

1. はじめに

Product Configuration とは、製品の構成要素である部品の定義や、接続関係などの部品間の制約条件が与えられたとき、ユーザの要求を満足する部品の組み合わせを求める問題である。近年、電子商取引の分野の発展や、ユーザの要求の多様化に伴い、この問題を対象とした研究が脚光を浴びており、様々なシステムやモデルの提案が行われている [3]。

人工知能の分野において、Configuration 自体の研究の歴史は古く、1980 年代初頭には、プロダクションルールを用いて、コンピュータシステムの Configuration を行う R1/XCON システムが提案されている [2]。XCON システムは、R1 の提案後も様々な改良が施され、実世界へ応用されたエキスパートシステムの成功例として認知されるようになった。しかしながら、以下に示すような XCON システムの問題点も明らかになった。

- 知識管理のコストが非常に高い
- システムの設計が、問題領域に強く依存しているため、他の問題領域への再利用が困難

例えば、最初の問題に対し、1989 年の XCON システムの知識ベースには、約 31,100 のコンポーネントと約 17,500 のルールが格納されているが、年間約 40% の知識の更新が必要であることが報告されている [1]。

Product Configuration に対する他の問題解決の手法として、制約充足を用いた研究も数多く行われている [4]。制約充足を用いた Configuration システムは、問題領域とは独立に問題解決を行うことができるため、再利用の容易性などの特長を持つ。しかしながら、多岐にわたるユーザの要求を満足する Configuration を行うためには、制約の重みや目的関数などを定義する必要があり、これらの知識の獲得や管理には高いコストがかかる

Product Configuration using Case-Based Reasoning
- CBR as a Support Method for Problem Solving -
Seishi Okamoto, Hiroya Inakoshi, Yuiko Ohta and Nobuhiro Yugami
FUJITSU LABORATORIES LTD.
E-mail: seishi@flab.fujitsu.co.jp

という問題を抱えている。

本論文では、CBR(Case-Based Reasoning) と制約充足を組み合わせることで、Product Configuration を行う新しい問題解決の枠組みを提案する。ここで、実際の Configuration は制約充足が行い、CBR は、過去の Configuration の事例を利用することで、目的関数などを自動的に生成する問題解決支援の役割を果たす。

2. 従来の CBR システム

CBR システムの一般的な枠組みについて、主な構成要素および問題解決の流れを図 1 に示す。

一般的な CBR システムでは、問題が入力されると、過去の問題解決事例が格納されている事例ベースから、入力された問題に類似した事例が検索される。次に、検索された類似事例の解を、現在の問題に適合するように修正を行い、解の出力を行う。ここで、解の修正に失敗した場合は、失敗の原因の解析などを通して、解の修復を行い、再び解の出力を行う場合もある。

このように、CBR システムは、類似事例に基づいて問題解決を行うため、ユーザの要求が不完全な場合でも、解の導出を行うことができるという特長を持つ。このユーザ支援に関する特長は、Product Configuration においても重要であり、CBR を直接利用した Configuration システムの提案も行われている [5]。しかしながら、従来の CBR を用いた Configuration システムは、解の修正を行うための知識の獲得や管理に、ボトルネックが生じてしまうといった問題を抱えている。

3. 提案手法

本論文では、CBR を問題解決自体に使用するのではなく、問題解決を行う制約充足器のラッパーとして使用する。このように、問題解決支援として使用する CBR のことを CBR Wrapper と呼ぶ。

Product Configuration に対して、本論文で提案する新しい問題解決の枠組みの構成要素と問題解決の流れを図 2 に示す。ここで、CBR Wrapper は、ユーザの要求を、実際の Configuration を行う制約充足器の入力へと

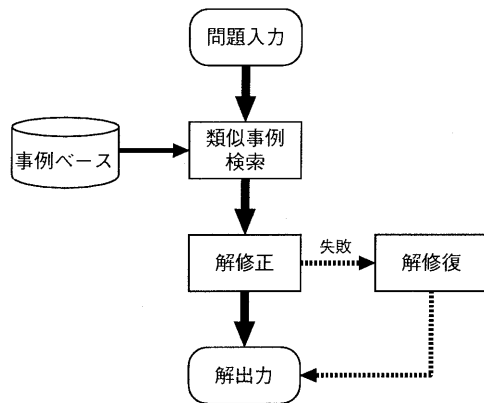


図 1: CBR による問題解決

変換する問題解決支援の役割を果たす。また、制約充足器は、目的関数と制約条件が与えられたときに、制約条件を満たし、目的関数の値を最大（最小）化する Configuration を求める、問題領域に依存しない問題解決器である。

まず、CBR Wrapper の各構成要素について、その機能的な役割の説明を行う。

事例ベース：過去に行われた Configuration の事例が格納されている。

類似事例検索部：ユーザの要求に類似した過去の Configuration の事例を事例ベースから検索する。機能的には、従来の CBR における類似事例検索と同じである。

問題補完部：類似事例を使用することで、ユーザの要求から、制約充足器の入力への変換を行う。具体的には、目的関数の作成などを行う。

問題修復部：制約充足器において、要求や制約を満たす Configuration が存在しない場合、類似事例を使用して、ユーザからの要求制約を緩和する。緩和した要求制約と目的関数を再び、制約充足器に渡すことで、代替案の出力を可能にする。

このように、CBR Wrapper は、問題補完部による目的関数の自動作成により、制約充足のみを用いた Configuration システムの問題点を解決することができる。また、ユーザの不完全な要求入力からでも、適切な問題解決を行うという従来の CBR システムの特長を有し、問題解決自体は制約充足器で行うため、CBR における解の修正に関する知識の獲得や管理のボトルネックが発生することはない。

次に、実際の Configuration を行う制約充足の各構成要素について、その機能的な役割の説明を行う。

部品 DB：製品の構成要素である部品の定義や、接続関

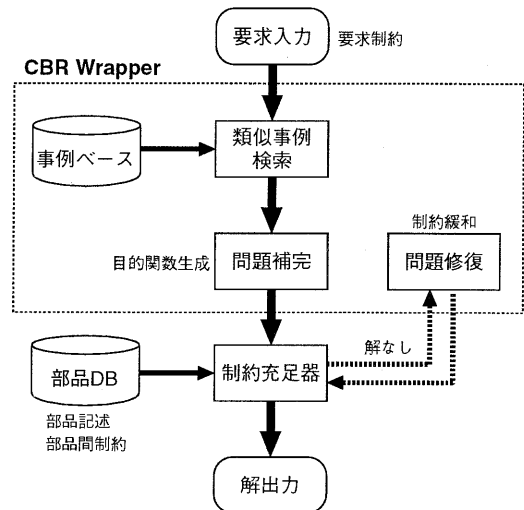


図 2: CBR Wrapper を用いた問題解決

係などの部品間の制約条件が格納されている。

制約充足器：ユーザの要求、部品 DB 中の部品間制約を満たす部品の組み合わせの中から、CBR Wrapper で作成された目的関数を最大（最小）化するものを求める。

以上述べたように、提案した問題解決の枠組みでは、知識の獲得や管理のボトルネックを発生させることなく、多岐にわたるユーザの要求を満たす Configuration を行うことができ、問題領域に依存しないシステムの設計が可能となる。

参考文献

- [1] V. E. Barker and D. E. O'Connor "Expert Systems for Configuration at Digital: XCON and Beyond". *CACM*, Vol.32, No.3, pp. 298-318, 1989.
- [2] J. McDermott "R1: A Rule-Based Configurer of Computer Systems". *Artificial Intelligence*, Vol.19, No.1, pp. 39-88, 1982.
- [3] D. Sabin and R. Weigel "Product Configuration Frameworks - A Survey". *IEEE Intelligent Systems*, Vol.13, No.4, pp. 42-49, 1998.
- [4] M. Stumptner, A. Haselböck, and G. Friedrich "Cocos: A Tool for Constraint-Based, Dynamic Configuration". *Proc. 10th IEEE Conf. AI Applications*, pp. 373-380, 1994.
- [5] I. Vollrath, W. Wilke, and R. Bergmann "Case-Based Reasoning Support for Online Catalog Sales". *IEEE Internet computing*, Vol.2, No.4, pp. 47-54, 1998.