

設計作業支援エキスパートシステムの知識表現

1 D-9

西野 篤夫

鹿島建設株式会社

1. はじめに

設計作業はエンジニアにとって本質的な仕事である。特に設計の初期の段階である概念設計段階は、設計者の経験と専門知識を最も必要とする、知識集約的色合いが強い作業である。その重要性にもかかわらず、この段階のコンピュータによる支援は、従来、数値計算やCAD的な作業のみに偏っていた。しかし、近年のエキスパートシステムの登場が、より本質的な設計業務のコンピュータ支援を可能とした。筆者は、概念設計段階の設計作業を支援するエキスパートシステムのシェル CONGEN (CONcept GENerator) を開発した。本稿では、CONGENで採用した設計モデルと設計知識の表現方法について報告する。

2. 設計モデル

設計問題は、一般に制約条件で表現される。制約とは、設計対象全体あるいはその一部に対して取りうる値の範囲を規定したり、評価の基準を提供するものである。設計問題を解くという行為は、制約を全て充足する最適な解を見つけることである。

また、設計作業をシステムティックに行うために、設計問題は小さな部分設計問題に分解することが可能であると仮定した。分解は、部位、部品、機能などにより行われる。全体の設計問題は、分解された小さな部分設計問題の集合として捉える。そして、各部分設計問題を、ある順番で解くことにより全体の設計問題が解かれると考える。

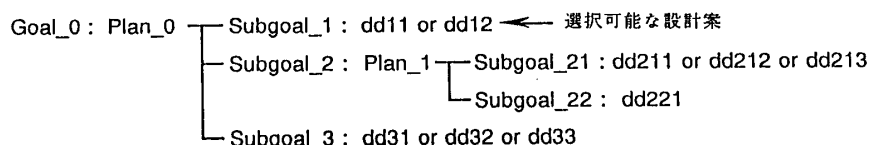


図-1 部分設計問題への分解

制約_1: IF dd11 is selected THEN dd212 is infeasible.
 制約_2: IF dd12 and dd212 are selected THEN dd32 is infeasible.
 制約_3: IF sp1 is specified THEN dd33 is infeasible.
 制約_4: IF sp1 is specified THEN dd213 is infeasible.

図-2 制約

Specification 1: sp1 is specified

図-3 初期入力条件

3. 設計知識の表現

CONGENは、設計作業のうち、制約を全て充足する解の集合を見つけるところまでを支援する。解の集合から最適解を見つける作業は、対象とする設計問題に大きく依存するため、汎用化するのが難しく、よってエキスパートシステムのシェルであるCONGENに組込むことを見合わせた。ここでは、CONGENで採用した設計知識の表現方法について、簡単な設計問題を例に説明する。

3.1 部分設計問題への分解

設計問題が部分設計問題に分解される場合、分解の方法及び各部分問題を解く順序は明示的に記述される必要がある。分解方法を記述するため、CONGENはゴール、プラン、サブゴールによる階層表現構造を用意した。図-1は、設計問題が部分設計問題に分解される場合の例を示す。Goal_0は設計問題全体を示しており、Goal_0を達成することがこの設計問題の最終目的である。このケースでは、Goal_0は部分設計問題Subgoal_1~3により達成される。部分設計問題への分解はPlan_0によって記述される。Plan_0はGoal_0を解くために満たすべき部分設計問題とその解放の順序を示す。

さらに、部分設計問題Subgoal_2はSubgoal_21とSubgoal_22に分解される。よって、Goal_0を達成するために、Subgoal_1 → Subgoal_21 → Subgoal_22 → Subgoal_3の順にサブゴールが達成される。

各サブゴールの達成は、そのサブゴールに対して設計案を見つけることにより行われる。各サブゴールで選択可能な設計案を、図中では各サブゴールの後ろに示した。

また、プランの変更が設計の途中で生じるような設計問題に対応するため、CONGENでは、プランを変更するためのルール記述が用意されている。

3.2 制約の記述

CONGENでは、制約はIF-THENルールで記述される。ルールの条件部には、数値的な条件式と記号的な条件式を記述することができる。また、結論部には、条件が成立したときに選択可能とする設計案(feasible)、条件が成立したとき唯一選択可能とする設計案(only_feasible)及び条件が成立したとき選択不可能とする設計案(infeasible)を記述することができる。図-2は、この設計例で使われる4種類の制約を示す。

3.3 初期入力値

設計を進める上で必要となる初期条件は、初期入力値としてCONGENに与える。図-3は、初期入力値としてSP1が宣言されていることを示す。

4. 設計案の生成

CONGENは、ゴール、プラン、サブゴールにより決定されるサブゴールの達成順に従って、制約を充足しながら各サブゴールの設計案を決定して行く。そして、サブゴールと採択された設計

案から木構造のDesign Solution Treeを生成する。図-4は、例題に対するSolution Treeを示す。図中で、□はゴール・ノードを、○は設計案のノードを示す。図中で、枝分かれしているゴール・ノードは、そのゴールに対して複数の設計案があることを示している。

設計問題に対する最終的な答えは、各サブゴールで選択された設計案の集合として表現される。最終設計案は、末端から最上層のノードまでSolution Treeを逆さにたどり、途中で遭遇するサブゴールの設計案を集めることにより見つけることができる。

5. 終わりに

概念設計作業を支援するエキスパートシステムのシェルCONGENの知識表現形式と設計案の生成方法について報告した。CONGENでは、設計モデルとして設計問題の部分問題への分解および制約の充足機構を採用した。

CONGENはAllegro Common Lispを用いて開発し、さらに建築物の構造設計問題に適用してシステムの有効性を確認している。

最後に、本研究を実施するにあたり、ご指導いただいた、マサチューセッツ工科大学Intelligent Engineering Systems Laboratoryのスリラム助教授に感謝いたします。

6. 参考文献

- 1) Pahl, G. and Beitz, W., *Engineering Design*, The Design Council, UK, 1984
- 2) Sriram, D., *Knowledge-Based Approaches for Structural Design*, Computational Mechanics Publications, UK, 1987

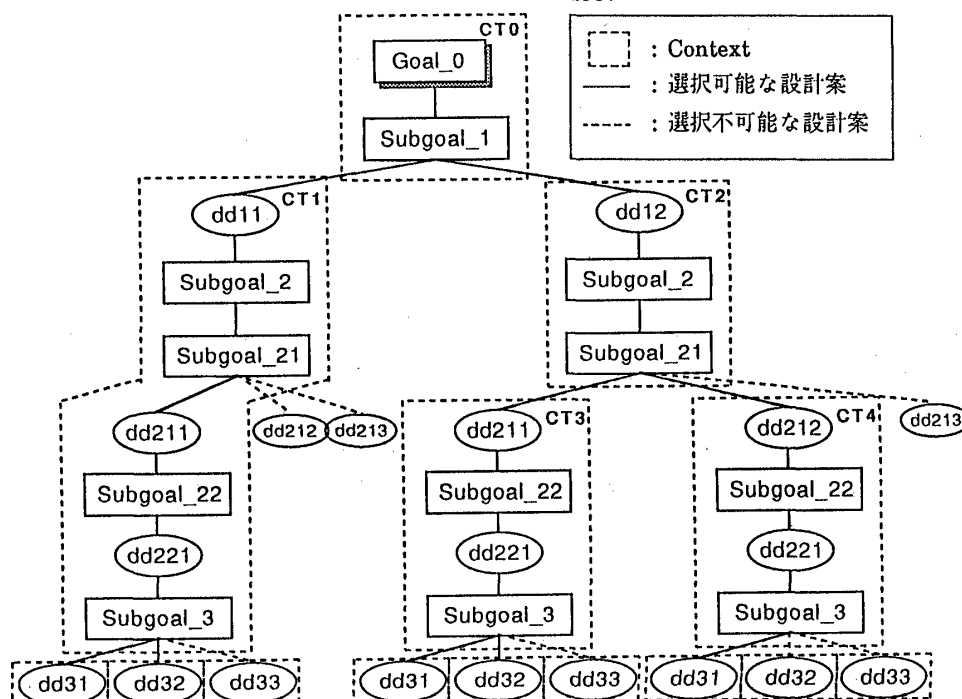


図-4 Design Solution Tree