

機械設計支援エキスパートシステム構築シェル: MAGIC†

中島裕生** 田原一徳** 加藤亨**

機械工業において、製品系列単位で設計支援エキスパートシステムを構築する際、共通に使えるシェルの存在は有益である。機械設計支援エキスパートシステムの実用段階では、次の機能、①大規模知識ベースに対する保守性、②設計問題向き専用推論エンジン、③エンジニアリング・データベースとの融合、④幾何形状操作に関する知識処理、⑤他のエキスパートシステムとの協調問題解決、⑥設計者の意思による自然な制御、⑦実用段階での性能、機能に対する柔軟なチューニング、⑧ユーザインタフェースの操作性の良さ等が必要である。これらの要件をもとに、筆者らは、属性モデリングによる設計対象のモデル化のパラダイムに沿って、知識処理可能なグラフィック・アイコン・ベース CAD、制約指向言語を基礎にした知的スプレッドシート、複合オブジェクトによる知識表現と一体になったオブジェクト指向データベース、およびこれらを相結合しダイナミックコントロールを行う設計プロセス制御部からなるシェル MAGIC を開発した。さらにこのシェルを使用して、油圧ポンプ・モータ設計支援エキスパートシステムを開発し、次の効果、①構築されるエキスパートシステムが、そのライフサイクルの過程で直面する知識、データの増加に対し、パフォーマンス向上のために柔軟に対処できること、②知識、データの追加・維持管理を設計者自身に実施してもらうための保守性向上、③設計支援エキスパートシステム構築の生産性向上、が達成されることを確認した。なお、本システムの大半は、オブジェクト指向言語 Flavors で実現されている。

1. はじめに

機械工業において、設計支援のためのエキスパートシステムを構築する場合、構築対象の粒度 (granularity) として、機械製品系列ごとを単位にしたほうが、知識ベースとしてのまとまり、実用可能性、運用面などから見て適当と考えられる。機械製品系列単位で構築するにしても、実際には、多品種少量で数多くの製品系列が存在するために、機械設計支援エキスパートシステム構築シェル**の存在は有益である。

これまで、機械設計を対象としたエキスパートシステムまたはコンセプトの実証として、それに類似したシステムが幾つか発表されている。これらには、長澤らによる制約条件解法の有用性¹⁾、伊藤による機械設計のための属性抽出を容易化する設計対象モデリング²⁾、Dixon らによる形状特徴 (Form Feature)³⁾、

Chandrasekaran らによるジェネリックタスク概念^{4),5)}、さらには Steinberg による詳細化と制約伝播の概念⁶⁾ などがある。こうした研究成果が提供、あるいは実証してくれるコンセプトは、現実の設計行為を計算機向きにモデル化する際、有益な視点を与えてくれる。その中でも、従来の形状処理を中心とする CAD に対し、新しい概念として Dixon らの "Form Feature" の概念を包含する「属性モデル」「属性モデリング」が提唱されている^{7),8)}。

この概念は、設計対象にはそれを表現するのに本質的かつ必要十分な属性集合が存在し、設計はその属性の操作によって実現されるというものである。ここで言う属性とは単に値だけでなく幾何形状情報も含んでいる。さらに、機械設計において、設計対象に本質的な属性集合と対応する幾何実体を洗い出すことが可能であり、これを行うことにより設計支援システムの貢献度合いを上げるというものである。このような概念にしたがった対象のモデル化を属性モデリング、そのモデルを属性モデルとよんでいる。筆者らも、すでに実用化されている油圧回路設計支援エキスパートシステム⁹⁾ の中でこうした概念をつかった知識処理を実現している。さらにこのエキスパートシステムの中で、

† A Shell for Building Mechanical Design Assistant Expert System: MAGIC by YUSEI NAKASHIMA, KAZUNORI TAHARA and TOHRU KATO (Development Team, Technology & Research Center, KAYABA Industry Co., Ltd.).

** カヤバ工業(株)技術研究所開発チーム

* 現在 ニチメンデータシステム(株)知能情報システム部
Artificial Intelligence & Information Eng. Dept.,
Nichimen Data Systems Corp.

** シェルとは通常エンドユーザがドメイン知識だけを入れると、エキスパートシステムを構築できるものをさすが、ここでは機械製品系列ごと、さらに運用段階での知識・データの増加に対応してシステムをチューニングできるようツールとしての特徴を持つシェルとして使用している。さらにこのシェルを使ってエキスパートシステムの実用版開発を実施するのは知識工学者を、運用段階ではエンドユーザを、また運用段階でのシステム・チューニングについて知識工学者を想定している。

** 当初 Dixon の Feature-based modeling を属性モデリングとして紹介されたが、Dixon は Form Feature (形状特徴) を主張しており、日本で提唱されている属性モデル、属性モデリングでは形状特徴を一属性として包含していると考えられる。ここでも日本で提唱されている意味の属性モデル、属性モデリングという言葉を使用し、特に断らないかぎり本文中の「属性」は属性モデルのそれを示す。英語表現では "feature" を使用する。

ATMS (Assumption-based Truth Maintenance System)¹⁰⁾, 制約条件伝播などの AI 技術, およびオブジェクト指向データベースなどの有用性を確認している。

本研究では, 設計対象知識, 設計操作知識, 設計制御知識と分類される設計知識¹¹⁾に対して, 設計対象知識には属性モデルを採用し, さらに残り二種類の知識と明確に分離することを特徴として, 実用化段階で求められる機能を実現した, **機械設計支援エキスパートシステム構築シェル: MAGIC**を開発した。ここでは, まず機械設計支援エキスパートシステムに対する要件を示し, 次にそれを実現するシェルのシステム・アーキテクチャについて述べる。

2. システム・アーキテクチャ

機械設計支援エキスパートシステムに求められる要件を, 基本要件と実用要件とに分類してみる。基本要件は次のようになる。

- ①幾何形状を介したマン・マシーン・コミュニケーション
- ②設計者とシステムの間で交互に主導権を持ち合い, 設計問題解決に当たる
- ③機械設計に特徴的な制約充足問題解決を効果的に達成できること
- ④機械設計は, 材料科学, 構造工学, 生産工学などの多くの周辺専門分野から支えられており, これらの知識をタイムリーに設計に活かせること
一方, 実用要件は次のようになる。
 - (1)大規模知識ベースに対する保守性
 - (2)エンジニアリング・データベースとの融合
 - (3)エキスパートシステムのライフサイクル過程で直面する知識, データの増加に対するパフォーマンス向上
 - (4)実用において頻繁に起きる, 設計変更管理に対応できること
 - (5)多様な設計者に受け入れられること
 - (6)従来 CAD との連携

こうしたエキスパートシステムに求められる要件は, それを実現するシェルのアーキテクチャに反映される。システム構成図を図1に示し, 先に述べた要件群と, システムの各モジュールとの関係を表1に示す。外部コミュニケーション・モジュールは, 他のエキスパートシステムとの分散協調問題解決を行うためのインタフェース機能, 従来の CAD へのデータ交換

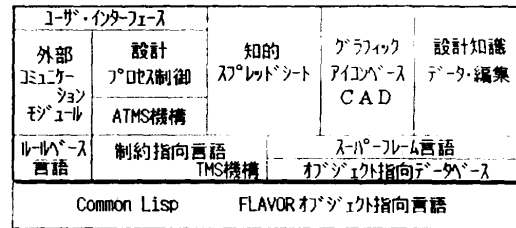


図1 機械設計支援エキスパートシステム構築シェル: MAGIC

Fig. 1 Configuration of a shell for building Mechanical Design Assistant Expert System: MAGIC.

表1 システム構成モジュールと対応する要件
Table 1 System's components and corresponding essential factors.

	基本要件	実用要件
設計知識・データ編集モジュール		(1)(2)
設計プロセス制御モジュール	②	(3)(4)(5)
知的グラフィック・エディタ	①	(6)
知的スプレッドシート	③	(4)
外部コミュニケーション・モジュール	④	(6)

のための IGES インタフェース機能, およびプロッタなどの周辺機器へのデータ伝達インタフェース機能からなる。ユーザである設計者は, 知的グラフィック・エディタと知的スプレッドシートを主に使用して設計作業を進め, データベースへのアクセス, および他のエキスパートシステムとのコミュニケーションを行う。

以下に主要モジュールの詳細について述べる。

2.1 属性モデル表現言語と設計知識・データ編集モジュール

設計対象を属性集合と捕えることを先に述べたが, 現実の設計対象を表現しようとする, その属性は極めて多量になり, 多様な属性値を持つ。したがって, 属性モデルを実際にどう表現するかと同時に保守を含めた量的問題を解決するという言語が必要になる。ここでは, このためにスーパーフレーム言語¹²⁾というものを使用している。これは, 宣言型知識表現とオブジェクト指向データベース構造生成を同時に行い, 知識表現単位として, スーパーフレームと呼ぶ複合オブジェクトによるクラス構造を持ち, そのスーパーフレーム間の多重継承関係が実現できるようになっている。従来の宣言型知識表現言語と違和感なく利用できながら, オブジェクト指向データベースのデータ構造生成を同

時に行うことが特徴である。したがって、スーパーフレーム言語を使用することによって次のようなオブジェクト指向データベースの特徴を享受することができる。

- ①可変長の豊富なデータタイプ (数値データ、文字データ、数式、ルール、制約条件、グラフ、図など)
- ②システム側が自動的にキー項目 (識別子) を内部で作るために、ユーザが特別に作成・管理する必要がない。
- ③データと手続きのカプセル化
- ④永続的オブジェクトの機能

①～④の特徴を、図2の多重継承関係にある、二つのスーパーフレームによる設計知識表現例で示す。「シューサブアッシー設計」というスーパーフレームは、「シューサブアッシー設計」の一つのフレーム・オブジェクト、「検討項目」、「関係設計式」、「参照実績データ」の三つのアトリビュート・オブジェクト、および「ピストンとシユの相対角検討」等々の複数のアイテム・オブジェクトから構成されている。フレーム・オブジェクトは独立オブジェクトであり、アトリビュート・オブジェクト、およびアイテム・オブジェクトはフレーム・オブジェクトに対する従属オブジェクトである。ここで、従属オブジェクトとは、その存在が他のオブジェクトの存在に依存するものを言い、ある一つのオブジェクトによって所有されているものである。フレーム・オブジェクトが定義されると二次記憶装置に④の機能を動かせることができる。「シューサブアッシー設計式」スーパーフレームの「設計式本体」アトリビュート・オブジェクトは、制約フォームのデータタイプ指定子を持つオブジェクトであり、それに従属するアイテム・オブジェクトのデータタイプを指定している。その定義例を図3に、また制約フォームというデータタイプ・オブジェクトの定義例と、そのオブジェクトに設定されるメソッド、つまり制約コンパイラを起動し、制約ネットワーク上に設計式を展開させる定義例を、図4に示す。このような①、③の特徴とともに、図2中の任意のアトリビュートに所属するアイテムの並びをキーに見出した検索 (例えば、「溶接部位径の式」の設計式本体は?) という検索) をスーパー

シューサブアッシー設計		
検討項目	関係設計式	参照実績データ
ピストンとシユの相対角検討	溶接部位径の式 溶接ピット外径の式 溶接ピット高さの式 ピストンとシユの相対角の式 シユ溶接部強度の式	製品実績表
シユの形状の検討	スチール厚MAXの式

多重継承関係

シューサブアッシー設計式					
設計式名	主パラメータ	従パラメータ	設計式本体	設計式に対する制約条件	適用種別
溶接部位径の式	溶接部位径 C1 mm ↓ $C1 = 2 \times (\text{SQRT}(DB \times H1 - H1^2))$		適用R-M1

図2 スーパーフレームによる設計知識表現例
Fig. 2 An example of super-frame based design knowledge representation.

```
(deframe シューサブアッシー設計式
  (設計式名 主パラメータ 従パラメータ 設計式本体
   設計式に対する制約条件 適用種別)
  :attribute-order
  ((溶接部位径の式 ..) (溶接部位径 C1 mm)
   ((スチール厚 DB mm) (スチール厚理め込み深さ H1 mm)
    (C1=2*(SQRT(DB*H1-H1^2))) (..) (..)))
```

(a)フレーム・オブジェクト「シューサブアッシー設計式」の定義

```
(defattribute 'has-formula-body :constraint-form
  (:has-formula-body-japanese-attribute-name "設計式本体"))
```

(b)制約フォームのデータタイプ指定子を持つ「設計式本体」のアトリビュート・オブジェクトの定義

図3 フレーム・オブジェクトとアトリビュート・オブジェクトの定義例

Fig. 3 An example of definition about frame object and attribute object.

```
(defattribute-flavor :constraint-form)
```

(a)制約フォームのデータタイプ・オブジェクトの定義

```
(defattribute-method execute-constraints-compiler
  :constraint-form () (execute-body))
```

(b)制約フォームのデータタイプ・オブジェクトに対し、制約コンパイラを起動し、制約ネットワーク上のデータ展開をさせるメソッドの定義

図4 制約フォームのデータタイプ定義例

Fig. 4 An example of definition about data object of constraint form type.

フレーム間に渡り行うことや、スーパーフレーム名からアトリビュートを経て直接アイテムにアクセスするという②が可能である。設計対象を表現する属性は、図2の場合、「主パラメータ」、および「従パラメータ」アトリビュートのアイテム (例えば、溶接部位径、ス

チールボール径，スチールボール埋め込み深さ)として記述される。こうした属性間の関係を示す制約関係の例が，設計式本体アトリビュートに含まれるアイテム，つまり設計式である。属性および制約関係をどう扱うかのデータ操作手続きが，先に述べた制約フォーム・データタイプ・オブジェクトに定義されたメソッドである。

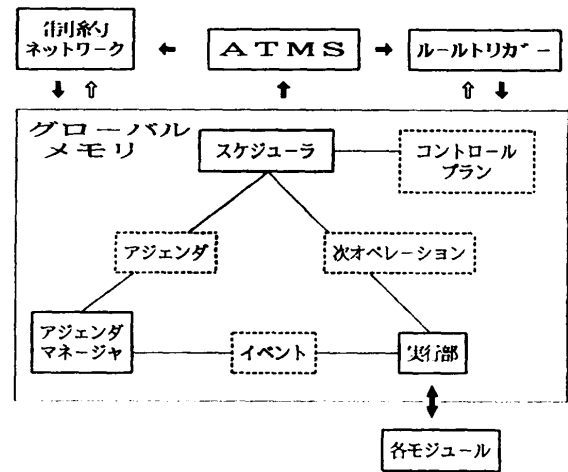
通常一つの機械部品を表現する設計属性を一つのスーパーフレーム内に収め，設計アクティビティというより抽象度の高い概念記述をしたスーパーフレームと多重継承関係を作り利用する。また設計式には，設計現場でよく使われている代数方程式を，設計式に対する制約条件アトリビュートには，代数不等式を，適用種別アトリビュートには，設計式選択，および計算結果に関する評価型ルール表現知識を入れる。また図2の参照実績データアトリビュートに含まれるアイテム，製品実績表は，過去の実績製品の属性データを収めるスーパーフレームへのポインタを示している。

一方，製品開発の企画段階で表れる品質—機能展開の品質特性，機能特性のように¹³⁾，設計対象を表現する属性の中には，代数方程式で示されるような制約関係を持たない属性があり，これらは例えば品質属性というスーパーフレームの中で実現される。この品質属性には，目的属性である要求仕様や，機能属性，実現構造属性，および幾何形状属性などのスーパーフレームが継承される。

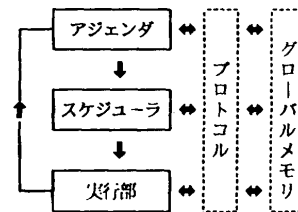
このような設計知識・データの構築，保守を支援するのが設計知識データ・編集モジュールである。このモジュールは，知的グラフィック・エディタとの連携により，幾何実体の抽出およびそれに対応した属性集合の特定作業という属性モデリング支援，さらに属性モデルのデータベース化という一連の作業に対するワークベンチの役割も果たしている。

2.2 設計プロセス制御モジュール

設計対象知識は，各設計者間で共通知識ベース化できるのに対して，設計操作，設計制御などの設計プロセス制御に関する知識は，設計者自身それを意識しにくいものであり，各設計者によって大きく異なる場合がある。また，各設計者のシステム依存度合いも実用当初から時間を経るに従い変化してくることも考慮すべき点となる。つまり，設計者によるシステムへの信頼獲得，およびシステムによる設計プロセス制御の利便性が求められるようになって，システムによる設計プロセス制御の割合を高めるといふ制御の柔軟性が必



(a) 情報の流れ
 →ATMSを使用する場合の情報の流れ
 ⇄ATMSを使用しない場合の情報の流れ



(b) グローバルメモリへのアクセス

図5 設計プロセス制御モジュール
 Fig. 5 Control module for design process.

要になる。

こうした点を踏まえ，図5(a)(b)に示す設計プロセス制御モジュールは，設計者の意思による設計プロセスのすみやかな遷移，設計者に対するリアルタイムなレスポンスを達成するために，ダイナミック・コントロール・アーキテクチャ^{14)~16)}の概念を受け継いでいる。設計プロセス制御モジュールは，設計作業に必要な情報を含んだグローバルメモリを持ち，モジュール自体が設計プロセス制御に関する知識ベースシステムであり，三段階の推論ステップ（アジェンダ・マネージャ→スケジューラ→実行部）を繰り返し行う。その基本動作として，アジェンダ・マネージャは，ユーザインタフェースからの入力イベント，推論ステップから発生するイベントにより次のタスク，あるいはオペレーションを特定する。スケジューラは，現在の制御プランに適合する次のオペレーションを選択する。実行部は，選択されたオペレーションに関した各モジュールを起動して個々のプロセスを開いた後，オペレーションを実行する。さらに制御プランも

含めグローバルメモリー内の情報を変更し、イベントを発行する。

この設計プロセス制御モジュールは、また他のモジュールを粗結合し、各モジュールを同時に実行させることができる。この同時に実行される各モジュールは、モジュールが実行されているセッションの始まりと終り、あるいはセッションの途中で、グローバルメモリーに記録、あるいはグローバルメモリーを見ることで、非同期にお互いに影響し合う。設計者が、他のモジュールを直接起動したい場合、アジェンダ・マネージャにその旨のイベントが発せられ、アジェンダ・マネージャはそれに対し高いプライオリティを付けたアジェンダをスケジューラに送る。スケジューラは、すでに実行すべきオペレーションが待機していても、この高いプライオリティのアジェンダに対応するオペレーションを実行するように実行部に指示する。このことは、設計者の意図に対して即時の反動的な応答をシステムが行うことになる。

グローバルメモリーのデータ保存構造設計に当たって、データ保存を最小限にして、さらに特定のデータの種類に対してロックアップタイムを最小にするために、用途に応じたデータ構造をインプリメントできるように留意した。標準は、ディスクリミネーション・ネットである。異なったデータ構造、例えばハッシュテーブル、アレイなどを共存させることもできるし、個々の問題に合わせてベストなデータ表現を選択することができる。このことは、これまで均一のデータ構造にするため、汎用の構造フレームだけを用意していたが、アーキテクチャ的には、シンプルになるものの、システムが実用レベルになると多量のデータを扱うことになり、パフォーマンス上の問題が出るためである。こうしたことは、エキスパートシステムのライフサイクルを考慮した場合、単調に増加していく知識・データに対して、ライフサイクルの何れかの時点において、パフォーマンスのチューニングが予想されるための方策である。また、このグローバルメモリーには ddb-guided ATMS¹⁷⁾⁻¹⁹⁾を装備し仮説展開型推論のサポートを行う。ddb-guided ATMS は、設計の解探索において、適切な選択肢を積極的に推奨し探索を制御できる場合に有効である。機械設計の場合、実績に基づいた選択肢を推奨する知識があり、これによって効果的な解探索を行う。一方、大きな設計問題の中の部分問題、例えばある部品設計において、その部品の制約パラメータ、制約式がすでに知識として完備さ

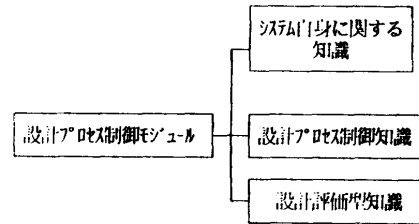


図 6 設計プロセス制御モジュールを構成する知識
Fig. 6 Decomposition of knowledge for controlling design process.

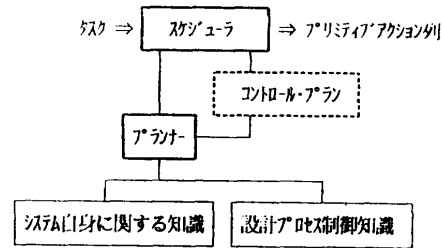
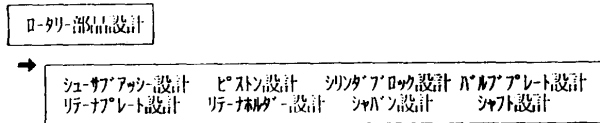


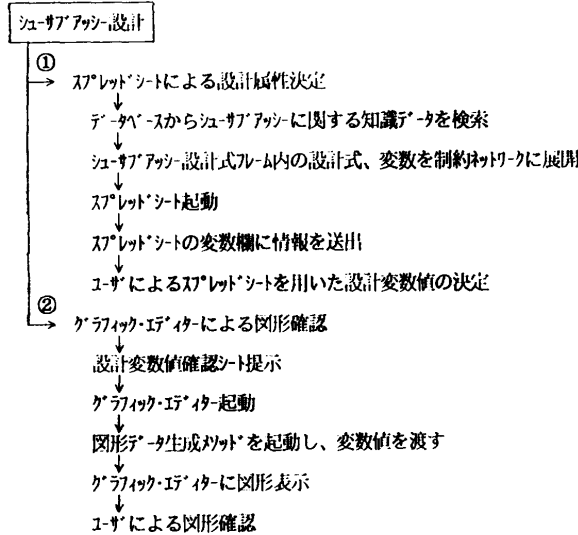
図 7 スケジューラにおける知識の使われ方
Fig. 7 Using knowledge for scheduler.

れており、他の部品との相互作用がほとんどない場合には、上記 ATMS を使用せずに LTMS 機能²⁰⁾だけを使い、制約ネットワーク上の制約伝播で問題解決を行う。こうした問題解決の戦略をとるのは、主にパフォーマンス向上のためであり、ルーチン設計と呼ばれている分野で使われる。

設計プロセス制御モジュールの知識ベースには、図 6 に示すように、システム自身に関する知識と、設計プロセスをどう制御するのかという知識、さらに過去の設計の失敗事例をもとにした設計評価型知識から構成される。システム自身に関する知識は、システムの持つリソースおよびネットワーク上の他のエキスパートシステムに関する知識である。これらの知識の使われ方をスケジューラについて説明したのが図 7 である。アジェンダ・マネージャからの入力タスクである場合、プランナーを使いそのタスクを実行するためのコントロール・プランを生成させ、実行部に対しオペレーションとしてプリミティブ・アクション列を送り出す。この際、プランナーはシステム自身に関する知識、設計プロセス制御知識を使用する。具体的には、図 8 の (a) に示すように、ロータリー部品設計というタスクの場合、それを八つの設計アクティビティに詳細化し、一つの任意のアクティビティをユーザに選択してもらい、その具体的なコントロール・プランを生成しようとする。図 8 (b) にシューサブアッシー設計というアクティビティに関する生成されたコントロー



(a) 詳細設計アクティビティへの展開例



(b) コントロールプラン生成例

図 8 プランナによるプラン生成例

Fig. 8 An example of plan generation by the planner.

ル・プランを示している。シュ-サブアッシ-設計というゴールに対して、図8中①、②で示されるサブゴールが生成され、それを達成するためのアクション列が作られる。実際には、図5の実行部が理解できるほどに、より詳細なプランが作られプリミティブ・アクション列として実行部に送られる。先に述べた“設計者の意思による設計プロセスのすみやかな遷移”とは、スケジューラがこうした一連のコントロール・プランを実行させようと実行部に命令を送っている最中に、例えばユーザがデータベース・アクセスなどをシステムに要求するなどに必要な、実行プライオリティの動的変更が可能であることを指す。

2.3 知的グラフィック・エディタ

知的グラフィック・エディタは、機械部品の形状に関する属性モデルをパラメトリック操作可能なグラフィック・アイコン化して幾何形状操作に使用するものである。グラフィック・アイコンを設計者が特定し、それを相互に結び付けることによって、逆に、属性モデルを特定して設計者の意図する対象に関する知識として、属性モデルのインスタンス集合を計算機内部で

表 2 動力伝達用平歯車の歯形属性
Table 2 A feature of tooth profile for spur gear.

モジュール
歯数
工具圧力角
歯先円直径
歯底円直径
転位係数
工具歯形歯先丸味半径
内径
トッピング歯先円方向
トッピング半径方向

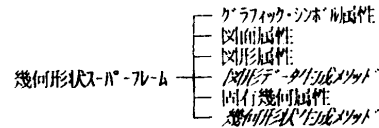


図 9 幾何形状スーパーフレームの主要構造
Fig. 9 Decomposition of super-frame for geometrical form.

作り出し、それに基づいた知識処理を施す。こうした知識処理可能な従来 CAD という特徴を持たせるために、インプリメンテーション・レベルでは、機械図面作成のためにこれまでの CAD が持っていたほとんどの機能を、オブジェクト指向言語で実現し、さらに知識処理可能なように各オブジェクトの機能を拡張している。また、システムに属性知識がなくて、設計者が図面を描く場合、システムとして設計者のタスクはそのまま実行してもらい、後で属性知識を入れてもらうオープンシステムにしている。この属性モデルの例として、機械部品でよく見られる動力伝達用平歯車の幾何形状である歯形属性についてみると、表2のようになる。これらの属性によって、歯車の歯プロフィールはインボリュート曲線でユニークに表現される。歯車の属性は表1の歯形属性だけではなく、歯車性能、面圧・曲げ強度、製作誤差・精度、干渉、歯形修正、測定などに大別される属性群からなる。知的グラフィック・エディタでは、こうした歯車の例に対して、歯車の図9に示す幾何形状スーパーフレームを用意し、その複合オブジェクトの中には歯車のグラフィック・シンボル属性（これはグラフィック・アイコンとしてユーザ・インタフェースに利用）、図面としての図面属性、さらに表2に示した歯形属性を含む固有幾何属性を持っている。歯車の場合、通常、図面としては完全な全体形状を表記せず、簡略表記をする。こうした方法は、スプライン形状の

伝達軸などでも見られる。これらの図面としての表記は、図面属性の中にはいる。有限要素法解析のように、歯形の詳細な形状を必要とする場合には、歯車の幾何形状スーパーフレームの幾何形状生成メソッドとして用意してあるインポリュート曲線生成メソッドを起動し、表2に示した歯形属性値を使い、幾何形状データを生成する。図9の図形属性、図形データ生成メソッドは、通常のCADで見られるパラメトリック作図のためのパラメータと生成器である。

2.4 知的スプレッドシート

制約指向言語と結びついたスプレッドシートは、知的スプレッドシートとして、ファイナンシャル分野²¹⁾で提案されている。ここで、制約とは、拘束関係を考慮しなければならない属性および属性間のことを指す。また機械設計分野で使用されるパラメータとは、属性の部分集合である。設計対象になる製品について、それを構成する部品、および機構が過去にない新規設計でないかぎり、かなりの部分が様々なパラメータおよびパラメータ間の拘束関係によって記述されている。これは、具体的には、実績ある部品、機構という類に当たる。実際、設計ノウハウと現場でいわれる知識は、こうしたパラメータの不等式、パラメータと幾つかの述語表現（例えば、near-member、これはパラメータが数値列に含まれる数値に近いものを取るといふ述語）、パラメータ間の拘束関係を示す設計計算式（例えば、機械構造の理論解析式、実験式、有限要素法解析でもある負荷に対して形状パラメータを変化させた場合の発生応力の回帰式を作成するなど）、フィールドで実績のある製品とパラメータ値との表、パラメータおよび設計式の使い方に関するヒューリスティクスからなる。機械設計では、扱うパラメータの数、パラメータ間の相互作用が多く、量的な問題が解決されれば、それらを知的スプレッドシートの形で使用し、設計作業を進めることは有効と考えられる。

本知的スプレッドシートは、ネットワーク型制約指向言語²²⁾と結びついており、大量のパラメータ、セル、柔軟なシート構成が扱えるようになっている。設計者はスプレッドシート上に表れるパラメータを操作することで、設計を進める。ネットワーク型制約指向言語は、スーパーフレーム上に宣言的にかかれた制約パラメータ、制約関係、制約条件をコンパイラにより自動的に制約ネットワークに展開する。ネットワーク上を制約が伝播して、未決定のパラメータは決定されたパラメータから計算され、ネットワークに展開したこ

とによって双方向計算機能を実現されている。この制約指向言語の基本的な戦略は、局所的伝播とその中で局所的無矛盾性を確保することである。ネットワーク型制約指向言語には、上記コンパイラ、ネットワークを視覚的に見ることのできるグラフィック機能、説明機能、矛盾情報を含めたステータス機能、デバッグ機能などを持つワークベンチが用意されている。

ネットワーク型制約指向言語において、制約ネットワーク上のパラメータに対するアクセスは任意の方向からできるため、ユーザ主導型のシステムが実現できる。つまり、ユーザである設計者は、最初に設計しようとする機能、部品、あるいは部品集合は、製品の持つどれでもよく、その後の設計プロセスで取り上げる機能、部品あるいは部品集合の順序付けは存在しない。例えば、図8(a)のロータリー部品と言われているものは、流体を介した動力伝達、あるいは機械的動力伝達を行うため、ピストン径、ピストン数などのパラメータが、シリンダブロック・ボア間距離、シリンダブロック PCD などのパラメータに直接影響する。このように、ある部品を表現するパラメータのネットワークは、他の部品のネットワークに対してインタフェースとなるパラメータを介した制約伝播を双方向に行っているために、先の順序付けがいらなくなる。

こうした知的スプレッドシートで決定されたパラメータは、先の図9の図形データ生成メソッドに渡され、グラフィック・エディタ上で視覚的に図形を確認できる。一方、逆に設計者がその図形を変更した場合には、変更された図形属性がグローバルメモリーに反映され、その後制約ネットワークに反映される。こうした、制約ネットワークに関する情報の流れを示したのが図10である。設計プロセス制御モジュールのグ

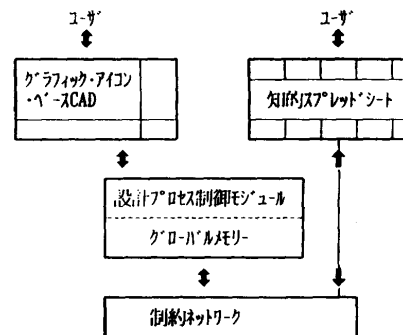


図10 制約ネットワークに関する情報の流れ
Fig. 10 Relationships among a constraint network, a control module for design process, a CAD interface, intelligent spread sheets.

ローバルメモリーに、局所的伝播が終了することに制約パラメータに関する情報を伝える。逆に、グラフィック・アイコン・ベース CAD で決定された属性情報で制約パラメータに関するものについては、局所的伝播が行われるセッションの前後の機会にグローバルメモリーを見に行き、制約ネットワーク上に反映させる。

3. おわりに

本システムの大部分は、オブジェクト指向言語である Flavors (Symbolics 社) で記述されているが、制約指向言語は Flavors 自身と同じレベルの Common-Lisp 言語でインプリメントされている。ハードウェアは、Symbolics 3600 シリーズ、および Symbolics XL 400 である。

本論文では、機械設計支援エキスパートシステム構築シェル: MAGIC について、基本概念、システムアーキテクチャについて述べた。この MAGIC を用いて、油圧ポンプ・モータ設計支援エキスパートシステムの構築を行った。このエキスパートシステム構築によって、次の効果、

- ①構築されるエキスパートシステムが、そのライフサイクルの過程で直面する知識、データの増加に対し、パフォーマンス向上のために柔軟に対処できること
 - ②知識、データの追加・維持管理を設計者自身に実施してもらうための保守性向上
 - ③設計支援エキスパートシステム構築の生産性向上
- などが確認された。これには、専用推論エンジンとしての制約指向言語をもとにした知的スプレッドシートが②、③に、設計プロセス制御モジュールが①に、知識ベースとデータベースとを一元化したスーパーフレーム言語が②、③に寄与している。一方、問題点として、属性モデリングの際、物理的な物の特徴に対応する属性名を付けるのが困難な場合があること、および設計者間で属性名に対するコンセンサスを取る必要があることがわかった。これに対し、機械的な属性名割り付けを行うと、設計者の記憶に残りにくいという面もあり、今後検討していきたい。また機能面では、MAGIC に定性推論エンジン、幾何推論エンジンなどの専用推論エンジンの種類を増やし、より効率的なシェルを開発していきたい。

謝辞 九州工業大学の長澤教授、および東京工業大学の伊藤氏には属性モデリングについて教示いた

いた。大阪大学産業科学研究所の溝口教授には ddb-guided ATMS について教示いただいた。これらの方々に深くお礼申し上げる。

参考文献

- 1) 長澤, 古川: 拘束条件リダクション法を用いた機械設計計算支援システム, 情報処理学会論文誌, Vol. 27, No. 1, pp. 112-120 (1986).
 - 2) Ito, M.: CONMOTO a Machine Part Description System Based on Designers' Mental Processes, in Yoshikawa, H. ed., *Design Theory for CAD*, North-Holland (1987).
 - 3) Cunningham, J. J. and Dixon, J. R.: Designing with Features: The Origin of Features, *Proc. of the 1988 ASME International Computers in Engineering Conference*, pp. 1-10, ASME (1988).
 - 4) Chandrasekaran, B.: Generic Tasks in Knowledge-based Reasoning; High-level Building Blocks for Expert System Design, *IEEE Expert*, pp. 23-30 (Fall 1986).
 - 5) Brown, D. C. and Chandrasekaran, B.: Plan Selection in Design Problem-Solving, Laboratory for Artificial Intelligence Research Technical Report, Ohio State Univ. (Apr. 1985).
 - 6) Steinberg, L.: Design as Refinement Plus Constraint Propagation: The VEXED Experience, *Proc. of the National Conference on Artificial Intelligence*, pp. 830-835, AAAI (Aug. 1987).
 - 7) 高度技術化に対応する機械製図システムの標準化のための調査研究報告書, 日本設計工学会 (1989).
 - 8) 伊藤: CAD システムの機能評価とこれからの見通す技術, PIXEL 10月-12月号, 図形情報処理センター (1989).
 - 9) Nakashima, Y. and Baba, T.: OHCS: Hydraulic Circuit Design Assistant, *Proc. of IAAI-89*, AAAI, pp. 124-130 (1989).
- 馬場, 中島: 仮説展開手法を用いた設計エキスパートシステム油圧回路設計への応用例, 第36回情報処理学会全国大会論文集, pp. 1565-1566 (1988).
- 10) de Kleer, J.: An Assumption-Based TMS, *Artif. Intell.*, Vol. 28, No. 2, pp. 127-162 (1986).
 - 11) 長澤: 設計エキスパートシステム, 情報処理, Vol. 28, No. 2, pp. 187-196 (1987).
 - 12) 中島, 田原, 加藤: 機械設計支援エキスパートシステム: MAGIC (2), 第40回情報処理学会全国大会論文集, pp. 249-250 (1990).
 - 13) 宮村, 土井, 中根, 安田: 新製品開発における情報の活用と伝達, 品質, Vol. 18, No. 4, pp. 34-54 (1988).

- 14) Hayes-Roth, B.: A Blackboard Architecture for Control, *Artif. Intell.*, Vol. 26, No. 3, pp. 251-321 (1985).
- 15) Hayes-Roth, B. and Hewett, M.: Building Systems in the bbl Architecture, in Engelmores, R. and Morgan, A. eds., *Blackboard Systems*, Addison-Wesley, London (1988).
- 16) Hayes-Roth, B.: Intelligent Monitoring and Control, *Proc. of the International Conference on Artificial Intelligence*, IJCAI-89 (1989).
- 17) de Kleer, J.: Back to Backtracking: Controlling the ATMS, *Proc. of AAAI '86*, pp. 910-917 (1986).
- 18) 西岡, 池田, 堀, 溝口: ATMS の拡張とその応用, 人工知能学会研究会資料, SIG-KBS-8805-5, pp. 41-50 (1989).
- 19) 西岡, 溝口: ATMS のラベル遅延更新アルゴリズム, 第 39 回情報処理学会全国大会論文集, pp. 327-328 (1989).
- 20) McAllester, D. A.: A Three Valued Truth Maintenance System, AI Memo 473, M. I. T. Artificial Intelligence Laboratory, Cambridge (May 1978).
- 21) Fischer, G. and Rathke, C.: Knowledge-Based Spread-sheets, *Proc. of the National Conference on Artificial Intelligence*, AAAI-88, pp. 802-807 (1988).
- 22) Sussman, G. J. and Steele, G. L., Jr.: Constraints—A Language for Expressing Almost-Hierarchical Descriptions, *Artif. Intell.*, Vol. 14, No. 1, pp. 1-39 (1980).

(平成 2 年 2 月 8 日受付)
(平成 2 年 9 月 11 日採録)



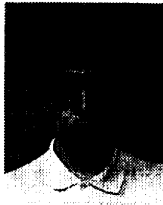
中島 裕生 (正会員)

1951 年生. 1976 年東京工業大学大学院材料科学専攻修士課程修了. 同年カヤバ工業(株)技術研究所勤務. 1990 年よりニチメンデータシステム(株)勤務. 現在, 知能情報システム部長. 知識工学応用システム, オブジェクト指向データベース応用開発に従事. 著書「計算力学と AI」(養賢堂, 共著). AAAI, 人工知能学会, OR 学会各会員.



田原 一徳 (正会員)

1958 年生. 1982 年長崎大学材料工学科卒業. 同年カヤバ工業(株)入社. 現在, カヤバ工業(株)技術研究所勤務. エキスパートシステム開発に従事.



加藤 亨 (正会員)

1963 年生. 1987 年日本電子専門学校国際コンピュータ技術研究科卒業. 同年カヤバ工業(株)入社. 現在, カヤバ工業(株)技術研究所勤務. エキスパートシステム開発に従事.