

# **Axiomatic Design And TRIZ: Compatibilities and Contradictions Part II**

Darrell Mann  
Systematic Innovation  
5A Yeo-Bank Business Park  
Kenn Road, Clevedon  
BS21 6UW, UK  
Phone: +44 (1275) 337500  
Fax: +44 (1275) 337509  
E-mail: Darrell.Mann@systematic-innovation.com

## **Introduction**

This article follows last month's brief look at Nam Suh's Axiomatic Design (AD) ideas and their possible relationship with TRIZ (1). The aim in this article will be to take a more detailed look at some of the AD tools and techniques in an attempt to demonstrate some of the benefits they may offer to TRIZ-based problem definition and problem solution processes.

## **Design Hierarchy**

In the design of complex systems, last month's article suggested three areas in which AD might benefit a TRIZ-based design approach:-

- 1) The need for a process of iteration between Functional Requirements (FRs) of a design and physical design attributes (Design Parameters – DPs) - i.e. the Design Parameters must be allowed to influence the form and content of the Functional Requirements.
- 2) AD places careful emphasis on the importance of recognising the hierarchical nature of design, and particularly to ensuring that the process of iteration between Function Requirements in the Functional Domain and selection of Design Parameters in the

Physical Domain (shown as the green arrows in Figure 1) is carried out in a systematic manner. As may be seen in the figure, this systemisation occurs through an essentially top-down approach; definition of System Level FRs permits derivation and iteration of System Level DPs (green arrow 'A') and then – most importantly – definition of the System Level DPs is necessary before FRs at the next level down in the hierarchy (blue arrow 'B') can be defined; and so on right through each level of the hierarchy. In effect, AD suggests that finalisation of top level FRs can only really be achieved after each layer of the problem hierarchy has been given due consideration and iterated accordingly.

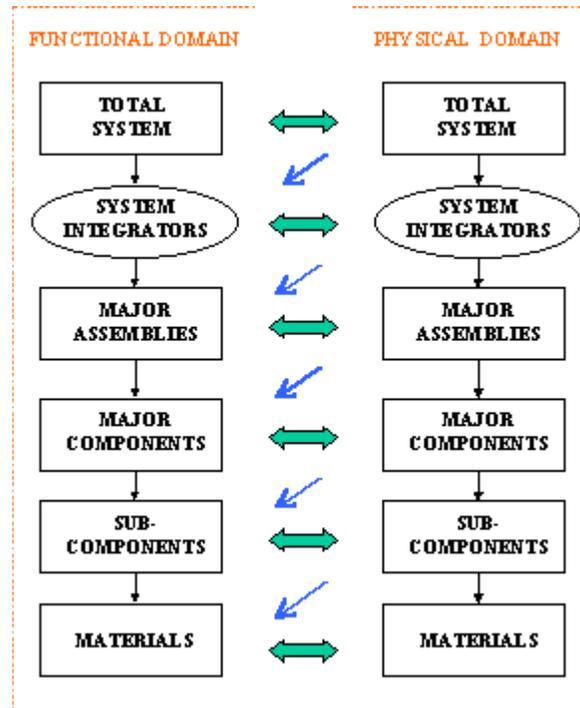


Figure 1: Hierarchical Nature of Functional Domain – Physical Domain Mapping

- 3) As well as recognising the relationship between Functional Domain and Physical Domain, Suh further extends the AD model to include what he describes the Process Domain (Reference 2, Section 4.10). In other words, AD demands that manufacturability issues are given appropriate consideration **during** the process of iterating to achieve the most appropriate form of the design Functional Requirements (i.e. we might add a third 'Process Domain' column to Figure 1 above – as shown now in Figure 2).

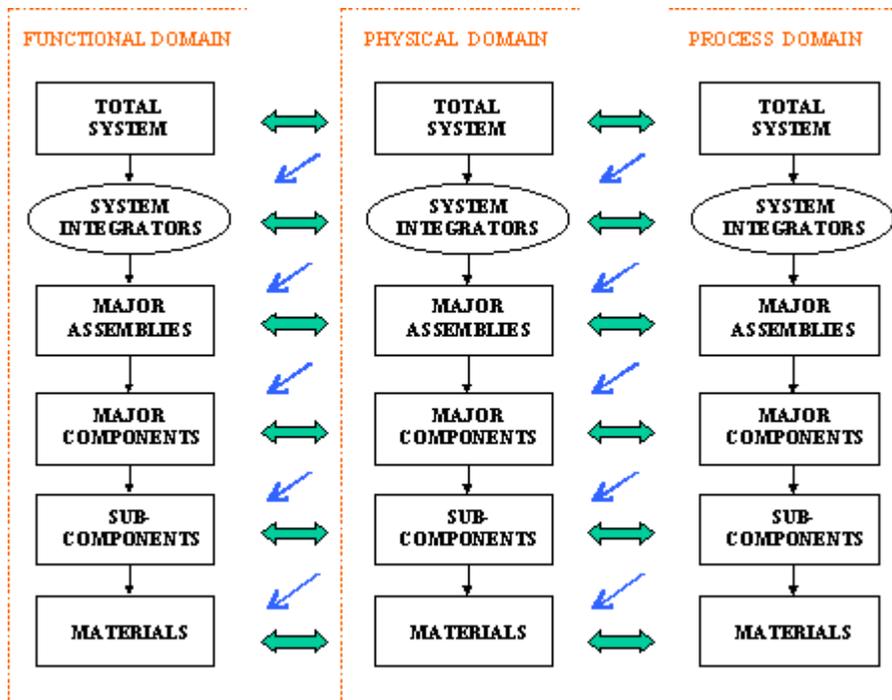


Figure 2: Hierarchical Nature of Functional Domain – Physical Domain Mapping

The hierarchical nature of design is one of the most important AD concepts. Suh uses the example of a metalworking lathe as a way of illustrating the hierarchy construction process in action. In this case, two hierarchies are constructed – the first a functional hierarchy (Figure 3), and the second, a physical hierarchy (Figure 4).

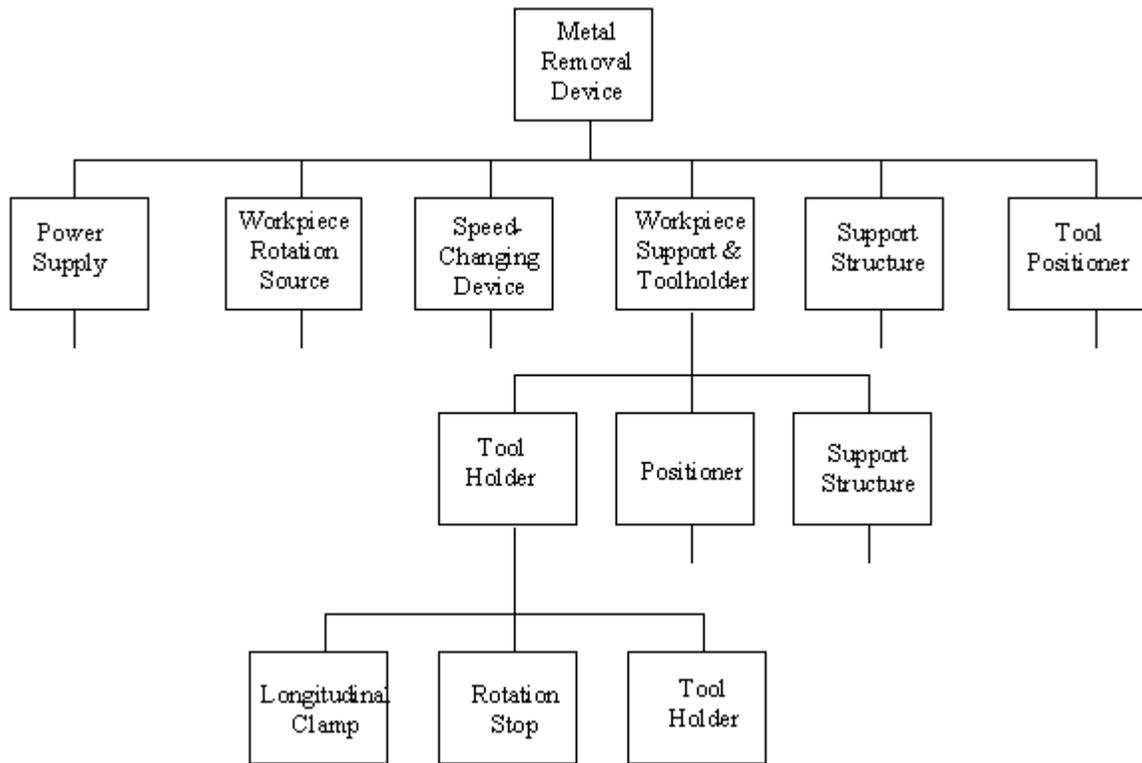


Figure 3: Lathe Functional Hierarchy (after Reference 2)

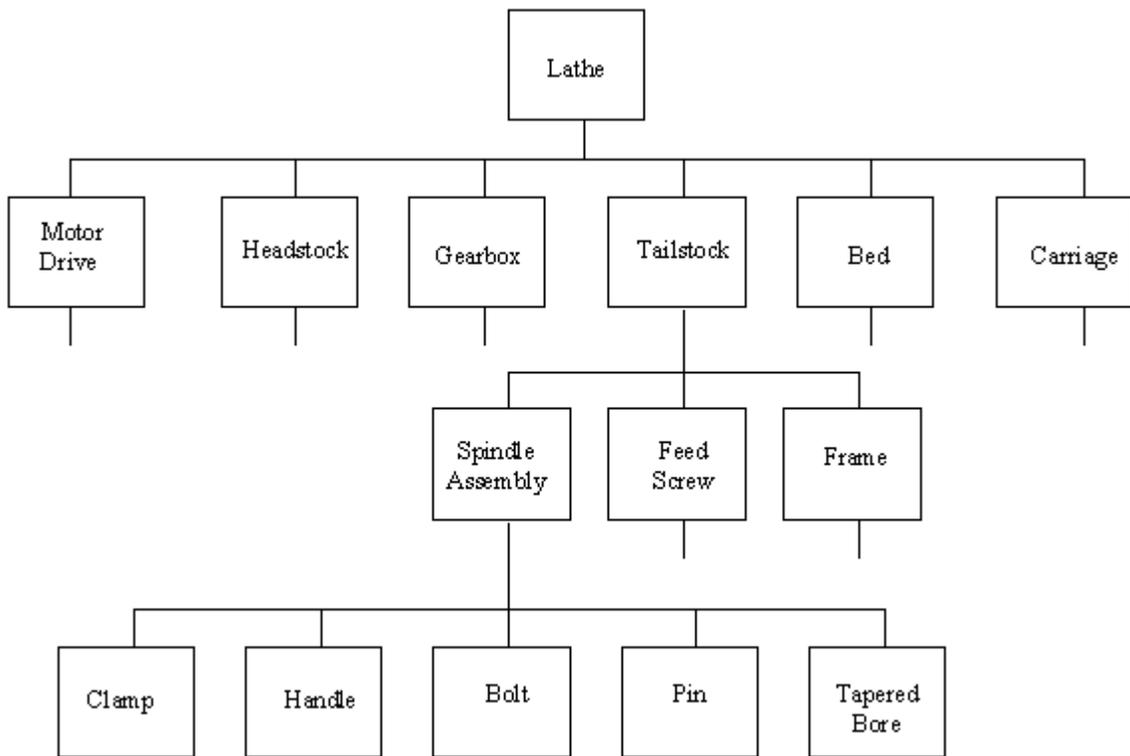


Figure 4: Lathe System Physical Hierarchy (after Reference 2)

One of the most important ideas in AD is that the functional and physical hierarchies are inter-dependent, and it is not possible, therefore, to construct the complete functional hierarchy without reading across to the physical domain at each corresponding level. Or, as Suh states in *The Principles of Design*, 'without having decided to use a *tailstock*, we could not have stated the three FRs: tool-holder, positioner, and support structure'.

We should also note the existence of a third (also interdependent) hierarchy associated with the process domain. For example, our ideas on the 'tool-holder' function and its corresponding physical manifestation in the 'spindle assembly' will clearly be affected by our ability or otherwise to be able to manufacture an appropriate 'tapered bore'.

While clear corollaries between the hierarchical ideas contained in AD and the sub-system/super-system concepts in TRIZ may be seen to exist, the **domain shifting** and **process domain** concepts found in AD appear to offer significant benefit to TRIZ methods.

## Constraints

No discussion of AD would be complete without consideration of the constraints which commonly attach themselves to any design exercise.

Suh defined two types of constraint; *input constraints* – constraints involving design specification (usually bounds on size, weight, cost, etc), and *system constraints* – involving constraints imposed by the system in which the design solution must function (e.g. geometric shape, environmental considerations, prime mover capacity, etc).

The distinction between a 'constraint' and a 'functional requirement' during problem specification can often be unclear. According to Suh, a constraint does not have to be independent of other constraints, its precise numeric value is often unimportant providing it is inside a limit, and it rarely has a tolerance associated with it.

'Cost' is a very commonly applied constraint. Especially in an overall context – for example, 'total product factory cost must not exceed \$x'. An important point here is that, unless we obtain a clear picture of the full design hierarchy in the physical (and usually process) domain(s), it is difficult or impossible to gauge whether the constraint has been met or not.

Constraints like this act on the entire hierarchy. According to Suh, as we pass up and down a hierarchy, the form of DPs and constraints at a given level may change (Reference 2, pp39-40), but for our current purposes, we shall merely acknowledge the facts that, a) overall problem constraints will exist and, b) that

they will have a potential effect of some kind at all levels of the functional, physical and process hierarchies.

### ***Aircraft Wing Design I***

Aircraft wings represent some of the most sophisticated systems to be found anywhere in the field of engineering endeavour, so, with due apologies to those who might be involved in the design and development of real wings, the analysis here is inevitably going to be somewhat simplified in order that we might concentrate on the AD/TRIZ learning points and not the design intricacies.

In the same way that a hierarchy of functional and physical requirements was constructed by Suh for the lathe example, we might construct a similar picture for an aircraft wing.

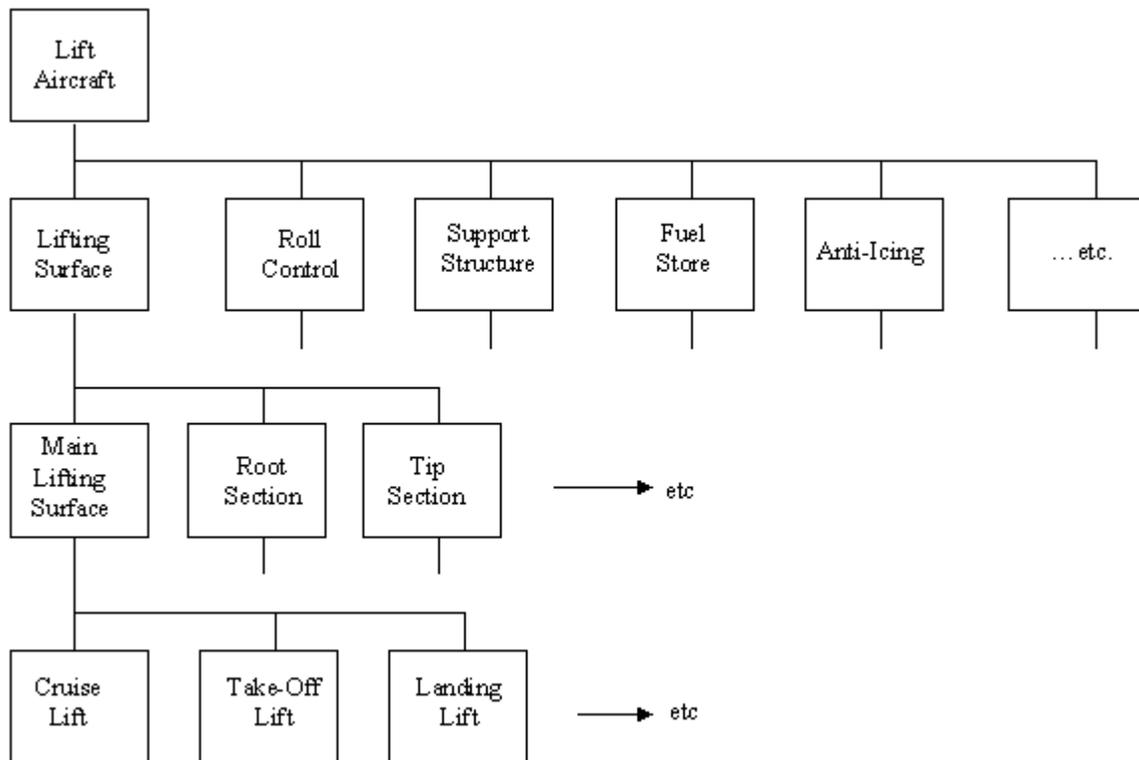


Figure 5: Simplified Aircraft Wing Functional Hierarchy

Figure 5 illustrates a possible (simplified) wing functional hierarchy. As with the previous lathe example, it is possible to picture how each function can map across to a given physical system or component.

In order to set the example in context, let us assume we set a Functional Requirement to increase wing lift performance by x%. It is quite likely that we will be expected to bound this FR with a number of constraints. Such constraints might include:-

- zero allowable increase in manufacture cost
- zero allowable decrease in wing life
- zero allowable increase in maintenance

Given the FR, we might expect to adopt a solution strategy which examines individual functions or functions which sit at the same level in the hierarchy. This is certainly consistent with a TRIZ-based solution approach. For example, we might look at how the wing tip can contribute to overall increased lift performance by constructing a technical contradiction: Thing we are trying to improve - Object Generated Harmful Factors (loss of lift due to passage of air over the side of the wing), thing which gets worse – Length (e.g. wing span increases), TRIZ recommends – ‘Another Dimension’ (e.g. incorporation of winglets).

Similarly, if we start from the ‘main lifting surface’ or ‘cruise lift’ functions, and we don’t consider any possible interactions with other parts of the hierarchy, we might derive a solution like that shown in Figure 6, where passive boundary layer control features are applied locally on appropriate surfaces of the wing in order to create air-flow conditions conducive to better lift performance.

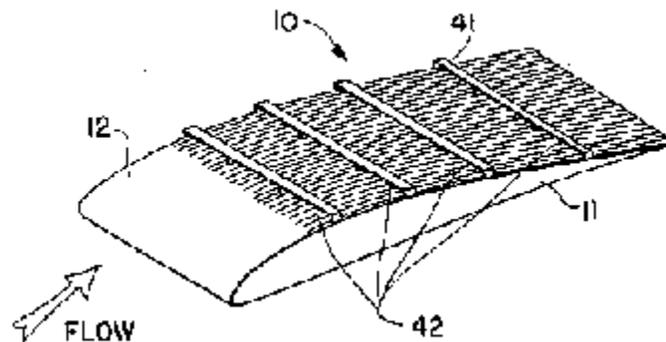


Figure 6: Example Passive BL Control on an Aircraft Wing (US Patent 4,706,910 (expired))

This idea of improving performance by concentrating on a particular area of the design hierarchy is a common, practical and usually valid method.

### ***Aircraft Wing Design II***

An alternative approach, is to examine the bigger picture. This can clearly be best done only if the complete hierarchy picture is constructed. The previous wing design example looked at passive means of achieving better wing lift performance. In this second example we will look at active methods of lift enhancement, and, in particular, the use of boundary layer suction to improve flow characteristics over the top surface of the wing. There are a number of forms active suction can take. Figure 7 illustrates an invention based on incorporation of myriad small holes through which unwelcome slow moving air in the wing boundary layer region is removed.

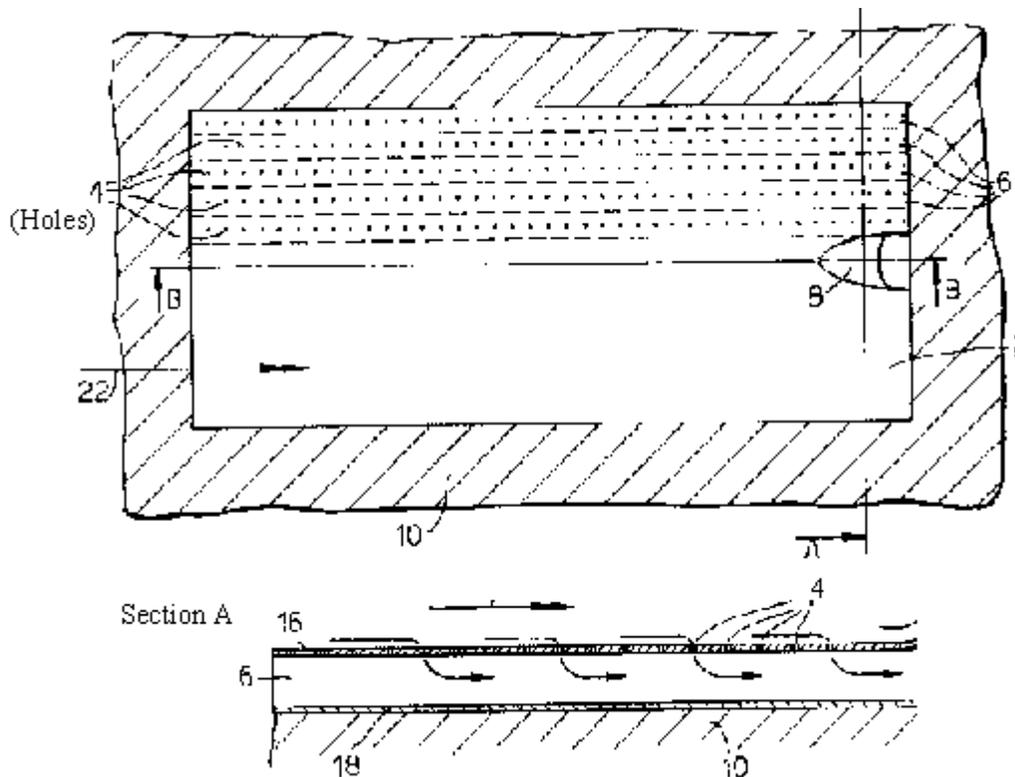


Figure 7: Example Active BL Control using Suction (US Patent 5,848,768)

Active suction is unlikely to emerge as a design possibility as a result of an analysis conducted at a single area of the hierarchy (or if it does, it very soon has to work its way through the whole hierarchy due to the emergence of, for example, the need for means of providing the suction, and means of integrating them with other elements of the wing design).

This 'complete picture' approach is one possible advantage which emerges through adoption of an AD approach. There are a number of points, however, which point to more significant benefits of the use of AD-based design analysis techniques:-

- a. Only through due consideration of the process domain will it be possible to assess the manufacturability implications of the suction holes. A requirement for high hole density is clearly going to have technical and cost implications. It is quite easy to imagine that a requirement for hole densities of the order of hundreds per square inch could quite easily increase manufacture cost to the point of non-viability. Unless, of course, significant innovations in the hole making process are made.
- b. Cost constraints are usually applied generally across the whole hierarchy structure rather than apportioned to individual parts in order to emphasise the bigger picture and the possibilities for penalties in one part of a design to be compensated for by benefits in others (e.g. incorporation of holes will increase manufacture cost, but

- the more efficient wing will see a reduction in fuel burn and therefore give rise to a reduction in overall operating cost).
- c. The connection between the idea of boundary layer suction using small holes and the need to provide an anti-icing capability might suggest a number of interaction problems which will have implications on the success or otherwise of the design. By plotting the ‘complete’ hierarchy, we will hopefully identify where such interactions occur and thus be able to do something to remedy any likely problems.
  - d. Generic constraints like ‘zero maintenance increase’ are again useful because they prompt consideration of the broadest possible range of potential problem areas – e.g. insect contamination can be a problem on many aerodynamic surfaces. It is clearly a maintenance issue. It is also clearly going to be a major issue if the aerodynamic surface in question is full of tiny holes. Maybe that is just another TRIZ contradiction to be solved, or maybe it is further justification for a design approach in which the ‘complete’ picture is drawn before any hardware development is instigated?

## **Discussion**

1. AD characterises ‘good’ design as that which achieves a one-to-one relationship between a function requirement and the physical design feature which achieves it – i.e. every component should perform a useful function. This, of course, takes no account of the ‘trimming’ ideas found in TRIZ and, particularly, the concept of maintaining functionality by trimming components and having their function performed by another part of the system.
2. Page 39 of Suh’s AD book discusses the point about the importance of minimising the number of FRs at each level of the design hierarchy. The book says, *‘Some designers are proud that their design products can perform more functions than were originally specified. In this case, they have over-designed the product. Consequently it is more complex, more costly, and less reliable than is necessary. The designer who creates such a solution should go back and search for a simpler solution’* (Reference 2). In light of the capabilities TRIZ offers, it is perhaps appropriate to modify this somewhat harsh assessment to reflect the knowledge that **function** per se need not correlate to increased cost or complexity or whatever, provided designers are inventive enough to achieve the required functionality without increasing the use of resources. In other words, designers ought to focus on **maximising value** rather than minimising functional requirements.
3. The aircraft wing function hierarchy figure (Figure 5) introduces an element of time dependency when it splits the lifting surface requirement into distinctly different aspects of the flight regime. In some cases, it is convenient to display such time-based requirements on one hierarchy picture. In others, it is often necessary to construct a complete hierarchy for each possible time scenario.

In either instance, it is interesting to note the strong correlation with Altshuller's '9 Windows' viewing perspective (Reference 3).

4. In the second wing design example, we saw the use of many holes to achieve a certain function. In light of the subsequently expressed concerns over the practical viability of manufacturing such a system economically, it is perhaps interesting to relate the problem to the TRIZ 'space segmentation' Evolution Trend to see where a more practical solution might emerge in the future.

5. In both wing design examples, it is important to note that at some stage in the design process, it is prudent to check the existence or otherwise of possible interaction effects with other parts of the hierarchy. For example, the winglet design may well have implications on the design of the overall wing anti-icing system.

### **Conclusions**

1. Axiomatic Design perspectives on functional, physical and process hierarchies in the design of a system offer a useful alternative to conventional TRIZ perspectives.
2. It is important to recognise and utilise the inter-dependencies which exist between both hierarchical layers **and** the different hierarchical regimes.
3. Definition and use of global design constraints provides a useful method for achieving a good overall design perspective.
4. Explicit use of the Process Domain and Domain shifting techniques of AD will offer significant enhancements to a TRIZ-based design approach.

### **References**

1. Mann, D.L. 'Axiomatic Design And TRIZ: Compatibilities and Contradictions', TRIZ Journal, June 1999.
2. Suh, N.P., 'The Principles of Design', (Oxford University Press, 1990).
3. Altshuller, G., 'Creativity As An Exact Science', (New York, Gordon And Breach, 1988).