

3C-3

設計プランに基づく設計タスク向き 知識表現モデルの検討

三樹 弘之 岩根 典之 木下 哲男
 沖電気工業株式会社 総合システム研究所

1. はじめに

コンピュータの適用範囲が広がるにつれ、近年、設計という創造的な活動に対してもシステム化のアプローチがなされている。人工知能の分野では、これは主に設計型知識表現言語及びそれを用いたエキスパートシステムという形で、設計プロセスの様々な面を统一的にモデル化する試みとして研究されている[1]。

我々は現在タスク指向アプローチに基づき、設計に特殊化された知識表現、推論機構を提供する設計タスク向き知識表現モデルについて検討を行なっている。そこでは、知識表現モデルの設計にあたり、設計の知識の多面性を促える為に設計プランという知識表現形式を導入し、様々な設計知識を集約する。これにより、異なる種類の設計問題に対して包括的な表現を提供できる。

本稿では、設計プランを用いた設計タスク向き知識表現モデルの、設計プランの表現形式、及びこの利用により構成される設計プロセスの概略について述べる。

2. 設計プラン表現の導入

従来、多くの設計エキスパートシステムでルール型の知識表現が用いられてきた。ルール表現は、簡単でしかも設計においても主要な役割の一つである状況駆動を提供するので、ある程度の有効性は発揮できる。しかし、設計においてはこれ以外に制約知識や知識の依存性が複雑に絡み合う。ルールでは知識をまとめる単位が細かすぎ、しかもサポートするメカニズムが少なすぎる。

一方、従来人工知能においては計画などのプロセスを構築する知識表現としては、プランという表現形式が用いられていた[2][3]。この枠組みは、知識の組み合わせ方を表現する枠組みとして適当であり、この面から、プラン表現を設計問題に適用する試みがなされている[1]。本モデルでは、プラン表現を採用し(図1)、ルール表現の欠点を効果的に補うことを目指している。

3. 設計プランに基づく設計知識の表現

3.1 設計方針

Knowledge Representation Model for Design Task based on Design Plan

Hiroyuki Miki, Noriyuki Iwane, Tetsuo Kinoshita
 Systems Lab., OKI Electric Industry Co., Ltd.

設計プロセスに対して設計プランを導入するために、以下の設計方針に基づいて設計プラン表現モデルを構築する。

- 1) 広くさまざまな設計をカバーする、包括的で柔軟な表現モデルとする。このための表現形式は3.4で述べる。
- 2) 多くのタスクベースツールと同様に、プロセスを構成するメインメカニズムは単純にし、そのサブメカニズムに多くのプリミティブ(タスク依存の機能を提供)を導入する。

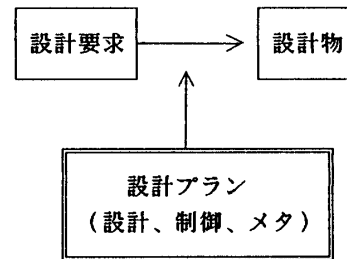


図1 設計と設計プラン

3.2 設計プロセスモデル

基本的にはプランの繰り返しの適用によって設計は進むが、本モデルではプランから次のプランを直接呼び出すことはしない。図2のように間にゴールを設定し、プランによる(サブ)ゴール生成、ゴールを1つ選択、当該ゴールに適するプランの選択・実行といったプロセスを踏む。ゴールの選択により「何をすべきか」を、プランの選択により「それをいかに行なうか」を判断する。

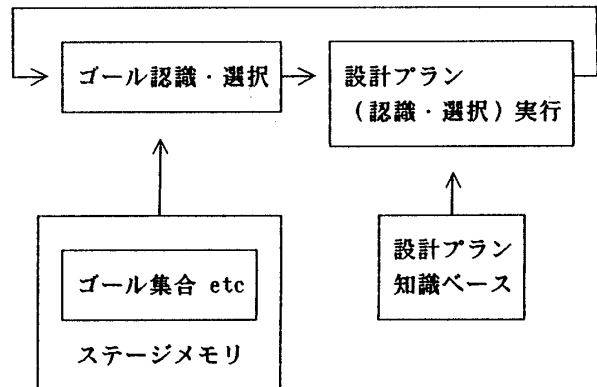


図2 ゴールと設計プランによる設計プロセス

3.3 設計プランによる統一的な知識表現形式

設計プランは10組の項目で定義される。

```
D_Plan = < DP_ID,   Type,
           Category, Goal_Statement,
           Constraint, PreCondition,
           Post_Condition, Sub_Goal,
           Action,   InterPlanRelation >
```

・DP_ID: 設計プラン名

・Type: プリミティブ/複合(ノコンパイル)

サブゴールの呼び出しを行わないある設計対象(のクラス)に対する一意な設計手順を表わした最も基本的なプランをプリミティブプランと呼ぶ。一方、自身のアクションを実行する前に実行すべきゴールが存在する(サブゴールの呼び出しを行う)プランを複合プランと呼ぶ。実行前にゴール・プランの参照関係が規定されている複合プランは、特にコンパイルプランと呼び、参照するプランとのリンクを張っておく。Typeは可読性のために付与する。

・Category: 設計/制御/メタ

当該プランで表現する知識の種類を示す。3.4 参照。

・Goal_Statement: 当該プランの起動のトリガーとなるゴールに関する情報を記述する。一つのゴールに対する設計プランの認識と選択の方法については、[4]を参照のこと。

・Constraint: 当該プラン実行中の制約を記述する。副次的に呼び出されるプランにも、この制約は引き継がれる。方程式で表わされる定量的制約、ならびにルールで表現される定性的制約の2種類が記述される。

・PreCondition: 当該プランの起動のための種々の条件を記述する。変数の未知/既知に関する指定や制約の設定/未設定に関する条件などを記述する。

・PostCondition: 当該プランの実行により生じる実行後の状況を記述する。値が決定される変数を明記したり、プロセス制御や他のプランの実行に影響を与える、グローバルな効果を記述する。

・Sub_Goal: いくつかの副次的な設計プランを設計サブゴールの生成により指定する。呼び出すゴールに対する制御情報として、独立、競合、協調、干渉などの情報が付加できる。たとえば協調ゴールのときには、(Cooperation (subgoal1..) (subgoal2..) ...) と記述する。このゴールの関係により判明する最適な戦略[5]は、その後の制御ステージで利用される。

・Action: Sub_Goal 実行後に行なう、当該ゴールを充足するための設計知識(手続き群など)を記述する。これ自身をプリミティブプランとして独立に記述して当該プランのSub Goalでコールしても構わない。

・InterPlanRelation: 設計知識相互間で成立する関

係に応じた上位-下位、全体-部分などの一般的な知識構造関係、あるいは設計知識の持つ類似性、因果性、排他性、代替性など、設計ドメインの知識に応じて設定される種々の相互関係を記述する。

3.4 プランで表現される3種類の知識

1) 設計知識

設計物を設計するための知識を、複合プランやプリミティブプランによって表現する。設計物をパラメータで表現した場合は、パラメータ値を決定するための一連の知識が表現される。

2) 制御知識

主にゴールの認識・選択方法を制御する戦略知識を記述する。主に一つのプリミティブプランで一つの制御知識を表わし、設計知識と違って設計サブゴールの生成による副次的な設計プランの呼び出しは行なわないことが多い。

3) メタ知識

要求仕様のわずかな違いや起動条件を緩和したりする場合に、設計知識(プリミティブ、複合)を修正したり、設計知識を組み合わせて新たに設計知識(複合設計プラン)を合成したりする知識である。

3.5 設計プランの利用

あらかじめ設計物の全体構造が明確となっており、設計においては部分構造のみを詳細化する様なルーチン設計では、一般に自動実行が可能なので、設計知識のみをプリミティブ、複合プランの組み合わせとして表現する。専門家と同じシーケンスを好んだりインタラクティブな使用をする時は制御知識を付与する。

4. おわりに

本論文では、設計プランを用いた設計タスク向き知識表現モデルの、設計プランの表現形式、及び、これを利用して構成される設計プロセスの概略について述べた。3種類のプランのシンタックス、制約処理を初めとした細かな制御方式などを詳細化し、事例に基づいて検証することが今後の課題である。

参考文献

- [1] Brown, C., et al.: Knowledge and Control for a Mechanical Design Expert System, IEEE COMPUTER JULY(1986)
- [2] Wilensky, R., : Meta-Planning, Cognitive Science No.5(1981)
- [3] Stefik, M., : Planning with Constraints, Artificial Intelligence, vol.16, No.2(1980)
- [4] 岩根、三樹、木下: 設計プランに基づく設計過程におけるプラン認識法の検討、本大会予稿集
- [5] Mostow, J., : Toward Better Models of The Design Process, AI Magazine Spring(1985)