2308 要求された製品挙動を実現可能なインタフェイス構造の設計支援システム

A design method of interface structure to achieve required product behavior

○学 古 賀 毅 (東京大学院)

正 青 山 和 浩 (東京大学院)

Tsuyoshi KOGA and Kazuhiro AOYAMA

Dept. of Environmental and Ocean Engineering, School of Tokyo, The University of Tokyo
7-3-1, Hongo, Bunkyo-ku, Tokyo 113-8656, JAPAN {tsuyoshi-koga, aoyama}@msel.t.u-tokyo.ac.jp

The design of interactions of product components, i.e., an interface structure, is a very difficult design problem, because the whole product behavior is determined by the interface structure. This paper proposes a new algorithm that can be used to derive the interface structure automatically to achieve the required product behaviors. An evaluation method for product behavior by comparison of the required product behavior is proposed.

We call this methodology the design system of Product Behavior by automatic calculation of the Interface Structure (PBIS). Here, we demonstrated the effectiveness of the PBIS algorithm in an implemented prototype system. PBIS enables the designer to determine the interface structure interactively based on a list of proposed interface structures that can realize the required product behavior. The methodology addresses: (1) how the interface structure problem should be formulated; (2) how the algorithm for automatic proposition of the interface structure achieves the required product behaviors; and (3) the evaluation method of the interface structure by comparison between the required product behavior and achieved product behavior.

Key Words: Interface Design, Behavior Design, Petri-Net, Reinforcement Learning, Heuristic Search

1. はじめに

製品の競争力の優位性を確保するためには、顧客要求を満足する製品仕様を決め、その仕様を満たす設計解を求めることが必要不可欠である。しかしながら、製品設計・開発の起点となる顧客要求を正確かつ十分に把握することは一般的に容易ではないため、製品仕様を具体的に求めることは困難とされる。この困難性に関しては要求工学でも指摘されるように、顧客要求を把握するためには、製品の使用目的、使用方法および状況などを顧客へ提案しながら、協調的に設計・開発を進める必要がある。

顧客は、実態としての製品自体を求めるだけではなく、その製品の振舞いによってもたらされる効果を求める。したがって、設計者は、製品が使用される状況を想定し、これを製品の具体的な振舞いに展開することによって必要な製品仕様を確定する。この様な設計の特徴を認識すると、設計とは機能空間の属性空間への写像とする吉川や Suh らのアプローチにおいて¹⁾²⁾、顧客が製品を使用する局面や状況(Scene)を考慮し、製品の振舞い(挙動)を実現化する実体を作りこむ仕組みを考慮する必要がある。

一方において、最近での製品の多機能化、複雑化の傾向を認識することができ、多種多様な製品要素が複雑に組合された製品システムが設計・開発されている。この様な状況下では、製品要素(コンポーネントや部品など)の適切な選択は勿論のこと、それらの製品要素間の相互作用およびインタフェイスを適切に設計することは重要である。しかしながら、多様な要素が存在し、それらの要素の複雑な組合せにおける適切なインタフェイスの設計問題は煩雑であり、何らかの支援方法が望まれているといっても過言ではない。また、上述した顧客要求の抽出において重要な製品の振舞いは製品要素とそのインタフェイスによる組合せに強く依存するため、必要十分な検討支援が要望される。

以上の背景をふまえて本研究では、製品設計・開発における顧客要求を把握し、実体として実現化する過程に着目し、顧客が要求する製品の振舞い(挙動)と製品を構成する製品要素間のインタフェイスの設計問題を対象とする。

2. 製品挙動設計とインタフェイス設計

2.1. 製品挙動設計に関する従来研究 前述のように、製品設計は、製品機能を実現する実体・属性へと写像する行為であると整理される。しかしながら、製品機能の統一的な定義は存在しないとの指摘から³⁾、製品挙動の記述と設計に関する多くの研究が存在する⁴⁾⁵⁾。例えば村上らは、製品挙動に関する特徴を抽出する方法として、コンフィギュレーション空間を用いることにより、製

品挙動を実現する機構原理の検索・表示を計算機支援によって可能としている⁶⁾。しかしながら、村上ら自らが指摘するように、単一の機構原理の検索だけでは不十分であり、複数の機構によって製品システム自体の挙動の設計を支援することが必要である。これを実現するためには、機構原理を有する製品要素の組合せにおけるインタフェイス設計は重要な課題と認識できる。

また、製品機能は挙動と明確に区別困難であり、機能は挙動を使用者が認識したものであるとの立場から、状態を用いて製品挙動を表したFBSモデル(Function-Behavior-State Modeler)が提唱されている⁷⁾。このFBSモデルによって定義された製品機能と挙動を表現するメタモデルを用いれば、様々な製品機能のライブラリ化が実現できる。たとえば吉岡らは、設計オブジェクトの挙動モデル・ライブラリを生成し、ビルディング・ブロックやボキャブラリの提供を行うことで概念設計を支援するKIEF: Knowledge Intensive Engineering Frameworkを提案している⁸⁾。しかし、このFBSの提案においては、製品挙動の設計において必要不可欠と考えられる段階的な詳細化手法は、今後の課題とされている。

2.2. 本研究の目的 本研究では、Fig.1に示すように、製品の使用状況を想定し記述しながら製品挙動を設計し、矛盾が無く、無駄な動作が少ない簡単な機構によって、意図する挙動を実現する製品構造を実現可能な設計手法を議論し、その手法を実装するプロトタイプ・システムを構築することを目的とする。

ところで著者らは、段階的な製品挙動の設計手法に関する研究において、2.1節で示した、複数の機構によって製品システム自体の挙動を設計する問題と、製品挙動設計の段階的な詳細化の問題に関して議論している⁹⁾。そこで本研究では、製品要素間に設計すべきインタフェイス構造を提案するアルゴリズムに関して検討する。このアルゴリズムは、要求する機構を検索する既存技術に加え、検索された機構を組合せて製品全体として意図する製品挙動を達成できるようなインタフェイス構造の提案と決

Fig.1 Design Concept of this Research

定とを支援することを可能とするものである。このアルゴリズムを実現するために、以下の具体的目的を設定する。

- 1) インタフェース設計問題の定式化
- 2) 意図する製品挙動を満足するインタフェース構造の提案アルゴリズム
- 3) 意図した挙動とシステムが提案した挙動との差に着目したインタフェース構造の評価点算出方法

3. インタフェース構造の設計

3.1. 製品挙動の表現モデル 本研究では、インタフェース構造は、製品の構成要素の動作の相互作用を明示化したものであると仮定する。そこで、インタフェース構造の設計問題を定式化するためには、A) 製品情報全体、B) 製品の構成要素の挙動、C) 製品の構成要素間の相互作用に関連する情報表現を定義することが必要となる。本研究では、Petri-Net によるモデル表現を導入し、以下に示すように、これらの情報を表現する。

- State: 製品の構成要素の状態 = Place オブジェクト (Cs; Circular nodes)
- Transition: 製品の構成要素の動作(属性変化) = Transition オブジェクト (Ca; Rectangle nodes)
- Entity: 製品の構成要素(サブ Petri-Net を持つ) = Entity オブジェクト (E; Rectangle area that have sub-Petri Net)
- Attribute: Entity の持つ属性 = Attribute オブジェクト (A; Entity は Attribute を持ち、Attribute は Entity のある時刻における状態により決定される)
- Interface: Entity 間の相互作用 = Interface オブジェクト (I; entities 間の相互作用)

Stapler を表現した例を Fig.2 に示す。ここでは、Stapler は 2 つの構成要素 Entity (Housing と Magazine Unit) から構成されている。State として Housing は状態 {Opened, Closed, Pushed} を有し、同様に、Magazine Unit は状態 {Charged, Empty} を持つ。さらに、それらの状態間の遷移が定義されている (Housing {open, close, push, release}, Magazine Unit {fire, charge}). また、Entity は Attribute を持ち、Attribute の値は State によって定まるものと仮定している。たとえば Housing は、形状という属性を持ち、状態に対応した属性値 (状態 {Opened, Closed, Pushed} に対応する属性値) を取る。製品の状態は、ある時刻を指定すると唯一に定まる。このことを、トークン (State が持つ黒丸) を用いて表現する。本研究では、製品の使用も、挙動を持つ一つのシステムとして捉え、Entity は構成要素だけでなく、使用者も表現する。

以上の Petri-Net 表現において、上に示した A)-C) に関連する情報は、それぞれ以下のように定義できる。

A) 製品情報全体: $G^{product} = \{G^{structure}(E; I, A), G^{behavior}(Cs, Ca; P^a)\}$

B) 製品要素の挙動: $G^{eb} \subseteq G^{behavior}$

C) 製品要素間の相互作用: $A^{if} = G^{behavior} - G^{eb}$

where $G^{product}$: 製品情報全体, $G^{structure}$: 製品の構造情報 (Entities, Attributes, and Interfaces), $G^{behavior}$: 製品の挙動情報 (States, Actions and conditions), G^{eb} : 製品の構成要素の挙動, A^{if} : 製品の構成要素間の相互作用

3.2. インタフェース構造の設計問題の定式化 目標とする製品全体の挙動が与えられたとき、製品の構成要素を決め、それを満足するような相互作用を決定する問題が、インタフェース構造の設計問題である。このインタフェース構造の設計問題は Fig.3 のように図示でき、同様に (式1) (式2) のように表現される。

$$G^{product} = G^{entity} + A^{if} \quad (式1)$$

$$R^{behavior} = \{Sn, En\} \quad (式2)$$

where G^{entity} : 製品構成要素 (Fig. 3 (A)), A^{if} : インタフェース構造 (Fig. 3 (B)), $G^{product}$: 製品情報 (Fig. 3 (C)), Sn: Scene Node (シーン・ノード) 製品がある時刻において置かれている状態, En: Event Node (イベント・ノード) 製品全体を一つのシステムとして捉えた際の製品の動作

この表現を用いれば、インタフェース構造の決定問題は、要求される製品挙動 (Fig.3(D)部) を実現することが可能なインタフェ

ース構造 (Fig.3(B)部) の決定問題となり、インタフェース構造の設計問題は、(式3) のように定式化することができる。

$$\text{インタフェース構造設計} = G^{entity} \text{ と } R^{behavior} \text{ が前提条件、要求として与えられたときの、} A^{if} \text{ の設計問題} \quad (式3)$$

3.3. 構成要素の相互作用の表現 本研究では、製品要素間の相互作用は、設計者がその相互作用を具体的なインタフェースとして実体化して考えていない影響関係と定義する。設計の進捗に伴い、この影響関係は、電線やリンケージ、あるいはアクチュエータなどといった実体概念へと変換されていく (可能性のあるもの) として取り扱う。本研究では、構成要素間の相互作用として、以下の二点の要素を基本として考える。

- a) ある構成要素の動作と別の構成要素の動作の同期: 同期アーク (Fig.4(A) a)). 同期アークは、トランジションとトランジションを結び、両者の同時発火を意味する
- b) ある構成要素の動作が別の構成要素の動作の条件となっているとき: テストアーク (Fig.4(A) b)). テストアークは、プレースとトランジションを結び、トランジションの発火の条件としてプレースにトークンがあることを意味する

これらの相互作用を、論理的に取りうるすべての関係に対して生成したものが、インタフェース情報の候補案になる。インタフェース情報の候補は、以下の手順によって生成される (Fig.4)。

Step_1 インタフェース構成要素のパターンの指定: 設計者は、製品の構成要素の間に存在するインタフェース構成要素のパターンを選択する (Fig. 4 (A))。

Step_2 取りうる全てのインタフェース候補案の生成: 製品の構成要素を決定し、その単体挙動を与える。その後、インタフェ

Fig.2 Interface Structure Design

Fig.3 Generation Process of Interface Candidates

Fig.4 Interface Candidates

イス毎に接続可能性を評価し、全てのインタフェース候補を生成する(Fig. 4 (B)).

Step.3 インタフェース候補案の絞り込み: 生成したインタフェース候補案は、実現不可能な候補も含んでいる。そこで、各インタフェース候補の意味を検討し、これを実現する機構を検索する。得られた結果を用い、さらに、各インタフェース候補に対して実現度やコストなどの評価を行い、実現不可能なインタフェース候補を削除する(Fig. 4 (C)).

以上に示すインタフェースの設計問題は、意図する製品挙動を実現するインタフェース構造の決定問題である。この問題は、前節に示した様に生成される製品における製品要素間の相互作用情報の全体から、インタフェース構造としてとりうる相互作用の選択問題として置き換えることができる。

3.4. インタフェース構造の評価 インタフェース構造の解空間の中には、意図する製品挙動を満たす解や、意図する製品挙動に近い解などが複数存在すると考えられる。意図する製品挙動との一致度が低い設計案であっても、実現性やコストを考慮すれば採用すべき案が含まれている可能性がある。

本研究では、設計者が、解空間の中から、設計案を適切に選択するために、1)意図する製品挙動との一致度、2)機構としての実現性・コストに関する評価指標を導入する。

評価指標1:意図する挙動との一致度(Performance): 意図した挙動が実現できるかを示す評価指標である。インタフェース構造をもつ製品情報から解析された製品挙動と、意図した製品挙動とを比較し、一致度を定量化することによって求める。

評価指標2:実現度, コスト(Feasibility, Cost): インタフェース構造を実際に製品上で実装するために要するコストを基準に評価する指標である。インタフェース構造の Feasibility $f^k_{feasibility}$ は、以下の式によって評価する。

$$f^k_{feasibility} = \prod_{i=1}^n f^i \times \delta^k_i \quad (式4)$$

where: $f^k_{feasibility}$: k番目のインタフェース構造の実現度

f^i : i番目のインタフェース候補の実現度

δ^k_i : インタフェース構造がi番目のインタフェース候補を持つとき1、持たないとき0

3.5. インタフェース構造のヒューリスティック・サーチ インタフェース構造の集合の大きさは、インタフェース候補が増大すると、指数的に大きくなり、インタフェース構造の空間のすべての要素を評価することは不可能となる(NP 困難問題)。そこで、上記に提案された評価手法を用いて、ランダムに生成した案に対し、得点を計算することにより、インタフェースの重要度として学習していく解探索アルゴリズムを提案する。このアルゴリズムは、強化学習に基づき、以下の4ステップにより構成される。

Step.1 インタフェース構造をランダムに生成: 各インタフェース候補に対し、重要度(0-1; w)を設定する(初期値は一律に一定値とする)。この重要度をインタフェース候補の選択率として扱い、インタフェース構造をランダムに生成する(Fig.5(B)).

Step.2 製品挙動を生成: インタフェース構造が生成されると、製品情報が確定し(Fig.5(C)), 可達木によって製品挙動を計算することができる(Fig.5(D)(E)).

Step.3 意図する製品挙動との比較: 得られた製品挙動が、意図する製品挙動とどれだけ一致するかを評価する。得られた可達木のグラフと、意図する製品挙動の状態遷移構造を比較し、Performanceを算出する。

Step.4 フィードバック・ラーニング: 得られた評価結果を用いて、インタフェースの重要度を変更する。Performanceが高い設計案に含まれるインタフェースの重要度を増加、Performanceが低い設計案に含まれる重要度を下げ、学習させる。

以上に示す解の生成、評価、学習(Step.1 から Step.4 まで)によって、インタフェースの重要度が変化するプロセスを Fig.6 に示す。一律に設定した初期の重要度 w_0 (Fig.6(A))が、学習の繰り返しにより次第に変化し(w_k) (Fig.6(B)), 最終的に必要なインタフェース候補と必要でない候補へと分かれる過程を確認できる。

Fig.5 Algorithm of Heuristic Search

3.6. 解のランキング 最終的に設計者に対し、計算によって得られた価値が高い案のランキングを提示する。ランキングは、以上の計算によって求められた、各インタフェース構造の解空間における Performance と Feasibility により生成する。設計解の価値は、評価式: $Value = Performance/Cost$ に従い算出する。インタフェース構造の解空間の全要素を、横軸 Performance、縦軸 Feasibility としてプロットした結果を、Fig.7に示す。曲線は、Value が一定となる曲線を表している。また、要素の色は、設計解の属するクラスを意味する。ここで、Value の高い順に並び替えることにより、設計解のランキングを提示し、推薦される設計解として設計者に提案する。

4. プロトタイプ・システムと実行例

以上に述べた手法を実装したプロトタイプ・システムを構築し、有用性を検証する。Fig.8 に示す様なトイ・パークプレーンの設計を例に、インタフェース構造が提案された結果を Fig.9 に示す。

4.1. パーク・プレーンに対する要求の設計 設計者は、上昇・下降・右旋回・左旋回が可能なパーク・プレーンを設計したいと考えたとする。そこで、この要求挙動に対し、存在するシーンとイベント、および製品構成の情報を、Fig.8(A)に示すように定義する。この製品状態に必要な動作(推力発生や方向転換など)から、構成要素(ダクトドファンユニット、レバー、ラダー、翼、胴体等)を設計する(静的な要素以外を Fig.8(B)に記述)。また、必要な構成要素の接続構造や形状材質などの属性を定義する(Fig.8(C))。この製品の構成情報および属性情報の設計に、トップダウン指向設計支援システムを用いた¹⁰⁾。

Fig.6 Learning Curve

Fig.7 Ranking of Solutions

4.2. 提案されたインタフェース構造 定義した問題設定に対して、探索によって示されたインタフェース構造は Fig.9(1-a)である。これは、インタフェースのタイプ(Fig.9(1-e))から生成したインタフェース構造の空間(Fig.9(1-d))に対し、シーンのオプション性の情報を追加し探索した結果得られたものである。意図する製品挙動(Fig.9(1-b))に対して、提案されたインタフェース構造の挙動を Fig.9(1-c)に示す。提案された製品は、以下のように動作する。

「レバーLを上げるとファンの出力が発生し、機体は上昇する。レバーLを下げることでファンの出力がダウンし、機体は下降する。レバーRを左右に倒すことで、左右のファンの出力差が発生し、このことにより機体は左右に旋回する。」

4.3. 提案された異なるタイプのインタフェース構造 同様に、4.1.において定義した問題設定に対して、探索によって示された設計解を Fig.9(2)に示す。この製品案は、同じ製品挙動を達成することが可能な設計解であるが、インタフェース構造が異なっている。Fig.9(2)にて提案された製品は、以下のように動作する。

「レバーLを上げるとファンの出力が発生し、機体は上昇する。レバーLを下げることでファンの出力がダウンし、機体は下降する。レバーRの左右の動作がラダーの左右の動作に同期し、このことにより機体は左右に旋回する。」

Fig.9(1)に示す案は、最終的にアンプ、サーボとリンケージにより実現されるが、Fig.9(2)ではデュアルアンプにより実現できる。Fig.9(1)の案は、市販のコンポーネントを組み合わせることで実現できるというメリットがある。一方、Fig.9(2)の案では、最終的な製品において可動機構が少なくなるため、信頼性や耐久性という観点から良い案である。設計者は、製品を特徴づけるこれらのインタフェース構造を、提案の中から選択することで決定していくことができる。

5. 結論と考察

本研究では、挙動を考慮した製品設計におけるインタフェース構造の決定問題の困難性から、インタフェース構造を探索によって提案する手法を提案した。まずインタフェース構造の設計問題を定式化し、インタフェース構造の評価軸を創出し、解空間の中から探索的にインタフェース構造を求めることで、意図する製品挙動を実現可能なインタフェース構造を提案可能なアルゴリズムを示した。このアルゴリズムを用いることによって、従来の機能や仕様を属性へと写像していく製品設計手法に対し、希望する製品挙動を実現可能な製品情報を生成するというアプローチによる新しい製品設計手法が可能になるものと考えられる。

本研究では、以下の問題が今後の課題として認識している。

- 決定論的なアプローチにより、設計者が製品挙動やインタフェース情報の消去、追加などといった操作時に、必要なアクションや、間違った決定に対する指示を与えることで、設計を支援する手法の構築

(1) Proposed Interface Structure type1

(2) Proposed Interface Structure type2

Fig.9 Two Types of Proposed Interface Structure

Fig.8 Example of Prototype System (A Design of Toy R/C Plane)

- 階層的なアプローチを用いた、インタフェース決定問題の部分分解への対応
- シーンやイベント・製品構成要素やその動作のモデルを用いた、オブジェクトおよび対応関係のボキャブラリの構築

参考文献

- [1]Yoshikawa, H. ,General design theory and a CAD system, Man-machine Communications in CAD/CAM, North-Holland, 1981.
- [2]Suh, N.P. , The Principles of Design, Oxford University Press Inc,USA,1990
- [3]Umeda, Y., Tomiyama, T. and Yoshikawa, H., Proposal of Function-Behavior-State Modeling to Support Functional Design (in Japanese), The Japan Society for Precision Engineering, Vol. 63, No. 6, pp. 795-800,1997
- [4]Umeda, Y., Takeda, H., Tomiyama, T. and Yoshikawa, H., Function, Behavior, And Structure, Applications of Artificial Intelligence in Engineering V. Ed. Gero, Proceedings of the 23 Fifth International Conference, Boston, Massachusetts,1990.
- [5]Brown, D.C., Functional, Behavioral and Structural Features, Proceedings of the ASME IDETC, DETC2003/DTM-48684,2003.
- [6]Murakami, T. and Nakajima, N., Mechanism Concept Retrieval Using Concept Space (in Japanese), Transactions of the Japan Society of Mechanical Engineers, Vol.60, No.580, pp.4400-4408,1994.
- [7]Shimomura, Y., Tanigawa, S., Takeda, H., Umeda, Y. and Tomiyama, T., Functional Evaluation Based on Function Concept, Proceedings of ASME DETC, 96-DETC:DTM-1412,1996
- [8]Yoshioka, M., Shamoto, Y. and Tomiyama, T., An application of the Knowledge Intensive Engineering Framework to Building Foundation Design, Knowledge Intensive Computer Aided Design, Susan Finger, pp. 197-212, Kluwer Academic Publishers, Dordrecht,1999.
- [9]Koga T. and Aoyama K., Product Behavior and Topological Structure Design System by Step-by-Step Decomposition, Proceedings of the ASME IDETC, DETC2004/DTM-57513,2004.
- [10]Aoyama K. and Koga T., Development of an Artificial Satellite Design Environment by Top-Down oriented Design Supporting System, Proceeding of the 24th ISTS,2004