

# An Active Symbols Theory of Chess Intuition

ALEXANDRE LINHARES

*EBAPE/FGV, Praia de Botafogo 190/426, Rio de Janeiro 22250-900, Brazil; National Institute of Space Research, LAC/INPE, Av Astronautas 1758, S.J. Campos SP12227-010, Brazil;  
E-mail: linhares@fgv.br*

**Abstract.** The well-known game of chess has traditionally been modeled in artificial intelligence studies by search engines with advanced pruning techniques. The models were thus centered on an inference engine manipulating passive symbols in the form of tokens. It is beyond doubt, however, that human players do not carry out such processes. Instead, chess masters instead carry out perceptual processes, carefully categorizing the chunks perceived in a position and gradually building complex dynamic structures to represent the subtle pressures embedded in the positions. In this paper we will consider two hypotheses concerning the underlying subcognitive processes and architecture. In the first hypothesis, a multiple-leveled chess representational structure is presented, which includes distance graphs (with varying levels of quality) between pieces, piece mobilities, and abstract roles. These representational schemes seem to account for numerous characteristics of human player's psychology. The second hypothesis concerns the extension of the architecture proposed in the Copycat project as central for modeling the emergent intuitive perception of a chess position. We provide a synthesis on how the postulated architecture models chess intuition as an emergent mixture of simultaneous distance estimations, chunk perceptions, abstract role awareness, and intention activations. This is an alternative model to the traditional AI approaches, focusing on the philosophy of active symbols.

**Key words:** active symbols, artificial intelligence, chess, cognitive modeling, psychology of intuition

## 1. Introduction

It finally took place on May 11th, 1997. Roughly four decades after Herbert Simon's 1957 bet that "within 10 years a digital computer will be the world's chess champion", the IBM massively parallel computer Deep Blue at last defeated world champion Gary Kasparov (Simon and Newell, 1958). This event, once considered far-fetched (Hofstadter, 1979), brought a substantial public man-versus-machine debate concerning topics as elusive as the ultimate limits of technological evolution and the demise of humanity – a debate that continues to this date. However, as the news spread around the planet, a consensus (as much as there can be any consensus in this type of debate) was rapidly reached: whatever merits Deep Blue happened to have, it was still incapable of exhibiting genuine intelligence.

After that episode, research on computer chess seems to have gradually faded to the background. Kurzweil (2002), for instance, points out that “...the level of interest in computer chess waned considerably after 1997. After all, the goal had been achieved, and there was little point in beating a dead horse. IBM cancelled work on the project, and there has been no work on specialized chess chips since that time.” But despite the slowdown, computer chess still holds much scientific interest. It may let us, for instance, take a closer look at the inner subcognitive workings of the human mind. The structure of the chess chunk may give us a glance of the organization of human memory and possibly of human concepts, the precise structuring of which still remains mysterious (Hofstadter and FARG, 1995; Margolis and Laurence, 1999).

In a remarkable research manifesto, Winkler and Fürnkranz (1997, 1998) discuss two distinct dimensions of ‘effort in AI research’, using the chess domain as an example of the chasm between *psychological knowledge and engineering advances*. They conclude by pointing out that the greatest value in research is precisely at the *junction between human-compatible processing and machine-compatible processing*, and that to obtain meaningful results at this junction requires more effort than that required to obtain results for either dimension alone. While we have seen a computer “world champion”, we still lack a computational theory of how even a good amateur plays chess. Nothing in Deep Blue models the system as having ideas. This is exactly the objective of this article: to analyze the key requirements brought by human chess psychology, carefully sketching a cognitive model for their implementation, and to detail an architecture capable of modeling the information processing involved when humans select a move (Figure 1).

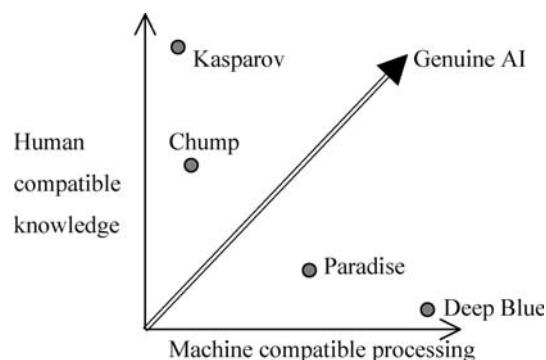


Figure 1. The chasm in AI research (After Winkler and Fürnkranz, 1997). The authors propose that the most valuable research leading to true AI lies at the junction between human-compatible psychological knowledge and machine compatible processing. They do, however, hypothesize that this is precisely the point that demands most research effort in order to obtain meaningful results.

Wilkins (1980) pointed out a perfect example of this chasm. Take the board in Figure 2, for instance. An experienced player will instantly recognize some of its broad patterns, which naturally lead to the solution. Note the following patterns:

- (i) The pawn chains block each other, and the white pawn on F6 is the only one capable of moving,
- (ii) The white pawn on F6 cannot promote as long as the king remains in such a short distance,
- (iii) The black king cannot move away from the pawn on F6 and thus cannot attack the white pawn structure at its unprotected end,
- (iv) The blocking pawn chains divide the board in two sides, the only passage between them lying on the queen rook file.

Having noticed these patterns, the solution becomes clear: white has a winning sequence by moving across the queen rook file and either forcing the black king to respond to a movement from the pawn at F6 or to distance itself from the white king, either way permitting white to safely promote at least one of its pawns.

After these broad patterns are perceived, expectations are formed, and even inexperienced players can intuitively understand such lines of reasoning. Computer chess, in its classical form at least, gets lost in the combinatorial explosion and thus has great difficulty with this position. Wilkins (1980) has noted that, "since white's advantage would take more than 20 ply to show up in most evaluation functions used by computer programs", these programs

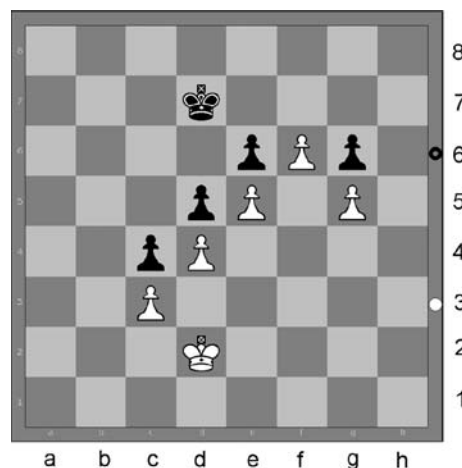


Figure 2. How can the intuitive understanding of a position arise? How does a strategy emerge? How is it carefully planned and evaluated? In this position, chess masters are able to immediately perceive some important features that lead to a winning strategy for white.

should probably decide to move the white king to a more centered position. Modern off-the-shelf computer software are able to eventually find the solution, though in an erratic manner, due to the horizon effect. Harry Foundalis (personal communication) points out that under many parameter settings, the programs have the white king performing ‘backward’ moves in its run for the mate. Furthermore, if we select the ‘solve for mate’ function of the Chessmaster 7000 program – in which the program must find the correct plan *from this starting state* without the help of having pieces moving around – then, after laboring for days, over three billion movement combinations were considered by the program without finding the right course of action. By now it should be clear that a program capable of using the concepts mentioned above would not even consider moving the white king to the center of the board. In this article, we will discuss how such a problem may be handled by an intuitive chess machine, a problem that perhaps has been best stated by Atkinson (1993):

“The master’s superior play is due to ‘sense of position’, an intuitive form of knowledge gained from experiencing a great variety of chess situations. Intuitive knowledge is perhaps the most important component of expertise in any human discipline, yet intuition has been one of our least understood phenomena. Intuition is an automatic, unconscious part of every act of perception. It is often associated with emotion: an intuitive realization can ‘feel right’ and evoke a sense of satisfaction or, equally, can elicit that sinking feeling of impending frustration. Intuition is nonverbal, yet trainable. Intuition is not part of present-day machine chess”.

Though this ‘intuitive strategy’ process still remains elusive after a decade, a considerable number of researchers have been working on pattern-analysis chess. For instance, Morales (1994, 1996) presents a first order system that can learn patterns in the form of Horn clauses, and is able to automatically construct a simple King and Rook against King (K RK) endgame strategy. However, the system is based on inductive logic programming and thus is not able to properly categorize complex cases, falling prey to the same criticisms pointed out in Linhares (2000) of symbolic systems applied to the problems posed by Bongard (1970). Another interesting approach is presented by Wilkins (1980), with the PARADISE system, which was able to find combinations as deep as 19 ply (in 1980) by encoding a large body of production-rule based knowledge. The system, however, does not present a cognitively plausible model, because it “still uses a systematic best-first search in the space of possible plans” (Winkler and Fürnkranz, 1997). Finkelstein and Markovitch (1998) present a model that has some resemblance to the model proposed here, as it is centered on direct attacks and defenses and indirect attacks and defenses (our model generalizes this notion by considering a

sparse distance graph – see Section 4 below). Their paper describes “a language for representing move patterns, and algorithms for learning, storing, retrieving, and using them”. Unfortunately, it still does not provide a psychologically plausible model that includes most of what is known about human chess cognition. For example, those move patterns do not seem to be able to account for long-distance relationships, such as the (numerous-movements long) upcoming white king attack on Wilkins’s example in Figure 2.

This paper is organized around the classical computer science distinction between *problem analysis* and *solution proposals*. For example, the field of *computational complexity* studies problem analysis, attempting to find answers to questions such as whether or not the traveling salesman problem can be solved in polynomial time, *independently of the solution method used*. On the other hand, studies of *solution proposals*, such as analysis of algorithms, consider precisely the dynamics of each particular approach, e.g., concluding that some particular method for the traveling salesman problem may be faster than others, or that it may give greater guarantees of optimality. Section 2 considers the part of *problem analysis*. Psychological plausibility is added as a restriction to the problem of devising an intuitive chess machine, by including some new architectural principles that should be accounted for by any intuitive machine chess system. In Section 3, we briefly review an example of a system based on an *active symbols* cognitive architecture. In Section 4, an analogous cognitive model is postulated for the case of chess; it is argued that such architecture of an intuitive chess machine can model the emergence of sense of position in a psychologically plausible way, enhancing our understanding of how humans select a move. Finally, some new research problems are elaborated and, it is claimed that the postulated system is able to account for the principles put forth in Section 2.

Let us begin with the argument that the core of the problem of chess intuition lies in perceptual processes.

## 2. Perception as the Key to Human Chess Capability

The Cuban World Chess Champion José Raul Capablanca once remarked on his personal, subjective, experience: “I know *at sight* what a position contains. What could happen? What is going to happen? You figure it out, *I know* it!” In another occasion, talking about the numerous possibilities that less-skilled players usually consider on each board position, he bluntly remarked: “I see only one move: The best one.” (Atkinson, 1993). It is this process of ‘just knowing’, or of ‘seeing only the best move’, that marks human intuition and which has been so elusive to computational models.

How can this be? Here is a question that borders on the mysterious: When selecting a move, what kind of subcognitive information processing takes place in an expert player's mind (Hofstadter, 1985)? How are these processes organized? And the crucial question: How can *meaning* emerge?

Chess has been traditionally viewed in artificial intelligence as a classic symbolic-paradigm search problem: given an initial state, and an overarching goal, the system searches through a myriad of candidate combinations in search of an optimum. Deep Blue, the paramount example, is said to have achieved astonishing speeds up to 330 million positions evaluated *per second* in the match against Kasparov (Campbell et al., 2002). However, as psychologists have repeatedly pointed out, humans do not play chess by brute-force search – indeed, it is well established that skilled human players hardly ‘search through’ a great number of nodes of the game tree, and upper bound estimations of this number reach at most 100 moves per position (de Groot, 1965; Simon and Chase, 1973; Chase and Simon, 1973a, b; Charness, 1981; Simon and Shaeffer, 1992; de Groot and Gobet, 1996; Gobet, 1998). It is understood that the key to understanding human chess capability lies not in search (or inference capacity) but really in the underlying subcognitive processes of knowledge acquisition and perception.

Since robust psychological studies have confirmed that humans do not employ any deep searching of the game tree (evaluating potential movements and their implications *ad nauseam*), but only sporadically resort to counting the number of movements, let us observe a first architectural principle for an intuitive chess machine:

**Architectural Principle 1.** *An intuitive program should not evaluate a large number of move combinations, and only sporadically evaluate move implications (i.e., search the game tree).*

This widely known difference between humans and machine players is hardly the only one. It has also been shown that humans carve board positions into sub-patterns referred to as *chunks*, and eye saccade studies have shown that effort is concentrated on the ‘most important’ regions of the board, with special emphasis on pieces under ‘chess relations’ – that is, attacks and defenses (see Simon and Chase, 1973). A recent study, for instance, points out that experts produce a greater proportion of fixations on ‘relevant’ pieces than novices, and experts also produce more fixations on empty squares than intermediates – leading to the argument that expert chess players perceptually encode chess configurations (instead of individual pieces) – see (Charness et al., 2001).

This is also in contrast to machines such as Deep Blue, which check billions of possibilities bureaucratically, by giving virtually the same static evaluation function to each evaluated board position (making the same ‘questions’, extracting the same information, and spending the same effort at each ply, even in diametrically distinct board positions) (Figure 3).

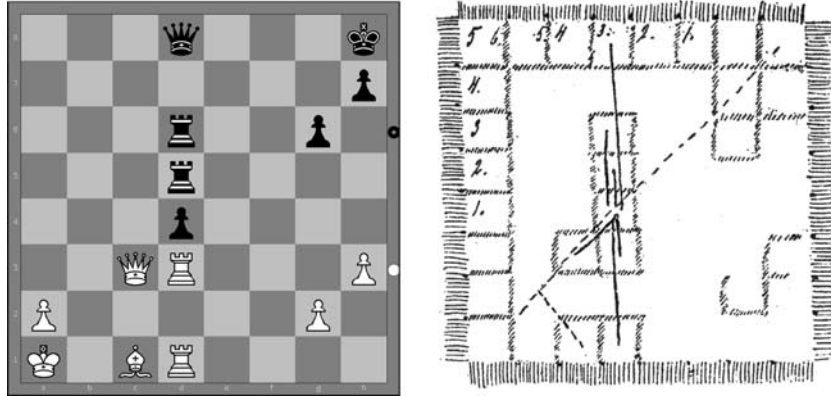


Figure 3. Representation of a position reported by an expert chess player (After Binet, 1894).

We are thus presented with a second architectural principle:

**Architectural Principle 2.** *An intuitive program should concentrate its attention (movement of eye saccades) to pieces in chess relations.*

While attention in psychological experiments can be roughly measured by studying eye saccades, let us leave the meaning of the term attention in an intuitive program open at this time, to be discussed later, in Section 5.

As mentioned, another stark contrast between human players and systems such as Deep Blue is that players rapidly and effectively ‘see’ the meaning of each board position, having an immediate intuition of where the game may be leading to: “*I know at sight what is going to happen*”; a knowledge that comes from the acquired chunk structures. But not all human players are able to extract meaning from board positions. This kind of knowledge must be acquired, and it may be probed. Simple experiments have demonstrated that human masters are exceptionally skilled in reconstructing the board positions after having looked at them for a few seconds (de Groot, 1965; Chase and Simon, 1973a, b; Simon and Chase, 1973). The results are clear: masters are able to reconstruct board positions with ease, while beginners have difficulty performing the task and do not achieve similar levels of performance. A subsequent experiment, however, dismantles the overtly simple hypothesis that such performance may be due to some higher faculty inaccessible to ‘weaker players’ (such as greater memory capacity). When asked to reconstruct *random* board positions (i.e., ones that do not arise naturally in the course of a game), the experts’s advantage rapidly drops, as masters are also prone to the limits of short-term memory (STM, Miller, 1956). Nonetheless, some skill effect remains, as the chunking theory would predict (see for instance Gobet and Simon, 1996). It is thus believed that humans perceive the board in terms of subpatterns that combine to make ‘dynamic complexes’, which entail the abstract relationships contained in each position

(de Groot, 1965; Chase and Simon, 1973a, b; Simon and Chase, 1973). It is widely believed that the encoding advantage of experts rests on their chess experience, as opposed to a greater perceptual or memory capability (Reingold et al., 2001b).

This is a major problem: to model the perception of a chess position as a set of dynamic complexes. Master players accumulate a great number of chunks, and over years of training they create a virtual ‘encyclopedia’ of these sub-patterns. Each of these chunks is treated like a single entity, like an object that holds abstract relations to other chunks identified on the board, to previous chunks it may have developed from, to forthcoming chunks it may evolve into, and also to estimates of the desirability of having such a chunk in a board position. What masters do over the years is to accumulate such knowledge. (Most modern cognitive theories are thus based on structures such as *chunks*, and *templates*, and some have argued that a higher level of description, based on structures known as *prototypes*, should be more representative of the underlying psychological processes – see for instance Gobet (1998), Gobet and Jackson (2002), Gobet et al. (2001), Gobet and Simon (1998, 2000).)

The effect perceived in these experiments may be intuitively understood in an area where we all are masters: the vocabulary of words we have acquired over the years. It is more natural to remember the information encoded in the term “psychology” than to remember the information encoded in a random sequence of the very same letters: “yyosglphco”. It seems obvious that one would not be able to remember accurately the latter with the same effort as the former after a rapid glance. This hidden “language of chess”, gradually acquired over the course of a player’s life, enables them to distinguish certain pressures on the board, which are prioritized over other pressures, and permits skilled players to see the dangers and opportunities overlooked by those who are less skilled. This ‘language’ of chunks denotes threats, blocks, defenses, forks, and a multitude of more subtle and complex combinations. It is an outstanding challenge for cognitive science to identify precisely and map such a ‘language’ (Linhares, 2002). As an optimistic note, estimations of its size are not notoriously high, as it has been proposed that the number of chunks probably is of the order of 50,000 chunks (Simon and Shaeffer, 1992) to 100,000 chunks and templates (Gobet and Simon, 1996).

We thus have an additional set of architectural principles:

**Architectural Principle 3.** *An intuitive program should be able to have familiarity with real board positions, while having difficulty interpreting implausible board configurations.*

This architectural principle would obviously correspond to human faculties. Since board interpretation is made in terms of previously acquired chunks, it demands the system to have a virtual “encyclopedia” of chunks in LTM and to keep acquiring new chunks over the course of new games.



Implausible board configurations will obviously not have the acquired chunks.

But even if masters do have 50,000 chunk items in long-term memory (LTM), they still remain limited in their handling of STM (Miller, 1956). It seems that novices and grandmasters are bounded in the number of chunks they can handle at any particular time; thus it seems that it is the chunk structure that can account for the increase in information (Chase and Simon, 1973a, b). The experiments clearly show that masters do not have greater memory skill, as stressed by their difficulty in reconstructing random (meaningless) positions. And since the results suggest that the number of chunks handled in STM are kept within the memory span proposed by Miller (1956), we have:

**Architectural Principle 4.** *An intuitive program should manipulate a relatively small number of chunks in short-term memory, despite having a relatively large encyclopedia of chunks stored in long-term memory.*

Let us now consider the problem of the coordination of the information acquisition. We may start with the Deep Blue example (Campbell et al., 2002): despite being labeled as a parallel and distributed system, Deep Blue is a top-down, centrally coordinated, search process. But is this the manner in which masters acquire information? It seems that the parallel and distributed perception process carried out by human masters exhibits instead a highly granular and fragmented mix of top-down and bottom-up processing. Let us start with the bottom up processes.

**Architectural Principle 5.** *An intuitive program should exhibit a bottom-up parallel processing, when it is presented with a board position.*

When facing a new position, it cannot be expected *a priori* that certain pieces (with the notable exception of the kings) or relations between pieces will be present. There can be few expectations *a priori*. It is thus crucial to acquire expectations during the perception of a chessboard. This process seems to be parallel, in that many pieces and relations are perceived simultaneously. Since there are no expectations, it starts as a purely data-driven process, and the system must work towards *knowing what to expect*. This is why at start we should have bottom-up processes operating: because the data drives the process. Robust psychological experiments have documented a perceptual interference effect, which “suggests automatic and parallel encoding procedures for chess relations in experts” (see Reingold et al., 2001a).

However, after the initial chunks have been identified, some expectations start to form, and these expectations deeply influence further perceptual processing. Expectations, ideas, hypotheses, and theories start to drive the process. Let us imagine that a first chunk is identified. What is the meaning of this particular chunk in this particular position? Is it an attack, or a defense? Which pieces are threatened and which pieces are secured? Is it desirable to

preserve the structure as it stands? What is likely to happen in the coming moves? Much information of this kind should be closely associated with the chunk in the LTM networks, and the identification of the chunk immediately propagates activation to such related nodes. As these expectations form, they trigger additional subcognitive urges to the perceptual process in order to further probe the position and clarify which details must receive immediate attention. These *expectation-based pressures* form a top-down – theory driven – part of the information processing.

This capability to fluidly mix bottom-up and top-down processes during information acquisition and description is precisely what enables attention (as measured by eye saccades) to be concentrated in the important areas of the board, instead of randomly switching from piece to piece, or of bureaucratically performing the very same tests irregardless of board position, *à la* Deep Blue. Moreover, the lack of expectations is precisely why expert players cannot reconstruct random boards. Since they simply happen to lack the chunks required to trigger the top-down processes, they are unable to create a set of top-down expectations to connect the chunks into a coherent and meaningful whole.

Cognitive psychology studies present us plenty of evidence that top-down pressures may influence and alter previously created representations. Chi (1978) points out for instance that “overlapping chunks share similar piece combinations”, which brings the obvious problem of identifying which chunk is the best alternative whenever two or more chunks overlap. Lories (1984, 1987) argues for a process where players “re-categorize chunks in order to achieve a global characterization of the position”. That is, a chunk previously identified by a data-driven, bottom-up process is later ‘destroyed’ and re-categorized in STM by an expectation-driven, top-down process, in order to create a coherent description of the position. Finally, experiments have demonstrated that “conceptual descriptions speed up recognition and interpretation of the board”, a result which has been used to support the argument for prototype theory (Gobet, 1998). This naturally leads to the following:

**Architectural Principle 6.** *An intuitive program should exhibit top-down processes, triggered by acquired expectations, which should enable the program to eventually create a coherent and meaningful description of the board.*

To understand the next principle, a final quote is of interest: “According to de Groot, chess masters do not encode the position as isolated pieces, but as large, mostly dynamic ‘complexes’. These complexes are generally made of pieces but may sometimes incorporate some empty squares that play an important role in the position” (Gobet, 1998, p. 127). These dynamic complexes are the result of the perception process, and, most interestingly, they should eventually incorporate empty spaces. This incorporation of empty spaces also has to be explained and modeled in any proposal for an intuitive chess machine.

**Architectural Principle 7.** *An intuitive program should construct dynamic complexes that contain pieces, relations between pieces, and eventually include empty squares of the board.*

At this point we may safely conclude (from the literature review) that perception seems to be the core function associated with human chess intelligence. This is in stark contrast to the vast majority of computational models supposedly based on ‘reason’ (i.e., implemented by inference search engines). So what kinds of structures would a chess program need in order to exhibit human-like intuitive perception? We can infer from the above mentioned experiments at least the following: (i) *chunks and subpatterns* (or a competing theory, such as that based on the prototypes notion, but still dealing with discrete structures); (ii) a STM, where a limited number of chunks are handled; (iii) a LTM – where a significantly larger number of chunks are stored; (iv) there should be a measure of ‘confusion’, of ‘disorder’, to reflect for instance the low performance of masters in remembering random positions; and finally (v) the core of the process should be based on perceptual processing – where ‘remembering’ and ‘analogy-making’ play a role in structuring each position as a unified dynamic complex. In the next section, we analyze a new cognitive architecture, evaluating how it may come to be adapted in order to develop an intuitive chess architecture.

### 3. The FARG Architecture for Pattern Conception

#### 3.1. PROBLEM DOMAIN

In Section 2 we have seen an analogy between the acquisition of chess chunks and the acquisition of a word vocabulary, presupposing that similar cognitive operations are carried out in both domains. In this section we extend this analogy and discuss a system devised to model the subcognitive operations underlying the human perception of sets of letter strings. At first the similarity with chess perception may not seem obvious. Consider the following, seemingly trivial, analogy problem:  $abc \rightarrow abd:ijk \rightarrow ?$ , that is, if the letter string “abc” changes to the letter string “abd”, how would the letter string “ijk” change “in the very same way”? This is the domain of the Copycat project (Mitchell, 1993; Marshall, 1999), and before we attempt a full description of the system, let us discuss in more detail some of the underlying intricacies. Most people will come up in this case with a rule of transformation that looks like: “Replace the rightmost letter by its successor in the alphabet”, the application of which would lead to  $ijl$ . This is a simple and straightforward example. But other examples bring us the full subtlety of this domain. The reader unfamiliar with the Copycat project is invited to consider the following problems:  $abc \rightarrow abd: ijkkkk \rightarrow ?$ ,  $abc \rightarrow abd: xyz \rightarrow ?$ ,

$abc \rightarrow abd$ :  $mrrkkk \rightarrow ?$ , among others (Mitchell, 2003) to acquire a sense of the myriad of subtle intuitions involved in solving these problems.

To solve this type of problem, one could come up with a scheme where the computer must first find a representation that models the change and then apply that change to the new string. This natural sequence of operations is not possible, however, because the transformation rule representing the change itself must bend to contextual cues and adapt to the particularities of the letter strings. For example, in the problem  $abc \rightarrow abd$ :  $xyz \rightarrow ?$ , the system may at first find a rule like “change rightmost letter to its successor in the alphabet”. However, this explicit rule cannot be carried out in this case, simply because  $z$  has no successor. This leads to an impasse, out of which the only alternative by the system is to use a flexible, context-sensitive, representational system.

The reader may have noticed that this cognitive processing bears some similarities to the process of chess perception. Perception obviously plays a significant role in letter-string analogies, as it is necessary to connect a set of individual units – in this case, letters – into a meaningful interpretation that stresses the underlying pressures of the analogy. In chess it is also necessary to connect disparate pieces into a meaningful description stressing the position’s pressures. But the most striking similarities with chess perception (in what concerns bounded rationality) seems to be the absolute lack of a single objectively correct answer. Instead, we have just an intuitive subjective feeling, given by the great number of simultaneous pressures arising in each problem.

In the previous section we have made reference to some studies considering multiple, incompatible chunks that emerge in chess positions. In letter strings this same problem appears. Consider for instance the following problem: If  $aabc \rightarrow aabd$ :  $ijkk \rightarrow ?$

One may chunk the initial strings as  $(a)(abc)$  and  $(a)(abd)$  and find a ‘corresponding’ chunk  $(ijk)(k)$ , which could lead to the following transformation rule: “change the last letter of the *increasing sequence* to its successor in the alphabet”. This interpretation would lead to the answer  $ijlk$ .

Alternatively, one may chunk the initial strings as  $(aa)(b)(c)$  and  $(aa)(b)(d)$  and find a counterpart string with the chunking  $(i)(j)(kk)$ . In this case, the mapping can even become inverted: The first letter-group ( $aa$ ) maps to the last letter-group ( $kk$ ), and this will also invert the other mappings, leading to  $(b)$  mapping to  $(j)$  and  $(c)$  mapping to  $(i)$ . Because this viewpoint substantially stresses the concept ‘*opposite*’, Copycat (detailed below) is able to create the transformation rule “change the first letter to its *predecessor* in the alphabet”, leading to the solution  $hjkk$ , which preserves symmetry between sizes of letter-groups and between successorship and predecessorship relations.

Other potential transformation rules could lead, in this problem, to  $ijkl$  (change the last letter to its successor in the alphabet),  $ijll$  (change the last

group of letters to its successor in the alphabet), or jjkk (change the first letter to its successor in the alphabet). This problem of many incompatible (and overlapping) chunkings is highly important here. The specific chunking of a problem is directly linked to its solution, because *chunks stress what is important on the underlying relations*. This is very much like chess perception, where chunks enhance perception of good moves and inhibit perception of irrelevant (and of bad) moves. It is a very context sensitive process. While in a hypothetical chess position a relation *X* may be seen to have greater relevance than a relation *Y*, there may still exist many different varieties of positions in which relation *Y* is more relevant than *X*. So a chess system, like the Copycat system, needs to (i) be able to conceive the multiplicity of relations from the underlying data, and (ii) evaluate which is more relevant to each specific case.

### 3.2. COPYCAT—A THEORY OF ACTIVE SYMBOLS

How does the Copycat system work? Before reviewing its underlying parts, let us bear in mind one of its principal philosophical points. Copycat is not intended solely as a letter-string analogy program. The intention of the project is the test of a theory, a theory of ‘*statistically emergent active symbols*’ (Hofstadter, 1979, 1985) which is diametrically opposite to the “*symbol system hypothesis*” (Newell, 1980; Simon, 1980). The major idea of active symbols is that instead of being tokens passively manipulated by programs, active symbols emerge from high numbers of interdependent subcognitive processes, which swarm over the system and drive its processing by triggering a complex ‘chain reaction of concepts’. The system is termed ‘subsymbolic’ because these processes are intended to correspond to subliminal human information processes of few milliseconds, such as a subtle activation of a concept (i.e., priming), or an unconscious urge to look for a particular object. So the models are of collective (or emergent) computation, where a multitude of local processes gradually build a context-sensitive representation of the problem. These symbols are *active* because *they* drive processing, leading a chain reaction of activation spreading, in which active concepts continuously trigger related concepts, and STM structures are construed to represent the symbol. In this philosophical view a token does not have any associated meaning, while a meaningful representation, a symbol, emerges from an interlocked interpretation of many subcognitive pressing urges.

This cognitively plausible architecture has been applied to numerous domains (see for instance Mitchell and Hofstadter, 1990; French, 1992; Mitchell, 1993; McGraw, 1995; Marshall, 1999; Rehling, 2001). It has five principal components:

- (i) *A workspace that interacts with external memory* – This is the working STM of the model. The workspace is where the representations are constructed, with innumerable pressing urges waiting for attention and their corresponding impulsive processes swarming over the representation, independently perceiving and creating many types of subpatterns. Common examples of such subpatterns are bonds between letters – such as *group* bonds between ‘a\*a’ or *successor* bonds between ‘a\*b’ – or relations between objects, awareness of abstract roles played by objects, and so on.
- (ii) *Pressing urges and impulsive processes* – The computational processes constructing the representations on STM are subcognitive impulsive processes named *codelets*. The system perceives a great number of subtle pressures that immediately invoke *subcognitive urges* to handle them. These urges will eventually become impulsive processes. Some of these impulsive processes may look for particular objects, some may look for particular relations between objects and create bonds between them, some may group objects into chunks, or associate descriptions to objects, etc. The collective computation of these impulsive processes, at any given time, stands for the working memory of the model. These processes can be described as impulsive for a number of reasons: first of all, they are *involuntary*, as there is no conscious decision required for their triggering. (As Daniel Dennett once put it, if I ask you “not to think of an elephant”, it is too late, you already have done so, in an involuntary way.) They are also *automatic*, as there is no need for conscious decisions to be taken in their internal processing; they simply know how to do their job without asking for help. They are *fast*, with only a few operations carried out. They accomplish direct connections between their micro-perceptions and their micro-actions. Processing is also *granular* and *fragmented* – as opposed to a linearly structured sequence of operations that cannot be interrupted (Linhares, 2004). Finally, they are *functional*, associated with a subpattern, and operate on a subsymbolic level (but not restricted to the manipulation of internal numerical parameters as opposed to most connectionist systems).
- (iii) *List of parallel priorities* – Each impulsive process executes a local, incremental change to the emerging representation, but the philosophy of the system is that all pressing urges are perceived simultaneously, in parallel. So there is at any point in time a list of subcognitive urges ready to execute, fighting for the attention of the system and waiting probabilistically to fire as an impulsive process. This list of parallel priorities is called the *codrack* in Copycat.
- (iv) *A semantic associative network undergoing constant flux* – The system has very limited basic knowledge: it knows the 26 letters of the alphabet, and the immediate successorship relations entailed (it does not, for instance, know that the shapes of lowercase letters p, b, q

bear some resemblance). The LTM of the system is embedded into a network of nodes representing concepts with links between nodes associating related concepts. This network is a crucial part for the formation of the chain reaction of conceptual activation: any specific concept, when activated, propagates activation to its related concepts, which will in turn launch top-down expectation-driven urges to look for those related concepts. This mode of computation not only enforces a context-sensitive search but also is the basis of the chain reaction of activation spreading – hence the term ‘active symbols’. This network is called the slipnet in Copycat. One of the most original features of the slipnet is the ability to “slip one concept into another”, in which analogies between concepts are made (for details see Mitchell, 1993; Hofstadter, 1995).

- (v) *A temperature measure* – It should be obvious that the system does not zoom in immediately and directly into a faultless representation. The process of representation construction is gradual and tentative, with numerous impulsive processes competing with each other. At the start, the system has no expectations of the content of letter strings, so it slowly wanders through many possibilities before converging on a specific interpretation, through a process called the *parallel terraced scan* (Hofstadter, 1995). Embedded within it is the control parameter of *temperature* that is similar in some aspects to that found in simulated annealing (Hofstadter, 1995; Cagan and Kotovsky, 1997). The temperature measures the global amount of disorder and misunderstanding contained in the situation. So at the beginning of the process, when no relevant information has been gathered, the temperature will be high, but it will gradually decrease as intricate relationships are perceived, first concepts are activated, the abstract roles played by letters and chunks are found, and meaning starts to emerge. Though other authors have proposed a relationship between temperature and understanding (Cagan and Kotovsky, 1997), there is still a crucial difference here (see Hofstadter, 1985, 1995): unlike the simulated annealing process that has a forcedly monotonically decreasing temperature schedule, the construction of a representation for these letter strings does not seem to get monotonically more complete as time flows. As in the ‘ $abc \rightarrow abd : xyz \rightarrow ?$ ’ problem, there are many instants when roadblocks are reached, when snags appear, and incompatible structures arise. At these moments, complexity (and entropy and confusion) grows, and so the temperature decrease is not monotonic (once again these non-monotonic temperature schedules are analogous to improvements on the simulated annealing algorithm used for optimization – Linhares et al. (1998, 1999) and Möbius et al. (1997)). Finally, temperature does not act as a control parameter dictated by the user, that is, forced to go either down or up,

but as a feedback mechanism to the system, which may reorganize itself, accepting or rejecting changes as temperature allows. As pressing urges are perceived, their corresponding impulses eventually propose changes in working memory, to construct or to destruct structures. How do these proposed changes get accepted? Temperature guides the process, very much like simulated annealing. At start it is high and the vast majority of proposed structures are built, but as it decreases it becomes increasingly more important for a proposed change to be compatible with the existing interpretation. And the system may thus focus on developing a particular viewpoint.

### 3.3. AN EXAMPLE RUN

Let us consider an example run of the Copycat system and look at some specific steps in its processing of the problem “ $abc \rightarrow abd : iijkk \rightarrow ?$ ”.

Figure 4 presents the working memory (workspace) after 110 impulsive processes (codelets) have been executed. The system at this point has not perceived much structure. It has perceived each individual letter, it has mapped the letters c and d between the original and target strings, and it has perceived some initial bonds between neighboring letters. Some of these bonds are sameness bonds (such as  $i*i$ ), some are successorship bonds (such as  $i*j$ ), and some are predecessorship bonds (such as  $b*c$ ). Note that there is confusion between the competing views of successorship and predecessorship relations in the string ‘abc’. These incompatible interpretations will occasionally compete against each other. The system is also mapping the leftmost letter a to the leftmost letter i.

Notice that a first chunk has been created in the group ‘jj’. Now this chunk is an individual object on its own, capable of bonding with (and relating to) other objects. Notice also that the system has not yet perceived – and built the corresponding bond between – the two k’s in succession. So perception in Copycat is granular, fragmented over large numbers of micro-perceptions (Figure 5).

After an additional 150 codelets have been executed (Figure 5), more structure is built: we now have three group chunks perceived; and there is also less confusion in the ‘abc’, as a ‘staircase’ relation is perceived: that is, the system now perceives ‘abc’ as a successorship group, another chunked object. Finally, an initial translation rule appears: replace letter category of rightmost letter by successor. If the system were to stop processing at this stage it would apply this rule rather crudely and obtain the answer  $iijjkl$ . Note that temperature is dropping as more structure is created.



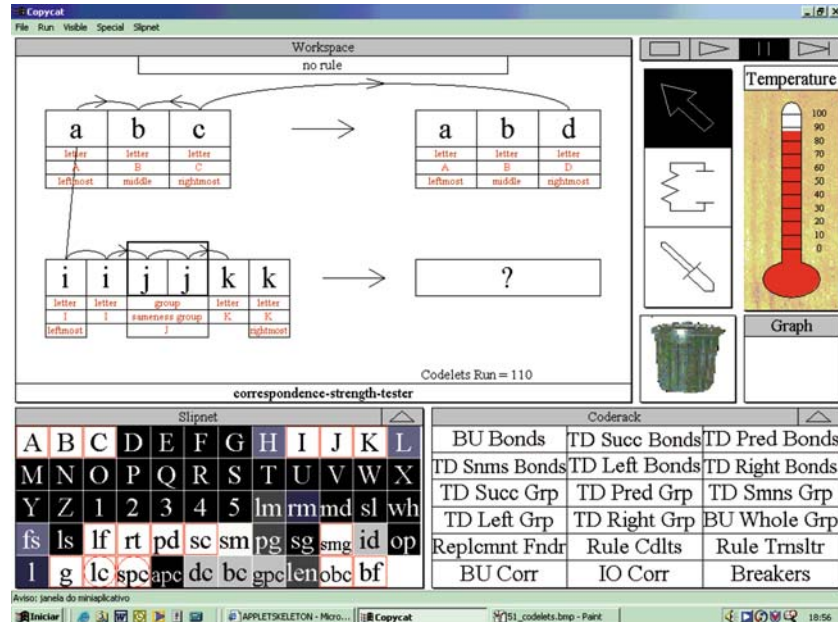


Figure 4. Copycat after 110 codelets have executed. This implementation was carried out by Scott Bolland from the University of Queensland, Australia (2003, available online).

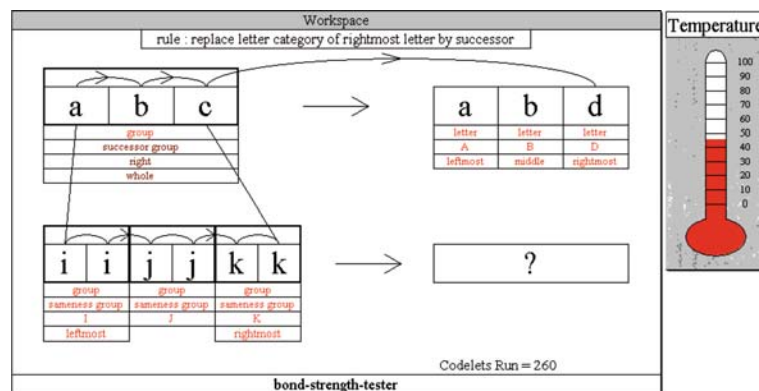


Figure 5. Copycat's working memory after the execution of 260 codelets.

Let us slow down our overview a little bit and return to Figure 6 after only 20 impulsive processes have run, to illustrate an important phenomenon: though *c* now will map to the *group* *kk*, which is an important discovery, the global temperature will still be higher than that of the previous point (Figure 5). This occurs because there is much added confusion arising from the *predecessorship* bond which was found between chunks *ii* and *jj*, which does

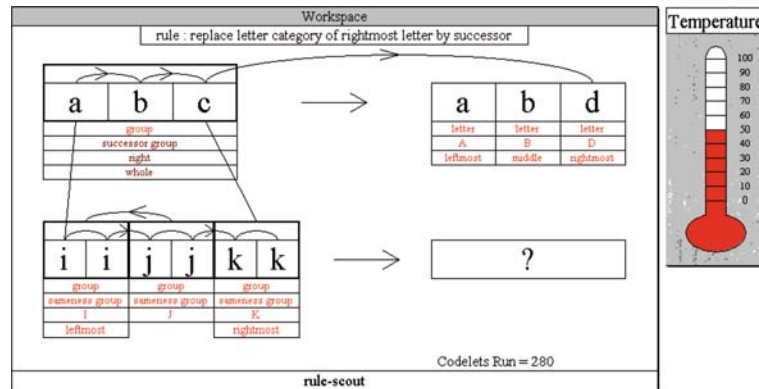


Figure 6. Copycat's working memory after the execution of 280 codelets.

not seem to fit well with all those successorship relations already perceived and with the high activation of the *successorship* concept. So temperature does not always drop monotonically (Figure 7).

On the next step we can perceive two important changes: first, the system perceives some successorship relations between the groups ii and jj and between the groups jj and kk, but these relations are perceived in isolation from each other. Another important discovery is that jj is interpreted as being in 'the middle of' iijkk, which will eventually lead to its mapping to the letter b in the original string (Figure 8).

The system finally perceives that the successorship relations between the ii, jj, and kk groups are not isolated and creates a single successorship group encompassing these three sameness groups. Thus two successor groups are perceived on the workspace, and a mapping between them is built. However, the letter a still maps to the letter i, instead of to the group ii, and the letter c still maps to the letter k, instead of to the group kk.

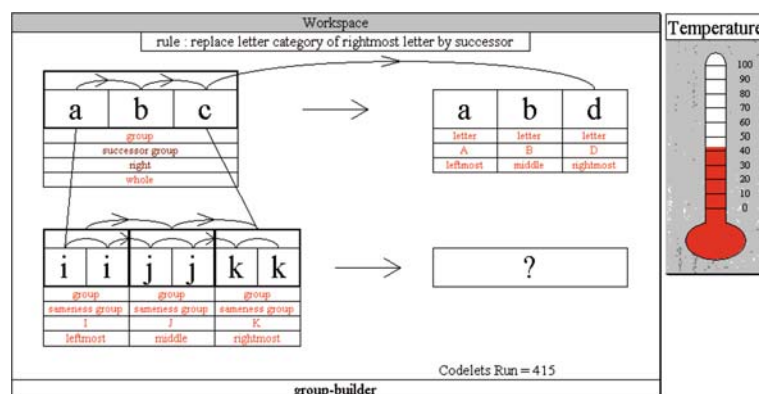


Figure 7. Copycat's working memory the after execution of 415 codelets.

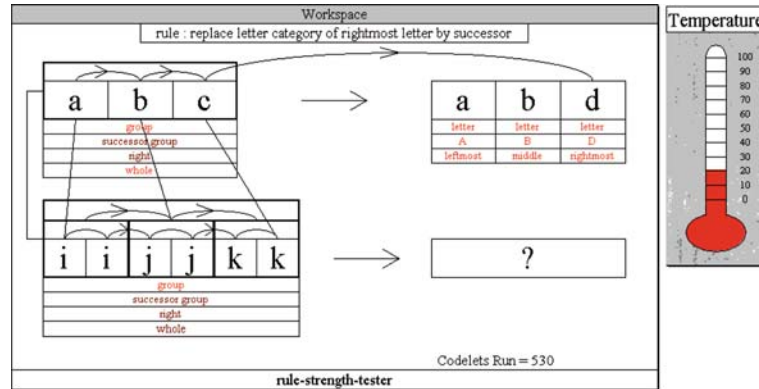


Figure 8. Copycat's working memory after the execution of 530 codelets.

From this stage it still remains for the letter a to map to the group ii and for the letter c to map to group kk, which will lead naturally to the translated rule “replace letter category of rightmost group to successor”, illustrating the slipping of the concept *letter* to the concept *group* (Figure 9).

After 695 impulsive processes (codelets), the system reaches the answer iijjll. The workspace may seem very clean and symmetric, but it has evolved from a great deal of disorder and from many microscopic ‘battles’ between incompatible interpretations.

The most important concepts activated in this example were *group* and *successor group*. Once some *sameness* bonds were constructed, they rapidly activated the concept *sameness group* which reinforced the search to find sameness groups, such as kk. Once the initial *successorship* bonds were created, the activation of the corresponding concept rapidly enabled the system to find other instances of successorship relations (between, for instance, the sameness groups jj and kk). Different problems would activate other sets

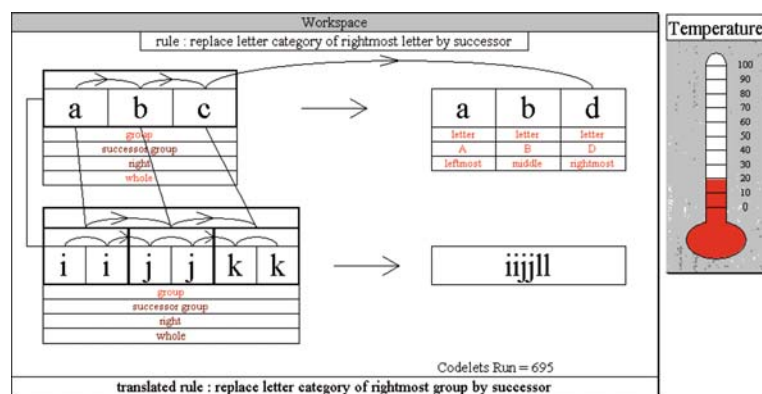


Figure 9. Final solution obtained after the execution of 695 codelets.

of concepts. For example, ‘ $abc \rightarrow abd: xyz \rightarrow ?$ ’ would probably activate the concept *opposite*. And ‘ $abc \rightarrow abd: mrrjjj \rightarrow ?$ ’ would probably activate the concept *length* (Mitchell, 1993). This rapid activation of concepts (and their top-down urges), with the associated propagation of activation to related concepts, creates a chain reaction of impulsive cognition, and is the key to active symbols theory. The reader is referred to Mitchell (1993) and to Marshall (1999) to have an idea of how the answers provided by Copycat resemble human intuition.

We may safely conclude at this point that there are many similarities between Copycat and the chess perception process, including: (i) an iterative ‘locking in’ process into a representation; (ii) smaller units bond and combine to form higher level, meaningfully coherent structures; (iii) the perception process is fragmented, granular, with great levels of confusion and entropy at start, but as time progresses it is able to gradually converge into a context-sensitive representation; (iv) there is a high interaction between an external memory, a limited size STM, and a LTM; and (v) this interaction is done simultaneously by bottom-up and top-down processes. This is the first part of our thesis: a Copycat-like chess system would exhibit architectural principles 4, 5, and 6 (these statements will be qualified in Section 5). It remains to be shown that principles 1, 2, 3 and 7 could also be followed by such a system. The next section will propose a specific chess representation in order to meet those principles.

In the next section an intuitive computer chess program based on these ideas is postulated. We may then come to see how the proposed architectural principles are satisfied by this model.

#### 4. Postulating a Cognitive Architecture for an Intuitive Chess Machine

How could the core principles of the Copycat cognitive architecture be applied to chess? Moreover, besides the general aspects of Copycat, what is needed for a system to exhibit ‘intuitive strategy’? Let us discuss the dynamics of the perception process of a position and its underlying chunk representation in a postulated Copycat-like chess system.

##### 4.1. THE LOCKING-IN PROCESS AND TEMPERATURE-BASED FEEDBACK

We are interested in the sub-conscious information processing that creates the intuition of a chess expert; a *locking in* process which at start has (from external memory) just the pieces present on the board and their specific squares. To conceive the board as a meaningful structure, the system then has to start gathering relevant data about the deeply hidden relationships of that position.

How does the process work? The main idea is for the system to gradually focus on a specific board structure, which is meaningful to the situation at hand, considers the complexity of the board, and clearly indicates the best movement possibilities, without performing an exhaustive tree search (see French, 1999). This is similar to the ‘locking-in’ process of Copycat, and there is an analogy to the perception of letters, syllables, words, and phrases, where meaningless units also ‘bond’ to form higher-level meaningful entities: how does a perception process transform a particular set of individual pieces (letters) on a coherent, meaningful, connected, complex dynamic of words and phrases? How does a perception process transform a particular arrangement of chess pieces into a unified, meaningful, coherent whole? These issues are clearly related, and are far from trivial.

As in Copycat, in our postulated system temperature is high at start, as there is a lot of entropy, and the system evaluates multiple potential possibilities (some of which are very unlikely), gradually gathering basic information about the problem. This initial information will evoke chunks (which should have corresponding nodes on a slipnet-like structure), previously acquired over earlier games. These evoked chunks act as highly active concepts, influencing further processing by triggering subcognitive processes (with high probability of execution) that either help confirm or negate that the current board position has an actual instance of the chunk in question. If these subcognitive tests pass, an instance of the chunk has been identified, its corresponding structures are built in working memory, and the temperature may drop accordingly. But if no test happens to pass, the system still remains able to wander over other possibilities, and temperature can even rise. Temperature thus sometimes drops quite rapidly (as chunk information is tested and confirmed), but not always – as the parameter measures the overall ‘understanding’ of the present situation.

As mentioned, chunks should be stored on a LTM chunk network, and their information should be accessed in a fragmented manner: a bit of data triggers activation of a possible chunk, which ‘tests’ a hypothesis by placing top-down impulsive processes looking for additional data which would be consistent with such a chunk (and that would reinforce the original hypothesis). In parallel, all the while, other impulsive processes carry on spreading activation to related chunks, as in Copycat. At times highly active chunks can also launch more sophisticated distance estimations between highly salient objects. Let us look closer at these differing levels of distance estimations.

#### 4.2. A DISTANCE-BASED REPRESENTATION

What may activate a chunk? Chase and Simon (1973b), when investigating a memory for sequences of chess moves, hypothesized attack and defense

relations in chunks. Let us generalize this idea by considering the concept of *distance between two pieces* that is, how many movements lay on the trajectory from piece X to piece Y? What is the minimum number of movements for the white knight to attack the black queen? This *distance metric* provides direct information on the pieces that are 'related' (in the sense of Simon and Chase, 1973): their distance is 1. Moreover, it also gives a more precise measure of the pressures involved in the chess board, by presenting those pieces that may attack (defend) each other in 2 moves, or 3 moves, or 4 moves, leading all the way to those which do not present any threat (or chance of defense) at all. So this metric captures one concrete aspect of board positions.

Figure 10 presents an example. According to traditional chunking theory (Chase and Simon, 1973a, b; Gobet, 1998), chunks are mostly based on proximity, color, type of piece, and occasionally on immediate attack and defense relations. In contrast, according to our standpoint, the basis of *semantics* should play a crucial part in the constitution of chunks. For example, in Figure 10, let us consider the white queen and the black king. They have different colors, they do not have a relation of high proximity, and there is no immediate attack relation (as neither is directly reachable from the other). So according to this traditional theory, a chunk containing these pieces could only be formed after a very large number of smaller chunks had been collected. In contrast, the proposal that semantics should play a definite part in the formation of chunks would build a chunk for these relations. Since the white queen is two moves away from the black king, and, moreover, one of her trajectories is strongly defended by the white rook, leading to a



Figure 10. White to move: mate in one. Distances form spatial graph representations manipulated in working memory. It may be of interest to investigate the role that these distances may play in forming chunks.

checkmate for white, this meaningful relation, obtained through distance analysis, seems crucial in forming a relevant chunk.

It is of course possible to test this hypothesis, by measuring the extent to which expert players would misplace the positions of the white king, the white rook, and the black king. This representational scheme would seem to be in line with the view that “chess players characterize the board spatially” (de Groot, 1965, p. 7), but it goes beyond considering the board visually as a simple Euclidean space, since the representation deals with the idiosyncrasies of each piece type’s movement capabilities.

Many abstract relations are described in such distance graphs. As each piece threatens to occupy some board squares in the following several moves, it is precisely which threats are perceived, which relations are ignored, and how the pieces interact as a *whole* that differentiates skilled from unskilled human players. Since there are always multiple candidate conceptions of how the pieces combine to defend and attack the opposing camp, it seems that chess perception is not an objective, one-to-one (position-to-representation) mapping, but a compromise, in which some features are identified, while others are ignored. In fact, *since so much structure is simply not perceived*, maybe the term perception itself is somewhat misleading. In previous works, we have referred to this as multiperception, and lately to a process of *pattern conception*, with the term *conception*’s natural meaning of a *conceived alternative* – *one specific conception* – *in detriment of many potential alternative conceptions* (Linhares, 2000). Saariluoma (1992, 2001) refers to a similar concept as *apperception* (see also Saariluoma and Hohlfeld, 1994).

Another aspect captured by this representation is the view of *analogy between chunks and analogy between chess positions*. Once the distance relations are mapped, the identity of each particular piece gradually loses importance: If we look at the distance graph and discard the identity of the pieces, we can obtain a set of positions that are analogous to this one in an abstract and profound sense: positions in which the “very same” pressures emanate from a radically distinct variety of piece combinations. Should we imagine a whole set of transformations of this type, which change the actual board while still preserving the large scale structure of the distance graph, then after some steps we should have a board position that looks very different from the original one but which still remains in a deep sense analogous to it. This raises intriguing possibilities for new psychological experiments concerning “analogies between positions”: if chess representation in experts is encoded in a structure such as this, it can clearly be subject to empirical enquiry.

But even if it is plausible that distance metrics play a significant role in experts’ coding of positions, we should not prematurely presuppose that such metrics are simple to devise. In fact, there are many possible gradations of such functions, varying in terms of computation speed and precision,

from an extremely fast lookup table to a full depth intractable combinatorial search. It is possible that a significant part (or even most) of the intelligence of experts comes from identifying when to select each specific gradation to use in each particular case. Let us consider 5 levels of distance metric quality:

- (i) *Heuristic glance*. This level is the fastest possible distance evaluation. It can be designed by implementing a set of lookup tables with the following information: the type of piece and its square, and the destination square; the lookup table would provide a first estimate of how many movements that piece remains from the destination. Notice the obvious limitations of this estimation, as it does not include blocking pieces, intermediate squares under immediate threat (which probably could not be used), and, most importantly, potential opponent responses. Note that this  $O(1)$  computation can be performed for all piece pairs in a chess position in fractions of a second, giving an initial estimate of both attack and defense distances involved, and *stressing the* (direct, 1-move) *immediate attacks and defenses* involved, which are usually the relations that demand immediate attention in a position.
- (ii) *Estimate considering blocks*. Because not all 1-move attacks (or defenses) pointed out by a heuristic glance can actually be executed in a single move, the next step up would be a fast algorithm to compute the minimum path to the distance, *while considering blocks*, that is, occupied squares. This would demand the use of a backtracking technique, take much more time than the heuristic glance, but it could still be done in a small number of operations, since the number of backtracking operations is bounded by the number of pieces present on the board.
- (iii) *Estimate considering blocks and threats*. This level of distance metric quality would function exactly as the ‘estimate considering blocks’, except by its consideration of potential threats by the opponent. This would require that the distance from the opponent’s pieces towards each position on the path be calculated, which would obviously add to processing time.
- (iv) *Estimate considering multiple levels of response*. This estimation would be considerably more complex, by employing an anytime tree searching algorithm that may consider multiple levels of opponent responses. It could take large amounts of time, but it could, for some cases, and for highly salient relations, at strategic points in the game, compute the exact minimum distance between a piece and its destination.
- (v) *Full combinatorial search*. For the sake of completeness, an extreme possibility would be the application of a full scale combinatorial search of all potential movements and all opponent responses, which would place enormous demands on computational time, but could



eventually obtain winning movement combinations, especially on endgames.

The cognitive chess model proposed here should operate with a mix of these levels of distance metric quality. It seems plausible that analogous mechanisms of differing levels of distance evaluation exist in the chess processing of experts. Four key reasons support this view:

First, as we have seen, psychological experiments (Chase and Simon, 1973b) have documented that chess players concentrate eye movements between related pieces; this corresponds to a *heuristic glance level of distance quality*, by merely perceiving that the distance between a piece and its destination equals 1. In order to perceive this, there must be eye saccades between the origin and the destination pieces. This is done instantly by human chess players, as Reingold et al. (2001a) put it, it is an “automatic and parallel” process; and this instantaneousness is also reflected on the metric.

However, in endgames with complicated positions, human chess players often reflect in a somewhat distinct manner, by carefully counting the piece distances and systematically considering potential opponent movements. A glimpse at Figure 3 shows the underlying logic, where the movements of the king “trying to catch the onrushing pawn” (de Groot, 1965) are carefully and deliberately counted, while other relations do not have such an explicit detail. At those moments, it seems that the subcognitive mechanisms more closely resemble those postulated as an ‘estimate considering multiple levels of response’, in which distance is carefully evaluated in a square-by-square, move-by-move basis. So it may be that humans employ a mix of those levels of quality of distance evaluations. This is a second reason for distance graphs. There is also evidence that human players make use of inter-piece distances also in findings about the time required to verify a check relation as a function of distance (Church and Church, 1983).

The third reason comes from a related feature of this representation: it enables a much more selective search than a classical tree model, because the focus here is to search movement combinations *starting from piece X and leading to piece Y* in order to measure the distance (obviously considering movement responses from the opponent). This is severely more restricted than iteratively searching each open movement possibility and each open opponent response, *ad nauseum*.

A final reason explains what some competing theories cannot: the presence of empty spaces in chunks and in representations – as had been proposed by Charness et al. (2001). If piece A is 2 moves away from piece B in a sophisticated evaluation level, and if this is an important relation in the context of a position, then there is an empty space between A and B which is

obviously relevant for representation of the ‘dynamic complex’. Once again, this can be subject to empirical psychological testing.

#### 4.3. A CHUNK NETWORK

Though the distance graph model seems to account for many characteristics of human chess cognition, it is, however, still restricted for our purposes. Chunks may be accessed and activated by a structure such as the distance graph, but it is clear that chunks should contain more information than their mere distance structure. Their access by the distance graph enables the immediate perception of forks, pins, blocks, attacks, skewers etc., but chunks should still have additional “knowledge” such as:

- (i) Links to other chunks of high co-incidence (sometimes through “higher level” chunks which expect to find specific combinations of these “subchunks”). These links enable activation to spread to them and to create expectations for the position. These same functions have been fulfilled by templates in (Gobet and Simon, 1996, Gobet 1997). Like the Copycat system, top down impulses ask whether or not these highly co-incident chunks are present in the configuration.
- (ii) An estimate of the salience of the chunk also analogous to the Copycat system.
- (iii) Links to potential upcoming positions, which can create a subtle ‘feeling’ of the upcoming events. These become important whenever many chunks conspire to reinforce a particular upcoming configuration. That is, because activation spreads only to a select subset of possible upcoming positions, the system is able to, rather like human players, ignore the vast majority of possibilities entailed by the combinatorial explosion. Mechanisms for learning also apply here: if the system creates expectations which are not subsequently met, it should be able in principle to automatically and incrementally update the LTM embedded in the chunk network in order to foresee a future similar case. (This obviously leads us to the challenging research problems of devising the appropriate mechanisms).
- (iv) An estimate of the overall “desirability” of the chunk; that is, a numerical estimate of whether or not a specific chunk is weakening a particular position (and thus needs to be changed), or, alternatively, if that chunk is actually expected to bring on some advantages.
- (v) An estimate of the “normalcy” of the chunk. Is it rare? Is it common? This information of frequency of occurrence should be important especially in games against experienced adversaries where one-of-a-kind piece combinations seem more likely to occur.

The principal role of chunks under this view is to highlight the abstract roles played by pieces and to activate suitable intentions.

#### 4.4. FROM CHUNKS TO ABSTRACT ROLES TO INTENTIONS

Beyond estimating distances and perceiving chunks, our postulated system also needs to perceive abstract roles in order to conceive lines of play. In order to do this, a new type of object is created in working memory: an *abstract role* structure. Associated with each piece (or chunk) is a set of data structures that intends to capture the abstract roles that the piece is playing on the position. For example, a piece can be a guardian of some other piece. Or it may be an attacker, or a blocker. The structure also intends to estimate the level of which each role is found. If a queen is a guardian of a pawn, it is generally a weak guardian, because in most circumstances it is not desirable to exchange the queen for the pawn. So roles can be of weak, strong, or potential guardians, attackers, and blockers. On a related issue, the *mobility* of each piece also should take part of its role. Mobility is computed gradually, and the territory around the piece is mapped in an iterative process where a piece gradually “floodfills” the next squares reachable. So an individual piece may at any point in time be playing multiple roles, as, for instance: “piece A is highly mobile, strongly defended by piece B, weak guardian of piece C, but strong guardian of piece D, and a strong attacker of piece E”. These data structures enable the system to conduct *a form of ‘abstract thought’ completely unattached to specific pieces, distances, and trajectories*. Hence it enables the system to respond to general abstract (and even vague) ideas, for example such as those implementable by the top-down impulse: “if a piece is guardian of two pieces in different regions, try to force it to compromise for the piece towards which it is the weakest guardian”. Thus, these abstract roles gradually hand over meaning to a position.

These abstract roles should be associated with such a set of top-down impulses that play parts of ‘tactics’. Saariluoma (1995) has classified 10 general tactics (see Table 1), which are referred to as *functional constraints*, and has claimed that this set of functional constraints can account for “the [...human...] calculation process in any protocol we have met so far”. Though it “may be possible to find some additional rules”, these would not change the principle of a meaningful content integration in the experts’ development of their intentions.

These tactics should be immediately activated after the perception of the correct abstract roles. In Copycat, perception of the letter ‘a’ evokes the concepts of ‘A-ness’, and of ‘first letter of the alphabet’, and (thus) indirectly, in a smaller scale of activation, the concept ‘1’. Perception of a single ‘a’ triggers a small chain reaction of conceptual activation. A similar phenomenon

Table 1. Functional constraints directing move generation (After Saariluoma, 1995)

<i>Transfer</i>	A piece is moved to get it to the path for some subsequent move
<i>Exchange</i>	An active piece is taken
<i>Blockade</i>	An active piece is prevented from achieving a key square by placing a piece between its original and destination square
<i>Escape</i>	The target piece is moved to another square
<i>Pin</i>	An active piece is prevented from moving, by placing a piece so that its movement is illegal (absolute pin) or would be too costly (relative pin)
<i>Unblockade</i>	A piece is moved to allow some other piece to make an active move
<i>Clearance</i>	An enemy piece supporting some key square is exchanged or forced to lose control over a key square
<i>Decoy</i>	A target piece is forced to move into an undesirable square
<i>Threat</i>	A piece is moved to achieve a goal in the next move
<i>Counter-action</i>	A move which is made to achieve some independent goal

of activation spreading seems needed to account for chess intuition. Note how each of the tactics classified by Saariluoma directly affects the distance network in its own specific way – the tactics are all directly related to some particular underlying distance structures. So some chess chunks, once identified, should immediately activate particular tactics, such as *decoy*, *pin*, or *threat*, and in our postulated system there should be a ‘network’ of these tactics (or in Saariluoma’s terminology, *functional constraints*), which are directly related to the abstract roles perceived. This fluid activation of such tactics, and how they should be applied in each case, also corresponds to Copycat’s fluid slipping of concepts (e.g., we have seen in Section 3 an example where the concept ‘letter’ is able to slip to the concept ‘letter group’).

The underlying logic of the system is this: multiple impulsive processes are executing simultaneously, each of them with a specific micro-perception and a specific micro-action. Some of these processes will be evaluating distances between pieces and distances from pieces to key squares. Now, some of these distances, once evaluated, will take part in the formation of chunks. And these chunks, once created, will trigger activation of the conceptual network that relates chunk structures, abstract roles, and tactics for play. Impulses that recognize an abstract role for a piece will immediately evoke urges for the particular tactics. The activation of such tactics, in turn, will once again create top-down pressing urges for additional (more detailed) distance estimations and chunk reconfigurations. For example, if a blockade tactic has been highly activated by a particular chunk, the system now has to estimate new distances from several pieces (from moderate to high mobility) to all the open squares that could potentially block the desired path, in order to find

the best possible implementation of the blockade. Thus there are rapid advances in this chain reaction of distance estimations, chunk perceptions, and tactic conceptions, to a point where the three developments are all proceeding simultaneously, completely controlled by temperature feedback. Emergent from these chaotic sets of estimated distances, subcognitive urges, mapped mobilities, subtle pressures, structured chunks, perceived interceptions, abstract roles, and active intentions, is *the meaning* of the situation at hand – that particular ‘*feeling*’ or ‘*sense of position*’ which has seemed so elusive to computational models to this date.

Up to this point, our discussion has been placed in rather abstract terms, so let us clarify the proposal by returning to Wilkins’s (1980) example and considering how an ‘intuitive strategy’ may emerge after some subcognitive impulsive processing.

#### 4.5. THE EMERGENCE OF AN ‘INTUITIVE STRATEGY’

Let us ponder how understanding of the pressures involved in the example of the introduction – such as (i) the blocked pawn structures, (ii) the interlocked black king and pawn at F6, (iii) and the sole safe passage through queen rook line, etc. – may emerge in the postulated Copycat-like chess processor.

Let us consider how the contents of working memory may look at some specific moments. The initial objects are the pieces and their positions, so at first pieces present on the board are gradually recognized, creating the corresponding discrete structures in STM, and triggering impulsive checks for their potential movements and threats. Figure 11(a) above represents this initial phase of processing, maybe just after the very first milliseconds, where some of the pieces have been randomly recognized and no distance relation has been built yet (pieces are displayed on their relative positions for illustrative purposes only). At this point, temperature is at its maximum – which means that the system is “open-minded”, capable of perceiving (i.e., interpreting some relations as) attacks, blocks, forks, skewers, etc. But temperature gradually falls as the initial relations are grasped and more information is acquired. Figure 11(b) displays some of these initial relations, as some preliminary distance estimations between random pairs of pieces start to form. These estimations are executed very quickly by “distance glance” impulses, which give only a first approximation of the distance for most pairs of pieces, but become exact numbers in the case of immediate attacks and defenses (that is, when piece X is at a distance of 1-move to piece Y, as stressed for some pairs of pieces in the figure). After these computed distances enter working memory as objects on their own, some ‘bonds between pieces’ start to form, as closely related pieces (those with smallest distances) are identified.

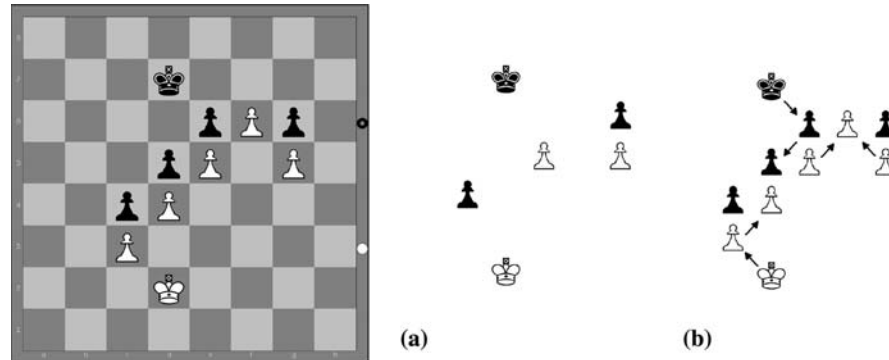


Figure 11. Working memory during initial stages of the perception process.

The salience of these bonds should be given by the value of each piece type and the underlying distance relation. If it is a low-salience pawn attacking a (valuable, high salience) queen, this is a relation that needs immediate attention (and very likely it should lead to the building of a chunk). If, however, it is a regular pawn chain or a valuable piece attacking a low-value one, then the salience of the bond should not be as high, and in some cases may not even lead to the construction of a chunk.

Once again, the reader may have noticed that this process of mapping the immediate 1-move relations corresponds to “eye saccades between related pieces” found over numerous psychological experiments. Note also that at this point we have not given recourse to chunks (or to templates, prototypes, etc.), but solely reflected on the processing which follows from the distance analysis. Chunks (or templates) provide even more knowledge and flexibility to the system by enabling changes on the list of parallel priorities.

At this point an initial abstract role is created: the white king is perceived as a guardian of white’s unprotected end of pawn chain (at C3). The king is considered a weak *guardian*, because it cannot afford to be traded by any other piece. The pawns of the pawn chain are also perceived as guardians, and these are the first abstract roles created on the position. But these initially perceived roles are fluid, easily reconfigurable, as the philosophy of active symbols constantly re-describes the roles taken by the pieces. Subsequently, the white king is perceived to have high mobility, while the pawns are perceived to be ‘stuck’.

Thus in the parallel and distributed processing of those impulsive processes, a relatively small distance between two individual pieces may trigger the construction of a bond between them (analogous to a bond between individual neighboring letters in Copycat). There may be, at any point in time, multiple, interconnected chains of such bonds. But these distributed representations must be organized as a whole, that is, a unique structure that

gives priority to the most important relations of the position. So some impulses begin to transform a number of these bond chains into chunks. That is, the processes impose a “membrane” around a specific bond chain, making it an object on its own. These chunks may be linked to other objects (either other chunks or other individual pieces) and they should not brake easily – but only sporadically, given high outside pressure from competitive incompatible interpretations. This is another research problem that arises in this model: we will find at numerous times the problem of *multiple overlapping chunks*. In other words, since construction of chunks occurs in a bottom-up, pattern-driven fashion, some parallel processes may be working under competing, incompatible hypotheses, and only the overall context of the position can tell which chunk is the best interpretation in each case. This occurs regularly in Copycat, and what controls processing is precisely the activation levels of context-sensitive concepts (slipnet nodes). Thus this is a critical problem for the implementation of the present model: to devise a LTM chunk network that is capable of properly categorizing chunks (given bond chains) while considering the outside contextual pressures of a position. The implemented solution of this problem would be a significant step in machine chess research and is of considerable theoretical importance to chess cognition models.

The next figure presents a point of processing in which some notable facts are perceived: there are two blocked pawn chains, leading to the construction of two respective pawn chain chunks, and one additional chunk connecting them (which represents the fact that the pawn chains are blocked). As soon as these chunks are perceived, the system begins to acquire important information, such as: (i) it is pointless to attempt to attack a pawn chain at any of its points except the unprotected one (which inhibits more complex distance evaluations from other pieces to the ‘strongly guarded’ pawns); (ii) the pawns are blocked (which inhibits complex distance evaluations from the pawns to other pieces), etc. In this particular case the subcognitive impulses evaluating movement possibilities for these pawns (obvious exception of F6) gradually disappear from the list of parallel priorities (coderack) (Figure 12).

The identification of chunks leads to the formation of expectations, which influence further processing in a top-down manner, making the system less likely, for example, to (i) break chunks, or to (ii) exert effort inferring the distance between pieces within a chunk, or to (iii) infer distances using only glances; so the system will discard any potential movement to a square which is blocked by the opponent. As we will see, this is crucial for the emergence of a strategy.

After some blocked pieces are perceived, the system’s emergent behavior gradually shifts to working with more elaborate distance analysis, which consider, for instance, blocked paths and threatened squares. As a consequence of this, two facts naturally come into view: the distance from the

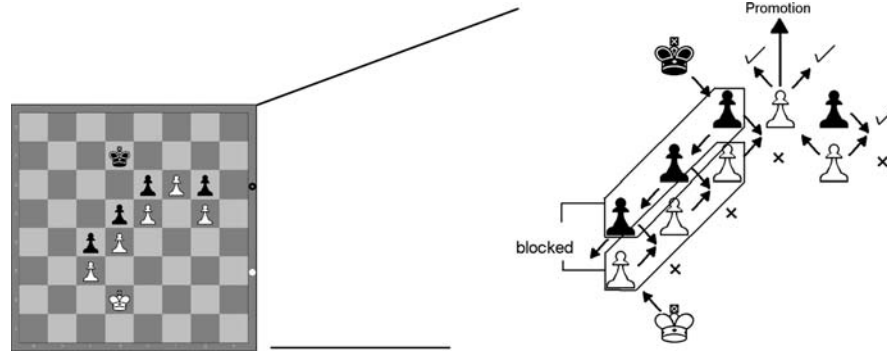


Figure 12. A subsequent point in the perception process: first chunks are identified, with the blocked 3-pawn structures (but still not the blocked 1-pawn structures). Crosses and marks represent the perceived open squares under attack and protected, respectively, from white's point of view.

white pawn at F6 to its promotion is only two moves, and the distance from the black king to that promotion point is also 2 moves. The underlying meaning of this relation is that the black king must guard from the promotion threat of the pawn at F6; the following “perception of the interlocked chunk” expresses this relation:

$$D(\text{black\_king}, \text{white\_promotion\_point}) \\ \leq D(\text{white\_pawn\_at\_F6}, \text{white\_promotion\_point}),$$

that is, in order to prevent from white's promotion, black has to keep the distance from its king to the promotion point at most equal to the white pawn's distance. Two new abstract roles are created: The first role is that the white pawn at F6 is perceived as a *strong attacker*, since a promotion would lead to a great advantage to white's overall evaluation. Square F8, however, is defended by the black king at a distance of 2 moves. *When this guardian of promotion at F8 role is perceived*, an urge will be triggered to drastically reduce the black king mobility to squares where it loses this guardian role. As a result, the constrained area for the king is displayed in Figure 13.

Another impulse is closely coupled with (and immediately triggered after) this mobility-reducing impulse. Since the mobility of the black king is drastically reduced, it can no longer attack any of white's pieces. So its abstract role is reclassified as having no potential as attacker. Furthermore, at this stage there will be no abstract role of attacker from the black side, and thus there is no need for white to preserve the current status of any highly mobile piece classified as unprotected guardian. It is an impulse that – when eventually fired – will lead to a new role for a piece at the other side of the board, the white king.



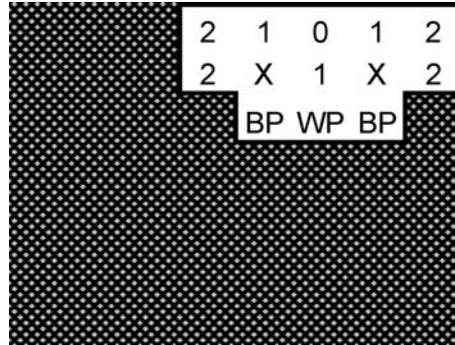


Figure 13. Mobility of the black king is dramatically reduced after an impulse recognizes that – because of white’s promotion threat – its principal abstract role is that of a guardian, “anchored” at a distance of 2 moves from square F8. (BP = Black Pawn; WP = White Pawn; X = guarded square; numbers denote how far the king is from the threat posed by the promotion; shaded squares, if reached, would trigger impulses conflicting with the role of guardian).

Let us refer to this relation between the black king and the pawn at F6 as a 2-move interlockedness relation, as the pawn, in its turn, cannot move from the protected square while the black king remains at this close distance; and so the pieces are interlocked and the situation seems stable – until one considers the upcoming attack of the white king. (Figure 14)

But how can the idea for this attack emerge? Let us proceed slowly: Since the processing of these previous impulses led to the perception that the white king cannot move away from F8 (and thus it cannot attack the unprotected ends of the white pawn chains), there is no more need for the white king to be perceived as a guardian of the pawn at C3. So an impulse deletes that role from the white king, and searches for other, new roles it could be playing. The white king can now find a role of attacker. But to find out to which piece it should direct its attack, a new urge will look for Black’s relatively unprotected pieces, and these impulses will eventually find out that the black pawn at E6 is ‘weakly protected’.

As the chunks tell us, white can only move 2 pieces (the king and the pawn at F6), but the interlocked relation between the pawn at F6 and the black king currently leave white restricted to moving its king. But where should the king go to? Once again, the knowledge embedded in the chunks tells us that there are few options for an attack: the black king, and the two pawns from the unprotected ends of black’s pawn chains – which are not safely reachable. The distance using a heuristic glance is 4 moves to each pawn, and 5 moves to the king, and so it becomes of increasing importance to elaborate on these estimates and also to find a possible ‘safe’ path for the attack. These urges should lead the system to include more complex corresponding distance measures as a high-priority task in the list of parallel priorities.

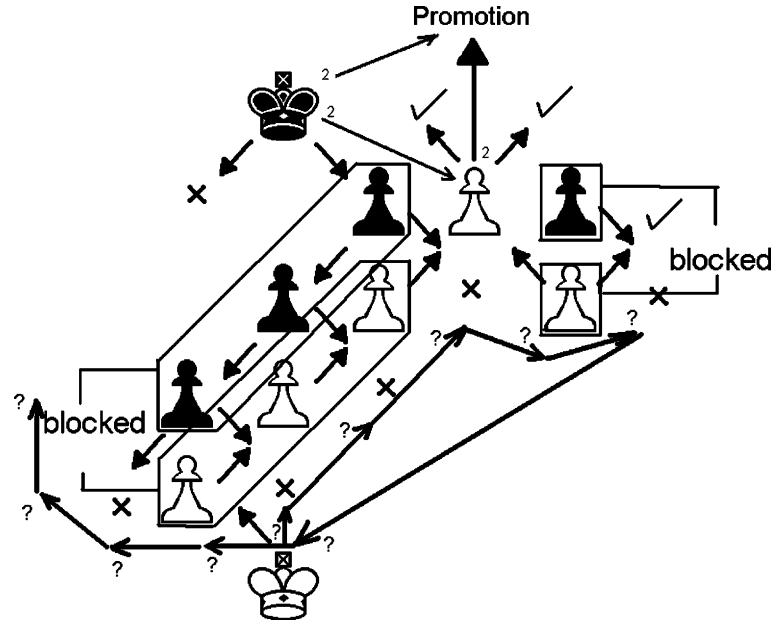


Figure 14. Distance estimations considering multiple response levels start to be triggered. Notice the backtracking required to measure the white king to black king distance.

Note that this computation strategy, to precisely measure distance from piece A to piece B, is much more focused than exploring the full combinatorial tree of all movements. It also corresponds to the integrated devising of a plan (“what is the distance to that king anyway?”) and to the careful planning of its execution (“and what is the best path for such an approach?”). In the process, since more complex distance algorithms rely on backtracking, the system is able to implicitly find out that the only safe passage lies on the queen rook file. Another important relation is perceived, as black’s pawn chain is also defended by black’s king:

$$\begin{aligned} &D(\text{black\_king}, \text{unprotected\_black\_pawn}) \\ &\leq D(\text{white\_king}, \text{unprotected\_black\_pawn}) \end{aligned}$$

That is, the system perceives that black must preserve the distance from its king to the ‘unprotected’ pawn of the chain as at most equal to the distance from white’s king to the same pawn, at the risk of losing the whole chain structure.

Meanwhile, a (higher level) impulse will be acting not on pieces or distances, but on *abstract roles* perceived. An impulse will look for pieces with multiple incompatible roles, and then perceive that the Black King has such

property: it is a weak guardian of the E6 pawn, and a guardian of square F8. This perception will trigger a tactic to make the black king compromise, i.e., to make it choose to defend either square E6 or F8.

The black king is playing two abstract roles: it must defend not only its pawn structure but also the potential promotion from white. Destruction of either role will lead to superiority of white. Perception of this relation should activate the appropriate tactics and is the culminating point in the emergence of a strategy for white. After some backtracking, the distance metric considering responses should be able to perceive that (i) either the black king will move away from the white king's path – leading to the capture of black's unprotected pawn, or (ii) the black king will block the white king's trajectory, but *it may still be forced to move* by advancing the white pawn towards a promotion – which could lead to a sequence of captures of black's 3-pawn structure. As the white king's distance to the unprotected black pawn decreases, white can force the black king to be at two places at the same time. And thus a strategy emerges without recourse to exhaustive combinatorial search (Figure 15).

#### 4.6. MOVE STACKING, MOVE COLLAPSING, AND A SINGLE COMPLEX DYNAMIC

The previous example demonstrated a clear case where distance estimations can ultimately lead to the activation of the right intentions and the emergence of a strategy. But it was a calm position, in which there were no exchanges taking place, and so the board simply would remain stable on the succeeding movements. Let us now consider the extreme opposite case, an example of a rather 'fiery' position, in which a furious exchange is taking place. A particularly interesting one was that provided in Figure 3. Let us consider how that representation might actually arise. And more specifically, let us analyze how the precise counting of the 'king trying to capture the onrushing pawn' may come to form, since our postulated system would not have any *a priori* reason to spend effort computing the distance of the black king to open square A8.

At start of processing, the distance glances immediately stress a problem for white: a low-value pawn is within 1-movement of white's valuable queen. After some processing it becomes clear that this is obviously the most pressing urge involved in the position, and it should immediately trigger an active intention to prevent capture of the queen. Two tactics will be activated and fight for attention: *exchange* and *escape*.

The escape tactic will consider the possibilities of movement for the queen. Notice that in its actual square, the queen is preventing the black queen from moving to A5, which, in combination with a black rook move to A6, could

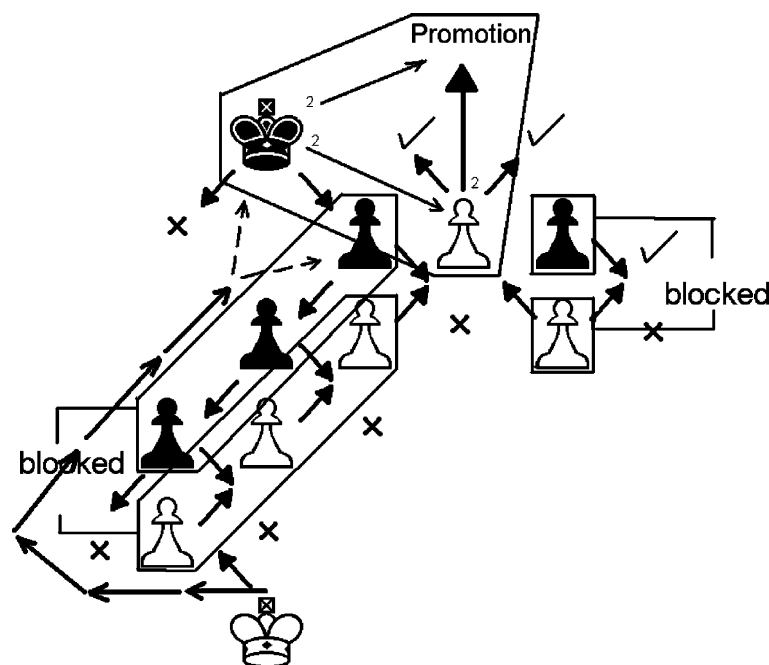


Figure 15. A complex dynamic is constructed, and a strategy emerges from the white king's distance to the black king and to the unprotected end of black's 3-pawn chain.

lead to a checkmate for black. So it is imperative to protect key square A5. But since the processing of the system is fragmented, while it is trying to consider this possibility of escape, other considerations are taking place simultaneously, and suddenly an impulse presents the possibility for an exchange: white already has a rook attacking the pawn, so it is natural to proceed with the possibility of *exchange* (in detriment of the *escape* tactic). The exchange tactic may get higher priority (and probabilistic attention) because of this perceived direct attack.

At this point a new mechanism needs to be put to use. The white rook will capture the black pawn, and this act will interfere with the distance evaluations that either (i) 'pass' through the corresponding squares, or (ii) with those distances and paths already constructed that could potentially pass through these squares. Though *distances that consider blocks* can sometimes be incrementally updated, that is not the case for the vast majority of distance estimations. After a set of exchanges, distances and piece mobilities need to be re-evaluated. So it will be necessary at particular points to re-evaluate distances after the consideration of such an impending movement. This brings us mechanisms for efficiently re-organizing positions for new distance estimations: move stacking and move collapsing.

The idea behind move stacking is to temporarily copy the contents of the emerging complex dynamic in working memory to a new area, incrementally

alter the piece positions corresponding to a single movement, and re-evaluate estimated distances, piece mobilities, perceived chunks, abstract roles and activated intentions.

It is well known that human players often consider alternative positions, and expectations of this looking ahead range from a dozen to an upper bound of a hundred positions evaluated at each time. At some particular points in the game, it is simply necessary to employ deeper look-ahead. Our postulated system must have a corresponding mechanism, such as move stacking, which will stack those potential subsequent moves.

A fundamental theoretical question is whether or not chess experts actually conceive multiple positions superposed.<sup>1</sup> For example, when a master player looks ahead for 18 positions, does the player mentally 'see' all these 18 positions superposed? Or does the player mentally 'see' one rapidly reconfiguring representation of the embedded pressures as each of the 18 positions are considered? Copycat and other active symbols projects have focused on a single representation. One argument is of interest here (Hofstadter, 1995): when we face ambiguous figures (i.e., Necker cube, vases versus facing men, etc.), it seems that the figures impulsively activate one coherent view, which is capable of being completely destroyed very rapidly and almost immediately reconfigured in new form. People do not perceive two *simultaneous alternative* superposed views of the Necker cube. People perceive them *alternating*. It is quite possible that when masters employ search and effectively look ahead, they do so in a way that retains a single representation, but a *fluid* one, in which, at each ply, whole parts of it collapse and are dynamically rebuilt (what would be an 'example collapse' is shown on Figure 17). This hypothesis seems more plausible than that of masters conceiving simultaneously all 18 positions 'mentally superposed'. This is the reckoning behind the mechanisms of move stacking and move collapsing: to preserve a single meaningful representational structure that undergoes dramatic change as moves are considered.

There are two major differences between the deep look-ahead of traditional programs and move stacking. First, traditional look-ahead programs obviously explore all potentially promising paths. Move stacking arises only sporadically – in fact, it seems to be necessary in only two types of situations: (i) exchanges, which obviously alter the number of pieces, the 'balance of power', and the underlying large-scale structure of the distance graphs; and (ii) moves which interfere with previously calculated (*under high quality evaluations*) distances, such as movements that block a particular path previously estimated as important.

The second difference between move-stacking and traditional tree search is that, in move stacking, each move on the stack represents *local changes* to the board position. Instead of having explicit representations of complete board positions, as carried out at each branch of the tree in traditional

programs, move stacking considers only the local transformation: that is, which piece moved from which origin square to which destination square. There is simply no need for a full board description. As each move may have global implications in terms of distances and chunk descriptions, it may be necessary to employ this form of tracking at times.

The most fundamental importance of move-stacking in reality comes from the fact that by looking ahead at limited times the system may re-evaluate how each of the originally perceived chunks will respond to the test of time, which pressures do not seem important at first, etc. In evaluating the new complex dynamic with move stacks, the postulated system may obtain a more precise view of what is essential to consider in each board position. Returning to our example, Figure 16 displays a move stack (from left to right) of some of the moves following from the position in Figure 3. Though not the fastest possible win for white, this move stack is a natural for this position: it is created by simply perceiving the immediate (1-move) attacks which will defend the most valuable pieces under threat at each step. Notice that it reflects the representation obtained by the original psychological studies of Binet (1894). As the figure makes clear, after the black queen captures the white queen, it is only a matter of time before the ‘onrushing pawn’ becomes free to run for its promotion. And this leads to the original representation given by experts, which will carry out – from *this new vantage point* – the careful counting of the movements between the white pawn and the black king.

## 5. Discussion

Let us take a moment to review the ideas proposed in this article. Two questions seem particularly important. First, how does the system reflect the architectural principles raised above, in Section 2? And second, what are the

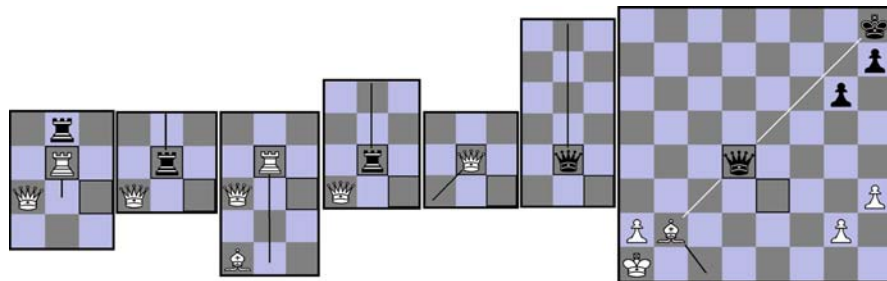


Figure 16. Move stacking: Initial look-ahead moves following from the position of Figure 3 (stack formed from left to right). These movements gradually alter the underlying distance relations of the original position.

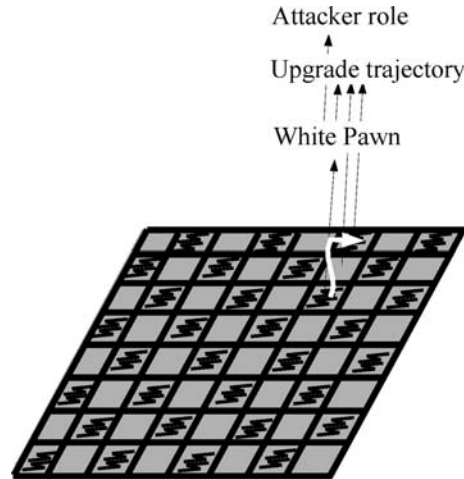


Figure 17. Move Collapsing. Associated with each move on the move stack are the original and final piece squares. Working memory is organized in such form that all constructed objects relating to those squares – from the most basic and concrete to the most elusive and abstract – are instantly accessible for immediate collapse (destruction), and a local reprocessing should rapidly rebuild new distance estimations, new-found piece mobilities, and fresh abstract roles. Mechanisms of move stacking and move collapsing enable the system to have a single representation – a single, but rapidly changing, complex dynamic – of a board position.

immediate research problems brought up by the postulated system? Let us start with a brief review of the core ideas of the postulated system.

As in Copycat, the major philosophical ideas of the postulated system are that of perception being at the core of intelligence and the associated image of mind as a collective, parallel, impulsive entity (such as proposed by Minsky, 1986). An ‘intuitive strategy’ naturally appears to emerge from the postulated system, without recourse to any exhaustive tree-search engine. In contrast, there is an emergent ‘feeling’ which comes forward from the very large number of simultaneously pressing urges from the distance estimations, chunk perceptions, and from the activation and activation-spreading of a number of intentions, or tactics. In review, the system is composed of:

- (i) multiple, automatic, fast, “impulsive” processes capable to collectively perceive a very large number of subcognitive pressures.
- (ii) Temperature-based feedback and control;
- (iii) Higher-level features (chunks) constructed out of smaller units. Instead of letters that bond with their neighbors, as in Copycat, this model has chess pieces that stochastically bond according to their perceived distances.
- (iv) Knowledge is embedded in chunks and in a chunk network that may activate the proper tactics (which correspond to slipnet nodes in

Copycat). Each individual piece and underlying relation between pieces holds a level of salience, which stochastically drives attention (i.e., information processing) to it.

- (v) Activation spreading in a chunk network creates a ‘feeling’ of what is important and of what is coming up in the game – this dynamic space of possibilities is very sensible to contextual changes and seems to be a plausible basis for a future ‘cognitive theory of intuition’.
- (vi) Chunks are encoded by distance structures and piece mobilities.
- (vii) Fluid intentions (i.e., tactics) are activated from a chunk network.

The philosophy behind these ideas has been summarized with the term “statistically emergent active symbols”. They are *active* symbols because *they* propagate subsequent control of processing – as opposed to “passive” symbols manipulated by top-down programs (see for instance the discussion in Hofstadter, 1985, chapter 26). And they are *statistically* emergent because the meaningful symbolic structure, the dynamic complex that in fact symbolizes the interlocked complexities of a position, emerges from a myriad of granular, impulsive processes.

How does the system reflect the architectural principles raised in Section 2? Let us look at each of them in turn:

*Principle 1: An intuitive program should not evaluate a large number of move combinations, only sporadically evaluate move implications (i.e., search the game tree).* It should be clear by now that the postulated model does not search the game tree in depth, as it does not evaluate most open movement possibilities. Instead, it evaluates movement implications sporadically – in fact, only when it estimates distances using a higher quality function. This estimation is also highly selective, as it considers moves *from piece X to piece Y*. The model is built on the idea of organizing the perception of a chess position in order to stress the pressures involved and perceive the main lines of action.

*Principle 2: An intuitive program should concentrate its attention (movement of eye saccades) to pieces in chess relations.* The model concentrates its attention to pieces in chess relations – that is, pieces with a single movement of distance. Attention in the model cannot be measured in terms of eye saccades, but it may be measured as the number of impulses that a specific structure received in working memory. Since the model based on distances specifically stresses out those pieces under chess relations, it could hardly be otherwise that attention is concentrated there.

*Principle 3: An intuitive program should be able to have familiarity with real board positions, while having difficulty of interpretation with implausible board configurations.* This should also be relatively easy to explain, since the chunk network maps relations arising in real positions played in real games, and random positions will simply not trigger many relevant chunks. This lack of



chunk presence in LTM creates huge difficulties in sustaining the chain reaction of activation spreading, which in turn intensifies the difficulties of interpretation.

*Principle 4: An intuitive program should manipulate a relatively small number of chunks (or other discrete structures) in short-term memory, despite having a relatively large encyclopedia of chunks stored in long-term memory.* A complex dynamic is composed out of smaller entities. Though a relatively small number of chunks will take part in each specific position, there is a large repository of potential chunks (which may possibly come from a relatively smaller set of general concepts such as forks, skewers, and other, non-verbalizable chunks).

*Principle 5: An intuitive program should exhibit a bottom-up, parallel, processing, as it is presented with a board position.* Analogously to Copycat, the system starts execution in a pure bottom-up, parallel, data-driven fashion. This mode of processing is gradually transformed to a mixed mode in which top-down (hypothesis-driven) pressing urges also influence processing, effectively dealing with the next principle.

*Principle 6: An intuitive program should exhibit top-down pressures, given by its previous expectations, which should enable the program to create a coherent and meaningful description of the board.* Top-down pressing urges are placed in the list of parallel priorities by highly activated chunks. Suppose the position is a natural one for the system to search for a skewer, that is, a chunk that fits the distance relations embedded in a skewer. If this concept of skewer eventually becomes activated, the system will rapidly have a top-down urge to test for the presence of such relationship. Now imagine that (i) there actually is such a skewer in the position and (ii) part of its underlying relation has already been chunked – but in an ‘overlapping’ chunk which is not necessarily compatible with the view of a skewer as a high priority item present in the board. It is possible that the system will have to ‘destroy’ the previous chunk in order to create the new chunk that stresses out the skewer relationship. This top-down re-description process occurs continually in Copycat, and in the domain of chess this process of chunk reconstruction has been carefully documented by Lories (1984, 1987).

*Principle 7: An intuitive program should construct dynamic complexes that contain pieces, relations between pieces, and eventually include empty squares of the board.* The system constructs ‘dynamic complexes’ based on a specific model of chunks. These chunks are composed of pieces, distance relations between pieces, (some) empty spaces, mobilities, and have a ‘membrane’-like structure that gives them an internal strength against the external pressure brought by competing incompatible interpretations. Empty spaces arise naturally in this representation as a byproduct of distance evaluations larger than a single movement.

And what are some research problems that immediately emerge in the development of such an “active symbols” chess program? Let us examine eight concrete problems:

- (i) At first, anytime algorithms for the evaluation of multiple levels of distance quality obviously need to be addressed; a question of particular interest here is: how many levels are necessary? In our postulated system we have discussed 5 levels of quality. It is possible that these levels suffice, but it might be necessary to have more levels of quality.
- (ii) Because the postulated system employs a mix of distance metric qualities, the problem of selecting the quality level to use at each step is central, and part of the intelligence of the system will undoubtedly reside in its solution. Obviously, the lower the estimated distance between two pieces, the higher the probability that a better estimate should be considered. Once again, we may look at the Copycat project for further clues. In Copycat, a certain letter, letter category, relation (or any object in general) receives attention probabilistically, according to its relative salience, and its *unhappiness* – a measure for how much ‘disconnected’ the object seems from the emerging complex dynamic. These variables should also be considered, and a piece’s (or chunk’s) relative value in a position also seems to be relevant in considering which distance relations must be most carefully calculated.
- (iii) Temperature-like mechanisms are very important in modeling chess perception. The reason is that at the initial glance of a board position, the system should be open-minded and register contradictory evidence, in order to find out the main pressures involved in the board. However, after some tactics for play become highly active, the system should concentrate on evaluating and testing these tactics, searching for attacks, interceptions, blocks, etc – since at this point it would be very counter-productive to continuously attempt interpretations incompatible with the increasingly active tactics. So it must be focused and reject most new ideas at this stage. A crucial research question concerns how temperature should be managed. It seems that in any implementation there should be a careful fine-tuning of the temperature-based management. Some simple experiments in Copycat provide very interesting erratic behaviors, such as forcing temperature to artificially low or artificially high levels during the course of a run – providing, respectively, an intensely focused ‘fundamentalist’ state (where new ideas are rejected and the system seems unable to perceive and attend to pressures inconsistent with the emerging interpretation) and a ‘blue sky baloney’ state, in which the system wanders through numerous concepts but is incapable of completion of a meaningfully coherent interpretation. Similar experiments in a chess project may be particularly intriguing.

- (iv) The problem of multiple overlapping chunks, and the associated ‘battles’ between incompatible interpretations is a core issue in such a postulated system. At any point in time, there will be many potential distinct conceptions to interpret each of the perceived pressures. So there will be, for each created chunk, a form of ‘tension’, in which external pressures want to destroy the chunk in order to replace it by a distinct one, and there is thus the associated problem of devising an ‘internal strength’ for establishing the breaking point of each chunk.
- (v) The chunk network. Is it necessary to create a network with those estimated tens of thousands (Simon and Shaeffer, 1992) of nodes? Or, alternatively, does it suffice to have few dozen *general* nodes, which include distance relations, but not piece squares and piece identities, leading to the general concepts of forks, skewers, pawn chains, blocks etc? Furthermore, each active tactic must be bound to a particular piece, relation, or chunk on the board.<sup>2</sup> Copycat could not handle such complexities, but other active symbol architectures such as Letter Spirit or Phaeaco are capable of managing such problems, so this should not be a critical obstacle.
- (vi) Learning mechanisms also should bring forth numerous difficulties of development and implementation. First of all, we must distinguish between (i) the acquisition of new chunks, (ii) the creation of new links between previously unrelated chunks in LTM, and (iii) the specific parameter tuning of each particular chunk’s estimated salience, ‘desirability’, ‘normalcy’, distance to related chunks etc. These are three different types of information that can be learned, and an advanced perception-based machine should handle each one of them. It is possible that there are ‘integrated’ algorithms to deal with the three in conjunction. But it is also possible that there are specific isolated mechanisms to deal with each type of information in LTM. So this issue remains unclear. The reader should note that Copycat does not model learning in any of those kinds. So it is still possible in principle to leave the learning problem for the future and deal exclusively with issues of perception and position interpretation in a first stage of examination of the ‘sense of position’ problem.
- (vii) In a related issue, the reader may recall another computational model that also seems to fulfill the architectural principles discussed here: the CHREST architecture (Gobet and Simon, 1996, 2000; Gobet and Jackson, 2002). The goal of this system is to model (mostly expert, but lately also novice) players’ perception, memory, and learning. These goals are shared with our postulated system, but CHREST is designed with the objective of *recall*, that is, of reconstructing a board after a brief opportunity of perception, instead of with the objective of move selection. Simulation results display an impressive match with human

data, recalling similar number of pieces and corresponding to the estimated times and known limitations of human memory. The system philosophy shares some similarities with the current framework, such as the assumption that perception is at the core of intelligence and intuition, and the central function played by chunks. But since the objectives of the systems are so different, it is natural to find architectural distinctions. One such distinction is that, in our postulated framework, *semantics* and the *abstract roles* played by pieces in a position are the principal considerations; this is in stark contrast to the mechanisms classically associated with the chunking theory, such as discrimination nets, which use a more syntactic – i.e., less concerned with *meaningful* information – means to access chunks in LTM. It is obvious that the postulated system will eventually need to have mechanisms for acquisition of chunks in LTM, but due to these differences it is debatable whether or not CHREST chunks and templates may be directly employed in such a Copycat-like chess architecture.

- (viii) Finally, on a theoretical note, it may be possible to prove mathematically a type of equivalence between our *distance graph* proposal and the classic *game tree*. A distance evaluation of full combinatorial search type at the beginning position of the game seems to share numerous properties with a complete game tree. Is there a theorem stating a form of equivalence between these representations? Are there implications for game theoretical models? These seem to be natural questions for this type of representation.

There are thus numerous pending issues for further research. Some of these issues have been under study for a long time, but from radically different perspectives than those postulated here. Others seem to be entirely new issues to be thoroughly analyzed.

The skeptical reader may argue that “this postulated system is very far from being proved; it is in fact being proposed in principle, and not as a comprehensive computer implementation; so it may not be clear what type of scientific contribution is actually being attempted here.” That is partially true: there are now more initiating questions than concluding answers. But one cannot refrain from pointing out that that was also true of the very first computer chess papers of Shannon, (1950a, 1950b) and Turing (1953): those were also works meant to initiate new research programmes. Herein is the very first detailed discussion of a computer architecture that would be capable of displaying the elusive question of sense of position. We do not have full answers, but we *do* have better questions. Furthermore, the questions brought up here lead us to a psychologically plausible synthesis of mechanisms involved in human chess intuition. We may be far from the postulated, Copycat-like, active symbols system, but now we may visualize it

on the horizon. There is at this point absolutely no system capable of performing this highly selective, granular, perception-based chess processing. And its proposal has indeed been presented in descriptive terms, as opposed to a comprehensive, formalized, implemented model (which is obviously a long-term research goal). The intention here, however, is not of speculation: even if the objective of this article is not the *test* of a hypothesis, the reader may find a noteworthy contribution in the form of the *proposal* of two new, outstanding, hypotheses:

**Hypothesis 1.** *Human experts access chunks by the perception of abstract roles. Chunks are created when a set of abstract roles are perceived to be played by the relevant piece, groups of pieces, or squares. These abstract roles emerge from levels upon levels of subtly perceived pressures, such as pieces, empty squares, piece mobilities, attack, defense and distance relations. Chunks are composed of sets of abstract roles, and their perception leads to a strategic vision of a position.*

This hypothesis is direct in opposition to current chess perception theories (which state that chunks are based on the very specific squares that specific pieces occupy on the board). That is, suppose we move all pieces of Wilkins position in Figure 2 one square down. According to the above hypothesis, humans would perceive the abstract relations and *the abstract roles played by each piece*. Since these abstract roles do not change after shifting the pieces, the perception of the board is simply ‘the same’. Nothing has changed, according to hypothesis 1. However, according to traditional chunking theory (Chase and Simon, 1973a, b; Gobet 1998), *this new board would bear no similarity at all to the original*, because no two chunks appear on both boards!

This hypothesis brings us to one of the *core theories* behind Copycat and all other active symbols architectures, such as Tabletop (French, 1992, 1995), Metacat (Marshall, 1999), Letter Spirit (Rehling, 2001), or Phaeaco (Fouldalis, 2004, personal communication). The *core function of perception is to find out the abstract roles that entities play*, be it in a restaurant table, a letter-string analogy, a particular font style, or a visual figure (Hofstadter, 1985, chapter 24; Linhares, 2000). In order to obtain the abstract perception of the role that each piece (or set of pieces) plays in a chess position, it is necessary to register the piece positions, register the fields of mobility of each piece, register the attack and defense relations, and also to have evaluations of distances between pieces.

Differing distance evaluations seem to account for some notable characteristics of human play. Sometimes, when there are fast eye saccades between related pieces, as modeled by a distance glance level of quality, such evaluations seem to occur “automatically and in parallel” (Reingold et al., 2001a). At other times, it seems that players are carefully and deliberately, counting the number of movements between pieces – which corresponds to higher quality distance estimations. This type of distance evaluation also can be seen

as a context-sensitive, highly selective tree-search, a widely known fact of human play, as opposed to the exhaustive exploration of combinatorial possibilities. Finally, distances stress some critical empty spaces on the chessboard, thereby accounting for a vital feature of human chess representations.

This hypothesis means that programs should make specific predictions about human players' performance. How are specific positions interpreted by most chess experts? What are the most relevant features? Which chunks are identified? Which empty spaces should be seen as vital? If in fact humans memorize boards from information related to these distance evaluations, this may clearly be tested by careful psychological experiments. For example, as we have already mentioned above, imagine two very distinct board positions  $P_1$  and  $P_2$  in which different sets of pieces appear. Consider the possibility that, on the surface, position  $P_1$  does not seem to have any similarity with position  $P_2$ . This is simple, because  $P_1$  and  $P_2$  do not share many pieces. But, as we start to evaluate distances, both positions gradually converge to a similar abstract distance graph between pieces. If this distance-network memory organization theory is correct, then necessarily human players must report that they see positions  $P_1$  and  $P_2$  as analogous, and maybe even occasionally report that "they seem in fact to be, structurally, *the same position*" (French, 1992). This conceptual model of the chess chunk can be tested empirically by generating board positions that share a deeply similar architecture, while a shallow dissimilar layout, and experts would be able to map pieces from one board into the other. Control experiments could include positions with the same pieces of  $P_1$  or  $P_2$ , but significantly distinct underlying distance structures, and subjects would not be expected to report positions as having deep similarities (despite the surface appearance brought by use of the same pieces). It is thus possible to devise numerous experiments to test this initial hypothesis. Let us turn now to a second hypothesis:

**Hypothesis 2.** *Copycat presents a psychologically plausible architecture for chess intuition.*

Table 2 summarizes the theoretical standpoint of the postulated system and its deep similarity to the Copycat architecture.

As we have seen above, the postulated active symbols model seems to account for numerous characteristics of human chess play. In other words, it is possible that the innovative Copycat architecture lies at the center-stage of an intuitive perception-based chess model. One can only be incredulous to find dozens of computer chess references with no mention of the insightful Copycat project and all its entailing possibilities. The very prospect that this project is of central theoretical importance to computer chess models is certainly an idea that deserves to be duly discussed, thoroughly scrutinized, and eventually tested.

Table 2. Synthesis of an active symbols chess architecture: A meaningful sense of position emerges from parallel chains of impulsive processing operating on multiple levels

Copypcat	Postulated System	Level
Letters perceived	Pieces perceived	Concrete manipulation of discrete structures
Bonds between letters evaluated	Estimated distances	
Created bonds	Trajectories and interceptions created	
Constructed chunks (groups, successors, etc.)	Constructed chunks (pieces, roles, trajectories, interceptions, etc.)	
Abstract roles played by letters & groups	Abstract roles played by pieces & trajectories	
Fluid concepts activated	Fluid tactics activated	
Continuous propagation (& decay of activation) of fluid concepts	Continuous propagation (& decay of activation) of fluid tactics	Flow of abstract conceptual feelings

As a final observation, the postulated system could not be more diametrically opposed to Deep Blue. Deep Blue calculates an *objective* function; while our postulated system models the subjective process of position interpretation. Deep Blue searches billions over billions of possibilities; while our system evaluates movement implications only sporadically and locally, *starting from piece X and leading to piece Y*. Deep Blue is a centrally coordinated process, a top-down system in which subroutines call other subroutines, which call still other subroutines in a nested hierarchical form of control. Everything in Deep Blue is specified *a priori*. It is ‘coldly’ efficient. There is no hesitation, no wasted motion – and one cannot refrain from comparing all this to the RF4 system’s handling of Bongard problems (Linhares, 2000). On the other hand, in the postulated system, control is initially distributed in a bottom-up, parallel, pressing urges fashion, where initial pieces and relations activate a chain reaction of impulsive processes. There is a lot of wasted motion. The system seems ‘hot’, confused, constantly dealing with inconsistencies, incompatible interpretations, and a noticeable level of disorder. Many pathways are considered erratically. These impulsive pressures (and the corresponding processes which perceive and respond to them) operate on a concrete STM, constructing and destructing structures in the gradual locking-in into a meaningful and coherent complex dynamic configuration. But they also operate in an abstract LTM network, in which chunks activate tactics, which propagate activation to related tactics, thereby taking charge of the next stages of processing in an emergent mix of simultaneous bottom-up and top-down influences.

But let us not get carried away. It will not be easy to test these hypotheses. In fact, the opposite is probably true. It may be even harder to develop the postulated system than a winning brute-force chess program has been. But if Winkler and Fürnkranz (1997, 1998) turn out to be right, this may be the best route towards our most valuable research. And it just might bring us close to understanding the rich unconscious subtlety involved whenever an intuitive strategy suddenly comes up right before our very eyes.

### Acknowledgements

This work has been greatly improved from the Copycat figures obtained with permission from Scott Bolland of University of Queensland, and from richly detailed comments from Douglas Hofstadter, Fernand Gobet, and Neil Charness, and an anonymous referee. The author is especially grateful for a series of discussions with Harry Foundalis back in 2000 – which spawned the project in the first place. Harry also undertook an incredibly excruciating proofreading of the manuscript, for which I am deeply indebted. The work has been financed by grants from the CAPES, FAPERJ and FAPESP foundations, and from the PROPESQUISA fund of FGV.

### Notes

<sup>1</sup> This question is obviously distinct from playing multiple simultaneous games or mentally storing multiple unrelated positions.

<sup>2</sup> I am grateful to an anonymous referee for pointing this out.

### References

- Atkinson, G. (1993), *Chess and Machine Intuition*, Norwood, NJ: Ablex Publishing.
- Binet, A. (1894), *Psychologie des grands calculateurs et joueurs d'échecs*, Paris: Hachette [Reedited by Slatkine Ressources, Paris, 1981].
- Bongard, M.M. (1970), *Pattern Recognition*, New York: Spartan Books.
- Bolland, S. (2003), A Copycat Java Implementation, available online at <http://www2.psy.uq.edu.au/CogPsych/Copycat/>, University of Queensland, Australia, April 2003.
- Cagan, J. and Kotovsky, K. (1997), Simulated Annealing and the Generation of the Objective Function: A Model of Learning during Problem Solving, *Computational Intelligence* 13, pp. 534–581.
- Campbell, M., Joseph Hoane, A. Jr. and Hsu, F.-H. (2002), Deep Blue, *Artificial Intelligence* 134, pp. 57–83.
- Chase, W.G. and Simon, H.A. (1973a), Perception in Chess, *Cognitive Psychology* 4, pp. 55–81.
- Chase, W.G. and Simon, H.A. (1973b), The Mind's Eye in Chess, in W.G. Chase, ed., *Visual Information Processing*, New York: Academic Press.



- Charness, N. (1981), Search in Chess: Age and Skill Differences, *Journal of Experimental Psychology: Human Perception and Performance* 7, pp. 467–476.
- Charness, N. Reingold, E.M. Pomplun, M. and Stampe, D.M. (2001), The Perceptual Aspect of Skilled Performance in Chess: Evidence from Eye Movements, *Memory & Cognition* 29, pp. 1146–1152.
- Chi, M.T.H. (1978), Knowledge Structure and Memory Development, in R. Siegler, ed., *Children's Thinking: What Develops?*, Hillsdale, NJ: Erlbaum, pp. 73–96.
- Church, R.M. and Church, K.W. (1983), Plans, Goals, and Search Strategies for the Selection of a Move in Chess, in P.W. Frey, ed., *Chess Skill in Man and Machine*, (2nd edn). New York: Springer-Verlag, pp. 131–156.
- de Groot, A.D. (1965), *Thought and Choice in Chess*, New York: Mouton.
- de Groot, A.D. (1986), Intuition in Chess, *International Computer Chess Association Journal* 9, pp. 67–75.
- de Groot, A.D. and Gobet, F. (1996), *Perception and Memory in Chess: Studies in the Heuristics of the Professional Eye*, Van Gorcum: Assen.
- Finkelstein, L. and Markovitch, S. (1998), Learning to Play Chess Selectively by Acquiring Move Patterns, *International Computer Chess Association Journal* 21, pp. 100–119.
- French, R.M. (1992), Tabletop: An emergent stochastic computer model of analogy making. PhD Thesis, University of Michigan, Ann Arbor, Michigan.
- French, R.M. (1995), *The Subtlety of Sameness*, Cambridge: MIT Press.
- French, R.M. (1999), Interactively Converging on Context-sensitive Representations: A Solution to the Frame Problem, *Revue Internationale de Philosophie* 3, pp. 365–385.
- Gobet, F. (1993), A computer model of chess memory, XV annual conference of the cognitive science society.
- Gobet, F. (1997), A Pattern-recognition Theory of Search in Expert Problem Solving, *Thinking and Reasoning* 3, pp. 291–313.
- Gobet, F. (1998), Expert Memory: A Comparison of Four Theories, *Cognition* 66, pp. 115–152.
- Gobet, F. and Jackson, S. (2002), In Search of Templates, *Cognitive Systems Research* 3, pp. 35–44.
- Gobet, F. Lane, P.C.R. Croker, S. Cheng, P.C-H. Jones, G. Oliver, I. and Pine, J.M. (2001), Chunking Mechanisms in Human Learning, *Trends in Cognitive Sciences* 5, pp. 236–243.
- Gobet, F. and Simon, H.A. (1996), Recall of Rapidly Presented Random Chess Positions is a Function of Skill, *Psychonomic Bulletin & Review* 3, pp. 159–163.
- Gobet, F. and Simon, H.A. (1996), Templates in Chess Memory: A Mechanism for Recalling Several Boards, *Cognitive Psychology* 31, pp. 1–40.
- Gobet, F. and Simon, H.A. (1996), Recall of Random and Distorted Chess Positions: Implications for the Theory of Expertise, *Memory and Cognition* 24, pp. 493–503.
- Gobet, F. and Simon, H.A. (1998), Expert Chess Memory: Revisiting the Chunking Hypothesis, *Memory* 6, pp. 225–255.
- Gobet, F. and Simon, H.A. (2000), Five Seconds or Sixty? Presentation Time in Expert Memory, *Cognitive Science* 24, pp. 651–582.
- Hofstadter, D.R. (1979), *Gödel, Escher, Bach: an Eternal Golden Braid*, New York: Basic Books.
- Hofstadter, D.R. (1985), *Metamagical Themas*, New York: Basic Books.
- Hofstadter, D. and FARG (1995), *Fluid Concepts and Creative Analogies: Computer Models of the Fundamental Mechanisms of Thought*, New York: Basic Books.
- Kurzweil, R. (2002) Deep Fritz Draws: Are Humans Getting Smarter, or Are Computers Getting Stupider? Published online at [www.KurzweilAI.net](http://www.KurzweilAI.net), October 19, 2002.
- Linhares, A. and Torreão, J.R.A. (1998), Microcanonical Optimization Applied to the Traveling Salesman Problem, *International Journal of Modern Physics C* 9, pp. 133–146.

- Linhares, A. Yanasse, H.H. and Torreão, J.R.A. (1999), Linear Gate Assignment: A Fast Statistical Mechanics Approach, *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems* 18, pp. 1750–1758.
- Linhares, A. (2000), A Glimpse at the Metaphysics of Bongard Problems, *Artificial Intelligence* 121, pp. 251–270.
- Linhares, A. (2002), 'Data Mining, Bongard Problems, and the Concept of Pattern Conception', in A. Zanazi, C.A. Brebbia, N.F.F. Ebecken and P. Melli, eds., *Data Mining III*, pp. 603–611.
- Linhares, A., (2003), 'Impulsive models of intelligence', Unpublished manuscript.
- Lories, G. (1984), La mémoire de joueurs d'échecs: revue critique, *L'Année Psychologique* 84, pp. 95–122.
- Lories, G. (1987), Recall of Random and Non-random Chess Positions in Strong and Weak Chess Players, *Psychologica Belgica* 27, pp. 153–159.
- McGraw, G. (1995), *Letter Spirit (Part one): Emergent High-level Perception of Letters using Fluid Concepts*, Bloomington: Indiana University, PhD Thesis.
- Margolis, E. and Laurence, S. (eds) (1999), *Concepts: Core Readings*, MIT Press.
- Marshall, J. (1999), *Metacat: A Self-watching Cognitive Architecture for Analogy-making and High Level Perception*, Bloomington: Indiana University, PhD thesis.
- Miller, G.A. (1956), The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information, *Psychological Review* 63, pp. 71–97.
- Minsky, M. (1986), *The Society of Mind*, Simon & Schuster.
- Mitchell, M. (1993), *Analogy-making as Perception*, Cambridge: MIT Press.
- Mitchell, M. (2003), (available online) letter string analogy problems available online from <http://www.cse.ogi.edu/~mm/analogy-problems.html>, March 2003.
- Mitchell, M. and Hofstadter, D.R. (1990), The Emergence of Understanding in a Computer Model of Concepts and Analogy-making, *Physica D* 42, pp. 322–334.
- Möbius, A. Nekliudov, A. Díaz-Sanchez, A. Hoffmann, K.H. Fachat, A. and Schreiber, M. (1997), Optimization by Thermal Cycling, *Physical Review Letters* 79, pp. 4297–4301.
- Morales, E. (1994), Learning Patterns for Playing Strategies, *International Computer Chess Association Journal* 17, pp. 15–26.
- Morales, E. (1996), Learning Playing Strategies in Chess, *Computational Intelligence* 12, pp. 65–87.
- Newell, A. (1980), Physical Symbol Systems, *Cognitive Science* 4, pp. 135–183.
- Rehling, J.A. (2001), Letter Spirit (Part two): Modeling Creativity in a Visual Domain. PhD Thesis, Indiana University, Bloomington.
- Reingold, E.M. Charness, N. Schultetus, R.S. and Stampe, D.M. (2001a), Perceptual Automaticity in Expert Chess Players: Parallel Encoding of Chess Relations, *Psychonomic Bulletin & Review* 8, pp. 504–510.
- Reingold, E.M. Charness, N. Pomplun, M. and Stampe, D.M. (2001b), Visual Span in Expert Chess Players: Evidence from Eye Movements, *Psychological Science* 12, pp. 48–55.
- Saariluoma, P. (1992), Error in Chess: Apperception Restructuring View, *Psychological Research* 54, pp. 17–26.
- Saariluoma, P. (1995), *Chess Players' Thinking*, London: Routledge.
- Saariluoma, P. and Hohlfeld, M. (1994), Apperception in Chess Players Long-range Planning, *European Journal of Cognitive Psychology* 6, pp. 1–22.
- Saariluoma, P. (2001), Chess and Content-oriented Psychology of Thinking, *Psicológica* 22, pp. 143–164.
- Shannon, C.E. (1950a), A Chess-playing Machine, *Scientific American* 182, pp. 48–51.
- Shannon, C.E. (1950b), Programming a Computer for Playing Chess, *Philosophical Magazine* 41, pp. 256–275.

- Simon, H. (1980), Cognitive Science: The Newest Science of the Artificial, *Cognitive Science* 4, pp. 33–46.
- Simon, H.A. (1995), ‘Explaining the Ineffable: AI on the Topics of Intuition, Insight, and Inspiration’, Paper presented at the proceedings of the fourteenth international joint conference in artificial intelligence.
- Simon, H.A. and Chase, W.G. (1973), Skill in Chess, *American Scientist* 61, pp. 394–403.
- Simon, H.A. and Newell, A. (1958), Heuristic Problem Solving: The Next Advance in Operations Research, *Operations Research* 6, pp. 1–10.
- Simon, H. and Schaeffer, J. (1992), ‘The Game of Chess’, in R. Aumann and S. Hart, eds., *Handbook of Game Theory with Economic Applications*, North Holland.
- Turing, A.M. (1953), ‘Chess. Digital Computers Applied to Games’, Bowden, ed., *Faster than Thought*, pp. 286–310.
- Wilkins, D. (1980), Using Patterns and Plans in Chess, *Artificial Intelligence* 14, pp. 165–203.
- Winkler, F.-G. and Fürnkranz, J. (1997), On Effort in AI Research: A Description Along Two Dimensions, in R. Morris, ed., *Deep Blue Versus Kasparov: The Significance for Artificial Intelligence: Papers from the 1997 AAAI Workshop*, Providence: AAAI Press, pp. 56–62.
- Winkler, F.-G. and Fürnkranz, J. (1998), A Hypothesis on the Divergence of AI Research, *International Computer Chess Association Journal* 21, pp. 3–13.