
結 言

統合的電力系統解析支援システムの構築に関する研究

平成11年3月

歌 谷 昌 弘

緒 言

電力系統は、現代の極めて重要な支持基盤である電気エネルギーを経済的かつ高い信頼性で供給する責務を有している。そのため電力系統の運用計画に対し、基本的各種電力系統解析法を駆使する必要がある。

従来、系統解析は個々の解析プログラムを個別に適用することにより行われてきたが、必要なデータや操作方法の違いから、その利用は取り扱いに熟練した技術者に限定されがちである。ユーザを熟練者と同じレベルにまで教育するには、時間とコストが必要になる。したがって、解析プログラムの操作に詳しくないユーザが熟練したユーザと同レベルの作業を実行できる解析支援環境が現在求められている。

本論文は、上述の解析支援システムを構成する上で重要な鍵となる (i) 既存解析プログラムの有効利用、(ii) データベースの共通化、(iii) マンマシンインタフェースについての研究結果をまとめたものである。

第1章においては、研究の背景ならびに研究の目的を明確化するために、現状における解析プログラムの操作、および入力データの管理に関する本研究の概要を述べている。

第2章では、まず1章で指摘した問題点を解決する解析支援システムを提案している。提案する解析支援システムは、オブジェクト指向アプローチに基づいて分析・設計・開発することで、他の研究者らによって提案されているものより保守・拡張性の点で特に優れたものとなっている。その他にも、統合的解析プログラムの操作を効果的に支援するために、解析目的に適したグラフィカルユーザインタフェース (GUI) を用意し、解析データの入力、プログラムの実行、解析結果表示までを一貫して支援する機能を用意した。特に解析結果を系統図上に表示したり、グラフ上にプロットする機能は、ユーザが解析結果を理解する際に大いに役立つものである。さらに、既存の解析プログラムの有効利用を図る際に問題となる入出力データフォーマット、操作方法の差異に対処するため、解析プログラム毎にアプリケーションプログラムインタフェース (API) を設け、適切な入力データファイル作成と実行結果の取り込み処理を行わせ、既存の解析プログラムを修正することなくシステムに組み込めるようにする。本章の最後では、これらの機能を実際に組み込んだシステムの構築例を紹介する。

第3章では、前章のシステムのもつ支援機能だけでは十分に対処しきれていない解析作業を支援するための機能とシステムの管理のために必要と思われる機能を加えたシステムを提案している。十分に対処しきれていなかったのは、運用データをパラメータにとり解析を繰り返す作業である。前章

のシステムでは、一連の解析結果を一画面にプロットできないため、断面毎に解析結果を表示することになり、ユーザは表示画面を手動で切り替え、解析結果を理解してゆかねばならない。一方システム管理のために必要があると判断したのは、提案システムへの既存解析プログラムの追加登録を支援する機能である。解析プログラムを新たにシステムに追加するには、専用のAPIをあらたにオブジェクト指向プログラミングしなければならないが、オブジェクト指向言語は、まだFortranやC言語ほど広く使用されていないため、一般のユーザには解析プログラムの追加が困難だからである。本章はこれらの2つの点をそれぞれ一連の解析結果を一括表示するための機能と解析プログラムの登録支援インタフェースを追加することで解決している。

第4章では、接続状況を示すデータから直接手書きで描いていたため多大な労力を費やしていた電力系統の単線結線図を自動作成するための一手法を提案している。この手法は、VLSIの配線設計で使用されている混雑度の概念に基づく最短経路探索法を利用することにより、ブランチの交差数ができるだけ少ない系統図の自動作成を行うものである。さらに、得られた系統図においてブランチの経路をさらに見やすくするためにラバーバンドとスポークを利用するアルゴリズムを用いて経路の整形を行っている。本システムではこれらの手法を用いて、ユーザがノードの配置位置を指定するだけで、自動的に線路の交差数の少ない系統図の作成を行うことができる。さらにノード配置に不都合がある場合、画面上で対象ノードを移動することができ、移動ノードに接続する線路についてのみ配線アルゴリズムを再実行することにより、よりわかりやすい系統図が得られる。インタラクティブな手法として開発されているものとして、CRT上にノードや送電線を逐一配線していく方法は実用化されているが、これは簡単なモデル系統を描画することはできても、数十ノード以上の系統に対しては極めて繁雑であり、実用的とは言い難い。したがって本章の方法は大きな前進と考えられる。

第5章では、2章、3章で提案してきた解析支援システムをもとに電力系統計画立案作業に適した新たな解析支援システムを提案する。ここで電力系統計画立案作業とは、年々増加する電力需要に対応してゆくために立案される電力設備増強の方針を立案することである。方針を決定する際には、様々な増設案に対して様々な運用条件を設定し、解析プログラムを使用して解析を行い、妥当な増設案を選択してゆく。提案システムは、この増設案の比較検討作業を支援するためのシステムである。このシステムは、電力系統の表現に適したオブジェクト指向データベース管理システム(OODBMS: Object-Oriented Data Base Management System) およびGUIを基本とし、適切に計画担当者を支援することを目指しており、以下のような特徴がある。第一は、オブジェクト指向の概念に基づき解析プログラムの入力データを効率よく管理できることである。具体的には、各増設プランのデータとして、元系統のデータとの差分のみを格納することにより、同一データを大量に格納す

る無駄を排除した。また、元システムの構成変更と各増設プランのデータとは無関係になるため、データベースの一貫性を容易に保持することができる。第二の特徴は、一連の解析プログラムを一つのアプリケーションとして自由に登録できる点である。これにより、計画者自身の希望する解析過程を自動化することができるため、操作性が極めて良くなり、プログラム操作上のミスもほぼ根絶できる。

第6章においては、各章の主要な結果を要約し、今後の課題を整理して結論とした。

目次

緒 言	11
第1章 序 論	1
1.1 研究の背景	1
1.2 本研究の目的	2
1.3 本論文の研究概要	3
1.3.1 統合的電力系統解析システムの提案	3
1.3.2 解析支援環境の高度化の概要	3
1.3.3 電力系統図編集作業の自動化の概要	4
1.3.4 電力系統計画立案支援環境の概要	4
第2章 統合的電力系統解析支援システムの提案	5
2.1 電力系統解析環境	5
2.1.1 解析環境の問題点	5
2.1.2 問題解決のアプローチ	5
2.2 解析支援システムの提案	6
2.2.1 システムの構成	6
2.2.2 解析支援インタフェース	7
2.2.3 APIによる既存解析プログラムの有効利用	8
2.2.4 電力系統データベース	9
2.3 潮流計算プログラムの操作例	11
2.3.1 システムモデルの登録	12
2.3.2 運用データの入力	13
2.3.3 解析の実行	14
2.3.4 結果表示	14
2.4 結言	14
第3章 解析支援機能の高度化	16
3.1 解析支援環境の再検討	16
3.1.1 高度化に必要な機能	16
3.2 解析支援環境の高度化	17
3.2.1 解析支援環境の構成	17
3.2.2 解析プログラムの登録支援	19
3.2.3 連続時間シミュレーション	21
3.3 提案環境の操作例	22
3.3.1 解析プログラムの登録	22
3.3.2 連続時間シミュレーション結果の表示	24
3.4 結言	25
第4章 電力系統図編集作業の自動化	26
4.1 最短経路探索法による系統図の作成	26

4.1.1	データ入力	27
4.1.2	混雑コスト計算	27
4.1.3	最短経路の探索	28
4.2	配線後の系統図の整形	29
4.2.1	配線領域の3角形分割	29
4.2.2	系統図のラバーバンド	30
4.2.3	スポークの挿入	31
4.3	実行結果	32
4.3.1	系統図の表示	32
4.3.2	系統図の変更	32
4.4	結言	33
第5章	電力系統計画立案支援環境の提案	34
5.1	電力系統計画	35
5.1.1	計画立案作業の概要	35
5.1.2	系統計画立案作業に必要な機能	35
5.2	計画立案支援環境の提案	36
5.2.1	構築方針	36
5.2.2	支援環境の構成	36
5.2.3	支援機能の説明	37
5.3	電力系統データベースのバージョン管理	38
5.3.1	バージョン管理機能	38
5.3.2	Vmodelの複合オブジェクトへの拡張	38
5.3.3	複合オブジェクトと拡張Vmodelによる電力系統データ表現	38
5.3.4	スナップショットによるバージョンデータ群の管理	39
5.4	電力系統計画支援システムの実行例	40
5.5	結言	42
第6章	結 論	44
	謝 辞	47
	参考文献	48
	付 録	51
A.	オブジェクト指向の概要	51
B.	MVCモデル	64
C.	電力系統モデルのオブジェクト指向分析	66
D.	電力系統解析支援インタフェースのオブジェクト指向分析	70
	研究業績	77

第1章 序 論

本章では、本研究の背景と目的について述べ、研究の位置づけを明らかにする。

1.1 研究の背景

電力系統は、現代社会の最も重要な支持基盤である電気エネルギーを経済的にかつ高い信頼性で供給する責務を有している。そのため電力系統の計画・運用などのあらゆる業務において、基本的各種電力系統解析法を駆使する必要がある。

従来、系統解析は個々の解析プログラムを個別に適用することにより行われてきたが、必要なデータや操作方法の違いから、その利用は取り扱いに熟練した技術者に限定されがちである。ユーザを熟練者と同じレベルにまで教育するには、時間とコストが必要になる。したがって、解析プログラムの操作に詳しくないユーザが熟練したユーザと同レベルの作業を実行できる解析環境が望まれている。

このような解析支援システムを構成する上で重要な鍵となるのは (i) 既存解析プログラムの有効利用、(ii) データベースの共通化、(iii) マンマシンインタフェースであり、本研究はこれらの項目についての考察を加え、これを解決するための研究を行っている。

まず、提案するシステムの位置づけを明確にするため、解析プログラムの使用に関する一般的な問題、ならびにこれまでに発表された類似の研究の問題点について概括しておく。

(1) 既存解析プログラムの使用上の問題点 現在の系統解析プログラムは長年に亘る開発努力のたまものであり、当然それぞれの開発当時の計算機環境に制約されていて、最近の高性能な計算機環境を十分に活用することができない。例えば、データベースによる統一的データ管理がなされていないため類似の複数のファイルの作成が要求されること、データ形式の不統一、解析結果の表示が視覚的でないことなどが挙げられる。しかし、これまでプログラムの開発と保守に多大の時間、費用、労力が費やされているため、既存の解析プログラムの有効利用が望ましい。

既存のプログラムを新しい環境に適合させるべく大幅な修正を加えると、新たなバグ発生の原因となる。したがって、修正量を極小化するか、もし可能ならばそのまま利用すべきである。

(2) 解析用データ管理の問題点 既存の解析プログラムの多くは、それぞれ入力されるデータ形式が異なり、しかもそのデータはファイルで管理されることから、ユーザは同一の系統に対し複数の入力データファイルを作成しなければならない。したがって、一つのデータ項目を修正することは、関連するすべてのファイル中該当項目データの修正を意味することになり、膨大な労力や修正ミスが起り得る。

この問題の解決法には、データベースに一括して系統データを蓄え、データベース管理システム(DBMS)により情報を管理(登録・検索・修正)する方法が提案されてきた^(1,2)。現在、電力会社の

設備データなどにリレーショナルデータベース (RDB) が最も多く使用されている。この RDB は、母線や線路などの機器毎に表を設け情報を管理している。例えば線路の表では、各行に個々の線路を割り当て、列項目 (キー項目) にインピーダンス、線路容量、接続先などの属性を割り当てることにより情報を管理している。一般に、RDB はデータの重複を最小とすることを目的として表を分割する (正規化) が、これは変電所、開閉所など物理的意味のある分類とは全く無関係に行われるため、物理的実体との対応が見つからないだけでなく、問合せ手続きが複雑となり応答時間が非常に遅い。そのため、系統情報の問合せに対し十分な応答速度が得られるよう、正規化していない系統情報管理専用の RDB を構築すると、同一情報を重複して格納することになりメモリの浪費、データ更新の無謬性の欠落という問題が発生する。また系統解析においては電力系統の構成要素の接続関係の把握を基礎とするが、表の各行に格納されたキー項目の値を頼りにこれを求めていく RDB には不向きである。また最近の GUI の発達で、系統図を画面上に表示して、属性値の入力や接続関係の修正を行うことが一般的になっているが、GUI はオブジェクト指向を基本にしている。したがって、画面上の系統図を媒介としてデータを管理する場合、RDB モデルと表示モデルの間のギャップを埋める手続きに負担がかかることになる。

(3) 解析支援環境の不統一 文献 (3, 4) では、EWS をネットワークで接続し、分散処理によって処理能力を向上させる環境⁽³⁾、データベースと解析プログラムとの間に設けたインタフェースを介して、複数のプログラムが 1 つのデータベースを共有する環境⁽⁴⁾を構築し、解析操作の支援はグラフィカルユーザインタフェース (GUI) で行っている。しかし、別な環境下で開発された解析プログラムを利用するには、これらのシステムの使用しているデータベース、GUI、分散処理環境などに対応できるよう修正しなければならず、プログラムミスなど予想外の問題で必要以上に労力と時間を必要とする事態が予想される。系統解析は、データの準備、プログラムの操作、出力結果の解釈の 3 段階を経て行われるが、複数の解析プログラムを統一的に支援するシステムは実現されていない。また、解析の実行を支援する方法もプログラム毎に当然異なっており、ユーザにとって使いやすいものではない。

1.2 本研究の目的

上述の問題を解決するためにユーザにとって使いやすい解析支援システムの構築を目指して一連の研究を行ってきた^(5,7)。すなわち解析プログラムの操作を効果的に支援するために、解析目的に適した GUI を用意し、解析データの入力、プログラムの実行、解析結果表示までを一貫して支援する。既存の解析プログラムの有効利用を図る際に問題となる入出力データフォーマット、操作方法の差異に対処するため、解析プログラム毎にアプリケーションプログラムインタフェース (API) を設け、適切な入力データ作成と実行結果の処理を行わせる。電力系統データの管理には、物理的実体の忠実なモデル化、データと手続の一体化による問合せ手続きの簡略化といった RDB にない特徴を持つ

オブジェクト指向データベース (Object-Oriented Data Base : OODB) を採用する。そしてそのデータベース管理には、系統作成から属性管理までを支援する描画機能を備えた GUI を設け、ユーザが容易に電力系統データ管理を行えるようにする。また本システムを通信処理に優れた UNIX ワークステーション上に構築することで、将来の分散処理環境やオープンシステム化へ対応することも考慮している。

1.3 本論文の研究概要

本節では各章の主な内容を含めて本論文の概要を述べる。

1.3.1 統合的電力系統解析システムの提案

第2章では、1章で述べた解析プログラムの操作、入力用データの管理に関する問題点を解決するための具体的なアプローチと提案する解析支援システムの基本構成を紹介する。提案する解析支援システムは、オブジェクト指向アプローチに基づいて分析・設計・開発することで、他の研究者らによって提案されているものより保守・拡張性の点で特に優れたものとなっている。基本構成の紹介のあとには解析支援システムがオブジェクト指向の概念を利用してどのように構築されているかを紹介する。本章の最後では、解析支援システムの構築例と潮流計算プログラムの実行例も紹介する。

1.3.2 解析支援環境の高度化の概要

第3章では、2章で提案した解析支援システムでは対処できない解析作業について検討を加え、その解決方法を提案している。2章で提案したシステムは、統合的解析プログラムの操作を効果的に支援するために、解析目的に適した GUI を用意し、解析データの入力、プログラムの実行、解析結果表示までを一貫して支援しており、特に解析結果を系統図上に表示したり、グラフ上にプロットする機能は、ユーザが解析結果を理解する際に大いに役立っている。ただし、その表示機能とは運用データのある値に設定したときの解析結果を表示するようにしたものである。運用データをパラメータとして解析を行う場合、これまでのシステムでは、解析結果を一画面にプロットできないため、断面毎に解析結果を表示することになる。そしてユーザは、表示画面を手動で切り替え、解析結果を理解してゆかねばならない。

またこれまで我々の提案してきた解析支援システムでは、既存の解析プログラムの有効利用を図る際に問題となる入出力データフォーマット、操作方法の差異に対処するため、解析プログラム毎にアプリケーションプログラムインタフェース (API) を設け、適切な入力データファイル作成と実行結果の取り込み処理を行わせている。しかし、解析プログラムを新たにシステムに追加するには、専用の API をあらたにオブジェクト指向プログラミングを行わなければならないため、一般のユーザには解析プログラムの追加が困難である。

本章はこれらの2つの問題点をそれぞれ、解析結果の動的側面を表示するための機能と解析プロ

グラムの登録支援インタフェースを追加することで解決している。本章の最後には構築した支援システムの実行例を示し、意図した結果が得られていることを紹介する。

1.3.3 電力系統図編集作業の自動化の概要

第4章では、接続状況を示すデータから直接手書きで描いていたため多大な労力を費やしていた電力系統の単線結線図を自動作成するための一手法を提案している。その手法とは、VLSIの配線設計で使用されている混雑度の概念に基づく最短経路探索法を利用することにより、ブランチの交差数ができるだけ少ない系統図の自動作成を行うものである。さらに、得られた系統図においてブランチの経路をさらに見やすくするためにラバーバンドとスポークを利用するアルゴリズム⁽²⁷⁾を用いて経路の整形を行っている。本システムではこれらの手法を用いて、ユーザがノードの配置位置を指定するだけで、自動的に線路の交差数の少ない系統図の作成を行うことができる。さらにノード配置に不都合がある場合、画面上で対象ノードを移動することができ、移動ノードに接続する線路についてのみ配線アルゴリズムを再実行することにより、よりわかりやすい系統図が得られる。

本論文で提案する方法で生成される系統図は常に電力系統解析に適したものとばかりではないが、系統図作成の自動化・高速化には非常に有用である。また、原理的にはノード数およびブランチ数に制限がないため、かなり大規模な系統にも適応できるシステムが構築できる。

1.3.4 電力系統計画立案支援環境の概要

第5章では、2章、3章で提案してきた解析支援システムをもとに電力系統計画立案作業に適した新たな解析支援システムを提案する。ここで電力系統計画立案作業とは、近年の着実な電力需要の増加に対して適切な電力供給設備の拡張方針を立案してゆく作業のことである。この作業の実行に際しては、様々な電力設備増設プランに対して多様な運用条件や故障条件を設定しながら、電力系統解析プログラムで解析を行った結果から、妥当とみなされる増設プランを選択してゆき、最終的な設備拡張方針を決定してゆく。提案システムは、電力系統の表現に適したオブジェクト指向データベース管理システム(Object-oriented database management system : OODBMS) および GUI を基本とし、適切に計画担当者の解析作業を支援することを目指しており、以下のような特徴がある。第一は、オブジェクト指向の概念に基づき解析プログラムの入力データを効率よく管理できることである。具体的には、各増設プランのデータとして、元系統のデータとの差分のみを格納することにより、同一データを大量に格納する無駄を排除した。また、元系統の構成変更と各増設プランのデータとは無関係になるため、データベースの一貫性を容易に保持することができる。第二の特徴は、一連の解析プログラムを一つのアプリケーションとして自由に登録できる点である。これにより、計画者自身の希望する解析過程を自動化することができるため、操作性が極めて良くなり、プログラム操作上のミスもほぼ根絶できる。これらの機能の有効性を提案システムの実行例で紹介する。

第2章 統合的電力系統解析支援システムの提案

本章では、1章で指摘した現状の解析プログラムに関する問題点、すなわち操作および入力用データの管理に関する問題点を解決するためのアプローチと提案システムの基本構成について説明する。提案する解析支援システムは、オブジェクト指向アプローチ（付録A参照）に基づいて分析・設計・開発することで、他の研究者らによって提案されているものより保守・拡張性の点で優れたものとなっている。次いで解析支援システムがオブジェクト指向の概念を利用してどのように構築されているかということ、およびオブジェクト指向に基づくシステム開発によって得られた効果について記述する。最後に、解析支援システムの構築例と潮流計算プログラムの実行例を紹介し、その有効性を確認する。

2.1 電力系統解析環境

2.1.1 解析環境の問題点

計算機の処理能力と表示装置の発達で、複数プログラムの同時処理機能や高度なグラフィカルユーザインタフェース (GUI) が利用できるようになり、こうした機能を応用した解析環境の構築が試みられているが、既存のプログラムと新しい計算機環境は決して相性の良いものでない。統一的なプラットフォーム上で解析支援システムを構築する上での問題点は既に第1章で指摘した通りであり、主として以下の3項目である。

- (a) 既存解析プログラム使用上の問題点
- (b) 解析用データ管理の問題点
- (c) 解析支援環境の不統一

さらに、系統解析のあらゆる局面で電力系統図を使用することが解析作業の効率化と解析結果の理解を助けるために有効であるが、現在実用に足る系統図を用いたインタラクティブなインタフェースがないことも指摘しておく。

2.1.2 問題解決のアプローチ

(1) 既存解析プログラムの有効利用 既存の解析プログラムと新しい計算機環境との間のギャップを埋めるために、アプリケーションプログラムインタフェース (API) を解析プログラム毎に設け、入力ファイルの作成、プログラムの実行支援、解析結果の効果的提示などの機能を持たせる。これにより、既存のプログラムの修正量を極小化できるので、修正に伴うバグの発生を抑制すると共に、ユーザの負担を軽減することができる。またAPIは、オブジェクト指向言語によるプログラミングで共通部分を汎化し、新たなAPIの追加にともなう負担増加の軽減を図る。

(2) 効率的なデータ管理 入手可能な商用のデータベースで、前節で述べた問題を解決できるものとしてオブジェクト指向データベース (Object-Oriented Data Base : OODB) がある。オブジェ

クトとは「情報（データ）とその処理手続き（メソッド）を一つの単位として組込んだもの」であり、OODBの主な特徴を列挙すると以下のとおりである。

- (a) オブジェクトは自身の所属するクラスに加え、その上位クラス、下位クラスを定義することができ、階層関係にある概念や事物の表現に適している。
- (b) オブジェクトはそれぞれの識別子により参照されるので、オブジェクト間の相互関係の記述が容易である。
- (c) オブジェクト内部の情報は、一定のアクセス手続きを介してのみ外部に公開されるため、オブジェクト内部の修正がシステム全体に波及することはない。

すなわち、特徴(a)は電力系統の階層的構造を表現するのに適しており、また(b)により系統の接続状態を簡単な手続きで調べることができる。したがって、本研究では、OODB管理システムで電力系統解析に必要な情報を一括管理する方法を採用する。こうして、本システムでは、実際の系統の構成機器やその階層構造を詳細に分析し、オブジェクトで電力系統をモデル化してある。ゆえに各プログラムからのデータ要求に対応可能である。この電力系統モデルは、オブジェクト指向プログラミングによって生成され、オブジェクトを永続的に管理するOODBに保存される。したがってOODBによる情報管理は、オブジェクト指向プログラミングとの相性がよい。

(3) ユーザ支援機能 系統解析は、系統の地理的な配置や接続関係を考慮しながら解析を進める必要があり、これには画面上の系統図を媒介として解析操作を支援する方法が最も有効である。本研究では、ウインドーシステムの持つ優れた視覚化の能力を利用して、高度でかつ使い易いプログラムの操作支援環境を構築する。具体的には、ウインドーシステムを系統図の表示、解析データの管理、解析プログラムの実行、出力データの視覚的表示などのインタフェースの構築に利用する。また、ユーザ支援用インタフェースもAPIと同じくオブジェクト指向プログラミングによってプログラミングの負担軽減を図る。

2.2 解析支援システムの提案

2.2.1 システムの構成

提案する解析支援システムの構成を図2.1に示す。本システムは、解析支援インタフェース、データベース管理支援インタフェース、解析プログラムとの仲介機能を持つAPI群、電力系統データベース（主データベース）、解析モデルや結果の一時保存用データベース（補助データベース）という5つのモジュールから構成される。現在、支援の対象としているのは、潮流計算、故障計算、最適潮流計算（OPF）、過渡安定度解析などの基本的な解析プログラムである。

解析支援インタフェースは、解析プログラムへのユーザからのデータの直接入力、プログラムの実行、解析結果の表示をウインドーシステムを通して行なえる環境を提供する。データベース管理支援インタフェースは、解析プログラムに必要なデータをユーザが容易に管理できるようにする役

割を担っているが、前節で説明したように電力系統データベースを直接管理しているのは、オブジェクト指向データベース管理システム（OODBMS）のVersant（Versant Object Technology 社製）である。このOODBMSが、本システムで生成したオブジェクトの電力系統データベースへの格納と抽出を管理している。本研究では、ウインドープログラミングの容易さ、オブジェクト指向DBとの相性の良さから、解析インターフェース、API、電力得系統データベースをオブジェクト指向言語"Smalltalk-80"（以下Smalltalkと略記）で構築した。グラフィックユーザインターフェースのプログラミングには、Smalltalkが標準で提供しているMVCモデル^(9,付録B)と呼ばれるインターフェース用のモデルを使用した。次節では、解析環境を構成する主要なモジュールの役割について説明する。

2.2.2 解析支援インターフェース

(1) 解析操作支援インターフェース 解析操作支援インターフェースはユーザからの支援要求に対して、対話的に応答していくモジュールである。ユーザはグラフィカルユーザインターフェース(GUI)を通して、希望する解析プログラムの選択、対象系統の選択、運用データの設定変更等を容易に行えるよう支援し、解析プログラムにより得られる結果を視覚的に表示することにより理解を助ける環境を提供する。運用データの設定は、データファイルを使用する方法の他に、スクリーン上の系統図から任意の機器を選択し、直接データを入力する機能も用意している。解析結果の表示は、解析プログラムが出力した数値データをリスト表示する以外に、画面上の系統図に重ねて表示すると共に、直感的に結果を把握できるようにメータやスライダーバーで図式的に表示する機能も付加した。

(2) データベース管理支援インターフェース このモジュールは、ウインドー上に表示された表示モデルを介してデータベースに格納されている機器への属性値の入力や修正といったデータの管

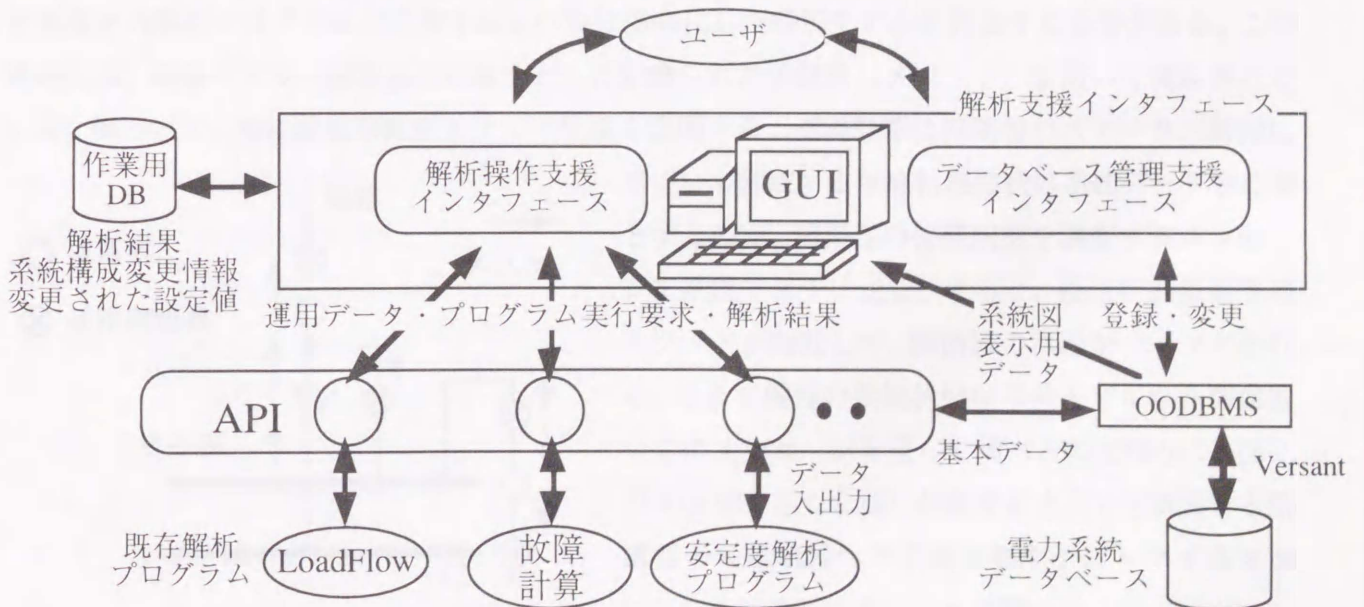


図2.1 解析支援システムの基本構成

理操作を行う。ユーザは常に系統図で編集情報を確認しながら作業を行うことができるので、データベースに既に格納されている電力系統の修正や小さなテスト系統の作成に適している。

一方、新たに大規模な電力系統をGUIを用いて手作業で登録するのは能率的でないので、本ツールは既存の解析プログラムのデータ入力ファイルを基に、自動的に電力系統モデルを生成し、データを登録する機能も備えている。

また、潮流計算と最適潮流計算においては、デフォルトまたはユーザ指定の電圧上下制限約を違反したノードの色を反転して表示する機能も付している。安定度解析のような電力系統の動的な特性を解析するプログラムには、解析結果を動画で表示する機能を用意した。

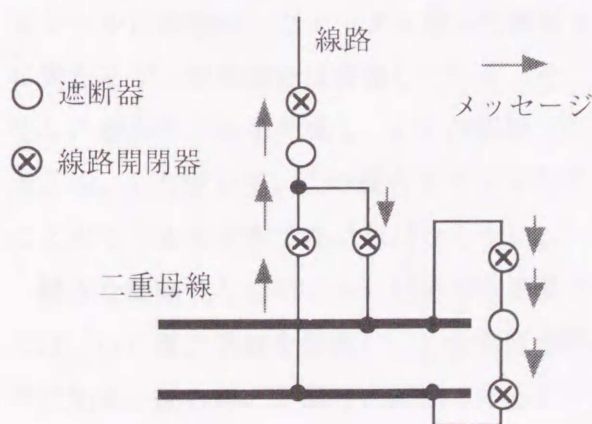
2.2.3 APIによる既存解析プログラムの有効利用

(1) APIの機能

APIは、電力系統データベースへのアクセス、入力データファイルの作成、プログラムの実行、解析結果のインタフェースへの転送など解析の実行に必要な機能をサポートするものであり、各解析プログラム毎に一つ用意される。以下でその機能の詳細を説明する。

入力ファイル作成： APIは、データ入力ファイルの仕様情報(フォーマット)に基づいて、データベース上の系統モデルから得られるネットワーク情報(機器の接続情報)、および機器に固有な情報(機器の物理的要因から決定される値)と補助データベースから得られる別のプログラムの解析結果や運用データ(ユーザが繰り返し解析を実行する場合にパラメータのように扱われるデータ)から解析プログラムへデータを入力するファイルを作成する。

解析モデルの準備： 主データベース上の系統データは系統状態を忠実に表すため、解析には直接必要のない機器の状態まで詳細に表現している。しかし、系統解析を行う際にはこの詳細なデータを各々の解析プログラムが要求するレベルに簡略化した解析モデルを作成する必要がある。この簡略化は、機器モデル(開閉器や母線など)に定義された手続き(メソッド)を用いて実現されている。例として、潮流計算用解析モデルの生成を説明する。潮流計算は開閉器のステータス情報に



基づいて作成する単線結線図状の系統データを必要とするので、開閉器の接続状態を調査するメッセージを系統モデルに送る。すると、接続状態調査用のメソッドが起動して、開閉器の状態がチェックされる。ここで機器の接続状態は系統モデルから開閉器全てにメッセージを送って調べるのではなく、図2.2の矢印で示した様に母線を起点として隣接する開閉器から開閉器へと開閉状態をチェックするメッセージを伝達してゆくことで調べる。

この方法は、全ての処理がメッセージパッシング

図2.2 接続状態の調査

で行なわれるオブジェクト指向プログラミングでは最も効率がよく自然な方法である。他の解析プログラムに対しても同様にして所望の系統データを作成することができる。

出力データ処理: 解析プログラムはそれぞれ固有なフォーマットで解析結果を出力するので、出力データの仕様情報を参照して結果を読み込み、インタフェース側に送り返す。

(2) APIの設計方針

解析プログラム毎に用意するAPIの最も重要な機能はデータ入出力ファイルの処理である。APIをなるべく統一的な仕様の下に、最小の労力で作成するために以下の設計方針を採用する。

本システムへ新しく追加する解析プログラムに対するAPIを設計する場合、ソースプログラムが公開されており、プログラムを改造してファイルを介さず直接リンクすることができるケースとオブジェクトコードのみが利用でき、改造をまったく行うことができないケースに分けて考える必要がある。実行速度の面から見ると前者が望ましいが、必ずしもソースコードが公開されるとは限らないの上、改造にかかるコストも無視することができない。一方、後者の場合については、設計上必要な入出力ファイルの仕様を明らかにしておく必要がある。

まず、ソースプログラムを対象とするときは、プログラムに対して構文解析を行い、入出力文と対応するフォーマット文から、入出力ファイルにおけるデータ項目とフォーマットおよびそれらの並びを読み取り、APIに必要なファイル処理部分のプログラムの自動生成を行う。現在システムに組み込まれている解析プログラムは、広島大学電力工学研究室で一貫した方針の下に開発されたプログラムであるので、容易に実現されている。しかし、他所で開発されたプログラムを同様に処理するためには、より高度な構文解析アルゴリズムが必要とされる。

次に、オブジェクトコードのみの使用権が得られた場合は、当然のことながら、入出力ファイルの完全な仕様をもとに手作業でAPIを作成することになる。

2.2.4 電力系統データベース

電力系統データベースの概要: 解析プログラムは、電力系統をそのプログラムで必要とされるレベルに簡略化したデータを使って解析を行う。それらのデータは、解析手法(プログラム)毎に異なるが、相当部分は重複している。そこで本研究では、電力系統をできるかぎり詳細にモデル化した統合モデルを作成し、全ての解析プログラムはこの統合モデルを利用して解析を行うことを考える。したがって、この統合モデルは全ての解析プログラムが必要としているデータを作成することができるモデルでなければならない。

様々な簡略化したモデルに対応できる統合モデルを管理するデータベースに要求される機能としては、(1) 電力系統を忠実に、しかも直感的にわかりやすく表現できること、(2) 電力系統のデータに加え、振る舞いも同時に記述できること、(3) 保守および拡張が容易であること、が挙げられる。こうした要求を満たすことのできるデータベースとして最も適しているのは、実世界の物や事象(データ)、動作、振る舞いを忠実にモデル化できるオブジェクト指向データベースである。オブ

ジェクト指向データベースとは、オブジェクト指向言語において過渡的なものとして取り扱われているオブジェクトを二次記憶装置上に永続化し、それらを管理運用するデータベースとしての機能を備えたシステムである。

電力系統データベースのデータモデル： 電力系統は、数多くの部品の組み合わせによって構成されていることから、複合オブジェクト（オブジェクトによる入れ子構造で、集約とも呼ばれる。）を利用すると直感的にわかりやすく表現することができる。本研究では、Rumbaughらの提唱するオブジェクト指向システム分析（OMT）⁽¹⁰⁾の手法に基づいて電力系統に存在するオブジェクトを抽出し、複合オブジェクトによって忠実に電力系統モデルを構築する。OMTに基づいて電力系統のオブジェクト指向分析を行った結果は付録Cに示し、ここではその概要を紹介する。

電力系統を詳細に記述するために、図2.3に示すように電力系統を3階層に分割した。3つの階層はそれぞれ、電力系統、電力系統を構成する施設、施設を構成する機器に対応しており、以下にそれらの内容と機能を説明する。

まず最上位階層は、当然のことながら電力系統というオブジェクトが抽出される。電力系統は、発電所、変電所などの施設と送電線路によって構成されているので、次の階層には、電気所（変電所、発電所、開閉所）、送電線路などのオブジェクトが抽出される。最下位の階層からは、母線、遮断器、変圧器、発電機などの機器オブジェクトが抽出される。各機器オブジェクトには、機器に固有な情報が格納され、オブジェクト内で定義された手続き（メソッド）によってその機器に関する情報を

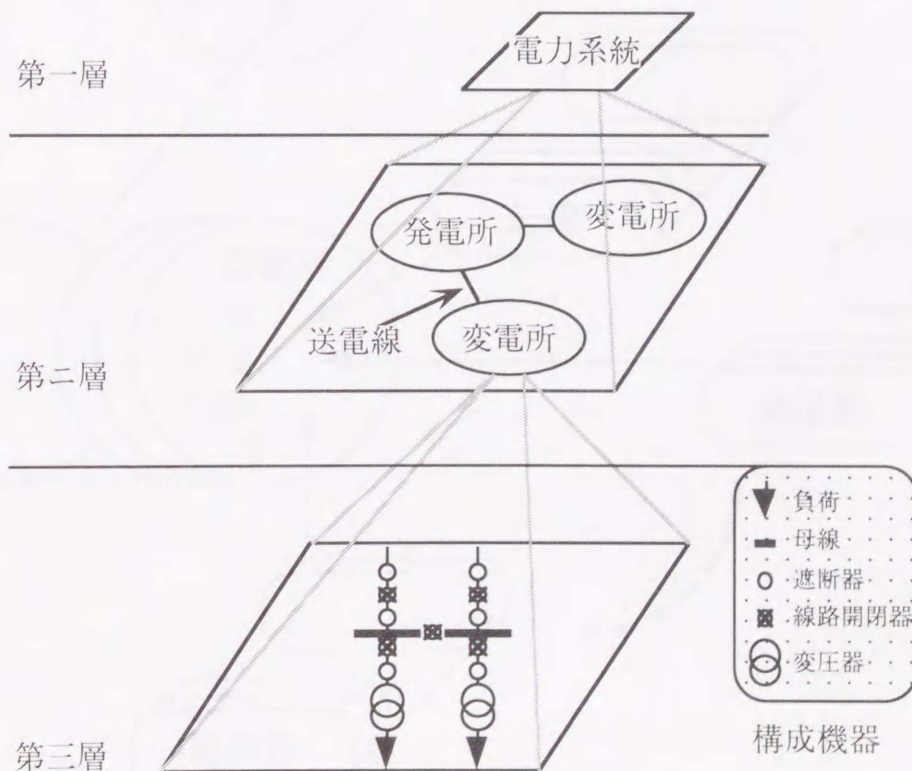


図2.3 電力系統の階層構造

返したり、固有の処理を行う。前節で述べた施設内の機器の接続状態の調査はこのメソッドを起動することによって実現される。

図2.4は図2.3で示した各層から抽出したオブジェクトを組み合わせる簡単な電力系統モデルを構築する例を示している。図2.4のように電力系統オブジェクトは、その属性値として送電線路、電気所オブジェクト等を持つ。そして電気所オブジェクトは、その属性値として母線、変圧器、遮断器といったオブジェクトを持つ。このモデルの構成方法は、現実の電力系統の構成を忠実に表している。

2.3 潮流計算プログラムの操作例

潮流計算を例にとって解析プログラムの実行手順を説明する。解析は、系統モデルの登録、運用データの入力、解析実行、結果表示の順に進む。図2.5、2.6は、IEEE118母線系統に対して潮流計算を適用した例である。

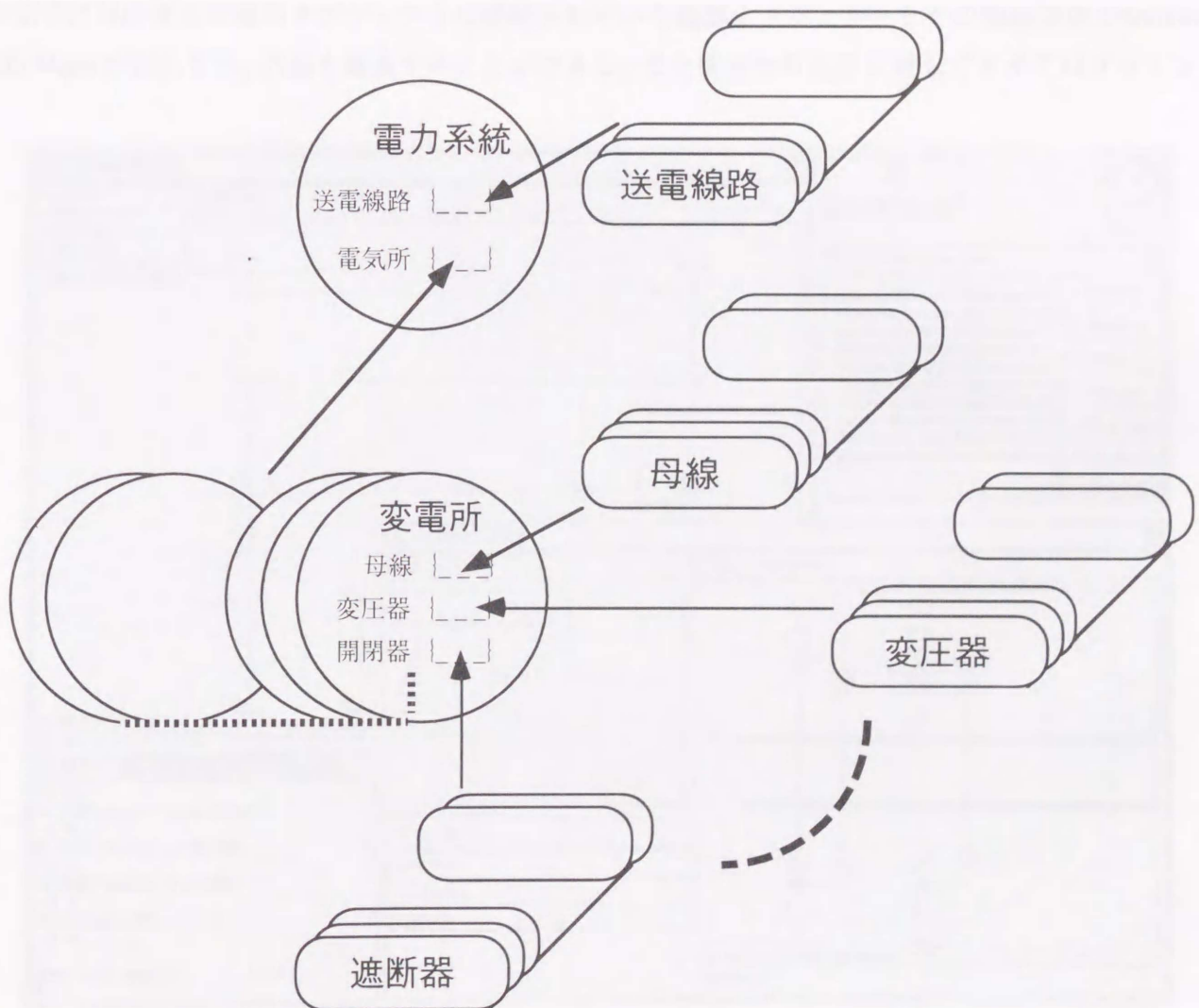


図2.4 電力系統モデルの構成

2.3.1 系統モデルの登録

まず、解析に先立って必要な系統モデルを作成し、データベースに登録しておく必要がある。モデルの作成方法としては、2.2.2節で述べた既存のデータファイル上のモデルを一括してオブジェクト指向モデルに変換するものと、電力用の図記号（アイコン）を使用した系統図編集でオブジェクト指向モデルを構築するものがある。前者は大規模な系統の登録を支援するための機能で、これは、入力フォーマットやデータの並びが既知のデータファイルから読み込んだデータを、対応するオブジェクトにマッピングすることで実現されている。図2.5のSystem Editorは、電気所をノード、線路をブランチとして表現した単線結線図を編集するツールである。Station Editorは、先のSystem Editorで入力した電気所を構成するオブジェクトを編集するためのツールで、Model Managerは、編集対象の系統の選択や削除を行うためのツールである。これらツールによる系統作成では、まず電気所間の接続関係のデータを編集した後、電気所内の機器の接続関係を編集してゆく。図2.5のSystem View（単線結線図）から編集対象の変電所（ここでは母線2）をマウスで選択すると、その記号に対応する変電所オブジェクトに格納されている機器オブジェクトとその接続関係がStation Edit Viewに表示され、内部を編集することができる。また画面から記号に対してダイアログウイン

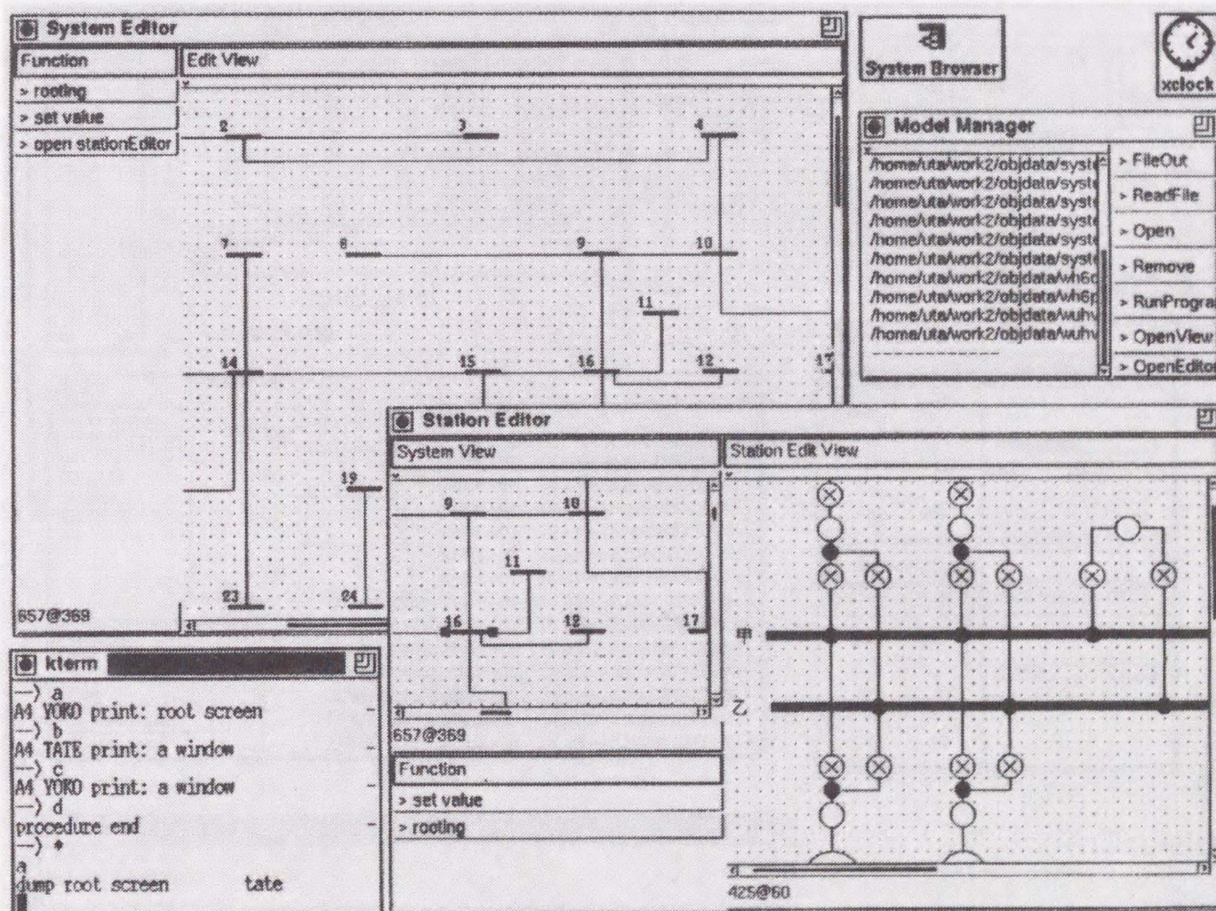


図2.5 データベース管理

ドで値を設定すると、対応するオブジェクトにデータが入力されるようになっている。図2.5の例は、"Bus"とラベル付けされたダイアログウインドーを通して、母線番号や公称電圧等を登録している様子である。

2.3.2 運用データの入力

運用データはユーザインタフェースを通じて直接入力できるのであるが、ユーザの入力ミスの削減を支援することを目的として、対話型の入力インタフェースツールを用意した。図2.6のEdit Viewは、運用データをインタラクティブに設定するために用意されている。Edit Viewの下にあるPFとラベル付けされたウインドーは、解析対象の系統選択、解析実行、解析結果のリスト表示などを行うツールである。まず、ツールを起動して解析対象の電力系統を選択すると図2.6右上のような系統図が表示される。この系統図は、運用データの設定を視覚的に支援するためのインタフェースとしての役割を持ち、マウスで選択した任意のノードに対し、図中央左のようなダイアログウインドーを通してデータが設定できる。さらに、図中央左上の"Line Table"とラベル付けされたテーブルによるデータの一括入力インタフェースを用い、多くのデータをダイアログで入力すると画面が見難くなる問題を解消した。

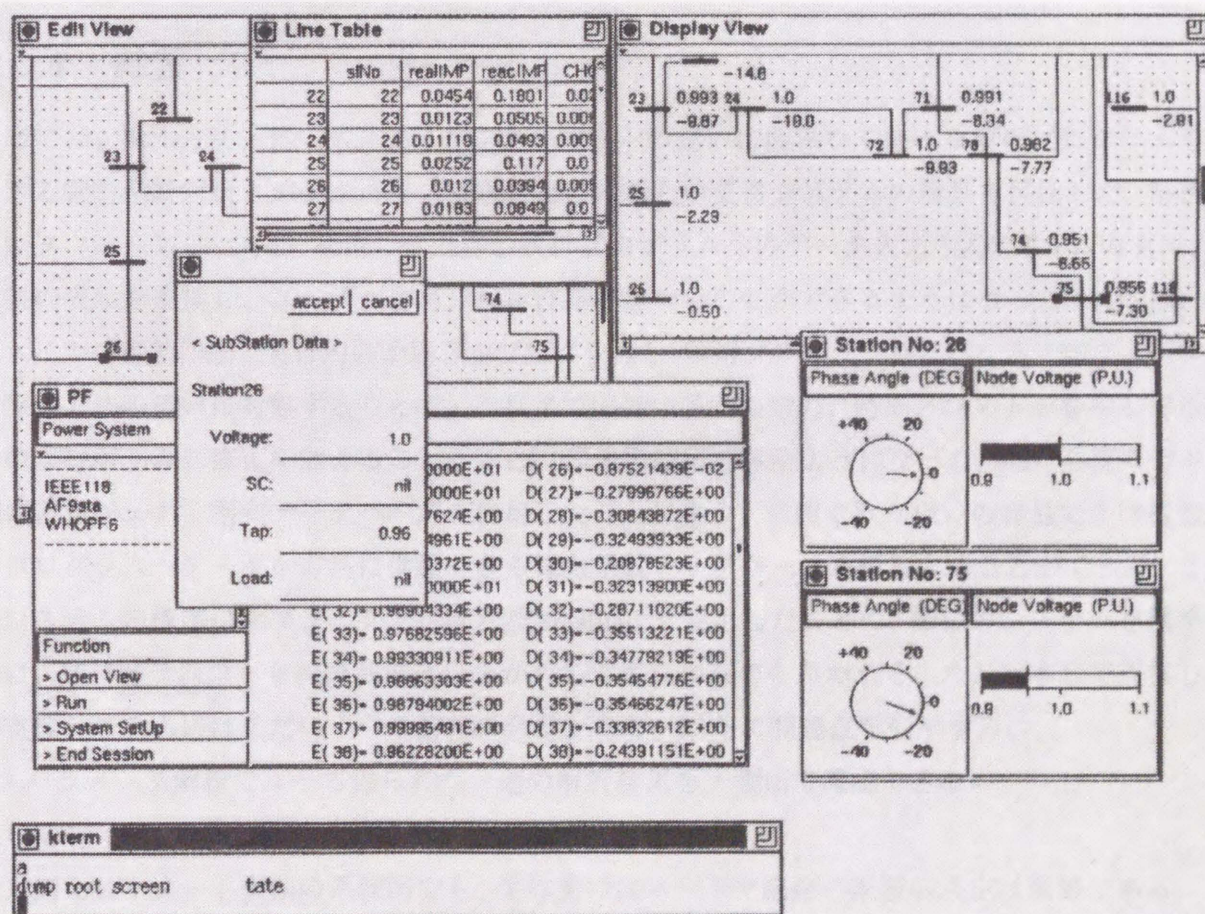


図2.6 潮流計算実行例

2.3.3 解析の実行

運用データを設定した後に解析に移る。解析に伴うファイルの作成とプログラムの実行は、APIにおいて定義されている手続きにより自動的に処理されるため、ユーザは、図2.6のインタフェース群の中にある解析実行ボタンをマウスでクリックするだけでよい。IEEE118母線系統に対して潮流計算を適用した際の処理時間は、データベースアクセスを含めたファイル作成に約3[sec]、解析に約1[sec]、結果の読み込みと表示に約5[sec]である。使用計算機はSun SPARCstation IPX (24.5MIPS)である。

2.3.4 結果表示

解析が終了すると、APIは解析結果を出力ファイルから読み込み、インタフェース側からの要求に応じて結果を返す準備を行う。図2.6のDisplay Viewは、解析結果を表示するウインドーで、各ノードの側にノード電圧と位相角を表示する機能の他に、電圧上下限制約に違反したノードを赤で表示する機能を備えている。Display Viewの下にある2つのウインドーは、解析結果のノード電圧と位相角をそれぞれスライダーバーとメーターで表示するウインドーである。解析結果をグラフィカルに表示することで、ユーザの理解支援を行っている。解析を続ける場合は、ふたたび運用データをインタフェースで設定し、繰り返すことができる。

2.4 結言

本章では、操作方法、データファイルフォーマットの違いに関係なく様々な解析プログラムを使用できる電力系統解析とデータベース管理のための新しい支援システムの構築を目指した。解析支援インタフェースは、GUIを利用して既存の解析プログラムへのデータ入力、プログラムの実行、解析結果の表示を支援し、ユーザが容易に解析作業を進めることができるようになった。APIは、入力ファイルの自動作成と実行用の手続きをプログラムし、解析プログラム毎に異なる入力フォーマットと操作方法の違いに対処することで、これまで開発に多くの労力、時間、コストを費やしてきた既存のプログラムに修正を加えることなくGUI環境で使用できるようになった。電力系統モデルの管理にオブジェクト指向データベースを使用したことにより、管理や取り扱いの容易でかつ複数の解析プログラムへデータを容易に供給できる電力系統データベースを構築することができた。また提案システムの構築にオブジェクト指向言語Smalltalkを使用したことで、類似のシステムを構築する際に、プログラムコードの再利用というかたちで役立つものと思われる。ただし本章で提案した解析支援システムにはまだいくつか改善の余地がある。以下に問題点を列挙する。

- (1) パラメータ解析によって得られた一連の解析結果を一画面で確認できない。
- (2) APIのプログラミングは一般のユーザには困難である。
- (3) 例え数十ノード規模の系統図でも、手作業ではノードや線路の配置の決定は困難である。
- (4) 設備拡張計画立案作業では、大部分のデータが元系統と同じ拡張案を多数作成し、管理しな

ければならない。

提案システムは、基本的な解析操作支援を目的として設計しており、解析プログラムを実行する毎に解析結果を表示するようになっている。従って負荷容量や発電機出力などをパラメータとして解析を繰り返した際に、(1)で挙げたように一連の結果を一画面に表示することができない。また提案システムに組み込んだデータ入出力文構文解析によってAPIを作成する機能は動作保証が限定されたものであり、解析プログラムのソースコードも公開されない場合も多いので、データ入出力ファイルの仕様情報からAPIをプログラムすることが重要となる。しかし、APIのプログラミングに使用したオブジェクト指向言語"Smalltalk"は一般にはあまり使用されていないため、(2)のようにユーザが直接APIをプログラミングすることは困難である。3章では、この(1)と(2)を解決する方法について検討する。

一方、本章で系統図を介して系統データの管理や解析結果の表示を行うことがデータ管理の操作性の向上と解析結果の理解に役立つことを強調した。しかし提案システムは、系統図をユーザが手作業で編集画面上で作成してゆくか、既存の系統図からノードや線路などの位置データをファイルで入力して系統図を登録することしかできない。しかも手作業で系統図を作成する場合、(3)で挙げたように、ノード数が増えれば増える程時間をかけてもなかなか解析結果等を理解しやすくデータ管理等で扱いやすい系統図を得ることは困難となる。4章では、最小限の労力で妥当な系統図を自動的に作成する方法を検討する。

最後に提案システムは、電力系統計画立案作業のようにオリジナルの電力系統を部分修正して多数の設備拡張案を作成し、設備拡張案の解析結果を比較する作業にも使用可能である。ところが、提案システムの電力系統データベースは、一つの完全な電力系統モデルを管理するように設計されており、上記の設備拡張案も一つの完全な電力系統モデルでなければならない。すると設備拡張案はオリジナルの電力系統を部分修正したものなので、上記(4)のように大部分同一の情報を含むことになり、設備拡張案のデータの一貫性保持にたいへんな労力が必要となる。5章では、解析支援システムを電力系統計画立案作業にも使用できるように、設備拡張案を効率良く電力系統データベースで管理する方法を検討する。

第3章 解析支援機能の高度化

第2章においてユーザにとって使いやすい電力系統解析支援システムの提案を行った⁽¹⁾。このシステムは、統合的解析プログラムの操作を効果的に支援するために、解析目的に適したグラフィカルユーザインタフェース (GUI) を用意し、解析データの入力、プログラムの実行、解析結果表示までを一貫して支援しており、特に解析結果を系統図上に表示したり、グラフ上にプロットする機能が、ユーザが解析結果を理解するさいに大いに役立っている。ただし、その表示機能とは運用データのある値に設定したときの解析結果を表示するようにしたものである。運用データをパラメータにとり解析を行う場合、これまでのシステムでは、解析結果を一画面にプロットできないため、断面毎に解析結果を表示することになる。そしてユーザは、表示画面を手動で切り替え、解析結果を理解してゆかねばならない。運用データをパラメータにとり解析を行った場合の結果を一括表示できるようにすることで、さらに提案システムを使いやすいものにすることができる。

また2章で提案した解析支援システムでは、既存の解析プログラムの有効利用を図る際に問題となる入出力データフォーマット、操作方法の差異に対処するため、解析プログラム毎にアプリケーションプログラムインタフェース (API) を設け、適切な入力データファイル作成と実行結果の取り込み処理を行わせている。しかし、解析プログラムを新たにシステムに追加するには、専用のAPIをあらたにオブジェクト指向プログラミングしなければならないため、一般のユーザには解析プログラムの追加が困難である。インタラクティブに解析プログラムの入出力データに関する情報を入力するだけで必要なAPIを自動作成し、解析プログラムの支援システムの追加を支援するインタフェースを用意すれば、一般のユーザにも解析プログラムの追加作業が行えるようになる。

本章は上述のような機能を追加した解析支援システムの概要と構築例を紹介するものである。3.1節では、2章で紹介した解析支援システムを高度化するために必要となる機能について論じる。3.2節では、今回提案する解析支援システムの概要を示し、問題解決のために新たに追加した解析結果の動的側面を表示するための機能と解析プログラムの登録支援インタフェースに言及する。3.3節では、構築した支援システムの実行例を紹介する。

3.1 解析支援環境の再検討

3.1.1 高度化に必要な機能

第2章で述べた解析支援システムは、基礎的な解析プログラムの操作支援を目的として設計されたものであり、比較的完成度は高いものの、解析を多数回繰り返し、それらの出力データをお互いに比較するなどといった複雑な解析操作を支援することは困難である。次のような解析環境の改善を行うことでより支援システムを使いやすいものとすることができる。

(1) 解析プログラムの登録支援 これまでに提案した解析支援システムでは、解析プログラム

毎にAPIを設け、解析支援システムとのデータ交換を一括管理させている。つまりAPIにおいてデータ交換を一括管理することで、解析プログラムを修正することなく解析プログラムの追加を可能にしている。このAPIのプログラムには、オブジェクト指向言語を使用し、その言語の持つコードの再利用性の良さを利用してプログラムの負担軽減が図られている。

このAPIのプログラミングが容易であれば、一般のユーザが必要に応じて解析プログラムを支援システムに登録してゆくことができる。しかし、APIのプログラミングに使用されているオブジェクト指向言語は現在それほど一般的に使用されていないため、プログラマ以外がAPIを作成し解析プログラムに登録してゆくことは困難である。インタラクティブにAPIを作成し、支援システムへの解析プログラムの登録を行うことができれば、登録作業を一般のユーザが行い、プログラマの負担を軽減することができる。

(2) 解析結果表示の改善 これまでに提案した解析支援システムは、基礎的な解析支援を目的に開発されており、機器の物理定数や運用データのある値に固定して解析を行った結果を画面上にグラフ表示する機能は備えていたが、機器の物理定数や運用データをパラメータとして解析を実行し、その結果をユーザにわかりやすく表示する機能は備えていない。機器の物理定数や運用データをパラメータとして解析を実行した場合、解析結果は各ステップ毎に表示することしかできないため、ユーザは、画面を切り替えて変化の様子を把握してゆかねばならない。そのため、ユーザは結果が連続的に変化する様子が直接確認できず、結果の理解に時間がかかってしまう。連続的に変化するデータは、一画面上に一括して表示する方が理解が容易である。

3.2 解析支援環境の高度化

3.2.1 解析支援環境の構成

本章で提案する解析支援システムの構成は、2章で提案したシステムを改良したものである。図3.1の解析支援インタフェースは、運用データの設定、解析プログラムの実行、結果表示を、データベース管理支援インタフェースは、データベースへ格納されるシステムモデルの作成・保守をGUIを使ってインタラクティブに行うようになっている。この解析支援インタフェースに新たに2つの機能を追加する。一つは、新たな解析プログラムの登録をユーザが容易に行えるようにするための機能である。この機能は、既存の解析プログラム、インタフェース、電力系統データベース間のデータ交換を一括管理するAPIをGUIでインタラクティブに作成することを目指している。二つ目は、連続的に負荷などの運用データを変えて解析を実行する機能とその解析結果を表示する機能である。電力系統データベースでは、系統構成、線路インピーダンスや変圧器のタップ比といった機器のデータ、系統図の座標データを管理している。補助データベースは、解析中の機器データや運用データの設定・変更によって電力系統データベースに保存されている共有データを変更してしまわないために、系統構成の修正や運用データの設定を一時的に保存する。この補助データベースは、解析結

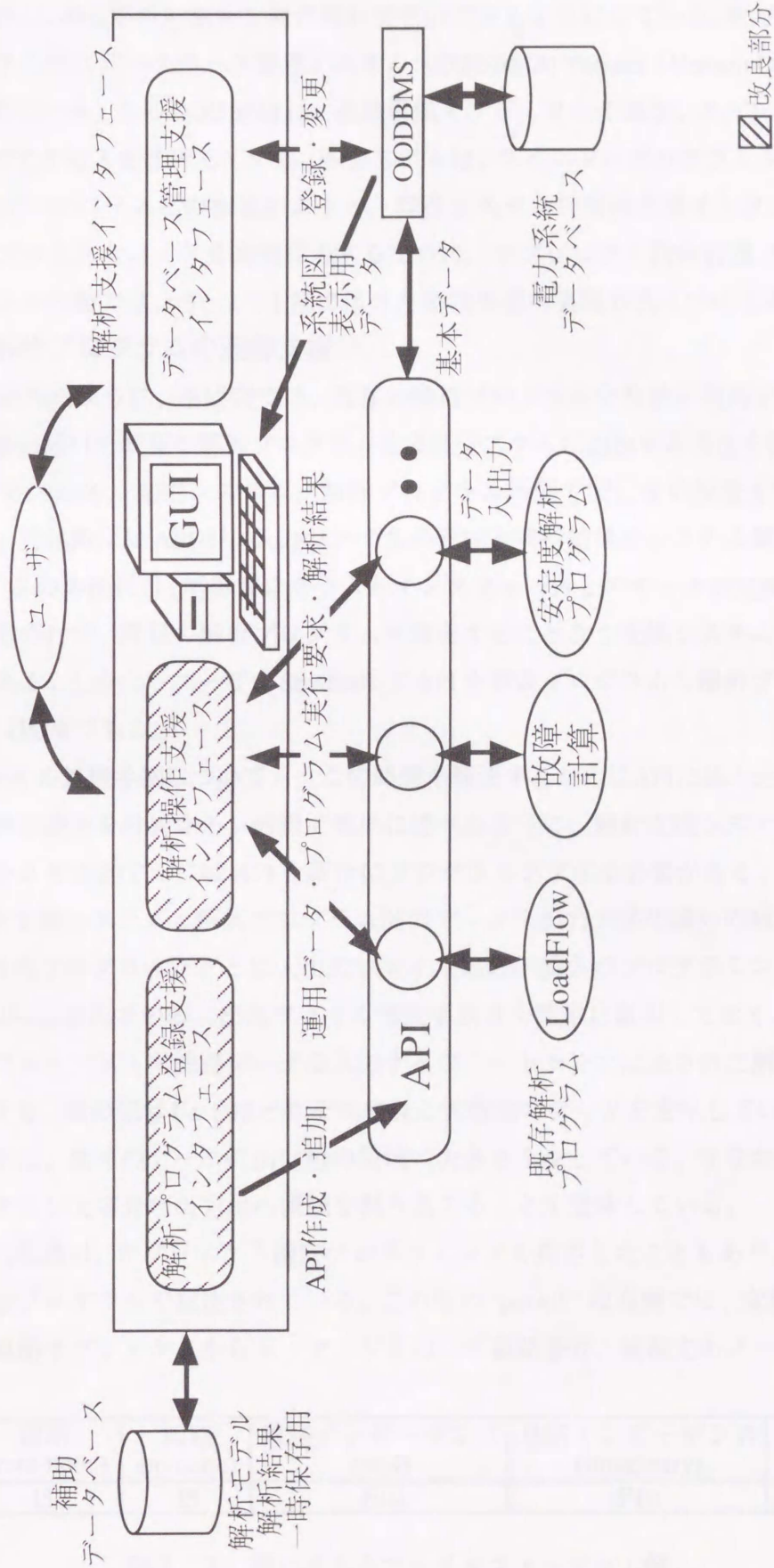


図3.1 改良された解析支援システムの構成

果も一時的に保存し、ある解析の結果を別の解析で利用できるようにしている。電力システムデータベースは、オブジェクト指向データベース管理システム (OODBMS) Versant (Versant Object Technology 社製) で管理されている。このOODBMSは、系統構成オブジェクトの追加、オブジェクトの属性値の修正、属性へのアクセスを管理している。本システムは、ウインドープログラミングの容易さ、オブジェクト指向データベースとの相性のよさと、既存システムの解析支援インタフェース、API、データベースのプログラムコードを有効利用するために、オブジェクト指向言語 (Smalltalk) で構築されている。以下の節では、3.1.1節で述べた要求事項の実現方法について述べる。

3.2.2 解析プログラムの登録支援⁽¹²⁾

3.1.1節で述べたように、本研究では、既存の解析プログラムを有効に利用するためにAPIを解析プログラム毎に設けて既存の解析プログラムを支援システムに追加する方法を採用している。このAPIは、データベース、支援システム、解析プログラムの中でデータの授受を管理するインタフェースである。具体的にはAPIが、入力ファイルの作成と解析結果をシステム側へ読み込みを管理するのである。この方法には、解析プログラムとインタフェースとのデータの交換を管理するAPIをプログラムするだけで、既存の解析プログラムを修正することなく支援システムに組み込むことができる利点がある。しかし、ユーザがSmalltalkでAPIを直接プログラムし解析プログラムの登録作業を行うことは困難である。

(1) ファイル処理手続について この問題を解決するためにAPIに組み込まれているファイル処理手続きを分析する。前節で簡単に述べたように、解析支援システムに新たな解析プログラムを追加するにはAPIを新たにプログラミングする必要がある。APIの主な役割は、解析支援システムと解析プログラム間のデータ入出力方法の違いの吸収にあることから、APIのプログラミングとは入出力ファイル処理手続きのプログラミングである。

APIに組み込まれている入出力ファイル作成手続きを簡単に説明しておく。図3.2に示すようなフォーマットで線路データを入力するフォートランで記述された解析プログラムがあるとす。図の記号IとFはそれぞれ整数と実数型のデータを意味している。この記号に続く数字は、個々のデータの出力用の領域の大きさを表している。すなわち、I5は整数型のデータに5文字分の出力用の領域を割り当てることを意味している。

手続きの記述は、オブジェクト指向プログラミングを採用したこともあり、図3.3のように単純なプログラムで記述されている。この図の"putAll:"の右側では、変数"aLine"に格納された線路オブジェクトからメッセージを用いて線路番号、接続先のノード番号、線路イ

線路番号	線路 (from node)	線路 (to node)	線路インピーダンス (real)	線路インピーダンス (imaginary)	線路充電容量
I5	I5	I5	F10	F10	F10

図3.2 データ入力ファイルフォーマット例

インピーダンス、線路充電容量データが抽出されている。こうして抽出されたデータは、変数 "aFileStream" に格納されているファイル管理用のオブジェクトにメッセージ "format: putAll:" を送り、ファイルに出力されている。ここで、"format:" の右側の数字は、"putAll:" の後に指定された出力データの文字列の長さを指定している。線路以外の母線、変圧器、発電機などのデータについても同様な手続きで入出力ファイル処理手続きを記述することができる。この図から入出力ファイル処理を行う際に必要となる、機器の属性、入出力時の並びがわかれば容易にプログラムすることができる。すなわち、解析プログラムのデータ入出力に関する情報を GUI を通して入力するだけで、データの入出力処理手続きを組み込んだ API を作成できるのである。しかし、属性の選択・登録に関して次のような問題がある。

(2) 属性の選択について 電力系統は、複数の電気所と送電線で構成されている。そして、電気所は変圧器、母線などの機器で構成されているという階層構造を持っている。本研究では、電力系統のこうした構造をオブジェクト指向プログラミングにおける集約の概念を用いてモデル化している。集約とは、下位の概念オブジェクト（機器オブジェクト）の集合で上位の概念（電力系統）をモデル化することである。具体的には、機器毎にオブジェクトを定義し、それらオブジェクトを階層構造に従って関係付け電力系統をモデル化することである。これら各々の機器オブジェクトには、対応する機器に関するデータを属性値として格納してある。そのためファイル処理に使用する属性を多くの機器オブジェクトの中から見つけだすのは、かなり困難な作業となる。

そこで本研究では、この電力系統モデルが実系統の階層構造までも忠実にモデル化してあることに注目した。なぜならこの階層構造は、広く電力系統解析に従事する人々に受け入れられているからである。この階層構造に類似した階層型のメニューでユーザに機器オブジェクトに格納されている属性を提供すれば属性の選択が容易になるはずである。

以上のような手段を用いて解析プログラムの登録支援を行う。機器オブジェクトに分散格納されている属性の選択には、オブジェクト指向に基づき忠実にモデル化した電力系統の階層構造と階層

aFileStream format:	5	putAll:	(aLine 線路番号).
aFileStream format:	5	putAll:	(aLine 線路 -fromNodeNo).
aFileStream format:	5	putAll:	(aLine 線路 -toNodeNo).
aFileStream format:	10	putAll:	(aLine 線路インピーダンス -real).
aFileStream format:	10	putAll:	(aLine 線路インピーダンス -imaginary).
aFileStream format:	10	putAll:	(aLine 線路充電容量).

図3.3 ファイル作成手続き

型のメニューの類似性を利用する。この様子を図3.4に示す。図の階層メニューのトップレベルには、系統全体に関係のある属性名と次のレベルのメニュー（系統を構成する電気所や送電線に関する属性名を選択メニュー）へのポインタが用意されている。電気所に関する選択メニューには、同様に電気所に関する属性名と電気所を構成する機器に関する属性名を選択メニューへのポインタが用意されている。このメニューによってユーザは、ファイル処理に使用する属性名を記憶しておかなくてもファイル処理で使用する属性の登録作業を行うことができる。そして、入力された属性やフォーマットの情報を用いてデータの入出力処理手続きを組み込んだAPIを容易に作成することができる。

3.2.3 連続時間シミュレーション

3.1節でも述べたように、これまでに提案した解析支援システムは、基礎的な解析支援を目的に開発されており、機器の物理定数や運用データをある値に固定して解析を行った結果を画面上にグラフ表示する機能は備えているが、機器の物理定数や運用データをパラメータとして解析を実行し、その結果をユーザにわかりやすく表示する機能は備えていない。そこで負荷や発電機出力を断続的に変化させ解析を実行し、その時の解析結果を表示する機能を追加する。プログラム自体が設定値

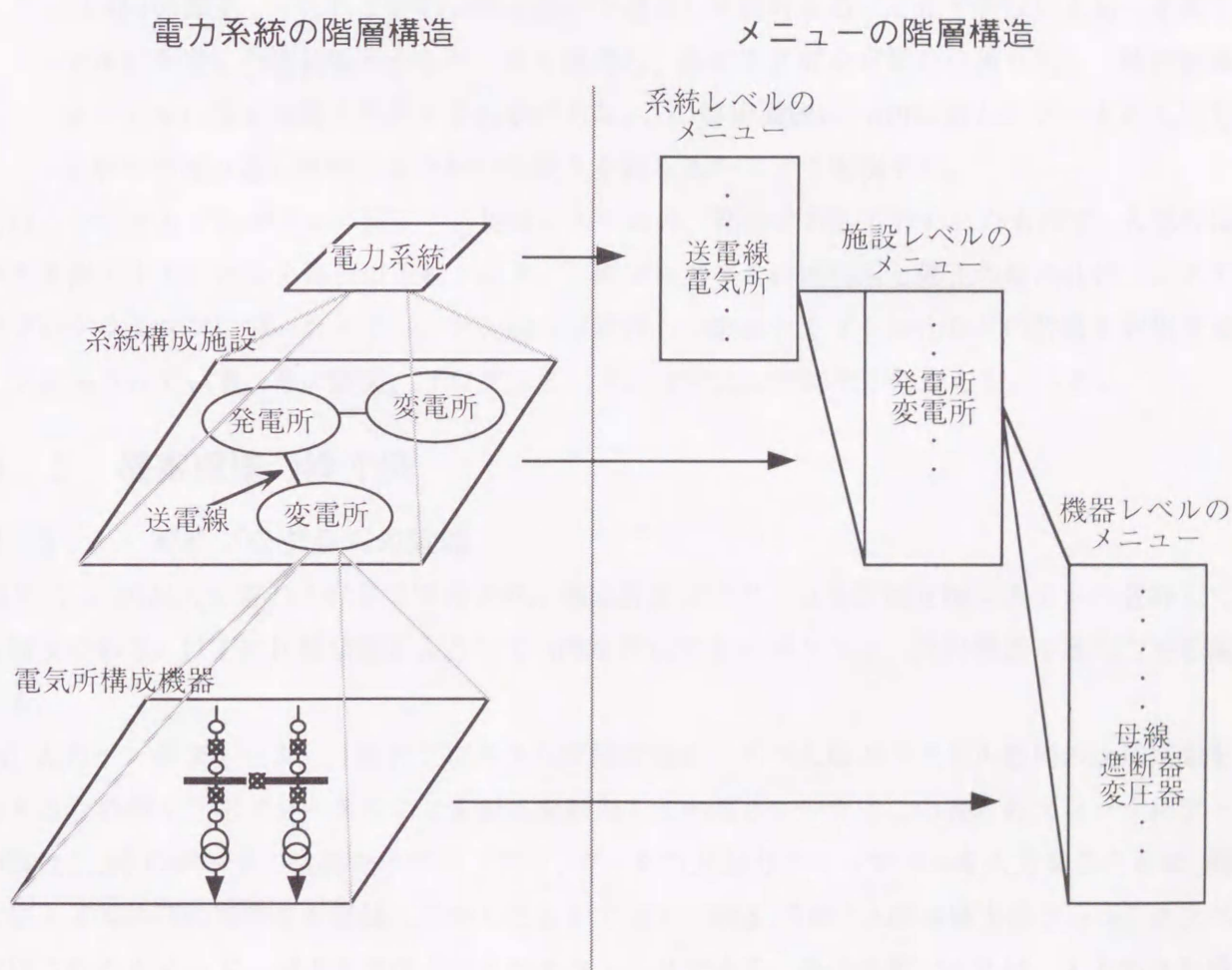


図3.4 階層メニュー

を連続的に変化させて解析を行うようになってきているものを使う方法もあるが、プログラムの使用が特化されてしまい別種の解析に使い難くなるなどの問題がある。そこで本研究では、特別な機能を持たない基本的な解析プログラムに対して、設定値を変えるたびに入力データを作成し、解析を繰り返すことで、設定値が連続的に変化したときの様子を解析できるようにする。このとき出力された結果は、毎回APIを通してシステムモデルに用意した配列に取り込むようにした。こうして解析終了後に、特定の断面の結果のみを表示したり、全断面の結果を一括表示やアニメーション表示ができるようになってきている。上記の機能を実現するためには次のようなプログラムの修正や追加を行う必要がある。

○ **インタフェースの修正** 運用データを連続的に変化させる解析には、負荷や発電機出力などの連続変化するデータを入力する機能と入力されたデータを解析した結果を表示する機能が必要となる。これまでのインタフェースには、ある断面のデータの入力と解析結果表示機能は既に用意されているので、新たに連続して変化するデータを入力するためのインタフェースと解析結果表示用のウインドーを追加する。これらのウインドーの追加には、既存のインタフェースを一部修正することで対応する。

○ **APIの修正** これまでのAPIは解析を連続して実行することができないため、システムモデルに用意した格納場所からデータを獲得し、解析を必要な回数だけ繰り返し、解析結果をモデルに返す機能を用意する必要がある。この解析機能は、APIに新たにデータの入出力と解析を繰り返し実行するための手続きを組み込むことで実現する。

これらシステムプログラムの修正や追加はシステムの一部に対して行われたもので、大部分はそのまま修正することなく再利用されている。このプログラムの再利用と修正の局所化は、システムのプログラミングに用いたオブジェクト指向言語のもつ継承やカプセル化などの特徴を利用することで実現されている。その結果、プログラミングにはそれほど時間がかからなかった。

3.3 提案環境の操作例

3.3.1 解析プログラムの登録

図3.5は、広島大学電力研究室で開発された潮流計算プログラムを計画支援システムへ登録している様子である。以下に仕様情報を入力してAPIを作成するインタフェースの構造や操作方法を説明する。

(a) **入力インタフェース** 解析プログラム登録支援ツールへ入出力ファイル処理の仕様情報を入力するためのインタフェースとして表形式を利用している。ユーザはこの表の各ブロックにデータの属性名、そのデータの格納先オブジェクト、データの入出力フォーマットを入力することで、同時にファイル処理の順序をも登録してゆくことができる。図3.5の”API編修支援ツール”とラベル付けされたウインドーが表形式の入力インタフェースである。登録支援ツールは、入力された情

報をもとに各行毎に図3.3のような入出力データ処理手続きを作成しAPIへ組み込んでゆく。この表への属性、属性格納先オブジェクト、フォーマットの登録は、次に説明するメニューを用いて行われる。

(b) 表への入力支援 入出力ファイル処理の仕様情報・登録作業を簡略化するために次のような登録支援用のインタフェースを用意している。このインタフェース(図3.5の"属性登録支援メニュー"とラベル付けされたウインドー)は、同じフォーマットやオプションが続く場合の設定を容易にするために、属性登録の際に予め設定されたフォーマットとオプションを自動的に登録する機能を備えている。属性登録メニュー上のラベル"現在のフォーマット"と"現在のオプション"の下に表示されているメッセージが、現在設定されているフォーマットとオプションである。この状態で属性を選択するとその設定値が自動的に登録される。また入力位置を常にマウスで指定して登録作業を行うのは非効率的であることから、登録が完了すると自動的に右隣の位置に入力位置を移動する機能もこのインタフェースに用意されている。

(c) 入力メニューの選択支援 図3.5の"属性登録支援メニュー"とラベル付けされたウイン

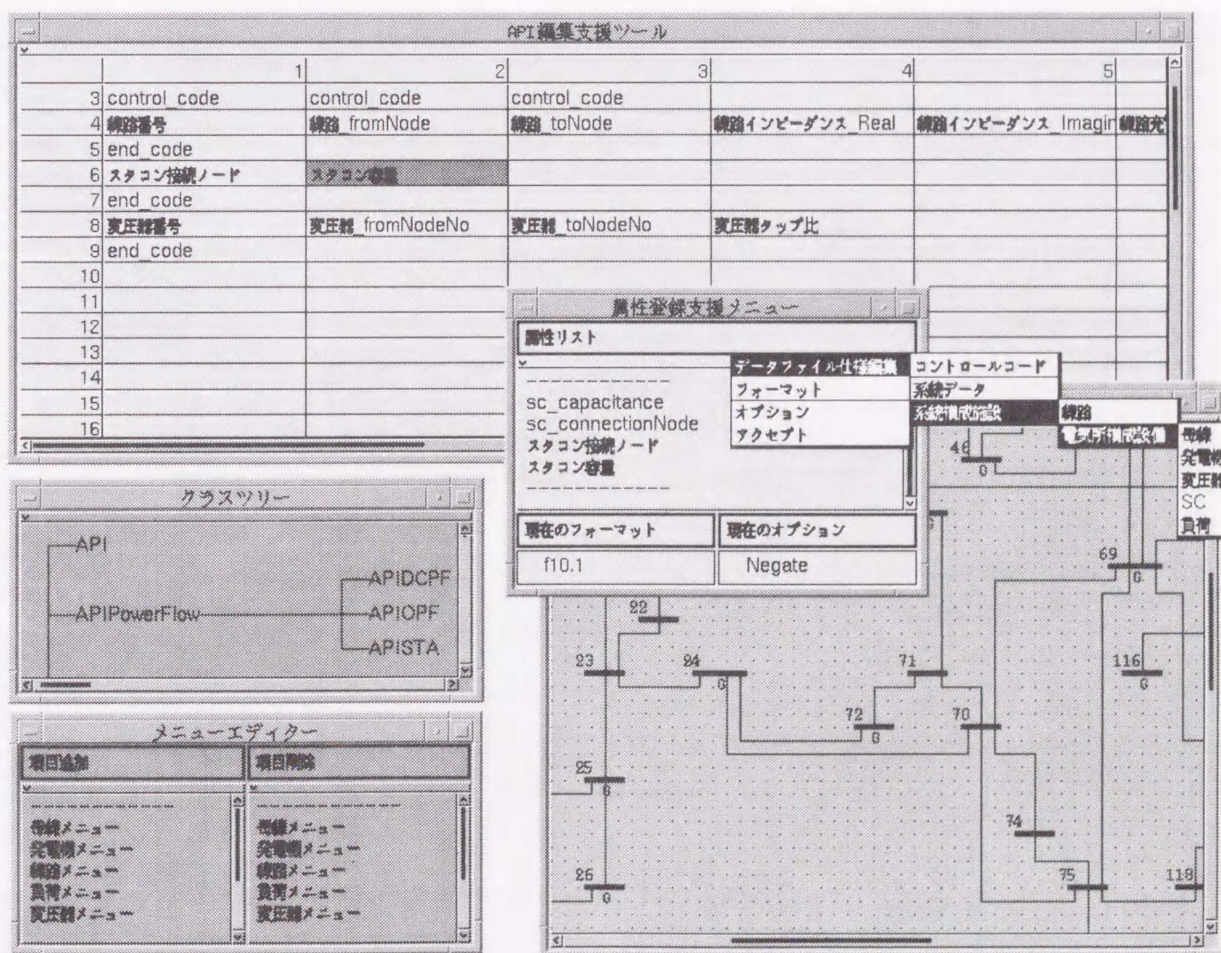


図3.5 解析プログラム登録支援

ドールにあるのが、属性選択のために設けた階層型のメニューである。このメニューは、忠実に系統の持つ構造をモデル化したことによる現実の系統と系統モデルの類似性と系統モデルの持つ層構造と階層メニューの類似性を利用して系統構成オブジェクトの検索効率の向上を図っている。図は、最下層の機器オブジェクトを選択するときの様子である。表示されたメニューの中から目的のオブジェクト名を選択すると、登録支援ツールが属性の選択・登録を行うための属性一覧を表示し、ユーザに選択を促す仕組みとなっている。またこのメニューは、一度に表示する選択項目数を限定し、段階的に属性のある階層へユーザを導くことで、ユーザがどのオブジェクトにどのような属性が格納されているかを詳しく知らなくてもスムーズに選択作業が行えるようになっている。

3.3.2 連続時間シミュレーション結果の表示

図3.6は、実際に安定度解析を実行した様子である。図右下の"Substation 53 24時間データ"とラベル付けされたウインドーは、24時間分の負荷の変化を登録するためのものである。このウインドーは、"System Editor"とラベル付けされた系統図表示画面上のノードをマウスを使って開いたものである。容易に各負荷に対して任意の負荷変化を設定することができるようになっている。発電機に対しても時間毎の出力変化を登録する同様なウインドーが用意されている。

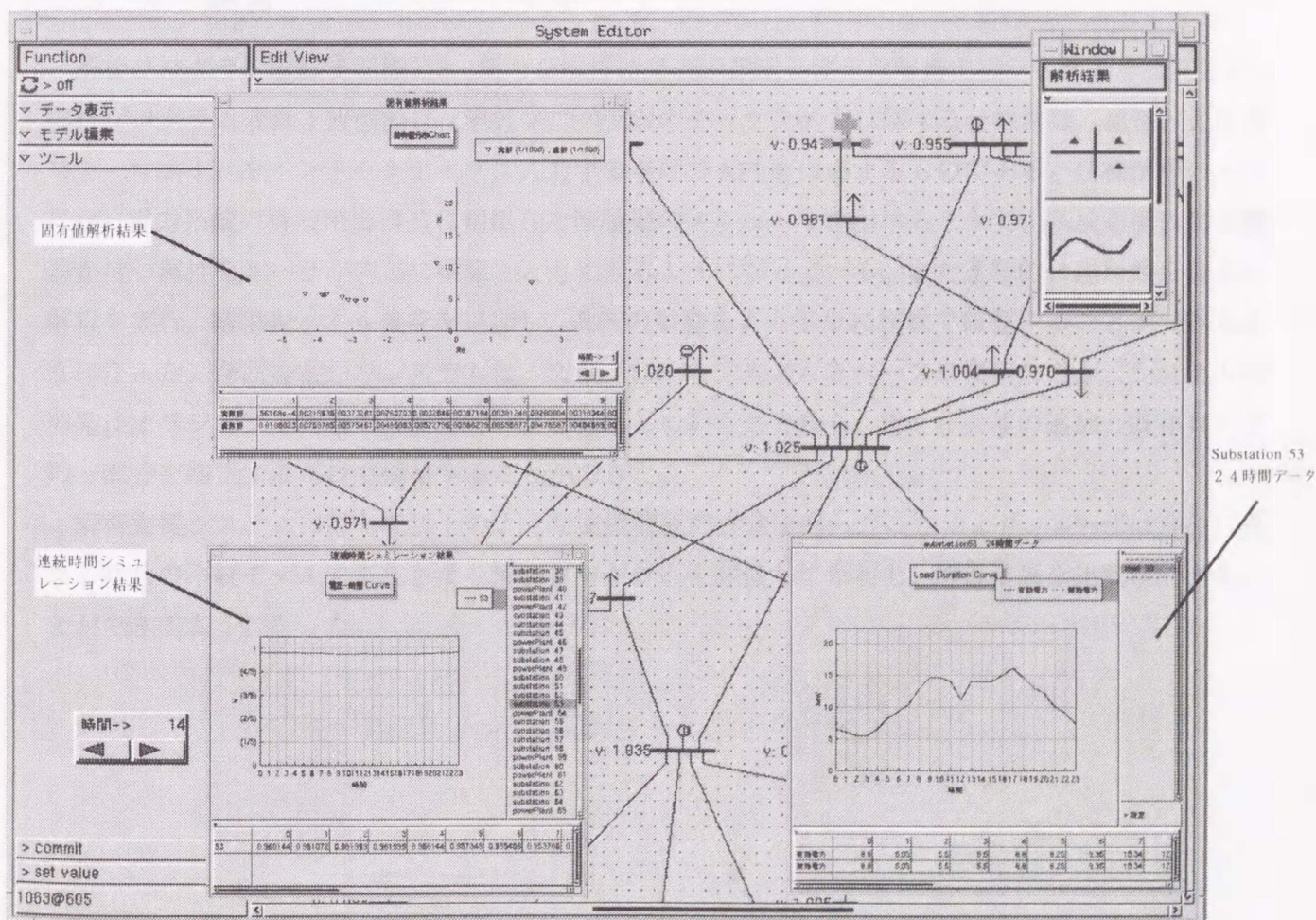


図3.6 解析プログラム実行例

運用データの設定が終わり解析を実行すると結果の表示へ移る。図の "System Editor" は、安定度解析の前処理の段階で実行された潮流計算の結果を表示している。エディター上には最初の時刻の電圧と位相角が表示されているが、時刻を設定するためのスイッチがあり、任意の時刻のデータを表示することができる。図左下の "連続時間シミュレーション結果" とラベル付けされたウインドーは、電圧変化の様子を表示している。このウインドーは、"System Editor" 上のノードをマウスで選択して開いたものである。開いた直後は、マウスで選択したノードのデータのみを表示するが、あとから表示画面右側にあるノードリストから比較したいノードをグラフ上に追加表示することができる。画面下にあるテーブルは、数値データを確認するためのものである。図左上の "固有値解析結果" とラベル付けされたウインドーは、定態安定度解析プログラムで求めた固有値を表示したものである。図のように求めた固有値が複素空間上に表示される。ウインドー上の時刻ボタンをクリックすれば任意の時刻の固有値の分布を確認することができる。このほかに固有値の変化の軌跡をアニメーション表示することもできる。この表示にも先の電圧表示画面と同様に、数値データを確認するためのテーブルが用意されている。

3.4 結言

解析プログラム登録支援機能と、新たな解析結果表示機能を備えた解析支援システムを提案した。解析プログラム登録支援機能は、解析プログラムのデータ入出力に関する仕様情報（属性、入出力フォーマット）をインタラクティブに入力するだけでAPIを作成するものである。仕様情報の入力には、電力系統の持つ階層構造に類似した階層型のメニューを採用することで、系統を構成する機器が持つ属性をユーザが容易に検索・入力できるようになっている。また連続的に運用値を変えた解析を実行、結果表示する機能を追加し、系統の挙動をより簡単に視覚で確認することができるようになった。今回提案したシステムは、文献(11)のシステムをベースに構築したが、システムの開発にオブジェクト指向プログラミングを採用していたことから、新たな機能の追加と既存コードの一部を修正するだけで構築することができた。

解析支援システムに対して以上のような支援機能の改善を行ったことにより、ユーザは容易に異なる複数の系統モデルをさまざまな解析プログラムを駆使して解析し、解析結果を比較検討することができるようになった。

第4章 電力系統図編集作業の自動化

電力系統は極めて複雑・大規模であるため実験的に解析を行うことは現在ほぼ不可能であり、計算機シミュレーションによる系統解析が一般的である。解析結果を直観的に理解し易くするためには電力系統図に重畳して表示する必要がある、さらに、解析対象系統の構成を把握するためにも系統図の作成が必要となる。

従来、電力系統図は接続状況を示すデータから直接手書きで描いていたため多大な労力を費やしていた。したがって、系統図の書き換えにも多くの時間がかかり、必然的にミスも発生していた。そこで、作図処理の時間短縮と労力軽減を目的として、電力系統図自動作成システムの開発が行われている^{(14)~(18)}。これらの研究の主目的は表示用データ作成の高速化と表示の見易さを追求している。電力系統図の自動作画システムとしては文献(19)があるが、対象となる系統を基本的に放射状系統に限定しており、系統図表示に関してはフラクタル木表現を用いている。

見易さの尺度の1つは、表示された系統図の線路(ブランチ)どうしの交差数が少ないことである。つまり、系統図には現実の系統の持つ地理的情報を忠実に反映することよりも、若干位置関係がずれても系統構成の認識しやすさが求められる。電力系統図表示における線路交差数最小化は文献(14)で若干考慮されているものの、これを主題とした研究は著者らの知る限り見当たらない。ただし、本主題に関する研究としては平面グラフ^{(20)~(23)}、多段グラフ⁽²⁴⁾、VLSI・プリント基盤などの配線問題^(25, 26)における表示法についての結果がある。

本論文ではVLSIの配線設計で使用されている混雑度の概念に基づく最短経路探索法を利用することにより、ブランチの交差数ができるだけ少ない系統図の自動作成を行う。さらに、得られた系統図においてブランチの経路をさらに見やすくするためにラバーバンドとスポークを利用するアルゴリズム⁽²⁷⁾を用いて経路の整形を行っている。本システムではこれらの手法を用いて、ユーザがノードの配置位置を指定するだけで、自動的に線路の交差数の少ない系統図の作成を行うことができる。さらにノード配置に不都合がある場合、画面上で対象ノードを移動することができ、移動ノードに接続する線路についてのみ配線アルゴリズムを再実行することにより、よりわかりやすい系統図が得られる。

本章で提案する方法⁽¹³⁾は電力系統図作成の特性とは必ずしも完全に合致するものではないが、系統図作成の自動化・高速化には非常に有用である。また、原理的にはノード数およびブランチ数に制限がないため、かなり大規模な系統にも適応できるシステムが構築可能である。

4.1 最短経路探索法による系統図の作成

本システムでの系統図作成は、まずノードと線路の接続状況を与える入力データに対し、図4.1に示す多重2層格子グラフ(以下"格子グラフ"と呼ぶ)の大きさ、および各ノードの格子グラフ

上の配置位置を指定する。次に、各格子枝に混雑度を考慮したコストを割り当て、接続されるべきノード対間のコストが最小となるように最初の2層格子に対してブランチの経路を決定する。この時、経路が決定できなかったブランチが存在する場合、2層格子を順次追加することで配線経路を決定するため、概念的には図4.1の多重2層格子グラフを考慮する。決定されたブランチの経路を格子グラフ上に埋め込み、最終的に系統図の表示を得る。以下、各手順を説明する。

4.1.1 データ入力

まず、ノード対を基本的情報とする線路データを入力する。次いで、格子グラフの大きさを入力した後、ノードの配置位置をディスプレイ上でマウスをクリックするか、または配置位置を指定したファイルを読み込むことによって指定する。多重割り当てを防ぐため、使われた格子点は使用禁止としておく。

4.1.2 混雑コスト計算

与えられた格子グラフにおいて、図4.2に示す2方向の枝カット群 C_1 、 C_2 を割り当てる。ここで、各ノードが指定されている格子点を端子と呼び、各ブランチの両端ノードに対応する端子対を接続要求と呼ぶ。各接続要求について、これを横切るすべてのカット $K \in C_1 \cup C_2$ について以下の操作を反復して、各格子枝 $e \in K$ にコスト $C(e)$ をつける。

$$C(e) \leftarrow C(e) + (1 / |K|)$$

ただし、 $|K|$ は K が通過する総格子枝数で、すべての格子枝 e' に対して初期値 $C(e') \leftarrow 0$ とする。

具体例として、図4.3の格子グラフに接続要求 $\{a_1, a_2\}$ 、 $\{b_1, b_2\}$ がある場合のコスト計算法を説明する。図4.3の実線は $C_1 \cup C_2$ のカットのうち、与えられた接続要求を横切るカットを示している。まず、図4.3(a)の枝カット群 C_1 についてコスト計算を行う。カット $K_1 \in C_1$ は、接続要求 $\{a_1, a_2\}$ のみを分割し、かつ格子枝を2つ横切っているので、その格子枝にそれぞれ $(1/2) \times 1$ のコストを与える。また、カット K_2 は2つの接続要求 $\{a_1, a_2\}$ 、 $\{b_1, b_2\}$ を分割し、かつ格子枝を4つ横切っているため、各格子枝にそれぞれ $(1/4) \times 2$ のコストを付与する。すなわち、すべての $K \in$

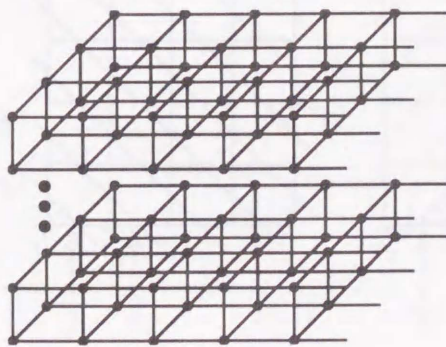
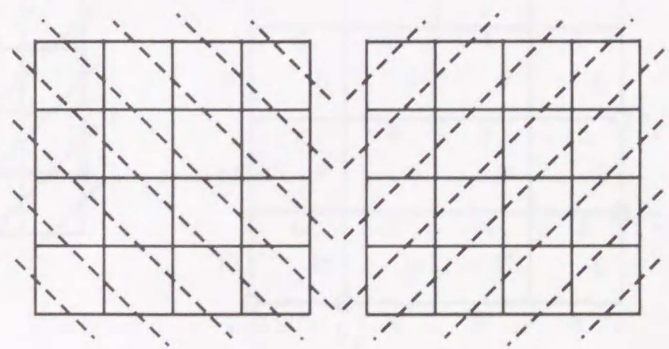


図4.1 多重2層格子グラフ



(a) カット群 C_1 (b) カット群 C_2

図4.2 使用される枝カット群

$C_1\Pi \cup C_2\Pi$ に対して、コスト

$$(1/|K|) \times (K \text{ が横切る接続要求数})$$

を K 中の各格子枝に与える。同様に、 $C_1\Pi$ 中の他のカット、ならびに図 4.3 (b) の枝カット群 $C_2\Pi$ についてコスト計算を行う。図 4.3 の接続要求についてコスト計算を行い、24 倍して正規化した値を図 4.4 に示す。

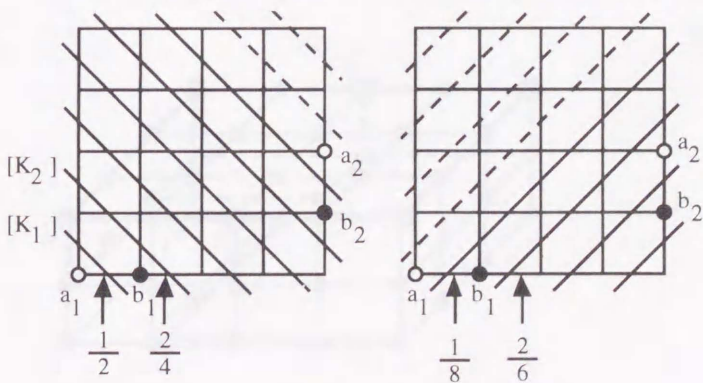
ここで求めた格子枝のコスト $C(e)$ は e の混雑度と呼ばれ、混雑度の大きい格子枝ほどその格子枝を通る接続要求が多いことを示している。1つの接続要求は格子グラフ上の1本の経路として実現(表示)されるが、混雑度の小さい格子枝をできるだけ通過するように経路を選んで接続要求を実現すれば、ブランチ交差数(重なり数)の少ない表示が得られることが予想できる。

第2層の格子枝のコストは対応する第1層のコストと同じ値とする。また、第1層にできるだけ多く配線するために、第1層と2層を接続する格子枝のコストは大きくして、最短経路探索において1層から2層へまたがる経路が選ばれるのを防止するようにする。

4.1.3 最短経路の探索

次に、前項で求めた格子枝のコストを使い Dijkstra 法によって最短経路を求める。探索においては、ノードが置かれた格子点を経路が通らないようにノードが割当てられた格子点をすべて使用禁止としておく。順次、結線しようとする両端子に相当する格子点を使用可能として Dijkstra 法で最短経路を求めていく。

最短経路を求めるには、例えば u を出発点とし、格子点 u, v 間の最短距離 $d(u, v)$ が得られているとき、 v に隣接している各格子点 w に対し各格子枝 $e=(v, w)$ が $d(u, v) - d(v, w) = C(e)$ を満たせば e を最短経路上の枝とする。なお、条件を満たす格子枝が複数存在するならば、経路の曲がり角が最小の格子枝を選ぶ。求めた最短経路を通るすべての格子点を使用禁止として、次の接続要求の結線を行う。このとき、第1層と2層の線路との重なりを少なくするために、第1層の最短経路上にある格



(a) カット群 C_1' (実線) (b) カット群 C_2' (実線) (各格子枝の横の数字が得られたコスト)

図 4.3 コスト計算例

	6	4	0	0
6	6	4	0	3
8	6	6	7	8
12	8	9	14	10
12	12	11	14	12
	12	15	16	12
	12	15	20	14
	12	15	20	14
	12	15	20	6
	a_1	b_1		

図 4.4 コスト計算結果

子枝に対応する第2層の格子枝には大きめのコストを与える。逆に第2層上を最短経路が通るときには、第1層に対して同じ処置を施す。図4.4のコストについて接続要求 $\{a_1, a_2\}$ の a_1 を出発点とする各格子点までのコストを求めると図4.5となる。また、接続要求 $\{a_1, a_2\}$ の最短経路を求めると図4.5の太線で示す線路が得られる。

1回目の経路探索で100%配線が得られなかった場合は、配線済みの2層格子グラフの下に新たに2層格子グラフを追加して得られる2重2層格子グラフ(3層格子グラフ)の3層目に対して、未結線接続要求(ノード対)のみを対象として最短経路探索を行う。図4.6にこの例を示すが、網かけ線で示した経路が新しく追加された格子グラフ上での配線である。このときの既にブランチが通っている格子枝に対応する新しい層の格子枝には大きいコストをつけることにより、ブランチの重複を防いでいる。

4.2 配線後の系統図の整形

第4.1節で自動配線した系統におけるブランチは縦方向と横方向の線分のみで表されているが、各母線間を直線的に結ぶことでより実用的な系統図となる。そこで、系統図のブランチの経路をラ

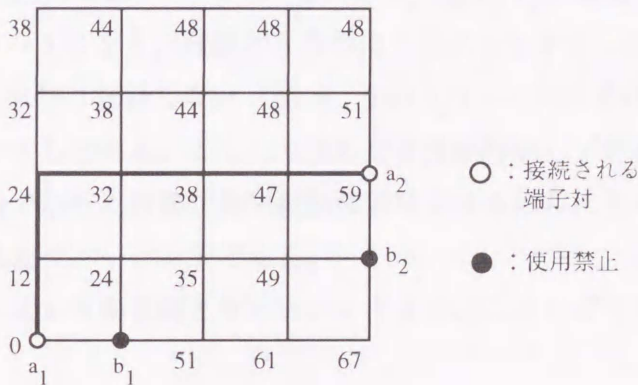


図4.5 図4.4のコストに関する最短経路

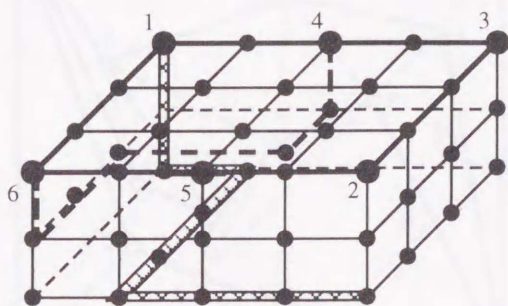
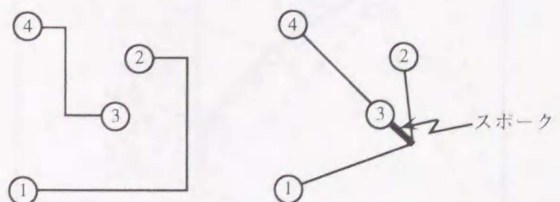


図4.6 3層目での配線
(網かけ線で示した経路)

バー(ゴム)バンドとスポークを求めるアルゴリズムを用いて整形する。ここで前者は経路をゴムひもと考えると、図4.7(a)の経路を図4.7(b)のように変換することであり、後者はノードと線路の間隔を空けるために挿入するスペースで、図4.7(b)の太い鎖線で示す。

4.2.1 配線領域の3角形分割

系統図のラバーバンドを計算するための準備として、図4.8に示すように系統図の配線領域を各ノードを頂点とした3角形に分割する。配線領域全体を3角形に分割するため、全ノ



(a) グリッドに沿った経路 (b) ラバーバンド表現

図4.7 ラバーバンドとスポーク

ドを内部に含む矩形を考え、その4隅の頂点を n_1, n_2, n_3, n_4 とする。ただし、 n_1, n_2, n_3, n_4 はそれぞれ左下、左上、右上、右下の頂点である。また、 $n_1 \sim n_4$ も含めた矩形内のノードの集合を V と表す。本論文では3角形分割を求める方法としてパス探索に基づく方法を用いる。パス探索法は、次の条件を満たすパス P_0, P_1, \dots, P_k ($k \geq 2$) を求めることにより3角形分割を行う。

- (1) どの P_i も n_2 から始まり n_4 に終わる。
- (2) どの2本のパス P_i, P_j ($i \neq j$) も n_2, n_4 以外には点を共有しない。
- (3) V のどの点も P_0, \dots, P_k のどれか1本のパスに含まれる。
- (4) $P_0 = \{n_2, n_1, n_4\}$ かつ $P_k = \{n_2, n_3, n_4\}$ である。

図4.8では $k=4$ で $P_1 = \{n_2, 2, 1, n_4\}$, $P_2 = \{n_2, 4, 3, 5, 8, n_4\}$, $P_3 = \{n_2, 6, 7, 9, n_4\}$ である。各 i ($1 \leq i \leq k$) に対して、 P_{i-1} を参照しながら P_i を求めるが、その過程で P_{i-1} と P_i にはさまれた領域内を3角形に分割していく。

4.2.2 系統図のラバーバンド

ラバーバンドを求めるには、ブランチを表す経路を辿りながら、通過した3角形と共に配線領域の境界線を記録しておく。例えば、図4.9では s から t に向かって $T_1 \sim T_8$ の順に通過する。また、3角形 T_1, T_2, T_3, T_4 を通過した後の左側境界線は $s \rightarrow 6$ 、右側境界線は $s \rightarrow 3 \rightarrow 5$ となる。経路が3角形に一旦入ってすぐに出る、つまりこの3角形を通過しないときはその3角形は記録しない(図4.9の T_6)。これはその3角形内で不必要な回り道をしているに過ぎないからである。こうして求めた3角形列からブランチのラバーバンドを形成する。図4.9では実線で示した経路が最短経路探索により求められたグリッドに沿う経路で、太線で示した経路が得られたラバーバンドである。

次に3角形列とラバーバンドを求めるアルゴリズムを説明する。左側と右側の境界線を示す頂点

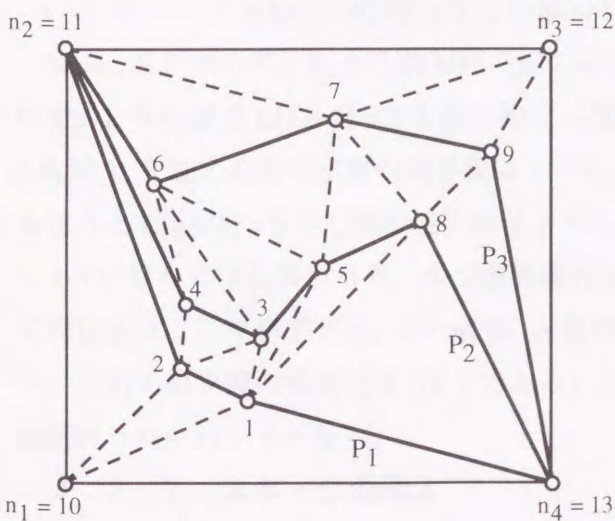


図4.8 配線領域の3角形分割

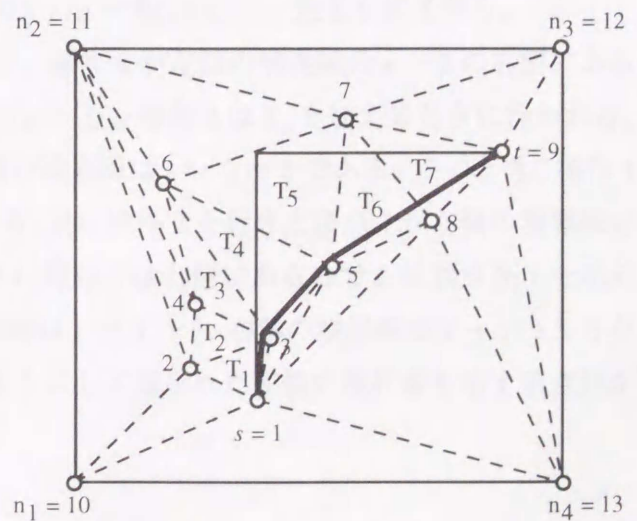


図4.9 ブランチの経路のラバーバンド

列を $L[1, \dots, l], R[1, \dots, r]$ とする。ただし、 l, r は現時点での左側、および右側の境界線上の頂点数とする。ブランチの初期端点 s を1つの頂点とする3角形の中からブランチの経路と交差する辺をもつ3角形(図4.9の T_1)を3角形列の先頭とする。また、初期端点 s を左側と右側の境界線の先頭とし($L[1]=s, R[1]=s$)、 T_1 の残りの2つの頂点について経路の左側にある頂点を左側の境界線に加え、右側にある頂点を右側の境界線に加える(初期値は図4.9では $L[2]=2, R[2]=3$)。また、左右の境界線が初めて合流する点を q とし、初期値 $q \leftarrow s$ とする。合流点とは、例えば図4.9において3角形 T_5 まで通過した後の左側の境界線は $s \rightarrow 3 \rightarrow 7$ 、右側境界線は $s \rightarrow 3 \rightarrow 5$ となるが、 $s \rightarrow 3$ までは左側と右側は境界線を共有しており、頂点3から以降に分かれているので、この時点での合流点は $q=3$ となる。

次に T_i までを処理済みとし、ブランチの経路と交差している辺を1辺とする3角形 T_{i+1} の新しい頂点 u がブランチの経路に対して左右どちらの側にあるかを調べる。もし、 u が左側にあるときは以下に述べる操作1,2を行う。逆に頂点 u が右側にあるときは操作1,2に対して左右対称な操作を行う。

操作1: $L[l]$ から $L[q]$ へと走査(変数 v を l から q へと変化)させながら以下を実行する。

- (1) u が境界線分 $L[l-1] \rightarrow L[l]$ よりも左側にあるならば u を L に加えて走査を終了する。
- (2) u が $L[v] \rightarrow L[v+1]$ の右側にあるときは $L[v+1]$ を L から除き、さらにもし u が $L[v-1] \rightarrow L[v]$ の左側にあるならば u を L に加えてから走査を終了する。

操作2: $R[r]$ から $R[q]$ へと走査(変数 w を r から q へと変化)させながら以下を実行する。

- (1) u が境界線分 $R[r-1] \rightarrow R[r]$ の右側にあるならば $R[q+1], \dots, R[r], u$ をこの順に L に加え $q \leftarrow u$ とする。
- (2) u が $R[w] \rightarrow R[w+1]$ の左側にあり、 u が $R[w-1] \rightarrow R[w]$ の右側にあるならば $R[q+1], \dots, R[w], u$ をこの順に L に加え、 $q \leftarrow R[w]$ とし、走査を終了する。

図4.9において、 T_2 を3角形列に加えるときに、頂点4が左側の境界線分 $s \rightarrow 2$ の右側にあるので、 L から頂点 $L[v+1]=2$ を取り除く(操作1)。一方、操作2は T_5 を加えるときに行われる。3角形 T_5 を加える前の左側の境界線は $s \rightarrow 6$ 、右側の境界線は $s \rightarrow 3 \rightarrow 5$ である。このとき、操作1を行うと $L[v+1]=6$ で左側の境界線は s だけとなる。次に操作2を行うと頂点7が右側の境界線分 $3 \rightarrow 5$ に対しては左側にあり、かつ境界線分 $s \rightarrow 3$ に対しては右側にあるので L に頂点3,7を加えて操作2はここで終了する。この結果、左側の境界線は $s \rightarrow 3 \rightarrow 7$ 、右側の境界線は $s \rightarrow 3 \rightarrow 5$ となり、左右の境界線の合流点は $q=3$ である。このようにして得られた左側の境界線を示す頂点列が経路のラバーバンドとなる。

4.2.3 スポークの挿入

ラバーバンド表示のみでは、接続していないノードにブランチが接続されているように見えるの

で、スポークと呼ばれる障害物を使ってブランチの経路に制約を与える。例えば、図 4.9 では、ノード対 $\{s, t\}$ 間のブランチがノード 3, 5 と接続しているように見えるので、ノード 3, 5 とブランチとに間隔を置くためにスポークを用いる。ここで使用するスポークは傾きが $45^\circ, 135^\circ, 225^\circ, 315^\circ$ の 4 方向である。

まず最初に、ブランチの経路が途中の非接続ノードにより曲げられているときにスポークを挿入するが、複数のブランチの経路が曲げられているときは、スポークの長さを変えてブランチどうしの交差を避ける必要がある。すなわち、挿入済みのスポークの上に新しいスポークをつけて、その上をブランチを通す。その際、経路の偏向角度の小さいものほどスポークの長さを短くして内側を通す。この例を図 4.10 に示す。すなわち、ノード n 上を 2 つの経路 $\{a_1, a_2\}$ と $\{b_1, b_2\}$ が通過しているが、 $\angle a_1 n a_2 < \angle b_1 n b_2$ なので、 $\{a_1, a_2\}$ は $\{b_1, b_2\}$ の内側を通している。

スポークの挿入により移動した経路が割当て前の経路と交差する可能性があるので、スポークを挿入した後に、全経路を走査して交差しているブランチを探す。交差しているときは、そのスポークをさらに延ばして経路を変更する。このような一連の操作により経路どうしの中に一定の間隔をもたせ、系統図をさらに見やすくすることが可能となる。

4.3 実行結果

4.3.1 系統図の表示

系統図の表示には X ウィンドウを使用し、対話機能には X Toolkit を使用している。また、負荷ノード、発電機ノードを一般的な記号を用いて表示する。さらに、拡大、縮小（ズーム）機能の他、中心ノードを指定することによる部分表示もできるようにしている。図 4.11 に IEEE 118 母線系統での出力結果を示す。

4.3.2 系統図の変更

自動配線が完了したあとの系統図において、ノードの配置位置を変更したい場合には、変更するノードをマウスでクリックして、つぎにその移動先を指定すればよい。このとき移動されるノードに接続しているブランチが割り当てられている格子点をすべて使用可能とし、すべての格子枝のコ

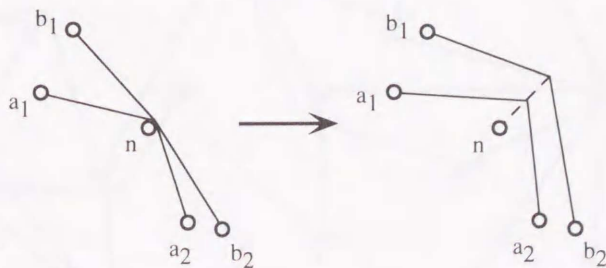


図 4.10 スポークの挿入

ストを 1（最小値）としておく。つぎに、そのノードに接続しているブランチについてのみ再び最短経路探索を行う。すなわち、ノードを移動させることによって影響を受けない経路はそのままにしておく。これから再びラバーバンドの計算を行い、スポークを挿入する。この手順により、ノードの配置位置を変更するとき

の計算速度を高速化できた。

Sun SPARC station 10 (約 100 Mips) 上での配線経路決定に要する計算時間は IEEE30 母線系統で 4.17 秒、IEEE57 母線系統で 23.82 秒、IEEE118 母線系統で 29.38 秒である。それぞれ格子の大きさは 70×80 、 150×140 、 152×124 であることから計算時間は格子の大きさと系統規模に依存することがわかる。

4.4 結言

本章では電力系統単線結線図の自動描画を目的として、VLSI の自動配線に使用されている混雑度を基本とし、線路の交差が少ない系統図が得られる手法を提案した。本システムは、電力系統図のようにノードの配置がある程度固定されているネットワークの表示には十分有用である。また、格子グラフ上に埋め込まれている系統図のブランチの経路をラバーバンドとスポークのアルゴリズムを用いて整形することにより、系統図がさらに見やすいものとなっている。

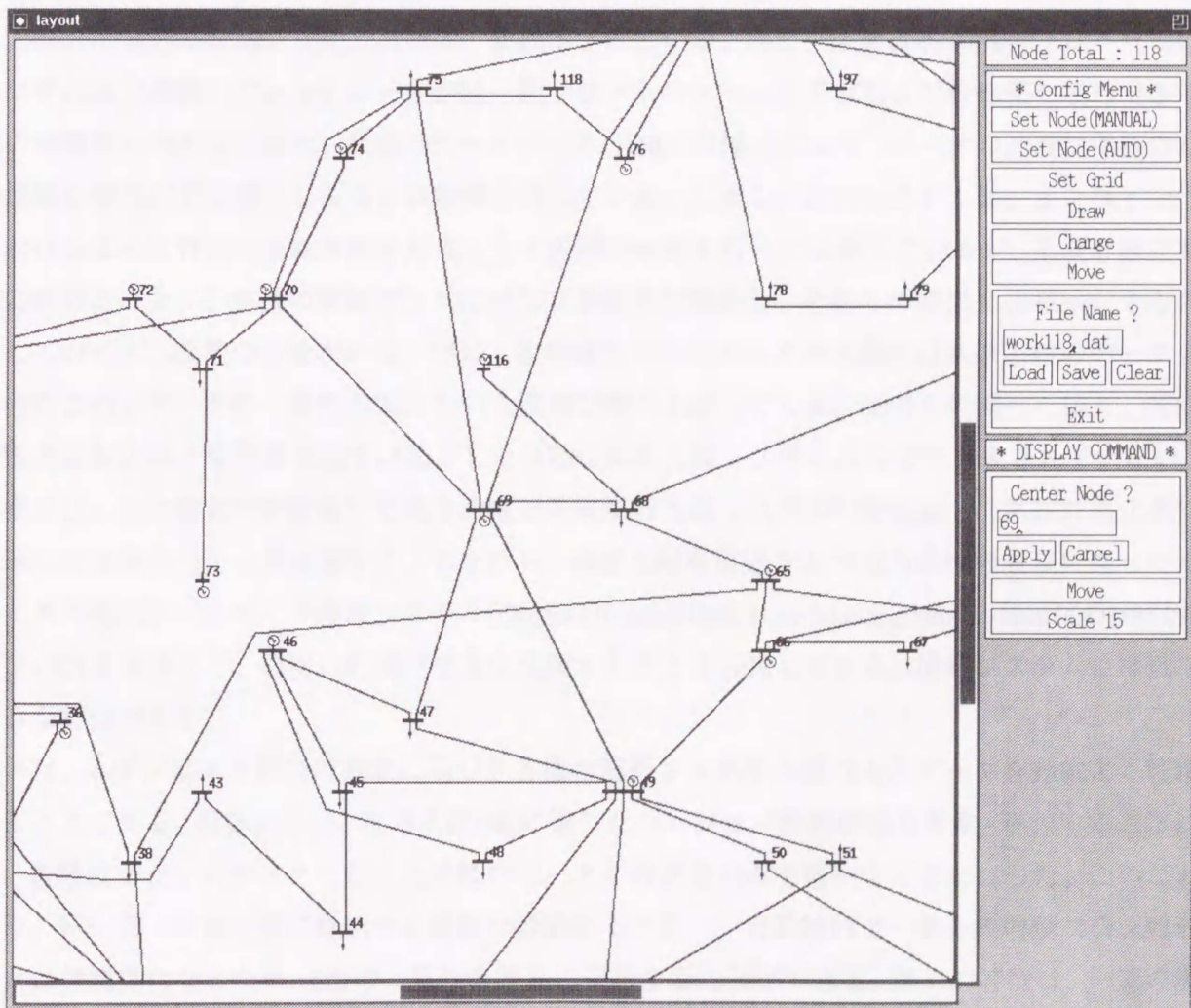


図 4.11 IEEE 118 母線系統での表示結果

第5章 電力系統計画立案支援環境の提案

電力各社は、近年の着実な電力需要の増加に対応して適切に電力設備を増強する必要性に迫られている。本研究で対象とする流通設備拡張計画（系統計画）の立案に際しては、各増設プランに対してコストや安定性の評価を行い妥当性を検証する必要がある。まず、電力需要の増加に対応した増設プランを多数用意し、各増設プランに対してオペレータが多様な運用条件や故障条件を設定しながら、潮流計算、安定度解析、故障計算等の一連のプログラムを用いて解析する。そして、妥当とみなされる増設プランをいくつか選択し、最終的な計画決定段階へと進んでいく。このように、系統計画作業では、数種類の解析プログラムを多数の計画案に対して連続的に適用する必要があり、それらの実行回数は莫大なものとなる。

一方、各種解析ソフトの操作性の悪さはかねてより指摘されており、このためオブジェクト指向に基づいたグラフィカルユーザインタフェース（Graphical User Interface: GUI）を用いてプログラム操作を支援するシステム⁽²⁸⁾、および筆者らによるオブジェクト指向データベース（OODB）を中核とした統合的電力系統解析支援システム⁽¹¹⁾が提案されている。特に、後者は各種解析プログラムの操作にそれほど精通していないユーザが統一的平台上で容易に解析作業を実行できる環境の構築を目的としており、共通のデータベース（DB）の採用により、データの一貫性保持のための煩雑な操作が不必要となるなどの特徴を有している。しかし、このシステムは、オンライン運用におけるように特定の系統構成を対象として各種の解析を行うには適しているが、系統計画立案作業におけるごとく、多数の増設プランに対して多面的に解析し、それらの結果を比較検討するには適しておらず、改善の余地がある。特に、各増設プランのデータの大部分は基準系統のデータを多く含むため、データの一貫性の保持という難題が持ち上がってくる。筆者らの調べた限り、国内では電力系統計画立案作業支援を目的とした有効な解析支援システムは今まで発表されていない。

本章では、先に筆者らが提案した統合的電力系統解析支援システム⁽¹¹⁾を拡張し、系統計画立案作業に適した支援システムを提案する。すなわち、複雑な階層構造をもつ電力系統の表現に適したオブジェクト指向データベース管理システム（Object Oriented Data Base Management System: OODBMS）およびGUIを基本とし、適切に計画担当者を支援することを目指している。提案システムには以下のような特徴がある。

第一は、オブジェクト指向の概念に基づき各種の解析プログラム用の入力データを効率よく管理できることである。具体的には、電力系統DBに適したバージョン管理機能を考案・導入することにより、各増設プランのデータとして元系統のデータとの差分のみを格納することにした。このことにより、同一データを大量に格納する無駄を排除すると共に、元系統のデータと各増設プランのデータとは無関係になるため、DBの一貫性を容易に保持することができる。第二の特徴は、一連の解析プログラムを一つのアプリケーションとして自由に登録できる点である。これにより、計画者自

身の希望する解析過程を自動化することができるため操作性が極めて良くなり、プログラム操作上のミスもほぼ根絶できる。最後に、系統計画案の評価結果の理解が容易に出来るように、視覚的に表示する方法を採用していることである。

提案システムはOODBMS VersantとSmalltalkを用いてUNIXマシン上に構築しているが、その大部分は先に開発した統合的電力系統解析支援システム⁽¹¹⁾のプログラムコードを再利用する。システム構築言語としてオブジェクト指向言語を採用したことがより効果的な再利用を可能としている。

5.1 電力系統計画^(30, 31)

5.1.1 計画立案作業の概要

図5.1に系統計画策定の概要を示す。最初に、将来の需要を予測し、過負荷あるいは運用指標値を逸脱した箇所がないか調べる。問題があれば系統構成を変更し、所定の解析プログラムを用いて平常時と事故時の系統特性を調べる。潮流面、電圧面、供給信頼度面から判断して、設備増強が必要な箇所を抽出し、設備増強案を選定する。次いで、過剰な設備増強とならないように、将来の需要動向や経済性を十分検討して必要容量を決定すると共に、工期、関連工事との整合性、資金計画等を総合的に検討し設備の増強時期を定める。続いて、系統の性能面、運用面、工期や費用面を考慮して系統構成を詳細に検討する。最後に、作成した増強案を経済性、施工面、運用面から検討し、最も適切な計画案を選択する。その後、方針変更等によって状況に変化が生じた場合は、再び計画の見直しを行う。

立案作業で最も重要なのが増設プランの供給信頼性評価作業である。これには、数種類の解析プログラムを全ての増設プランに対して適用した結果を比較検討しなければならないため、解析操作の繰り返しだけでなく、データの後処理もユーザにとって大きな負担となる。これらの問題を解決

するための機能を次項で検討する。

5.1.2 系統計画立案作業に必要な機能

多くの増設計画案を比較・検討するために、解析プログラムへの入力データと解析結果を個別に管理しておく必要がある。各増設プランは基準系統を一部修正して作成するので、それらのデータの大部分は基準系統のデータと同一である。したがって、もし増設プランのデータを別々に格納すると、DB上に同一データが数多く格納されることになり、データの一貫性の保持が厄介になるだけでなく、メモリの無駄遣いでもある。これを避けるため、基準系統データとの差分のみを格納・

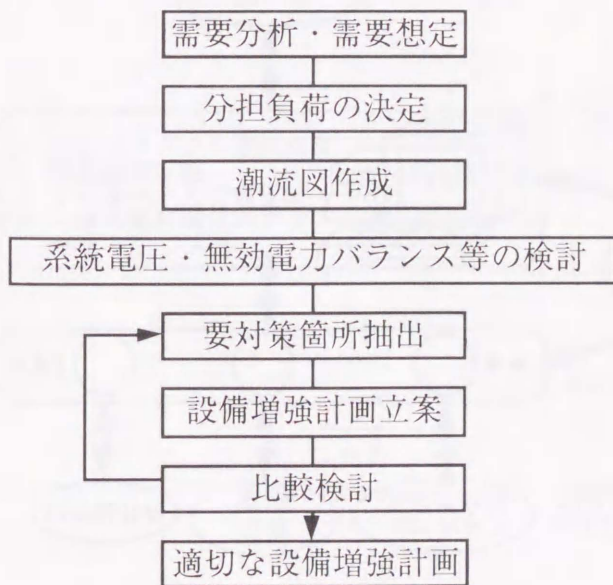


図5.1 電力系統計画立案作業

管理する仕組みが必要である。

次に、信頼性評価作業を効率的に実行するために、一連の解析操作をアプリケーションとして登録し、全ての増設プランをこのアプリケーションで自動的に解析し、結果をユーザに理解しやすい形で提供する機能が求められる。

本章では、これらの要求を満たす電力系統計画支援システムを提案する。

5.2 計画立案支援環境の提案

5.2.1 構築方針

本研究では先に挙げた問題点を踏まえて、次の方針に従ってシステムの構築を試みる。

方針1 解析プログラムへの入力データをDBで一括管理する。すなわち、ファイル形式で分散管理されている入力データをDBで一元管理することで、データ管理の負担を軽減し、メモリの有効利用を図ることを目的とする。

方針2 GUIを用いて対話的に計画立案作業（解析操作やデータ管理）を実行する。これは、ユーザが使用する計算機システムに対して、それほど詳細な知識や技能を必要としない環境を整えるためである。

5.2.2 支援環境の構成

提案システムは図5.2に示すように、ユーザ支援のためのGUI、電力系統DB、アプリケーションプログラムインタフェース(API)、および解析プログラム群から構成される。既に述べたように、本システムは、筆者らが開発した統合的電力系統解析支援システム⁽¹⁾のプログラムコードを再利用して構築している。図右のDBはOODBMS Versnatを用いて作成している。OODBはデータモデル

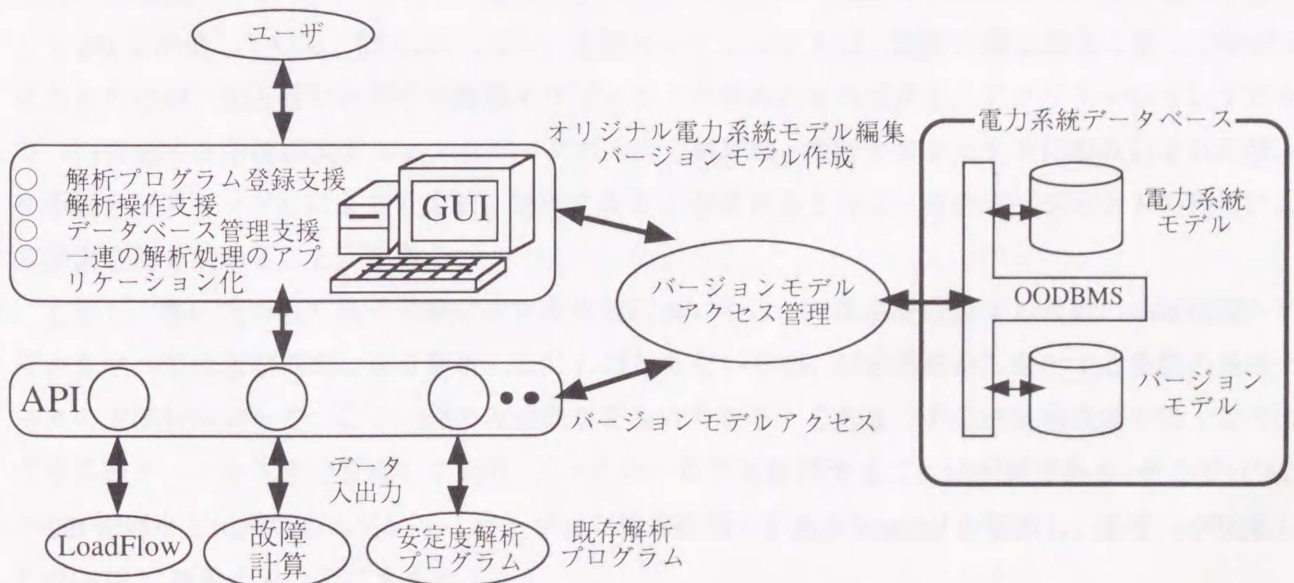


図5.2 電力系統計画支援システムの構成

を構成する機器オブジェクトに、その属性に加えて属性を操作する手続き（メソッド）を組み込むことができるので、非常に柔軟なデータ表現が可能である。このことから、関係DBでは難しかった機器の接続関係（部品構造関係）も効率よく管理できる。

5.2.3 支援機能の説明

○ ユーザインタフェース： 本システムでは、オペレータがGUIとの対話により計画案作成・評価作業を視覚的に操作することを可能にし、かつ理解しにくい数値データをグラフィックス表示することで直感的な理解が得られるように構築されている。また、後述のDBのバージョン管理機能にGUIインタフェースを用いることで、統一的データ管理を行うことができる。

○ 解析プログラムとAPI： 既存の電力系統解析プログラムはそれぞれ独自の入出力フォーマットを有しており、GUIと連携して使用するためには解析プログラム自身を書き換えるか、GUIに専用の入出力プログラムを作成する必要があった。著者らが提案した系統解析支援システム⁽¹¹⁾ではGUIと解析プログラムの間にAPIを設けて、その非互換性を吸収するように設計されている。さらに、このAPIはSmalltalkで記述されているので、オブジェクト指向言語の再利用性を生かし、APIの自動作成が可能となるシステムの提案も行った⁽²⁹⁾。このように、APIを解析プログラム毎に作成し、DBとのデータ交換を担当させることで、任意の解析プログラムをシステムに容易に追加することが可能となっている。

○ 電力系統DB： 著者らが先に構築した電力系統解析支援システム用DB⁽¹¹⁾は、一つの電力系統を複数の解析プログラムにより解析するためのものであり、オブジェクト指向モデリング技術を採用している。これは複雑な階層構造をなす電力系統を都合良くモデリングするためである。したがって、系統を構成する個々の機器を同じレベルのオブジェクトとしてモデル化するのではなく、これらの機器オブジェクトをある程度機能的にまとめたもの（複合オブジェクト）を、基本単位としてDBを構築している。例えば、（複合）変電所オブジェクトは、機器の構成関係に従ってモデル化された母線、変圧器など多くの機器オブジェクトを含めたものである。アプリケーションプログラムの要求する系統構成チェックやデータの入出力処理は、機器オブジェクトに組み込まれた様々な手続き（メソッド）によって柔軟に処理することができると共に、複合オブジェクトの利用により探索空間を狭めることができる。

しかし、本研究で取り扱う系統計画立案作業においては、対象系統に対する複数の系統増設シナリオをすべての考察期間に亘り解析しなければならないため、対象系統から派生する多数の系統データの大部分は同一で、ごく一部のみが異なるものとなる。これまでの系統計画業務では、このような系統データを人手で管理しており、データの一貫性を維持することは困難である。そこで、CADのDB管理などに用いられている「バージョン管理機構」であるVmodelを拡張し、著者らが提案したOODBに適用することにする。

5.3 電力系統データベースのバージョン管理

5.3.1 バージョン管理機能

バージョン管理機能は、CADを用いる設計業務において、多くの設計案を効率的に比較検討することを目的として研究・開発されてきた。この機能を実現するモデルはいくつか発表^(32, 33, 34)されているが、大別するとDBの機能の一部として組み込まれている一体型と、既存のDBに追加できるアドオン型とがある。Versant⁽³³⁾の管理モデルは一体型で融通性に欠け、Odin⁽³⁴⁾はアドオン型であるが、OODB上のインスタンスのみを対象としているためクラスの管理ができないという欠点がある。一方、Vmodelと呼ばれるバージョン管理モデル^(32, 付録A)は、設計・計画システムにおける試行錯誤の支援を行うため、データの履歴管理だけでなく、旧データの再利用に重点を置いて設計されており、複数の増設シナリオを管理する本システムに都合が良い。そこで、本研究ではVmodelを採用することとし、系統計画立案作業に必須の支援機能を追加すると共に、本システムのOODBで用いられている複合オブジェクトに対応できるよう拡張を行う。

5.3.2 Vmodelの複合オブジェクトへの拡張

5.2節で述べたように、複雑な階層構造をなしている電力系統を効率良くモデル化するため、複合オブジェクトによるDBの構築を行ってきた⁽¹⁾。複合オブジェクトとは、部分を表すオブジェクトを全体を表すオブジェクトに集約してひとつのオブジェクトをモデル化する手法であり、集約には入れ子構造が利用される。2章で概説した単一オブジェクトを対象としたVmodel⁽³²⁾では複合オブジェクトで表現された電力系統DBを適切に管理することができない。そこで本研究ではVmodelに以下の2点の機能拡張を行う。

1. OMが複合オブジェクトのバージョン情報を取り扱えるように拡張する。
2. 子オブジェクトからのバージョン情報を親オブジェクトに伝達するためのメソッドを複合オブジェクトのVMに実装する。

本研究で複合オブジェクトを取り扱えるよう拡張したVmodel（拡張Vmodelと略記）の例を図5.3に示す。図中のOM1は複合オブジェクトに対するOMであり、OM2はOM1の子オブジェクトのOMを表わしている。OM2のバージョン情報はOM1のVMであるVM1のメソッドを通してOM2からOM1へ伝達される。図5.3では2階層の複合オブジェクトの例を示しているが、3層以上の場合でも、もちろん適用可能である。

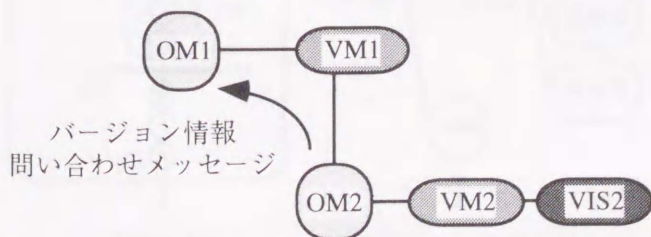


図5.3 バージョン管理

5.3.3 複合オブジェクトと拡張Vmodelによる電力系統データ表現

ある変電所に変圧器バンクが増設された場合のデータ表現を例題として用いる。変電所は複合オブジェクトとして記述されて

おり、さまざまな部品の機器オブジェクトから構成されている。また、機器オブジェクト自身も複合オブジェクトで構成されている場合もある。図5.4 (a) は母線に変圧器バンクが増設された場合の系統図的表現を、図5.4 (b) には変圧器バンクオブジェクトの構成を示している。変圧器バンクオブジェクト自身も複合オブジェクトで表わされており、この図の例では変圧器オブジェクト、遮断器オブジェクト、断路器オブジェクトという3種類の機器オブジェクトから構成されている。変圧器バンクの追加が生じたときの拡張Vmodelによるバージョン管理モデルは図5.4 (c) となる。図中のOM1は変電所オブジェクトX_Stationを管理するOMであり、OM2は変圧器バンクオブジェクトY_TrBankを管理するOMである。前述のように、Y_TrBankも変圧器オブジェクトY_Trans、遮断器オブジェクトY_CB、断路器オブジェクトY_LSから構成されており、変圧器バンクが追加された場合には、それぞれのオブジェクトに対するVM、VISが生成される。これら単一オブジェクトの状態変更は、上位の親オブジェクトのVMにメソッドを用いてバージョン管理情報の伝達が再帰的に行われる。

5.3.4 スナップショットによるバージョンデータ群の管理

前節までに述べた拡張Vmodelを用いることで電力系統DBの個々のオブジェクトの追加・修正に対するバージョン管理が可能となった。しかし、系統計画立案作業では複数のデータ追加・修正をまとめて1つの系統増設計画とするのが一般的であり、その増設計画内のオブジェクト間のバージョン情報には意味がない。このような増設計画を効率良く取り扱うために、複数の差分情報を一つにまとめて、ひとかたまりで取り扱うことのできるスナップショット⁽³²⁾と呼ばれる手法を導入する。

Vmodelのスナップショットはあるモデルを構成する全てのオブジェクト群と使用中のバージョン(VIS)を一つにまとめスナップショットID (SnapID) で管理するというものである。本システムでは複合オブジェクトで電力系統をモデル化しており、入れ子構造の最上位に位置するオブジェクト

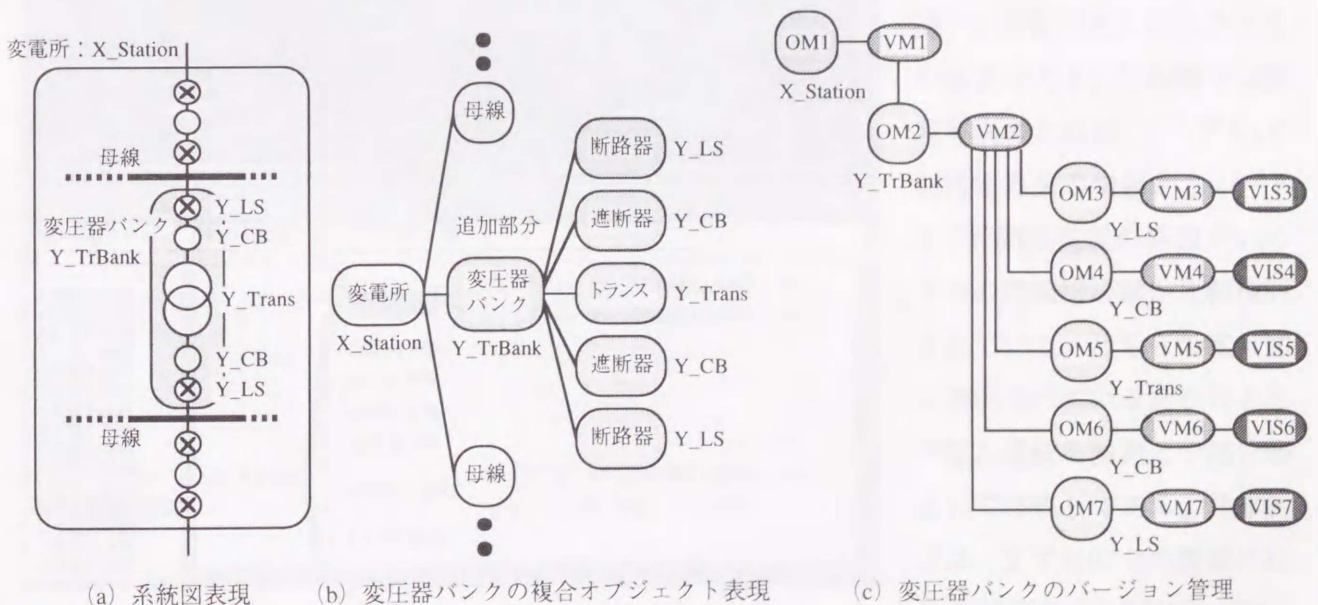


図5.4 バージョン管理例

のIDがあれば電力系統モデル構成オブジェクト全てを参照できる。そこで拡張Vmodelではスナップショットを電力系統モデルのID（最上位のオブジェクトのID）と使用中のVISをSnapIDでまとめることで実現している。このスナップショットを用いて差分情報（現在使用中の機器オブジェクトのバージョン情報）をあたかも一つの電力系統として取り扱える。

5.4 電力系統計画支援システムの実行例

まず最初に本システムのバージョン管理機能を利用した電力系統モデル管理の例を紹介する。図5.5下側のウィンドウは現在DBに登録されている電力系統モデル（以下、電力系統と略記）を参照するためのツール（バージョンView）である。同ウィンドウ左端のリストは、オリジナルの電力系統を一覧表示しており、現在C北電力系統が選択されている。同ウィンドウ右側には、選択されたC北電力系統から派生した電力系統がツリー表示されている。このツリーは、5.3節で説明した

スナップショットを利用して管理されている電力系統を表示したものである。このツリーの中で年号の記されているものは、各年度の予測需要を適用した電力系統である。図5.5中央の系統図は、2003年度系統で、ノード6（KY変電所）の負荷増加とともない送電線（ノード6-5間）が容量不足に陥ったときの様子である。この例では問題解決のために、ノード6-5間送電容量の増強と、ノード1-6間送電線の新設という2つの設備増設案が比較検討されている。どちらの案も図に表示されている2003年の電力系統を画面上で部分修正して作成してある。具体的には、まず画面上の機器に対して修正が施される度に修正

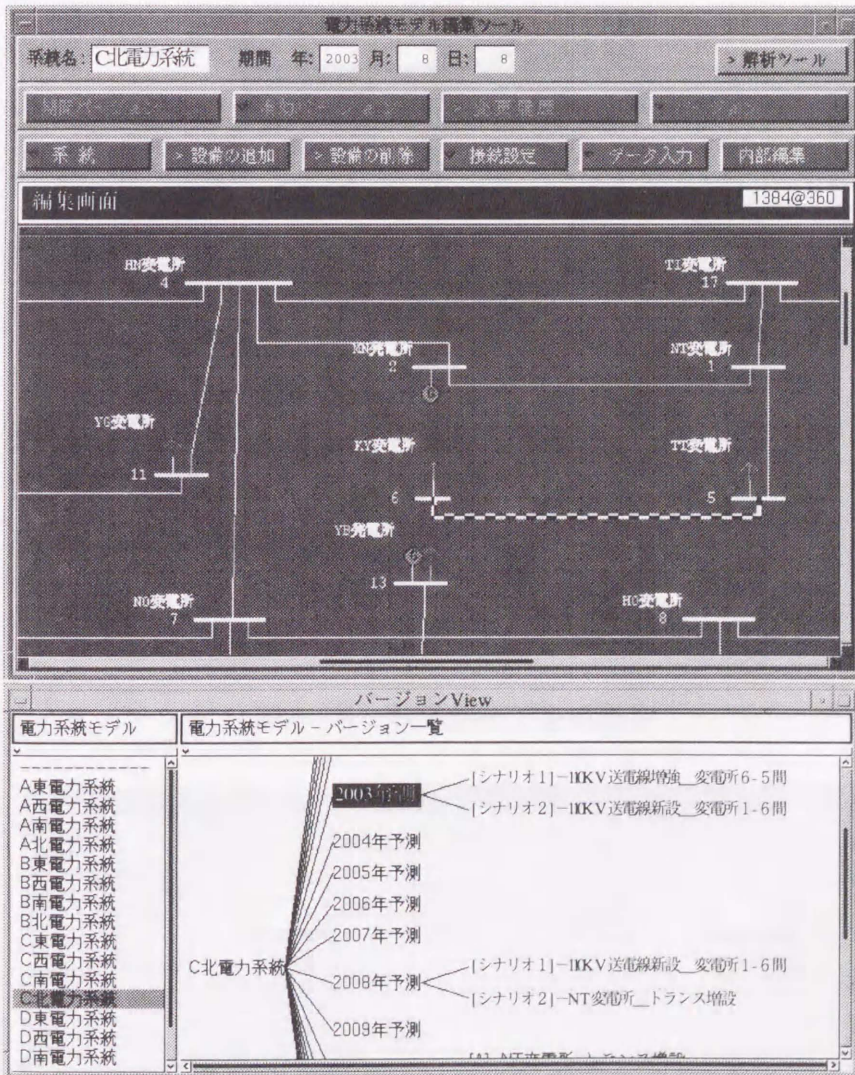


図5.5 電力系統モデルのバージョン管理

内容を保存した新たな機器のバージョンを5.3節で紹介したバージョン管理モデルを使用して作成してゆく。そして最後に作成した機器のバージョンをスナップショットを利用して一つにまとめ、2003年度系統から派生した電力系統（設備増設案）を作成してある。バージョンView上には、新たに追加された電力系統が、図のように派生元となった2003年度系統の右側に表示され、導出関係が一目でわかるようになっている。以上のような手順で準備された電力系統を解析してゆくために、本システムには以下に挙げるような幾つかの機能が用意されている。

本システムは、図5.6のように発電機の制御器モデルをユーザが自由に編集する機能を備え、デフォルトの制御器モデル以外を解析に使用することができる。編集作業は、図上側にあるブロック記号のパレットから必要な記号をマウスで選択し、図下側のブロック図編集画面に配置・接続することで行う。ブロック記号には、そのブロックに定義されているパラメータを設定するためのダイアログウィンドウが用意されており、ユーザが値を自由に設定できる。ブロック図編集画面左隅にある2つのリストは既にDBに登録されている制御器モデルを編集画面上に呼び出すために使用する。

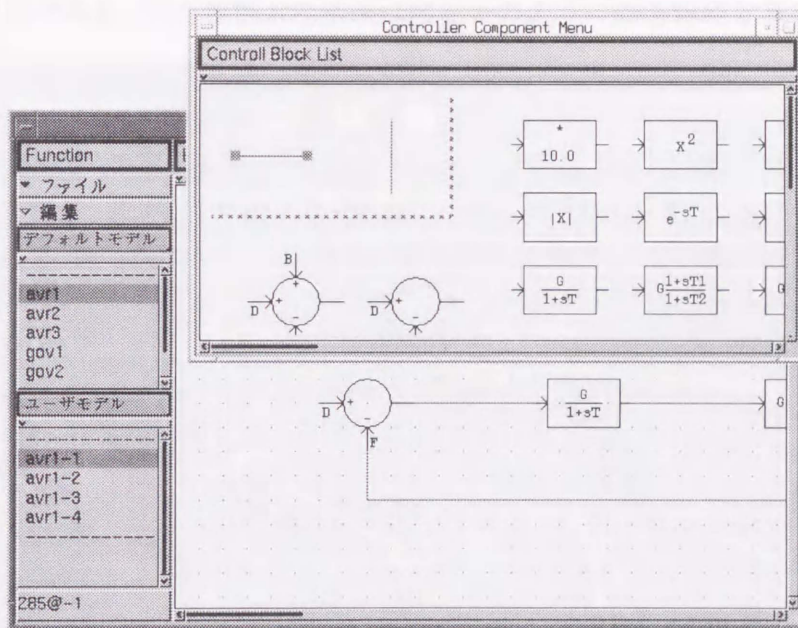


図5.6 制御器モデルの編集

上側はデフォルトの制御器モデルの一覧を表示するためのリストである。このリストからモデルを選択すると下側のリストに、デフォルトモデルから派生したユーザモデルの一覧が表示される。ユーザモデルの一つをマウスで選べば、画面上に対応する制御器モデルが表示される。ユーザはこの制御器モデルに修正を加えてそのまま保存することもできるし、修正後のモデルに新たに名前を付けて別のユーザモデルとして保存することもできる。

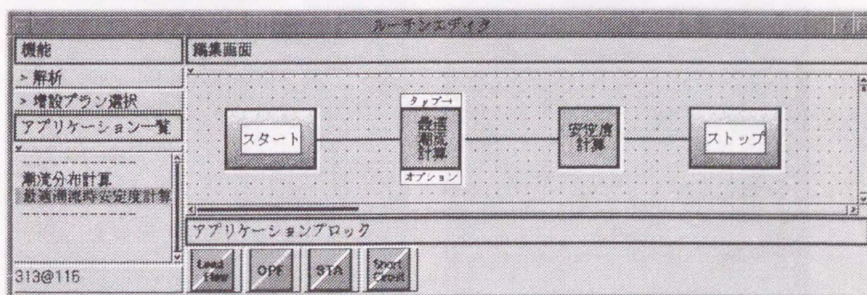


図5.7 制御器モデルの編集

解析作業は、ユーザが逐次解析プログラムを選択して実行してゆくほかに、図5.7の"ルーチンエディタ"を使って一連の解析作業をバッチ処理

することができる。一連の解析作業の登録は、「ルーチンエディタ」上でアイコンパレットから選んだアプリケーションブロックをマウスでウインドウ上に配置し、互いに結び付けることで容易に実現できる。図の例は最適潮流計算実行後に安定度解析を行うという操作を一つのアプリケーション「最適潮流時安定度計算」として登録した様子を示している。このアプリケーションには、単純なバッチ処理だけでなく前段の解析結果を次段の入力データとして利用する解析や各ブロックで機器のパラメータを変化させる解析が実行可能である。こうした処理が必要な場合は、ブロックをマウスで選択してオプションウインドウを開いて設定を行う。設定を行うと標準のアイコンが図のようなオプション指定されているものに変更され、ユーザにオプション設定の有無が一目でわかるようになっている。

以上のような機能を利用して得た解析結果はこれまでに構築してきた表示画面を利用して、数値データをグラフィックス表示することができる。図5.8の例は、線路潮流の方向を矢印、大きさを線路の太さで図示した様子である。ノード電圧や位相角についても棒グラフや円グラフの形式で表示可能である。固有値解析の結果は図左下のように固有値の分布状態をグラフ上にプロットして確認

できる。本システムは、1時間毎24時間分の負荷と発電機出力が与えられれば、各時間断面を自動解析し、ノード電圧の時間変化等を図右下のような折れ線グラフで表示することもできる。

5.5 結言

本章において提案した電力系統計画支援システムは、複雑な電力系統計画を支援するためのもので、先に提案した統合的電力系統解析支援システムを拡張したものである。今回提案したシステムにおいては、DBの一貫性を保持し、かつ計画代替案に含まれる元系統と同一のデータを繰り返して登録することを避けた

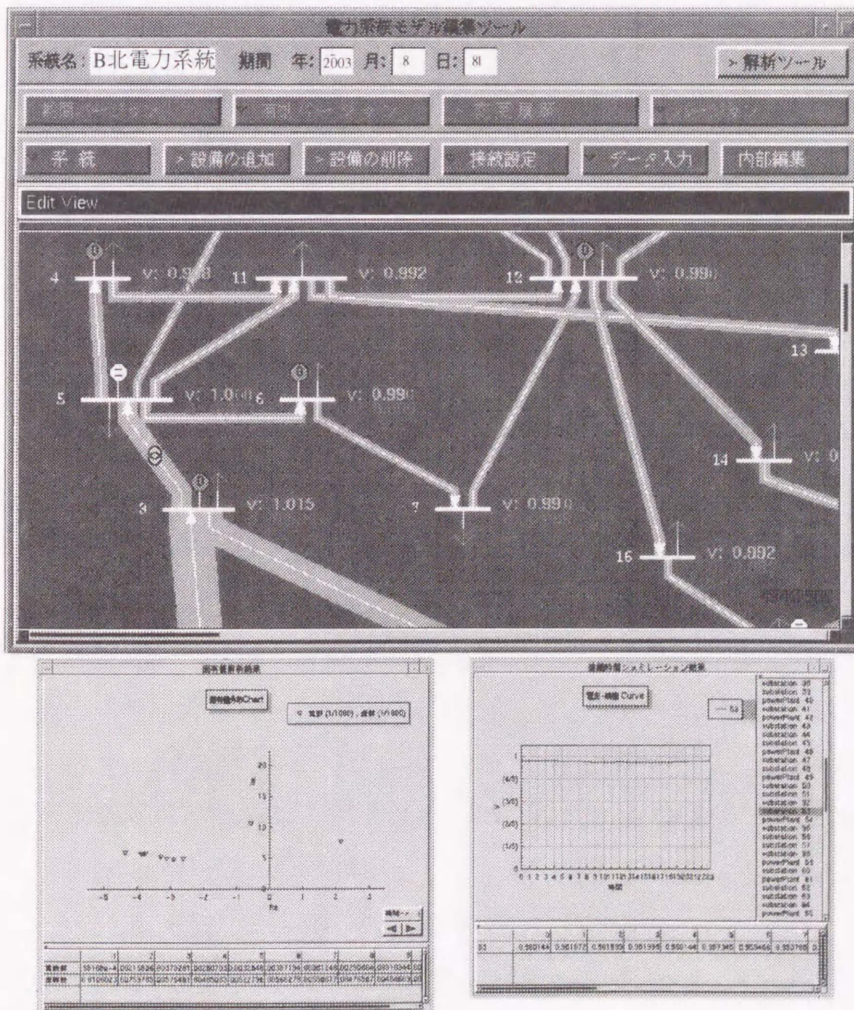


図5.8 電力系統解析実行例

め、代替案のデータは元システムデータとの差分のみを対象として登録する方法（バージョン管理）を採用した。このバージョン管理は、CADなどの設計DBで差分情報の履歴管理を行うことで、この機能を組み込んだシステムがすでに幾つか発表されている。著者らはそれらの中でも、バージョン管理機能がDB管理システムとは独立しているため、機能自体をプログラミング環境や要求事項の変化に合わせて容易にカスタマイズできる等の理由から、Vmodelと呼ばれるシステムに注目した。ただこのVmodelは著者らが構築してきた電力系統DBで取り扱う複合オブジェクトをバージョン管理できないので、新たに複合オブジェクトを取り扱うことができる拡張Vmodelを提案し、電力系統計画支援システムに組み込んだ。さらに制御機器を自由にユーザが編集・管理する機能や一連の解析作業をアプリケーション化して登録管理する機能を組み込み、計画立案者が複数の電力系統を詳細かつ容易に解析できるようにした。本章で提案したシステムは一応初期の目的を達成しているが、電源計画プログラムなどを組み込むことで本システムの有用性はさらに高まると思われる。

第6章 結 論

電力系統の解析支援を目的とした支援システムを提案した。1章では、背景と電力系統解析の問題点と問題解決へ向けた方針を示した。

以下、各章毎に内容と得られた結果を要約する。

(第2章) 統合的電力系統解析支援システムの概要

本稿の2章では、操作方法、データファイルフォーマットの違いに関係なく様々な解析プログラムを使用できる電力系統解析とデータベース管理のための新しい支援システムが提案された。このシステムの解析支援インタフェースは、グラフィカルユーザインタフェース (GUI) を利用して既存の解析プログラムへのデータ入力、プログラムの実行、解析結果の表示を支援できるようになっている。また既存解析プログラムを有効利用するためにAPIを設け、インタフェースとデータベース間のデータ交換が一括管理されている。こうして解析プログラム毎に異なる入力フォーマットと操作方法の違いに対処し、これまで開発に多くの労力、時間、コストを費やしてきた既存のプログラムの有効利用を図った。これらのインタフェースとアプリケーションプログラムインタフェース (API) のプログラミングには、オブジェクト指向言語を採用し、開発と保守が効率よく行えるようにした。解析プログラムの入力データ管理には、電力系統のもつ階層構造のモデル化に適し、かつオブジェクト指向言語との相性のよいオブジェクト指向データベースを採用した。

(第3章) 解析支援環境の高度化の概要

3章では、解析プログラム登録支援機能と新たな解析結果表示機能を追加した解析支援システムを提案した。解析プログラム登録支援機能は、解析プログラムのデータ入出力に関する仕様情報(属性、入出力フォーマット)をインタラクティブに入力するだけでAPIを作成するものである。仕様情報の入力には、電力系統の持つ階層構造に類似した階層型のメニューを採用することで、システムを構成する機器が持つ属性をユーザが容易に検索・入力できるようにした。また連続的に運用値を変えた解析を実行、結果表示する機能を追加し、システムの挙動をより簡単に視覚で確認することができるようになった。これらの機能の追加は、システムの開発にオブジェクト指向プログラムを採用したことで、プログラムの差分を追加ないし再定義するだけで実現できた。既存プログラムの多くは、そのまま再利用することができたので、システムの開発にそれほど時間はかからなかった。

(第4章) 電力系統図編集作業の自動化の概要

本章では、電力系統単線結線図の自動描画を目的として、VLSIの自動配線に使用されている混雑度を基本とし、線路の交差が少ない系統図が得られる手法を提案した。本システムは、電力系統図

のようにノードの配置がある程度固定されているネットワークの表示には十分有用である。また、格子グラフ上に埋め込まれている系統図のブランチの経路をラバーバンドとスポークのアルゴリズムを用いて整形することにより、系統図がさらに見やすいものとなることが確認できた。

(第5章) 電力系統計画立案支援環境の概要

5章では、電力系統計画作業支援システムを提案した。本システムは、基礎解析プログラムの操作支援を目的とした解析支援システムとは異なり、計画面作成のために頻繁に行われ代替案の作成と管理を行う機能を備えている。この代替案の管理は、オブジェクト指向データベースに電力系統計画用にカスタマイズしたバージョン管理機構を組み込むことで実現している。このバージョン管理機構を導入することで、代替案のデータは元系統データとの差分をあたかも一つの電力系統モデルとして取り扱うことができる。データベースの一貫性を保持し、かつ計画代替案に含まれる元系統と同一のデータを繰り返し登録することを避けることができる。さらに制御機器を自由にユーザが編集・管理する機能や一連の解析作業をアプリケーション化して登録管理する機能を組み込み、計画立案者が複数の電力系統を詳細かつ容易に解析することもできる。

本論文では、既存の解析プログラムを有効に活用しながらインタラクティブな解析作業環境を実現する検討を行ないある程度の成果を収めることができたと考えているが、不十分と考えられる部分も少なくない。今後の課題として研究が必要であると考えられる事項を以下に示す。

(1) 電力系統図自動表示システムについて

- (i) 格子グラフ上に埋め込まれているブランチの経路をすべてラバーバンドを使って変換しているが、ブランチによってはラバーバンドによる変換を行わないほうがかえって見やすい場合もあるので、もとの経路の折れ曲がりの数などによる基準を導入してブランチによってはラバーバンドによる変換を行わないようにする。
- (ii) ノードの配置位置が可変である場合のブランチ交差数の最小化を目的としたシステムの開発を行う。
- (iii) 系統図表示システムを使って、系統図上でデータの入力、潮流計算などの系統解析を対話的に行うことのできるシステムを開発する。

(2) 電力系統解析支援システムについて

本論文で提案したシステムは一応所期の目的を達成しているが、改良点の一つとして制御機器を自由にユーザが編集・管理する機能を備えることが挙げられる。具体的には、MATLABやEMTPのTACSのように、ユーザが各種の制御要素を自由に組み込めるようにすれば、本システムの有用性は

非常に高まると思われる。

以上、本論文で示した電力系統解析支援システムに関する研究は、基礎的な電力系統解析プログラムの操作支援に対して新しい知見を提示することができたと考えている。本論文の内容は、電力系統解析作業全体からすれば狭い範囲をカバーするにすぎないが、電力系統解析支援システムの発展に対して一助となれば幸いである。

謝 辞

本研究は、広島大学工学部 佐々木博司教授の御指導のもとに遂行されたものであり、終始時宜にかなった御教示と御鞭撻を賜りました。ここに深甚なる感謝の意を表わす次第であります。

本研究を進めるにあたり有益な御討論と御助言をいただきました広島大学工学部 山下英生教授、尾崎俊治教授、中村信人教授に厚く御礼申し上げます。

また本研究遂行中に終始御激励を賜りました広島大学工学部 餘利野直人助教授、久保川淳司講師、造賀芳文助手、輝平盛重技官、福山大学工学部 川原耕治助手に深く感謝の意を表わします。

また、広島電機大学 葉佐井博巳電気工学科主任教授をはじめ、立川清兵衛教授、岡野繁雄助教授、そのほかの教職員の皆様には、終始変わらぬ暖かい御支援と御協力を賜りました。厚く御礼申し上げます。

更に、本研究を行う上で多大なる便宜と貴重な援助を頂きました 中国電力株式会社 工務部 水津卓也様をはじめ関係各位に対して、心から感謝いたします。

最後に、本論文の作成にあたってお世話になった広島大学工学部電力研究室の研究室の方々に変にお世話になりました。厚く御礼申し上げます。

参 考 文 献

- (1) B.C.Schwab, K.Hemmaplardh, S.A.Sackett, "A Developed Database Management System for Operation and Planning Applications", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-104, No. 7, July 1985
- (2) B.C.Schwab, K.Hemmaplardh, S.A.Sackett "Software Features of a DBMS Kernel in Large Real-Time EMS Systems", IEEE Transactions on Power Systems, vol. 3, No. 1, February 1988
- (3) 小海, 谷津, 真鳥:「オープンシステムを用いた分散形電力系統解析システム」, 平成4年電気学会全国大会, p. 9-266
- (4) 林, 浅野, 中村:「ワークステーションを用いた系統解析支援システムの開発」, 平成5年電気学会全国大会, p. 9-168
- (5) 佐々木, 歌谷, 久保川:「電力系統解析支援用統合環境の開発－III」, 平成5年電気学会全国大会, pp. 9-171~9-172
- (6) 佐々木, 歌谷, 久保川:「統合的電力系統解析支援システムの構築」, 平成5年電気学会電力・エネルギー部門大会, pp. 136~137
- (7) 佐々木, 歌谷, 久保川:「電力系統解析支援用統合環境の開発－IV」, 平成5年電気学会電力技術研究会, PE-93-17
- (8) 河村一樹:「データベース要論－関係データベースとオブジェクト指向データベース－」, 啓学出版, 1994年2月
- (9) 小林:「オブジェクト指向と Smalltalk」, CQ 出版, 1990年2月
- (10) 本位田, 山城:「オブジェクト指向分析・設計」, 情報処理学会雑誌, Vol. 35, No. 5, 1994年2月
- (11) 佐々木, 歌谷, 久保川:「統合的電力系統解析支援システムの構築」, 電気学会論文誌B, 115, pp.734~740, (平7-7)
- (12) 佐々木, 歌谷, 久保川:「解析プログラム登録支援ツールの開発」, 平成7年電気学会電力技術研究会, PE-95-173
- (13) 佐々木, 歌谷, 竹田:「電力系統自動表示システムの開発」, 電気学会論文誌B, 116, pp.35~41, (平8-1)
- (14) 今村, 横島, 磯野:「会話型系統解析のための系統図作画プログラム」, 昭和56年電気学会全国大会, p.1123
- (15) 鈴木, 土井, 阿部, 細島, 神津:「電力系統解析用系統図作成プログラムの開発」, 昭和58年電気学会全国大会, p.1238
- (16) 佐藤, 猪俣, 相原:「電力系統図作成システムの開発と運用」, 昭和62年電気学会全国

- 大会, pp.1212~1213
- (17) 田中, 金輪, 庄司:「グラフィック CRT を用いた配電系統図高速表示システムの開発」, 昭和 62 年電学論 D, Vol. 107, No. 4, pp.479~486
- (18) 辻, 忠鉢, 吉田, 飯島, 渡辺, 池田:「配電線路自動作成システムの開発」, 昭和 61 年電気学会全国大会, pp.1408~1409
- (19) 伊庭, 川崎, 熊野:「フラクタル木表現を用いた電力系統の系統図自動作画手法」電学論 C 94 年 2 月号
- (20) 小野口, 千葉, 西関:「平面グラフの描画アルゴリズム」, 信学技報 CAS 83-105, pp.73~78
- (21) 西口, 白川, 尾崎:「平面グラフの自動描画プログラムについて」, 信学技報 CAS 79-136, pp.23~28
- (22) 小澤, 関口:「平面グラフの一描写法とその応用」, 信学技報 CAS 80-48, pp.33~40
- (23) 汪, 宮尾, 菊野, 吉田:「構成規則に基づく直並列グラフの平面描画について」, 信学技報 COMP 87-49, pp.43~50
- (24) Sugiyama, et. al., "Methods for Visual Understanding of Hierarchical System Structures", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-11, No. 2, February, 1981, pp.109-125
- (25) Watanabe, et. al., "A Congestion -Cost- Directed Router for VLSI Switchboxes", *Proc. IEEE ISCAS*, 1990, pp.1684-1687
- (26) Iwamoto, et. al., "PRIDE : A Printed Writing Board Designing System for Analog Circuit by Graph-Planarization and Rectangler-Dualization", 情報研報, DA 58-4, 1991, pp.1-8
- (27) Charles E. Leiserson, et. al., "Algorithms for Routing and Testing Routability of Planar VLSI Layouts", in Proc. of the 17th Annual ACM Symposium on Theory of Computing, 1985, pp. 69-78
- (28) Mike Foley, Anjan Bose, "An Object Based Graphical User Interface for Power Systems", IEEE Transactions on Power Systems, Vol. 8, No. 1, pp. 97-104, February 1993
- (29) M. Utatani, et al, "An Integrated Support System for Power System Analyses with Improved Application Program Interface(API)," ICEE 95, Taejon, Korea, July 1995
- (30) 宮田秀介:「電力系統の計画と運用」, 電気書院, 1970 年 2 月 20 日
- (31) 関根泰次:「電力系統工学」, 電気書院, 1976 年 6 月 20 日
- (32) 吉沢直美, 小野美由紀, 石川博, 「オブジェクト指向データベースにおけるバージョン管理モデルの設計と実装」, 情報処理学会論文誌, Vol.37, No.4, Apr, 1996, pp556~567
- (33) Versant Object Database Management System Release 1.6, 「システムリファレンスマニュアル」, Versant Object Technology Corporation, 1991

-
-
- (34) 木村, 鶴岡:「オブジェクト指向データベース管理システム Odin の機能と構成」, 日経 AI 別冊 1991 夏号, 1991
 - (35) J. ランボー, M. ブラハ:「オブジェクト指向方法論 OMT モデル化と設計」, トップアン, 1994 年 7 月
 - (36) Chou,H.T.and Kim,W., "Version and Change Notification in an Object-Oriented Database System", Proc.25th ACM/IEEE Design Automation Conference, pp275-281, 1988
 - (37) Ahmed, R. and Navathe, S.B., "Version Mangemant of Composite Objects in CAD Data-base", Proc. ACM SIGMOD, Denver, pp218-227, 1991
 - (38) 北川博之, 田中肇他, 「履歴データを用いた版管理データモデルの提案」, 情報処理学会論文誌, Vol.34, No.5, Apr, 1993, pp1031~1044

付 録

A. オブジェクト指向の概要

従来の手続き型言語でのソフトウェア開発には、いくつかの問題点があった。一つは、手続き（機能）がコンピュータ上での実現単位であることから、ソフトウェアの分析・設計が機能中心になることである。実世界の問題は、機能よりもモノ中心にとらえる方が自然であるため、現実の問題を手続き型の言語で記述するのは、困難な作業となる。また既存の手続きやデータ構造の拡張に手間がかかることが挙げられる。例えば、プログラムの一部を修正する場合、プログラムのコードをコピーし、コピーに対して修正・追加を行うため、類似したコードを持つプログラムが複数作成されることになる。このとき、ベースとなったコードにバグがあるとそのバグが他のコードにそのままコピーされることになり、いくつものプログラムを修正しなくてはならなくなる。こうした問題を解決する新しいソフトウェア開発の方法として考案されたのがオブジェクト指向アプローチである(9, 10, 35)。

本章では、本研究で提案する解析支援システムの設計・開発に使用したオブジェクト指向の概念、オブジェクト指向システムの分析・設計手法からオブジェクト指向プログラミング、オブジェクト指向データベース（OODB）、オブジェクトのバージョン管理に言及する。

A. 1 オブジェクト指向プログラミングの基礎⁽⁹⁾

現実の世界は、様々なモノによって構成されている。これらのモノは、それぞれ独自の役割を持ち、自分でできることは独立して作業を進めるが、複雑な作業は複数のモノが互いに作業を分担して実現されている。そこで、現実の世界のモノ（＝オブジェクト）とモノどうしのつながり（＝関係）に着目し、個々のオブジェクトに作業を割り振り、オブジェクトどおしがお互いに作業を依頼しながら機能するようにしたのが、オブジェクト指向アプローチである。すなわちオブジェクト指向とは、現実のモノどおしの関係をそのままソフトウェアで記述することによって、現実世界の仕組みをコンピュータ上で再現（モデル化）するアプローチである。現実類似した形でシステムをモデル化できることが、システムの構造をより直感的で分かりやすくし、プログラムの生産性の向上等に役立つ。本節ではオブジェクト指向プログラミングで使用される用語の簡単な説明を行う。

A. 1.1 オブジェクトとインスタンス

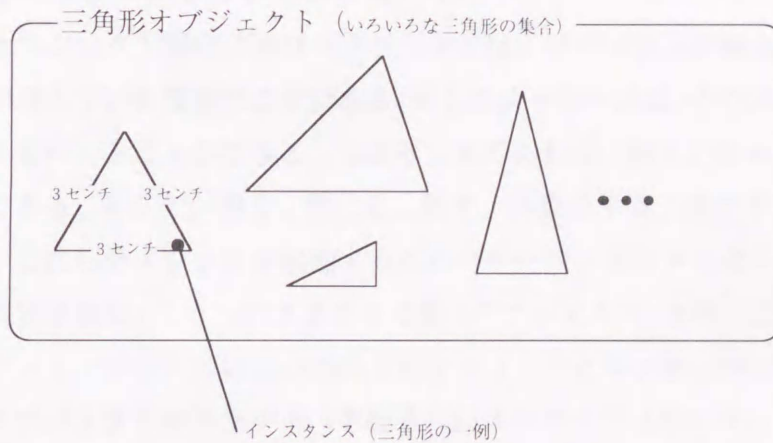
オブジェクト指向におけるオブジェクトとは同じ性質をもつものの集合を意味している。そしてその集合を構成する各々の要素をインスタンスと呼ぶ。この関係を三角形を例に挙げて説明する。三角形にはさまざまな大きさのものがある。例えば、図A.1のような一辺3センチあるいは5センチの正三角形などである。この一辺3センチの正三角形は、三角形というグループのなかの一例であ

る。オブジェクト指向では、こうした一例をインスタンスと呼ぶのである。それに対して、三辺で囲まれた図形という性質を持ったインスタンスの集合をオブジェクト（この場合は三角形オブジェクト）と呼ぶ。オブジェクトとはインスタンスを抽象化した概念であるといえる。抽象化とは図A. 2のようにある物や事象を重要な側面に注目して分類することである。

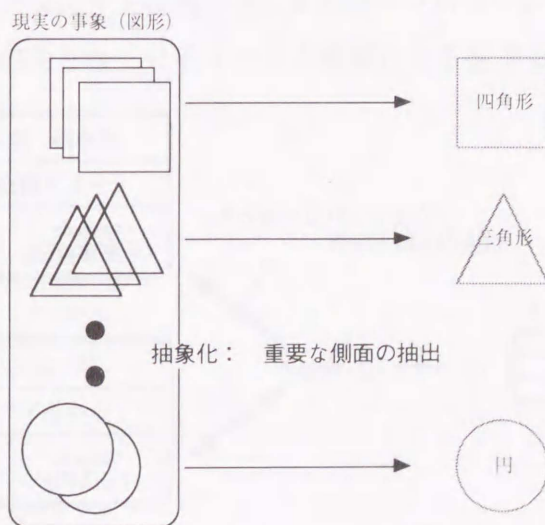
オブジェクト指向プログラミングではこうしたオブジェクトを基本単位としてプログラミングが行われる。すなわち問題領域のオブジェクトをコンピュータ上にモデル化し、それらオブジェクトの相互作用によって問題解決が行われるのである。その結果、現実のものや概念を忠実にコンピュータ上にモデル化することができる。したがってオブジェクト指向プログラミングにおいては、問題領域にどのようなオブジェクトがあり、それらがどのような役割を演じているかを特定することが重要である。この問題領域の分析方法をA. 2節で紹介する。

A. 1. 2 クラスとは

前項で述べたオブジェクトを実現するためにクラスと言う概念が必要となる。このクラスは、オブジェクトのインスタンス



図A. 1 オブジェクトとインスタンス



図A. 2 抽象化

のテンプレートとなるもので、オブジェクトが共通して備えている属性、振る舞い（メソッド：次項で紹介する。）が集められる。ここで属性とは、オブジェクトによって保持されるデータのことである。例えば人オブジェクトを考えてみる。人が備えている共通の属性としては名前、年齢、体重などがある。人オブジェクトのクラスには、名前、年齢、体重などの格納場所、属性が定義されることになる。この人クラスの属性に特定の値が設定されると、識別可能な人オブジェクトのインスタンスが誕生することになる。一方振る舞い

とは、オブジェクトに固有な手続きのことで、属性値を処理する手続きなどがこれに相当する。オブジェクト指向プログラミングの大半はこのクラスの定義に費やされることになる。

このクラスによって生み出される特徴の一つに情報隠蔽（カプセル化）がある。先にオブジェクトには、クラス定義を利用して属性と手続きが同時に組み込まれることを述べた。適切にモデル化されたオブジェクトでは、このクラス定義のしくみが次のような利点を生み出す。それは、オブジェクトの外部からは直接オブジェクトの内部を見ることも操作することもできないようにできるのである。つまり限られたインタフェース（メソッドとメッセージ：次項で紹介する）を通してしかオブジェクトへのアクセスが行えないのである。一見不便そうであるが、この仕組みにはオブジェクトに定義したメソッドの修正の際に、その影響が他のオブジェクトのメソッドに波及するのを防止する効果がある。なぜなら、あるオブジェクト内部のメソッドにどのような変更があっても、インタフェース部分（データの受け渡しの方法）が変わらなければ、そのメソッドを呼び出す別のオブジェクト側のメソッドに影響を及ぼさないからである。

A.1.3 メソッドとメッセージ

オブジェクト指向におけるメソッドとはオブジェクトの振る舞いを指す。つまりそのオブジェクトが備えている機能のことである。そしてメッセージは、そのメソッドを実行させるためのキーワードととらえることができる。つまり、人であれば、歩く、話す、聞く、が人オブジェクトのメソッドである。そして、開く、閉じる、隠す、再表示する、はウインドーオブジェクトのメソッドである。これらのメソッドを起動するためのキーワードがメッセージである。

現実事象は、こうしたさまざまな物（オブジェクト）が相互に作用しあってひきおこされている。オブジェクト指向ではこの処理方法をメソッドとメッセージで模擬しているのである。つまり、あるオブジェクトのメソッド（手続き）にメッセージ（キーワード）を送ると、処理の結果に応じて次の処理を別のオブジェクトに送るという作業を繰り返すことで、最終的な結果を得るのである。このことをオブジェクト指向では、メッセージパッシングとよぶ。図A.3はメッセージによってオブジェクトに定義されているメソッドを実行する様子を示している。

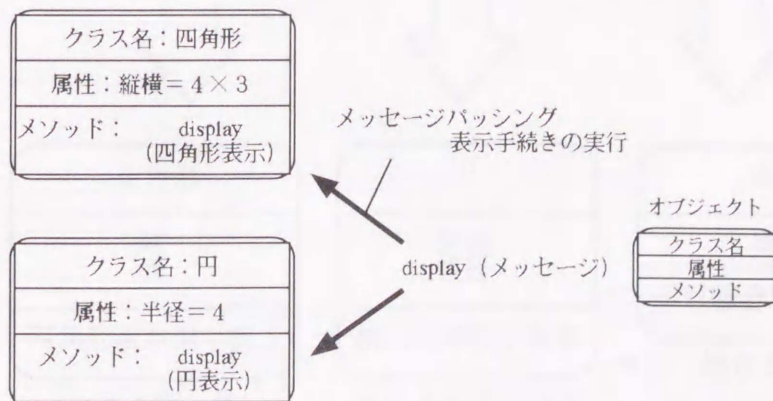


図 A.3 オブジェクトとメッセージ

A.1.4 リンクと関連

現実世界において物は自分以外のものと何らかの関係を持っている。オブジェクト指向ではこの物（オブジェクト）とおしりの関係をリンクと関連で表現する。まずリンクとはインスタンス間の物理的・概念的な結合関係のことである。例えば、「鈴木は広

島大学で勉強する」などがリンクに相当する。一方、関連とは共通の構造と意味を持ったグループ（クラス）間の結合関係のことである。例えば、「人が大学で勉強する」などが関連に相当する。この二つの例からリンクとは関連の一例（インスタンス）であることがわかる。この関連には、いま例で紹介したような単純なオブジェクト間の従属関係を表すものの他に以下のようなものもある。

関連のクラスとしてのモデル化 (is-a関係)： 通常我々は、事象や物を分類しグループ化することで理解を深めようとする。例えば図形であれば、多角形のグループに属する四角形や三角形、楕円に属する円と言った分類である。すなわち四角形や三角形は多角形の特殊なものであり、円は楕円の特殊なものであるという考えに基づいた分類である。オブジェクト指向ではこうした抽象的なものから具体的な物へという階層関係を is-a 関係と呼び、クラス階層を構築する際に使用している。この関係はオブジェクト指向プログラミングの特徴である差分プログラミングを実現するために重要な役割を果たしている。差分プログラミングと継承の関係を図 A. 4 に示す。図はクラス階層を通して属性とメソッドがどのように継承されるかを示している。図の例では長方形クラスには縦横二辺の長さを格納する属性と描画のためのメソッドが定義されている。このとき長方形クラスの特別なクラス（サブクラス）として正方形クラスを図のように定義する。すると正方形クラスは属性やメソッドを定義しなくても図下側のようにスーパークラスである長方形で定義した属性とメソッドを持つクラスとして振る舞う。すなわちサブクラスには、スーパークラスの属性とメソッドが継承

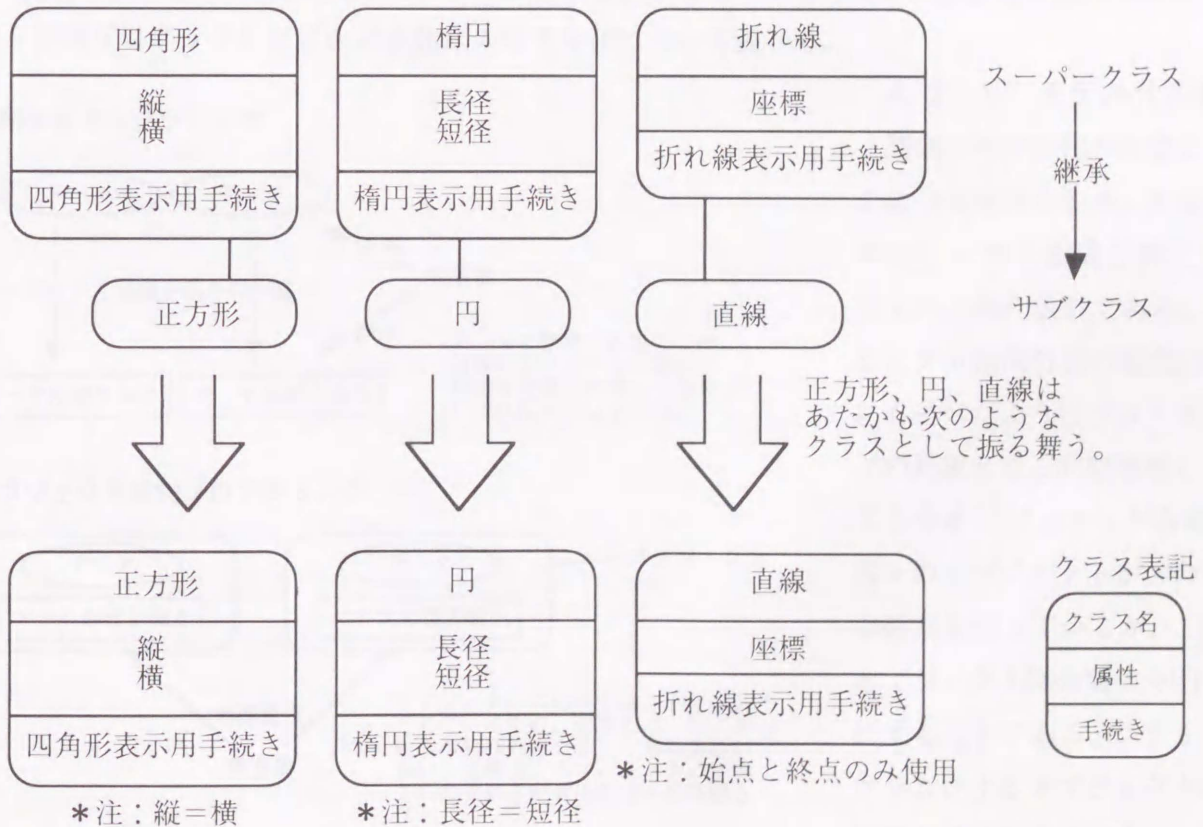


図 A. 4 差分プログラミングと継承

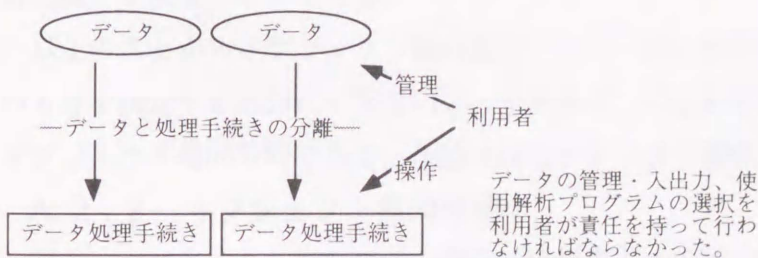
されるのである。従ってサブクラスは、スーパークラスに定義してないものだけを付け加えてゆくことでプログラミングしてゆくことができる。is-a関係に基づいてクラス間の関連をモデル化することはプログラミングの負担軽減に大きく役立つ。

集約 (part-of関係)： 現実の物は、通常いくつもの部品の組み合わせで形作られている。例えば、自動車というオブジェクトは、ホイール、エンジン、ボディーなどの部品オブジェクトの集合である。つまり、ホイールは、自動車の一部であり、エンジン、ボディーも同様である。この部品構造関係をオブジェクト指向ではpart-ofの関係と呼び、先に挙げたオブジェクト間の従属関係とは区別している。部品構造関係をもつ対象を忠実にモデル化する際に役立つ概念である。

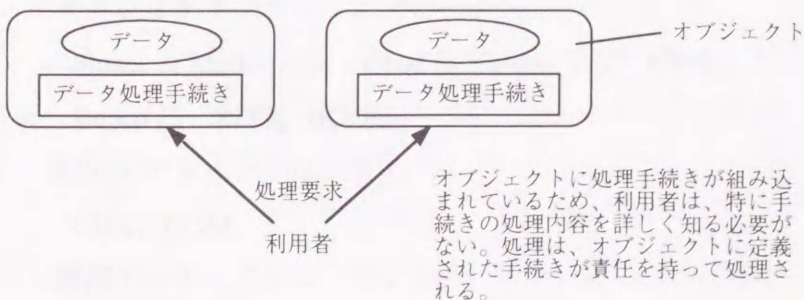
A.2 オブジェクト指向プログラミングについて⁽¹⁰⁾

プログラミングとは、ある問題の解決を目的として事象をモデル化することである。これまでは手続き型のプログラミング言語が広く使われてきた。この手続き型言語は、事象を手続きに注目してモデル化し問題解決を図っている。この手続き型の言語は、モジュール化が難しく、あるモジュールの一部を変更するとその影響が他のモジュールに波及しやすい欠点がある。一方オブジェクト指向プログラミングでは、事象を手続きではなく物に注目してモデル化を実行する。つまりもともと現実の世界でモジュールとして捉えられているものをそのままオブジェクトとしてモジュール化し、これらオブジェクトの相互作用によって問題を解決するのである(図A.5参照)。本節ではオブジェクト指向プログラミングにおける問題のモデル化について述べる。

手続き型プログラミング



オブジェクト指向プログラミング



A.2.1 モデル化の概要

問題のモデル化の作業は大きく2つに分けられる。すなわちオブジェクト指向分析とオブジェクト指向設計である。オブジェクト指向分析の目的は、先にも述べたようにプログラミングの対象となる問題領域にどのようなオブジェクトが存在し、各々のオブジェクトがどのような役割を担っているか、これらオブジェクト間の関係を明らかにすることである。つまりプログラム化するオブジェクトの仕様書を作成するのである。この

図A.5 手続き型プログラミングとオブジェクト指向プログラミングの違い

仕様書の出来（分析の出来）は、システムの性能や拡張性に大きく影響してくるので、注意深く行う必要がある。分析の方法はいくつか発表されているが、どの手法も共通してプログラマが問題領域を正確に把握するために、次のような3つの側面を分析するようになっている。

1. 対象の静的な構造の分析
2. 対象の動的な振る舞いの分析
3. 対象におけるデータの流れの分析

1.では、モデル化する問題領域に含まれるオブジェクトとそのオブジェクトの内部構造やオブジェクト間の関係を把握する。2.では、1.で抽出されたオブジェクトにどのような状態があり、その状態が何によってどのように変化するかを把握する。3.では、オブジェクト内部でデータがどのような操作を受けるかを把握する。これら3つの項目のどの部分に重点を置いて分析を進めるかによって、作成されるモデルの性質や能力が大きくかわってくる。実際に重点の置き方の異なるいくつかのオブジェクト指向分析の手法が提案されている。

一方オブジェクト指向設計は、分析フェーズで得られた仕様情報をもとにプログラム化の具体的な方法を決定し、プログラミング・フェーズに渡すことを目的としている。したがってオブジェクト指向分析とプログラミング作業の中間に位置する。オブジェクト指向設計は論理設計（あるいは基本設計）および物理設計（あるいは詳細設計）の二つから構成されている。論理設計フェーズでは、分析で抽出されたオブジェクトに対して、コンピュータに実現するために必要な、ユーザ・インタフェース部分、データ管理部分、タスク管理部分の情報がつけ加えられる。次に、物理設計フェーズでは、それらオブジェクトをプログラミング言語で記述する上での制約や方法（詳細な設計仕様）を決定するのである。

以上のことからオブジェクト指向設計は、オブジェクト指向分析の結果をプログラミングする際の方針を決定する部分にすぎないことがわかる。すなわち問題のモデル化において最も重要なのはオブジェクト指向分析である。次節ではオブジェクト指向分析の具体的内容について述べる。

A.2.2 オブジェクト指向分析

現在、オブジェクト指向分析・設計には数多くの手法が提案されている。これらの手法は以下のように三つに分類することができる。

- ・オブジェクトモデル・アプローチ：

Shlaer & Mellor 方法、Coad & Yordon 手法、OMT、

Booch 法、RDD、HOOD

- ・動的モデル・アプローチ：

OBA、OOSE

- ・機能モデル・アプローチ：

OOSD、Martin & Odell 手法、OOS

これらのアプローチは、A. 1 節で述べた三つのモデル（オブジェクトモデル、動的モデル、機能モデル）のどのモデルの構築に重点を置くかによって分類されている。まず、オブジェクトモデル・アプローチは、最初に実体のあるオブジェクトを見つける手法である。このアプローチでは、まず、与えられた要求仕様から、名詞とそれに対する様々な動作を記述した動詞を探し出す作業を行う。こうして得られた名詞はオブジェクトあるいは属性となり、動詞はメソッドあるいは関係になる。次に、特定したオブジェクトがどのような動作をするか把握する。最後に、オブジェクト内に組み込む機能を明らかにしてゆく。

この手法で分析を進めるには、完成度の高い仕様書の存在が前提となるが、大規模なシステムでは、そうした仕様書が存在しないことが多い。そこで、最近注目されているのが動的モデル・アプローチである。このアプローチでは、最初に把握するのは実体ではなく、まずシステムに何が起こるか（振舞い）を把握し、この振舞いを実行するオブジェクトを特定してゆくことでモデル化を進める。機能モデル・アプローチは、機能モデルの構築を重視する手法である。したがって、従来の手続き型プログラミングで用いられる構造化分析・設計手法をベースにオブジェクト指向の考え方を導入した手法も含まれる。そうした手法は、構造化手法の発展形と位置づけられ、オブジェクト指向方法論とは区別されることもある。

以下に、OMT⁽³⁵⁾手法を例にとってモデル化の流れを簡単にまとめる。付録C、Dには、この手法に基づいて電力系統解析支援システムを分析した結果を示す。

[1] オブジェクト指向分析

[1-1] 要求事項の列挙

- 問題の範囲（モデル化するシステムの役割）
- そのシステムに固有な条件（必要とするデータなど）
- 性能要求（例：将来の拡張性を確保する。）

[1-2] システムを構成するオブジェクトの静的構造の分析

（オブジェクト図の作成）

- (1) 分析の視点（モデル化の方針）の決定
- (2) オブジェクトの列挙（モデル化する対象を形作っているオブジェクト）
- (3) 列挙したオブジェクトの役割の列挙（使用方法、属性、操作）
- (4) 関連の定義
- (5) 属性の定義
- (6) 操作の定義（役割、機能、振る舞いをオブジェクトに割り当てる。）
- (7) 継承によるクラスの階層化（共通な属性を共有する。）

[1－3] システムを構成するオブジェクトの動的側面の分析

- (1) シナリオの書き出し（1－2（6）のシナリオ）
- (2) 事象トレース図作成（メッセージ送受信のタイミングの分析）
- (3) 状態図作成（オブジェクトの状態遷移とその遷移を引き起こす作業の関係を明らかにする。）

[1－4] システム内のデータの流れの分析

- (1) 入出力値のリストアップ
- (2) データフロー図による分析（システムの中をデータがどう流れるかを分析する。）

見落としていた操作を見つける。

その操作がどのオブジェクトのものであるかを特定する。

ある操作を行うときに必要なデータを明らかにする。

[1－5] オブジェクト図の完成

1－3、1－4の分析から1－2で見落とした操作をオブジェクトに付け加え、オブジェクト図を完成する。

[2] オブジェクト指向設計（具体的な実現方法の決定）

[2－1] 操作を実現するためのアルゴリズムの調査

アルゴリズムで使用するデータ構造の選択

中間結果保存のために新たに必要になったオブジェクトの追加

[2－2] 関連の実現方法の決定

（関連の記述方法：キー、ポインタ）

（検索方向：単方向、双方向）

A. 2. 3 オブジェクト指向プログラミングの利点

オブジェクト指向プログラミングを利用することで得られる利点としては、まず継承によるプログラムコードの再利用、モジュールかによる修正の影響の波及の防止などの効果をあげることができる。例えば、系統解析支援システムには、同様な機能を持ったインタフェースをいくつもそろえることになる。最初からインタフェースをプログラムしていたのではたいへんである。継承を利用

すれば、差分のみのプログラミングで、目的の機能を備えたインタフェースを用意することができる。

またカプセル化は、内部構造の異なるオブジェクトであっても入出力の仕様が統一されていれば、ユーザのアプリケーションを変更することなく使用することを可能にする。将来ある機器オブジェクトに新たな機能を組み込む必要が生じて、既存のオブジェクトと入出力の仕様が統一されていればオブジェクトの置き換えだけで、既存のアプリケーションに修正することなく要求に対処することができる。

さらにメッセージパッシング処理により得られる利点もある。ウインドーシステムでは、画面上に表示されたウインドー、図や記号に対してマウスなどを使用して処理を要求する。するとアクションを要求されたウインドーなどの持つサービスがユーザに返される。この一連の操作は、オブジェクト指向のメッセージパッシング処理に類似している。ウインドーシステムは、オブジェクト指向の概念に基づいたシステムであり、オブジェクト指向プログラミングとの相性がよい。

A. 3 オブジェクト指向プログラミングの応用

A. 3.1 オブジェクト指向データベース⁽⁸⁾

プログラミングによって作成されたオブジェクトはプログラムの実行中にのみメモリ上に存在し、一旦プログラムが終了してしまえばオブジェクトの存在は保証されなくなってしまう。ところが現実世界のオブジェクトは永続的に存在し、情報を保持し続けるのが一般的である。オブジェクト指向に基づいてモデル化されたオブジェクト指向モデルでデータを永続管理することができれば、より現実の問題を忠実にモデル化できるようになる。このオブジェクトを永続管理するためのシステムがオブジェクト指向データベース管理システム (OODBMS) である。ここで OODB の特徴を明らかにするために現在広く利用されているリレーショナルデータベース (RDB) と比較する。

RDB は、テーブルで情報を管理するデータベース (DB) である。このテーブルモデルはもともと帳票などに記録されたデータを管理するのに適している。このテーブルモデルで電力系統データを管理することを考えてみる。電力系統データには機器に固有な属性値の他に機器どうしの接続関係や構造関係がある。機器に固有な属性値はテーブルでの管理に向いているが接続関係・構造関係の管理はあまり適していない。ある機器毎の属性値はユーザ定義のレコード ID を付加してテーブルに格納する。接続関係はこのレコード ID を別の機器のレコード中に属性値として保存することで表現する。そのため接続関係を調べるためにはつねにレコード ID を使った検索を関係のあるテーブルの全てのレコードにたいして繰り返さなければならない。複雑な部品構造関係をもつ電力系統の情報を RDB で管理するのは問い合わせの応答速度の面で問題がある。

これに対して OODB の場合は、関係をユーザ定義の ID ではなくデータベース管理システムが一元管理するオブジェクト ID で表現する。このオブジェクト ID はプログラミングの世界で言えばポイ

ンタのようなもので、オブジェクトIDを参照すればデータベース管理システムが対応するオブジェクトへのアクセスを保証している。レコードIDの管理や検索方法がユーザ任せのRDB比べて、部品構造関係をもつ対象の管理に適している。

以上のような特徴をもつOODBを電力系統のデータ管理に適用した場合の効果について検討してみた。以下にOODBを電力系統データ管理に用いる利点をまとめる。

1. データのみでなくそのデータを処理する手続もDBへ格納される。

オブジェクトは、データと手続きを合わせ持った構造を持ち、そのオブジェクトをDB上で永続管理することからこの特徴が生まれる。

2. DB上のオブジェクトとプログラミングで生成されたオブジェクトの間には、大きな違いはなく、プログラミングでデータベース上のオブジェクトが使用できる。

プログラミングで生成されたオブジェクトが、そのままDB上で永続管理されるためである。

3. 1.と2.からこれまでのようなプログラミング言語を複数使い分ける必要がなくなる。その結果プログラムの保守が容易になる。

RDBの場合はSQLと呼ばれる専用の問い合わせ言語を使用しなければならない。

以上のようにOODBには電力系統のデータ管理にも大きな効果があると考えられる。しかし、これはOODBの静的な状態を保存する機能の利点だけである。現在動的な状態を保存し管理する機能(バージョン管理機能)に関する研究が進められている。

A.3.2 バージョン管理機能

バージョン(版)管理機能は、設計DBに必要不可欠になってきている機能の一つである。設計DBは、試行錯誤を繰り返す設計支援を目的としたDBのことであり、CADシステムやソフトウェア開発等における計算機利用の高度化に伴いその重要性が認識されている。設計DBにおいては、通常、設計中の試行錯誤、設計作業の進歩、製品の改良、製品系列の拡充等の要因によって、時間の経過に伴い様々な版が作成される。この設計データの複数の版を記録し、後に必要な版の検索・再利用を可能とするためのものがバージョン管理機能である。以下にバージョン管理を電力系統データの管理に適用する利点について検討してみた。

バージョン管理の概要： まずバージョン管理機能の特徴を以下に列挙する。

- (1) 電力系統解析において重要な構成や運用条件を変えながら解析結果を比較する際に役立つ。(容易に以前の構成や運用条件を再現できる。)
- (2) 構成や運用条件を修正した電力系統をオリジナルの電力系統のコピーを利用して管理した場合に比べて、バージョン管理機能は、オリジナルの系統と修正した系統の差分のみを管理するので記憶領域を浪費しない。
- (3) バージョン管理機能を利用すると、バージョンの検索等の機能が利用できるため、構成や運

用条件を変えながら電力系統を解析するアプリケーションが、使用しなかった場合と比べてはるかに作成しやすい。

これらの利点から電力系統データの管理にバージョン管理機能を組み込むことで、解析作業の支援、電力系統データの管理、アプリケーションの作成に役立つことがわかる。現在、OODBには、すでに版管理を実装したものがあり、本研究で使用する Versant にも実装されている。しかし、基本的な機能を与えているだけのものがほとんどで、版の導出関係の管理や旧版の頻繁な再利用が十分にサポートできない。特に、複合オブジェクトを多く含む複雑な構造のデータモデルが対象となるとき、オブジェクト間のリンクの状態を正確に復元することが困難である。そのため、各応用分野に適応した版管理機能を実現するためには、ユーザが版管理のためのプログラムを定義する必要がある。そこで、現在、ユーザが版管理のプログラムを構築するにあたっての基本的な枠組みを与えることを目的とした開発・研究が行われており、いくつかモデルが提案されている^(32, 36-38)。このモデルの中には、OODB以外のDBを対象としたものもあるが、本来、OODBは設計DBとしての能力が高いことから、OODBを対象としたものが多い^(32, 36, 37)。つづいて本研究で採用しているバージョン管理システムについて紹介する。

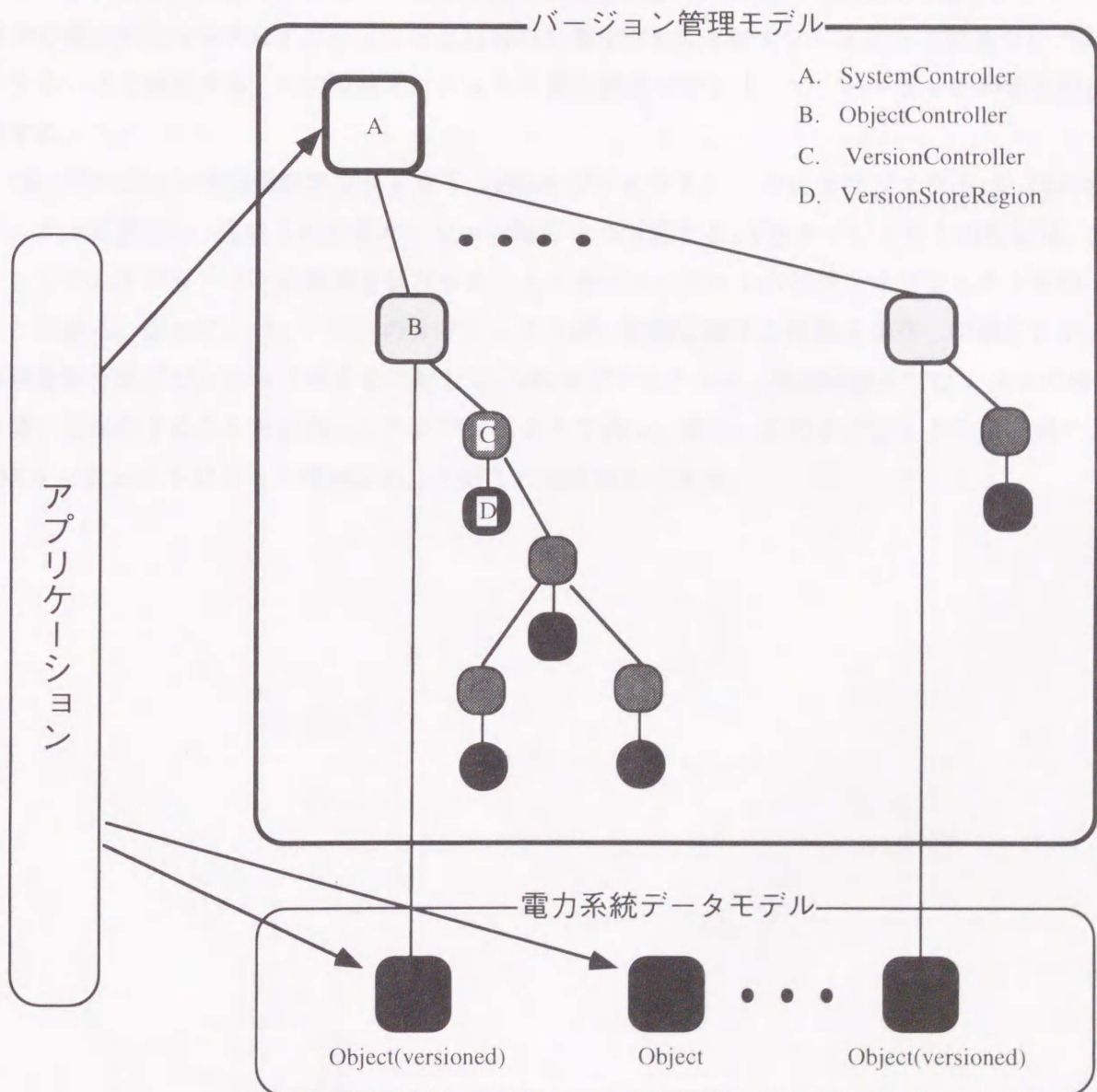
Vmodelの基本構造： 本研究では、電力系統DBのバージョン管理のために Vmodel⁽³²⁾と呼ばれるバージョン管理システムを採用している。まずこのシステムを採用する理由を簡単にまとめておく。図A.6の中での○は、DB上のオブジェクト（インスタンス）を表している。ラインは、オブジェクト間のリンクを表している。図に示すように、本モデルは、データモデルのオブジェクトやアプリケーションから独立したオブジェクトにより構成している。そして、機能提供に必要な情報管理や制御処理は、これらのオブジェクト群が全て行う。つまり本モデルにはバージョン管理に関する機能のみがモジュール化されており、既存のオブジェクト指向DBにこのモジュールを適用することでDBのバージョン管理を実現することができる。言い換えると、バージョン機能の適応にあたって既存のDBの仕様を変更する必要がないのである。そのため、今後の機能拡張が簡単に実現できるシステムを構築できる。これに対して、文献(36)などのモデルでは、データモデルのオブジェクトにバージョン管理機能に必要な属性や操作を定義しており、データモデルとバージョン管理モデルが一体化している。このような方法は、機能を拡張することでデータモデルの通常操作に影響を与えやすく、データモデルを利用するアプリケーション側に変更を強いる恐れがある。また、バージョン機能を必要としない場合に実行性能を低下させる恐れもある。

Vmodelを構成するこれらのオブジェクトは、情報管理や制御処理を役割分担して行うため、「システム管理オブジェクト」「オブジェクト管理オブジェクト」「バージョン管理オブジェクト」「バージョン情報保存オブジェクト」の4つに分類されている。以下に、各オブジェクトの役割を簡単に説明しておく。

(1) システム管理オブジェクト (SM オブジェクト) SM オブジェクトは、図の中のAのオブ

ジェクトに対応するものであり、DB一つに対して一つ用意する。SMオブジェクトの役割は、モデル全体を管理することであり、全ての処理の発信源となる。従って、バージョン機能を利用するアプリケーションは、SMオブジェクトにのみ処理を依頼するだけで、バージョン管理モデル内部の構造を知る必要はない。

(2) オブジェクト管理オブジェクト (OMオブジェクト) OMオブジェクトは、図のBオブジェクトに相当し、データモデル内のバージョン機能を適応するオブジェクト (適用対象オブジェクト) 毎に一つ用意する。OMオブジェクトの役割は、適用対象オブジェクトの全てのバージョンを管理することである。また、図に示すように、OMオブジェクトは、唯一データモデル内のオブジェクトとの関係をもつ。この関係は、1方向の関連づけになっており、DB内のオブジェクトからの関係



図A. 6 バージョン管理モデルの基本構造図

付けはしない。バージョン管理において既存のデータモデルは常に情報の受け手となるので双方向の関連付けは必要としていない。逆方向（管理対象からOMへ）の関連付けも行うということは既存のデータモデルに修正を加えることになり、アドオン方式のシステムであるという特徴を失うことになる。このオブジェクトの管理は、SMオブジェクトが行う。

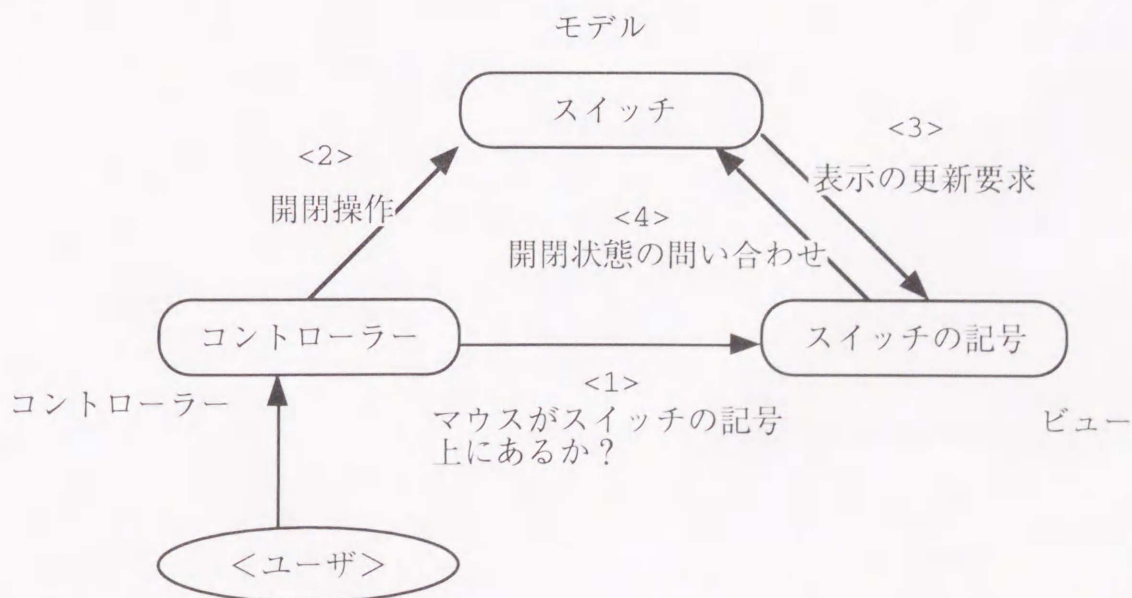
(3) バージョン管理オブジェクト (VMオブジェクト) VMオブジェクトは、図のCオブジェクトに相当し、適応対象オブジェクトに作成された各バージョン毎に一つ用意する。そのVMオブジェクトの役割は、生成された各バージョンを管理することである。そのため、各バージョンの制御に関する情報の保存と処理を行う機能が組み込まれている。このオブジェクトの管理は、OMオブジェクトが行う。また、各バージョンには、導出元となるバージョンや導出先となる複数のバージョンが存在する。このようなバージョン間の関係は、図A.6に示すように、VMオブジェクトが、自身の導出元となるVMオブジェクトと自身から導出されたVMオブジェクトと関連づけ（双方向）をすることで保存する。このVMオブジェクト間の関連づけによって、バージョンの導出関係を保存する。

(4) バージョン情報保存オブジェクト (VISオブジェクト) VISオブジェクトは、図のDオブジェクトに相当し、作成された各バージョン毎に一つ用意する。VISオブジェクトの役割は、各バージョンでのオブジェクトの状態を保存することとそのバージョンの状態にオブジェクトを復元することである。従って、(1)～(3)のオブジェクトが、制御に関する情報を保存し制御を目的とした処理を行うオブジェクトであるのに対して、VISオブジェクトは、適応対象オブジェクトの情報（属性値）を保存することを目的としたオブジェクトである。また、このオブジェクトは、図のようにVMオブジェクトによって管理され、1対1に対応付けされる。

B. MVCモデル

Smalltalkにはモデル・ビュー・コントローラー（MVC）と呼ばれるウインドープログラムを作成する指針がある。モデルは、作成するウインドーによって表示されるべき内容を表現するオブジェクトである。ビューは、モデルの表示を司るオブジェクトである。コントローラーは、モデルやビューに対する操作や指示を処理するオブジェクトである。画面上でマウスを使ってスイッチの開閉を設定するインターフェースをMVCモデルでプログラム化することを考えてみる。モデルは、スイッチを表現するオブジェクトで、状態を保存する属性と<3>スイッチ状態の更新要求を促すメッセージを周囲のオブジェクトにブロードキャスト（放送）するメソッドが組み込まれている。ビューは、スイッチの状態を表示するオブジェクトで、モデルからスイッチの更新要求メッセージを受信したら<4>モデルの状態を参照してスイッチの表示を更新するメソッドが組み込まれている。コントローラーは、スイッチの状態を操作するオブジェクトで、<1>マウスがスイッチの記号上にあるかどうかを調べるメソッドと<2>スイッチのメニューで選択した状態をモデルに設定するメソッドが格納されている。この枠組みを図B.1に示す。

いまユーザがマウスをスイッチの記号の上に移動させるとコントローラーは、図のメソッド<1>の処理を実行する。つづいてユーザがマウスボタン等でスイッチの状態を設定するとコントローラーは、メソッド<2>を実行し、選択した状態をモデルへと登録する。モデルは、新しいスイッチの状態が設定されると、メソッド<3>を起動し、自分の周囲のビューに対して表示の更新要求を促すメッセージをブロードキャストする。モデルからブロードキャストされたメッセージを受信するように設定されたビューは、メソッド<4>を起動してモデルの状態を調べ、表示を更新する。



図B.1 スイッチのMVCモデル

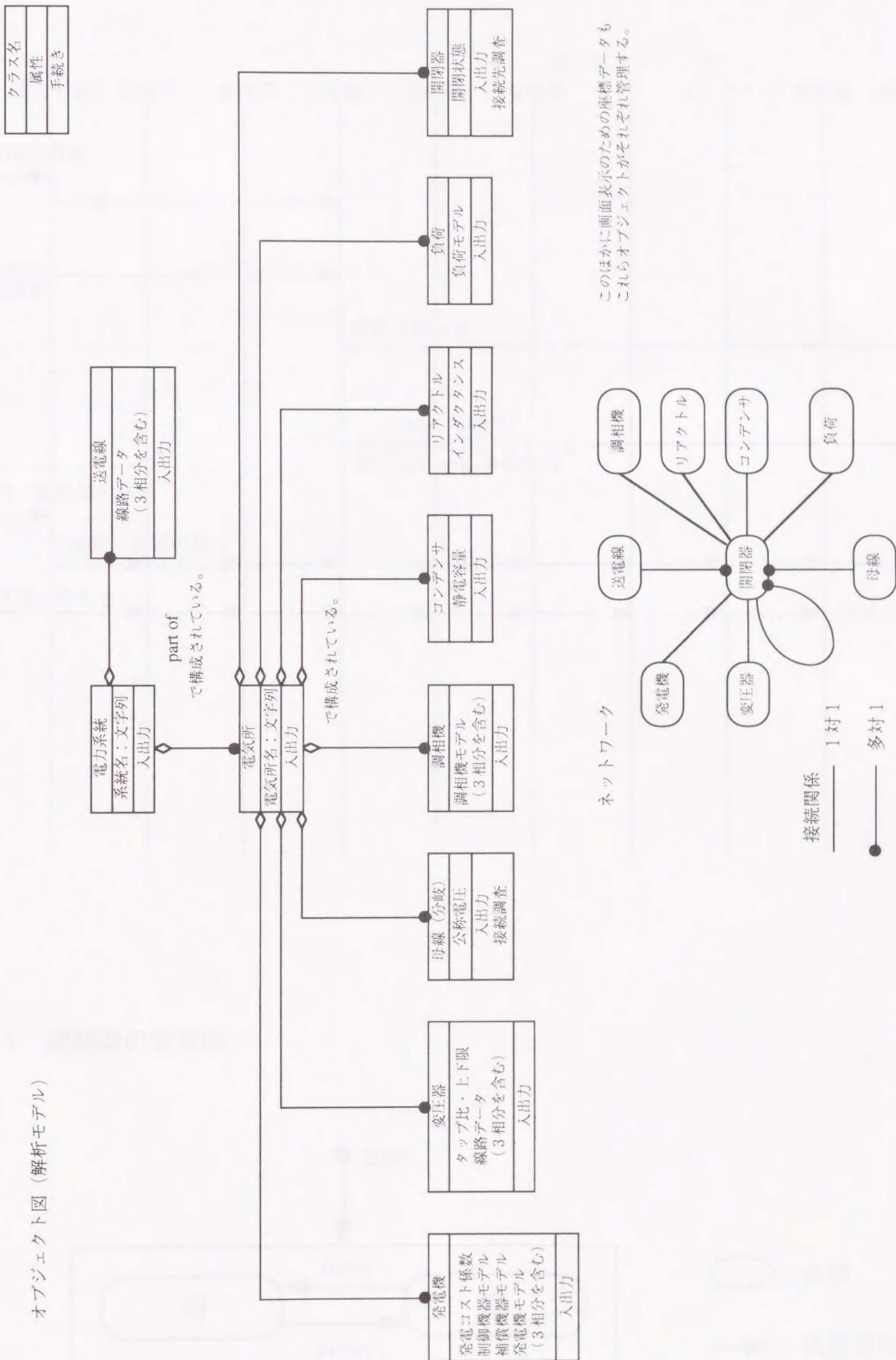
このような形態にすると、モデルは表示のための情報を持たないため、モデル側のコードを変更することなく自由にビューを取り替えることができる。同様にモデルはコントローラーを制御するメソッドを持たないので、コントローラーの取り替えも自由である。MVCモデルは、モデル・ビュー・コントローラー内部の変更による影響が他のオブジェクトに非常に伝搬しにくいモデルである。このモデル化手法に従えば、完成度の高いインターフェースを短時間に構築することが可能である。

C. 電力系統モデルのオブジェクト指向分析

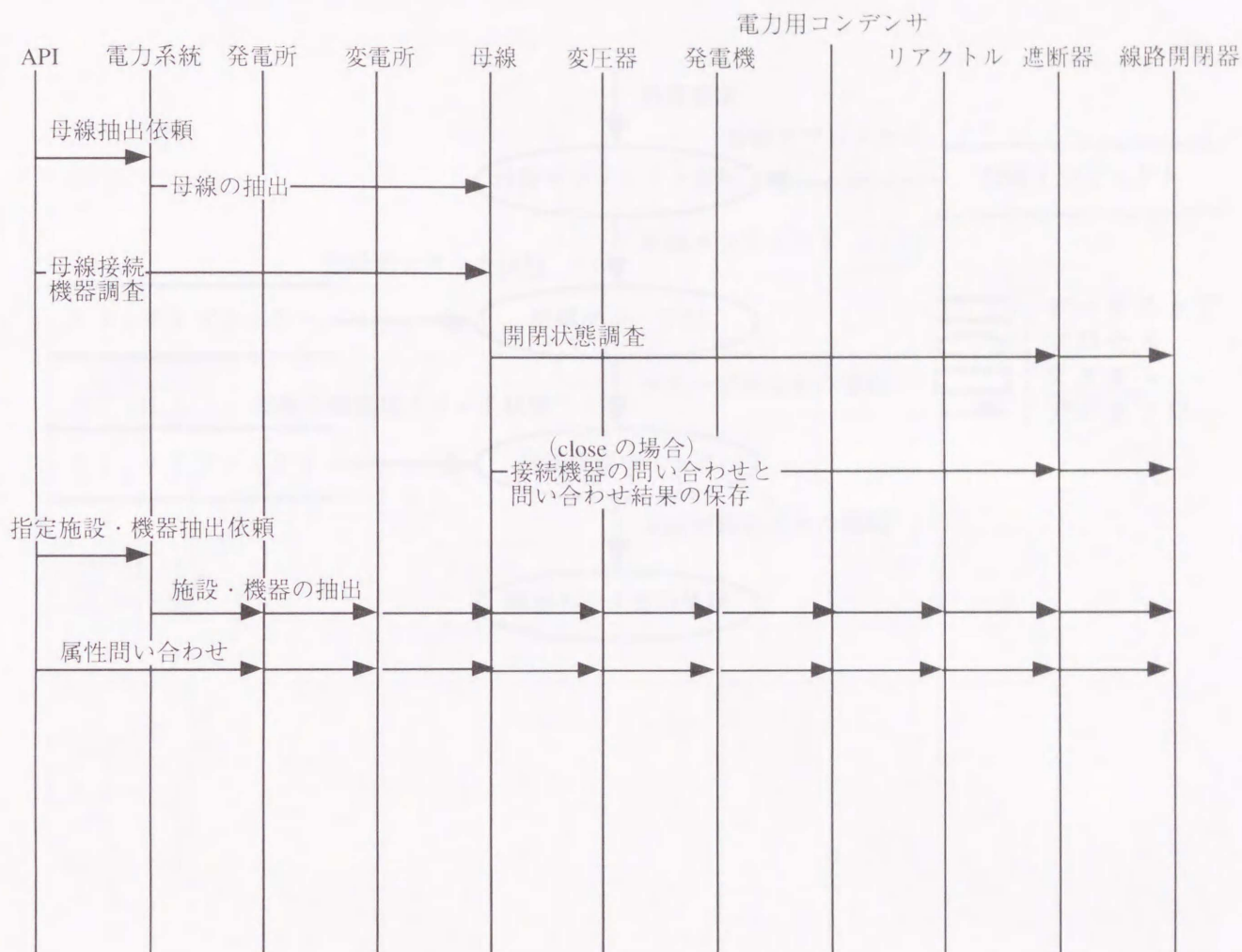
2章でまとめたOMTによるオブジェクト指向分析手法を用いて、オブジェクト指向データベース用電力系統モデルの分析結果を示す。本研究では、この結果に基づいて電力系統モデルをオブジェクト指向プログラミングしてある。



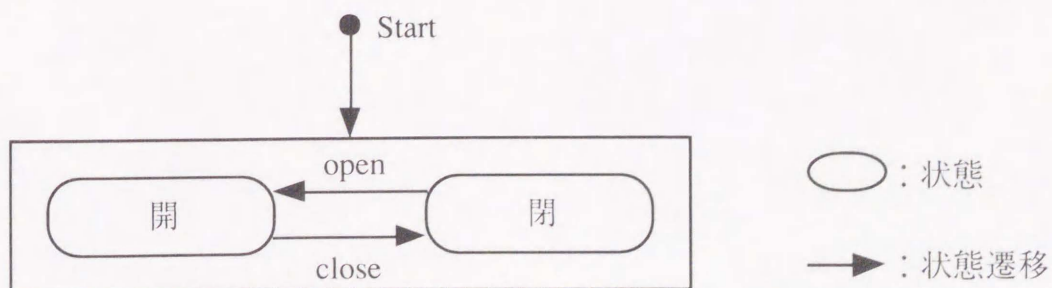
C.1 電力系統のオブジェクト図 (データベースモデル)



C.2 電力系統モデルのイベントトレース図

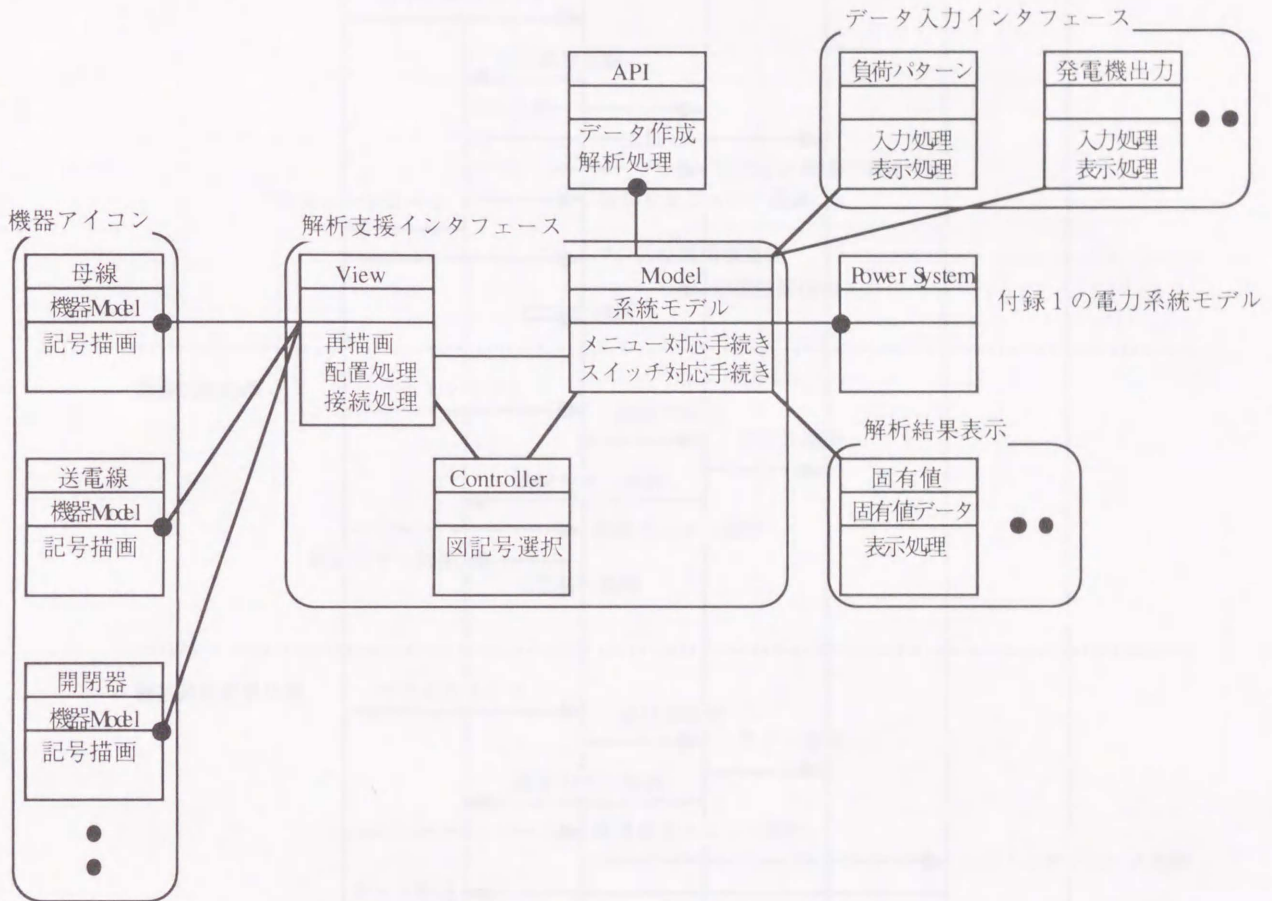


C.3 開閉器の状態図

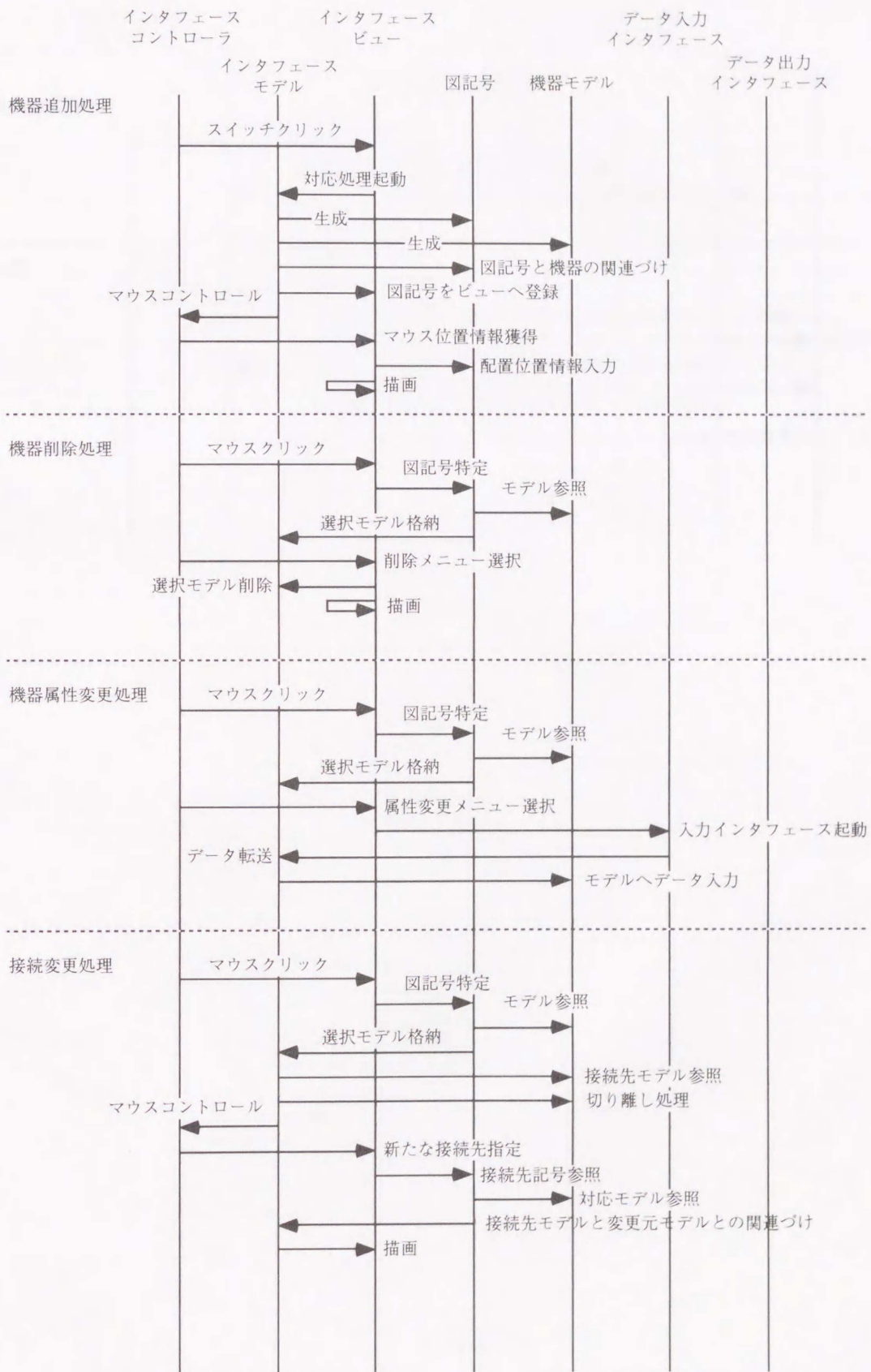


D. 電力系統解析支援インタフェースのオブジェクト指向分析

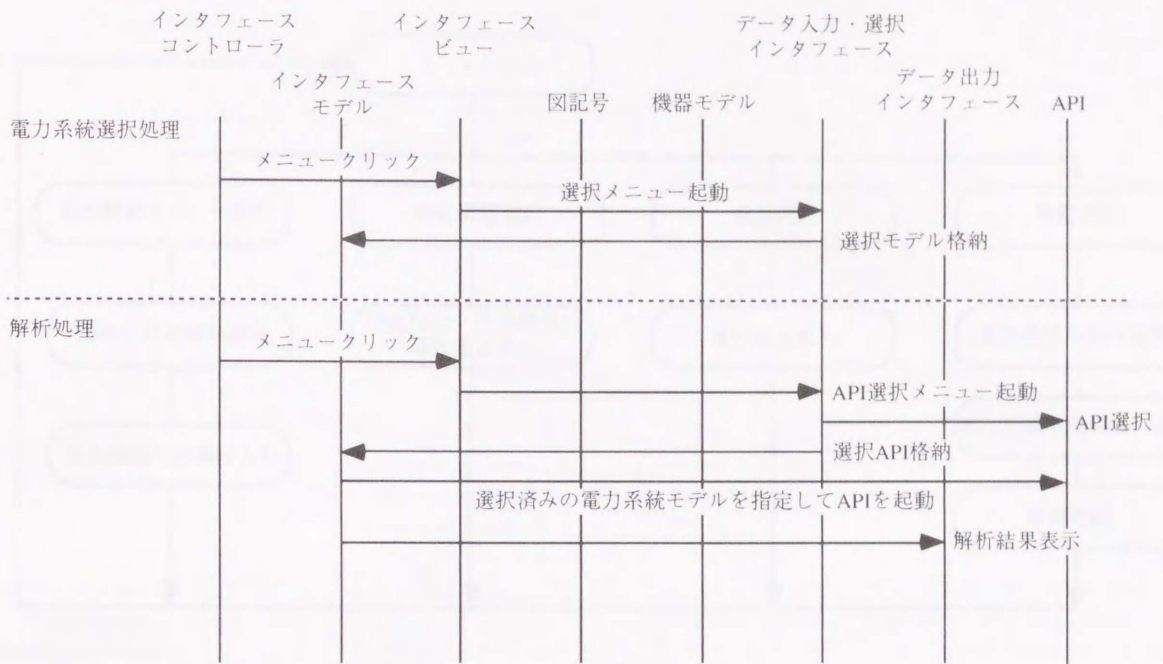
D.1 電力系統解析支援インタフェースのオブジェクト図



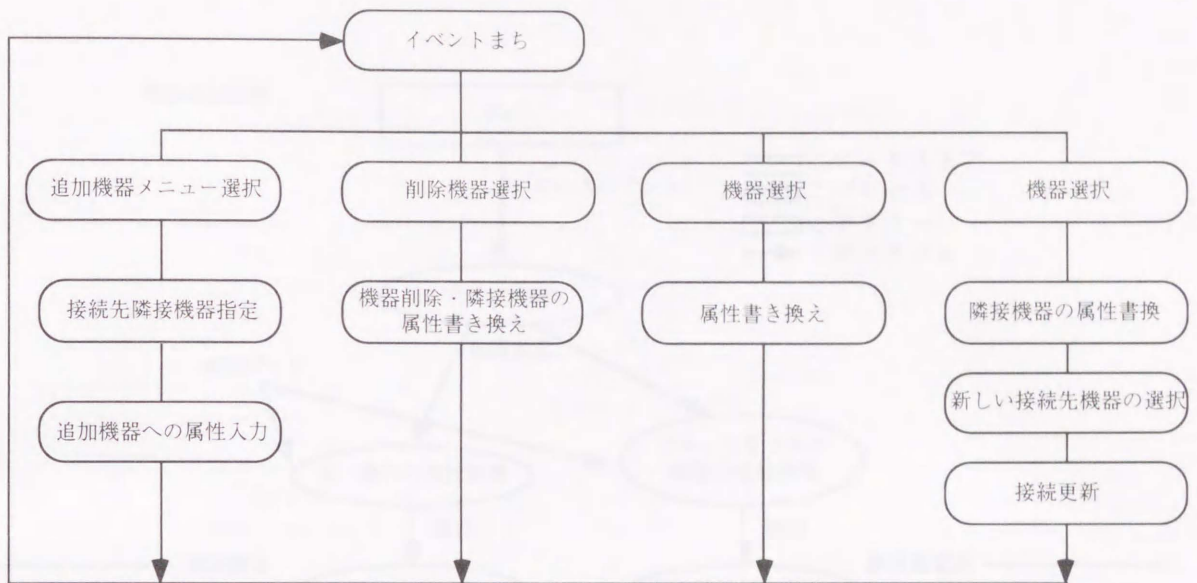
D.2 電力系統解析支援インタフェースのイベントトレース図その1



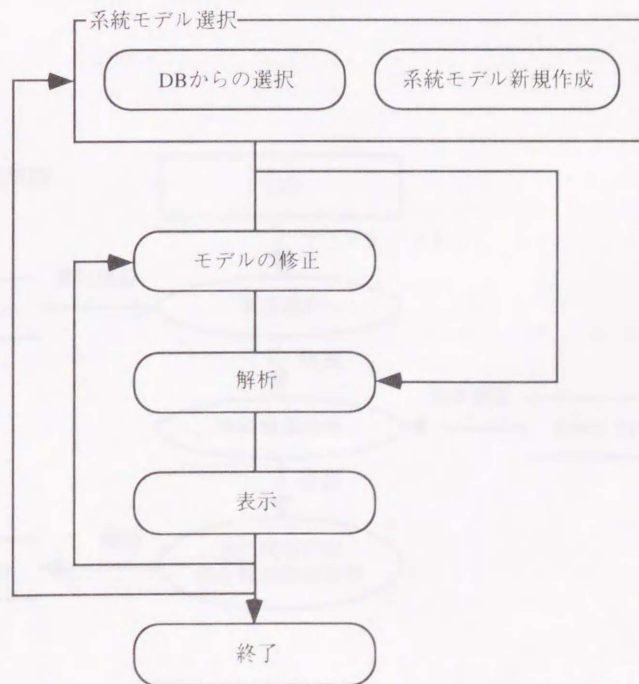
D.3 電力系統解析支援インタフェースのイベントトレース図その2



D. 4 電力系統解析支援インタフェースの状態図

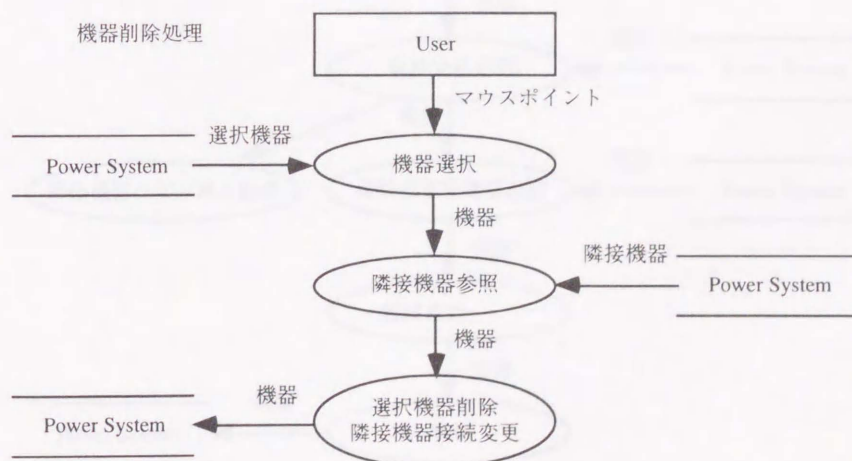
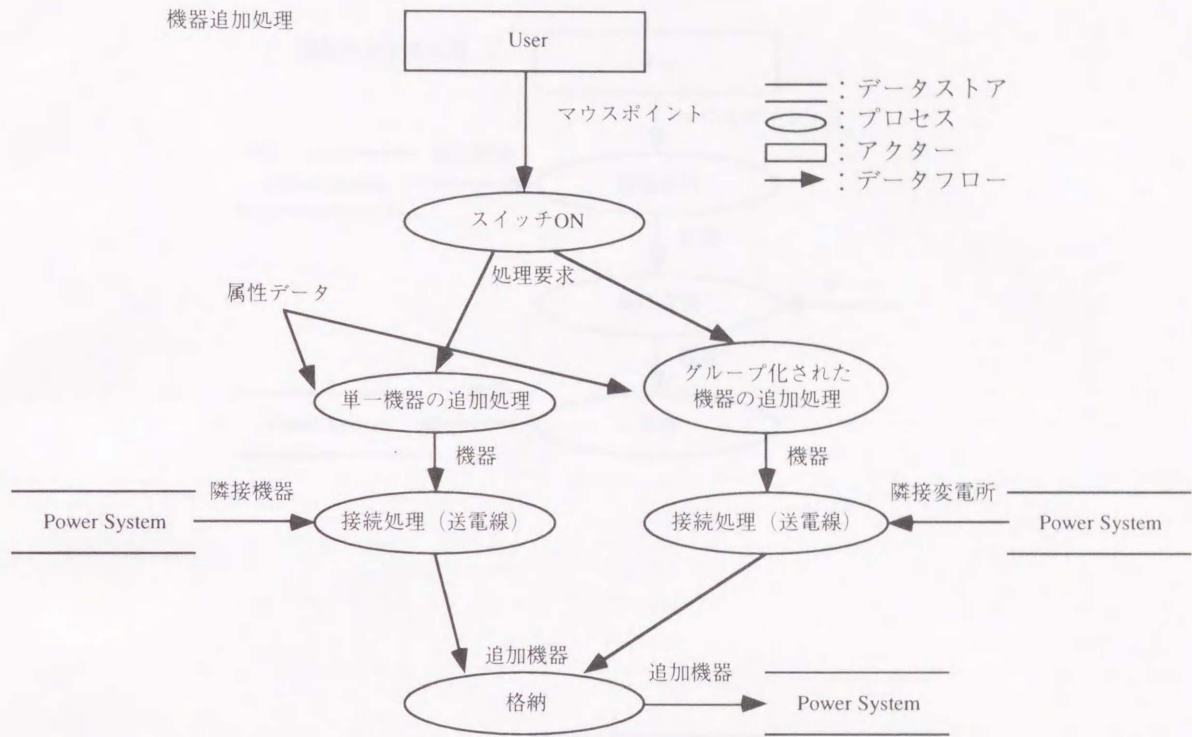


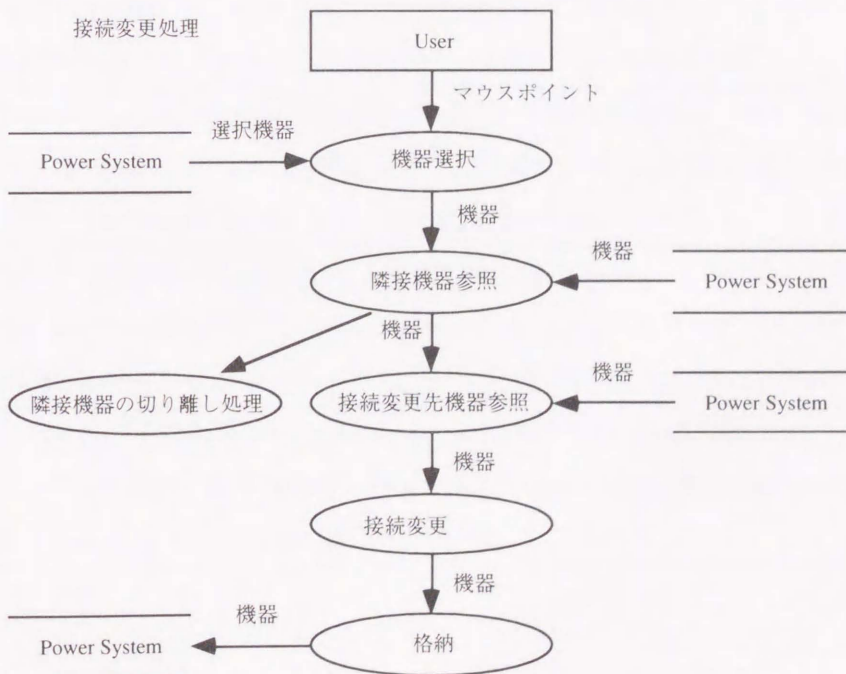
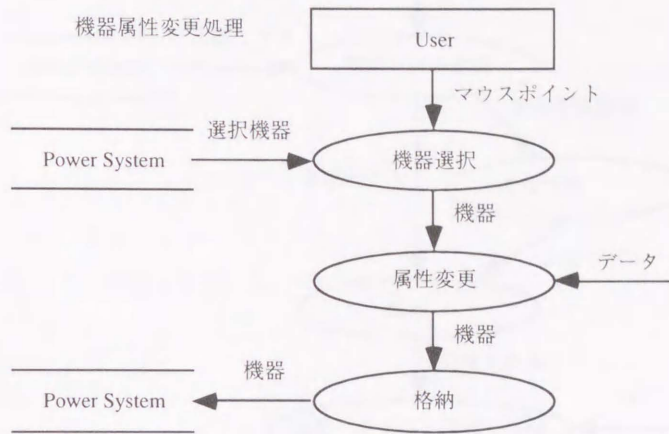
(a) 編集時の状態図

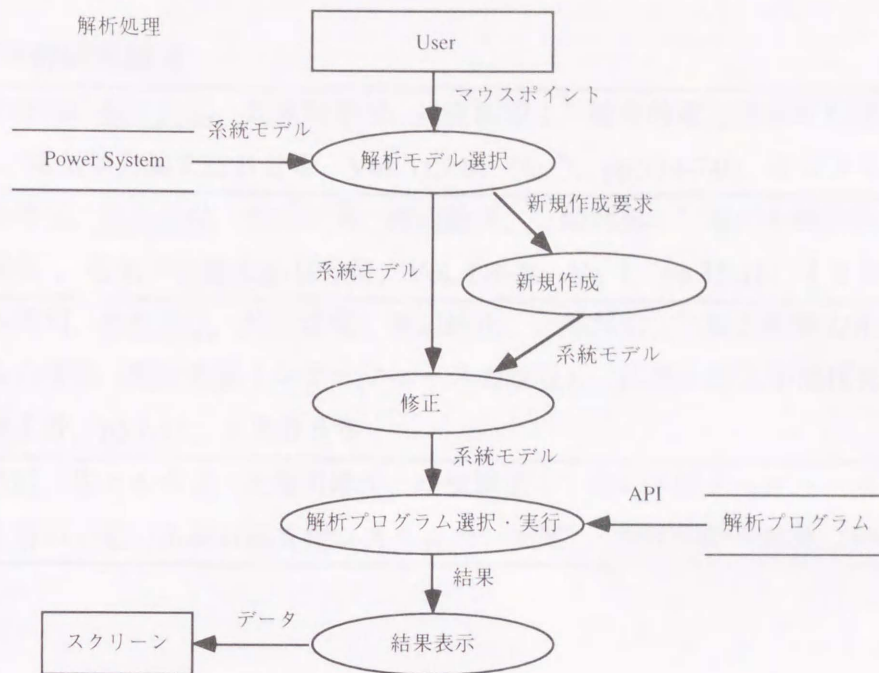


(b) 解析時の状態図

D.5 電力系統解析支援インタフェースのデータフローダイアグラム







研 究 業 績

1. 学会誌等学術研究論文

[1]	佐々木博司, 歌谷昌弘, 久保川淳司, 川原耕治: "統合的電力系統解析支援システムの構築", 電気学会論文誌B分冊, Vol. 115-B, No. 7, pp.734-740, 1995年7月
[2]	佐々木博司, 歌谷昌弘, 竹田浩明, 渡辺敏正, 三崎邦男: "電力系統図自動表示システムの開発", 電気学会論文誌B分冊, Vol. 116-B, No. 1, pp.35-41, 1996年1月
[3]	佐々木博司, 歌谷昌弘, 竹田浩明, 渡辺敏正, 三崎邦男: "統合的電力系統解析支援システムの構築(解析支援インターフェースの改良)", 広島大学工学部研究報告, 第45巻, 第1号, pp.7-13, 1996年
[4]	歌谷昌弘, 佐々木博司, 久保川淳司, 川原耕治: "電力系統データベースにバージョン管理を用いた電力系統計画支援システムの一提案", 電気学会論文誌B分冊, 1999年6月掲載予定

2. 国際会議発表研究論文

[1]	Hiroshi Sasaki, Masahiro Utatani, Junji Kubokawa, Koji Kawahara, "Development of an Integrated Environment to Support Power System Analyses", 9TH Conference on Electric Power Supply Industry (9th CEPSI, Hong Kong), Vol.4, pp.389-397, November, 1992
[2]	Hiroshi Sasaki, Masahiro Utatani, Junji Kubokawa, Koji Kawahara, "A Construction of an Integrated Support System for Power System Analyses", 10TH Conference on Electric Power Supply Industry (10th CEPSI, New Zealand), Vol.3, pp.181-187, September, 1994
[3]	Hiroshi Sasaki, Masahiro Utatani, Junji Kubokawa, Koji Kawahara, "An Integrated Support System for Power System Analyses with an Improved Application Program Interface(API)", The International Conference on Electrical Engineering (ICEE'95, Korea), pp.256-260, July, 1995
[4]	Masahiro Utatani, Hiroshi Sasaki, Junji Kubokawa, Koji Kawahara, "A Development of a Support System for Power System Planning", The International Conference on Electrical Engineering (ICEE'97, Japan), pp.14-17, July, 1997

3. 学会技術研究会等研究発表

(1)	佐々木博司, 歌谷昌弘, 竹田浩明, 久保川淳司, 川原耕治: "電力系統解析支援用統合環境の開発-II", 電気学会電力技術研究会資料, PE-92-138, pp.91-99, (1992)
(2)	佐々木博司, 歌谷昌弘, 竹田浩明, 久保川淳司, 餘利野直人, 川原耕治: "統合的電力系統解析支援システムの構築", 平成5年電気学会電力・エネルギー部門大会, 170, pp.136-137, (1993)
(3)	佐々木博司, 歌谷昌弘, 竹田浩明, 久保川淳司, 川原耕治: "電力系統解析支援用統合環境の開発-IV", 電気学会電力技術研究会資料, PE-93-17, pp.1-10, (1993)
(4)	佐々木博司, 歌谷昌弘, 久保川淳司, 川原耕治: "電力系統解析支援用統合環境の開発-VI", 電気学会電力技術研究会資料, PE-94-42, pp.57-64, (1994)
(5)	佐々木博司, 歌谷昌弘, 久保川淳司, 川原耕治: "解析プログラム登録支援ツールの開発", 電気学会電力技術研究会資料, PE-95-173, pp.117-125, (1995)
(6)	歌谷昌弘, 佐々木博司, 久保川淳司, 川原耕治: "系統計画支援システムの開発(システムの概要)", 電気学会電力技術研究会資料, PE-96-91, pp.59-67, (1996)
(7)	歌谷昌弘, 佐々木博司, 久保川淳司, 川原耕治: "系統計画支援システムの構築(アプリケーション登録支援機能について)", 平成9年電気学会電力・エネルギー部門大会, 191, pp.180-181, (1997)
(8)	歌谷昌弘, 佐々木博司, 久保川淳司, 川原耕治: "電力系統計画支援システムの開発(制御器登録支援エディタ)", 電気学会電力技術研究会資料, PE-97-119, pp.53-58, (1997)
(9)	歌谷昌弘, 佐々木博司, 久保川淳司, 川原耕治: "電力系統計画支援システム(複合オブジェクトのバージョン管理)", 電気学会電力技術研究会資料, PE-98-122, pp.21-25, (1998)