APPENDIX C

A Conceptualisation of Model Synthesis

This appendix contains the author's own conceptualisation of model synthesis to fill the void in the modelling literature. There are many possibilities for synthesis but no criteria or strategies to this end. The decision support literature, e.g. Dolk (1993), discusses model integration and model management systems with respect to developing new modelling languages and environments but not ways to synthesize existing techniques and models. The host of conceptual issues suggests that model synthesis is not a trivial undertaking by any means.

This appendix is organised as follows. Section C.1 clarifies the concept of model synthesis, e.g. the difference between a technique and a model. Section C.2 shows the potential for synthesis between various techniques by illustrating the similarities between them. Section C.3 examines model structuring issues of technique choice, ordering, and linkage. Section C.4 proposes a distinction between weak and strong forms of synthesis as a taxonomy for levels of integration. Section C.5 offers different strategies for synthesis.

C.1 Definitions

A ***technique*** is *a proven, standardised method which solves a specific class of problems*. Techniques are defined by their ***functionality*** such as optimisation, simulation, and other types of analysis. Many operational research methods and financial techniques have been incorporated into commercially available software tools, e.g. Excel add-ins.

A ***model*** is *an application of a technique to solve a more specific problem*. A technique becomes a model when input and output variables are specified and data is applied. A technique is the engine without the data, whereas a model employs a technique with data. A model evolves into a new technique when its algorithm or driving engine becomes generic, i.e. applicable to a class of problems but not confined to a context-specific problem. A model is a smaller and simpler interpretation of a bigger and more complex reality. [An ***approach*** refers to a method, such as the use of a technique or model, or even a combination of techniques and models to solve a particular problem.]

A ***composite model*** (Kydes and Rubin, 1981) is any model which is made up of separate components, each independently developed and not originally designed to be compatible, and built by integrating (linking) two or more separate, dissimilar types of methodologies. ***Model synthesis*** concerns *the use of more than one technique or model to build a composite model.* A related term, ***integrated modelling*** (Geoffrion, 1987) refers to the coordinated unification of two or more distinct models, enabling results and insights that cannot be achieved by separate models, hence a need to preserve the conceptual integrity of sub-models or components. Model synthesis refers to synthesis of methods or features of approaches, not just at the output level of, say, ***combining forecasts***.

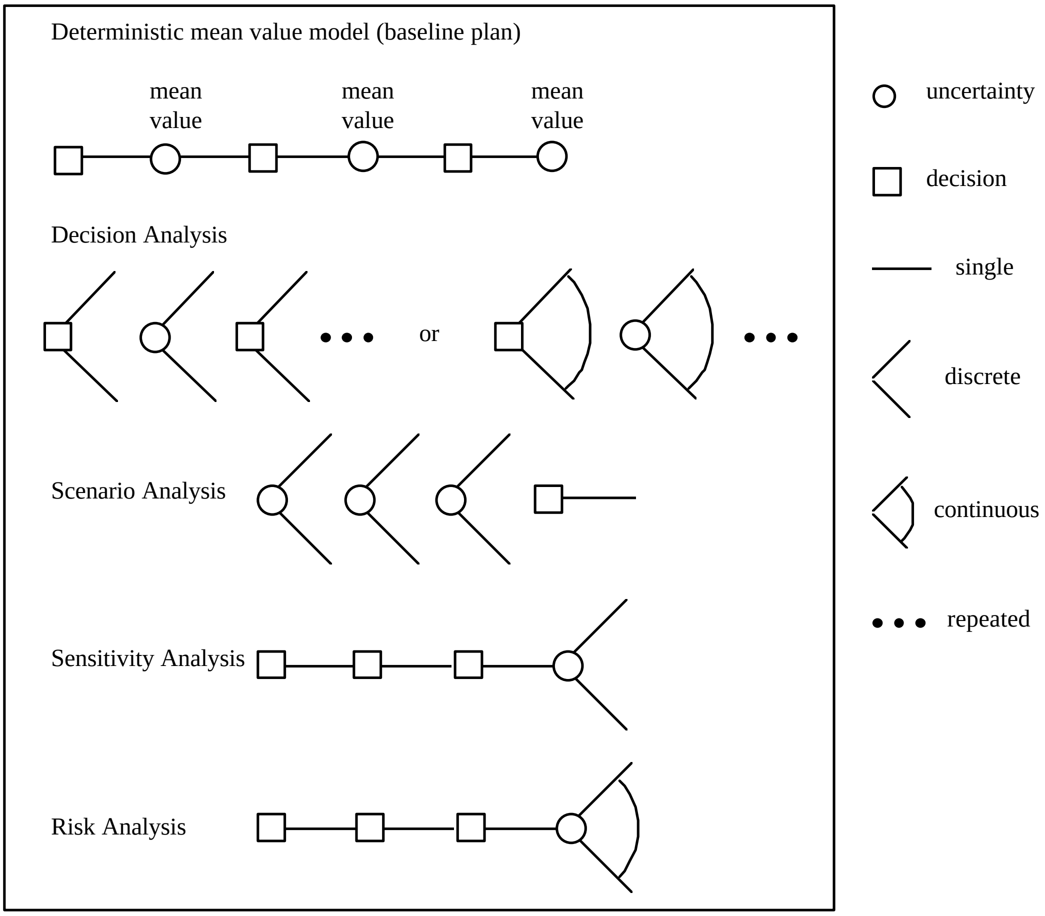
C.2 Synergies Between Techniques

C.2.1 Decision and Uncertainty Nodes

If represented as a sequence of decisions and uncertainties in figure C.1, the techniques of ***decision analysis***, ***risk analysis***, ***scenario analysis***, ***and sensitivity analysis*** have some similarities, suggesting possibilities for synthesis. The deterministic base case model, where single mean values represent uncertainties, is limited in scope when compared to other models of decision analysis, scenario analysis, sensitivity analysis, and risk analysis. Scenario analysis, sensitivity analysis, and risk analysis are single staged while decision analysis is multi-staged.

Another difference lies in the order of deterministic and uncertain nodes. Decision analysis and risk analysis are able to consider continuous probability distributions while scenario analysis and sensitivity analysis are limited to the discrete. The decision maker's attitude to risk is lacking in all except decision analysis.

Figure C.1 Similarities of Techniques



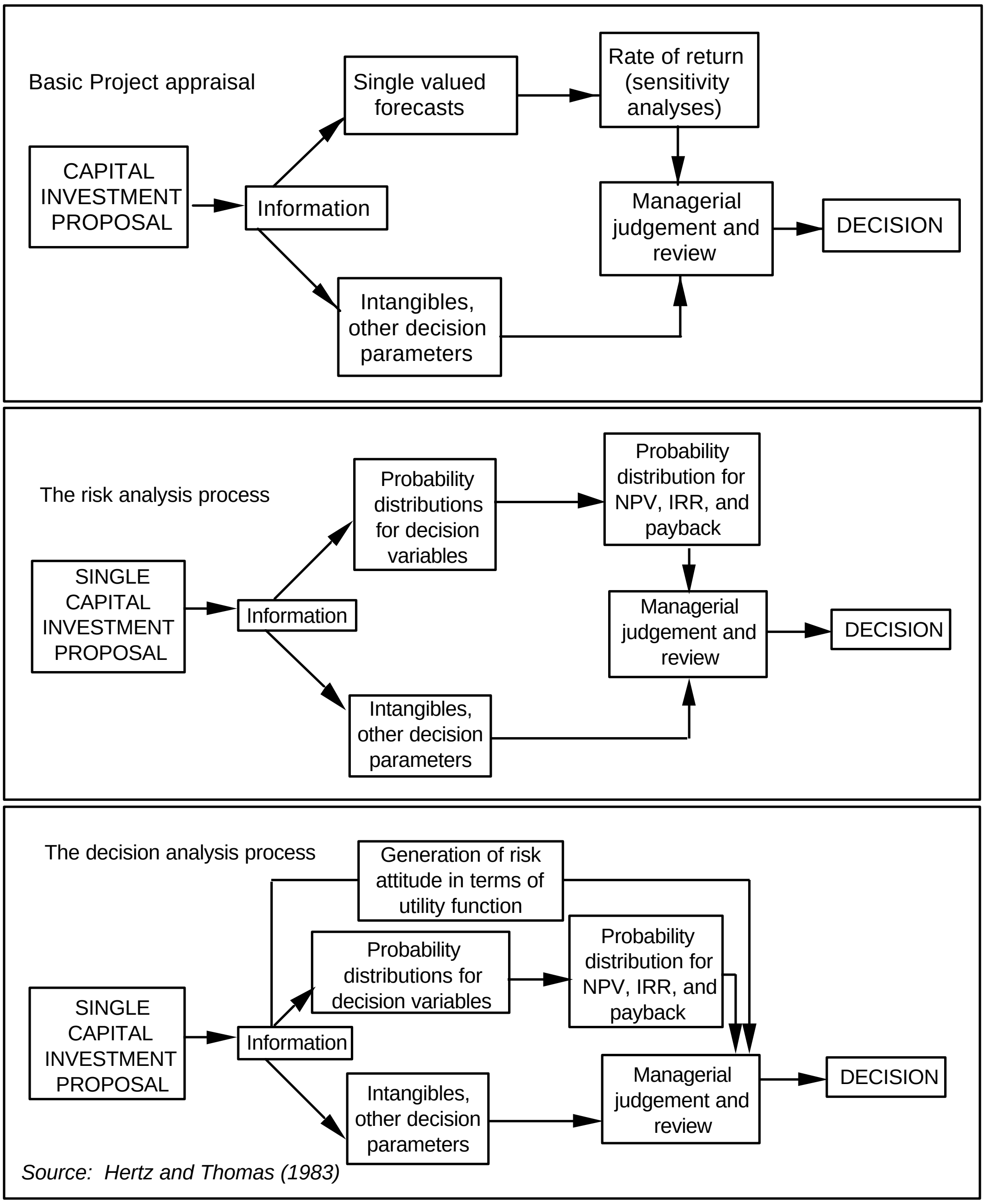
The above differences suggest a complementarity between ***probabilistic*** and ***deterministic*** methods. Deterministic methods by themselves fail to consider the likelihood of possible outcomes and, in the case of power planning, may result in selecting a technology even when its cost advantage is much smaller than the degree of uncertainty. Risk simulation promises the rigorous uncertainty analysis that formal algorithms are unable to offer. Some compromise may be achieved by a synthesis of probabilistic and deterministic models. In fact, the Sizewell B public inquiry (Layfield, 1987) concluded that probabilistic analysis is a helpful and complementary method for analysing uncertainty and therefore has been recommended in making proposals for future power stations.

The computational rigour of ***optimisation*** algorithms is a complement to the structural strength of ***decision analysis***. Optimisation can handle many quantitative variables while decision analysis can deal with the non-quantitative, subjective, and judgmental variables, thus incorporating the preferences and values of the decision maker. These two techniques offer a balance of hard/soft, prescriptive/descriptive, and deterministic/probabilistic characteristics. Davis and West (1987) suggest using linear programming to pre-generate alternatives for decision analysis to assess. This approach is similar to the modelling-to-generate alternatives decision support system of Brill et al (1990). There is a trade-off between complete analysis of all potential feasible solutions through linear programming and partial analysis of limited alternatives with decision analysis.

C.2.2 Sensitivity Analysis, Risk Analysis, Decision Analysis

Similarities in ***project appraisal***, ***risk analysis***, and ***decision analysis*** processes suggest a merger in figure C.2. In risk analysis, probability distributions replace single point forecasts, thereby adding more information to the analysis of basic project appraisal. Decision analysis brings in the values and preferences of the decision maker through utility functions which reflect risk attitude. These functional similarities strengthen the structural synergies between decision analysis and risk analysis in the previous figure C.1.

Figure C.2 Risk Analysis and Decision Analysis



Source: Hertz and Thomas (1983)

When the chance nodes of a conventional decision tree are replaced with continuous probability distributions, the analysis becomes that of a stochastic decision tree, as first described in Hespos and Strassman (1965). All quantities and factors, including chance events, can be represented by continuous, empirical probability distributions. The information about the results from any or all possible combinations of decisions made at sequential points in time can be obtained in a probabilistic form. The probability distribution of possible results from any particular combination of decisions can be analysed using concepts of utility and risk. However, like most decision trees, it quickly becomes messy, too large and cumbersome. The rapid increase in size is compounded by the assessment of additional uncertain quantities and the existence of time-series dependence between variables. Reduction methods which screen out dominated options would be helpful. Again, as seen in other modelling approaches, there is a trade-off between model complexity and assessment efficiency.

C.3 Structuring

C.3.1 Selection of Components

Synthesis is achieved within the context of available techniques or models. Rather than developing new model components, existing available models or techniques are used so that the main task of synthesis becomes that of ***structuring*** and ***integrating***. The selection of which technique to use is mainly driven by ***functionality***, e.g. optimisation, scheduling, simulation, etc. However, matching existing models or techniques to the problem also depends on ***execution costs***, ***input requirements***, ***applicability***, and other appraisal criteria (Ghosh and Agarwal, 1991.) These factors determine the ***number*** and ***kinds*** of techniques to use and the ***method*** of synthesis.

The high cost of model development (Balci, 1986) is one of the main reasons for using commercially available software packages that provide the algorithms and environments for model synthesis. In recent years, a proliferation of such software has facilitated rapid model development as well as model synthesis. The use of such software pushes verification and validation to others, reduces the overall validation effort, eliminates extensive programming, and allows model builders to concentrate on careful problem analysis, formulation, and sound data collection.

Therefore, ***software availability*** is an important determinant of technique selection. With regard to multi-technique software, Excel is beginning to do for modelling what statistical packages have done for data analysis.

The modeller's ***familiarity*** with model components greatly reduces modelling time and effort. Recent trends in user-friendliness, speediness of software upgrades, and standardisation of software and hardware help to flatten the learning curve and accelerate familiarity. However, familiarity produces a ***technique-driven bias***, steering the modeller away from those components that may be more conducive to the problem at hand. A model-building team, made up of different technique experts, instead of a single modeller may be needed to absolve this technique-driven bias.

The level of detail that can be incorporated by each technique varies greatly. Some are better at specifying technical and operational detail, while others are more capable of explicit uncertainty treatment. A ***manageable level of detail*** calls for technique specialisation, i.e. selecting the right techniques to represent the following types of detail: accounting, financial, causal, intangibles, non-linear effects, dependence, uncertainty, and time dynamics. In fact, this is one of the main reasons for synthesis, i.e. each model component is selected for its functional specialisation. It is necessary to keep the different levels of detail in check and to avoid "runaway complexity" to maintain manageability. The intention (Greenberger, 1981) is to keep the model as simple as possible while still capturing the essence of the problem.

The right mix of functionality also helps to achieve completeness in problem specification. Comprehensive modelling exhibits holism and balance so that no single aspect is distorted (Goldberg, 1987). Chapter 3 proposed ***complementarity*** by way of balancing the hard and soft, descriptive and prescriptive, and the deterministic and probabilistic for greater model completeness. Complementary techniques compromise their individual capabilities and limitations. For example, the combination of linear programming followed by sensitivity analysis exhibits a balance of hard, prescriptive, and deterministic linear programme with the soft and descriptive functionality of the latter technique. Such a sequence of scenario analysis, linear programming, and sensitivity analysis was actively used by the CEGB in the public energy inquiries.

To utilise the results of different components, ***compatibility*** is required. Compatibility means the ability to co-exist and work together. In model synthesis, compatibility resides at the data and theoretical levels. At the data level, techniques must be able to share data in the same form or convert them into the form they need. At the theoretical level, basic axioms must not be violated. Compatibility between components is a function of the communicability between different interfaces and protocols. Much of this depends on the interaction between the components. For instance, the heavy data demands of linear programming may not be met by the simplistic results of scenario analysis; therefore they are not compatible at the data interface level.

C.3.2 Ordering

The order in which model components are activated in the synthesis relates to the strategy of synthesis, which is discussed later in section C.5. In this section, we discuss two main kinds of ordering: ***increasing complexity*** and ***most relevant aspect first***.

The strategy of ***increasing complexity***, prescribed by Kendall (1969), starts with a simple model and works towards a more complex model by integration. By "simplicity," it is not clear whether we should build a crude but symbolic version of the target model and add incremental detail, such as the way we add flesh to the skeletal frame, or start with the smallest and simplest part of the problem to model and increase in scope and complexity. From a technique point of view, it seems reasonable to capture the main elements of the problem in a simple model initially and then refine by adding greater detail incrementally.

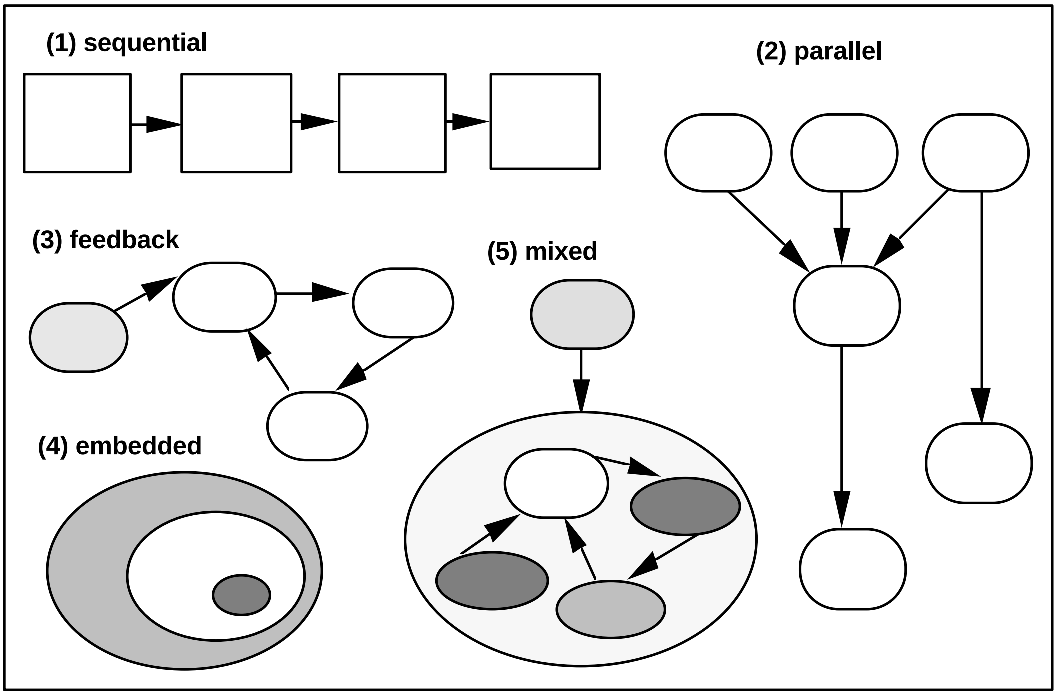
The strategy of decreasing importance calls for first capturing the ***most relevant aspect*** of the problem. Each subsequent technique or model-component addresses a less important aspect. For example, if the technology choice decision is the most important aspect of capacity planning, cost-benefit analysis or multiple attribute decision analysis should be selected. On the other hand, if the investment and retirement of plants over the forty year planning horizon is more important, we will need a scheduling device or resource allocation technique.

There are other beginnings. Using most “intuitive” model first to get the decision makers involved ensures that the basis for the model is user-driven and as a result credible. Scenario analysis suggests starting with “peripheral” models to avoid “anchoring bias.”

C.3.3 Linkage

A composite model organises the components so that the problem can be addressed in whole. Models that coexist in a given framework are part of a larger composite model only if their total contribution is greater than the sum of each. Model synthesis reflects the definition of a system: “the whole is greater than the sum of its parts.” While such components may co-exist and still stand alone and not interact, some linkage is required to pull the outputs together. In most cases, the components are linked in one or more of the following ways as illustrated in figure C.3. Two techniques are linkable if they are compatible, that is, communicable.

Figure C.3 Types of Model Linkages



1) The easiest method of integration is ***sequential***, whereby the results from one component are fed into the next. The sequential manner in which data is passed limits the number of model linkages. However, sequential modelling is quite time consuming, because a new stage cannot begin until previous stages have ended. Once a component has passed its output to the next, it can be "shut down" or freed to process more data in an assembly line fashion.

2) Computational speed can be increased by using more than one computer or a system with multi-tasking ability. Modularity also permits a kind of parallel processing, i.e. several models are run simultaneously, and the results fed into a final model. Because computer costs are quite high, in reality this ***parallel*** method is achieved sequentially, with the results of each model saved and entered into the final model at the end. Models in the same stage of analysis can commence in any order.

3) A variation of the sequential method is the incorporation of ***feedback or iteration***. Here, results from one model are fed into a previous model, increasing the number of interfaces to three if the previous model is not the first model used. Feedbacks occur in real life; thus, feedback modelling helps to refine the data and correct earlier assumptions.

4) In an ***embedded***, or nested synthesis, a smaller model resides in a larger model. The embedded model provides results which are needed by the larger model. In a concentric structure like the layers of an onion, outer components are highly dependent on inner components.

5) A ***multi-level*** modelling approach to deal with the long range capacity expansion problem proposed by Butler et al (1992) overcomes some of the difficulties and inadequacies of certain stand-alone techniques. For example, optimisation models are not effective in predicting performance based on short term uncertainties or fluctuations. Simulation would be more suitable, even though it is very data intensive. A hierarchical modelling process, i.e. modelling at more than one level, gives the opportunity to test the consistency of various types of decisions. Components at lower levels report or output to those at higher levels.

Other methods of synthesis are exemplified in practice. For example, the ***integrating*** module of the large energy model NEMS (DOE, 1994) is solely dedicated to converting, linking, and coordinating other components. In computer networking, ***gateways*** exhibit the same characteristics. Their primary task is to link different communication networks and protocols. These dedicated “integrators” act as middle-men or translators.

Possibilities for synthesis increase with the number of techniques. The greater the number of techniques and linkage methods, the greater are the number of permutable linkages. Linkages are restricted by the level of complexity we can handle and the operational implications of the components themselves. We have shown the possibilities, but there is a need for guidelines concerning which ordering to follow, which linkage is best, the circumstances under which model synthesis is preferred to model development, and whether a dynamic or static linkage is required. Other issues include unidirectional or bi-directional data transfer; interfacing; exact or reduced form for data conversions; when to standardise the inputs and output and types of inputs and outputs; different subsets of data passed to different modules or same data output to more than one module.

C.4 Weak and Strong Forms

We propose a distinction into weak and strong forms of synthesis to reflect the levels of integration and to characterise model synthesis in a more formal way. Whether or not a model is weak or strong depends on the factors addressed in the previous sections, i.e. to do with technique selection, ordering, and linkage.

A composite model is weakly composed (integrated) if the model components are not highly dependent on each other. The weakest form is given by a model which consists of stand-alone components, which can be run in isolation or in parallel and as such can be run independently of each other.

A model constructed from a combination of techniques or models refers to the existence of two or more components which are not necessarily linked. An integrated model is a combination model with linkage, hence a stronger form. In the strong form, model components are tightly integrated and contribute towards each other's informational and functional needs. Factors that contribute towards the strength of synthesis include the degree of cohesion, interaction, communication, contribution, and dependence. The stronger the synthesis, the greater is the integrity of the overall whole.

C.5 Strategies for Synthesis

The questions we have raised in model structuring pertain mostly to trading off obvious contradictory criteria, such as completeness and simplicity. Do we employ a top-down or bottom-up approach? Do we follow pre-specified instructions or do we follow our instincts? These issues of style and approach relate to an overall modelling strategy whose determinants are not yet clear. We suggest three strategies for synthesis: modular, hierarchical, and evolutionary.

C.5.1 Modular

Model synthesis by its very nature of combining different components is modular in approach. Miller and Katz (1986) recommends a modularisation scheme in which components are worked on and developed individually. ***Modularisation*** allows parts of the model to be changed without affecting the rest. It is easy to expand and contract. Different people can work on different parts of the model without having to understand each other. Modularising over time is equivalent to the staged approach where modules can be run in stages if necessary. Each module represents a complete, enclosed aspect of the problem. Both modular and staged approaches help to reduce the complexity and increase the manageability. Because different modules have different assumptions, some standardisation is required otherwise cognitive adjustments are needed.

C.5.2 Hierarchical

The concept of hierarchies is related to modularity but with the added dimensions of order, rank, and organisation. Thus a hierarchical synthesis is a more organised and stronger form of synthesis than modularisation.

Among the many approaches, Thompson and Davis (1990) describe the problem-driven method. A problem is broken into a series of decision levels, with the highest being ***aggregate***, that is, containing the smallest number of variables by grouping similar resources and subdividing the planning horizon. Linked hierarchically, each model-component addresses one of the decision levels. Hierarchical means “not equal,” so not everything can be passed up without filtering, screening, and condensing the data.

Nested techniques follow the hierarchical approach. Those at the top level are dependent on those at the bottom. Geoffrion (1987) supports this method of getting the big picture right and adding the details later.

C.5.3 Evolutionary

Evolutionary means becoming more ***developed***, more ***complex***, more ***differentiated***, more ***advanced***, and more ***integrated***. Balci (1986) conceives of an evolving model which is repeatedly redefined to reflect the new and increased understanding of the problem, the changing objectives, and the availability of new data. Ward (1989) suggests that models generating different levels of detail should be developed and introduced in an evolutionary manner to meet that level of ***integrative complexity*** most optimal or acceptable to the user.

An evolutionary approach reduces the effort involved in model synthesis by ***incremental additions*** in model detail. At each step, the level of complexity is kept manageable. Starting from a simple model with few parameters but encapsulating the big picture, additional factors and dimensions are introduced with a view to test the feasibility and attractiveness of different techniques. The exploratory way in which increasing level of detail is added enables the examination of intricate interactions between model parameters. The evolutionary approach facilitates a thorough analysis of uncertainty as the absence of a prescriptive element is conducive to learning and testing different possibilities.

In spite of these favourable characteristics, this method of investigation has several shortcomings. The biggest drawback to an evolutionary approach is the time commitment. Being exploratory in nature, modelling without a time constraint runs the risk of never finding anything. Modelling without an end goal or without defining the boundaries or standards beforehand is indefinite and inappropriate.

C.5.4 Other Approaches

In building decision support systems, Sprague and Carlson (1982) describes three tactical options: the quick hit, staged development, and the complete system. 1) The ***quick hit*** has the lowest risk in the short run but no re-assurance of *re-usability*, *flexibility*, or *generalisability*, as it is the approach of developing a specific model using whatever is available quickly and without any plans for upgrades. This lack of foresight means that it is likely to require much maintenance over time. 2) ***Staged development*** is *iterative* leading to an accumulation of knowledge over time. It is similar to the *evolutionary* approach which allows frequent opportunities to change direction of modelling. 3) Finally, the ***complete system*** approach is most comprehensive and ambitious and by default most time-consuming. It requires a lot of *foresight* and *planning* but bears the risk of technological obsolescence.

The need for a uniform modelling framework led Geoffrion (1987) to develop what is known as ***structured modelling***. It encompasses a formal mathematical framework and computer-based environment for conceiving, representing, and manipulating a wide variety of models that are hierarchically organised and partitioned. This modelling language differs from Lendaris' (1980***) structural modelling***, which refers to a collection of elements and their relationships with emphasis on qualitative structural (geometric and topological) rather than exact numerical or statistical properties. Both modelling paradigms have been developed to address the fragmented modelling world where low productivity and poor managerial acceptance prevail. Structured modelling is a bold attempt to reduce the multiple representation of models, interfacing problems, proliferation of different types of models, and resulting difficulties in model communication. The structured modelling language aims to provide ease of software integration. However, it is not commercially available at time of writing.

Model synthesis requires adjustment to different terminology within different modelling environments. The lack of a common modelling language and framework means that a consistent level of detail and scope cannot be maintained easily. Just as sensitivity analysis is used to identify a subset of factors, we need to develop criteria and methods to extract a subset of models from the grand design. Without sufficient empirical evidence and theoretical foundation, we are unable to give an exhaustive list of criteria and strategies for model synthesis. Nonetheless, these conceptual issues provide the basis for further research into model synthesis and the dimensions of composite models.