周辺動水勾配 0.05

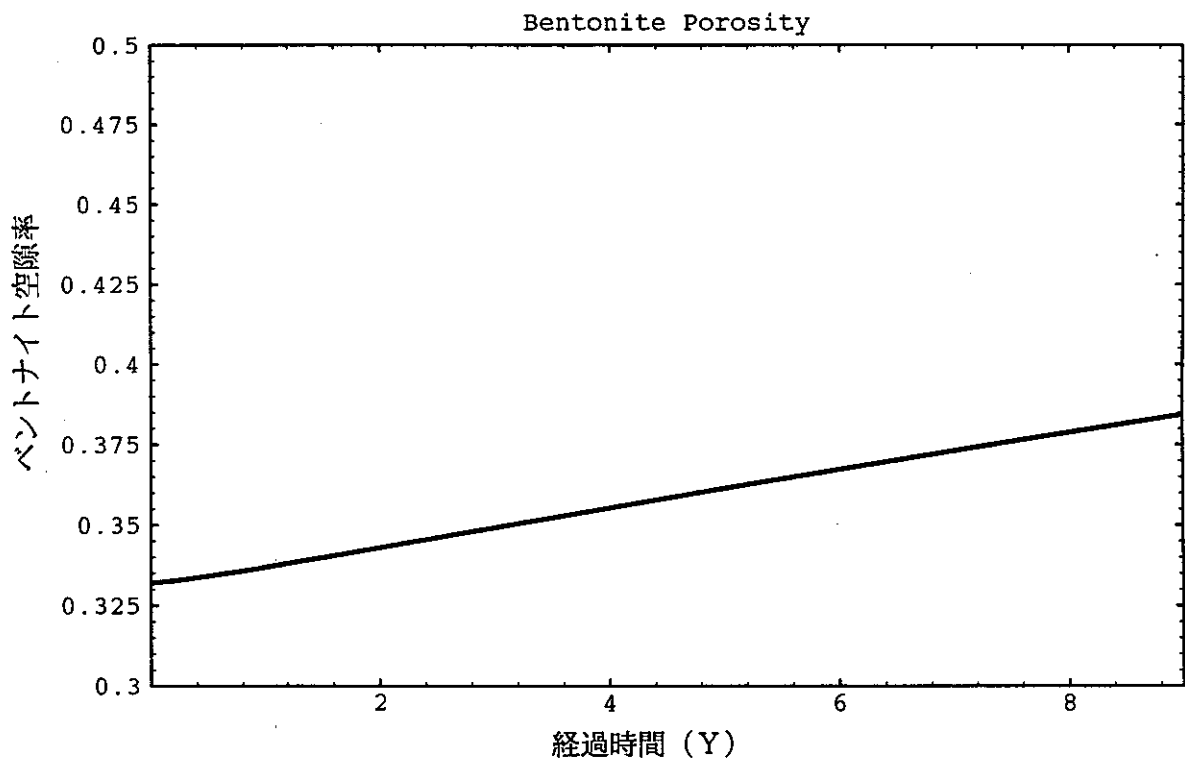
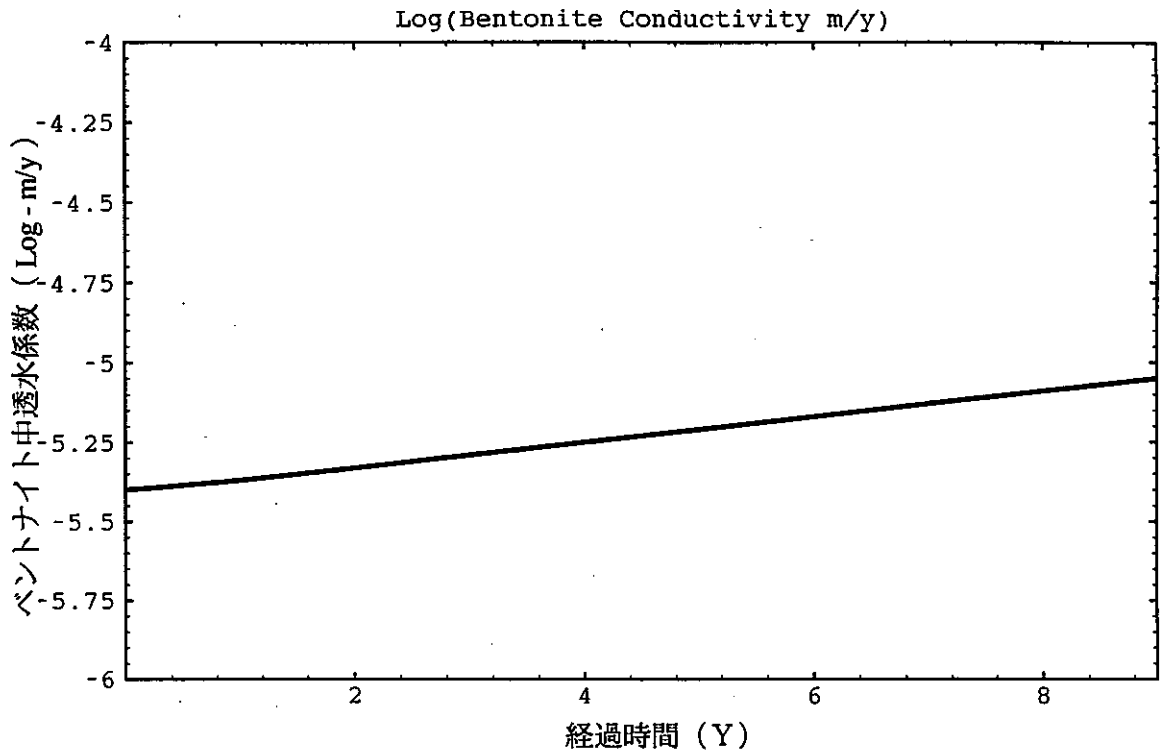
亀裂間隔を固化体配置間隔とする (亀裂開口幅 0.88mm、亀裂内流速 4.0×10^{-6} m/s)



周辺母岩透水係数 0.315 m/y

周辺動水勾配 0.05

亀裂間隔を100mとする (亀裂開口幅 3.3mm、亀裂内流速 1.5×10^{-5} m/s)



広義拡散方程式上で α を瞬時平衡、線型吸着、溶解度限界のモデルに対応するように設定したケース(標準ケース)について、以下の解析的な計算結果と比較することによる検証を行った。

- ・標準ケースにおける最大核種放出については、その上限は溶解度にゆるみ域の地下水流量を乗じたものとして算出される。また、流量が 0.71 l と小さいものであることから、標準ケースの最大放出率はこの数値に十分近いものであるべきである。この観点からは、標準ケースの結果が 6.94 E-13 であるのに対して、その上限値は 7.1 E-13 であり、SPADE の解析が精度良く行われていることを示している。
- ・定常状態においては、人工バリア外に放出されておりかつ崩壊していない核種の総量もまた定常となる。この定常時の系外残留核種量については厳密な値が計算可能であり、算出された数値である 2.1 E-7 mol は標準ケースでの計算結果と完全に一致する。

付録-4

SPADE に内蔵されたビルトイン関数のリストを以下に示す。

abs(x)... 絶対値

arccos(x)... 逆三角関数

arcsin(x)... "

arctan(x)... "

cos(x)

cosh(x)... 双曲線関数

d(a, x)... 空間についての一階微分

d2(a, x) ... " 二階微分

exp(x)

heavy(x)... ヘビサイドステップ関数

hsmooth(x, eps)... ヘビサイドステップ関数の連続関数による近似

log(x)

log10(x)

min(.)

max(.)

sin(x)

sqrt(x)

swich(x, a, b)... スイッチ関数(xが正の時はa、それ以外の場合はb という値をとる)

tan(x)

tanh(x)

なお、hsmooth(x)は以下の式で定義される。

$$\text{hsmooth}(x, \text{eps}) = 0 \quad (x < 0)$$

$$\exp(-x^2/\text{eps}^2) \quad (x \geq 0)$$

付録-5

決定論的感度解析の試行において用いたSPADE 入力ファイル(SPFファイル)の一部の例を以下に添付する。なお、各ケースでのパラメータ設定値については、次頁の一覧表を参照されたい。

SPFファイルのまとめ

No.	CASE NAME	CASE No.	kpcl	kpc2	ktcl	ktc2	rf2	qf	qw
1	central	1	0	0	0	0	2.8	0	-
2	colloids	1	0	0	0.1	0.1	2.8	0	-
3	glacier	1	0	0	0	0	2.8	1.00E+04	1.00E+05
4		2	0	0	0	0	2.8	1.00E+04	1.00E+04
5	mixcolloids	1	0	0.1	0.1	0.1	2.8	0	-
6		2	0	0.1	0.1	0.1	280	0	-
7	pocolloids	1	0	0.1	0	0	2.8	0	-
8		2	0	0.1	0	0	280	0	-
9	radiocolloids	1	0	0	0.1	0.1	2.8	-	-
10	radiolysis	1	0	0	0	0	2.8	-	-
11	saline	1	0	0	0	0	2.8	-	1.00E+04
12		2	0	0	0	0	2.8	-	1.00E+05

```

*
* central.spf
*
* SPF file for Task 1.1 central case (IM3246)
*
* This file models the diffusive transport of Technetium Tc-99 through the
* near field of a glass/overpack corrosion product/bentonite system.
*
* JAW 02/02/94
*
* Version 1.8
*
* Modification history:
* Vn Who When What
* 1.0 JAW 02/02/94 First version
* 1.1 JAW 03/02/94 Improve variable scaling and layout of file
* 1.2 JAW 04/02/94 Add colloids and pseudo-colloids
* 1.3 JAW 08/02/94 Further minor layout changes and added "outside"
* 1.4 JAW 10/02/94 Add monitor variable drop_time
* 1.5 JAW 11/02/94 Add monitor variable peak_outside
* 1.6 JAW 14/02/94 Improve inner boundary condition
* 1.7 JAW 15/02/94 Further change to inner boundary condition
*                   + fixed error in (7.2)
* 1.8 JAW 22/02/94 Corrected a number of small errors
*
* General notes:
* (a) Units.
* The time unit is years. Distances are in metres. Amount of contaminant
* remaining in the glass or escaping from the system is measured in moles
* of Tc-99.
* Concentrations of contaminant in the transported species are measured in
* moles per cubic metre *of water* multiplied by a scaling factor ascale.
* This factor serves to improve the numerical conditioning of the problem:
* without it typical transported concentrations would be many orders of
* magnitude smaller than the remaining inventory.
*
* (b) Geometry.
* The problem is posed in a 1D cylindrical coordinate system, assuming
* cylindrical symmetry. The height of the cylinder is taken as the height
* of the central glass waste canister.
* There are three distinct regions in the model:
* Region Inner Outer
* radius (m) radius (m)
* Glass 0 0.22
* Overpack 0.22 0.52
* Bentonite 0.52 1.50
* The glass is treated as a source term which forms the inner boundary
* condition.
* It is assumed that contaminant from the whole volume of the glass is
* instantaneously in equilibrium with the surrounding water.
* The overpack (strictly, overpack corrosion products) layer and bentonite
* layer are treated as two separate regions with a flux condition at
* their shared boundary.
* The outer boundary of the bentonite is a decompressed zone. This is
* represented by a mixing-cell treatment which forms the outer boundary
* condition.
*
* (Block 1) Control info.
*
LISTFILE; * generate list output
SPARSE; * use sparse matrix solver
ALGERR; * include errors on algebraic variables in timestep monitor
CONSTANT epsstep=1e-5; * value for absolute error tolerance per timestep
EPSSTEP epsstep; * set absolute error tolerance
INITSTEP 1.e-3; * initial timestep
MINTIME 10; * Set output timesteps (in years)
MAXTIME 1e7; * - write out every 10000 years
GEOMSTEP 1.258925412; * from t=0 to t=1e7 years
*ARITHSTEP 1e5; * from t=0 to t=1e7 years
*
* (Block 2) Fundamental constants
*
CONSTANT pi=4*arctan(1);
CONSTANT SecsPerYear=365.2422*24*60*60; * number of seconds in one year
CONSTANT ascale=1e10; * multiplicative factor for all molar concentrations
* in transported species - improves conditioning
CONSTANT thalf=2.1e5; * Half-life of Tc-99 in years
CONSTANT Kdecay=log(2)/thalf; * Equivalent decay constant
CONSTANT Cstar=1e-9*ascale; * Solubility limit of Tc-99 in mol/m3

```

```

CONSTANT R1=1; * Retardation factor of Tc-99 in inner (overpack) layer
CONSTANT R2=2.8; * Retardation factor of Tc-99 in outer (bentonite) layer
CONSTANT Rglass=1; * Retardation factor of Tc-99 in glass
CONSTANT Dsol=3e-10*SecsPerYear; * diffusion coefficient of Tc solute in
* both bentonite and overpack
CONSTANT Dpc1=Dsol/10; * diffusion coefficient of Tc pseudocolloid
* in overpack
CONSTANT Dpc2=Dsol/10; * diffusion coefficient of Tc pseudocolloid
* in bentonite
CONSTANT Dtc1=Dsol/2; * diffusion coefficient of Tc true colloid
* in overpack
CONSTANT Dtc2=Dsol/2; * diffusion coefficient of Tc true colloid
* in bentonite
CONSTANT thetag=0.33; * Porosity of glass
CONSTANT theta1=0.33; * Porosity of overpack
CONSTANT theta2=0.33; * Porosity of bentonite
CONSTANT ZTc=99; * atomic weight of Tc
CONSTANT invmass=1.06e3; * Inventory at calculation start in g
CONSTANT invmoles=invmass/ZTc; * Initial inventory in moles
CONSTANT rglass=0.22; * outer glass radius
CONSTANT rover1=0.52; * actual outer overpack radius
CONSTANT rbent1=1.50; * actual outer bentonite radius
CONSTANT h=1.35; * height of cylinder
CONSTANT vglass=pi*h*rglass*rglass*thetag; * volume of glass
*
* (Block 3) Mixing cell constants
*
CONSTANT q0=0.71*1e-3; * mean flow through mixing cell in m3/year
CONSTANT qfrac=0; * fractional variation in mixing cell flow, used to
* derive q - central case value is qfrac=0
CONSTANT q1=q0*qfrac; * amplitude of variation in q
CONSTANT Tq=2000; * period of variation in q (in years)
CONSTANT wq=2*pi/Tq; * equivalent angular frequency
SIDE q=q0+q1*sin(wq*t); * mixing cell flow rate (potentially time dependent)
*
* (Block 4) Define geometry.
*
* The problem is posed in 1D cylindrical coordinates. Although
* Spade does not understand about these at present, we can use
* 1D cartesian coordinates because we discretize the
* spatial derivatives ourselves.
*
* The spatial variables Rin1 and Rin2 are chosen to represent the INNER
* radius of each ring. We therefore set the grid spacing so that the
* OUTER radius of the outermost cylinder in each region matches the
* specified dimensions.
*
CONSTANT nover=5; * number of grid points in overpack
CONSTANT nbent=10; * number of grid points in bentonite
CONSTANT rover=rglass+(rover1-rglass)*(nover-1)/nover; * grid limit
* for overpack
CONSTANT rbent=rover1+(rbent1-rover1)*(nbent-1)/nbent; * grid limit
* for bentonite
CONSTANT dr1=(rover1-rglass)/nover; * grid spacing in overpack
CONSTANT dr2=(rbent1-rover1)/nbent; * grid spacing in bentonite
SPATIAL Rin1,UNIFORM,rglass,rover,nover; * INNER radius of each ring
SPATIAL Rin2,UNIFORM,rover1,rbent,nbent;
REGION over,CART1D,Rin1; * overpack grid
REGION bent,CART1D,Rin2; * bentonite grid
VECTOR Rout1[over]; * OUTER radius of each ring in overpack
VECTOR Rout2[bent]; * OUTER radius of each ring in bentonite
CONSTANT Rout1=Rin1+dr1;
CONSTANT Rout2=Rin2+dr2;
VECTOR V1[over]; * volume of WATER in each ring in overpack
VECTOR V2[bent]; * volume of WATER in each ring in bentonite
CONSTANT V1=theta1*pi*h*((Rout1*Rout1)-(Rin1*Rin1));
CONSTANT V2=theta2*pi*h*((Rout2*Rout2)-(Rin2*Rin2));
VECTOR SA1[over]; * OUTER surface area of each ring in overpack
VECTOR SA2[bent]; * OUTER surface area of each ring in bentonite
CONSTANT SA1=2*pi*h*Rout1;
CONSTANT SA2=2*pi*h*Rout2;
CONSTANT SAglass=2*pi*h*rglass; * outer surface area of glass
SIDE Cglass=ascale*inventory/vglass; * concentration in glass
*
* (Block 5) Declare remaining vector quantities
*
* the concentration vector within each region
VECTOR C1[over];
VECTOR C2[bent];

```

```

* the effective diffusivity
VECTOR alpha1[over];
VECTOR alpha2[bent];
* the flux across each outer boundary
VECTOR fluxout1[over];
VECTOR fluxout2[bent];
*
* (Block 6) Define alpha. The form of alpha varies depending on the assumed
* distribution of mobile species as a function of concentration.
* For the central case with no colloids or pseudocolloids, k1=k2=kpc1=kpc2=0.
*
CONSTANT k1=0; * fraction of true colloid in overpack
CONSTANT k2=0; * fraction of true colloid in bentonite
CONSTANT kpc1=0; * fraction of pseudocolloid in overpack
CONSTANT kpc2=0; * fraction of pseudocolloid in bentonite
* (6.1) alpha in overpack
SIDE alpha1=switch(C1-Cstar*R1,
                  Dsol/R1+kpc1*(R1-1)*Dpc1/R1,
                  (Dsol*Cstar+kpc1*(R1-1)*Cstar*Dpc1+k1*Dtc1*(C1-Cstar*R1))/C1);
* (6.2) alpha in bentonite
SIDE alpha2=switch(C2-Cstar*R2,
                  Dsol/R2+kpc2*(R2-1)*Dpc2/R2,
                  (Dsol*Cstar+kpc2*(R2-1)*Cstar*Dpc2+k2*Dtc2*(C2-Cstar*R2))/C2);

* (6.3) alpha in glass
SIDE alphaglass=switch(Cglass-Cstar*Rglass,
                      Dsol/Rglass+kpc1*(Rglass-1)*Dpc1/Rglass,
                      (Dsol*Cstar+kpc1*(Rglass-1)*Cstar*Dpc1+k1*Dtc1*(Cglass-Cstar*Rglass))/Cglass);
*
* (Block 7) Define fluxout, which gives the flux of Tc across each outer
* face (positive = net outward flux).
* The outer boundary and the boundary between the overpack and bentonite
* have to be handled separately.
*
* (7.1) Flow through overpack
SIDE fluxout1[Rin1<rover]=--SA1*(C1[Rin1=Rin1+dr1]*alpha1[Rin1=Rin1+dr1] -
                               C1*alpha1)/dr1;
* (7.2) Flow through bentonite
SIDE fluxout2[Rin2<rbent]=--SA2*(C2[Rin2=Rin2+dr2]*alpha2[Rin2=Rin2+dr2] -
                               C2*alpha2)/dr2;
* (7.3) Flow from overpack to bentonite
SIDE fluxout1[Rin1=rover]=--SA1*(C2[Rin2=rover1]*alpha2[Rin2=rover1] -
                               C1*alpha1)/(0.5*(dr1+dr2));
* (7.4) Flow from outermost cell - mixing cell approximation
* (q may be a function of time)
SIDE fluxout2[Rin2=rbent]=switch(C2-Cstar*R2, C2/R2+kpc2*(R2-1)*C2/R2,
                                Cstar+kpc2*(R2-1)*Cstar+k2*(C2-Cstar*R2))*q;
* (7.5) Flux from glass into innermost cell of overpack
SIDE fluxglass=-SAglass*(C1[Rin1=rglass]*alpha1[Rin1=rglass]
                        -Cglass*alphaglass)/((dr1+rglass)/2);
*
* (Block 8) Evolution equations
*
* (8.1) Tc in glass decays and is transported out into overpack inner ring
EQUATION d(inventory, t)=-Kdecay*inventory-(fluxglass/ascale);
* (8.2) Innermost ring of overpack receives influx from glass
EQUATION d(C1, t)=- (Kdecay*C1[Rin1<=rglass]+(fluxout1[Rin1=rglass]-fluxglass)
                    / V1);
* (8.3) Transport through overpack
EQUATION d(C1, t)=- (Kdecay*C1+(fluxout1-fluxout1[Rin1=Rin1-dr1, Rin1>rglass])
                    / V1);
* (8.4) Transport from overpack to bentonite
EQUATION d(C2, t)=- (Kdecay*C2[Rin2>rover1]+
                    (fluxout2-fluxout2[Rin2=Rin2-dr2])/V2);
* (8.5) Transport through bentonite (including loss from outermost ring)
EQUATION d(C2, t)=- (Kdecay*C2+
                    (fluxout2[Rin2=rover1]-fluxout1[Rin1=rover])/V2);
* (8.6) Amount escaping is equal to loss from outermost ring
EQUATION d(escaped, t)=fluxout2[Rin2=rbent]/ascale;
* (8.7) Amount escaped and surviving is equal to amount escaped less decay
EQUATION d(outside, t)=fluxout2[Rin2=rbent]/ascale-Kdecay*outside;
*
* (Block 9) Initial values
*
INITIAL inventory=invmoles;
INITIAL C1=0;
INITIAL C2=0;
INITIAL escaped=0;
INITIAL outside=0;

```

```

INITIAL q=q0+q1*sin(wq*t);
*
* (Block 10) Weights - chosen to give correct relative scaling for variables
*
WEIGHT C1=1/(C1+Cstar);
WEIGHT C2=1/(C2+Cstar);
WEIGHT inventory=1/(inventory+Cstar*vglass/ascale);
WEIGHT escaped=1/(escaped+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
WEIGHT outside=1/(outside+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
*
* (Block 11) Monitor time and magnitude of peak flux and time of
* "drop off"
*
* The current escaping flux is only considered to be a new peak value if it
* is "significantly" larger than the previous peak,
* i.e. if it exceeds it by more than cutoff times the solver error tolerance.
* The monitor variable drop_time records the time at which the escaping
* flux first drops below a fixed fraction (e.g. 95%) of the peak flux.
* The monitor variable peak_outside records the maximum amount of
* contaminant surviving outside at any time.
*
CONSTANT cutoff=3; * multiple of epsstep required for significance
MONITOR peak_time=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        t, peak_time);
INITIAL peak_time=t;
MONITOR peak_flux=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        fluxout2[Rin2=rbent]/ascale, peak_flux);
INITIAL peak_flux=0;
CONSTANT drop_frac=0.95; * fraction of peak flux
MONITOR drop_time=switch(peak_flux*drop_frac-(fluxout2[Rin2=rbent]/ascale),
                        t, drop_time);
INITIAL drop_time=t;
MONITOR peak_outside=max(outside, peak_outside);
INITIAL peak_outside=0;
*
* (End of central.spf)
*

```

```

*
* central.spf
*
* SPF file for Task 1.1 central case (IM3246)
*
* This file models the diffusive transport of Technetium Tc-99 through the
* near field of a glass/overpack corrosion product/bentonite system.
*
* JAW 02/02/94
*
* Version 1.8
*
* Modification history:
* Vn Who When What
* 1.0 JAW 02/02/94 First version
* 1.1 JAW 03/02/94 Improve variable scaling and layout of file
* 1.2 JAW 04/02/94 Add colloids and pseudo-colloids
* 1.3 JAW 08/02/94 Further minor layout changes and added "outside"
* 1.4 JAW 10/02/94 Add monitor variable drop_time
* 1.5 JAW 11/02/94 Add monitor variable peak_outside
* 1.6 JAW 14/02/94 Improve inner boundary condition
* 1.7 JAW 15/02/94 Further change to inner boundary condition
* + fixed error in (7.2)
* 1.8 JAW 22/02/94 Corrected a number of small errors
*
* General notes:
* (a) Units.
* The time unit is years. Distances are in metres. Amount of contaminant
* remaining in the glass or escaping from the system is measured in moles
* of Tc-99.
* Concentrations of contaminant in the transported species are measured in
* moles per cubic metre *of water* multiplied by a scaling factor ascale.
* This factor serves to improve the numerical conditioning of the problem:
* without it typical transported concentrations would be many orders of
* magnitude smaller than the remaining inventory.
*
* (b) Geometry.
* The problem is posed in a 1D cylindrical coordinate system, assuming
* cylindrical symmetry. The height of the cylinder is taken as the height
* of the central glass waste canister.
* There are three distinct regions in the model:
* Region Inner Outer
* radius (m) radius (m)
* Glass 0 0.22
* Overpack 0.22 0.52
* Bentonite 0.52 1.50
* The glass is treated as a source term which forms the inner boundary
* condition.
* It is assumed that contaminant from the whole volume of the glass is
* instantaneously in equilibrium with the surrounding water.
* The overpack (strictly, overpack corrosion products) layer and bentonite
* layer are treated as two separate regions with a flux condition at
* their shared boundary.
* The outer boundary of the bentonite is a decompressed zone. This is
* represented by a mixing-cell treatment which forms the outer boundary
* condition.
*
* (Block 1) Control info.
*
LISTFILE: * generate list output
SPARSE: * use sparse matrix solver
ALGERR: * include errors on algebraic variables in timestep monitor
CONSTANT epsstep=1e-5; * value for absolute error tolerance per timestep
EPSSTEP epsstep; * set absolute error tolerance
INITSTEP 1.e-3; * initial timestep
MINTIME 10; * Set output timesteps (in years)
MAXTIME 1e7; * - write out every 10000 years
GEOMSTEP 1.258925412; * from t=0 to t=1e7 years
*ARITHSTEP 1e5; * from t=0 to t=1e7 years
*
* (Block 2) Fundamental constants
*
CONSTANT pi=4*arctan(1);
CONSTANT SecsPerYear=365.2422*24*60*60; * number of seconds in one year
CONSTANT ascale=1e10; * multiplicative factor for all molar concentrations
* in transported species - improves conditioning
CONSTANT thalf=2.1e5; * Half-life of Tc-99 in years
CONSTANT Kdecay=log(2)/thalf; * Equivalent decay constant
CONSTANT Cstar=1e-9*ascale; * Solubility limit of Tc-99 in mol/m3

```



```

CONSTANT R1=1; * Retardation factor of Tc-99 in inner (overpack) layer
CONSTANT R2=2.8; * Retardation factor of Tc-99 in outer (bentonite) layer
CONSTANT Rglass=1; * Retardation factor of Tc-99 in glass
CONSTANT Dsol=3e-10*SecsPerYear; * diffusion coefficient of Tc solute in
* both bentonite and overpack
CONSTANT Dpc1=Dsol/10; * diffusion coefficient of Tc pseudocolloid
* in overpack
CONSTANT Dpc2=Dsol/10; * diffusion coefficient of Tc pseudocolloid
* in bentonite
CONSTANT Dtc1=Dsol/2; * diffusion coefficient of Tc true colloid
* in overpack
CONSTANT Dtc2=Dsol/2; * diffusion coefficient of Tc true colloid
* in bentonite
CONSTANT thetag=0.33; * Porosity of glass
CONSTANT theta1=0.33; * Porosity of overpack
CONSTANT theta2=0.33; * Porosity of bentonite
CONSTANT ZTc=99; * atomic weight of Tc
CONSTANT invmass=1.06e3; * Inventory at calculation start in g
CONSTANT invmoles=invmass/ZTc; * Initial inventory in moles
CONSTANT rglass=0.22; * outer glass radius
CONSTANT rover1=0.52; * actual outer overpack radius
CONSTANT rbent1=1.50; * actual outer bentonite radius
CONSTANT h=1.35; * height of cylinder
CONSTANT vglass=pi*h*rglass*rglass*thetag; * volume of glass
*
* (Block 3) Mixing cell constants
*
CONSTANT q0=0.71*1e-3; * mean flow through mixing cell in m3/year
CONSTANT qfrac=0; * fractional variation in mixing cell flow, used to
* derive q - central case value is qfrac=0
CONSTANT q1=q0*qfrac; * amplitude of variation in q
CONSTANT Tq=2000; * period of variation in q (in years)
CONSTANT wq=2*pi/Tq; * equivalent angular frequency
SIDE q=q0+q1*sin(wq*t); * mixing cell flow rate (potentially time dependent)
*
* (Block 4) Define geometry.
*
* The problem is posed in 1D cylindrical coordinates. Although
* Spade does not understand about these at present, we can use
* 1D cartesian coordinates because we discretize the
* spatial derivatives ourselves.
*
* The spatial variables Rin1 and Rin2 are chosen to represent the INNER
* radius of each ring. We therefore set the grid spacing so that the
* OUTER radius of the outermost cylinder in each region matches the
* specified dimensions.
*
CONSTANT nover=5; * number of grid points in overpack
CONSTANT nbent=10; * number of grid points in bentonite
CONSTANT rover=rglass+(rover1-rglass)*(nover-1)/nover; * grid limit
* for overpack
CONSTANT rbent=rover1+(rbent1-rover1)*(nbent-1)/nbent; * grid limit
* for bentonite
CONSTANT dr1=(rover1-rglass)/nover; * grid spacing in overpack
CONSTANT dr2=(rbent1-rover1)/nbent; * grid spacing in bentonite
SPATIAL Rin1,UNIFORM,rglass,rover,nover; * INNER radius of each ring
SPATIAL Rin2,UNIFORM,rover1,rbent,nbent;
REGION over,CART1D,Rin1; * overpack grid
REGION bent,CART1D,Rin2; * bentonite grid
VECTOR Rout1[over]; * OUTER radius of each ring in overpack
VECTOR Rout2[bent]; * OUTER radius of each ring in bentonite
CONSTANT Rout1=Rin1+dr1;
CONSTANT Rout2=Rin2+dr2;
VECTOR V1[over]; * volume of WATER in each ring in overpack
VECTOR V2[bent]; * volume of WATER in each ring in bentonite
CONSTANT V1=theta1*pi*h*((Rout1*Rout1)-(Rin1*Rin1));
CONSTANT V2=theta2*pi*h*((Rout2*Rout2)-(Rin2*Rin2));
VECTOR SA1[over]; * OUTER surface area of each ring in overpack
VECTOR SA2[bent]; * OUTER surface area of each ring in bentonite
CONSTANT SA1=2*pi*h*Rout1;
CONSTANT SA2=2*pi*h*Rout2;
CONSTANT SAglass=2*pi*h*rglass; * outer surface area of glass
SIDE Cglass=ascale*inventory/vglass; * concentration in glass
*
* (Block 5) Declare remaining vector quantities
*
* the concentration vector within each region
VECTOR C1[over];
VECTOR C2[bent];

```

```

* the effective diffusivity
VECTOR alpha1[over];
VECTOR alpha2[bent];
* the flux across each outer boundary
VECTOR fluxout1[over];
VECTOR fluxout2[bent];
*
* (Block 6) Define alpha. The form of alpha varies depending on the assumed
* distribution of mobile species as a function of concentration.
* For the central case with no colloids or pseudocolloids, k1=k2=kpc1=kpc2=0.
*
CONSTANT k1=0.1; * fraction of true colloid in overpack
CONSTANT k2=0.1; * fraction of true colloid in bentonite
CONSTANT kpc1=0; * fraction of pseudocolloid in overpack
CONSTANT kpc2=0; * fraction of pseudocolloid in bentonite
* (6.1) alpha in overpack
SIDE alpha1=switch(C1-Cstar*R1,
    Dsol/R1+kpc1*(R1-1)*Dpc1/R1,
    (Dsol*Cstar+kpc1*(R1-1)*Cstar*Dpc1+k1*Dtc1*(C1-Cstar*R1))/C1);
* (6.2) alpha in bentonite
SIDE alpha2=switch(C2-Cstar*R2,
    Dsol/R2+kpc2*(R2-1)*Dpc2/R2,
    (Dsol*Cstar+kpc2*(R2-1)*Cstar*Dpc2+k2*Dtc2*(C2-Cstar*R2))/C2);
* (6.3) alpha in glass
SIDE alphaglass=switch(Cglass-Cstar*Rglass,
    Dsol/Rglass+kpc1*(Rglass-1)*Dpc1/Rglass,
    (Dsol*Cstar+kpc1*(Rglass-1)*Cstar*Dpc1+k1*Dtc1*(Cglass-Cstar*Rglass))/Cglass);
*
* (Block 7) Define fluxout, which gives the flux of Tc across each outer
* face (positive = net outward flux).
* The outer boundary and the boundary between the overpack and bentonite
* have to be handled separately.
*
* (7.1) Flow through overpack
SIDE fluxout1[Rin1<rover]= -SA1*(C1[Rin1=Rin1+dr1]*alpha1[Rin1=Rin1+dr1] -
    C1*alpha1)/dr1;
* (7.2) Flow through bentonite
SIDE fluxout2[Rin2<rbent]= -SA2*(C2[Rin2=Rin2+dr2]*alpha2[Rin2=Rin2+dr2] -
    C2*alpha2)/dr2;
* (7.3) Flow from overpack to bentonite
SIDE fluxout1[Rin1=rover]= -SA1*(C2[Rin2=rover1]*alpha2[Rin2=rover1] -
    C1*alpha1)/(0.5*(dr1+dr2));
* (7.4) Flow from outermost cell - mixing cell approximation
* (q may be a function of time)
SIDE fluxout2[Rin2=rbent]=switch(C2-Cstar*R2, C2/R2+kpc2*(R2-1)*C2/R2,
    Cstar+kpc2*(R2-1)*Cstar+k2*(C2-Cstar*R2))*q;
* (7.5) Flux from glass into innermost cell of overpack
SIDE fluxglass= -SAglass*(C1[Rin1=rglass]*alpha1[Rin1=rglass]
    -Cglass*alphaglass)/((dr1+rglass)/2);
*
* (Block 8) Evolution equations
*
* (8.1) Tc in glass decays and is transported out into overpack inner ring
EQUATION d(inventory, t)= -Kdecay*inventory - (fluxglass/ascale);
* (8.2) Innermost ring of overpack receives influx from glass
EQUATION d(C1, t)= -(Kdecay*C1[Rin1<=rglass] + (fluxout1[Rin1=rglass] - fluxglass)
    / V1);
* (8.3) Transport through overpack
EQUATION d(C1, t)= -(Kdecay*C1 + (fluxout1 - fluxout1[Rin1=Rin1-dr1, Rin1>rglass])
    / V1);
* (8.4) Transport from overpack to bentonite
EQUATION d(C2, t)= -(Kdecay*C2[Rin2>rover1] +
    (fluxout2 - fluxout2[Rin2=Rin2-dr2])/V2);
* (8.5) Transport through bentonite (including loss from outermost ring)
EQUATION d(C2, t)= -(Kdecay*C2 +
    (fluxout2[Rin2=rover1] - fluxout1[Rin1=rover])/V2);
* (8.6) Amount escaping is equal to loss from outermost ring
EQUATION d(escaped, t)= fluxout2[Rin2=rbent]/ascale;
* (8.7) Amount escaped and surviving is equal to amount escaped less decay
EQUATION d(outside, t)= fluxout2[Rin2=rbent]/ascale - Kdecay*outside;
*
* (Block 9) Initial values
*
INITIAL inventory=invmoles;
INITIAL C1=0;
INITIAL C2=0;
INITIAL escaped=0;
INITIAL outside=0;

```

```

INITIAL q=q0+q1*sin(wq*t);
*
* (Block 10) Weights - chosen to give correct relative scaling for variables
*
WEIGHT C1=1/(C1+Cstar);
WEIGHT C2=1/(C2+Cstar);
WEIGHT inventory=1/(inventory+Cstar*vglass/ascale);
WEIGHT escaped=1/(escaped+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
WEIGHT outside=1/(outside+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
*
* (Block 11) Monitor time and magnitude of peak flux and time of
* "drop off"
*
* The current escaping flux is only considered to be a new peak value if it
* is "significantly" larger than the previous peak,
* i.e. if it exceeds it by more than cutoff times the solver error tolerance.
* The monitor variable drop_time records the time at which the escaping
* flux first drops below a fixed fraction (e.g. 95%) of the peak flux.
* The monitor variable peak_outside records the maximum amount of
* contaminant surviving outside at any time.
*
CONSTANT cutoff=3; * multiple of epsstep required for significance
MONITOR peak_time=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        t,peak_time);
INITIAL peak_time=t;
MONITOR peak_flux=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        fluxout2[Rin2=rbent]/ascale,peak_flux);
INITIAL peak_flux=0;
CONSTANT drop_frac=0.95; * fraction of peak flux
MONITOR drop_time=switch(peak_flux*drop_frac-(fluxout2[Rin2=rbent]/ascale),
                        t,drop_time);
INITIAL drop_time=t;
MONITOR peak_outside=max(outside,peak_outside);
INITIAL peak_outside=0;
*
* (End of central.spf)
*

```

```

*
* central.spf
*
* SPF file for Task 1.1 central case (IM3246)
*
* This file models the diffusive transport of Technetium Tc-99 through the
* near field of a glass/overpack corrosion product/bentonite system.
*
* JAW 02/02/94
*
* Version 1.7
*
* Modification history:
* Vn Who When What
* 1.0 JAW 02/02/94 First version
* 1.1 JAW 03/02/94 Improve variable scaling and layout of file
* 1.2 JAW 04/02/94 Add colloids and pseudo-colloids
* 1.3 JAW 08/02/94 Further minor layout changes and added "outside"
* 1.4 JAW 10/02/94 Add monitor variable drop_time
* 1.5 JAW 11/02/94 Add monitor variable peak_outside
* 1.6 JAW 14/02/94 Improve inner boundary condition
* 1.7 JAW 15/02/94 Further change to inner boundary condition
*                   + fixed error in (7.2)
*
* General notes:
* (a) Units.
* The time unit is years. Distances are in metres. Amount of contaminant
* remaining in the glass or escaping from the system is measured in moles
* of Tc-99.
* Concentrations of contaminant in the transported species are measured in
* moles per cubic metre *of water* multiplied by a scaling factor ascale.
* This factor serves to improve the numerical conditioning of the problem:
* without it typical transported concentrations would be many orders of
* magnitude smaller than the remaining inventory.
*
* (b) Geometry.
* The problem is posed in a 1D cylindrical coordinate system, assuming
* cylindrical symmetry. The height of the cylinder is taken as the height
* of the central glass waste canister.
* There are three distinct regions in the model:
* Region Inner Outer
* radius (m) radius (m)
* Glass 0 0.22
* Overpack 0.22 0.52
* Bentonite 0.52 1.50
* The glass is treated as a source term which forms the inner boundary
* condition.
* It is assumed that contaminant from the whole volume of the glass is
* instantaneously in equilibrium with the surrounding water.
* The overpack (strictly, overpack corrosion products) layer and bentonite
* layer are treated as two separate regions with a flux condition at
* their shared boundary.
* The outer boundary of the bentonite is a decompressed zone. This is
* represented by a mixing-cell treatment which forms the outer boundary
* condition.
*
* (Block 1) Control info.
*
LISTFILE; * generate list output
SPARSE; * use sparse matrix solver
ALGERR; * include errors on algebraic variables in timestep monitor
CONSTANT epsstep=1e-5; * value for absolute error tolerance per timestep
EPSSTEP epsstep; * set absolute error tolerance
INITSTEP 1.e-3; * initial timestep
MINTIME 0; * Set output timesteps (in years)
MAXTIME 6e6; * - write out every 10000 years
ARITHSTEP 1e5; * from t=0 to t=1e7 years
*
* (Block 2) Fundamental constants
*
CONSTANT pi=4*arctan(1);
CONSTANT SecsPerYear=365.2422*24*60*60; * number of seconds in one year
CONSTANT ascale=1e10; * multiplicative factor for all molar concentrations
* in transported species - improves conditioning
CONSTANT thalf=2.1e5; * Half-life of Tc-99 in years
CONSTANT Kdecay=log(2)/thalf; * Equivalent decay constant
CONSTANT Cstar=1e-9*ascale; * Solubility limit of Tc-99 in mol/m3
CONSTANT R1=1; * Retardation factor of Tc-99 in inner (overpack) layer
CONSTANT R2=2.8; * Retardation factor of Tc-99 in outer (bentonite) layer

```

```

CONSTANT Rglass=1; * Retardation factor of Tc-99 in glass
CONSTANT Dsol=3e-10*SecsPerYear; * diffusion coefficient of Tc solute in
* both bentonite and overpack
CONSTANT Dpc1=Dsol/2; * diffusion coefficient of Tc pseudocolloid
* in overpack
CONSTANT Dpc2=Dsol/2; * diffusion coefficient of Tc pseudocolloid
* in bentonite
CONSTANT Dtc1=Dsol/2; * diffusion coefficient of Tc true colloid
* in overpack
CONSTANT Dtc2=Dsol/2; * diffusion coefficient of Tc true colloid
* in bentonite
CONSTANT thetag=0.33; * Porosity of glass
CONSTANT theta1=0.33; * Porosity of overpack
CONSTANT theta2=0.33; * Porosity of bentonite
CONSTANT ZTc=99; * atomic weight of Tc
CONSTANT invmass=1.06e3; * Inventory at calculation start in g
CONSTANT invmoles=invmass/ZTc; * Initial inventory in moles
CONSTANT rglass=0.22; * outer glass radius
CONSTANT rover1=0.52; * actual outer overpack radius
CONSTANT rbent1=1.50; * actual outer bentonite radius
CONSTANT h=1.35; * height of cylinder
CONSTANT vglass=pi*h*rglass*rglass*thetag; * volume of glass
*
* (Block 3) Mixing cell constants
*
CONSTANT q0=0.71*1e-3; * mean flow through mixing cell in m3/year
CONSTANT qfrac=1e4; * fractional variation in mixing cell flow, used to
* derive q - central case value is qfrac=0
CONSTANT q1=q0*qfrac;
CONSTANT qwyear=1e5; * half-width of gaussian peak in years
CONSTANT Tq=30000; * period of variation in q (in years)
CONSTANT qwidth=qwyear/Tq; * fractional width of gaussian peak
CONSTANT qamp=q0*qfrac; * amplitude of variation in q
CONSTANT wq=2*pi/Tq; * equivalent angular frequency
EQUATION qpar=(1/pi)*((pi/2)+arcsin(sin(wq*t-(pi/2))))); * varies linearly from 0 to 1 and
* back every Tq years
INITIAL qpar=0;
WEIGHT qpar=1/1000;
SIDE qadd=exp(-qpar*qpar/(qwidth*qwidth));
INITIAL qadd=1;
EQUATION q=q0+q1*qadd; * mixing cell flow rate (potentially time dependent)
INITIAL q=q0+q1*qadd;
WEIGHT q=1/(q*100);
*
* (Block 4) Define geometry.
*
* The problem is posed in 1D cylindrical coordinates. Although
* Spade does not understand about these at present, we can use
* 1D cartesian coordinates because we discretize the
* spatial derivatives ourselves.
*
* The spatial variables Rin1 and Rin2 are chosen to represent the INNER
* radius of each ring. We therefore set the grid spacing so that the
* OUTER radius of the outermost cylinder in each region matches the
* specified dimensions.
*
CONSTANT nover=5; * number of grid points in overpack
CONSTANT nbent=10; * number of grid points in bentonite
CONSTANT rover=rglass+(rover1-rglass)*(nover-1)/nover; * grid limit
* for overpack
CONSTANT rbent=rover1+(rbent1-rover1)*(nbent-1)/nbent; * grid limit
* for bentonite
CONSTANT dr1=(rover1-rglass)/nover; * grid spacing in overpack
CONSTANT dr2=(rbent1-rover1)/nbent; * grid spacing in bentonite
SPATIAL Rin1, UNIFORM, rglass, rover, nover; * INNER radius of each ring
SPATIAL Rin2, UNIFORM, rover1, rbent, nbent;
REGION over, CART1D, Rin1; * overpack grid
REGION bent, CART1D, Rin2; * bentonite grid
VECTOR Rout1[over]; * OUTER radius of each ring in overpack
VECTOR Rout2[bent]; * OUTER radius of each ring in bentonite
CONSTANT Rout1=Rin1+dr1;
CONSTANT Rout2=Rin2+dr2;
VECTOR V1[over]; * volume of WATER in each ring in overpack
VECTOR V2[bent]; * volume of WATER in each ring in bentonite
CONSTANT V1=theta1*pi*h*((Rout1*Rout1)-(Rin1*Rin1));
CONSTANT V2=theta2*pi*h*((Rout2*Rout2)-(Rin2*Rin2));
VECTOR SA1[over]; * OUTER surface area of each ring in overpack
VECTOR SA2[bent]; * OUTER surface area of each ring in bentonite
CONSTANT SA1=2*pi*h*Rout1;

```

```

CONSTANT SA2=2*pi*h*Rout2;
CONSTANT SAglass=2*pi*h*rglass; * outer surface area of glass
SIDE Cglass=ascale*inventory/vglass; * concentration in glass
*
* (Block 5) Declare remaining vector quantities
*
* the concentration vector within each region
VECTOR C1[over];
VECTOR C2[bent];
* the effective diffusivity
VECTOR alpha1[over];
VECTOR alpha2[bent];
* the flux across each outer boundary
VECTOR fluxout1[over];
VECTOR fluxout2[bent];
*
* (Block 6) Define alpha. The form of alpha varies depending on the assumed
* distribution of mobile species as a function of concentration.
* For the central case with no colloids or pseudocolloids, k1=k2=kpc1=kpc2=0.
*
CONSTANT k1=0; * fraction of true colloid in overpack
CONSTANT k2=0; * fraction of true colloid in bentonite
CONSTANT kpc1=0; * fraction of pseudocolloid in overpack
CONSTANT kpc2=0; * fraction of pseudocolloid in bentonite
* (6.1) alpha in overpack
SIDE alpha1=switch(C1-Cstar*R1,
    Dsol/R1+kpc1*(R1-1)*C1*Dpc1/R1,
    (Dsol*Cstar+kpc1*(R1-1)*Cstar*Dpc1+k1*Dtc1*(C1-Cstar*R1))/C1);
* (6.2) alpha in bentonite
SIDE alpha2=switch(C2-Cstar*R2,
    Dsol/R2+kpc2*(R2-1)*C2*Dpc2/R2,
    (Dsol*Cstar+kpc2*(R2-1)*Cstar*Dpc2+k2*Dtc2*(C2-Cstar*R2))/C2);
* (6.3) alpha in glass
SIDE alphaglass=switch(Cglass-Cstar*Rglass,
    Dsol/Rglass+kpc1*(Rglass-1)*Cglass*Dpc1/Rglass,
    (Dsol*Cstar+kpc1*(Rglass-1)*Cstar*Dpc1+k1*Dtc1*(Cglass-Cstar*Rglass))/Cglass);
*
* (Block 7) Define fluxout, which gives the flux of Tc across each outer
* face (positive = net outward flux).
* The outer boundary and the boundary between the overpack and bentonite
* have to be handled separately.
*
* (7.1) Flow through overpack
SIDE fluxout1[Rin1<rover]=SA1*(C1[Rin1=Rin1+dr1]*alpha1[Rin1=Rin1+dr1] -
    C1*alpha1)/dr1;
* (7.2) Flow through bentonite
SIDE fluxout2[Rin2<rbent]=SA2*(C2[Rin2=Rin2+dr2]*alpha2[Rin2=Rin2+dr2] -
    C2*alpha2)/dr2;
* (7.3) Flow from overpack to bentonite
SIDE fluxout1[Rin1=rover]=SA1*(C2[Rin2=rover1]*alpha2[Rin2=rover1] -
    C1*alpha1)/(0.5*(dr1+dr2));
* (7.4) Flow from outermost cell - mixing cell approximation
* (q may be a function of time)
SIDE fluxout2[Rin2=rbent]=switch(C2-Cstar*R2, C2/R2+kpc2*(R2-1)*C2/R2,
    Cstar+kpc2*(R2-1)*Cstar+k2*(C2-Cstar*R2))*q;
* (7.5) Flux from glass into innermost cell of overpack
SIDE fluxglass=-SAglass*(C1[Rin1=rglass]*alpha1[Rin1=rglass]
    -Cglass*alphaglass)/((dr1+rglass)/2);
*
* (Block 8) Evolution equations
*
* (8.1) Tc in glass decays and is transported out into overpack inner ring
EQUATION d(inventory, t)=-Kdecay*inventory-(fluxglass/ascale);
* (8.2) Innermost ring of overpack receives influx from glass
EQUATION d(C1, t)=- (Kdecay*C1[Rin1<=rglass]+(fluxout1[Rin1=rglass]-fluxglass)
    / V1);
* (8.3) Transport through overpack
EQUATION d(C1, t)=- (Kdecay*C1+(fluxout1-fluxout1[Rin1=Rin1-dr1, Rin1>rglass])
    / V1);
* (8.4) Transport from overpack to bentonite
EQUATION d(C2, t)=- (Kdecay*C2[Rin2>rover1]+
    (fluxout2-fluxout2[Rin2=Rin2-dr2])/V2);
* (8.5) Transport through bentonite (including loss from outermost ring)
EQUATION d(C2, t)=- (Kdecay*C2+
    (fluxout2[Rin2=rover1]-fluxout1[Rin1=rover])/V2);
* (8.6) Amount escaping is equal to loss from outermost ring
EQUATION d(escaped, t)=fluxout2[Rin2=rbent]/ascale;
* (8.7) Amount escaped and surviving is equal to amount escaped less decay

```

```

EQUATION d(outside, t)=fluxout2 [Rin2=rbent]/ascale-Kdecay*outside;
*
* (Block 9) Initial values
*
INITIAL inventory=invmoles;
INITIAL C1=0;
INITIAL C2=0;
INITIAL escaped=0;
INITIAL outside=0;
INITIAL q=q0+q1*sin(wq*t);
*
* (Block 10) Weights - chosen to give correct relative scaling for variables
*
WEIGHT C1=1/(C1+Cstar);
WEIGHT C2=1/(C2+Cstar);
WEIGHT inventory=1/(inventory+Cstar*vglass/ascale);
WEIGHT escaped=1/(escaped+C2 [Rin2=rbent]/ascale+Cstar*vglass/ascale);
WEIGHT outside=1/(outside+C2 [Rin2=rbent]/ascale+Cstar*vglass/ascale);
*
* (Block 11) Monitor time and magnitude of peak flux and time of
* "drop off"
*
* The current escaping flux is only considered to be a new peak value if it
* is "significantly" larger than the previous peak,
* i.e. if it exceeds it by more than cutoff times the solver error tolerance.
* The monitor variable drop_time records the time at which the escaping
* flux first drops below a fixed fraction (e.g. 95%) of the peak flux.
* The monitor variable peak_outside records the maximum amount of
* contaminant surviving outside at any time.
*
CONSTANT cutoff=3; * multiple of epsstep required for significance
MONITOR peak_time=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2 [Rin2=rbent]/ascale),
                        t, peak_time);
INITIAL peak_time=t;
MONITOR peak_flux=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2 [Rin2=rbent]/ascale),
                        fluxout2 [Rin2=rbent]/ascale, peak_flux);
INITIAL peak_flux=0;
CONSTANT drop_frac=0.95; * fraction of peak flux
MONITOR drop_time=switch(peak_flux*drop_frac-(fluxout2 [Rin2=rbent]/ascale),
                        t, drop_time);
INITIAL drop_time=t;
MONITOR peak_outside=max(outside, peak_outside);
INITIAL peak_outside=0;
*
* (End of central.spf)
*

```

```

*
* central.spf
*
* SPF file for Task 1.1 central case (IM3246)
*
* This file models the diffusive transport of Technetium Tc-99 through the
* near field of a glass/overpack corrosion product/bentonite system.
*
* JAW 02/02/94
*
* Version 1.7
*
* Modification history:
* Vn Who When What
* 1.0 JAW 02/02/94 First version
* 1.1 JAW 03/02/94 Improve variable scaling and layout of file
* 1.2 JAW 04/02/94 Add colloids and pseudo-colloids
* 1.3 JAW 08/02/94 Further minor layout changes and added "outside"
* 1.4 JAW 10/02/94 Add monitor variable drop_time
* 1.5 JAW 11/02/94 Add monitor variable peak_outside
* 1.6 JAW 14/02/94 Improve inner boundary condition
* 1.7 JAW 15/02/94 Further change to inner boundary condition
*                   + fixed error in (7.2)
*
* General notes:
* (a) Units.
* The time unit is years. Distances are in metres. Amount of contaminant
* remaining in the glass or escaping from the system is measured in moles
* of Tc-99.
* Concentrations of contaminant in the transported species are measured in
* moles per cubic metre *of water* multiplied by a scaling factor ascale.
* This factor serves to improve the numerical conditioning of the problem:
* without it typical transported concentrations would be many orders of
* magnitude smaller than the remaining inventory.
*
* (b) Geometry.
* The problem is posed in a 1D cylindrical coordinate system, assuming
* cylindrical symmetry. The height of the cylinder is taken as the height
* of the central glass waste canister.
* There are three distinct regions in the model:
* Region Inner Outer
* radius (m) radius (m)
* Glass 0 0.22
* Overpack 0.22 0.52
* Bentonite 0.52 1.50
* The glass is treated as a source term which forms the inner boundary
* condition.
* It is assumed that contaminant from the whole volume of the glass is
* instantaneously in equilibrium with the surrounding water.
* The overpack (strictly, overpack corrosion products) layer and bentonite
* layer are treated as two separate regions with a flux condition at
* their shared boundary.
* The outer boundary of the bentonite is a decompressed zone. This is
* represented by a mixing-cell treatment which forms the outer boundary
* condition.
*
* (Block 1) Control info.
*
LISTFILE; * generate list output
SPARSE; * use sparse matrix solver
ALGERR; * include errors on algebraic variables in timestep monitor
CONSTANT epsstep=1e-5; * value for absolute error tolerance per timestep
EPSSTEP epsstep; * set absolute error tolerance
INITSTEP 1.e-3; * initial timestep
MINTIME 10; * Set output timesteps (in years)
MAXTIME 1e7; * - write out every 10000 years
GEOMSTEP 1.258925412; * from t=0 to t=1e7 years
*ARITHSTEP 1e5; * from t=0 to t=1e7 years
*
* (Block 2) Fundamental constants
*
CONSTANT pi=4*arctan(1);
CONSTANT SecsPerYear=365.2422*24*60*60; * number of seconds in one year
CONSTANT ascale=1e10; * multiplicative factor for all molar concentrations
* in transported species - improves conditioning
CONSTANT thalf=2.1e5; * Half-life of Tc-99 in years
CONSTANT Kdecay=log(2)/thalf; * Equivalent decay constant
CONSTANT Cstar=1e-9*ascale; * Solubility limit of Tc-99 in mol/m3
CONSTANT R1=1; * Retardation factor of Tc-99 in inner (overpack) layer

```



```

CONSTANT R2=2.8; * Retardation factor of Tc-99 in outer (bentonite) layer
CONSTANT Rglass=1; * Retardation factor of Tc-99 in glass
CONSTANT Dsol=3e-10*SecsPerYear; * diffusion coefficient of Tc solute in
    * both bentonite and overpack
CONSTANT Dpc1=Dsol/10; * diffusion coefficient of Tc pseudocolloid
    * in overpack
CONSTANT Dpc2=Dsol/10; * diffusion coefficient of Tc pseudocolloid
    * in bentonite
CONSTANT Dtc1=Dsol/2; * diffusion coefficient of Tc true colloid
    * in overpack
CONSTANT Dtc2=Dsol/2; * diffusion coefficient of Tc true colloid
    * in bentonite
CONSTANT thetag=0.33; * Porosity of glass
CONSTANT theta1=0.33; * Porosity of overpack
CONSTANT theta2=0.33; * Porosity of bentonite
CONSTANT ZTc=99; * atomic weight of Tc
CONSTANT invmass=1.06e3; * Inventory at calculation start in g
CONSTANT invmoles=invmass/ZTc; * Initial inventory in moles
CONSTANT rglass=0.22; * outer glass radius
CONSTANT rover1=0.52; * actual outer overpack radius
CONSTANT rbent1=1.50; * actual outer bentonite radius
CONSTANT h=1.35; * height of cylinder
CONSTANT vglass=pi*h*rglass*rglass*thetag; * volume of glass
*
* (Block 3) Mixing cell constants
*
CONSTANT q0=0.71*1e-3; * mean flow through mixing cell in m3/year
CONSTANT qfrac=0; * fractional variation in mixing cell flow, used to
    * derive q - central case value is qfrac=0
CONSTANT q1=q0*qfrac; * amplitude of variation in q
CONSTANT Tq=2000; * period of variation in q (in years)
CONSTANT wq=2*pi/Tq; * equivalent angular frequency
SIDE q=q0+q1*sin(wq*t); * mixing cell flow rate (potentially time dependent)
*
* (Block 4) Define geometry.
*
* The problem is posed in 1D cylindrical coordinates. Although
* Spade does not understand about these at present, we can use
* 1D cartesian coordinates because we discretize the
* spatial derivatives ourselves.
*
* The spatial variables Rin1 and Rin2 are chosen to represent the INNER
* radius of each ring. We therefore set the grid spacing so that the
* OUTER radius of the outermost cylinder in each region matches the
* specified dimensions.
*
CONSTANT nover=5; * number of grid points in overpack
CONSTANT nbent=10; * number of grid points in bentonite
CONSTANT rover=rglass+(rover1-rglass)*(nover-1)/nover; * grid limit
    * for overpack
CONSTANT rbent=rover1+(rbent1-rover1)*(nbent-1)/nbent; * grid limit
    * for bentonite

CONSTANT dr1=(rover1-rglass)/nover; * grid spacing in overpack
CONSTANT dr2=(rbent1-rover1)/nbent; * grid spacing in bentonite
SPATIAL Rin1, UNIFORM, rglass, rover, nover; * INNER radius of each ring
SPATIAL Rin2, UNIFORM, rover1, rbent, nbent;
REGION over, CART1D, Rin1; * overpack grid
REGION bent, CART1D, Rin2; * bentonite grid
VECTOR Rout1[over]; * OUTER radius of each ring in overpack
VECTOR Rout2[bent]; * OUTER radius of each ring in bentonite
CONSTANT Rout1=Rin1+dr1;
CONSTANT Rout2=Rin2+dr2;
VECTOR V1[over]; * volume of WATER in each ring in overpack
VECTOR V2[bent]; * volume of WATER in each ring in bentonite
CONSTANT V1=theta1*pi*h*((Rout1*Rout1) - (Rin1*Rin1));
CONSTANT V2=theta2*pi*h*((Rout2*Rout2) - (Rin2*Rin2));
VECTOR SA1[over]; * OUTER surface area of each ring in overpack
VECTOR SA2[bent]; * OUTER surface area of each ring in bentonite
CONSTANT SA1=2*pi*h*Rout1;
CONSTANT SA2=2*pi*h*Rout2;
CONSTANT SAglass=2*pi*h*rglass; * outer surface area of glass
SIDE Cglass=ascale*inventory/vglass; * concentration in glass
*
* (Block 5) Declare remaining vector quantities
*
* the concentration vector within each region
VECTOR C1[over];
VECTOR C2[bent];
* the effective diffusivity

```

```

VECTOR alpha1[over];
VECTOR alpha2[bent];
* the flux across each outer boundary
VECTOR fluxout1[over];
VECTOR fluxout2[bent];
*
* (Block 6) Define alpha. The form of alpha varies depending on the assumed
* distribution of mobile species as a function of concentration.
* For the central case with no colloids or pseudocolloids, k1=k2=kpc1=kpc2=0.
*
CONSTANT k1=0.1; * fraction of true colloid in overpack
CONSTANT k2=0.1; * fraction of true colloid in bentonite
CONSTANT kpc1=0; * fraction of pseudocolloid in overpack
CONSTANT kpc2=0.1; * fraction of pseudocolloid in bentonite
* (6.1) alpha in overpack
SIDE alpha1=switch(C1-Cstar*R1,
    Dsol/R1+kpc1*(R1-1)*Dpc1/R1,
    (Dsol*Cstar+kpc1*(R1-1)*Cstar*Dpc1+k1*Dtc1*(C1-Cstar*R1))/C1);
* (6.2) alpha in bentonite
SIDE alpha2=switch(C2-Cstar*R2,
    Dsol/R2+kpc2*(R2-1)*Dpc2/R2,
    (Dsol*Cstar+kpc2*(R2-1)*Cstar*Dpc2+k2*Dtc2*(C2-Cstar*R2))/C2);
* (6.3) alpha in glass
SIDE alphaglass=switch(Cglass-Cstar*Rglass,
    Dsol/Rglass+kpc1*(Rglass-1)*Dpc1/Rglass,
    (Dsol*Cstar+kpc1*(Rglass-1)*Cstar*Dpc1+k1*Dtc1*(Cglass-Cstar*Rglass))/Cglass);
*
* (Block 7) Define fluxout, which gives the flux of Tc across each outer
* face (positive = net outward flux).
* The outer boundary and the boundary between the overpack and bentonite
* have to be handled separately.
*
* (7.1) Flow through overpack
SIDE fluxout1[Rin1<rover]=-SA1*(C1[Rin1=Rin1+dr1]*alpha1[Rin1=Rin1+dr1] -
    C1*alpha1)/dr1;
* (7.2) Flow through bentonite
SIDE fluxout2[Rin2<rbent]=-SA2*(C2[Rin2=Rin2+dr2]*alpha2[Rin2=Rin2+dr2] -
    C2*alpha2)/dr2;
* (7.3) Flow from overpack to bentonite
SIDE fluxout1[Rin1=rover]=-SA1*(C2[Rin2=rover1]*alpha2[Rin2=rover1] -
    C1*alpha1)/(0.5*(dr1+dr2));
* (7.4) Flow from outermost cell - mixing cell approximation
* (q may be a function of time)
SIDE fluxout2[Rin2=rbent]=switch(C2-Cstar*R2, C2/R2+kpc2*(R2-1)*C2/R2,
    Cstar+kpc2*(R2-1)*Cstar+k2*(C2-Cstar*R2))*q;
* (7.5) Flux from glass into innermost cell of overpack
SIDE fluxglass=-SAglass*(C1[Rin1=rglass]*alpha1[Rin1=rglass]
    -Cglass*alphaglass)/((dr1+rglass)/2);
*
* (Block 8) Evolution equations
*
* (8.1) Tc in glass decays and is transported out into overpack inner ring
EQUATION d(inventory, t)=-Kdecay*inventory-(fluxglass/ascale);
* (8.2) Innermost ring of overpack receives influx from glass
EQUATION d(C1, t)=-Kdecay*C1[Rin1<=rglass]+(fluxout1[Rin1=rglass]-fluxglass)
    / V1);
* (8.3) Transport through overpack
EQUATION d(C1, t)=-Kdecay*C1+(fluxout1-fluxout1[Rin1=Rin1-dr1, Rin1>rglass])
    / V1);
* (8.4) Transport from overpack to bentonite
EQUATION d(C2, t)=-Kdecay*C2[Rin2>rover1]+
    (fluxout2-fluxout2[Rin2=Rin2-dr2])/V2);
* (8.5) Transport through bentonite (including loss from outermost ring)
EQUATION d(C2, t)=-Kdecay*C2+
    (fluxout2[Rin2=rover1]-fluxout1[Rin1=rover])/V2);
* (8.6) Amount escaping is equal to loss from outermost ring
EQUATION d(escaped, t)=fluxout2[Rin2=rbent]/ascale;
* (8.7) Amount escaped and surviving is equal to amount escaped less decay
EQUATION d(outside, t)=fluxout2[Rin2=rbent]/ascale-Kdecay*outside;
*
* (Block 9) Initial values
*
INITIAL inventory=invmoles;
INITIAL C1=0;
INITIAL C2=0;
INITIAL escaped=0;
INITIAL outside=0;
INITIAL q=q0+q1*sin(wq*t);

```

```

*
* (Block 10) Weights - chosen to give correct relative scaling for variables
*
WEIGHT C1=1/(C1+Cstar);
WEIGHT C2=1/(C2+Cstar);
WEIGHT inventory=1/(inventory+Cstar*vglass/ascale);
WEIGHT escaped=1/(escaped+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
WEIGHT outside=1/(outside+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
*
* (Block 11) Monitor time and magnitude of peak flux and time of
* "drop off"
*
* The current escaping flux is only considered to be a new peak value if it
* is "significantly" larger than the previous peak,
* i.e. if it exceeds it by more than cutoff times the solver error tolerance.
* The monitor variable drop_time records the time at which the escaping
* flux first drops below a fixed fraction (e.g. 95%) of the peak flux.
* The monitor variable peak_outside records the maximum amount of
* contaminant surviving outside at any time.
*
CONSTANT cutoff=3; * multiple of epsstep required for significance
MONITOR peak_time=switch(peak_flux*(1+cutoff*epsstep)-
    (fluxout2[Rin2=rbent]/ascale),
    t, peak_time);
INITIAL peak_time=t;
MONITOR peak_flux=switch(peak_flux*(1+cutoff*epsstep)-
    (fluxout2[Rin2=rbent]/ascale),
    fluxout2[Rin2=rbent]/ascale, peak_flux);
INITIAL peak_flux=0;
CONSTANT drop_frac=0.95; * fraction of peak flux
MONITOR drop_time=switch(peak_flux*drop_frac-(fluxout2[Rin2=rbent]/ascale),
    t, drop_time);
INITIAL drop_time=t;
MONITOR peak_outside=max(outside, peak_outside);
INITIAL peak_outside=0;
*
* (End of central.spf)
*

```

```

*
* central.spf
*
* SPF file for Task 1.1 central case (IM3246)
*
* This file models the diffusive transport of Technetium Tc-99 through the
* near field of a glass/overpack corrosion product/bentonite system.
*
* JAW 02/02/94
*
* Version 1.7
*
* Modification history:
* Vn Who When What
* 1.0 JAW 02/02/94 First version
* 1.1 JAW 03/02/94 Improve variable scaling and layout of file
* 1.2 JAW 04/02/94 Add colloids and pseudo-colloids
* 1.3 JAW 08/02/94 Further minor layout changes and added "outside"
* 1.4 JAW 10/02/94 Add monitor variable drop_time
* 1.5 JAW 11/02/94 Add monitor variable peak_outside
* 1.6 JAW 14/02/94 Improve inner boundary condition
* 1.7 JAW 15/02/94 Further change to inner boundary condition
*                   + fixed error in (7.2)
*
* General notes:
* (a) Units.
* The time unit is years. Distances are in metres. Amount of contaminant
* remaining in the glass or escaping from the system is measured in moles
* of Tc-99.
* Concentrations of contaminant in the transported species are measured in
* moles per cubic metre *of water* multiplied by a scaling factor ascale.
* This factor serves to improve the numerical conditioning of the problem:
* without it typical transported concentrations would be many orders of
* magnitude smaller than the remaining inventory.
*
* (b) Geometry.
* The problem is posed in a 1D cylindrical coordinate system, assuming
* cylindrical symmetry. The height of the cylinder is taken as the height
* of the central glass waste canister.
* There are three distinct regions in the model:
* Region Inner Outer
* radius (m) radius (m)
* Glass 0 0.22
* Overpack 0.22 0.52
* Bentonite 0.52 1.50
* The glass is treated as a source term which forms the inner boundary
* condition.
* It is assumed that contaminant from the whole volume of the glass is
* instantaneously in equilibrium with the surrounding water.
* The overpack (strictly, overpack corrosion products) layer and bentonite
* layer are treated as two separate regions with a flux condition at
* their shared boundary.
* The outer boundary of the bentonite is a decompressed zone. This is
* represented by a mixing-cell treatment which forms the outer boundary
* condition.
*
* (Block 1) Control info.
*
LISTFILE; * generate list output
SPARSE; * use sparse matrix solver
ALGERR; * include errors on algebraic variables in timestep monitor
CONSTANT epsstep=1e-5; * value for absolute error tolerance per timestep
EPSSTEP epsstep; * set absolute error tolerance
INITSTEP 1.e-3; * initial timestep
MINTIME 10; * Set output timesteps (in years)
MAXTIME 1e7; * - write out every 10000 years
GEOMSTEP 1.258925412; * from t=0 to t=1e7 years
*ARITHSTEP 1e5; * from t=0 to t=1e7 years
*
* (Block 2) Fundamental constants
*
CONSTANT pi=4*arctan(1);
CONSTANT SecsPerYear=365.2422*24*60*60; * number of seconds in one year
CONSTANT ascale=1e10; * multiplicative factor for all molar concentrations
* in transported species - improves conditioning
CONSTANT thalf=2.1e5; * Half-life of Tc-99 in years
CONSTANT Kdecay=log(2)/thalf; * Equivalent decay constant
CONSTANT Cstar=1e-9*ascale; * Solubility limit of Tc-99 in mol/m3
CONSTANT R1=1; * Retardation factor of Tc-99 in inner (overpack) layer

```

```

CONSTANT R2=2.8; * Retardation factor of Tc-99 in outer (bentonite) layer
CONSTANT Rglass=1; * Retardation factor of Tc-99 in glass
CONSTANT Dsol=3e-10*SecsPerYear; * diffusion coefficient of Tc solute in
    * both bentonite and overpack
CONSTANT Dpc1=Dsol/10; * diffusion coefficient of Tc pseudocolloid
    * in overpack
CONSTANT Dpc2=Dsol/10; * diffusion coefficient of Tc pseudocolloid
    * in bentonite
CONSTANT Dtc1=Dsol/2; * diffusion coefficient of Tc true colloid
    * in overpack
CONSTANT Dtc2=Dsol/2; * diffusion coefficient of Tc true colloid
    * in bentonite
CONSTANT thetag=0.33; * Porosity of glass
CONSTANT theta1=0.33; * Porosity of overpack
CONSTANT theta2=0.33; * Porosity of bentonite
CONSTANT ZTc=99; * atomic weight of Tc
CONSTANT invmass=1.06e3; * Inventory at calculation start in g
CONSTANT invmoles=invmass/ZTc; * Initial inventory in moles
CONSTANT rglass=0.22; * outer glass radius
CONSTANT rover1=0.52; * actual outer overpack radius
CONSTANT rbent1=1.50; * actual outer bentonite radius
CONSTANT h=1.35; * height of cylinder
CONSTANT vglass=pi*h*rglass*rglass*thetag; * volume of glass
*
* (Block 3) Mixing cell constants
*
CONSTANT q0=0.71*1e-3; * mean flow through mixing cell in m3/year
CONSTANT qfrac=0; * fractional variation in mixing cell flow, used to
    * derive q - central case value is qfrac=0
CONSTANT q1=q0*qfrac; * amplitude of variation in q
CONSTANT Tq=2000; * period of variation in q (in years)
CONSTANT wq=2*pi/Tq; * equivalent angular frequency
SIDE q=q0+q1*sin(wq*t); * mixing cell flow rate (potentially time dependent)
*
* (Block 4) Define geometry.
*
* The problem is posed in 1D cylindrical coordinates. Although
* Spade does not understand about these at present, we can use
* 1D cartesian coordinates because we discretize the
* spatial derivatives ourselves.
*
* The spatial variables Rin1 and Rin2 are chosen to represent the INNER
* radius of each ring. We therefore set the grid spacing so that the
* OUTER radius of the outermost cylinder in each region matches the
* specified dimensions.
*
CONSTANT nover=5; * number of grid points in overpack
CONSTANT nbent=10; * number of grid points in bentonite
CONSTANT rover=rglass+(rover1-rglass)*(nover-1)/nover; * grid limit
    * for overpack
CONSTANT rbent=rover1+(rbent1-rover1)*(nbent-1)/nbent; * grid limit
    * for bentonite

CONSTANT dr1=(rover1-rglass)/nover; * grid spacing in overpack
CONSTANT dr2=(rbent1-rover1)/nbent; * grid spacing in bentonite
SPATIAL Rin1, UNIFORM, rglass, rover, nover; * INNER radius of each ring
SPATIAL Rin2, UNIFORM, rover1, rbent, nbent;
REGION over, CART1D, Rin1; * overpack grid
REGION bent, CART1D, Rin2; * bentonite grid
VECTOR Rout1[over]; * OUTER radius of each ring in overpack
VECTOR Rout2[bent]; * OUTER radius of each ring in bentonite
CONSTANT Rout1=Rin1+dr1;
CONSTANT Rout2=Rin2+dr2;
VECTOR V1[over]; * volume of WATER in each ring in overpack
VECTOR V2[bent]; * volume of WATER in each ring in bentonite
CONSTANT V1=theta1*pi*h*((Rout1*Rout1) - (Rin1*Rin1));
CONSTANT V2=theta2*pi*h*((Rout2*Rout2) - (Rin2*Rin2));
VECTOR SA1[over]; * OUTER surface area of each ring in overpack
VECTOR SA2[bent]; * OUTER surface area of each ring in bentonite
CONSTANT SA1=2*pi*h*Rout1;
CONSTANT SA2=2*pi*h*Rout2;
CONSTANT SAglass=2*pi*h*rglass; * outer surface area of glass
SIDE Cglass=ascale*inventory/vglass; * concentration in glass
*
* (Block 5) Declare remaining vector quantities
*
* the concentration vector within each region
VECTOR C1[over];
VECTOR C2[bent];
* the effective diffusivity

```

```

VECTOR alpha1[over];
VECTOR alpha2[bent];
* the flux across each outer boundary
VECTOR fluxout1[over];
VECTOR fluxout2[bent];
*
* (Block 6) Define alpha. The form of alpha varies depending on the assumed
* distribution of mobile species as a function of concentration.
* For the central case with no colloids or pseudocolloids, k1=k2=kpc1=kpc2=0.
*
CONSTANT k1=0; * fraction of true colloid in overpack
CONSTANT k2=0; * fraction of true colloid in bentonite
CONSTANT kpc1=0; * fraction of pseudocolloid in overpack
CONSTANT kpc2=0.1; * fraction of pseudocolloid in bentonite
* (6.1) alpha in overpack
SIDE alpha1=switch(C1-Cstar*R1,
    Dsol/R1+kpc1*(R1-1)*Dpc1/R1,
    (Dsol*Cstar+kpc1*(R1-1)*Cstar*Dpc1+k1*Dtc1*(C1-Cstar*R1))/C1);
* (6.2) alpha in bentonite
SIDE alpha2=switch(C2-Cstar*R2,
    Dsol/R2+kpc2*(R2-1)*Dpc2/R2,
    (Dsol*Cstar+kpc2*(R2-1)*Cstar*Dpc2+k2*Dtc2*(C2-Cstar*R2))/C2);

* (6.3) alpha in glass
SIDE alphaglass=switch(Cglass-Cstar*Rglass,
    Dsol/Rglass+kpc1*(Rglass-1)*Dpc1/Rglass,
    (Dsol*Cstar+kpc1*(Rglass-1)*Cstar*Dpc1+k1*Dtc1*(Cglass-Cstar*Rglass))/Cglass);
*
* (Block 7) Define fluxout, which gives the flux of Tc across each outer
* face (positive = net outward flux).
* The outer boundary and the boundary between the overpack and bentonite
* have to be handled separately.
*
* (7.1) Flow through overpack
SIDE fluxout1[Rin1<rover]=--SA1*(C1[Rin1=Rin1+dr1]*alpha1[Rin1=Rin1+dr1] -
    C1*alpha1)/dr1;
* (7.2) Flow through bentonite
SIDE fluxout2[Rin2<rbent]=--SA2*(C2[Rin2=Rin2+dr2]*alpha2[Rin2=Rin2+dr2] -
    C2*alpha2)/dr2;
* (7.3) Flow from overpack to bentonite
SIDE fluxout1[Rin1=rover]=--SA1*(C2[Rin2=rover1]*alpha2[Rin2=rover1] -
    C1*alpha1)/(0.5*(dr1+dr2));
* (7.4) Flow from outermost cell - mixing cell approximation
* (q may be a function of time)
SIDE fluxout2[Rin2=rbent]=switch(C2-Cstar*R2, C2/R2+kpc2*(R2-1)*C2/R2,
    Cstar+kpc2*(R2-1)*Cstar+k2*(C2-Cstar*R2))*q;
* (7.5) Flux from glass into innermost cell of overpack
SIDE fluxglass=-SAglass*(C1[Rin1=rglass]*alpha1[Rin1=rglass]
    -Cglass*alphaglass)/((dr1+rglass)/2);
*
* (Block 8) Evolution equations
*
* (8.1) Tc in glass decays and is transported out into overpack inner ring
EQUATION d(inventory, t)=-Kdecay*inventory-(fluxglass/ascale);
* (8.2) Innermost ring of overpack receives influx from glass
EQUATION d(C1, t)=-((Kdecay*C1[Rin1<=rglass]+(fluxout1[Rin1=rglass]-fluxglass)
    / V1);
* (8.3) Transport through overpack
EQUATION d(C1, t)=-((Kdecay*C1+(fluxout1-fluxout1[Rin1=Rin1-dr1, Rin1>rglass])
    / V1);
* (8.4) Transport from overpack to bentonite
EQUATION d(C2, t)=-((Kdecay*C2[Rin2>rover1]+
    (fluxout2-fluxout2[Rin2=Rin2-dr2])/V2);
* (8.5) Transport through bentonite (including loss from outermost ring)
EQUATION d(C2, t)=-((Kdecay*C2+
    (fluxout2[Rin2=rover1]-fluxout1[Rin1=rover])/V2);
* (8.6) Amount escaping is equal to loss from outermost ring
EQUATION d(escaped, t)=fluxout2[Rin2=rbent]/ascale;
* (8.7) Amount escaped and surviving is equal to amount escaped less decay
EQUATION d(outside, t)=fluxout2[Rin2=rbent]/ascale-Kdecay*outside;
*
* (Block 9) Initial values
*
INITIAL inventory=invmoles;
INITIAL C1=0;
INITIAL C2=0;
INITIAL escaped=0;
INITIAL outside=0;
INITIAL q=q0+q1*sin(wq*t);

```

```

*
* (Block 10) Weights - chosen to give correct relative scaling for variables
*
WEIGHT C1=1/(C1+Cstar);
WEIGHT C2=1/(C2+Cstar);
WEIGHT inventory=1/(inventory+Cstar*vglass/ascale);
WEIGHT escaped=1/(escaped+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
WEIGHT outside=1/(outside+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
*
* (Block 11) Monitor time and magnitude of peak flux and time of
* "drop off"
*
* The current escaping flux is only considered to be a new peak value if it
* is "significantly" larger than the previous peak,
* i.e. if it exceeds it by more than cutoff times the solver error tolerance.
* The monitor variable drop_time records the time at which the escaping
* flux first drops below a fixed fraction (e.g. 95%) of the peak flux.
* The monitor variable peak_outside records the maximum amount of
* contaminant surviving outside at any time.
*
CONSTANT cutoff=3; * multiple of epsstep required for significance
MONITOR peak_time=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        t, peak_time);

INITIAL peak_time=t;
MONITOR peak_flux=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        fluxout2[Rin2=rbent]/ascale, peak_flux);

INITIAL peak_flux=0;
CONSTANT drop_frac=0.95; * fraction of peak flux
MONITOR drop_time=switch(peak_flux*drop_frac-(fluxout2[Rin2=rbent]/ascale),
                        t, drop_time);

INITIAL drop_time=t;
MONITOR peak_outside=max(outside, peak_outside);
INITIAL peak_outside=0;
*
* (End of central.spf)
*

```

```

*
* Case with radiolysis and true colloids
* central.spf
*
* SPF file for Task 1.1 central case (IM3246)
*
* This file models the diffusive transport of Technetium Tc-99 through the
* near field of a glass/overpack corrosion product/bentonite system.
*
* JAW 02/02/94
*
* Version 1.7
*
* Modification history:
* Vn Who When What
* 1.0 JAW 02/02/94 First version
* 1.1 JAW 03/02/94 Improve variable scaling and layout of file
* 1.2 JAW 04/02/94 Add colloids and pseudo-colloids
* 1.3 JAW 08/02/94 Further minor layout changes and added "outside"
* 1.4 JAW 10/02/94 Add monitor variable drop_time
* 1.5 JAW 11/02/94 Add monitor variable peak_outside
* 1.6 JAW 14/02/94 Improve inner boundary condition
* 1.7 JAW 15/02/94 Further change to inner boundary condition
* + fixed error in (7.2)
*
* General notes:
* (a) Units.
* The time unit is years. Distances are in metres. Amount of contaminant
* remaining in the glass or escaping from the system is measured in moles
* of Tc-99.
* Concentrations of contaminant in the transported species are measured in
* moles per cubic metre *of water* multiplied by a scaling factor ascale.
* This factor serves to improve the numerical conditioning of the problem:
* without it typical transported concentrations would be many orders of
* magnitude smaller than the remaining inventory.
*
* (b) Geometry.
* The problem is posed in a 1D cylindrical coordinate system, assuming
* cylindrical symmetry. The height of the cylinder is taken as the height
* of the central glass waste canister.
* There are three distinct regions in the model:
* Region Inner Outer
* radius (m) radius (m)
* Glass 0 0.22
* Overpack 0.22 0.52
* Bentonite 0.52 1.50
* The glass is treated as a source term which forms the inner boundary
* condition.
* It is assumed that contaminant from the whole volume of the glass is
* instantaneously in equilibrium with the surrounding water.
* The overpack (strictly, overpack corrosion products) layer and bentonite
* layer are treated as two separate regions with a flux condition at
* their shared boundary.
* The outer boundary of the bentonite is a decompressed zone. This is
* represented by a mixing-cell treatment which forms the outer boundary
* condition.
*
* (Block 1) Control info.
*
LISTFILE: * generate list output
SPARSE: * use sparse matrix solver
ALGERR: * include errors on algebraic variables in timestep monitor
CONSTANT epsstep=1e-5; * value for absolute error tolerance per timestep
EPSSTEP epsstep; * set absolute error tolerance
INITSTEP 1.e-3; * initial timestep
MINTIME 10; * Set output timesteps (in years)
MAXTIME 1e7; * - write out every 10000 years
GEOMSTEP 1.258925412; * from t=0 to t=1e7 years
*ARITHSTEP 1e5; * from t=0 to t=1e7 years
*MAXTIME 1e4; * - write out every 10000 years
*ARITHSTEP 1e1; * from t=0 to t=1e7 years
*
* (Block 2) Fundamental constants
*
CONSTANT pi=4*arctan(1);
CONSTANT SecsPerYear=365.2422*24*60*60; * number of seconds in one year
CONSTANT ascale=1e10; * multiplicative factor for all molar concentrations
* in transported species - improves conditioning
CONSTANT thalf=2.1e5; * Half-life of Tc-99 in years

```



```

CONSTANT Kdecay=log(2)/thalf; * Equivalent decay constant
CONSTANT R1=1; * Retardation factor of Tc-99 in inner (overpack) layer
CONSTANT R2=2.8; * Retardation factor of Tc-99 in outer (bentonite) layer
CONSTANT Rglass=1; * Retardation factor of Tc-99 in glass
CONSTANT Dsol=3e-10*SecsPerYear; * diffusion coefficient of Tc solute in
    * both bentonite and overpack
CONSTANT Dpc1=Dsol/2; * diffusion coefficient of Tc pseudocolloid
    * in overpack
CONSTANT Dpc2=Dsol/2; * diffusion coefficient of Tc pseudocolloid
    * in bentonite
CONSTANT Dtc1=Dsol/2; * diffusion coefficient of Tc true colloid
    * in overpack
CONSTANT Dtc2=Dsol/2; * diffusion coefficient of Tc true colloid
    * in bentonite
CONSTANT thetag=0.33; * Porosity of glass
CONSTANT theta1=0.33; * Porosity of overpack
CONSTANT theta2=0.33; * Porosity of bentonite
CONSTANT ZTc=99; * atomic weight of Tc
CONSTANT invmass=1.06e3; * Inventory at calculation start in g
CONSTANT invmoles=invmass/ZTc; * Initial inventory in moles
CONSTANT rglass=0.22; * outer glass radius
CONSTANT rover1=0.52; * actual outer overpack radius
CONSTANT rbent1=1.50; * actual outer bentonite radius
CONSTANT h=1.35; * height of cylinder
CONSTANT vglass=pi*h*rglass*rglass*thetag; * volume of glass
*
* (Block 3) Mixing cell constants
*
CONSTANT q=0.71*1e-3; * mean flow through mixing cell in m3/year
CONSTANT Cstar0=1e-9*ascale; * Solubility limit of Tc-99 in mol/m3 in bentonite
*
* (Block 4) Define geometry.
*
* The problem is posed in 1D cylindrical coordinates. Although
* Spade does not understand about these at present, we can use
* 1D cartesian coordinates because we discretize the
* spatial derivatives ourselves.
*
* The spatial variables Rin1 and Rin2 are chosen to represent the INNER
* radius of each ring. We therefore set the grid spacing so that the
* OUTER radius of the outermost cylinder in each region matches the
* specified dimensions.
*
CONSTANT nover=5; * number of grid points in overpack
CONSTANT nbent=10; * number of grid points in bentonite
CONSTANT rover=rglass+(rover1-rglass)*(nover-1)/nover; * grid limit
    * for overpack
CONSTANT rbent=rover1+(rbent1-rover1)*(nbent-1)/nbent; * grid limit
    * for bentonite
CONSTANT dr1=(rover1-rglass)/nover; * grid spacing in overpack
CONSTANT dr2=(rbent1-rover1)/nbent; * grid spacing in bentonite
SPATIAL Rin1,UNIFORM,rglass,rover,nover; * INNER radius of each ring
SPATIAL Rin2,UNIFORM,rover1,rbent,nbent;
REGION over,CART1D,Rin1; * overpack grid
REGION bent,CART1D,Rin2; * bentonite grid
VECTOR Rout1[over]; * OUTER radius of each ring in overpack
VECTOR Rout2[bent]; * OUTER radius of each ring in bentonite
CONSTANT Rout1=Rin1+dr1;
CONSTANT Rout2=Rin2+dr2;
VECTOR V1[over]; * volume of WATER in each ring in overpack
VECTOR V2[bent]; * volume of WATER in each ring in bentonite
CONSTANT V1=theta1*pi*h*((Rout1*Rout1)-(Rin1*Rin1));
CONSTANT V2=theta2*pi*h*((Rout2*Rout2)-(Rin2*Rin2));
VECTOR SA1[over]; * OUTER surface area of each ring in overpack
VECTOR SA2[bent]; * OUTER surface area of each ring in bentonite
CONSTANT SA1=2*pi*h*Rout1;
CONSTANT SA2=2*pi*h*Rout2;
CONSTANT SAglass=2*pi*h*rglass; * outer surface area of glass
SIDE Cglass=ascale*inventory/vglass; * concentration in glass
*
* Variable Cstar1 for radiolysis model
*
VECTOR Cstar1[over];
CONSTANT CsT1=200; * rise time for radiolysis
CONSTANT CsT2=1000; * fall time for radiolysis
*CONSTANT Cstar1=Cstar0*(1+1e6*(1-((Rin1-rglass)/(rover-rglass)))));
*SIDE Cstar1=Cstar0*(1+1e6*(1-((Rin1-rglass)/(rover-rglass))))
* *(heavy(t)-heavy(CsT2-t));

```

```

EQUATION Cstar1=Cstar0*(1+1e6*(1-((Rin1-rglass)/(rover-rglass)))
      *(1-exp(-t/CsT1)-(1-exp(-t/CsT2))));
INITIAL Cstar1=Cstar0;
WEIGHT Cstar1=1/Cstar1;

*
* (Block 5) Declare remaining vector quantities
*
* the concentration vector within each region
VECTOR C1[over];
VECTOR C2[bent];
* the effective diffusivity
VECTOR alpha1[over];
VECTOR alpha2[bent];
* the flux across each outer boundary
VECTOR fluxout1[over];
VECTOR fluxout2[bent];
*
* (Block 6) Define alpha. The form of alpha varies depending on the assumed
* distribution of mobile species as a function of concentration.
* For the central case with no colloids or pseudocolloids, k1=k2=kpc1=kpc2=0.
*
CONSTANT k1=0.1; * fraction of true colloid in overpack
CONSTANT k2=0.1; * fraction of true colloid in bentonite
CONSTANT kpc1=0; * fraction of pseudocolloid in overpack
CONSTANT kpc2=0; * fraction of pseudocolloid in bentonite
* (6.1) alpha in overpack
SIDE alpha1=switch(C1-Cstar1*R1,
      Dsol/R1+kpc1*(R1-1)*C1*Dpc1/R1,
      (Dsol*Cstar1+kpc1*(R1-1)*Cstar1*Dpc1+k1*Dtc1*(C1-Cstar1*R1))/C1);
* (6.2) alpha in bentonite
SIDE alpha2=switch(C2-Cstar0*R2,
      Dsol/R2+kpc2*(R2-1)*C2*Dpc2/R2,
      (Dsol*Cstar0+kpc2*(R2-1)*Cstar0*Dpc2+k2*Dtc2*(C2-Cstar0*R2))/C2);
* (6.3) alpha in glass
SIDE alphaglass=switch(Cglass-Cstar1[Rin1=rglass]*Rglass,
      Dsol/Rglass+kpc1*(Rglass-1)*Cglass*Dpc1/Rglass,
      (Dsol*Cstar1[Rin1=rglass]+kpc1*(Rglass-1)*Cstar1[Rin1=rglass]*Dpc1+k1*Dtc1*(Cglass-Cstar1[Rin1=rgl
      ass]*Rglass))/Cglass);
*
* (Block 7) Define fluxout, which gives the flux of Tc across each outer
* face (positive = net outward flux).
* The outer boundary and the boundary between the overpack and bentonite
* have to be handled separately.
*
* (7.1) Flow through overpack
SIDE fluxout1[Rin1<rover]==SA1*(C1[Rin1=Rin1+dr1]*alpha1[Rin1=Rin1+dr1] -
      C1*alpha1)/dr1;
* (7.2) Flow through bentonite
SIDE fluxout2[Rin2<rbent]==SA2*(C2[Rin2=Rin2+dr2]*alpha2[Rin2=Rin2+dr2] -
      C2*alpha2)/dr2;
* (7.3) Flow from overpack to bentonite
SIDE fluxout1[Rin1=rover]==SA1*(C2[Rin2=rover1]*alpha2[Rin2=rover1] -
      C1*alpha1)/(0.5*(dr1+dr2));
* (7.4) Flow from outermost cell - mixing cell approximation
* (q may be a function of time)
SIDE fluxout2[Rin2=rbent]=switch(C2-Cstar0*R2, C2/R2+kpc2*(R2-1)*C2/R2,
      Cstar0+kpc2*(R2-1)*Cstar0+k2*(C2-Cstar0*R2))*q;
* (7.5) Flux from glass into innermost cell of overpack
SIDE fluxglass=-SAglass*(C1[Rin1=rglass]*alpha1[Rin1=rglass]
      -Cglass*alphaglass)/((dr1+rglass)/2);
*
* (Block 8) Evolution equations
*
* (8.1) Tc in glass decays and is transported out into overpack inner ring
EQUATION d(inventory, t)=-Kdecay*inventory-(fluxglass/ascale);
* (8.2) Innermost ring of overpack receives influx from glass
EQUATION d(C1, t)=-Kdecay*C1[Rin1<=rglass]+(fluxout1[Rin1=rglass]-fluxglass)
      / V1);
* (8.3) Transport through overpack
EQUATION d(C1, t)=-Kdecay*C1+(fluxout1-fluxout1[Rin1=Rin1-dr1, Rin1>rglass])
      / V1);
* (8.4) Transport from overpack to bentonite
EQUATION d(C2, t)=-Kdecay*C2[Rin2>rover1]+
      (fluxout2-fluxout2[Rin2=Rin2-dr2])/V2);
* (8.5) Transport through bentonite (including loss from outermost ring)
EQUATION d(C2, t)=-Kdecay*C2+
      (fluxout2[Rin2=rover1]-fluxout1[Rin1=rover])/V2);

```

```

* (8.6) Amount escaping is equal to loss from outermost ring
EQUATION d(escaped,t)=fluxout2[Rin2=rbent]/ascale;
* (8.7) Amount escaped and surviving is equal to amount escaped less decay
EQUATION d(outside,t)=fluxout2[Rin2=rbent]/ascale-Kdecay*outside;
*
* (Block 9) Initial values
*
INITIAL inventory=invmoles;
INITIAL C1=0;
INITIAL C2=0;
INITIAL escaped=0;
INITIAL outside=0;
*
* (Block 10) Weights - chosen to give correct relative scaling for variables
*
WEIGHT C1=1/(C1+Cstar0);
WEIGHT C2=1/(C2+Cstar0);
WEIGHT inventory=1/(inventory+Cstar0*vglass/ascale);
WEIGHT escaped=1/(escaped+C2[Rin2=rbent]/ascale+Cstar0*vglass/ascale);
WEIGHT outside=1/(outside+C2[Rin2=rbent]/ascale+Cstar0*vglass/ascale);
*
* (Block 11) Monitor time and magnitude of peak flux and time of
* "drop off"
*
* The current escaping flux is only considered to be a new peak value if it
* is "significantly" larger than the previous peak,
* i.e. if it exceeds it by more than cutoff times the solver error tolerance.
* The monitor variable drop_time records the time at which the escaping
* flux first drops below a fixed fraction (e.g. 95%) of the peak flux.
* The monitor variable peak_outside records the maximum amount of
* contaminant surviving outside at any time.
*
CONSTANT cutoff=3; * multiple of epsstep required for significance
MONITOR peak_time=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        t,peak_time);

INITIAL peak_time=t;
MONITOR peak_flux=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        fluxout2[Rin2=rbent]/ascale,peak_flux);

INITIAL peak_flux=0;
CONSTANT drop_frac=0.95; * fraction of peak flux
MONITOR drop_time=switch(peak_flux*drop_frac-(fluxout2[Rin2=rbent]/ascale),
                        t,drop_time);

INITIAL drop_time=t;
MONITOR peak_outside=max(outside,peak_outside);
INITIAL peak_outside=0;
*
* (End of central.spf)
*

```

```

*
* central.spf
*
* SPF file for Task 1.1 central case (IM3246)
*
* This file models the diffusive transport of Technetium Tc-99 through the
* near field of a glass/overpack corrosion product/bentonite system.
*
* JAW 02/02/94
*
* Version 1.7
*
* Modification history:
* Vn Who When What
* 1.0 JAW 02/02/94 First version
* 1.1 JAW 03/02/94 Improve variable scaling and layout of file
* 1.2 JAW 04/02/94 Add colloids and pseudo-colloids
* 1.3 JAW 08/02/94 Further minor layout changes and added "outside"
* 1.4 JAW 10/02/94 Add monitor variable drop_time
* 1.5 JAW 11/02/94 Add monitor variable peak_outside
* 1.6 JAW 14/02/94 Improve inner boundary condition
* 1.7 JAW 15/02/94 Further change to inner boundary condition
*                   + fixed error in (7.2)
*
* General notes:
* (a) Units.
* The time unit is years. Distances are in metres. Amount of contaminant
* remaining in the glass or escaping from the system is measured in moles
* of Tc-99.
* Concentrations of contaminant in the transported species are measured in
* moles per cubic metre *of water* multiplied by a scaling factor ascale.
* This factor serves to improve the numerical conditioning of the problem:
* without it typical transported concentrations would be many orders of
* magnitude smaller than the remaining inventory.
*
* (b) Geometry.
* The problem is posed in a 1D cylindrical coordinate system, assuming
* cylindrical symmetry. The height of the cylinder is taken as the height
* of the central glass waste canister.
* There are three distinct regions in the model:
* Region Inner Outer
* radius (m) radius (m)
* Glass 0 0.22
* Overpack 0.22 0.52
* Bentonite 0.52 1.50
* The glass is treated as a source term which forms the inner boundary
* condition.
* It is assumed that contaminant from the whole volume of the glass is
* instantaneously in equilibrium with the surrounding water.
* The overpack (strictly, overpack corrosion products) layer and bentonite
* layer are treated as two separate regions with a flux condition at
* their shared boundary.
* The outer boundary of the bentonite is a decompressed zone. This is
* represented by a mixing-cell treatment which forms the outer boundary
* condition.
*
* (Block 1) Control info.
*
LISTFILE; * generate list output
SPARSE; * use sparse matrix solver
ALGERR; * include errors on algebraic variables in timestep monitor
CONSTANT epsstep=1e-5; * value for absolute error tolerance per timestep
EPSSTEP epsstep; * set absolute error tolerance
INITSTEP 1.e-3; * initial timestep
MINTIME 10; * Set output timesteps (in years)
MAXTIME 1e7; * - write out every 10000 years
GEOMSTEP 1.258925412; * from t=0 to t=1e7 years
*ARITHSTEP 1e5; * from t=0 to t=1e7 years
*MAXTIME 1e4; * - write out every 10000 years
*ARITHSTEP 1e1; * from t=0 to t=1e7 years
*
* (Block 2) Fundamental constants
*
CONSTANT pi=4*arctan(1);
CONSTANT SecsPerYear=365.2422*24*60*60; * number of seconds in one year
CONSTANT ascale=1e10; * multiplicative factor for all molar concentrations
* in transported species - improves conditioning
CONSTANT thalf=2.1e5; * Half-life of Tc-99 in years
CONSTANT Kdecay=log(2)/thalf; * Equivalent decay constant

```

```

CONSTANT R1=1; * Retardation factor of Tc-99 in inner (overpack) layer
CONSTANT R2=2.8; * Retardation factor of Tc-99 in outer (bentonite) layer
CONSTANT Rglass=1; * Retardation factor of Tc-99 in glass
CONSTANT Dsol=3e-10*SecsPerYear; * diffusion coefficient of Tc solute in
    * both bentonite and overpack
CONSTANT Dpc1=Dsol/2; * diffusion coefficient of Tc pseudocolloid
    * in overpack
CONSTANT Dpc2=Dsol/2; * diffusion coefficient of Tc pseudocolloid
    * in bentonite
CONSTANT Dtc1=Dsol/2; * diffusion coefficient of Tc true colloid
    * in overpack
CONSTANT Dtc2=Dsol/2; * diffusion coefficient of Tc true colloid
    * in bentonite
CONSTANT thetag=0.33; * Porosity of glass
CONSTANT theta1=0.33; * Porosity of overpack
CONSTANT theta2=0.33; * Porosity of bentonite
CONSTANT ZTc=99; * atomic weight of Tc
CONSTANT invmass=1.06e3; * Inventory at calculation start in g
CONSTANT invmoles=invmass/ZTc; * Initial inventory in moles
CONSTANT rglass=0.22; * outer glass radius
CONSTANT rover1=0.52; * actual outer overpack radius
CONSTANT rbent1=1.50; * actual outer bentonite radius
CONSTANT h=1.35; * height of cylinder
CONSTANT vglass=pi*h*rglass*rglass*thetag; * volume of glass
*
* (Block 3) Mixing cell constants
*
CONSTANT q=0.71*1e-3; * mean flow through mixing cell in m3/year
CONSTANT Cstar0=1e-9*ascale; * Solubility limit of Tc-99 in mol/m3 in bentonite
*
* (Block 4) Define geometry.
*
* The problem is posed in 1D cylindrical coordinates. Although
* Spade does not understand about these at present, we can use
* 1D cartesian coordinates because we discretize the
* spatial derivatives ourselves.
*
* The spatial variables Rin1 and Rin2 are chosen to represent the INNER
* radius of each ring. We therefore set the grid spacing so that the
* OUTER radius of the outermost cylinder in each region matches the
* specified dimensions.
*
CONSTANT nover=5; * number of grid points in overpack
CONSTANT nbent=10; * number of grid points in bentonite
CONSTANT rover=rglass+(rover1-rglass)*(nover-1)/nover; * grid limit
    * for overpack
CONSTANT rbent=rover1+(rbent1-rover1)*(nbent-1)/nbent; * grid limit
    * for bentonite
CONSTANT dr1=(rover1-rglass)/nover; * grid spacing in overpack
CONSTANT dr2=(rbent1-rover1)/nbent; * grid spacing in bentonite
SPATIAL Rin1, UNIFORM, rglass, rover, nover; * INNER radius of each ring
SPATIAL Rin2, UNIFORM, rover1, rbent, nbent;
REGION over, CART1D, Rin1; * overpack grid
REGION bent, CART1D, Rin2; * bentonite grid
VECTOR Rout1[over]; * OUTER radius of each ring in overpack
VECTOR Rout2[bent]; * OUTER radius of each ring in bentonite
CONSTANT Rout1=Rin1+dr1;
CONSTANT Rout2=Rin2+dr2;
VECTOR V1[over]; * volume of WATER in each ring in overpack
VECTOR V2[bent]; * volume of WATER in each ring in bentonite
CONSTANT V1=theta1*pi*h*((Rout1*Rout1) - (Rin1*Rin1));
CONSTANT V2=theta2*pi*h*((Rout2*Rout2) - (Rin2*Rin2));
VECTOR SA1[over]; * OUTER surface area of each ring in overpack
VECTOR SA2[bent]; * OUTER surface area of each ring in bentonite
CONSTANT SA1=2*pi*h*Rout1;
CONSTANT SA2=2*pi*h*Rout2;
CONSTANT SAglass=2*pi*h*rglass; * outer surface area of glass
SIDE Cglass=ascale*inventory/vglass; * concentration in glass
*
* Variable Cstar1 for radiolysis model
*
VECTOR Cstar1[over];
CONSTANT CsT1=200; * rise time for radiolysis
CONSTANT CsT2=1000; * fall time for radiolysis
*CONSTANT Cstar1=Cstar0*(1+1e6*(1-((Rin1-rglass)/(rover-rglass)))));
*SIDE Cstar1=Cstar0*(1+1e6*(1-((Rin1-rglass)/(rover-rglass))))
* * (heavy(t)-heavy(CsT2-t));
EQUATION Cstar1=Cstar0*(1+1e6*(1-((Rin1-rglass)/(rover-rglass)))

```

```

*(1-exp(-t/CsT1)-(1-exp(-t/CsT2))));
INITIAL Cstar1=Cstar0;
WEIGHT Cstar1=1/Cstar1;

*
* (Block 5) Declare remaining vector quantities
*
* the concentration vector within each region
VECTOR C1[over];
VECTOR C2[bent];
* the effective diffusivity
VECTOR alpha1[over];
VECTOR alpha2[bent];
* the flux across each outer boundary
VECTOR fluxout1[over];
VECTOR fluxout2[bent];
*
* (Block 6) Define alpha. The form of alpha varies depending on the assumed
* distribution of mobile species as a function of concentration.
* For the central case with no colloids or pseudocolloids, k1=k2=kpc1=kpc2=0.
*
CONSTANT k1=0; * fraction of true colloid in overpack
CONSTANT k2=0; * fraction of true colloid in bentonite
CONSTANT kpc1=0; * fraction of pseudocolloid in overpack
CONSTANT kpc2=0; * fraction of pseudocolloid in bentonite
* (6.1) alpha in overpack
SIDE alpha1=switch(C1-Cstar1*R1,
    Dsol/R1+kpc1*(R1-1)*C1*Dpc1/R1,
    (Dsol*Cstar1+kpc1*(R1-1)*Cstar1*Dpc1+k1*Dtc1*(C1-Cstar1*R1))/C1);
* (6.2) alpha in bentonite
SIDE alpha2=switch(C2-Cstar0*R2,
    Dsol/R2+kpc2*(R2-1)*C2*Dpc2/R2,
    (Dsol*Cstar0+kpc2*(R2-1)*Cstar0*Dpc2+k2*Dtc2*(C2-Cstar0*R2))/C2);
* (6.3) alpha in glass
SIDE alphaglass=switch(Cglass-Cstar1[Rin1=rglass]*Rglass,
    Dsol/Rglass+kpc1*(Rglass-1)*Cglass*Dpc1/Rglass,
    (Dsol*Cstar1[Rin1=rglass]+kpc1*(Rglass-1)*Cstar1[Rin1=rglass]*Dpc1+k1*Dtc1*(Cglass-Cstar1[Rin1=rgl
ass]*Rglass))/Cglass);
*
* (Block 7) Define fluxout, which gives the flux of Tc across each outer
* face (positive = net outward flux).
* The outer boundary and the boundary between the overpack and bentonite
* have to be handled separately.
*
* (7.1) Flow through overpack
SIDE fluxout1[Rin1<rover]=-SA1*(C1[Rin1=Rin1+dr1]*alpha1[Rin1=Rin1+dr1] -
    C1*alpha1)/dr1;
* (7.2) Flow through bentonite
SIDE fluxout2[Rin2<rbent]=-SA2*(C2[Rin2=Rin2+dr2]*alpha2[Rin2=Rin2+dr2] -
    C2*alpha2)/dr2;
* (7.3) Flow from overpack to bentonite
SIDE fluxout1[Rin1=rover]=-SA1*(C2[Rin2=rover1]*alpha2[Rin2=rover1] -
    C1*alpha1)/(0.5*(dr1+dr2));
* (7.4) Flow from outermost cell - mixing cell approximation
* (q may be a function of time)
SIDE fluxout2[Rin2=rbent]=switch(C2-Cstar0*R2, C2/R2+kpc2*(R2-1)*C2/R2,
    Cstar0+kpc2*(R2-1)*Cstar0+k2*(C2-Cstar0*R2)*q;
* (7.5) Flux from glass into innermost cell of overpack
SIDE fluxglass=-SAglass*(C1[Rin1=rglass]*alpha1[Rin1=rglass]
    -Cglass*alphaglass)/((dr1+rglass)/2);
*
* (Block 8) Evolution equations
*
* (8.1) Tc in glass decays and is transported out into overpack inner ring
EQUATION d(inventory, t)=-Kdecay*inventory-(fluxglass/ascale);
* (8.2) Innermost ring of overpack receives influx from glass
EQUATION d(C1, t)=- (Kdecay*C1[Rin1<=rglass]+(fluxout1[Rin1=rglass]-fluxglass)
    / V1);
* (8.3) Transport through overpack
EQUATION d(C1, t)=- (Kdecay*C1+(fluxout1-fluxout1[Rin1=Rin1-dr1, Rin1>rglass])
    / V1);
* (8.4) Transport from overpack to bentonite
EQUATION d(C2, t)=- (Kdecay*C2[Rin2>rover1]+
    (fluxout2-fluxout2[Rin2=Rin2-dr2])/V2);
* (8.5) Transport through bentonite (including loss from outermost ring)
EQUATION d(C2, t)=- (Kdecay*C2+
    (fluxout2[Rin2=rover1]-fluxout1[Rin1=rover])/V2);
* (8.6) Amount escaping is equal to loss from outermost ring

```

```

EQUATION d(escaped,t)=fluxout2[Rin2=rbent]/ascale;
* (8.7) Amount escaped and surviving is equal to amount escaped less decay
EQUATION d(outside,t)=fluxout2[Rin2=rbent]/ascale-Kdecay*outside;
*
* (Block 9) Initial values
*
INITIAL inventory=invmoles;
INITIAL C1=0;
INITIAL C2=0;
INITIAL escaped=0;
INITIAL outside=0;
*
* (Block 10) Weights - chosen to give correct relative scaling for variables
*
WEIGHT C1=1/(C1+Cstar0);
WEIGHT C2=1/(C2+Cstar0);
WEIGHT inventory=1/(inventory+Cstar0*vglass/ascale);
WEIGHT escaped=1/(escaped+C2[Rin2=rbent]/ascale+Cstar0*vglass/ascale);
WEIGHT outside=1/(outside+C2[Rin2=rbent]/ascale+Cstar0*vglass/ascale);
*
* (Block 11) Monitor time and magnitude of peak flux and time of
* "drop off"
*
* The current escaping flux is only considered to be a new peak value if it
* is "significantly" larger than the previous peak,
* i.e. if it exceeds it by more than cutoff times the solver error tolerance.
* The monitor variable drop_time records the time at which the escaping
* flux first drops below a fixed fraction (e.g. 95%) of the peak flux.
* The monitor variable peak_outside records the maximum amount of
* contaminant surviving outside at any time.
*
CONSTANT cutoff=3; * multiple of epsstep required for significance
MONITOR peak_time=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        t,peak_time);

INITIAL peak_time=t;
MONITOR peak_flux=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        fluxout2[Rin2=rbent]/ascale,peak_flux);

INITIAL peak_flux=0;
CONSTANT drop_frac=0.95; * fraction of peak flux
MONITOR drop_time=switch(peak_flux*drop_frac-(fluxout2[Rin2=rbent]/ascale),
                        t,drop_time);

INITIAL drop_time=t;
MONITOR peak_outside=max(outside,peak_outside);
INITIAL peak_outside=0;
*
* (End of central.spf)
*

```

```

*
* central.spf
*
* SPF file for Task 1.1 central case (IM3246)
*
* This file models the diffusive transport of Technetium Tc-99 through the
* near field of a glass/overpack corrosion product/bentonite system.
*
* JAW 02/02/94
*
* Version 1.7
*
* Modification history:
* Vn Who When What
* 1.0 JAW 02/02/94 First version
* 1.1 JAW 03/02/94 Improve variable scaling and layout of file
* 1.2 JAW 04/02/94 Add colloids and pseudo-colloids
* 1.3 JAW 08/02/94 Further minor layout changes and added "outside"
* 1.4 JAW 10/02/94 Add monitor variable drop_time
* 1.5 JAW 11/02/94 Add monitor variable peak_outside
* 1.6 JAW 14/02/94 Improve inner boundary condition
* 1.7 JAW 15/02/94 Further change to inner boundary condition
*                   + fixed error in (7.2)
*
* General notes:
* (a) Units.
* The time unit is years. Distances are in metres. Amount of contaminant
* remaining in the glass or escaping from the system is measured in moles
* of Tc-99.
* Concentrations of contaminant in the transported species are measured in
* moles per cubic metre *of water* multiplied by a scaling factor ascale.
* This factor serves to improve the numerical conditioning of the problem:
* without it typical transported concentrations would be many orders of
* magnitude smaller than the remaining inventory.
*
* (b) Geometry.
* The problem is posed in a 1D cylindrical coordinate system, assuming
* cylindrical symmetry. The height of the cylinder is taken as the height
* of the central glass waste canister.
* There are three distinct regions in the model:
* Region      Inner      Outer
*              radius (m)  radius (m)
* Glass       0          0.22
* Overpack    0.22       0.52
* Bentonite   0.52       1.50
* The glass is treated as a source term which forms the inner boundary
* condition.
* It is assumed that contaminant from the whole volume of the glass is
* instantaneously in equilibrium with the surrounding water.
* The overpack (strictly, overpack corrosion products) layer and bentonite
* layer are treated as two separate regions with a flux condition at
* their shared boundary.
* The outer boundary of the bentonite is a decompressed zone. This is
* represented by a mixing-cell treatment which forms the outer boundary
* condition.
*
* (Block 1) Control info.
*
LISTFILE; * generate list output
SPARSE; * use sparse matrix solver
ALGERR; * include errors on algebraic variables in timestep monitor
CONSTANT epsstep=1e-5; * value for absolute error tolerance per timestep
EPSSTEP epsstep; * set absolute error tolerance
INITSTEP 1.e-3; * initial timestep
MINTIME 10; * Set output timesteps (in years)
MAXTIME 1e7; * - write out every 10000 years
GEOMSTEP 1.258925412; * from t=0 to t=1e7 years
*ARITHSTEP 1e5; * from t=0 to t=1e7 years
*
* (Block 2) Fundamental constants
*
CONSTANT pi=4*arctan(1);
CONSTANT SecsPerYear=365.2422*24*60*60; * number of seconds in one year
CONSTANT ascale=1e10; * multiplicative factor for all molar concentrations
* in transported species - improves conditioning
CONSTANT thalf=2.1e5; * Half-life of Tc-99 in years
CONSTANT Kdecay=log(2)/thalf; * Equivalent decay constant
CONSTANT R1=1; * Retardation factor of Tc-99 in inner (overpack) layer
CONSTANT R2=2.8; * Retardation factor of Tc-99 in outer (bentonite) layer

```



```

CONSTANT Rglass=1; * Retardation factor of Tc-99 in glass
CONSTANT Dsol=3e-10*SecsPerYear; * diffusion coefficient of Tc solute in
* both bentonite and overpack
CONSTANT Dpc1=Dsol/2; * diffusion coefficient of Tc pseudocolloid
* in overpack
CONSTANT Dpc2=Dsol/2; * diffusion coefficient of Tc pseudocolloid
* in bentonite
CONSTANT Dtc1=Dsol/2; * diffusion coefficient of Tc true colloid
* in overpack
CONSTANT Dtc2=Dsol/2; * diffusion coefficient of Tc true colloid
* in bentonite
CONSTANT thetag=0.33; * Porosity of glass
CONSTANT theta1=0.33; * Porosity of overpack
CONSTANT theta2=0.33; * Porosity of bentonite
CONSTANT ZTc=99; * atomic weight of Tc
CONSTANT invmass=1.06e3; * Inventory at calculation start in g
CONSTANT invmoles=invmass/ZTc; * Initial inventory in moles
CONSTANT rglass=0.22; * outer glass radius
CONSTANT rover1=0.52; * actual outer overpack radius
CONSTANT rbent1=1.50; * actual outer bentonite radius
CONSTANT h=1.35; * height of cylinder
CONSTANT vglass=pi*h*rglass*rglass*thetag; * volume of glass
*
* (Block 3) Mixing cell constants
*
CONSTANT q=0.71*1e-3; * mean flow through mixing cell in m3/year
CONSTANT qyear=1e4; * half-width of gaussian peak in years
CONSTANT Tq=100000; * period of variation in q (in years)
CONSTANT qwidth=qyear/Tq; * fractional width of gaussian peak
CONSTANT wq=2*pi/Tq; * equivalent angular frequency
EQUATION qpar=(1/pi)*((pi/2)+arcsin(sin(wq*t-(pi/2)))); * varies linearly from 0 to 1 and
INITIAL qpar=0;
WEIGHT qpar=1/1000;
* back every Tq years
SIDE qadd=exp(-qpar*qpar/(qwidth*qwidth));
INITIAL qadd=1;
EQUATION Cstar=1e-9*ascale*(1+1e8*qadd); * Solubility limit of Tc-99 in mol/m3
INITIAL Cstar=1e-9*ascale*1e8;
WEIGHT Cstar=1/(100*Cstar);
*
* (Block 4) Define geometry.
*
* The problem is posed in 1D cylindrical coordinates. Although
* Spade does not understand about these at present, we can use
* 1D cartesian coordinates because we discretize the
* spatial derivatives ourselves.
*
* The spatial variables Rin1 and Rin2 are chosen to represent the INNER
* radius of each ring. We therefore set the grid spacing so that the
* OUTER radius of the outermost cylinder in each region matches the
* specified dimensions.
*
CONSTANT nover=5; * number of grid points in overpack
CONSTANT nbent=10; * number of grid points in bentonite
CONSTANT rover=rglass+(rover1-rglass)*(nover-1)/nover; * grid limit
* for overpack
CONSTANT rbent=rover1+(rbent1-rover1)*(nbent-1)/nbent; * grid limit
* for bentonite
CONSTANT dr1=(rover1-rglass)/nover; * grid spacing in overpack
CONSTANT dr2=(rbent1-rover1)/nbent; * grid spacing in bentonite
SPATIAL Rin1,UNIFORM,rglass,rover,nover; * INNER radius of each ring
SPATIAL Rin2,UNIFORM,rover1,rbent,nbent;
REGION over,CART1D,Rin1; * overpack grid
REGION bent,CART1D,Rin2; * bentonite grid
VECTOR Rout1[over]; * OUTER radius of each ring in overpack
VECTOR Rout2[bent]; * OUTER radius of each ring in bentonite
CONSTANT Rout1=Rin1+dr1;
CONSTANT Rout2=Rin2+dr2;
VECTOR V1[over]; * volume of WATER in each ring in overpack
VECTOR V2[bent]; * volume of WATER in each ring in bentonite
CONSTANT V1=theta1*pi*h*((Rout1*Rout1)-(Rin1*Rin1));
CONSTANT V2=theta2*pi*h*((Rout2*Rout2)-(Rin2*Rin2));
VECTOR SA1[over]; * OUTER surface area of each ring in overpack
VECTOR SA2[bent]; * OUTER surface area of each ring in bentonite
CONSTANT SA1=2*pi*h*Rout1;
CONSTANT SA2=2*pi*h*Rout2;
CONSTANT SAglass=2*pi*h*rglass; * outer surface area of glass
SIDE Cglass=ascale*inventory/vglass; * concentration in glass
*

```

```

* (Block 5) Declare remaining vector quantities
*
* the concentration vector within each region
VECTOR C1[over];
VECTOR C2[bent];
* the effective diffusivity
VECTOR alpha1[over];
VECTOR alpha2[bent];
* the flux across each outer boundary
VECTOR fluxout1[over];
VECTOR fluxout2[bent];
*
* (Block 6) Define alpha. The form of alpha varies depending on the assumed
* distribution of mobile species as a function of concentration.
* For the central case with no colloids or pseudocolloids, k1=k2=kpc1=kpc2=0.
*
CONSTANT k1=0; * fraction of true colloid in overpack
CONSTANT k2=0; * fraction of true colloid in bentonite
CONSTANT kpc1=0; * fraction of pseudocolloid in overpack
CONSTANT kpc2=0; * fraction of pseudocolloid in bentonite
* (6.1) alpha in overpack
SIDE alpha1=switch(C1-Cstar*R1,
                  Dsol/R1+kpc1*(R1-1)*C1*Dpc1/R1,
                  (Dsol*Cstar+kpc1*(R1-1)*Cstar*Dpc1+k1*Dtc1*(C1-Cstar*R1))/C1);
* (6.2) alpha in bentonite
SIDE alpha2=switch(C2-Cstar*R2,
                  Dsol/R2+kpc2*(R2-1)*C2*Dpc2/R2,
                  (Dsol*Cstar+kpc2*(R2-1)*Cstar*Dpc2+k2*Dtc2*(C2-Cstar*R2))/C2);
* (6.3) alpha in glass
SIDE alphaglass=switch(Cglass-Cstar*Rglass,
                      Dsol/Rglass+kpc1*(Rglass-1)*Cglass*Dpc1/Rglass,
                      (Dsol*Cstar+kpc1*(Rglass-1)*Cstar*Dpc1+k1*Dtc1*(Cglass-Cstar*Rglass))/Cglass);
*
* (Block 7) Define fluxout, which gives the flux of Tc across each outer
* face (positive = net outward flux).
* The outer boundary and the boundary between the overpack and bentonite
* have to be handled separately.
*
* (7.1) Flow through overpack
SIDE fluxout1[Rin1<rover]=--SA1*(C1[Rin1=Rin1+dr1]*alpha1[Rin1=Rin1+dr1] -
                               C1*alpha1)/dr1;
* (7.2) Flow through bentonite
SIDE fluxout2[Rin2<rbent]=--SA2*(C2[Rin2=Rin2+dr2]*alpha2[Rin2=Rin2+dr2] -
                               C2*alpha2)/dr2;
* (7.3) Flow from overpack to bentonite
SIDE fluxout1[Rin1=rover]=--SA1*(C2[Rin2=rover1]*alpha2[Rin2=rover1] -
                               C1*alpha1)/(0.5*(dr1+dr2));
* (7.4) Flow from outermost cell - mixing cell approximation
* (q may be a function of time)
SIDE fluxout2[Rin2=rbent]=switch(C2-Cstar*R2, C2/R2+kpc2*(R2-1)*C2/R2,
                                Cstar+kpc2*(R2-1)*Cstar+k2*(C2-Cstar*R2))*q;
* (7.5) Flux from glass into innermost cell of overpack
SIDE fluxglass=--SAglass*(C1[Rin1=rglass]*alpha1[Rin1=rglass]
                          -Cglass*alphaglass)/((dr1+rglass)/2);
*
* (Block 8) Evolution equations
*
* (8.1) Tc in glass decays and is transported out into overpack inner ring
EQUATION d(inventory, t)=-Kdecay*inventory-(fluxglass/ascale);
* (8.2) Innermost ring of overpack receives influx from glass
EQUATION d(C1, t)=-(Kdecay*C1[Rin1<=rglass]+(fluxout1[Rin1=rglass]-fluxglass)
                  / V1);
* (8.3) Transport through overpack
EQUATION d(C1, t)=-(Kdecay*C1+(fluxout1-fluxout1[Rin1=Rin1-dr1, Rin1>rglass])
                  / V1);
* (8.4) Transport from overpack to bentonite
EQUATION d(C2, t)=-(Kdecay*C2[Rin2>rover1]+
                  (fluxout2-fluxout2[Rin2=Rin2-dr2])/V2);
* (8.5) Transport through bentonite (including loss from outermost ring)
EQUATION d(C2, t)=-(Kdecay*C2+
                  (fluxout2[Rin2=rover1]-fluxout1[Rin1=rover])/V2);
* (8.6) Amount escaping is equal to loss from outermost ring
EQUATION d(escaped, t)=fluxout2[Rin2=rbent]/ascale;
* (8.7) Amount escaped and surviving is equal to amount escaped less decay
EQUATION d(outside, t)=fluxout2[Rin2=rbent]/ascale-Kdecay*outside;
*
* (Block 9) Initial values
*

```

```

INITIAL inventory=invmoles;
INITIAL C1=0;
INITIAL C2=0;
INITIAL escaped=0;
INITIAL outside=0;
*
* (Block 10) Weights - chosen to give correct relative scaling for variables
*
WEIGHT C1=1/(C1+Cstar);
WEIGHT C2=1/(C2+Cstar);
WEIGHT inventory=1/(inventory+Cstar*vglass/ascale);
WEIGHT escaped=1/(escaped+C2 [Rin2=rbent]/ascale+Cstar*vglass/ascale);
WEIGHT outside=1/(outside+C2 [Rin2=rbent]/ascale+Cstar*vglass/ascale);
*
* (Block 11) Monitor time and magnitude of peak flux and time of
* "drop off"
*
* The current escaping flux is only considered to be a new peak value if it
* is "significantly" larger than the previous peak,
* i.e. if it exceeds it by more than cutoff times the solver error tolerance.
* The monitor variable drop_time records the time at which the escaping
* flux first drops below a fixed fraction (e.g. 95%) of the peak flux.
* The monitor variable peak_outside records the maximum amount of
* contaminant surviving outside at any time.
*
CONSTANT cutoff=3; * multiple of epsstep required for significance
MONITOR peak_time=switch(peak_flux*(1+cutoff*epsstep)-
    (fluxout2 [Rin2=rbent]/ascale),
    t, peak_time);

INITIAL peak_time=t;
MONITOR peak_flux=switch(peak_flux*(1+cutoff*epsstep)-
    (fluxout2 [Rin2=rbent]/ascale),
    fluxout2 [Rin2=rbent]/ascale, peak_flux);

INITIAL peak_flux=0;
CONSTANT drop_frac=0.95; * fraction of peak flux
MONITOR drop_time=switch(peak_flux*drop_frac-(fluxout2 [Rin2=rbent]/ascale),
    t, drop_time);

INITIAL drop_time=t;
MONITOR peak_outside=max(outside, peak_outside);
INITIAL peak_outside=0;
*
* (End of central.spf)
*

```

```

*
* central.spf
*
* SPF file for Task 1.1 central case (IM3246)
*
* This file models the diffusive transport of Technetium Tc-99 through the
* near field of a glass/overpack corrosion product/bentonite system.
*
* JAW 02/02/94
*
* Version 1.7
*
* Modification history:
* Vn Who When What
* 1.0 JAW 02/02/94 First version
* 1.1 JAW 03/02/94 Improve variable scaling and layout of file
* 1.2 JAW 04/02/94 Add colloids and pseudo-colloids
* 1.3 JAW 08/02/94 Further minor layout changes and added "outside"
* 1.4 JAW 10/02/94 Add monitor variable drop_time
* 1.5 JAW 11/02/94 Add monitor variable peak_outside
* 1.6 JAW 14/02/94 Improve inner boundary condition
* 1.7 JAW 15/02/94 Further change to inner boundary condition
*                   + fixed error in (7.2)
*
* General notes:
* (a) Units.
* The time unit is years. Distances are in metres. Amount of contaminant
* remaining in the glass or escaping from the system is measured in moles
* of Tc-99.
* Concentrations of contaminant in the transported species are measured in
* moles per cubic metre *of water* multiplied by a scaling factor ascale.
* This factor serves to improve the numerical conditioning of the problem:
* without it typical transported concentrations would be many orders of
* magnitude smaller than the remaining inventory.
*
* (b) Geometry.
* The problem is posed in a 1D cylindrical coordinate system, assuming
* cylindrical symmetry. The height of the cylinder is taken as the height
* of the central glass waste canister.
* There are three distinct regions in the model:
* Region Inner Outer
* radius (m) radius (m)
* Glass 0 0.22
* Overpack 0.22 0.52
* Bentonite 0.52 1.50
* The glass is treated as a source term which forms the inner boundary
* condition.
* It is assumed that contaminant from the whole volume of the glass is
* instantaneously in equilibrium with the surrounding water.
* The overpack (strictly, overpack corrosion products) layer and bentonite
* layer are treated as two separate regions with a flux condition at
* their shared boundary.
* The outer boundary of the bentonite is a decompressed zone. This is
* represented by a mixing-cell treatment which forms the outer boundary
* condition.
*
* (Block 1) Control info.
*
LISTFILE; * generate list output
SPARSE; * use sparse matrix solver
ALGERR; * include errors on algebraic variables in timestep monitor
CONSTANT epsstep=1e-5; * value for absolute error tolerance per timestep
EPSSTEP epsstep; * set absolute error tolerance
INITSTEP 1.e-3; * initial timestep
MINTIME 0; * Set output timesteps (in years)
MAXTIME 6e6; * - write out every 10000 years
ARITHSTEP 1e5; * from t=0 to t=1e7 years
*
* (Block 2) Fundamental constants
*
CONSTANT pi=4*arctan(1);
CONSTANT SecsPerYear=365.2422*24*60*60; * number of seconds in one year
CONSTANT ascale=1e10; * multiplicative factor for all molar concentrations
* in transported species - improves conditioning
CONSTANT thalf=2.1e5; * Half-life of Tc-99 in years
CONSTANT Kdecay=log(2)/thalf; * Equivalent decay constant
CONSTANT Cstar=1e-9*ascale; * Solubility limit of Tc-99 in mol/m3
CONSTANT R1=1; * Retardation factor of Tc-99 in inner (overpack) layer
CONSTANT R2=2.8; * Retardation factor of Tc-99 in outer (bentonite) layer

```

```

CONSTANT Rglass=1; * Retardation factor of Tc-99 in glass
CONSTANT Dsol=3e-10*SecsPerYear; * diffusion coefficient of Tc solute in
      * both bentonite and overpack
CONSTANT Dpc1=Dsol/2; * diffusion coefficient of Tc pseudocolloid
      * in overpack
CONSTANT Dpc2=Dsol/2; * diffusion coefficient of Tc pseudocolloid
      * in bentonite
CONSTANT Dtc1=Dsol/2; * diffusion coefficient of Tc true colloid
      * in overpack
CONSTANT Dtc2=Dsol/2; * diffusion coefficient of Tc true colloid
      * in bentonite
CONSTANT thetag=0.33; * Porosity of glass
CONSTANT theta1=0.33; * Porosity of overpack
CONSTANT theta2=0.33; * Porosity of bentonite
CONSTANT ZTc=99; * atomic weight of Tc
CONSTANT invmass=1.06e3; * Inventory at calculation start in g
CONSTANT invmoles=invmass/ZTc; * Initial inventory in moles
CONSTANT rglass=0.22; * outer glass radius
CONSTANT rover1=0.52; * actual outer overpack radius
CONSTANT rbent1=1.50; * actual outer bentonite radius
CONSTANT h=1.35; * height of cylinder
CONSTANT vglass=pi*h*rglass*rglass*thetag; * volume of glass
*
* (Block 3) Mixing cell constants
*
CONSTANT q0=0.71*1e-3; * mean flow through mixing cell in m3/year
CONSTANT qfrac=1e4; * fractional variation in mixing cell flow, used to
      * derive q - central case value is qfrac=0
CONSTANT q1=q0*qfrac;
CONSTANT qyear=1e4; * half-width of gaussian peak in years
CONSTANT Tq=30000; * period of variation in q (in years)
CONSTANT qwidth=qyear/Tq; * fractional width of gaussian peak
CONSTANT qamp=q0*qfrac; * amplitude of variation in q
CONSTANT wq=2*pi/Tq; * equivalent angular frequency
EQUATION qpar=(1/pi)*((pi/2)+arcsin(sin(wq*t-(pi/2))))); * varies linearly from 0 to 1 and
      * back every Tq years

INITIAL qpar=0;
WEIGHT qpar=1/1000;
SIDE qadd=exp(-qpar*qpar/(qwidth*qwidth));
INITIAL qadd=1;
EQUATION q=q0+q1*qadd; * mixing cell flow rate (potentially time dependent)
INITIAL q=q0+q1*qadd;
WEIGHT q=1/(q*100);
*
* (Block 4) Define geometry.
*
* The problem is posed in 1D cylindrical coordinates. Although
* Spade does not understand about these at present, we can use
* 1D cartesian coordinates because we discretize the
* spatial derivatives ourselves.
*
* The spatial variables Rin1 and Rin2 are chosen to represent the INNER
* radius of each ring. We therefore set the grid spacing so that the
* OUTER radius of the outermost cylinder in each region matches the
* specified dimensions.
*
CONSTANT nover=5; * number of grid points in overpack
CONSTANT nbent=10; * number of grid points in bentonite
CONSTANT rover=rglass+(rover1-rglass)*(nover-1)/nover; * grid limit
      * for overpack
CONSTANT rbent=rover1+(rbent1-rover1)*(nbent-1)/nbent; * grid limit
      * for bentonite

CONSTANT dr1=(rover1-rglass)/nover; * grid spacing in overpack
CONSTANT dr2=(rbent1-rover1)/nbent; * grid spacing in bentonite
SPATIAL Rin1,UNIFORM,rglass,rover,nover; * INNER radius of each ring
SPATIAL Rin2,UNIFORM,rover1,rbent,nbent;
REGION over,CART1D,Rin1; * overpack grid
REGION bent,CART1D,Rin2; * bentonite grid
VECTOR Rout1[over]; * OUTER radius of each ring in overpack
VECTOR Rout2[bent]; * OUTER radius of each ring in bentonite
CONSTANT Rout1=Rin1+dr1;
CONSTANT Rout2=Rin2+dr2;
VECTOR V1[over]; * volume of WATER in each ring in overpack
VECTOR V2[bent]; * volume of WATER in each ring in bentonite
CONSTANT V1=theta1*pi*h*((Rout1*Rout1)-(Rin1*Rin1));
CONSTANT V2=theta2*pi*h*((Rout2*Rout2)-(Rin2*Rin2));
VECTOR SA1[over]; * OUTER surface area of each ring in overpack
VECTOR SA2[bent]; * OUTER surface area of each ring in bentonite
CONSTANT SA1=2*pi*h*Rout1;

```

```

CONSTANT SA2=2*pi*h*Rout2;
CONSTANT SAglass=2*pi*h*rglass; * outer surface area of glass
SIDE Cglass=ascale*inventory/vglass; * concentration in glass
*
* (Block 5) Declare remaining vector quantities
*
* the concentration vector within each region
VECTOR C1[over];
VECTOR C2[bent];
* the effective diffusivity
VECTOR alpha1[over];
VECTOR alpha2[bent];
* the flux across each outer boundary
VECTOR fluxout1[over];
VECTOR fluxout2[bent];
*
* (Block 6) Define alpha. The form of alpha varies depending on the assumed
* distribution of mobile species as a function of concentration.
* For the central case with no colloids or pseudocolloids, k1=k2=kpc1=kpc2=0.
*
CONSTANT k1=0; * fraction of true colloid in overpack
CONSTANT k2=0; * fraction of true colloid in bentonite
CONSTANT kpc1=0; * fraction of pseudocolloid in overpack
CONSTANT kpc2=0; * fraction of pseudocolloid in bentonite
* (6.1) alpha in overpack
SIDE alpha1=switch(C1-Cstar*R1,
    Dsol/R1+kpc1*(R1-1)*C1*Dpc1/R1,
    (Dsol*Cstar+kpc1*(R1-1)*Cstar*Dpc1+k1*Dtc1*(C1-Cstar*R1))/C1);
* (6.2) alpha in bentonite
SIDE alpha2=switch(C2-Cstar*R2,
    Dsol/R2+kpc2*(R2-1)*C2*Dpc2/R2,
    (Dsol*Cstar+kpc2*(R2-1)*Cstar*Dpc2+k2*Dtc2*(C2-Cstar*R2))/C2);
* (6.3) alpha in glass
SIDE alphaglass=switch(Cglass-Cstar*Rglass,
    Dsol/Rglass+kpc1*(Rglass-1)*Cglass*Dpc1/Rglass,
    (Dsol*Cstar+kpc1*(Rglass-1)*Cstar*Dpc1+k1*Dtc1*(Cglass-Cstar*Rglass))/Cglass);
*
* (Block 7) Define fluxout, which gives the flux of Tc across each outer
* face (positive = net outward flux).
* The outer boundary and the boundary between the overpack and bentonite
* have to be handled separately.
*
* (7.1) Flow through overpack
SIDE fluxout1[Rin1<rover]=-SA1*(C1[Rin1=Rin1+dr1]*alpha1[Rin1=Rin1+dr1] -
    C1*alpha1)/dr1;
* (7.2) Flow through bentonite
SIDE fluxout2[Rin2<rbent]=-SA2*(C2[Rin2=Rin2+dr2]*alpha2[Rin2=Rin2+dr2] -
    C2*alpha2)/dr2;
* (7.3) Flow from overpack to bentonite
SIDE fluxout1[Rin1=rover]=-SA1*(C2[Rin2=rover1]*alpha2[Rin2=rover1] -
    C1*alpha1)/(0.5*(dr1+dr2));
* (7.4) Flow from outermost cell - mixing cell approximation
* (q may be a function of time)
SIDE fluxout2[Rin2=rbent]=switch(C2-Cstar*R2, C2/R2+kpc2*(R2-1)*C2/R2,
    Cstar+kpc2*(R2-1)*Cstar+k2*(C2-Cstar*R2))*q;
* (7.5) Flux from glass into innermost cell of overpack
SIDE fluxglass=-SAglass*(C1[Rin1=rglass]*alpha1[Rin1=rglass]
    -Cglass*alphaglass)/((dr1+rglass)/2);
*
* (Block 8) Evolution equations
*
* (8.1) Tc in glass decays and is transported out into overpack inner ring
EQUATION d(inventory, t)=-Kdecay*inventory-(fluxglass/ascale);
* (8.2) Innermost ring of overpack receives influx from glass
EQUATION d(C1, t)=- (Kdecay*C1[Rin1<=rglass]+(fluxout1[Rin1=rglass]-fluxglass)
    / V1);
* (8.3) Transport through overpack
EQUATION d(C1, t)=- (Kdecay*C1+(fluxout1-fluxout1[Rin1=Rin1-dr1, Rin1>rglass])
    / V1);
* (8.4) Transport from overpack to bentonite
EQUATION d(C2, t)=- (Kdecay*C2[Rin2>rover1]+
    (fluxout2-fluxout2[Rin2=Rin2-dr2])/V2);
* (8.5) Transport through bentonite (including loss from outermost ring)
EQUATION d(C2, t)=- (Kdecay*C2+
    (fluxout2[Rin2=rover1]-fluxout1[Rin1=rover1])/V2);
* (8.6) Amount escaping is equal to loss from outermost ring
EQUATION d(escaped, t)=fluxout2[Rin2=rbent]/ascale;
* (8.7) Amount escaped and surviving is equal to amount escaped less decay

```

```

EQUATION d(outside,t)=fluxout2[Rin2=rbent]/ascale-Kdecay*outside;
*
* (Block 9) Initial values
*
INITIAL inventory=invmoles;
INITIAL C1=0;
INITIAL C2=0;
INITIAL escaped=0;
INITIAL outside=0;
INITIAL q=q0+q1*sin(wq*t);
*
* (Block 10) Weights - chosen to give correct relative scaling for variables
*
WEIGHT C1=1/(C1+Cstar);
WEIGHT C2=1/(C2+Cstar);
WEIGHT inventory=1/(inventory+Cstar*vglass/ascale);
WEIGHT escaped=1/(escaped+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
WEIGHT outside=1/(outside+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
*
* (Block 11) Monitor time and magnitude of peak flux and time of
* "drop off"
*
* The current escaping flux is only considered to be a new peak value if it
* is "significantly" larger than the previous peak,
* i.e. if it exceeds it by more than cutoff times the solver error tolerance.
* The monitor variable drop_time records the time at which the escaping
* flux first drops below a fixed fraction (e.g. 95%) of the peak flux.
* The monitor variable peak_outside records the maximum amount of
* contaminant surviving outside at any time.
*
CONSTANT cutoff=3; * multiple of epsstep required for significance
MONITOR peak_time=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        t,peak_time);
INITIAL peak_time=t;
MONITOR peak_flux=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        fluxout2[Rin2=rbent]/ascale,peak_flux);
INITIAL peak_flux=0;
CONSTANT drop_frac=0.95; * fraction of peak flux
MONITOR drop_time=switch(peak_flux*drop_frac-(fluxout2[Rin2=rbent]/ascale),
                        t,drop_time);
INITIAL drop_time=t;
MONITOR peak_outside=max(outside,peak_outside);
INITIAL peak_outside=0;
*
* (End of central.spf)
*

```

```

*
* central.spf
*
* SPF file for Task 1.1 central case (IM3246)
*
* This file models the diffusive transport of Technetium Tc-99 through the
* near field of a glass/overpack corrosion product/bentonite system.
*
* JAW 02/02/94
*
* Version 1.7
*
* Modification history:
* Vn Who When What
* 1.0 JAW 02/02/94 First version
* 1.1 JAW 03/02/94 Improve variable scaling and layout of file
* 1.2 JAW 04/02/94 Add colloids and pseudo-colloids
* 1.3 JAW 08/02/94 Further minor layout changes and added "outside"
* 1.4 JAW 10/02/94 Add monitor variable drop_time
* 1.5 JAW 11/02/94 Add monitor variable peak_outside
* 1.6 JAW 14/02/94 Improve inner boundary condition
* 1.7 JAW 15/02/94 Further change to inner boundary condition
* + fixed error in (7.2)
*
* General notes:
* (a) Units.
* The time unit is years. Distances are in metres. Amount of contaminant
* remaining in the glass or escaping from the system is measured in moles
* of Tc-99.
* Concentrations of contaminant in the transported species are measured in
* moles per cubic metre *of water* multiplied by a scaling factor ascale.
* This factor serves to improve the numerical conditioning of the problem:
* without it typical transported concentrations would be many orders of
* magnitude smaller than the remaining inventory.
*
* (b) Geometry.
* The problem is posed in a 1D cylindrical coordinate system, assuming
* cylindrical symmetry. The height of the cylinder is taken as the height
* of the central glass waste canister.
* There are three distinct regions in the model:
* Region Inner Outer
* radius (m) radius (m)
* Glass 0 0.22
* Overpack 0.22 0.52
* Bentonite 0.52 1.50
* The glass is treated as a source term which forms the inner boundary
* condition.
* It is assumed that contaminant from the whole volume of the glass is
* instantaneously in equilibrium with the surrounding water.
* The overpack (strictly, overpack corrosion products) layer and bentonite
* layer are treated as two separate regions with a flux condition at
* their shared boundary.
* The outer boundary of the bentonite is a decompressed zone. This is
* represented by a mixing-cell treatment which forms the outer boundary
* condition.
*
* (Block 1) Control info.
*
LISTFILE; * generate list output
SPARSE; * use sparse matrix solver
ALGERR; * include errors on algebraic variables in timestep monitor
CONSTANT epsstep=1e-5; * value for absolute error tolerance per timestep
EPSSTEP epsstep; * set absolute error tolerance
INITSTEP 1.e-3; * initial timestep
MINTIME 10; * Set output timesteps (in years)
MAXTIME 1e7; * - write out every 10000 years
GEOMSTEP 1.258925412; * from t=0 to t=1e7 years
*ARITHSTEP 1e5; * from t=0 to t=1e7 years
*
* (Block 2) Fundamental constants
*
CONSTANT pi=4*arctan(1);
CONSTANT SecsPerYear=365.2422*24*60*60; * number of seconds in one year
CONSTANT ascale=1e10; * multiplicative factor for all molar concentrations
* in transported species - improves conditioning
CONSTANT thalf=2.1e5; * Half-life of Tc-99 in years
CONSTANT Kdecay=log(2)/thalf; * Equivalent decay constant
CONSTANT Cstar=1e-9*ascale; * Solubility limit of Tc-99 in mol/m3
CONSTANT R1=1; * Retardation factor of Tc-99 in inner (overpack) layer

```



```

CONSTANT R2=280; * Retardation factor of Tc-99 in outer (bentonite) layer
CONSTANT Rglass=1; * Retardation factor of Tc-99 in glass
CONSTANT Dsol=3e-10*SecsPerYear; * diffusion coefficient of Tc solute in
    * both bentonite and overpack
CONSTANT Dpc1=Dsol/10; * diffusion coefficient of Tc pseudocolloid
    * in overpack
CONSTANT Dpc2=Dsol/10; * diffusion coefficient of Tc pseudocolloid
    * in bentonite
CONSTANT Dtc1=Dsol/2; * diffusion coefficient of Tc true colloid
    * in overpack
CONSTANT Dtc2=Dsol/2; * diffusion coefficient of Tc true colloid
    * in bentonite
CONSTANT thetag=0.33; * Porosity of glass
CONSTANT theta1=0.33; * Porosity of overpack
CONSTANT theta2=0.33; * Porosity of bentonite
CONSTANT ZTc=99; * atomic weight of Tc
CONSTANT invmass=1.06e3; * Inventory at calculation start in g
CONSTANT invmoles=invmass/ZTc; * Initial inventory in moles
CONSTANT rglass=0.22; * outer glass radius
CONSTANT rover1=0.52; * actual outer overpack radius
CONSTANT rbent1=1.50; * actual outer bentonite radius
CONSTANT h=1.35; * height of cylinder
CONSTANT vglass=pi*h*rglass*rglass*thetag; * volume of glass
*
* (Block 3) Mixing cell constants
*
CONSTANT q0=0.71e-3; * mean flow through mixing cell in m3/year
CONSTANT qfrac=0; * fractional variation in mixing cell flow, used to
    * derive q - central case value is qfrac=0
CONSTANT q1=q0*qfrac; * amplitude of variation in q
CONSTANT Tq=2000; * period of variation in q (in years)
CONSTANT wq=2*pi/Tq; * equivalent angular frequency
SIDE q=q0+q1*sin(wq*t); * mixing cell flow rate (potentially time dependent)
*
* (Block 4) Define geometry.
*
* The problem is posed in 1D cylindrical coordinates. Although
* Spade does not understand about these at present, we can use
* 1D cartesian coordinates because we discretize the
* spatial derivatives ourselves.
*
* The spatial variables Rin1 and Rin2 are chosen to represent the INNER
* radius of each ring. We therefore set the grid spacing so that the
* OUTER radius of the outermost cylinder in each region matches the
* specified dimensions.
*
CONSTANT nover=5; * number of grid points in overpack
CONSTANT nbent=10; * number of grid points in bentonite
CONSTANT rover=rglass+(rover1-rglass)*(nover-1)/nover; * grid limit
    * for overpack
CONSTANT rbent=rover1+(rbent1-rover1)*(nbent-1)/nbent; * grid limit
    * for bentonite

CONSTANT dr1=(rover1-rglass)/nover; * grid spacing in overpack
CONSTANT dr2=(rbent1-rover1)/nbent; * grid spacing in bentonite
SPATIAL Rin1,UNIFORM,rglass,rover,nover; * INNER radius of each ring
SPATIAL Rin2,UNIFORM,rover1,rbent,nbent;
REGION over,CART1D,Rin1; * overpack grid
REGION bent,CART1D,Rin2; * bentonite grid
VECTOR Rout1[over]; * OUTER radius of each ring in overpack
VECTOR Rout2[bent]; * OUTER radius of each ring in bentonite
CONSTANT Rout1=Rin1+dr1;
CONSTANT Rout2=Rin2+dr2;
VECTOR V1[over]; * volume of WATER in each ring in overpack
VECTOR V2[bent]; * volume of WATER in each ring in bentonite
CONSTANT V1=theta1*pi*h*((Rout1*Rout1) - (Rin1*Rin1));
CONSTANT V2=theta2*pi*h*((Rout2*Rout2) - (Rin2*Rin2));
VECTOR SA1[over]; * OUTER surface area of each ring in overpack
VECTOR SA2[bent]; * OUTER surface area of each ring in bentonite
CONSTANT SA1=2*pi*h*Rout1;
CONSTANT SA2=2*pi*h*Rout2;
CONSTANT SAglass=2*pi*h*rglass; * outer surface area of glass
SIDE Cglass=ascale*inventory/vglass; * concentration in glass
*
* (Block 5) Declare remaining vector quantities
*
* the concentration vector within each region
VECTOR C1[over];
VECTOR C2[bent];
* the effective diffusivity

```

```

VECTOR alpha1[over];
VECTOR alpha2[bent];
* the flux across each outer boundary
VECTOR fluxout1[over];
VECTOR fluxout2[bent];
*
* (Block 6) Define alpha. The form of alpha varies depending on the assumed
* distribution of mobile species as a function of concentration.
* For the central case with no colloids or pseudocolloids, k1=k2=kpc1=kpc2=0.
*
CONSTANT k1=0.1; * fraction of true colloid in overpack
CONSTANT k2=0.1; * fraction of true colloid in bentonite
CONSTANT kpc1=0; * fraction of pseudocolloid in overpack
CONSTANT kpc2=0.1; * fraction of pseudocolloid in bentonite
* (6.1) alpha in overpack
SIDE alpha1=switch(C1-Cstar*R1,
    Dsol/R1+kpc1*(R1-1)*Dpc1/R1,
    (Dsol*Cstar+kpc1*(R1-1)*Cstar*Dpc1+k1*Dtc1*(C1-Cstar*R1))/C1);
* (6.2) alpha in bentonite
SIDE alpha2=switch(C2-Cstar*R2,
    Dsol/R2+kpc2*(R2-1)*Dpc2/R2,
    (Dsol*Cstar+kpc2*(R2-1)*Cstar*Dpc2+k2*Dtc2*(C2-Cstar*R2))/C2);
* (6.3) alpha in glass
SIDE alphaglass=switch(Cglass-Cstar*Rglass,
    Dsol/Rglass+kpc1*(Rglass-1)*Dpc1/Rglass,
    (Dsol*Cstar+kpc1*(Rglass-1)*Cstar*Dpc1+k1*Dtc1*(Cglass-Cstar*Rglass))/Cglass);
*
* (Block 7) Define fluxout, which gives the flux of To across each outer
* face (positive = net outward flux).
* The outer boundary and the boundary between the overpack and bentonite
* have to be handled separately.
*
* (7.1) Flow through overpack
SIDE fluxout1[Rin1<rover]=-SA1*(C1[Rin1=Rin1+dr1]*alpha1[Rin1=Rin1+dr1] -
    C1*alpha1)/dr1;
* (7.2) Flow through bentonite
SIDE fluxout2[Rin2<rbent]=-SA2*(C2[Rin2=Rin2+dr2]*alpha2[Rin2=Rin2+dr2] -
    C2*alpha2)/dr2;
* (7.3) Flow from overpack to bentonite
SIDE fluxout1[Rin1=rover]=-SA1*(C2[Rin2=rover1]*alpha2[Rin2=rover1] -
    C1*alpha1)/(0.5*(dr1+dr2));
* (7.4) Flow from outermost cell - mixing cell approximation
* (q may be a function of time)
SIDE fluxout2[Rin2=rbent]=switch(C2-Cstar*R2, C2/R2+kpc2*(R2-1)*C2/R2,
    Cstar+kpc2*(R2-1)*Cstar+k2*(C2-Cstar*R2))*q;
* (7.5) Flux from glass into innermost cell of overpack
SIDE fluxglass=-SAglass*(C1[Rin1=rglass]*alpha1[Rin1=rglass]
    -Cglass*alphaglass)/((dr1+rglass)/2);
*
* (Block 8) Evolution equations
*
* (8.1) To in glass decays and is transported out into overpack inner ring
EQUATION d(inventory, t)=-Kdecay*inventory-(fluxglass/ascale);
* (8.2) Innermost ring of overpack receives influx from glass
EQUATION d(C1, t)=- (Kdecay*C1[Rin1<=rglass]+(fluxout1[Rin1=rglass]-fluxglass)
    / V1);
* (8.3) Transport through overpack
EQUATION d(C1, t)=- (Kdecay*C1+(fluxout1-fluxout1[Rin1=Rin1-dr1, Rin1>rglass])
    / V1);
* (8.4) Transport from overpack to bentonite
EQUATION d(C2, t)=- (Kdecay*C2[Rin2>rover1]+
    (fluxout2-fluxout2[Rin2=Rin2-dr2])/V2);
* (8.5) Transport through bentonite (including loss from outermost ring)
EQUATION d(C2, t)=- (Kdecay*C2+
    (fluxout2[Rin2=rover1]-fluxout1[Rin1=rover])/V2);
* (8.6) Amount escaping is equal to loss from outermost ring
EQUATION d(escaped, t)=fluxout2[Rin2=rbent]/ascale;
* (8.7) Amount escaped and surviving is equal to amount escaped less decay
EQUATION d(outside, t)=fluxout2[Rin2=rbent]/ascale-Kdecay*outside;
*
* (Block 9) Initial values
*
INITIAL inventory=invmoles;
INITIAL C1=0;
INITIAL C2=0;
INITIAL escaped=0;
INITIAL outside=0;
INITIAL q=q0+q1*sin(wq*t);

```

```

*
* (Block 10) Weights - chosen to give correct relative scaling for variables
*
WEIGHT C1=1/(C1+Cstar);
WEIGHT C2=1/(C2+Cstar);
WEIGHT inventory=1/(inventory+Cstar*vglass/ascale);
WEIGHT escaped=1/(escaped+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
WEIGHT outside=1/(outside+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
*
* (Block 11) Monitor time and magnitude of peak flux and time of
* "drop off"
*
* The current escaping flux is only considered to be a new peak value if it
* is "significantly" larger than the previous peak,
* i.e. if it exceeds it by more than cutoff times the solver error tolerance.
* The monitor variable drop_time records the time at which the escaping
* flux first drops below a fixed fraction (e.g. 95%) of the peak flux.
* The monitor variable peak_outside records the maximum amount of
* contaminant surviving outside at any time.
*
CONSTANT cutoff=3; * multiple of epsstep required for significance
MONITOR peak_time=switch(peak_flux*(1+cutoff*epsstep)-
    (fluxout2[Rin2=rbent]/ascale),
    t, peak_time);

INITIAL peak_time=t;
MONITOR peak_flux=switch(peak_flux*(1+cutoff*epsstep)-
    (fluxout2[Rin2=rbent]/ascale),
    fluxout2[Rin2=rbent]/ascale, peak_flux);

INITIAL peak_flux=0;
CONSTANT drop_frac=0.95; * fraction of peak flux
MONITOR drop_time=switch(peak_flux*drop_frac-(fluxout2[Rin2=rbent]/ascale),
    t, drop_time);

INITIAL drop_time=t;
MONITOR peak_outside=max(outside, peak_outside);
INITIAL peak_outside=0;
*
* (End of central.spf)
*

```

```

*
* central.spf
*
* SPF file for Task 1.1 central case (IM3246)
*
* This file models the diffusive transport of Technetium Tc-99 through the
* near field of a glass/overpack corrosion product/bentonite system.
*
* JAW 02/02/94
*
* Version 1.7
*
* Modification history:
* Vn Who When What
* 1.0 JAW 02/02/94 First version
* 1.1 JAW 03/02/94 Improve variable scaling and layout of file
* 1.2 JAW 04/02/94 Add colloids and pseudo-colloids
* 1.3 JAW 08/02/94 Further minor layout changes and added "outside"
* 1.4 JAW 10/02/94 Add monitor variable drop_time
* 1.5 JAW 11/02/94 Add monitor variable peak_outside
* 1.6 JAW 14/02/94 Improve inner boundary condition
* 1.7 JAW 15/02/94 Further change to inner boundary condition
* + fixed error in (7.2)
*
* General notes:
* (a) Units.
* The time unit is years. Distances are in metres. Amount of contaminant
* remaining in the glass or escaping from the system is measured in moles
* of Tc-99.
* Concentrations of contaminant in the transported species are measured in
* moles per cubic metre *of water* multiplied by a scaling factor ascale.
* This factor serves to improve the numerical conditioning of the problem:
* without it typical transported concentrations would be many orders of
* magnitude smaller than the remaining inventory.
*
* (b) Geometry.
* The problem is posed in a 1D cylindrical coordinate system, assuming
* cylindrical symmetry. The height of the cylinder is taken as the height
* of the central glass waste canister.
* There are three distinct regions in the model:
* Region Inner Outer
* radius (m) radius (m)
* Glass 0 0.22
* Overpack 0.22 0.52
* Bentonite 0.52 1.50
* The glass is treated as a source term which forms the inner boundary
* condition.
* It is assumed that contaminant from the whole volume of the glass is
* instantaneously in equilibrium with the surrounding water.
* The overpack (strictly, overpack corrosion products) layer and bentonite
* layer are treated as two separate regions with a flux condition at
* their shared boundary.
* The outer boundary of the bentonite is a decompressed zone. This is
* represented by a mixing-cell treatment which forms the outer boundary
* condition.
*
* (Block 1) Control info.
*
LISTFILE; * generate list output
SPARSE; * use sparse matrix solver
ALGERR; * include errors on algebraic variables in timestep monitor
CONSTANT epsstep=1e-5; * value for absolute error tolerance per timestep
EPSSTEP epsstep; * set absolute error tolerance
INITSTEP 1.e-3; * initial timestep
MINTIME 10; * Set output timesteps (in years)
MAXTIME 1e7; * - write out every 10000 years
GEOMSTEP 1.258925412; * from t=0 to t=1e7 years
*ARITHSTEP 1e5; * from t=0 to t=1e7 years
*
* (Block 2) Fundamental constants
*
CONSTANT pi=4*arctan(1);
CONSTANT SecsPerYear=365.2422*24*60*60; * number of seconds in one year
CONSTANT ascale=1e10; * multiplicative factor for all molar concentrations
* in transported species - improves conditioning
CONSTANT thalf=2.1e5; * Half-life of Tc-99 in years
CONSTANT Kdecay=log(2)/thalf; * Equivalent decay constant
CONSTANT Cstar=1e-9*ascale; * Solubility limit of Tc-99 in mol/m3
CONSTANT R1=1; * Retardation factor of Tc-99 in inner (overpack) layer

```

```

CONSTANT R2=280; * Retardation factor of Tc-99 in outer (bentonite) layer
CONSTANT Rglass=1; * Retardation factor of Tc-99 in glass
CONSTANT Dsol=3e-10*SecsPerYear; * diffusion coefficient of Tc solute in
    * both bentonite and overpack
CONSTANT Dpc1=Dsol/10; * diffusion coefficient of Tc pseudocolloid
    * in overpack
CONSTANT Dpc2=Dsol/10; * diffusion coefficient of Tc pseudocolloid
    * in bentonite
CONSTANT Dtc1=Dsol/2; * diffusion coefficient of Tc true colloid
    * in overpack
CONSTANT Dtc2=Dsol/2; * diffusion coefficient of Tc true colloid
    * in bentonite
CONSTANT thetag=0.33; * Porosity of glass
CONSTANT theta1=0.33; * Porosity of overpack
CONSTANT theta2=0.33; * Porosity of bentonite
CONSTANT ZTc=99; * atomic weight of Tc
CONSTANT invmass=1.06e3; * Inventory at calculation start in g
CONSTANT invmoles=invmass/ZTc; * Initial inventory in moles
CONSTANT rglass=0.22; * outer glass radius
CONSTANT rover1=0.52; * actual outer overpack radius
CONSTANT rbent1=1.50; * actual outer bentonite radius
CONSTANT h=1.35; * height of cylinder
CONSTANT vglass=pi*h*rglass*rglass*thetag; * volume of glass
*
* (Block 3) Mixing cell constants
*
CONSTANT q0=0.71*1e-3; * mean flow through mixing cell in m3/year
CONSTANT qfrac=0; * fractional variation in mixing cell flow, used to
    * derive q - central case value is qfrac=0
CONSTANT q1=q0*qfrac; * amplitude of variation in q
CONSTANT Tq=2000; * period of variation in q (in years)
CONSTANT wq=2*pi/Tq; * equivalent angular frequency
SIDE q=q0+q1*sin(wq*t); * mixing cell flow rate (potentially time dependent)
*
* (Block 4) Define geometry.
*
* The problem is posed in 1D cylindrical coordinates. Although
* Spade does not understand about these at present, we can use
* 1D cartesian coordinates because we discretize the
* spatial derivatives ourselves.
*
* The spatial variables Rin1 and Rin2 are chosen to represent the INNER
* radius of each ring. We therefore set the grid spacing so that the
* OUTER radius of the outermost cylinder in each region matches the
* specified dimensions.
*
CONSTANT nover=5; * number of grid points in overpack
CONSTANT nbent=10; * number of grid points in bentonite
CONSTANT rover=rglass+(rover1-rglass)*(nover-1)/nover; * grid limit
    * for overpack
CONSTANT rbent=rover1+(rbent1-rover1)*(nbent-1)/nbent; * grid limit
    * for bentonite

CONSTANT dr1=(rover1-rglass)/nover; * grid spacing in overpack
CONSTANT dr2=(rbent1-rover1)/nbent; * grid spacing in bentonite
SPATIAL Rin1,UNIFORM,rglass,rover,nover; * INNER radius of each ring
SPATIAL Rin2,UNIFORM,rover1,rbent,nbent;
REGION over,CART1D,Rin1; * overpack grid
REGION bent,CART1D,Rin2; * bentonite grid
VECTOR Rout1[over]; * OUTER radius of each ring in overpack
VECTOR Rout2[bent]; * OUTER radius of each ring in bentonite
CONSTANT Rout1=Rin1+dr1;
CONSTANT Rout2=Rin2+dr2;
VECTOR V1[over]; * volume of WATER in each ring in overpack
VECTOR V2[bent]; * volume of WATER in each ring in bentonite
CONSTANT V1=theta1*pi*h*((Rout1*Rout1) - (Rin1*Rin1));
CONSTANT V2=theta2*pi*h*((Rout2*Rout2) - (Rin2*Rin2));
VECTOR SA1[over]; * OUTER surface area of each ring in overpack
VECTOR SA2[bent]; * OUTER surface area of each ring in bentonite
CONSTANT SA1=2*pi*h*Rout1;
CONSTANT SA2=2*pi*h*Rout2;
CONSTANT SAglass=2*pi*h*rglass; * outer surface area of glass
SIDE Cglass=ascale*inventory/vglass; * concentration in glass
*
* (Block 5) Declare remaining vector quantities
*
* the concentration vector within each region
VECTOR C1[over];
VECTOR C2[bent];
* the effective diffusivity

```

```

VECTOR alpha1[over];
VECTOR alpha2[bent];
* the flux across each outer boundary
VECTOR fluxout1[over];
VECTOR fluxout2[bent];
*
* (Block 6) Define alpha. The form of alpha varies depending on the assumed
* distribution of mobile species as a function of concentration.
* For the central case with no colloids or pseudocolloids, k1=k2=kpc1=kpc2=0.
*
CONSTANT k1=0; * fraction of true colloid in overpack
CONSTANT k2=0; * fraction of true colloid in bentonite
CONSTANT kpc1=0; * fraction of pseudocolloid in overpack
CONSTANT kpc2=0.1; * fraction of pseudocolloid in bentonite
* (6.1) alpha in overpack
SIDE alpha1=switch(C1-Cstar*R1,
    Dsol/R1+kpc1*(R1-1)*Dpc1/R1,
    (Dsol*Cstar+kpc1*(R1-1)*Cstar*Dpc1+k1*Dtc1*(C1-Cstar*R1))/C1);
* (6.2) alpha in bentonite
SIDE alpha2=switch(C2-Cstar*R2,
    Dsol/R2+kpc2*(R2-1)*Dpc2/R2,
    (Dsol*Cstar+kpc2*(R2-1)*Cstar*Dpc2+k2*Dtc2*(C2-Cstar*R2))/C2);
* (6.3) alpha in glass
SIDE alphaglass=switch(Cglass-Cstar*Rglass,
    Dsol/Rglass+kpc1*(Rglass-1)*Dpc1/Rglass,
    (Dsol*Cstar+kpc1*(Rglass-1)*Cstar*Dpc1+k1*Dtc1*(Cglass-Cstar*Rglass))/Cglass);
*
* (Block 7) Define fluxout, which gives the flux of Tc across each outer
* face (positive = net outward flux).
* The outer boundary and the boundary between the overpack and bentonite
* have to be handled separately.
*
* (7.1) Flow through overpack
SIDE fluxout1[Rin1<rover]= -SA1*(C1[Rin1=Rin1+dr1]*alpha1[Rin1=Rin1+dr1] -
    C1*alpha1)/dr1;
* (7.2) Flow through bentonite
SIDE fluxout2[Rin2<rbent]= -SA2*(C2[Rin2=Rin2+dr2]*alpha2[Rin2=Rin2+dr2] -
    C2*alpha2)/dr2;
* (7.3) Flow from overpack to bentonite
SIDE fluxout1[Rin1=rover]= -SA1*(C2[Rin2=rover1]*alpha2[Rin2=rover1] -
    C1*alpha1)/(0.5*(dr1+dr2));
* (7.4) Flow from outermost cell - mixing cell approximation
* (q may be a function of time)
SIDE fluxout2[Rin2=rbent]=switch(C2-Cstar*R2, C2/R2+kpc2*(R2-1)*C2/R2,
    Cstar+kpc2*(R2-1)*Cstar+k2*(C2-Cstar*R2))*q;
* (7.5) Flux from glass into innermost cell of overpack
SIDE fluxglass= -SAglass*(C1[Rin1=rglass]*alpha1[Rin1=rglass]
    -Cglass*alphaglass)/((dr1+rglass)/2);
*
* (Block 8) Evolution equations
*
* (8.1) Tc in glass decays and is transported out into overpack inner ring
EQUATION d(inventory, t)= -Kdecay*inventory-(fluxglass/ascale);
* (8.2) Innermost ring of overpack receives influx from glass
EQUATION d(C1, t)= -(Kdecay*C1[Rin1<=rglass]+(fluxout1[Rin1=rglass]-fluxglass)
    / V1);
* (8.3) Transport through overpack
EQUATION d(C1, t)= -(Kdecay*C1+(fluxout1-fluxout1[Rin1=Rin1-dr1, Rin1>rglass])
    / V1);
* (8.4) Transport from overpack to bentonite
EQUATION d(C2, t)= -(Kdecay*C2[Rin2>rover1]+
    (fluxout2-fluxout2[Rin2=Rin2-dr2])/V2);
* (8.5) Transport through bentonite (including loss from outermost ring)
EQUATION d(C2, t)= -(Kdecay*C2+
    (fluxout2[Rin2=rover1]-fluxout1[Rin1=rover])/V2);
* (8.6) Amount escaping is equal to loss from outermost ring
EQUATION d(escaped, t)=fluxout2[Rin2=rbent]/ascale;
* (8.7) Amount escaped and surviving is equal to amount escaped less decay
EQUATION d(outside, t)=fluxout2[Rin2=rbent]/ascale-Kdecay*outside;
*
* (Block 9) Initial values
*
INITIAL inventory=invmoles;
INITIAL C1=0;
INITIAL C2=0;
INITIAL escaped=0;
INITIAL outside=0;
INITIAL q=q0+q1*sin(wq*t);

```

```

*
* (Block 10) Weights - chosen to give correct relative scaling for variables
*
WEIGHT C1=1/(C1+Cstar);
WEIGHT C2=1/(C2+Cstar);
WEIGHT inventory=1/(inventory+Cstar*vglass/ascale);
WEIGHT escaped=1/(escaped+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
WEIGHT outside=1/(outside+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
*
* (Block 11) Monitor time and magnitude of peak flux and time of
* "drop off"
*
* The current escaping flux is only considered to be a new peak value if it
* is "significantly" larger than the previous peak,
* i.e. if it exceeds it by more than cutoff times the solver error tolerance.
* The monitor variable drop_time records the time at which the escaping
* flux first drops below a fixed fraction (e.g. 95%) of the peak flux.
* The monitor variable peak_outside records the maximum amount of
* contaminant surviving outside at any time.
*
CONSTANT cutoff=3; * multiple of epsstep required for significance
MONITOR peak_time=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        t, peak_time);

INITIAL peak_time=t;
MONITOR peak_flux=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        fluxout2[Rin2=rbent]/ascale, peak_flux);

INITIAL peak_flux=0;
CONSTANT drop_frac=0.95; * fraction of peak flux
MONITOR drop_time=switch(peak_flux*drop_frac-(fluxout2[Rin2=rbent]/ascale),
                        t, drop_time);

INITIAL drop_time=t;
MONITOR peak_outside=max(outside, peak_outside);
INITIAL peak_outside=0;
*
* (End of central.spf)
*

```

```

*
* central.spf
*
* SPF file for Task 1.1 central case (IM3246)
*
* This file models the diffusive transport of Technetium Tc-99 through the
* near field of a glass/overpack corrosion product/bentonite system.
*
* JAW 02/02/94
*
* Version 1.7
*
* Modification history:
* Vn Who When What
* 1.0 JAW 02/02/94 First version
* 1.1 JAW 03/02/94 Improve variable scaling and layout of file
* 1.2 JAW 04/02/94 Add colloids and pseudo-colloids
* 1.3 JAW 08/02/94 Further minor layout changes and added "outside"
* 1.4 JAW 10/02/94 Add monitor variable drop_time
* 1.5 JAW 11/02/94 Add monitor variable peak_outside
* 1.6 JAW 14/02/94 Improve inner boundary condition
* 1.7 JAW 15/02/94 Further change to inner boundary condition
*                   + fixed error in (7.2)
*
* General notes:
* (a) Units.
* The time unit is years. Distances are in metres. Amount of contaminant
* remaining in the glass or escaping from the system is measured in moles
* of Tc-99.
* Concentrations of contaminant in the transported species are measured in
* moles per cubic metre *of water* multiplied by a scaling factor ascale.
* This factor serves to improve the numerical conditioning of the problem:
* without it typical transported concentrations would be many orders of
* magnitude smaller than the remaining inventory.
*
* (b) Geometry.
* The problem is posed in a 1D cylindrical coordinate system, assuming
* cylindrical symmetry. The height of the cylinder is taken as the height
* of the central glass waste canister.
* There are three distinct regions in the model:
* Region Inner Outer
* radius (m) radius (m)
* Glass 0 0.22
* Overpack 0.22 0.52
* Bentonite 0.52 1.50
* The glass is treated as a source term which forms the inner boundary
* condition.
* It is assumed that contaminant from the whole volume of the glass is
* instantaneously in equilibrium with the surrounding water.
* The overpack (strictly, overpack corrosion products) layer and bentonite
* layer are treated as two separate regions with a flux condition at
* their shared boundary.
* The outer boundary of the bentonite is a decompressed zone. This is
* represented by a mixing-cell treatment which forms the outer boundary
* condition.
*
* (Block 1) Control info.
*
LISTFILE; * generate list output
SPARSE; * use sparse matrix solver
ALGERR; * include errors on algebraic variables in timestep monitor
CONSTANT epsstep=1e-5; * value for absolute error tolerance per timestep
EPSSTEP epsstep; * set absolute error tolerance
INITSTEP 1.e-3; * initial timestep
MINTIME 10; * Set output timesteps (in years)
MAXTIME 1e7; * - write out every 10000 years
GEOMSTEP 1.258925412; * from t=0 to t=1e7 years
*ARITHSTEP 1e5; * from t=0 to t=1e7 years
*
* (Block 2) Fundamental constants
*
CONSTANT pi=4*arctan(1);
CONSTANT SecsPerYear=365.2422*24*60*60; * number of seconds in one year
CONSTANT ascale=1e10; * multiplicative factor for all molar concentrations
* in transported species - improves conditioning
CONSTANT thalf=2.1e5; * Half-life of Tc-99 in years
CONSTANT Kdecay=log(2)/thalf; * Equivalent decay constant
CONSTANT R1=1; * Retardation factor of Tc-99 in inner (overpack) layer
CONSTANT R2=2.8; * Retardation factor of Tc-99 in outer (bentonite) layer

```



```

CONSTANT Rglass=1; * Retardation factor of Tc-99 in glass
CONSTANT Dsol=3e-10*SecsPerYear; * diffusion coefficient of Tc solute in
      * both bentonite and overpack
CONSTANT Dpc1=Dsol/2; * diffusion coefficient of Tc pseudocolloid
      * in overpack
CONSTANT Dpc2=Dsol/2; * diffusion coefficient of Tc pseudocolloid
      * in bentonite
CONSTANT Dtc1=Dsol/2; * diffusion coefficient of Tc true colloid
      * in overpack
CONSTANT Dtc2=Dsol/2; * diffusion coefficient of Tc true colloid
      * in bentonite
CONSTANT thetag=0.33; * Porosity of glass
CONSTANT theta1=0.33; * Porosity of overpack
CONSTANT theta2=0.33; * Porosity of bentonite
CONSTANT ZTc=99; * atomic weight of Tc
CONSTANT invmass=1.06e3; * Inventory at calculation start in g
CONSTANT invmoles=invmass/ZTc; * Initial inventory in moles
CONSTANT rglass=0.22; * outer glass radius
CONSTANT rover1=0.52; * actual outer overpack radius
CONSTANT rbent1=1.50; * actual outer bentonite radius
CONSTANT h=1.35; * height of cylinder
CONSTANT vglass=pi*h*rglass*rglass*thetag; * volume of glass
*
* (Block 3) Mixing cell constants
*
CONSTANT q=0.71*1e-3; * mean flow through mixing cell in m3/year
CONSTANT qyear=1e5; * half-width of gaussian peak in years
CONSTANT Tq=100000; * period of variation in q (in years)
CONSTANT qwidth=qyear/Tq; * fractional width of gaussian peak
CONSTANT wq=2*pi/Tq; * equivalent angular frequency
EQUATION qpar=(1/pi)*((pi/2)+arcsin(sin(wq*t-(pi/2))))); * varies linearly from 0 to 1 and
INITIAL qpar=0;
WEIGHT qpar=1/1000;
      * back every Tq years
SIDE qadd=exp(-qpar*qpar/(qwidth*qwidth));
INITIAL qadd=1;
EQUATION Cstar=1e-9*ascale*(1+1e8*qadd); * Solubility limit of Tc-99 in mol/m3
INITIAL Cstar=1e-9*ascale*1e8;
WEIGHT Cstar=1/(100*Cstar);
*
* (Block 4) Define geometry.
*
* The problem is posed in 1D cylindrical coordinates. Although
* Spade does not understand about these at present, we can use
* 1D cartesian coordinates because we discretize the
* spatial derivatives ourselves.
*
* The spatial variables Rin1 and Rin2 are chosen to represent the INNER
* radius of each ring. We therefore set the grid spacing so that the
* OUTER radius of the outermost cylinder in each region matches the
* specified dimensions.
*
CONSTANT nover=5; * number of grid points in overpack
CONSTANT nbent=10; * number of grid points in bentonite
CONSTANT rover=rglass+(rover1-rglass)*(nover-1)/nover; * grid limit
      * for overpack
CONSTANT rbent=rover1+(rbent1-rover1)*(nbent-1)/nbent; * grid limit
      * for bentonite
CONSTANT dr1=(rover1-rglass)/nover; * grid spacing in overpack
CONSTANT dr2=(rbent1-rover1)/nbent; * grid spacing in bentonite
SPATIAL Rin1.UNIFORM,rglass,rover,nover; * INNER radius of each ring
SPATIAL Rin2.UNIFORM,rover1,rbent,nbent;
REGION over,CART1D,Rin1; * overpack grid
REGION bent,CART1D,Rin2; * bentonite grid
VECTOR Rout1[over]; * OUTER radius of each ring in overpack
VECTOR Rout2[bent]; * OUTER radius of each ring in bentonite
CONSTANT Rout1=Rin1+dr1;
CONSTANT Rout2=Rin2+dr2;
VECTOR V1[over]; * volume of WATER in each ring in overpack
VECTOR V2[bent]; * volume of WATER in each ring in bentonite
CONSTANT V1=theta1*pi*h*((Rout1*Rout1)-(Rin1*Rin1));
CONSTANT V2=theta2*pi*h*((Rout2*Rout2)-(Rin2*Rin2));
VECTOR SA1[over]; * OUTER surface area of each ring in overpack
VECTOR SA2[bent]; * OUTER surface area of each ring in bentonite
CONSTANT SA1=2*pi*h*Rout1;
CONSTANT SA2=2*pi*h*Rout2;
CONSTANT SAglass=2*pi*h*rglass; * outer surface area of glass
SIDE Cglass=ascale*inventory/vglass; * concentration in glass
*

```

```

* (Block 5) Declare remaining vector quantities
*
* the concentration vector within each region
VECTOR C1[over];
VECTOR C2[bent];
* the effective diffusivity
VECTOR alpha1[over];
VECTOR alpha2[bent];
* the flux across each outer boundary
VECTOR fluxout1[over];
VECTOR fluxout2[bent];
*
* (Block 6) Define alpha. The form of alpha varies depending on the assumed
* distribution of mobile species as a function of concentration.
* For the central case with no colloids or pseudocolloids, k1=k2=kpc1=kpc2=0.
*
CONSTANT k1=0; * fraction of true colloid in overpack
CONSTANT k2=0; * fraction of true colloid in bentonite
CONSTANT kpc1=0; * fraction of pseudocolloid in overpack
CONSTANT kpc2=0; * fraction of pseudocolloid in bentonite
* (6.1) alpha in overpack
SIDE alpha1=switch(C1-Cstar*R1,
    Dsol/R1+kpc1*(R1-1)*C1*Dpc1/R1,
    (Dsol*Cstar+kpc1*(R1-1)*Cstar*Dpc1+k1*Dtc1*(C1-Cstar*R1))/C1);
* (6.2) alpha in bentonite
SIDE alpha2=switch(C2-Cstar*R2,
    Dsol/R2+kpc2*(R2-1)*C2*Dpc2/R2,
    (Dsol*Cstar+kpc2*(R2-1)*Cstar*Dpc2+k2*Dtc2*(C2-Cstar*R2))/C2);
* (6.3) alpha in glass
SIDE alphaglass=switch(Cglass-Cstar*Rglass,
    Dsol/Rglass+kpc1*(Rglass-1)*Cglass*Dpc1/Rglass,
    (Dsol*Cstar+kpc1*(Rglass-1)*Cstar*Dpc1+k1*Dtc1*(Cglass-Cstar*Rglass))/Cglass);
*
* (Block 7) Define fluxout, which gives the flux of Tc across each outer
* face (positive = net outward flux).
* The outer boundary and the boundary between the overpack and bentonite
* have to be handled separately.
*
* (7.1) Flow through overpack
SIDE fluxout1[Rin1<rover]=--SA1*(C1[Rin1=Rin1+dr1]*alpha1[Rin1=Rin1+dr1] -
    C1*alpha1)/dr1;
* (7.2) Flow through bentonite
SIDE fluxout2[Rin2<rbent]=--SA2*(C2[Rin2=Rin2+dr2]*alpha2[Rin2=Rin2+dr2] -
    C2*alpha2)/dr2;
* (7.3) Flow from overpack to bentonite
SIDE fluxout1[Rin1=rover]=--SA1*(C2[Rin2=rover1]*alpha2[Rin2=rover1] -
    C1*alpha1)/(0.5*(dr1+dr2));
* (7.4) Flow from outermost cell - mixing cell approximation
* (q may be a function of time)
SIDE fluxout2[Rin2=rbent]=switch(C2-Cstar*R2, C2/R2+kpc2*(R2-1)*C2/R2,
    Cstar+kpc2*(R2-1)*Cstar+k2*(C2-Cstar*R2))*q;
* (7.5) Flux from glass into innermost cell of overpack
SIDE fluxglass=-SAglass*(C1[Rin1=rglass]*alpha1[Rin1=rglass]
    -Cglass*alphaglass)/((dr1+rglass)/2);
*
* (Block 8) Evolution equations
*
* (8.1) Tc in glass decays and is transported out into overpack inner ring
EQUATION d(inventory, t)=-Kdecay*inventory-(fluxglass/ascale);
* (8.2) Innermost ring of overpack receives influx from glass
EQUATION d(C1, t)=-Kdecay*C1[Rin1<=rglass]+(fluxout1[Rin1=rglass]-fluxglass)
    / V1);
* (8.3) Transport through overpack
EQUATION d(C1, t)=-Kdecay*C1+(fluxout1-fluxout1[Rin1=Rin1-dr1, Rin1>rglass])
    / V1);
* (8.4) Transport from overpack to bentonite
EQUATION d(C2, t)=-Kdecay*C2[Rin2>rover1]+
    (fluxout2-fluxout2[Rin2=Rin2-dr2])/V2);
* (8.5) Transport through bentonite (including loss from outermost ring)
EQUATION d(C2, t)=-Kdecay*C2+
    (fluxout2[Rin2=rover1]-fluxout1[Rin1=rover])/V2);
* (8.6) Amount escaping is equal to loss from outermost ring
EQUATION d(escaped, t)=fluxout2[Rin2=rbent]/ascale;
* (8.7) Amount escaped and surviving is equal to amount escaped less decay
EQUATION d(outside, t)=fluxout2[Rin2=rbent]/ascale-Kdecay*outside;
*
* (Block 9) Initial values
*

```

```

INITIAL inventory=invmoles;
INITIAL C1=0;
INITIAL C2=0;
INITIAL escaped=0;
INITIAL outside=0;
*
* (Block 10) Weights - chosen to give correct relative scaling for variables
*
WEIGHT C1=1/(C1+Cstar);
WEIGHT C2=1/(C2+Cstar);
WEIGHT inventory=1/(inventory+Cstar*vglass/ascale);
WEIGHT escaped=1/(escaped+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
WEIGHT outside=1/(outside+C2[Rin2=rbent]/ascale+Cstar*vglass/ascale);
*
* (Block 11) Monitor time and magnitude of peak flux and time of
* "drop off"
*
* The current escaping flux is only considered to be a new peak value if it
* is "significantly" larger than the previous peak,
* i.e. if it exceeds it by more than cutoff times the solver error tolerance.
* The monitor variable drop_time records the time at which the escaping
* flux first drops below a fixed fraction (e.g. 95%) of the peak flux.
* The monitor variable peak_outside records the maximum amount of
* contaminant surviving outside at any time.
*
CONSTANT cutoff=3; * multiple of epsstep required for significance
MONITOR peak_time=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        t, peak_time);

INITIAL peak_time=t;
MONITOR peak_flux=switch(peak_flux*(1+cutoff*epsstep)-
                        (fluxout2[Rin2=rbent]/ascale),
                        fluxout2[Rin2=rbent]/ascale, peak_flux);

INITIAL peak_flux=0;
CONSTANT drop_frac=0.95; * fraction of peak flux
MONITOR drop_time=switch(peak_flux*drop_frac-(fluxout2[Rin2=rbent]/ascale),
                        t, drop_time);

INITIAL drop_time=t;
MONITOR peak_outside=max(outside, peak_outside);
INITIAL peak_outside=0;
*
* (End of central.spf)
*

```