INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 01 GLASGOW, AUGUST 21-23, 2001

TOWARDS HYBRID METHODS FOR SYNTHESIS

A Chakrabarti

Keywords: human creativity, innovation methods, principles of invention, concept generation.

1 Introduction

Design synthesis research is often seen as having produced a set of competing approaches, falling largely under two seemingly conflicting philosophies. One is the design reuse philosophy which prescribes retrieval and reuse of past designs as much as possible, and suggests adaptation of past designs to the current purpose if complete reuse is not possible. The other is the compositional synthesis philosophy that prescribes development of a design by composing a set of building blocks.

Under each of these philosophies, the approaches available are either designer-led methods, or automated tools. For instance, the function-structures approach [1] is primarily meant to be a designer-led approach for compositional synthesis, while the functional synthesis approach [2] is one of its automated approaches.

Within the design reuse philosophy, similar divisions can be seen within its two major sub-areas: retrieval of designs, and change of a design to suit the current purpose. For instance, design catalogues by [3] is a designer-led approach to retrieval, while retrieval using analogy [4] is one of its automated approaches. Similarly, the support proposed based on a function model [5] can be taken as a designer-led approach to change and repair, while the evolutionary approach to architectural layout design [6] is one of its automated approaches.

The aim of this paper is to address the issue of polarity and specialism in the area of design synthesis research, and identify the scope of blending the current, apparently conflicting and competing approaches into more powerful synthesis schemes that should help make major advances in the capability of the area as a whole.

2 A review of the main approaches to synthesis in design

The main approaches to synthesis – compositional synthesis and design reuse (which includes retrieval and change) – are discussed below, with the aim of identifying their relative significance, and the main issues involved in terms of them providing creative designs that satisfy a given function.

2.1 Compositional synthesis

Compositional synthesis approaches start off with the overall function to be fulfilled by a design, and combine building blocks to fulfil these functions [2, 7, 8]. Usually compositional synthesis is seen as the process of generating compositions of building blocks by combining them from a pre-defined set. However, identification of building blocks itself is part of the broader, creative task of synthesis [9]. In order to be able to synthesise well, one must be able to identify appropriate building blocks; central to this are the issues of what building blocks to use, and how composite or basic they should be [10, 11].

Compositional synthesis has been hailed as having the potential of generating innovative designs, but at the cost of having high potential risks associated with them (as with innovative designs in general) and having a resource-intensive development process. Case based design approaches [12], which are based on retrieval and adaptation (change), are seen to be more resource efficient [13] and less risky [14] than compositional synthesis.

2.2 Design Reuse

Within a design reuse approach [4, 15, 16, 17], a design is retrieved and changed if necessary. The change may be due to (i) the need to generate alternative designs [4, 15], (ii) to modify an existing design to fulfil the current purpose [16], or (iii) to develop an optimal design [17, 18].

Retrieval

Retrieval is the first step in the design reuse approach. Often the central issue is how to identify the most appropriate cases for the current purpose. When purpose can be expressed in a quantitative way, the main task has been to identify whether a retrieved design is capable of providing the function [19, 20]; change in parametric values of the components of the retrieved designs are performed to test this. In others, where product functions are qualitative, analogy is often used as a means of retrieval, which can be at several levels such as functional, behavioural and structural [4].

In general, retrieval of complete designs is likely to lead to the least innovative designs, but has less cost associated with the development process. The more different the source and target domains are, the more innovative and risky the designs, and the more resource-intensive their development process are likely to be.

Change

One way of changing a design is to identify a set of rules the application of which creates different but valid designs. These can then be subjected to test to see if they satisfy the given function, which can be quantitative or qualitative, based on which further modifications are made. Change processes can be used to generate alternative designs [14]. They can also be used as a means of repair. Knowledge-lean processes such as simulated annealing can be used to make the changes [16], but so far they have proved more successful where the functions can be quantified and components used are similar in characteristics to each other, e.g., structural optimisation. In cases where functions are hard to quantify, knowledge-intensive processes (e.g., qualitative model based reasoning [15]) have been relatively more successful.

In general, for designs with functions that are hard to quantify, change processes for repair are likely to produce designs that are more innovative and risky than those directly retrieved, but less so than compositionally synthesised designs, especially if compositional synthesis is done using less composite elements as building blocks. The resource involved in development is also usually in-between, although not always, as seen from a comparison between a compositional synthesis process and a retrieval-based modification process [21].

3 Analysis

Possible combinations of the three processes underlying the main sub-approaches - compositional synthesis, retrieval and change - are systematically enumerated in this section. The change process is seen to take place in terms of two alternative ways: retrieval of a design part of which needs to be replaced, or creation of a design using compositional synthesis in order to use that as a replacement. Each combination enumerated can be viewed as a possible synthesis scheme. These combinations are then put together into a minimal graph that is a union of the combinations. Since this minimal graph has the individual combinations as its subsets, this graph can be treated as a generic synthesis framework from which all these synthesis approaches can be derived.

3.1 Combinations starting with compositional synthesis

Compositional synthesis (CS) starts with a function and no structure, and ends with one or many alternative structures – combinations of building blocks – that fulfil the function.

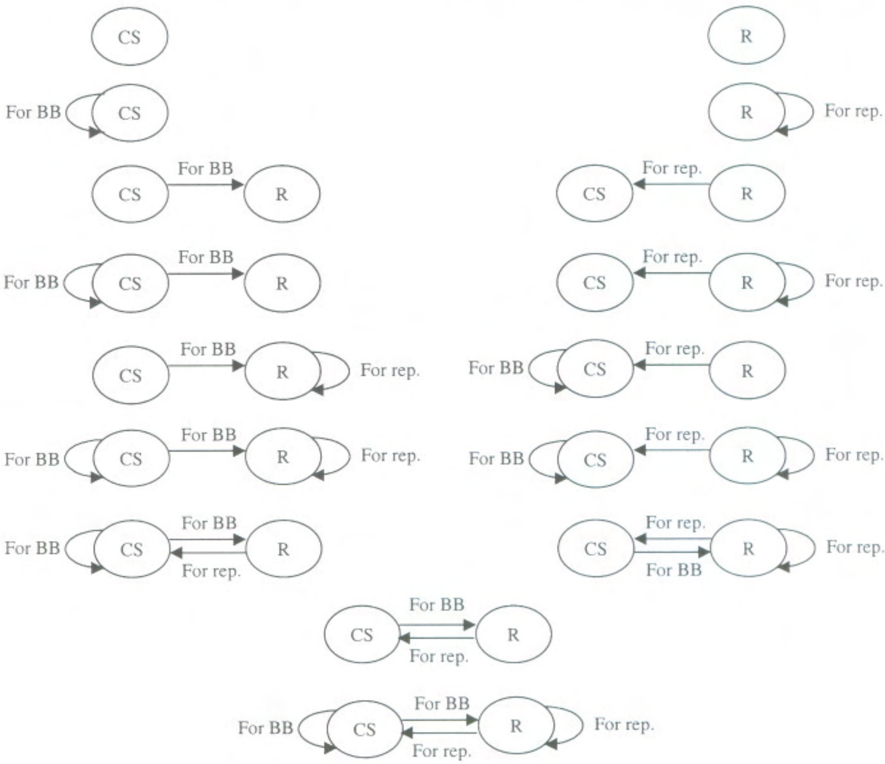

Figure 1. Hybrid synthesis possibilities

The first combination possibility is where a structure that can fulfil a given function can be developed using, or is constrained to remain within the set of building blocks with which synthesis begins. This is pure compositional synthesis, shown as the top left option in Fig. 1.

The second is the possibility where the existing building blocks are not sufficient to fulfil a certain sub-function, at least within the constraints imposed (e.g., the number of building blocks to be used in a solution). Some of these building blocks, therefore, are combined into more composite building blocks to be used in the synthesis (2nd down from top left, Fig. 1).

The third possibility applies where the existing building blocks are not sufficient to fulfil a certain sub-function. An existing design is retrieved (R) as a new building block, to be used in the composition process. This is a combination of synthesis and retrieval (3rd down from top left, Fig. 1).

The fourth possibility applies when the building block retrieved does not fulfil the function entirely and a change, to fulfil the remaining function for instance, is necessary. This is done by developing a building block using compositional synthesis (4th down from top left, Fig. 1).

The fifth is similar to the fourth, except that in this case, instead of creating new building blocks using compositional synthesis, the retrieved solution is changed using further retrieval for replacement of parts of it that do not work (5th down from top left, Fig. 1).

In the sixth case, new building blocks are created and used in compositional synthesis using both compositional synthesis and retrieval, and yet the function is not fully satisfied. So, the retrieved solution is further amended using further retrieval (6th down from top left, Fig. 1).

The next (next down) is similar to the sixth, except that instead of using further retrieval as the process of further amending the solution retrieved, retrieval uses compositional synthesis for this purpose.

The eighth case (last but one in the figure) starts off with compositional synthesis as the process to change the solution originally retrieved, and retrieval as the process for identifying further building blocks necessary in the compositional synthesis process.

The ninth (last down in the figure) case is similar to the eighth, but also includes processes for generating further building blocks using compositional synthesis, and processes for further change using further retrieval.

3.2 Combinations starting with retrieval

Retrieval (R) starts with a function and no structure, and ends with one or many alternative structures – retrieved from a case-base – that fulfil the function.

The first possibility is that a structure that can fulfil a given function is obtainable directly from the case base, shown as the top right option in Fig. 1.

The second is the possibility in which the solution originally retrieved is not sufficient to fulfil the function and is changed by replacing parts of it using further retrieval (2nd down from top right, Fig. 1).

The third is similar to the second, except that the change process for replacement is done using compositional synthesis rather than further retrieval. This is a combination of synthesis and retrieval (3rd down from top right, Fig. 1)

The fourth possibility involves change processes using compositional synthesis as well as further retrieval. This would be the case where the one done first does not quite fulfil the function and therefore further changes are necessary (4^{th} down from top right, Fig. 1).

The fifth is similar to the fourth, except that in this case all the changes are done using compositional synthesis, which uses compositional synthesis to develop further building blocks necessary (5^{th} down from top right, Fig. 1).

In the sixth case, further retrieval is tried first as the change process for amending the solution originally retrieved. However, this is not sufficient to fulfil the function in this case, and further changes are made using compositional synthesis that uses further compositional synthesis to develop building blocks (6^{th} down from top right, Fig. 1).

The next (next down) is similar to the sixth, except that instead of using further compositional synthesis as the process of developing further building blocks, compositional synthesis uses further retrieval for this purpose.

The eighth case (last but one in the figure) starts off with search for building blocks using retrieval, and its amendment using compositional synthesis.

The ninth (last down in the figure) case is similar to the eighth, but also includes processes for generating further building blocks using compositional synthesis, and processes for further change using further retrieval. Note that these last two cases are common in both the processes that start primarily as either compositional synthesis or retrieval.

3.3 Overall Framework

The cases described in Section 3.2 illustrate two things. The first is, since a complete design as dug out by a retrieval process can also be used as a building block, retrieval can be used as a sub-process of a compositional synthesis approach. Now, when a design retrieved by a retrieval process does not quite provide the intended function, it needs to be changed, and thus change approaches can be used as sub-processes of a compositional synthesis approach.

The second is, when a design is retrieved and needs changing, a possible change process may be a compositional synthesis approach, which may use a design reuse approach as one of its sub-processes. In other words, the two approaches can bootstrap, leading to more symbiotic, capable and possibly more complex schemes, in which the two original, apparently distinct and conflicting approaches can be sub-processes of each other.

This bootstrapping is what is indicated in the bottom four cases in Fig. 1 in all of which arrows connect compositional synthesis and retrieval in both directions. This symmetry allows the process to start primarily as a compositional synthesis or retrieval. This, for instance, can be done by re-interpreting the seventh case down from top left in Fig. 1 (originally interpreted primarily as compositional synthesis) as primarily a retrieval process. In this new interpretation, the process starts as retrieval which uses compositional synthesis as its sub-process for replacement of parts of the solution retrieved, which in turn uses further compositional synthesis as well as retrieval as sub-processes to develop further building blocks. Similarly, the seventh case down from right in the figure (originally interpreted primarily as retrieval) can be re-interpreted primarily as a compositional synthesis process, which uses retrieval to generate its building blocks, which in turn uses further retrieval as well as further compositional synthesis to amend the building blocks.

It should be noted that the case lowest down in Fig. 1 has all the other cases subsumed in it, and is the minimal graph that includes them all. This graph is taken here as the generic synthesis framework (see Fig. 2).

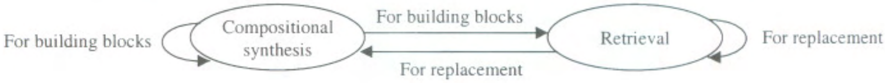

Figure 2. Generic synthesis framework

If the two ovals are considered in isolation, they show compositional synthesis and retrieval as alternative direct approaches to synthesis. However, if direct retrieval does not work, the retrieved design may need further retrieval for replacement (shown by arrow pointing back to retrieval, on right) or compositional synthesis in order to create the replacement of some of its parts (shown by arrow at the bottom). Similarly, if direct compositional synthesis does not work, it may require developing further building blocks (BB), generated by further synthesis (arrow on right), or further retrieval of designs to be used as building blocks (arrow on top). The process can go on several times, and the arrows together can be seen to represent iterative or recursive cycles.

4 Examples from existing synthesis approaches

The first two cases – of direct compositional synthesis (top left in Fig. 1) and retrieval (top right in Fig. 1) have many examples in design research literature [2, 4, 7, 8, 19, 20]. Retrieval and adaptation by retrieval (2nd down from top right, Fig. 1) can be seen for instance in [6] where design prototypes are retrieved from case memory, on which GA operators are applied to generate new designs. An example of limited case adaptation using both compositional synthesis and further retrieval (fourth down from right in Fig. 1) can be seen in [22]. However, there seems to be no example of the other combinations enumerated, which points to the gap in current synthesis methods and the scope for developing them along these lines.

5 Conclusions and further work

The main conclusions are:

- There are two apparently distinct and contrasting philosophies in design synthesis research: design reuse (retrieval and change) and compositional synthesis.
- There is a variety of approaches under each, both design-led and automated.
- It is possible to describe change in terms of synthesis and retrieval.
- The approaches can be explained as subsets of a generic synthesis framework consisting of synthesis and retrieval operations, with iterative or recursive cycles between the two.
- None of the existing approaches presently allow such seamless combination; however evidence suggests that designers already do some of these and it would help if approaches to help designers synthesise also had this flexibility.

This leads to a host of research issues: how can each such hybrid approach be supported? Where is each one applicable? How far should iterative cycles go on until it is understood that there can be no solution possible within a base of knowledge? The two potential challenges are knowledge representation and development of rules that are necessary to get to the relevant knowledge in an efficient way. These are future directions for this work.

References

[1] Pahl, G., and Beitz, W. Engineering Design: A Systematic Approach, Springer, London, 1996.

[2] Chakrabarti, A. Designing by Functions, PhD Thesis, Department of Engineering, University of Cambridge, 1991.

[3] Roth, K. Systematik der Maschinen und ihrer Mechanischen Elementaren Funktionen, Feinwerktechnik, 74, 453-460, 1970.

[4] Qian, L. and Gero, J. S. Function-behaviour-structure paths and their role in analogy based design. AIEDAM Special Issue on Representing Functionality in Design, A. Chakrabarti & L. Blessing (Eds.), 10(4), 289-312, 1996.

[5] Ullman, D.G. A New View on Function Modeling, Proc. International Conference on Engineering Design, 1, 21-28, The Hague, 1993.

[6] Gomez de Silva Garza, A., and Maher, M. L. A Process Model for Evolutionary Design Case Adaptation, J.S. Gero (Ed.) Artificial Intelligence in Design'00, 393-412, Kluwer Academic Publishers, 2000.

[7] Ulrich, K.T., Computation and Pre-Parametric Design, PhD Thesis, AI Laboratory, Massachusetts Institute of Technology, Cambridge, 1988.

[8] Bracewell, R. H., and Sharpe. J. E. E. Functional Representations Used in Computer Support for Qualitative Scheme Generation – Schemebuilder, AIEDAM Special Issue on Representing Functionality in Design, A. Chakrabarti & L. Blessing (Eds.), 10(4), 333-346, 1996.

[9] Smithers, T. Personal communication, 1999.

[10] Chakrabarti, A. and Bligh T. P. Exploring Basicness of Building Blocks for Supporting Innovative Design of Mechanical Systems using the Building Blocks Metaphor, Technical Report No. CUED/C-EDC/TR61-January 1998, University of Cambridge, 1998.

[11] Liu, Y.C. A Methodology for the Generation of Concepts in Mechanical Design, PhD Thesis, Department of Engineering, University of Cambridge, 2000.

[12] Kolodner, J. Case Based Reasoning, Morgan Kaufmann, San Mateo, 1993.

[13] Pu, P., and Purvis, L. Formalizing Case Adaptation in a Case Based Design System, J.S. Gero and F. Sudweeks (Eds.) Artificial Intelligence in Design'94, 77-91, Kluwer Academic Publishers, 1994.

[14] Hammond, K. J. CHEF: A Model of Case Based Planning, Proc. AAAI-86, Philadelphia, PA, 267-271, 1986.

[15] Finger, S., and Rinderle, J. R. A Transformational Approach for Mechanical Design Using a Bind Graph Grammar, EDRC Report No. 24-23-90, Carnegie Mellon University, USA, 1990.

[16] Sycara, K., and Navinchandra, D. Integrating Case Based Reasoning and Qualitative Reasoning in Engineering Design, J.S. Gero (Ed.) Artificial Intelligence in Design'90, Computational Mechanics Publications, Springer Verlag, 231-250, 1990.

[17] Shea, K. Essays of Discrete Structures: Purposeful Design of Grammatical Structures by Direct Stochastic Search, PhD Thesis, Canegie Mellon University, USA, 1997.

[18] Liu, J-S., Parks, G. T., and Clarkson, P. J. Metamorphic Development: A New Topology Optimisation Method for Truss Structures, The 40th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Paper AIAA-99-1387, St Louis, MO, USA, April 12-15, 1999.

[19] Medland, A. J., Mullineux, G., Twyman, B. R., and Rentoul, A. H. A Constraint Based Approach to the Design and Optimisation of Mechanisms Systems, IUTAM Symposium on Optimisation of Mechanical Systems, D. Bestle and W. Schiehlen (Eds.), Kluwer Academic, Netherlands, 1996.

[20] McGarva, J. R. Rapid Search and Selection of Path Generating Mechanisms from a Library, Mechanisms and Machine Theory, 29(2), 223-235, 1994.

[21] Chakrabarti, A., and Bligh, T.P. A Scheme for Functional Reasoning is Conceptual Design, Technical Report CUED/C-EDC/TR60, Cambridge University Engineering Department, January 1998.

[22] Hua, K., and Faltings, B. Exploring Case Based Building Design - CADRE, AI EDAM, 7(2), 135-143, 1993.

Corresponding author's name: Dr Amaresh Chakrabarti

Engineering Design Centre, Cambridge University Engineering Department, Trumpington Street, Cambridge CB2 1PZ, UK, Tel: +44 1223 766961, Fax: +44 1223 332662, E-mail: ac123@eng.cam.ac.uk, URL: http://www-edc.eng.cam.ac.uk/people/ac123.html