SEWPS

SPRU Electronic Working Paper Series

Paper No. 123

The Value and Costs of Modularity: A Cognitive Perspective

Stefano Brusoni*, Luigi Marengo, Andre Prencipe*** and Marco Valente******

(*CESPRI and CRORA, Bocconi University, **Università di Teramo, ***Università G. D'Annunzio di Pescara and SPRU, ****Università dell'Aquila and DRUID, Aalborg University)

August 2004

SPRU Science and Technology Policy Research

The Freeman Centre, University of Sussex,
Falmer, Brighton BN1 9QE, UK
Tel: +44 (0) 1273 877130
E-mail: a.prencipe@sussex.ac.uk
http://www.sussex.ac.uk/spru/

The Value and Costs of Modularity: A cognitive perspective

Stefano Brusoni* Luigi Marengo† Andrea Prencipe‡ Marco Valente§

Abstract

This paper discusses the issue of modularity from a problem-solving perspective. Modularity is in fact a decomposition heuristic, through which a complex problem is decomposed into independent or quasi-independent sub-problems. By means of a model of problem decomposition, this paper studies the trade-offs of modularity: on the one hand finer modules increase the speed of search, but on the other hand they usually determine lock-in into sub-optimal solutions. How effectively to balance this trade-off depends upon the problem environment and its complexity and volatility: we show that in stationary and complex environments there exists an evolutionary advantage to over-modularization, while in highly volatile – though "simple" – environments, contrary to usual wisdom, modular search is inefficient. The empirical relevance of our findings is discussed, especially with reference to the literature on system integration.

*CESPRI and CRORA, Bocconi University, stefano.brusoni@unibocconi.it
†Università di Teramo, marengo@unite.it
‡Università G. D'Annunzio di Pescara and SPRU, University of Sussex, a.prencipe@sussex.ac.uk
§Università dell'Aquila and DRUID, Aalborg University, mv@business.auc.dk

1 Introduction

In recent years, the concept of modularity has gained great visibility in a number of disciplines as diverse as management (e.g. Schilling, 2000), organisational sciences (e.g. Brusoni *et al.*, 2001), economics (e.g. Langlois, 2002; Marengo *et al.*, forthcoming), cognitive sciences (e.g. Fodor, 1983), American studies (e.g. Blair, 1988) and architectural and engineering design (e.g. Alexander, 1969; Suh, 1990). Overall, modularity has been proposed as a powerful organising principle of the evolutionary processes of both artificial and natural complex systems. Within social sciences, modularity principles have been applied to explain the evolution of products, organizational design and knowledge management, on the one hand; and the emerging patterns, and dynamics, of coordination and division of labour between firms, on the other. This paper focuses on the implications of modularity on problem-solving strategies.

It is often argued that, by adopting modular design strategies, firms can take responsibility for the design and development of separate modules. Thus, they can develop new products at a faster pace, as the integration of the final product is a matter of mix and match of 'black boxes' (e.g. Baldwin and Clark, 2001; Baldwin and Clark, 2000; Sanchez and Mahoney, 1996). This is made possible by advanced technological knowledge about component interactions that can be used to fully specify and standardise component interfaces and, therefore, to decouple the design of the product architecture (i.e. arrangement of functional elements) from the design of each module. Modularity, by simplifying design and development processes, would allow a greater division of labour across firms. As a consequence, firms can focus their capabilities on few modules or on the architecture. To fully exploit modularity, a 'grammar of action' (Argyres, 1999), or 'design rules' (Baldwin and Clark, 2000) have to be set in terms of neat and powerful routines that govern interfaces at product and organisational levels. Modularity emerges as a specific kind of problem-solving strategy, which entails a specific pattern of knowledge and task partitioning.

On this basis, formal models start from the already set grammar, or rules, and focuses on the exploitative aspects of modular design principles (Baldwin and Clark, 2000). Such grammar of action sets the rules of interaction across module, but also defines the organisation of problem-solving activities in terms of division of tasks, and their allocation to decoupled design teams. A modular architecture generates more options than an integral architecture because experimentation takes place at the level of the modules, rather than at the entire product (Baldwin and Clark, 2000). The finer the detail with which modules' interfaces are defined, the faster the pace of experimentation. That is to say, by adopting a modular product design strategy firms can speed up the process of exploitation of a given (modular) architecture.

When Baldwin and Clark (2001) discuss the advantages of modularity in terms of option value, they assume that a modular design is able to generate more options, and thus more value. They do so by analysing '*modularity as a financial force* [that] *adds options* to and thereby increases the financial value of complex design' (Baldwin and Clark, 2001, p.1, italics in original). We approach modularity as a specific example of problem-solving strategy. From this complementary standpoint, it becomes interesting to analyse what kind of problems can be framed and solved in a modular fashion (as opposed to an integral one), what are the characteristics of the search spaces generated by alternative problem-solving strategies, and what are the properties of the solutions so identified in the presence of varying degrees of environmental uncertainty.

More specifically, this paper builds upon Kauffman's work (Kauffman, 1993) exploring two related questions. Both questions aim at shedding light on the dynamic properties of modular search strategies. First, given the competitive environment, we want to understand whether there is a trade off between the 'speed' of search (enabled by modularity) and the 'breadth' of search (enabled by non-modular search strategies). Apparently, modular search strategies are indeed highly efficient in the short term (i.e., they provide 'higher value') enabling fast searches within a predefined search space. However, these gains might disappear in the long term, as 'slower' (i.e., less modular) search strategies catch up and reach better solutions as they can explore wider search spaces, exactly because they rely on less tightly defined 'design rules'. Second, and more fundamentally, we want to explore the relationship between alternative search strategies and changing competitive environments. Does modularity pay off in the presence of fundamental uncertainty? This is a basic question because, in dynamic terms, modularity may entail some risks: the more defined the grammar, and the search space it enables, the more structured, limited, and limiting is the search process. Firms may miss value-generating alternatives because they cannot escape the boundaries set by the existing modular design strategies.

This paper is organised as follows. Section 2 focuses on recent empirical research on modularity to highlight a few facts that demand an explanation. Section 3 introduced the model that we develop to approach a few key issues related to the dynamic properties of modularity. Section 4 presents the results and Section 5 concludes.

2 Modularity and problem-solving in organizations

One of the fundamental contributions of recent research on modularity is the identification of a series of constructs and key relationships that allows the connection to be studied between what firms do, how they do it, and what they need to know in order to it. Research on modularity reminds us about the complexity of the relationship between conceptualisations of firms as knowledge structures, and of firms as producers of goods and services. The need to disentangle this complex relationship is of paramount importance for both practical and theoretical reasons. First, in the context of increasingly globalised markets, ever more complex supply chains, and international manufacturing networks, corporate decision-making processes involve more and more actors, variables and criteria which lead to less and less transparency about who is deciding what, and on what basis. Second, and relatedly, the notion of 'means of production' has less and less to do with hardware, and more and more to do with information and specialised knowledge. Management tasks increasingly involve the monitoring, control and co-ordination of a widening range of useful, but highly heterogeneous, scientific and technological disciplines that are embodied in products of increasing complexity, in terms of components and functionalities.

Modularity, as a product and organisational design strategy, provides a possible answer to understand how this complex relationship is governed by modern corporations. Modularity allows the decoupling of complex artefacts into less complex, self-contained modules; each module, at the extreme, could become the sole business of a specialist firm, which would be only responsible over the specific module on which it focuses. Modularity makes complexity manageable by making it possible to run experiments at the level of modules, rather than the entire artefact, and in parallel (Baldwin and Clark, 2001). Moreover, modularity is 'tolerant of uncertainty' because particular elements of a modular design may be changed after unforeseen contingencies emerge, as long as the design rules are obeyed (ibid.).

The advantages of modularity seem to be particularly compelling in high technology settings, such as the aircraft engine industry. For example, two competing aircraft engine architectures are employed in the industry, namely two-shaft and three-shaft. According to industry experts, the three-shaft (launched by Rolls-Royce in the early 1970s) has turned out to be more effective in accommodating evolving customer requirements in terms of engine power due to its more modular architecture. In a three-shaft engine, the compression work is split across three compressors (low-, intermediate-, and high-pressure). Each compressor can be driven by its own turbine at its optimum speed. In a three-shaft design the mapping between functions and physical structures tends to be more one-to-one. A three-shaft engine, therefore, is more modular than a two-shaft. In a two-shaft design, in fact, the compression work is split between two compressors. The fan and the booster run on the same shaft. They rotate at relatively low speed to maintain the fan tip speed below supersonic. This limits the compression that can be achieved in the first part of the engine leaving high duty on the high-pressure compressor.

The embedded modularity of the three-shaft design enabled Rolls-Royce to exploit the same architecture, hence cutting the high development costs of new engines, to cater for a broader range of power requirements. The so-called thrust growth capability of the three-shaft has been much larger than the two-shaft. The thrust ratings of the JT9D and CF6 engine two-shaft families go from 44,250lb to 56,000lb and from 40,000lb to 64,500lb, respectively. Due to the exhausted growth potential of the JT9D and CF6 engines and to increasing thrust requirements asked by airlines to power larger aircraft, Pratt & Whitney and GE Aircraft Engines had to develop new engine families, the PW4000 and the GE90, respectively. Instead, the three-shaft architecture has been characterised by higher thrust growth. Rolls-Royce stretched the same engine architecture (the RB211) to develop engines from 42,000lb to 115,000lb. The in-built growth capability of the three-shaft design due to its more modular architecture gave Rolls-Royce a clear competitive advantage in terms of *speed* of development of new engines. In fact, they were able to introduce incremental changes in the original architecture (*mixing and matching* components) to meet a wider variety of aircraft makers' needs than their competitors. Emblematic is the example of the Trent engine 500 that has become the sole engine for the Airbus A340-500/600.

However, as noted by Baldwin and Clark (2000), the adoption of modular design strategies brings about costs too. First, the creation and dissemination of design rules is a rather expensive activity. Experimenting and testing on different modules is also costly. Moreover, increasing division of labour among firms also entails the traditional costs associated with the use of the market system (i.e., transaction costs) as well as agency costs related to the hold-up problem. The costs of creating the design rules deserve particular attention. Developing modular architectures is more difficult than developing integral ones. Achieving modularity requires a very precise understanding of the product functionalities, how they are allocated to components, and how the components interact. Thus, the choice of product architecture should be related to a company's product strategy. Ulrich (1995) argued that if a company wants to stress product performances, then the most appropriate choice would be the integral architecture, since global performance characteristics are optimised through this type of architecture. On the other hand, companies wanting to emphasise product change and variety, flexibility and upgradability, may well choose a modular architecture.

Furthermore, there are other costs related to developing a modular architecture. One must also consider the costs of the foregone opportunities that might have been exploited adopting different architectural choices. This problem is particularly cogent in an innovative context. The crucial point is the change of unit of selection that modularity implies with respect to an integral system: having 'fine' rather than 'coarse' units of selection makes the search process faster (selection operates on the finer scale of modules and therefore the selection environment is in a sense 'richer') but essentially 'local' and gets quickly locked into a local optimum, whereas in an integral system search is global, which implies that there is no lock-in, but search is much slower and in complex space there is a lot of wasteful search as nonsensical options can be generated.

Generally speaking, modularity can indeed highly increase the number of options generated and the speed of search for each module by creating standard interfaces between modules, but cannot avoid the lock-in. Modularity will make the system climb the local optimum faster, but cannot make it jump to another, higher valued, local optimum (Fleming and Sorenson, 2001). This can only be done by changing the architecture of modules. However, if companies embedded in a modular network design their interface, and specialised capabilities, around the current product architecture, how can they learn about alternative architectures? '[L]earning about changes in the architecture of the product is unlikely to occur naturally' (Henderson and Clark, 1990, p.28). Not only does learning about the architecture require dedicated efforts, but it also entails different kinds of organisations, people and skills (Brusoni and Prencipe, 2001). Moreover, 'architectural knowledge can emerge only after an organisation has developed sufficient experience with a problem to be able to fragment it into elements without losing critical information' (Henderson, 1992, p.127). Once an organisation recognises an architectural innovation as such, it has to change its 'orientation from one of refinement within a stable architecture to one of active search for new solutions.' (Henderson and Clark, 1990, p.17)

This change of orientation is what 'systems integrating' firms can do. Systems integrators are companies that rely on wide and dispersed networks of suppliers of specialised components and capabilities, yet maintain broad and deep in-house capabilities. These are firms that 'know more than they make' (Brusoni *et al.*, 2001) in order to be able to co-ordinate loosely coupled networks of suppliers, but also introduce new product architectures. The case of Fujitsu exemplifies the role played by systems integrators in the case of the hard-disk drive network. Fujitsu successfully managed the introduction of a new product architecture, stemming from a major technological breakthrough embodied into the magneto-resistive head, a component that displaced the pre-existing mechanical based technology. Relying on the modular architecture of the established product, Fujitsu, like other firms, relied on a decoupled network of external suppliers. However, unlike its competitors, Fujitsu continued to invest 'in systems knowledge and materials and component technology in its R&D labs.' (Chesbrough and Kusunoki, 2001, p.218) Fujitsu's systems knowledge went well beyond the range of products and components that the company produced in house. It enabled the firm to master the new, fast-moving technology and to navigate the dangerous waters of architectural innovation stemming from it. By knowing more than it needed for its own design and production, Fujitsu managed to avoid competency traps such as those described by Chesbrough and Kusunoki (2001) and Henderson and Clark (1990).

Brusoni *et al.* (2001) argued that cases like Fujitsu's show that decoupled, modular networks coordinated through markets and the exchange of codified knowledge (cfr. Sturgeon, 2002) are but particular cases of a more general model which link firms' knowledge and production boundaries. They argued that truly modular networks could emerge only when product interdependencies are predictable and when the specialised bodies of knowledge required are at the same stage of development. Interdependencies across components are predictable when a change in the design of one component entails a well-understood change in the design of other components and vice versa. The personal computer industry seems to fall into this situation (Langlois and Robertson, 1992). However, in the presence of product-level contingencies that cannot be fully predicted, 'co-ordinated activity is required to secure agreement about the estimates that will be used as a basis for action. Vertical integration facilitates such co-ordination.' (Teece, 1976, p.13). The automotive industry seems to fall into this category (Sako and Murray, 1999). Similarly, Davies (1999) studied the case of products characterized by unpredictable interdependencies across components as well as imbalances at the technological level: mobile phone systems. He showed that under high technological and environmental uncertainty, tightly coupled organizations in which integrated firms maintain in house both the knowledge and the production activities involved in the design and production of their final products and component units, have a competitive advantage. The advantage of a 'single vendor solution' lies in the supplier's experience in delivering 'a verified system in which all the components work well together, and can be integrated, tested and ready for service more rapidly than is possible in multivendor solutions.' (Davies, 1999, p.120) Specifically, Davies argued that the key advantage of Ericsson, the world leader throughout the 1990s, was its constant involvement in both architectural and component innovations, as well as its efforts to control production costs. It is worth noting that Ericsson adopted such a broad innovation strategy while introducing a modular approach to system design, in which core systems -centrally designed- were then adapted by regional 'competence centres.' (Edquist, 2003) However, this modular approach was accompanied, and enabled, by the development of in-house 'system competency', i.e. the competencies to design, build, market and support the entire system (McKelvey and Texier, 2000).

This is a key insight for understanding the dynamic trade off implied by modular search strategies. Companies like Ericsson, operating under conditions of fundamental technological and environmental uncertainty, have not disintegrated to be replaced by modular networks of specialised innovators. Quite the opposite, they have zealously invested into both the exploitation of the current standard, and into explorative activities to shape the next generation standard. Given the incredibly rapid rate of change in the technologies, regulatory environments, and competitive landscapes, firms like Ericsson could not run the risk of remaining stuck in any one specific research trajectory. Hence, the need to be involved throughout the 1990s in all the major development efforts that led the industry from the 1G mobile phone systems, to the still recent launch of 3G. Firms that followed more 'specialised' strategies lost their role of leaders despite their very early entry into the arena.

The above discussion informs the modeling exercise reported in the next section. First, we analyse formally the advantages brought about by modularity in terms of speed of adaptation to changing customer needs, highlighted by the Rolls Royce case. We show that speed of adaptation can give evolutionary advantages even though over-modular search strategies may not be the most efficient problem-solving strategy. Secondly, we explore the dynamic trade offs of modular and integral problem-solving strategies under conditions of fundamental uncertainty. Building upon the evidence summarized above, which focus on the role played by 'systems integrators', we show that integral problem-solving strategies may provide a way out to firms caught by surprise by unexpected changes in their competitive environment. Modular problem-solving strategies instead prevent organizations from rapidly abandoning their established way of doing things.

3 Model structure

Our model is made up of two elements: the **problem space**, which is exogenously given and characterized by a given degree of difficulty (expressed in terms of sub-problem decomposability) and the **problem solving organization** which searches in the problem space for superior solutions and tries to implement them. We assume that the organization is boundedly rational and therefore carries out its activities through a process of adaptive trial-and-error; at the same time, we also assume that this adaptive search is not purely random but is based on a (albeit possibly wrong) representation of the problem-space.

3.1 Problem Space

The problem-space is an extension and generalization of Kauffman's NK model of fitness landscapes (Kauffman, 1993). A fitness landscape is simply a mapping from a vector characterizing an entity's form to a payoff value. The original structure developed by Kauffman postulated a random interaction structure where a given element interacted with K randomly specified other elements. In the spirit of Simon's work on nearly decomposable systems and building on the modelling approaches of Marengo and Dosi (2003) and Marengo *et al.* (2000) we characterize problem environments as potentially consisting of more structured patterns of interaction. In particular, we develop and use two notions of complexity of a problem environment, namely decompositions and near-decompositions, which give a precise indication of the degree to which the problem can be decomposed into independent or quasi independent sub-problems (modules). As shown also in Frenken *et al.* (1999) Kauffman's 'K' can be a bad indicator of the decomposability of the problem: since blocks of epistatic interactions overlap and because of the randomness of fitness contributions, also problems with very low K values can be *de facto* non-decomposable.

More formally, the problem space is defined by N interdependent features which, for simplicity and without loss of generality, can assume only two states, labelled 0 and 1. The set of features comprising the problem space consists of $\aleph = \{x_1, x_2, \ldots, x_N\}$, with $x_i \in \{0, 1\}$. A particular configuration, that is a possible solution to the problem, is a string $x^i = x_1^i x_2^i \ldots x_N^i$. The set of configurations is characterized as: $X = \{x^1, x^2, \ldots, x^{2^N}\}$. The value, or fitness function, consists of a mapping from the set of configurations to the positive real numbers: $V : X \rightarrow \Re^+$. A problem is therefore defined by the couple (X, V).

As the size of the set of configurations is exponential in the number of components, whenever the latter is large enough, the state space of problem becomes much too vast to be extensively searched by agents with bounded computational capabilities. One way of reducing its size is to decompose it into sub-spaces[1]. Let $\Im = \{1, 2, \ldots, N\}$ be the set of indexes, and let a **block** $d_i \subseteq \Im$ be a non-empty subset of this set, and let $\mid d_i \mid$ be the **size of block** d_i, i.e., its cardinality.

We define a **decomposition scheme** (or simply **decomposition**) of the space $\aleph$ as a set of blocks:

$$D = \{d_1, d_2, \ldots, d_k\} \text{ such that } \bigcup_{i=1}^{k} d_i = \Im$$

Note that a decomposition does not necessarily have to be a partition; that is, there may be some overlap among the particular decompositions d_i.

Decompositions structure the nature of the organizational and technological search process. Search for alternative bases of action does not take place on a holistic, system-wide basis but tends to be local and to approach different facets of the problem in a sequential manner (Cyert and March, 1956). In this spirit, a new configuration is generated and tested by picking a block $d_j \in D$ at random and some (at least one and up to all) components in

[1] A decomposition can be considered as a special case of a search heuristic. Search heuristics are in fact ways of reducing the number of configurations to be considered in a search process.

this block (and only in this block) are mutated, obtaining a new configuration x^h which may differ from the original configuration x^i only in those components belonging to block d_i. If $V(x^h) \geq V(x^i)$, then x^h is retained and becomes the new current configuration; otherwise, x^h is discarded and x^i continues to be the current configuration.

We say that a decomposition scheme D^* is an optimal decomposition of the problem if multiple iterations of this search procedure are always able (after repeated random mutations) to locate the globally optimal configuration(s), starting from any initial configurations. That is, the scheme is such that there is no lock-in into suboptimal configurations. In general, there can be many optimal decomposition[2]. For instance, if D^* is an optimal decomposition, all decompositions which can be obtained by the union of some of its blocks will also be optimal decomposition. However, among the set of decompositions satisfying this criterion, we are particularly interested in the finest optimal decomposition(s), i.e., the one(s) whose blocks have minimal cardinality. Blocks in the finest optimal decompositions represent the finest sub-problems into which the overall problem can be decomposed and still be optimally solved.

We can classify problems in terms of their finest optimal decomposition. In particular, the following types will be widely referred to in our subsequent analysis:

1. Non-decomposable problem, for which the finest optimal decomposition is the degenerate one: $D^* = \{1, 2, \ldots, N\}$
2. Nearly-decomposable problems (Simon, 1981) whose finest optimal decomposition is made of non-disjointed (partially overlapping) blocks. Two cases are particularly interesting:
   - partially overlapping blocks, such as for instance:
   $$D^* = \{1, 2, 3, 4\}, \{3, 4, 5, 6\}, \{5, 6, 7, 8\}\}$$
   - nested blocks, such as for instance:
   $$D^* = \{1\}, \{1, 2\}, \{1, 2, 3\}, \{1, 2, 3, 4\}, \ldots, \{1, 2, 3, 4, 5, 6, 7, 8\}\}$$
3. Decomposable problems, whose optimal decomposition is made only of disjointed blocks. Furthermore, this decomposition of disjointed blocks can be:
   - coarse, if blocks are not all singletons (i.e., they contain more than one component)
   - fine, if all blocks are singletons (i.e., they contain only one component)

Only in this last case is the problem 'simple' and optimally solvable through N separate local search processes and therefore fully modularizable.

3.2 Techno-Organizational Problem-Solving

A decomposition scheme is a sort of template which determines how new configurations are generated and can therefore be tested by a selection mechanism. In large search spaces in which only a very small subset of all possible configurations can be generated and undergo testing, the procedure employed to generate such new configurations plays a key role in defining the set of attainable final configurations.

[2] See Marengo and Dosi (2003) for a more formal and detailed account of the properties of optimal and sub-optimal decompositions and for an algorithmic procedure which computes them.

Blocks in our model can be considered as a formalization of the notion of modules used by the growing literature on modularity in technologies and organizations (Baldwin and Clark, 2000) and decomposition schemes are a formalization of the notion of system architecture which defines the set of modules in which a technological system or an organization are decomposed

We will assume that boundedly rational agents can only search locally in directions which are given by the decomposition scheme: new configurations are generated and tested in the neighborhood of the given one, where neighbors are new configurations obtained by changing some (possibly all) components within a given module.

Among all the decomposition schemes of a given problem, we are especially interested in those for which the global optimum becomes reachable from any starting configuration. One such decomposition always exists, and is the degenerate decomposition $D = \{\{1, 2, 3, \ldots, N\}\}$ for which of course there exists only one local optimum and it coincides with the global one. But obviously we are interested in – if they exist – finer decompositions and in particular in those of minimum size. The latter decompositions represent the maximum extent to which the search space can be subdivided into independent modules coordinated by a simple selection mechanism, with the property that such selection processes can eventually lead to optimality from any starting condition. On the contrary, even finer decompositions will not in general (unless the starting configuration is "by chance" within the basin of attraction of the global optimum) allow decentralized selection processes to optimize.

Minimum size decomposition schemes can be found recursively with the following procedure which we describe informally[3]:

Let us re-arrange all the configurations in X by descending rank $X = \{x^0, x^1, \ldots, x^{2^N-1}\}$ where $x^i \succeq x^{i+1}$.

The minimum size decomposition can be computed as follows:

1. start with the finest decomposition $D^0 = \{\{1\}, \{2\}, \ldots, \{N\}\}$
2. check whether $x^0 \in P(x^i, D)\ \forall x^i\ i = 1, 2, \ldots, 2^N - 1$, i.e., if there is a path leading to the global optimum from every other configuration for decomposition D, if yes STOP
3. if no, build a new decomposition D^1 by union of the finest blocks for which condition 2 was violated and go back to 2.

Let us finally provide an example for illustration.

Example: consider a system of 3 binary components and imagine having a selection (value) landscape described by the following table:

CONFIGURATIONS	RANKING
100	1
010	2
110	3
011	4
001	5
000	6
111	7
101	8

[3]The complete algorithm is quite lengthy to describe in exhaustive and precise terms. Its Pascal and C++ implementations are available from the authors upon request. See also Marengo and Dosi (2003) for a more formal treatment of decompositions and their properties.

If the system is fully modular (i.e., there are three modules {1},{2} and {3}) and the current state is 001 then search will always be locked into the local optimum 010 and never reach the higher value solution 100. To see this just notice that there is no one-mutation and value-increasing path leading from 001 to 100: for instance the first module, which is initially set to 0 can never switch to its optimal configuration 1 because switching to 1 always decreases the value given to the other modules. In order to ensure that maximum value can always be achieved one needs coarser modules: for instance in this example the finest possible set of modules is composed of two modules: {1,2} and {3}.

3.3 Near decomposability

When building a decomposition scheme for a problem, we have looked so far for perfect decomposability, in the sense that we require that all blocks can be optimized in a totally independent way from the others. In this way we are guaranteed to decompose the problem into perfectly isolated components which can be solved independently. This is however very stringent a requirement: even when interdependencies are rather weak, but diffused across all components, we easily tend to observe problems for which no perfect decomposition exists.

One can soften the requirement of perfect decomposability into one of near-decomposability: one no longer requires the problem to be decomposed into completely separated sub-problems, i.e., sub-problems which fully contain all interdependencies, but only wants sub-problems to contain the most "relevant"interdependencies whereas less relevant ones can persist across sub-problems. In this way, optimizing each sub-problem independently will not necessarily lead to the global optimum, but to a "good"solution. In other words we construct **near-decompositions** which give a precise measure of the trade-off between decentralization and optimality: higher degrees of decentralization and market coordination, and therefore higher speed of adaptation, can be obtained at expenses of the optimality of the solutions which can be reached.

Let us re-arrange all the configurations in X by descending rank $X = \{x^0, x^1, \ldots, x^{2^N-1}\}$ where $x^i \succeq x^{i+1}$, and let $X_\mu = \{x^0, x^1, \ldots, x^{\mu-1}\}$ with $0 \leq \mu \leq 2^N - 1$ be the ordered set of the best μ configurations.

We say that X_μ is reachable from a configuration $y \notin X_\mu$ and for decomposition D if there exists a configuration $x^i \in X_\mu$ such that $x^i \in P(y, D)$.

We call basin of attraction $\Psi(X_\mu, D)$ of X_μ for decomposition D the set of all configurations from which X_μ is reachable. If $\Psi(X_\mu, D) = X$ we say that D is a **μ-decomposition** for the problem.

μ-decompositions of minimum size can be found algorithmically with a straightforward generalization of the above algorithm which computes minimum size decomposition schemes for optimal decompositions.

Higher degrees of decomposition and decentralization can be attained by giving up optimality and providing a precise measure for this trade-off. In order to provide an example we generated random problems[4] of size $N = 12$ all characterized by $|D| = 12$ (i.e., they are not decomposable). Figure 1 shows the sizes of the minimum size decomposition schemes as we vary the number m of acceptable configurations (average on 100 random landscapes).

[4]Here problems are random NK landscapes *à la* Kauffman (1993) with $N = 12$ and $K = 5$

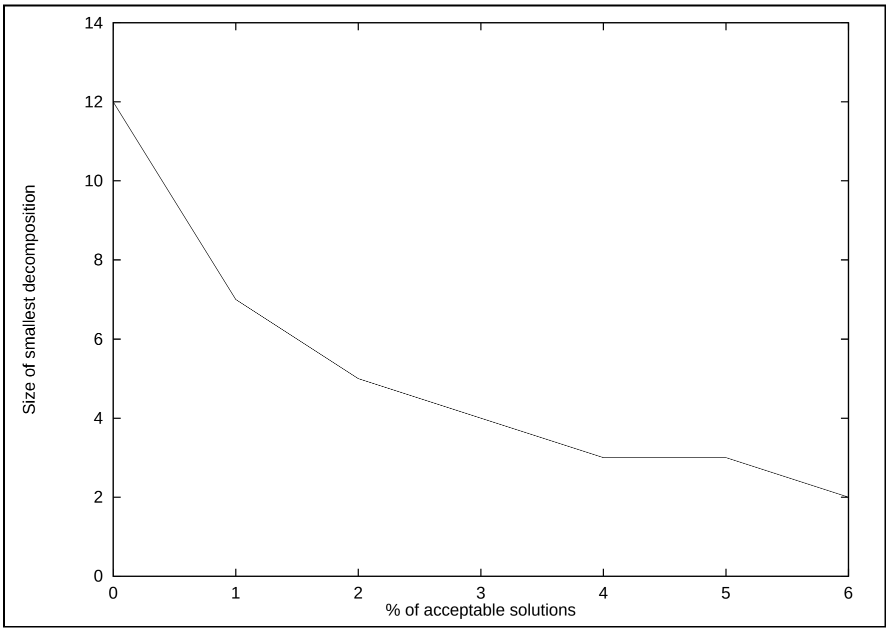

Figure 1: Near decomposability

This figure shows that sharp reductions of complexity and time of search[5] can be obtained by accepting sub-optimal "satisficing" solutions. Thus there is a trade-off between optimality and speed of search which has interesting implications which will be examined in the next section.

4 Speed and Optimality of Search strategies

4.1 The evolutionary advantage of excess modularity

So far we have characterized a system in terms of its decomposability. Now, using this toolbox we are able to construct a problem whose structure is perfectly known and then test the relative efficiency of different search strategies. We will concentrate on the comparison among research strategies based upon different degrees of modularity. A search strategy consists of a rule that produces a new configuration starting from a current one; if the new configuration is better than the previous one it is retained, otherwise it is discarded.

In general, fully modular search strategies, i.e. those in which each component is optimized independently of the others, are not optimal (Kauffman, 1993) as they can locate the globally optimal configuration only if there are no interdependencies among components.

In Frenken *et al.* (1999), the properties of other search strategies are analyzed based upon coarser modules. In fact, we can consider a search strategy that divides the N components of the configuration in modules, each containing a given number of components, say S[6]. The generalized S-search strategy consists of choosing one module (instead of a single component) and mutating one or more components in the module.

Put more precisely, the steps of a generalized S-search strategy are: 1) choose randomly

[5] Every reduction of 1 in the size of the decomposition schemes implies that the number of solutions to be tested and the expected time of search are cut down by one half

[6] Without loss of generality and for the sake of simplicity, we assume that each group module contains exactly S components and, therefore, that N/S is an integer number.

one of the N/S modules; 2) choose randomly an integer number Z in the range [1,S]; 3) choose randomly Z bits among the S of the chosen module; 4) switch the state of the selected bits; 5) test the fitness of the newly produced string; 6) accept the new string if it produces a higher fitness than the current string, or reject it otherwise[7].

We can draw a parallel between the complexity of the problem (decomposability) and the search strategy (modularity). For example, suppose that we consider a problem whose minimal decomposition is: 1,2,3,4,5, 6,7,8,9,10, ..., N-4,N-3,N-2,N-1,N. Obviously, a S-strategy based upon the same modules is always able to reach the optimal configuration. Instead, lower dimensional S-strategies (finer modules) are usually locked into local optima. Higher level S-strategies (larger modules) are still able to find the global maximum, but they take much longer.

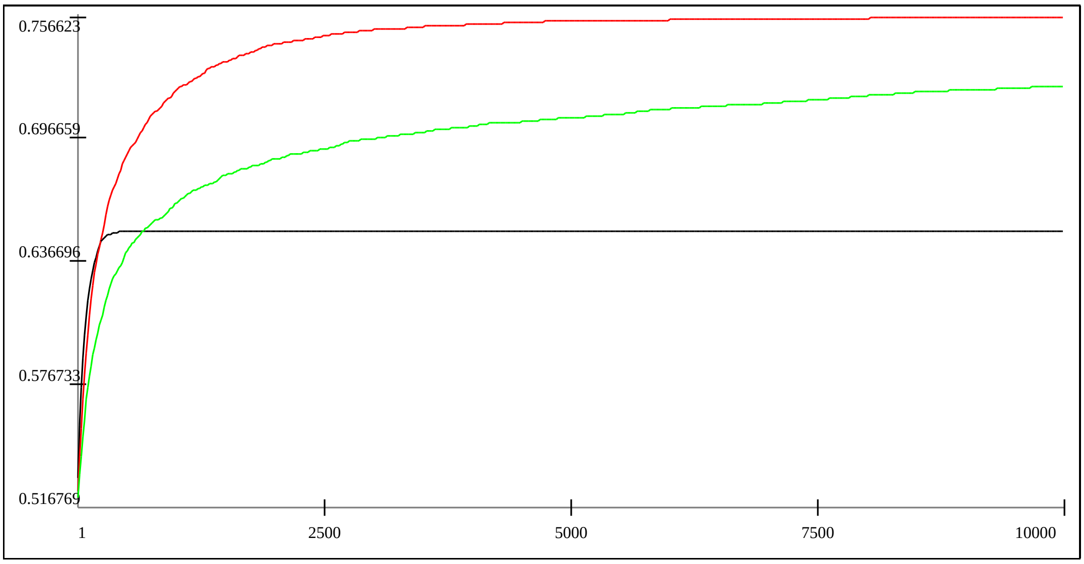

Figure 2: Average fitness values of three populations searching on a problem space with N=40. The first population (black series) adopts S=1, the second S=5 (red series) and the third S=10 (green series).

Figure 2 shows a simulation on a random problem à la Kauffman (1993) of three populations of 100 agents each, that independently search a random problem of size N=40 and optimally decomposable into 10 modules of size 5 each, starting from the same (randomly drawn) initial configuration. The three populations adopt search strategies based upon modules of size, respectively, 1,5 and 10. In other words the first class of agents are over-modularized, the second use optimal modules and the third are under-modularized.

All agents with the "optimal" strategy at the end of the simulation have managed to reach the global maximum, having explored 10,000 configurations (a portion of less than $1/10^{10}$ of the total number of configurations). Instead, none of the agents in the other two populations manage to reach the optimum in the same time. Over-modular agents in the first population quickly get stuck in different local optima, from which they are unable to unlock. The third population, though moving continuously up-hill, is very slow, since they explore a much larger portion of the search space.

This property of the S-strategy derives from an analytical result that shows that the maximum number of strings required to be tested in order to select with certainty the maximum fitness is a linear function of 2^{K+1} (Frenken *et al.*, 1999).

[7] Obviously, the S-search strategy with S=1 is the one bit mutation.

This result concerns “modular” worlds, where basic components are grouped into modules and components within a module influence each other’s performance, while having no relation with components outside their module. In these cases the correct way to search for the optimal configuration of the components is to act at the level of modules, since acting at the level of components is bound to provide sub-optimal solutions, whereas working at higher levels is too slow.

However, if we consider the initial steps of the simulations, reported in Figure 3, we can see that the first population of agents adopting the finest modules possesses a big initial advantage over the “optimal” strategy, lasting for many periods, although its agents are doomed to be stuck in a local optima. Why this temporary advantage? The reason lies in the quicker response of the modular strategy (S=1) in respect to the more integrated ones. The strategy aiming at testing smaller variations of the current string is able to test a higher number of possibilities, quickly improving the fitness in the beginning of the search.

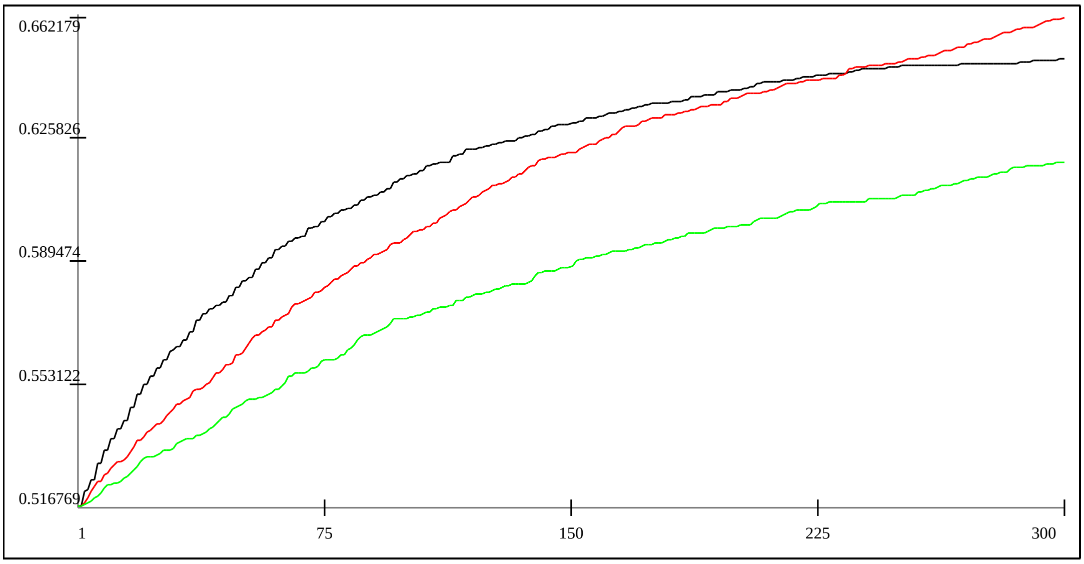

Figure 3: First 300 steps of the same results shown in Figure 2

The advantage of the more modular strategy in the first population for the initial period may produce interesting dynamic properties. Suppose that the search strategy is used as an evolutionary competition among individuals of different species. The fitness value of an agent provides its probability of being removed from the population and of producing an off-spring of its same species.

The selection mechanism can be simply represented by the removal of a fixed number of agents (the ones with the worst fitness values) and their replacement with “copies” of the best ones, where a copy is a new agent adopting the search strategy of the copied agent (i.e., the same modularity), and being assigned an initial random configuration. Figure 4 shows the number of agents in each population with the same settings as above, with the selection mechanism acting every 50 time steps.

Clearly, the first population can exploit its initial advantage, whereas the “optimal” strategy (and even more so less modular) do not have the time to unfold their superiority. In other terms, while strategy with S=5 is globally optimal, under the selection pressure it can become evolutionary dominated by a modular strategy that, though prevented from reaching the global optimum, is faster in reaching moderately higher results. This property gives the modular strategy an evolutionary edge over integrated strategies.

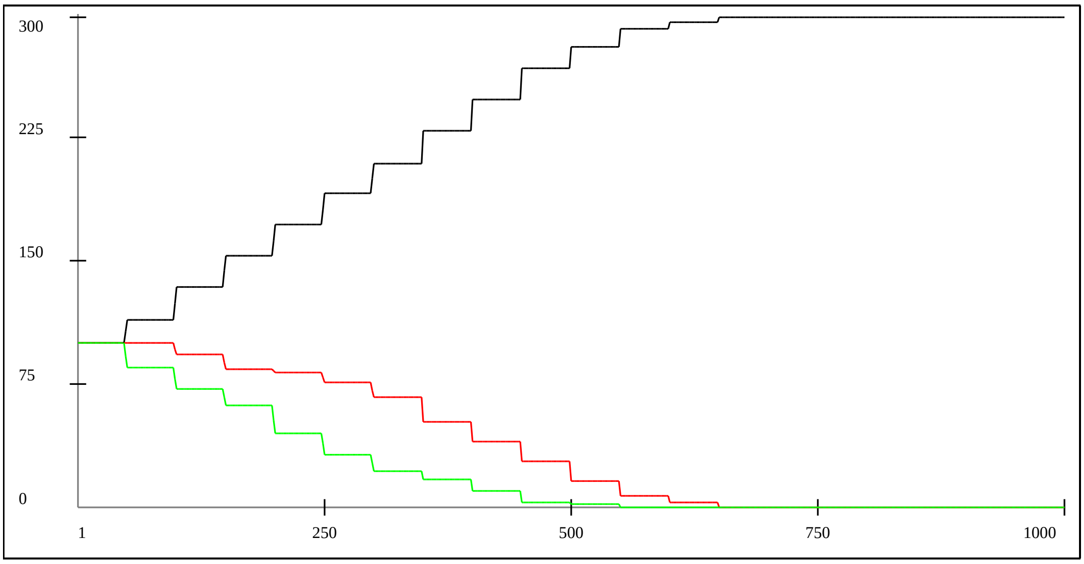

Figure 4: Number of agents in the three populations with S=1 (black), S=5 (red) and S=10 (green). Selection applies every 50 time steps, replacing 20 agents.

These results show that a trade-off exists between speed and optimality. Aiming at the optimal solution of a problem entails the necessity of taking into account account all the interdependencies among components. However, this enlarges enormously the space to be searched and therefore the time required to explore it. Conversely, an over-modular approach focused on the exploration of each component independently from the others, may be doomed to be limited in the maximum performance that can be finally obtained, but has the advantage of providing quick-and-dirty improvements that, in a highly competitive environment, may be the key to evolutionary success.

4.2 Volatility and the Revenge of Integrated Systems

We have seen that a modular search strategy enjoys an evolutionary advantage both when the problem space is actually decomposable and also when it is not so, because of higher speed of adaptation. There are, however, cases when an integrated search strategy outperforms modular ones: this happens contrary to the current wisdom[8], when the environment is highly volatile.

In this section in fact we show that when the fitness values of configurations change rapidly, then even if at that moment in time the problem space is fully decomposable, modular strategies are rapidly outperformed and selected out by integrated ones. A modular search strategy consistently climbs up from its current position with "steps" which are smaller the finer the modules. On the contrary, an integrated search strategy can "jump" to locations far away from the current one. In a stable environment the former strategy is more effective as it quickly climbs a local optimum while the latter spends a long time wandering around the problem space. But in a highly volatile environment, it can happen that an agent finds itself in a location with very low fitness (the bottom of a "well"): in this case short steps will be too slow a strategy for climbing out of the low fitness area, while long jumps have a high probability of quickly finding a higher fitness

[8]Baldwin and Clark (2000) for instance maintain that modularity is especially advantageous in uncertain and changing environments. See also Langlois (2002).

level[9]. Conversely, when agents are in a high fitness location, the integrated strategy has a much lower probability of bringing about a further improvement than a modular one, but the latter can only bring about small improvements.

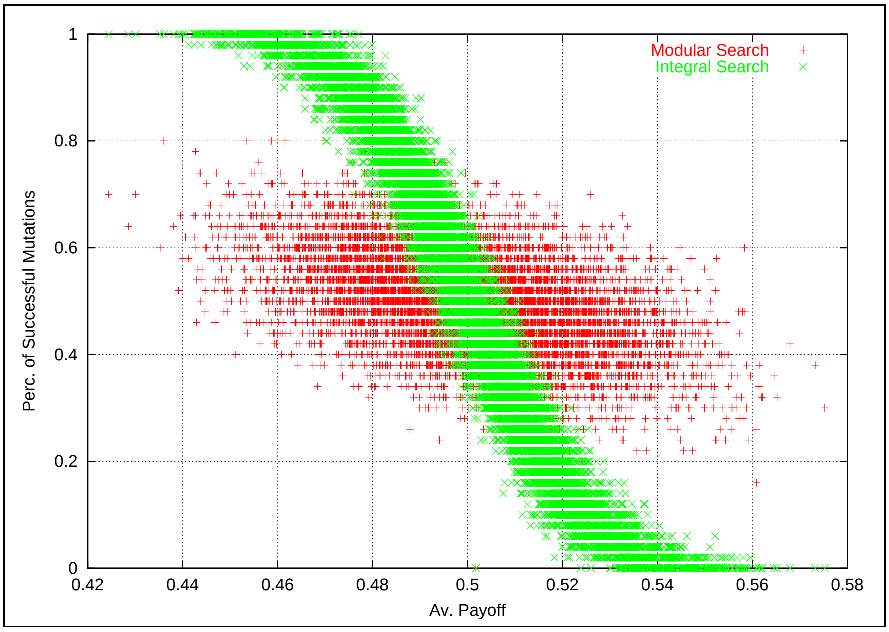

Figure 5: Average percentage of agents producing a successful mutation as a function of the average fitness values for two populations: the first applies a one-bit mutation strategy (Modular) and the second applies a N-bit search strategy (Integral). Simulation performed with N=1000, K=0. The two populations are made up of 50 agents each, running for 10,000 time steps. At each time step all agents are relocated in the same (randomly chosen) point

The superiority of integrated strategies is the outcome of two factors: probability of improvement and expected size of improvement. Concerning the probability, a modular strategy is only slightly more likely to provide a fitness improvement when applied to a low fitness point than to a high fitness one. Instead, integrated strategies are much more likely to provide improvements when the current fitness is low, and very much more unlikely when it is high. To confirm this statement, in Figure 5 we show the percentage of successful mutations produced in two populations using a modular and an integrated search strategy respectively.

Concerning the size of the improvement, the integrated strategy is likely to provide large improvements when starting from low fitness locations, whereas it will provide low improvements when the starting location has high fitness. The modular approach will provide small improvements in both circumstances (see Figure 6).

This result contradicts the common wisdom that modularity provides robustness against volatile environments. In fact, it shows that the opposite is true. In highly volatile environments, the systemic recombination allowed by integrated and non-modular search strategies allows for the radical change which is much more appropriate than the local search which characterizes modular search.

This result is also confirmed when the volatility concerns only part of the system and agents know which parts are affected. That is, we use a shock that modifies randomly

[9] A more formal treatment of this proposition can be found in Valente (2003), where it is shown that when fitness levels rapidly change, the expected gains of integrated strategies are higher than those of modular strategies.

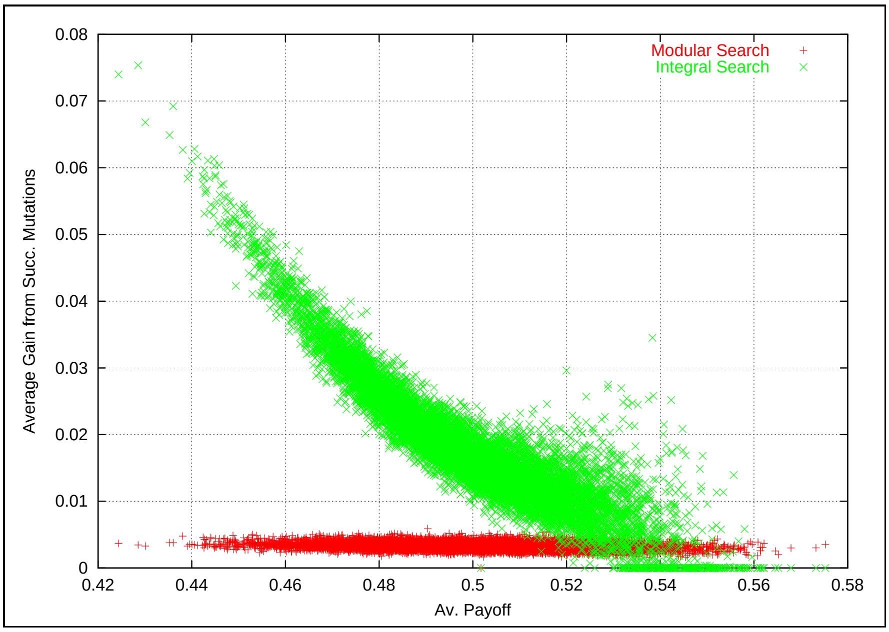

Figure 6: Average gain from successful mutations as a function of the average fitness values. Same conditions as in Fig. 5.

the fitness provided by half of the available dimensions, instead of modifying all of them. When the shock occurs, agents know which half of their environment has been affected and direct the mutation efforts to repair that portion of their current activities.

For this second experiment we defined two populations of agents, where the integrated strategy can modify either the first or the second half of the string. We allowed agents to apply their strategy for 20 time steps, after which a shock produces the modification of the fitness contributions provided by half of the bits. Immediately after the shock agents attempt a mutation choosing one (for modular strategies) or more (for integrated ones) of the bits affected by the shock.

The results do not change sensibly, though in this experiment modular agents were allowed to climb a stable environment for some time, and the shock affected only half of the environment. It can be noticed that the "clouds" of points is in fact centered above the expected fitness value of 0.5, since agents are able to climb somewhat from the average level during the 20 stable periods.

Figures 7 and 8 report the same statistics seen above for this new set up, considering only the time steps immediately after the shocks. Figure 8 reports the average gain from successful mutations whereas Figure 7 reports the percentage of successful mutations for integrated and modular strategies.

5 Discussion and Conclusions

Taking a complementary perspective to Baldwin and Clark (2001)'s financial view of modularity, this paper has attempted to single out the advantages and disadvantages of problem-solving strategies in relation to the time horizon of the search and the volatility of the environment. Simulation results show that modular search strategies are particularly efficient in the short-term. This is due to the fact that modular search strategies enable a greater number of possibilities to be quickly tested and therefore fast adaptation. Speed

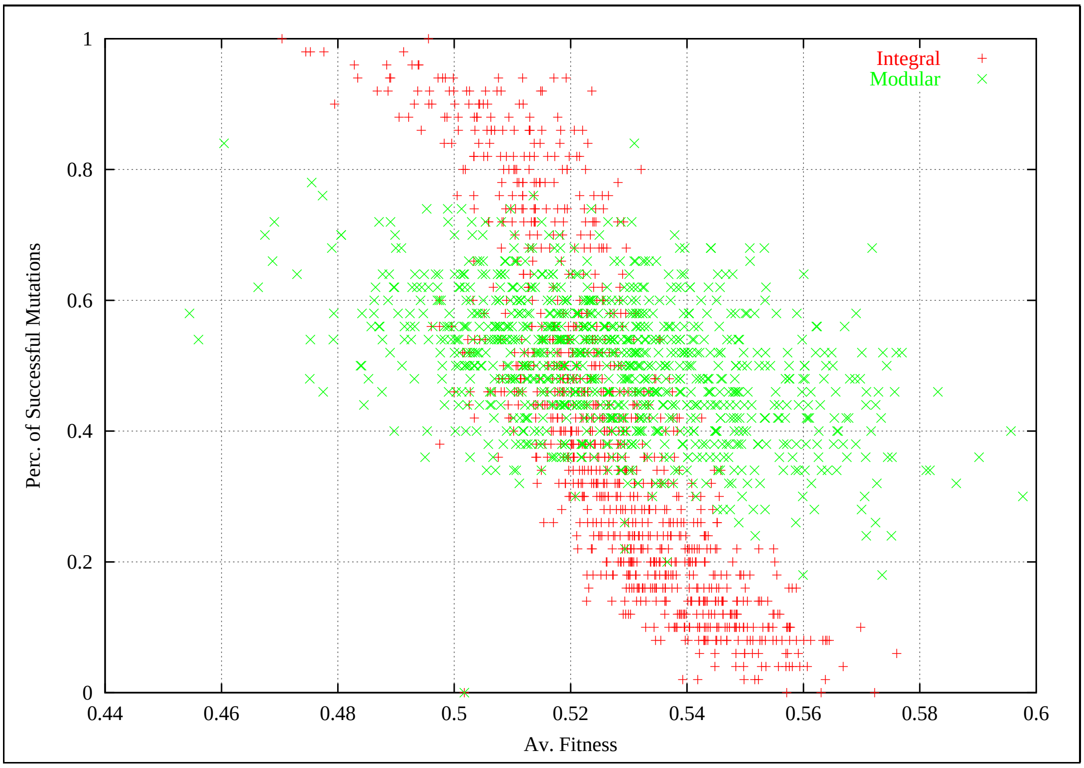

Figure 7: Percentage of successful mutations for two populations of "modular" and "integrated". N=100, K=50, and a shock affecting either the first or the second 50 dimensions occur every 20 steps. Agents "know" which half of the environment underwent the shock and attempt to repair that part. Data used for the figure concern only the periods just after a shock.

is therefore an important advantage of modularity. Our results also showed that there is a cogent trade-off between speed of search and breadth of search, however. In the long-term, integral search strategies reach higher picks. By enabling a broader search, integral search strategies slowly catch up with and eventually overtake modular ones in terms of solutions reached. According to our study, the advantages of modularity tend to be short-lived because of intrinsic limitations. Modular search strategies enable a quick adaptation and a rapid improvement of fitness, but this focuses the search process on close local optima and prevents searching in the broader context.

However, short-term advantages may lead to sustainable evolutionary advantages at the population level. That is to say, populations of organizations that rely on over-modular problem-solving strategies may come to dominate populations of organizations that know the 'right' pattern of modularization, even though their problem-solving strategies do not deliver the highest fitness solution. In a rather speculative manner, one might argue that this result captures a key feature of the competitive struggle in the PC industry. The open and modular architecture of IBM-compatible PCs has given Microsoft (and Intel) a great competitive advantage over Apple with its proprietary and more integrated architecture. Despite some experts' opinion that Apple can deliver better technical solutions, Microsoft definitely is the market leader.

The paper also analyzed the effectiveness of alternative search strategies in relation to different characteristics of the competitive environment. The dimension of the environment analyzed was the stability - volatility one. The simulation exercise showed that in stable

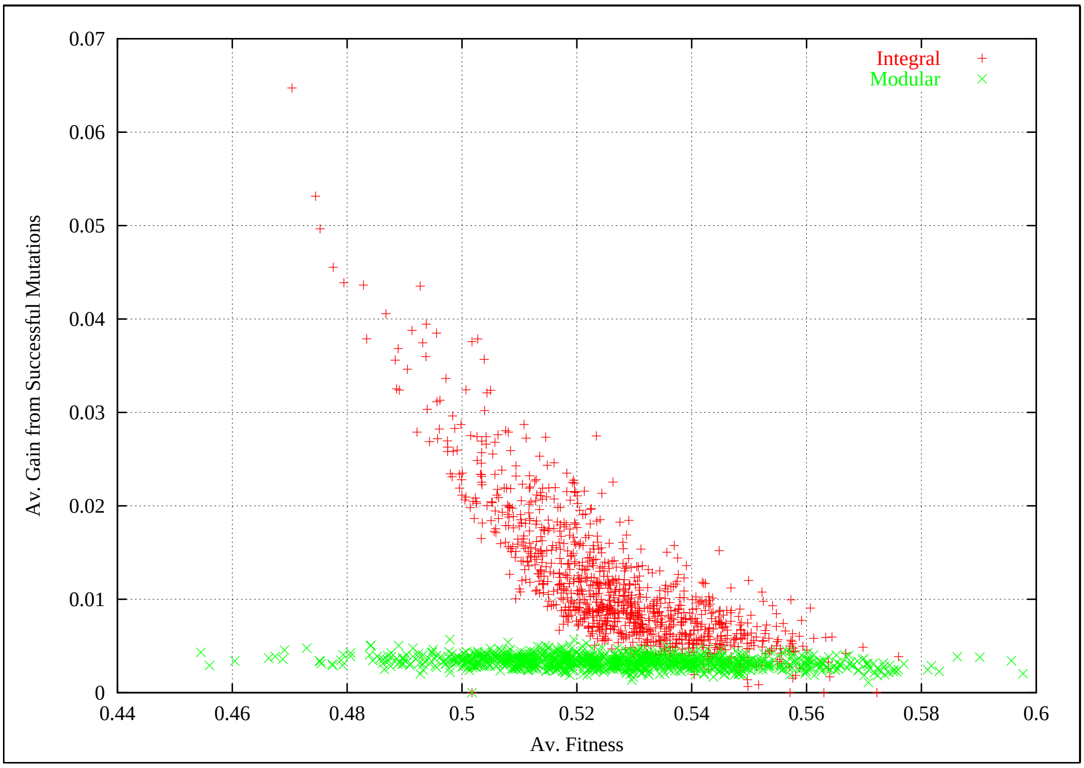

Figure 8: Average gain from successful mutations for two populations of "modular" and "integrated" agents with random changes of fitness contributions on half of the components. Same conditions as in fig. 7.

environments, modular search strategies are more effective because of the above-mentioned fast response attitude. For example, one might argue that this was the situation enjoyed by Rolls Royce in the 1980s and 1990s, i.e., two decades during which the civil aviation industry went through a phase of continuous and stable growth. Whether the competitive advantage built upon the modular architecture of the three-shaft engine is defendable in a more turbulent environment is to be seen. Our simulation results would suggest some (preliminary) skepticism.

In highly volatile environments, modular search strategies are shown to become trapped in local optima and incapable of moving to different ones. This is due to the fact that modular search strategies enable only small definite changes that do not allow jumping away from local optima that become wells when the competitive environment suddenly changes. On the contrary, an integral search strategy might well enable large improvements that are able to jump to locations distant from, and safer than, the current one. In our view, this result captures the key role played by systems integrating companies. For example, evidence from the mobile phone industry suggests that systems integrators need to remain involved in exploratory research that looks beyond the boundaries set by current architectures in order to be able to lead the process of development of successive generations of mobile telephony systems. The narrow, but fast, search enabled by over modular problem-solving strategies might lock incumbents into peaks that, once turned into wells, they will not be able to escape because of their inability to 'jump' toward new local optima.

The line of reasoning presented in this paper needs be extended in several directions. Two seem most promising to us. First, in this paper we have conceptualized, and modeled, organizations as 'pure' problem solvers. While problem solving is a fundamental activity performed by organizations, it is not the only one. It is necessary to build models capable of providing richer characterizations of firm behavior. This is not just a quest for realism. In a previous paper, we argued that decisions to outsource production (and other functions) are different from decisions to outsource technological knowledge. In other words, there is a gap between firm production and knowledge boundaries (Brusoni and Prencipe, 2001; Brusoni *et al.*, 2001). Such a gap is fundamental in explaining how incumbents manage to react to rapid technical change, like Fujitsu did in the hard disk drive industry. We need to develop new families of models capable of distinguishing and coping with the joint dynamics, of the division of labor and that of knowledge. A paper by Dosi *et al.* (2002) begins to explore these issues.

Second, and related to the first point, we need to explore further the relationship between firm organization and environmental changes. Specialization and loose coupling are often advocated as a suitable way of organizing business in the context of fast environmental changes. For example, Levinthal (1997) argued that '[t]ightly coupled organizations cannot engage in exploration without foregoing the benefits of exploitation' (p. 949). However, cases like Fujitsu and Ericsson, show that some incumbents can actually manage both exploration and exploitation activities. In our view, to make organizational sense of these issues one needs to clearly distinguish between the organization of manufacturing activities, and the organization of more knowledge-intensive activities. Moreover, learning and manufacturing processes are embedded in dense networks that link manufacturers of the final product to suppliers of components and specialized knowledge. In innovative, fast-changing environments it becomes more and more difficult to pinpoint firms (whether systems integrators or mere assemblers) as the correct unit of analysis. Problems are solved 'socially', and understanding how problem-solving strategies unfold within communities of specialists that cut across firm boundaries is a challenge to both practitioners and scholars.

References

Alexander, C. (1969), *Notes on the Synthesis of Form*, Harvard University Press, Cambridge, Mass.

Argyres, N. (1999), "The Impact of Information Technology on Coordination: Evidence from the B-2", *Organization Science*, **10**, pp. 162–180.

Baldwin, C. and Clark, K. (2001), "The Value and Costs of Modularity", mimeo, Harvard Business School.

Baldwin, C. Y. and Clark, K. B. (2000), *Design Rules: The Power of Modularity*, MIT Press, Cambridge, MA.

Blair, J. G. (1988), *Modular America Cross-Cultural Perspectives on the Emergence of an American Way*, Greenwood Press, Westport, Conn.

Brusoni, S. and Prencipe, A. (2001), "Unpacking the black box of modularity: Technologies, products, organisations", *Industrial and Corporate Change*, **10**, pp. 179–205.

Brusoni, S., Prencipe, A. and Pavitt, K. (2001), "Knowledge Specialisation, Organisational Coupling, and the Boundaries of the Firm: Why Do Firms Know More Than They Make?", *Administrative Science Quarterly*, **46**, pp. 597–621.

Chesbrough, H. and Kusunoki, K. (2001), "The modularity trap: innovation, technology phase-shifts, and the resulting limits of virtual organizations", in I. Nonaka and D. Teece, eds., "Managing Industrial Knowledge: Creation, Transfer and Utilization", Sage Publications, Thousand Oaks, CA.

Cyert, R. M. and March, J. G. (1956), *A Behavioral Theory of the Firm*, Prentice Hall, Englewood Cliffs, NJ.

Davies, A. (1999), "Innovation and competitiveness in Complex Product Systems: The case of mobile phone systems", in S. Mitter and M. Bastos, eds., "Europe and Developing Countries in the Globalised Information Economy", UNU Press, London.

Dosi, G., Marengo, L. and Levintahl, D. (2002), "The Uneasy Organizational Matching Between Distribution of Knowledge, Division of Labour and Incentive Governance", mimeo.

Edquist, C., ed. (2003), *'The Internet and Mobile Telecommunications System of Innovation: Developments in Equipment, Access and Content*, Edward Elgar, Cheltenham, UK.

Fleming, L. and Sorenson, O. (2001), "The Dangers of Modularity", *Harvard Business Review*, **September**, pp. 20–21.

Fodor, J. A. (1983), *The Modularity of Mind*, MIT Press, Cambrdige MA.

Frenken, K., Marengo, L. and Valente, M. (1999), "Interdependencies and Adaptation", in T. Brenner, ed., "Computational Techniques to Model Learning in Economics", pp. 145–165, Kluwer Academics.

Henderson, R. M. (1992), "Technological Change and the Management of Architectural Knowledge", in T. Kochan and M. Useem, eds., "Transforming organisations", Oxford University Press, Oxford.

HENDERSON, R. M. and CLARK, K. B. (1990), "Architectural innovation: The reconfiguration of existing product technologies and the failure of established firms", *Administrative Science Quarterly*, **35**, pp. 9–30.

KAUFFMAN, S. A. (1993), *The Origins of Order: Self-Organization and Selection in Evolution*, Oxford University Press.

LANGLOIS, R. N. (2002), "Modularity in technology and organization", *Journal of Economic Behavior and Organization*, **49**, pp. 19–37.

LANGLOIS, R. N. and ROBERTSON, P. L. (1992), "Networks and innovation in a modular system: Lessons from the microcomputer and stereo component industries", *Research Policy*, **21**, pp. 297–313.

LEVINTHAL, D. (1997), "Adaptation on Rugged Landscapes", *Management Science*, **43**, pp. 934–950.

MARENGO, L. and DOSI, G. (2003), "Division of Labor, Organizational Coordination and Market Mechanism in Collective Problem-Solving", Working paper 2003/04, St. Anna School for Advanced Studies, Pisa.

MARENGO, L., DOSI, G., LEGRENZI, P. and PASQUALI, C. (2000), "The Structure of Problem-solving Knowledge and the Structure of Organizations", *Industrial and Corporate Change*, **9**, pp. 757–788.

MARENGO, L., PASQUALI, C. and VALENTE, M. (forthcoming), "Decomposability and Modularity of Economic Interactions", in C. W. and D. RASSKIN-GUTMAN, eds., "Modularity: Understanding the Development and Evolution of Complex Natural Systems", .

MCKELVEY, M. and TEXIER, F. (2000), "Surviving Technological Discontinuities through Evolutionary Systems of Innovation: Ericsson and Mobile Telecommunication", in P. P. SAVIOTTI and B. NORBOOM, eds., "Technology and knowledge: from the firm to innovation system", Edward Elgar, Cheltenham, UK.

SAKO, M. and MURRAY, F. (1999), "Modules in design, production and use: Implications for the global automotive industry", mimeo, Paper presented at the International Motor Vehicle Program (IMVP) Annual Sponsors.

SANCHEZ, R. and MAHONEY, J. (1996), "Modularity, flexibility, and knowledge management in product and organization design", *Strategic Management Journal*, **17**, pp. 63–76.

SCHILLING, M. A. (2000), "Towards a General Modular Systems Theory and its Application to Inter-Firm Product Modularity", *Academy of Management Review*, **25**, pp. 312–324.

SIMON, H. A. (1981), *The sciences of the artificial*, MIT Press, Cambridge, MA, 2nd edition.

STURGEON, T. (2002), "Modular production networks: A new model of industrial organization", *Industrial and Corporate Change*, **11**, pp. 451–496.

SUH, N. (1990), *The Principles of Design*, Oxford University Press, New York.

TEECE, D. (1976), *Vertical Integration and Vertical Divestiture in the U.S. Oil Industry*, Stanford University Institute for Energy Studies, Stanford, CA.

ULRICH, K. T. (1995), "The role of product architecture in the manufacturing firm", *Research Policy*, **24**, pp. 419–440.

VALENTE, M. (2003), "Well Theorem: Failure of Modularity in Volatile Modular Landscapes", Mimeo.