

以案例式推理
建立產品功能結構之探討
Case-Based Reasoning for the Generation of Function
Structures +

計劃編號: NSC 89-2212-E-002-068
執行期限: 88年8月1日至89年7月31日
主持人: 尤春風 you@w3.me.ntu.edu.tw
執行單位: 臺灣大學機械系

一、摘要

本文以 Pahl & Beitz 的設計方法與案例式推理的架構為基礎，配合結構式比對的原則，進行電腦輔助產品功能結構求解的研究。產品功能求解屬於設計流程中概念設計的階段，主要是根據設計問題的規範合成可以滿足設計目標的功能。本研究主要內容有兩項，第一是建立一套可以表達設計案例的模型，將過去的設計經驗有系統地儲存在資料庫當中，以供後續使用；第二是建立一套系統化的流程，讓電腦根據工程師輸入的設計資訊，在過去案例資料庫中搜尋相似案例、移轉功能及組合功能，以輔助使用者建立產品功能結構，提高設計效率。文中以咖啡機設計問題為例，驗證本研究的可行性。

關鍵詞：功能結構合成、案例式推理、結構式比對、設計案例表示法、搜尋設計案例。

Based on Case-Based Reasoning(CBR) and Structure Mapping, a systematic approach

for generating the function structures of mechanical products is proposed. Generation of function structures is the main process of conceptual design stage, which determines the principle solution of design. In this stage, designers transform design specifications to more concrete concept represented by an assembly of functions which could achieve the purpose of design requirements.

CBR, as a design process, aims at assisting engineers by recalling and reusing the previous design experience. It includes the subtasks of representing, indexing, retrieving and adapting the design cases. Under the structure of CBR, a system is proposed that receives design specifications of problems as input, retrieves similar design case by using Structure Mapping, and returns the generated function structures as output. Some examples have been tested to verify the feasibility of this approach.

Keywords : conceptual design, function structure, case-based reasoning, structure mapping,

二、計劃緣由及目的

在市場競爭激烈的環境下，產品的生命週期快速縮短，消費者對產品的要求也越來越高，生產者若未能意識到產品設計的重要性，將為潮流所淘汰。因此，如何有效率地進行產品開發的工作，成為生產者的一大課題。

設計流程中包含頻繁的資訊流動，在這些資訊裏面，對日後產品開發最有幫助的，應屬設計過程中產生的經驗，這些經驗都是企業寶貴的智慧財產，不但可以增進企業的競爭力，也是新進人員的活教材，此外，日後若遇有類似狀況，可以直接運用這些經驗，減少嘗試錯誤的成本。這些設計經驗的表現型式各不相同，若沒有經過適當的規劃運用，往往會流失；而且沒有經過標準化的資訊，常會導致各部門間的溝通費時費力，造成不必要的錯誤。

目前一般設計資訊的保存方式，主要是將產品的幾何資料、材料清單以及設計變更等資訊儲存起來，極少有關於設計者思考過程的記錄。因此，當其他的設計者或是新進人員想要參考過去設計案例時，往往僅能「知其然而不知其所以然」。

由於在產品開發計畫進行的過程中，設計階段的結果決定產品的基本特性，對於產品的成敗有決定性的影響，因此，若能有效地運用電腦與資訊科技，使設計的流程及設計資訊電腦化，利用電腦強大的記憶功能，減少尋

覓和整合過去資料的時間，設計師的精力將可專注於產品本身的開發，達到提昇設計品質及縮短設計時間的目的。

機械概念設計有兩大特色，第一，它沒有唯一解，這也和設計問題本身不易定義完整有關；第二，它的解決過程是遞迴的，也就是說最初找到的解並不一定是最終的結果，而是需要一直尋求更好的解決方案。因此，設計問題所牽涉的層面既廣且深。此外，人類有一項非常寶貴的能力，是將過去的經驗和目前情境建立類比關係，透過既有概念的拓展、修改以及擴充，形成創新性的想法。但是人們往往會受限於本身對事物的既有印象，而難以採用新的觀點來解決問題，而且將針對某一問題得到的解決方案視為個案，造成日後利用過去解決方案處理其他相似問題的困難。在進行創新性設計時，應該採取一種開放的態度，將觸角向外延伸到其他領域，以期充分利用現有的資源形成新的觀念。

因此，要以較接近人們處理問題的方式解決機械概念設計的問題，通常無法利用類似專家系統的工具來解決，原因在於專家系統是將過去的設計經驗以法則的形式儲存，這可能造成兩個瓶頸：第一，知識的擷取不易；第二，當知識不斷擴充時，規則之間的矛盾不易管理。而案例式推理是以電腦模擬人類在進行設計時較自然的思考模式，也就是以回想過去設計經驗的方式從事新設計問題的解決，而不是每次都

從零開始。因此本研究認為案例式推理在機械概念設計的運用是值得探討的課題。

本研究主要探討如何利用電腦來輔助概念設計的進行，讓使用者可以藉由輸入設計規範的方式，得到滿足設計需求的功能結構輸出。整體流程如圖1所示。

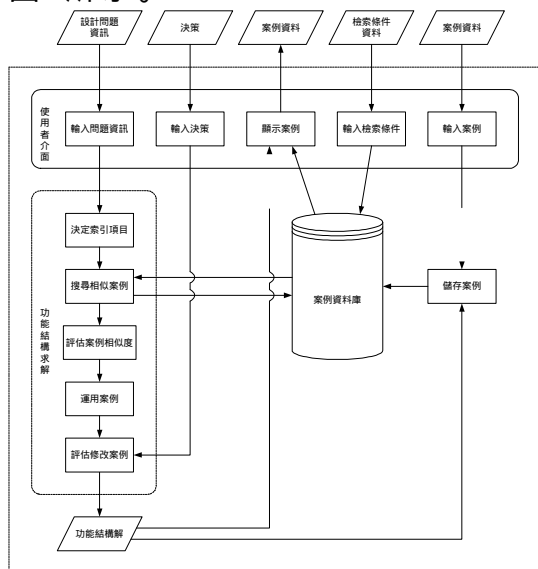


圖1 系統架構圖

本研究的設計流程是以 Pahl & Beitz 的設計方法為基礎；推理功能結構是以案例式推理為架構；將設計案例資訊分為規範、行為、功能以及實體四方面表示；以結構式比對進行案例的搜尋與運用。研究內容可歸納為以下項目：

1. 規劃設計案例表示法。案例中包含：規範、功能、物件間相互影響關係、實體結構等資訊。
2. 建構案例式推理的機制。其中包

括：決定索引項目、搜尋相似案例、評估案例相似度、運用案例等。

3. 制定語彙表示法並建立字典。
4. 建立結構式比對的機制。
5. 建立電腦合成功能結構的法則。
6. 利用物件導向式資料庫 ObjectStore 儲存設計案例。

三、研究方法

3.1 搜尋相似案例

在使用待解決問題規範項目搜尋相似案例時，是以這些規範項目的述詞來和案例規範項目的述詞進行比對，若案例規範項目的述詞具有和待解決規範項目相同或相似的述詞，則將此案例規範項目納入候選清單中。而各個待解決問題規範項目分別有一個候選清單，其中紀錄對應到這個規範項目的案例規範項目。

判斷兩個述詞是否相同或相似的方法，是藉由查詢兩個述詞各自所屬案例之字典，找出它們分別對應的字詞物件，再針對這兩個字詞進行比較，得到兩個字詞之間的相似程度。在這個階段比較兩個字詞是否相似的方法，是先比對兩個字詞的內容，也就是原本的述詞；若內容相同，則代表這兩個述詞相同；若內容不同，則進一步將兩個字詞對應的同

義詞加入比對，若仍然沒有找到相同者，則代表這兩個述詞不相似。

3.2 評估案例相似度

由「搜尋相似案例」步驟得到各個待解決規範項目的候選清單，接著要進一步比對清單中案例規範項目和對應的待解決規範項目間語句的相似程度。

本研究是以結合結構式比對以及語意比對的方式來進行語句相似度的評比。

3.2.1 語句相似度比對法則

本研究中規範項目語句比對是以待解決規範項目的語句結構為基準，依序由上層向下層、由左而右取得元素，進行和案例規範項目語句中對應位置元素相互比較。

3.2.2 字詞相似度比對法則

本研究進行字詞相似度比對時，是將兩個字詞的字詞本身、同義字詞、反義字詞皆納入考慮。比對的流程如圖3-6所示。若有兩個字詞分別為字詞₁與字詞₂，首先比較兩個字詞本身是否相同，如果不同，則接著進行「字詞₁本身與字詞₂同義詞」、「字詞₁同義詞與字詞₂本身」、「字詞₁同義詞與字詞₂同義詞」以及「字詞₁反義詞與字詞₂反義詞」等四個階段的比對。在每一個階段中，都是遞迴地進行字詞相似度比對的流程。

3.2.3 語句比對結果表示法

待解決問題規範項目與案例規範項目之間語句相似度的表示法，是建立一個和待解決問題規範項目樹狀圖架構相同的新樹狀圖，並且在此一新樹狀圖的節點上儲存待解決問題規範項目各元素與案例規範項目元素比對的結果。儲存的資料包括：待解決問題規範項目元素內容、對應的案例規範項目元素內容以及比對得分。

3.2.4 語句相似度計算法

在語句中各元素的字詞相似度皆已得知之後，便可進行語句相似度的計算。依此計算方式，兩個語句的相似度是由語句比對結果樹狀圖中底層的各元素開始，逐步向樹狀圖上方累加各元素的字詞相似度值，累加至最上方的節點所得之值，即為這兩個語句的相似度。

3.3 運用設計案例

尋得資料庫中相關案例並得到它們和各待解決問題規範的相似程度之後，進入運用設計案例的步驟建立新的功能結構，也就是使用過去的設計經驗來解決目前的問題。

運用設計案例分為兩個步驟，首先是根據各個待解決規範項目和案例規範項目的比對結果，分別以對應的案例功能結構建立新的功能結構解，稱之為移轉功能；接著是利用這

些功能結構解組成產品的整體功能結構，稱之為組合功能。

3.3.1 移轉功能

移轉功能的主要目的是以新的字詞代換案例功能中的字詞，以形成新的功能。逐一取出待解決規範項目，並根據其對應候選案例規範項目的相似度，由相似度高到相似度低依序取用案例規範項目，代換對應的案例功能結構字詞，以建立新功能結構。接著將新建立的功能結構加入可用功能結構的候選清單中，以供下一個步驟組合功能運用。

3.3.2 組合功能

得到各個規範項目的對應功能結構候選清單之後，進入組合功能的步驟。也就是組織可分別滿足規範項目的功能結構，形成一個產品的功能結構，以滿足規範的需求。

四、結論與成果

根據本研究的法則，電腦可以根據使用者輸入的設計規範，透過案例式推理的程序找出相似的設計案例，並且修改尋得的案例，以生成可以滿足設計需求的功能結構。要點整理如下。

1. 以案例式推理的架構為基礎，使得電腦能夠透過搜尋相似案例、運用設計案例的過程，建立產品功能結構。

2. 使用「規範—功能—行為—實體」模型表現設計案例，有系統地保存設計資訊以及設計資訊之間的關係。並且利用物件導向式資料庫儲存設計資訊，提升存取資訊的效率。

3. 使用結構式比對的方法，作為搜尋相似設計案例時判斷語句相似度的標準，以提高評估案例相似度的彈性。

4. 利用案例中關於功能組合的資訊以及功能輸出入的比對組合功能結構。

參考文獻

1. A. Terry Purcell and John S. Gero, "Design and Other Types of Fixation", *Design Studies*, Vol. 17, No. 4, pp. 363-383, 1996.
3. Ashwin Ram, Linda Wills, Eric Domeshek, Nancy Nersessian, Janet Kolodner, "Understanding the Creative Mind: A Review of Margaret Boden's Creative Mind", *Artificial Intelligence*, Vol. 79, pp. 111-128, 1995.
4. B. Raphael and B. Kumar, "Indexing and Retrieval of Cases in a Case-Based Design System", *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Vol. 10, No. 1, pp. 47-63, 1996.

5. G. Pahl and W. Beitz, *Engineering Design: A Systematic Approach, 2nd ed.* edited by K. Wallace, Springer-Verlag, London, 1996.
6. Gerard Jounghyun Kim, "Case-Based Design for Assembly", *Computer-Aided Design*, Vol. 29, No. 7, pp. 497-506, 1997.
7. Ian Watson and Srinath Perera, "Case-Based Design: A Review and Analysis of Building Design Applications", *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Vol. 11, No. 1, pp. 59-87, 1997.
8. Jerzy Surma and Bertrand Braunschweig, "Case-Based Retrieval in Process Engineering: Supporting Design by Reusing Flowsheets", *Engineering Applications of Artificial Intelligence*, Vol. 9, No. 4, pp. 385-391, 1996.
9. Mary Lou Maher and Andrés Gómez de Silva Garza, "Case-Based Reasoning in Design", *IEEE Expert*, Vol. 12, No.2, pp. 34-41, 1997.
10. Peter Jackson, *Introduction to Expert Systems*, Addison - Wesley, 1990.