Questions in Cognitive Mimetics

Antero Karvonen[1[0000-0001-8032-2947]]

[1] University of Jyväskylä, Jyväskylä, Finland
anilkarv@jyu.fi

Abstract. Human thinking advances through questions and answers. Any field of human endeavor is permeated by the presence of questions, answers and presuppositions. Questions have a kind of universality, whereby one can place the question marks on anything, including questions themselves. The process of asking the right questions about the right things and in the right way are key for the explication of an approach. Recently, we have begun thinking about an approach to the design of intelligent technology: Cognitive mimetics. In brief, the idea is to take inspiration of empirical human thinking in specific contexts to develop AI solutions. The purpose of this article is to question this approach from various angles to take steps towards specifying it as a methodology. Some of the questions are like the chips a sculptor would make on a rough unfinished piece of marble. They and their answers uncover a particular shape immanent in the broad idea we have presented. Other questions are like the tools the sculptor needs to turn an idea into reality, these are the questions which will have permanent applicability within the methodology itself. Finally, some questions are promissory notes, they concern issues that will need to be responded to as the method construction proceeds further. The purpose is to establish a waypoint of where our thinking stands at the moment and an idea of where it may lead. The way we present and answer some of the questions may be of broader interest for researchers involved in fundamental and practical questions in AI design.

Keywords: Artificial Intelligence, Design Methods, AI Design, Cognitive Mimetics.

1 Introduction

Poincare anticipated the frustration of an important group of would-be computer users when he said, "The question is not, 'What is the answer?' The question is, 'What is the question?'" —J. C. R. Licklider in Man-Computer Symbiosis *(1960) [1]*

The ability to ask questions is a basic feature of the human mind. The presence of questions permeates the sciences, as it is a necessary element in reasoning and argumentation. Indeed, some have called the interrogative method *the* theory of reasoning [2]. Questions and assumptions can be thought of as orienting systems for cognitive resources. A question, and how it is understood, sets the stage for a quest for an answer. This may lead to activities such as research and experiment in empirical science, thinking and discourse in philosophy, or the realization of a product in design. In all cases, it is the question plus assumptions that orient actors and communities in some way. Consequently, paradigms open up through questions and the refinement of assumptions.

Cognitive mimetics (CM) is about asking questions of expert performance in a selected domain and seeking to formulate the questions and answers in a way that can be translated or mapped into a computational form for the design of artificial intelligence. What does this mean? Questions can be seen as a way of focusing our mental lenses. If we ask a question like: When did it rain last time? or Why did it rain yesterday? our mental resources are geared towards quite different sorts of answers. The question may entail a particular sort of answer that is admissible. In this article, we seek to state and answer some central methodological and practical questions that have emerged from the very idea of CM. Questions and assumptions orient the practice of mimetics: the kinds of questions we raise and the answers we expect influence activity in the field. Because CM straddles science, research, and design, we need to establish a harmony between the different orientations, in different phases of the mimetic process. CM can be understood in a "loose" and in a "tight" sense. A loose sense means simply taking inspiration from human thinking in solving AI problems or constructing solutions based on human thinking, without specifying further. A tight sense means explicating presuppositions and methodological questions in a more detailed manner. Here our plan is to take steps toward the latter, although still on a relatively high level. The idea is to ask the questions important for going forward to a more systematic description of the method.

1.1 Mapping Relations

As a preliminary, it will clarify the discussion to introduce the mapping relation in CM. The mimetic process (in general, including biomimetics) can be structured as a mapping (→) relation between a source (S) and a target (T): S → T. One way to differentiate mimetic design types is to evaluate what (and how) they are using as a source [3]. The broader field of biomimetics has traditionally been more focused on the functions, structures, and substances of biological entities. For CM, the source is the information content and processing of humans, and the target is, generally speaking, the software of a computer system. In terms of questions, we are asking how does the source do what it does and then asking how this can be implemented in the artificial system that is the target (AI). The S → T schema is useful for guiding broad questions in terms of what (in our case, who) is being used as a source, what is our target system, or what kind of a mapping relation could be constructed between them, among other questions. However, on the practical level, as research and design begins, the relation between the two is iterative. We must consult both poles of the equation—for example, as implementation reveals blank spots in research and research outlines directions for design. We may delineate this difference by replacing the mapping symbol (→) with an iteration symbol (↺): S ↺ T. An important further point to note is that the iteration and mapping relation also involves a transformation, due to the differences between the source and the target. Perhaps most pressingly for us, these questions emerge with the construction of machine semantics based on human semantics. We give mental content a strong role in CM. Thus, the limits, problems, and possibilities of machine semantics define perhaps the most interesting frontier for CM to explore.

2 Fundamental Questions

The importance of questions and answers for human thinking can hardly be overstated. Even basic cognitive processes, such as the retrieval, inference, or generation of information, seem to loosely accommodate a question-answer structure. Whatever their types or other properties, all questions seem to relate to information. Information is, in different ways, in the background of the concept of questions and answers. However, the way information is generated or selected by questions is not completely straightforward. Answers provide information, and some question-answer processes create information. For example, in classic group game, binary yes-no answers divide (often recursively) the elements of the answer set until a single element is left or can be identified. This would be akin to the classic information-theoretic view of reduction in uncertainty [4]. A why-question, on the other hand, may involve the creation of new informational structures and contents, carving out aspects of reality before ill understood. This in turn is a very different notion of information: a far stronger one. This question type is often associated with science, broadly understood [5,6]. Indeed, questions may be of many kinds [5,7], and the same simple type of question can have quite different sorts of answers, depending on the approach and context. There is quite a bit of ambiguity in the difference between a why and a how question, even in scientific discourse. Put simply, a scientific answer seems to be an answer to a why question. But simply stating a why, without specifying a how, seems to yield an uninformative answer, as the classic blackness of ravens illustrates [8]. What ties them together, however, is their nature as an interrogative information process. Basic questions refer to general question types recognizable by all: binary yes-no questions and the so-called wh-questions (who, what, when, where, why, and how). Because basic questions are like functional operators employed across different contexts, care must be taken to explicate the presuppositions that influence how the question is understood and, indeed, what is supposed to count as answer. These are often understood in an intuitive fashion, which would benefit from explication [9]. As noted by Saariluoma, Canas, and Leikas [10], questions have a kind of permanence—owing to their status as operators—that answers do not. Thus, questions are a kind of universal tool for thinking. They can even be turned on themselves—or more precisely, on the presuppositions and contexts in which they are presented. In the following paper, we will address some of the fundamental questions of CM to bring to the surface and address some of our presuppositions. The motivation behind this broad discussion on fundamental issues is to show the specificities that get introduced to AI design when certain basic issues are combined with the S → T schema.

Let us begin at the beginning. What are the originary questions for computer science (CS) and AI? The origin of Turing Machines [11] was in seeking an answer to (a foundational mathematical) question proposed by Hilbert: the Entscheidungsproblem. The answer to the question was negative, but the by-product of the proof was something else: the Turing Machine (TM), which was the result of (conceptual) *design.* In other words, the TM is an *invention* [12]. By Turing's approach, thinking acquired a technological form. Later, Turing pioneered the emerging AI paradigm by presenting another question: "Can machines think?" [13]. His answer was that it was "too meaningless to deserve discussion." Instead, he replaced it with a new question, which was operationalized as a question-game: the imitation game, now commonly called the Turing test. This question explicitly turned attention to functional equivalence and provided room for AI and computer science to develop on their own terms—a significant precursor to the idea of multiple realizability (of intelligence). As an aside, note how Turing knew that understanding and answering (natural language) *questions* is a powerful indicator of intelligence.

Returning to the TM, what was Turing's question there? It was to answer in unambiguous terms what the computer (back then a human) was doing as an information processor. Turing based his model on his perception of mathematical thinking [14]. What Turing established was a mapping relation between (his idea of) human computation and an abstract technical system (the TM), such that the operations coincided completely with the operations of the machine. The question was, can human computation be so precisely and unambiguously described so that a machine can be made to do it, and by what architecture? In our thinking, Turing's work is an important and foundational example of what we have called CM [15,16]

Turing's thinking was extremely influential. One can hear the echoes of Turing in the 1956 Dartmouth proposal [17], which was founded on the conjecture that "every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it." Note that if Turing [11] established that effective computation could be so precisely described, the question behind the 1956 proposal was whether all aspects of intelligence could be described as computation (broadly, as orderly symbol manipulation, transformation, storage, etc.). This in turn has bifurcated into the pragmatic questions of how intelligence can be created from this (computational) basis in AI and the philosophical-cognitive questions of whether the *human* is performing computations at some deeper level, even when not engaged in computation per se. The AI design question has been fruitful irrespective of its contested philosophical foundations [18,19] because it is a design question rather than a purely theoretical one. The design question of AI is founded on the multiple realizability of (computational) information processes and the one-to-many qualities in design. This complex back-and-forth analogical mapping between computers and brains—or software and mind—has been a defining feature behind the spirit of past century [20,21]. In passing, one might even note here that the theoretical successes of the cognitivist paradigm exemplified in Newell and Simon [22] have been tied to the success of the implemented computer models that perform the information processing task under investigation [23]—thus, tacitly to design.

Today, at least based on the highly popular AI textbook of Russell and Norvig [24], AI seems to have somewhat removed itself from a mimetic approach. They organize AI into four categories based on approach. The first they call the "acting humanly" approach, which is to make AI systems behave in more or less exactly the same way as humans. The second is the "thinking humanly" approach, which is the classical cognitive science method, to do with computational modeling of human information processing. The third they call the "thinking rationally" approach, which emphasizes pure logic and the making of correct inferences. The fourth (and the approach they advocate) is the "acting rationally" approach, which focuses more on achieving (by whatever pragmatic means) a rational outcome for the actions of the agent, taking into account the impossibility of perfect rationality in practice. While the borders are somewhat fuzzy between categories, let us attempt to see where CM might be placed on this map. To state the obvious, our approach is indeed rooted in the second, "thinking humanly" approach, but we take a more nuanced position. As already indicated, the mimetic approach starts from this position, but by adding the mapping relation plus basic assumptions about the difference between minds and machines, we accept it as necessary to approach machine intelligence on its own terms, be it the "rational approach" [24] or some other method. However, we believe that much of value to AI research and practice can be gleaned by starting from an analysis of an empirical source—the human expert. In all domains of sufficient complexity, intelligent behavior is a result of bounded rationality [25]. Thus, it seems that the rational approach [24], or limited rationality (a concept that came from Simon and the cognitive investigation of human limitations), is not possible to solve for all domains in the abstract. It is a useful concept and goal, but without understanding how experts in empirical settings *actually* make judgments and choices, it seems to require at least something *like* CM. Interestingly, Russell [26] may have noted this himself and expresses doubts as to whether the standard method of "building calculatively rational agents and then speeding them up" will "enable the AI community to discover all of the design features needed for general intelligence." CM is not primarily about artificial *general* intelligence, but even if it was, it seems like the rational place to look for its features would be human cognition. The conceptual benefit (and a goal) of CM is to make some of these issues clearer. The practical benefit is that the necessary bounds to rationality are not introduced tacitly or ad hoc by AI designers or programmers but are based on empirical research.

This leads to the important question of the goals of CM. Let us answer this by simply saying what it is not. CM is not about artificial general intelligence (AGI) [27] or the attempt to achieve strong AI. If it were, the route we have sketched would probably focus more on explaining general cognitive faculties in a different sense. Put simply, we would be more focused on theory and have followed the traditional terrain of cognitive science. Craver [28] presented a critical discussion of explanatory depth and what he called phenomenal and explanatory models in science. Anticipating this perspective, one might question whether CM only focuses on the phenomenal properties of its source, making sketches of it akin to Ptolemy's models. The answer however is no. On the level at which we focus our attention, we would say that it is precisely the mental contents that explain the actions (correct or incorrect). The explanatory ground for human action is in the mental contents, and the reasons and reasoning behind human action is our source. Of course, this is not the sole explanatory ground, but one that we feel is the most relevant, pragmatic, and in a way most interesting for AI design purposes. The difference is that we are not attempting to explain *the contents*, as such, in the source, in the abstract. We are seeking to explain *action* with respect to a context. It also tacitly opens up differences (between subjects), not on the level of general cognitive faculties, but on the level of the mental representations that they have. In researching many contexts, the common representation among actors will likely be saturated quite quickly, but the differences will be in the depth and variety of tacit knowledge, among other factors. The CM perspective is not about uncovering abstract mechanisms behind mental phenomena, but about empirically exploring the context-specific mental contents, mental models, and other mental representations that, for us, are the most immediate explanatory ground for human intelligence in a domain. These are, of course, supported or based on a whole host of basic general abilities, but we start by abstracting these, rather than the contents.

Let us circle back to an issue that emerged earlier. From the classical cognitivist perspective, the computation metaphor and method was mapped onto the deepest explanatory levels of human cognition. CM does not, or does not need to, follow the idea that all facets of human mentality are somehow at base computational symbol manipulation. In fact, it seems more natural to say that the mind computes only when it computes and operates on symbolic structures only when it indeed does so [29]. However, that many mental processes can be described computationally is clear, as is the fact that significant parts of many mental processes seem to involve the creation and manipulation of symbolic constructs. What seems more likely, and certainly more useful for design thinking, is that there is a gradient of face-value applicability of the computation metaphor (or mode). For example, the applicability of TM operational logic to human (or a particular kind of) computation approximates 1:1. Note here we are speaking not even of similarity, but of applicability in the design context. Wells [30] noted how Turing's analysis was driven by the pursuit of the essential core in relation to the effective computation of a real number by the application of specified rules. In other words, there may be conative or affective mental states which are in effect in the real human in the process of calculation, but these are largely inessential in terms of the analysis Turing was seeking.

This question cannot be decided a priori, however, in the broader contexts where CM operates—and for CM the question is open until decided based on the task and the context of analysis. Obviously, the applicability of computation to emotion seems far more distant than for computation. Squaring this requires the forced mapping of a phenomenal event to something like a correspondence between inputs and outputs, or some other transformation into a symbolic abstract form. This is, of course, the classic functionalist move in cognitive science philosophy [31-33]. Now, discarding the identity of emotions as emotions in favor of a functional representation is a forced move owing to the computational cognition paradigm, which we do no need to make, philosophically. What we do need to accept from it, however, is that if and when an emotional state is important for the chosen action of a research subject, the mapping relation awaiting us in the mimetic part of the design will demand some kind of symbolic transformation. Practically, the emotion as such will need to be investigated in terms of its causes and effects. These will often reside in the tacit domain: subconscious intuitions and hunches or "bad feelings about that" are cases in point. Thus, whatever the ontological kind of the source mental phenomenon may be—tacit, emotional, intuitive, etc.—we know that to implement it on a machine, it needs to be explicated and mapped or converted to a symbolic form. Boldly put, for CM, the achievements of computational cognitive science become mimetic design methods for us—but not necessarily anything more. In fact, they become important precedents for how the mapping operation between the source and the target could proceed [34]. The end result of a CM design process would of course retain many features of the source and be a kind of model for it as an information-processing artifact. For us, however, it makes less difference whether it is plausible as a deep explanation of the behavior of the source as long as the artifact works. The criteria are simply different and come from the domain of design, rather than from (cognitive) science. It also distinguishes our work from work on cognitive modeling. These are not irrelevant and do inform our work, but to make the point crudely, it would seem senseless to imitate the limitations or fallibilities of human cognition in AI, as must be done in cognitive modeling.

Djikstra's [35] critique of mimetics illustrates this well. He stated that mimetic copying has the unfortunate connotation that one could not thereby improve limited and fallible human information processes. This is decisively not our intention. However, the danger here is that CM causes designers to fixate on sub-optimal strategies and solutions. This is something that we must guard against to see how it plays out empirically as the method construction proceeds further. Nevertheless, it is clear that AI is fundamentally about imitating—or at the very least replacing—human information processing. The problem is that this may proceed even more fallibly and in an ad hoc manner in specific contexts, if the source is not an actual expert but the imagination of a designer or programmer. Making this link explicit while managing presuppositions and allowing for freedom on the implementation level remains a valid path for AI.

Let us stress again, however, that our approach is not about foregoing explanatory depth. Indeed, to explain an action by referring to mental contents in any domain of expertise is at first glance an enormously complicated task, especially as common-sense knowledge begins to seep in from outside the specific domain. Which branches can be pruned and abstracted out is a question that can only be settled on a case-by-case basis. It is here that pragmatic design questions, the context of investigation, and the role of the designer using mimetics become important.

3 Design Questions

The importance of questions in design processes is obvious. Dym and Brown [36] illustrate this by the task of designing a "safe ladder." From the task immediate questions emerge. What is safety? For whom? What should the inclination of the ladder be? How much weight should the steps be able to bear? How can slipping be prevented? How can movability and stability be reconciled? What is the context of use? Each question may inspire further questions and, over the product design and development process, many more will be asked and answered [37]—either explicitly or tacitly. Each question is a step toward converging the many possible paths in design to some set of design problems to be solved, which of course inspires new questions. The goal is really to discover the right questions, as well as the right answers. The practical fact is that CM operates in the realm of (science-based) design, and the interdisciplinary nature of the work demands a kind of harmonization among perspectives. As noted in the introduction, CM can be understood in both a loose and a tight way. Following the loosest interpretation, CM offers an approach or perspective for AI design. A tighter formulation (which we are attempting here) begins to sharpen the idea towards a method or a collection of methods within an approach. CM is not there yet, but parts of it are taking shape. The importance of methodological work is clear [38,39] and likely to be needed to gain the cognitive information in a form suitable for AI development. Along the way, there are many methodological, practical, and empirical questions to settle, some of which we will outline below.

Let us start by addressing a very basic question: what is CM's relation to theory or, more broadly, science? Gregor [40] provides a discussion of some of the background issues—although from a different perspective and in a different context. Gregor focuses on theory (in information systems, IS), which makes the details of the conceptualization ill-suited for our purposes. Nevertheless, with a different interpretation and context, the core concepts are quite useful. Gregor outlines a taxonomy of five types of theories:

1. analysis (says what is)
2. explanation (says what is, how, why, when, and where)
3. prediction (says what is and what will be)
4. explanation and prediction (says what is, how, why, when, where, and what will be)
5. design and action (says how to do something)

Is CM a theory? It is clearly not a *theory* but *a design approach.* Still, we can identify which of the above are most important for this approach. CM is about analysis (of the source). It is about explanation (of the source). Finally, it is about design and action (how to investigate and analyze the source and how to map it onto a target).

Naturally, our perspective is theory-laden [41] as we maintain certain presuppositions such as the very existence of information processing as an explanatory level of analysis, the centrality of mental contents in explanation of action, and the importance of explicating tacit knowledge. Here, they are in the first instance exploratory viewpoints or tools, whose value is measured in the extent to which the method or activity of CM leads to successful design outcomes. The key question is how this viewpoint is communicated across the mimetic process.

We have prioritized context-specific thinking over general cognitive faculties. First, this is due to the presupposition that the contents of particular minds explain their success in particular domains. General faculties are of course necessary, but we choose to keep this aspect secondary, because it allows for more freedom on the implementation level in computers. Thus, generic results (methodological or concrete) of CM may relate mostly to the possibility of machine semantics based on human semantics. The core question is really how to analyze the source in action and, in a faithful way, abstract the mental contents, and further, re-introduce those contents in a computational setting. This is where CM has the potential to place the question marks deeper for AI: what are the computational equivalents of mental contents? Such questions must be answered in practice, somehow. If among those answers emerge new ways of thinking about or solving this problem, CM will have in fact made a theoretical contribution to AI discourse. However, we choose to proceed via practice (i.e., the design approach way). In summary, CM is as a design method normative rather than theoretic. Technical artifacts—including the results of CM—are judged on normative grounds [42]. However currently, it is too early to specify the criteria by which the method or its' results should be evaluated. Simple performance may lead to guiding the method on the wrong path, and it is not yet clear where and how exactly the perspective will deliver most value. Two distinct starting points in computing history may illustrate this point.

One question, which relates to design goals, could be called the starting point of CM. The history of computing shows two related but subtly different starting points for the relationship between natural and artificial intelligence, which we can call the AI and symbiotic augmentation approaches. The history of computing and computer design has several mimetic or analogical mapping examples. Turing [11] is of course one on them, while Shannon's mapping of logic to circuitry provides another. It is not exactly CM, but on the other hand, logic is, despite whatever platonic qualities it may have, in the end a mental phenomenon. We have previously [15] called the McCulloch and Pitts [43] paper on neural nets an example of *bio*mimetics, which seems plausible on the one hand, given that it is focused on idealized versions of biological, physical neurons. However, the example is more complex, given that McCulloch and Pitts were mapping logic to these idealized neurons and in fact looking for and proving their equivalence (equipotentiality) with TMs. What we may call the AI-approach to computing seeks to match or surpass human information processing in some chosen domain. When considered carefully, one can see that even the most powerful or quasi-autonomous AI systems today are, in the final analysis, mappings from human thought to computational systems. It simply that the mapping here is from the thinking of the designer to the computer, rather than from the thinking of a domain-expert via a designer to the computer. Be it hard-coded or learning algorithms, it seems clear that all AI systems are in the final analysis actually *displacements* of human intelligence.

A related but noticeably different strand in the history of computers is what we might call *symbiotic augmentation* [1,44,45]. This is related more to the human-computer interaction side of AI design. The spirit of this branch is that, rather than replace, we should seek to expand and increase the power of human thinking by complementing it with the strengths of computers. Of course, sometimes to increase the power of human thought, parts of it should be replaced, so the difference between the AI and symbiotic augmentation approaches is not hard and fast.

The mimetic approach is, in principle, naturally suited to both *AI* and *symbiotic augmentation* perspectives. However, our answer to the question of which approach to explicate for the rest of this paper is the AI-approach. We start from the premise that in (cognitive) mimetics, there exists a source that can, by way of information processing, exhibit success in a task that we wish to implement on a computer. Our design goal is to construct a computational system that is equally successful in the task by the mimetic method of taking inspiration from human information processing. The scope of mimetics here is to develop autonomy for the digital artifact within the bounds of the task. In practice, however, the limitations of digital intelligence mean a co-habitation or co-working between artificial and natural information processing. One here notices the value of holding the *symbiotic* perspective in mind as well. It may be best to consider the symbiotic perspective in a way parallel to the AI-approach, but in terms of methodology, they are likely to be different, so for the sake of scope we will focus on the AI approach here.

3.1 Research

Let us next present the way we have approached the development of the method in terms of questions. Let us assume that we have already been presented with a task context: operators of a paper machine. The company is seeking to find AI and/or automation potentials in the range of tasks that they perform in addition to ordinary work. This is on-going research, so we will discuss the findings on a rather broad level. On a general level, our first task is to identify a *system of regular actions* [10]. This is to be understood as a joint cognitive system [46], consisting of multiple actors interacting with technologies to complete overall tasks and goals. The natural question that follows is "who?" followed by "what?" and informed by a "where?" The who-question here refers to identification of those individuals who are involved and skilled in the tasks, giving us research subjects. The what-question refers to a description of the task: what they are doing, why, and how? The where-question is an approximation of contextual issues: where is the task to be done, by what means, and so on. Practically, it will be necessary to have someone with in-depth domain knowledge to facilitate translation and understanding between research subjects and researchers.

Next, we must ask the question of how and combine it with a distinct sense of the why. Essentially, having identified a system of regular actions and having sketched out some of its features, CM needs to penetrate deeper into the actions themselves. On a high level, action can be described by five structural elements: *goal, agent, artifact (tool), target,* and *context* [10]. Note that, in a particular sense, we have already answered these questions in our preliminary sketch of the research context, but the same elements can be used to probe more detailed questions with respect to very particular actions. For example, we could now begin to ask *why did the operator increase the RPM of the pump at time T?* We can see that the agent used a tool (a DCS in this case) to alter the state of a target (the pump RPM). To understand this action, we need to know what goal this action served to accomplish, what elements of the state of the system it affects, how those states relate to the operator's goals, and what the contextual issues that make this action reasonable at this stage are. These questions inspire further questions and begin to uncover the mental operations and contents that CM is seeking.

How can we gain information about mental contents for specific tasks and contexts? The answer must be empirical, so we have made use of observation and think-aloud protocols to gain research data. Here, and especially in follow-up interviews, questioning becomes a practical necessity. The idea is to probe and bring to the surface tacit knowledge concerning chosen actions. This is where depth of analysis is gained. This means that often the operators cannot provide an explanation for all of their behavior, although the behavior is correct and highly efficient. Thus, in research, we must triangulate between overt explanations from subjects, their behavior, verbal protocols, and contextual facts. The explanatory depth from CM emerges from discovering parsimonious content-level explanations for action. It is a content-based answer to a why-question.

3.2 Research Example

An example will illustrate this better. Let us observe that a human operator in a factory responds to an event by some series of actions. For convenience, let us say that the event is mapped to a meter reading, and the actions map to a set of remotely controlled pumps and valves. Cognitively, the meter reading is interpreted (mapped) as an event of some type by the operator. In fact, the operator is answering a tacit series of questions: what I am seeing, what does this mean, do I need to intervene, and by what operations can I do so—among others. The whole process is so efficient in an experienced operator that mere observation would likely only give us event-action pairs. The problem with expertise is that it seems to increase as a function of the degree to which to knowledge and skills become transparent (tacit) for the expert. Here is where iteration becomes important. Assuming we have observations of events and corresponding actions plus fragmented thoughts expressed in the protocols, to grasp the structure and the mental contents corresponding to action we need first to identify regularities and then to follow up with qualitative interviews. The purpose is to tune in to specific actions to discover their meaning. As stated, the operator's thinking seems to fit a kind of tacit self-questioning process, and thus questions like "why did you choose this action" or "why was this event significant" trigger outpourings of tacit knowledge. The operators can often answer these questions if asked.

The problem with researching actions in complex and dynamic settings is that it is difficult to establish a stable point of reference, such as is possible in many games like chess. Thus, discovering the "rules" and goals of the "game" is not definitively given beforehand, but must be discovered, and indeed is a major part of the results of the research. In practice, to establish a stable foundation we have searched for the following: goals, the space of possible measurements, and the space of possible actions. This triangulation provides one method of attack for the problems of relevance and action.

We have approached operator behavior in terms of goal seeking. This means that action and behavior can be explained in terms of reducing the distance between the current state of the system and a goal state. Events and observations are evaluated against deviations from this state or the path toward that state. This gives the behaviors of the operators an episodic structure: events occur and are dealt with if they violate the goals of the operator, which can be identified with a system state.

The state of the system maps onto measurements. For the measurements to make sense, the operator must have a mental model of the system, in which measurements map onto certain aspects of the system. On the face of it, the objective set of measurements is large and internally related. The cognitive answer to the question we are seeking is which measurements are relevant, which not, and when. The measurements simply map to facts about the system state, but how they relate to each other and to the state the operator is seeking is an open question before research. The goal is to discover relevance-based subsets of the measurements that can be explained by goals, mental models, and tacit knowledge.

It is often possible to compute the space of possible actions at least on the level of the interaction points between operator and machine. This could be thought of as the list of "moves" available to the operator. What we must seek to explain as a third part of the explanatory structure are the principles of the action selection. A goal or a goal-state maps onto the measurements. The moves, on the other hand, map onto system states and measurements. Thus, only subsets of moves (and their values) are either *relevant* for influencing the system state at some particular time, or *appropriate.* Again, we can foresee the problem of tacit knowledge here. The fact is that certain actions may have consequences beyond some small part of the system. Thus, the *path* towards the goal state chosen by the operator is likely constrained, or guided, by tacit knowledge of the interrelationships within the system and instantiated in a mental model of the system.

Triangulating and iterating between these perspectives is one method of uncovering the structure, properties, and contents of the cognitive system that currently keeps paper machines operational in our research context. Here we presented some viewpoints on how the method might be specified and some of the ways in which information can be gathered and classified. For us, this would *still* be a kind of initial sketch in terms of the explanation of actions. It provides a core framework in which to ask further questions. Some of those questions should come from design. The next stage is to establish a mapping relationship and step into the world of possible design solutions.

3.3 Implementation and Design

Next, we must ask the question of how—this time in a different sense, because we are now concerned with the mapping or implementation relation. So far, this part of the equation is speculative, as we have not engaged with AI designers or programmers. Let us assume that we have now established at least a direction for the idea of the tasks we wish to realize in AI. Our task going into the design phase would be to establish a principal solution, the key problems, and a set of conceptual design variations that answer those problems [39]. This is where the *problem* and *solution spaces* [38] should become informed by research. One may notice that we have already tried on, as it were, the designers' shoes by tacitly imagining a halfway abstraction between the source and the target. It seems almost impossible to think about these things without simultaneously imagining possible solution patterns [38]. This is not necessarily a problem, but something to be aware of, given Djikstra's [35] critique (for example) and the danger of fixating on sub-optimal solutions.

Our content-specific presupposition carries over to our presuppositions on design. Visser [47] summarized the problems with an oversystematic and formal approach to design. Design problems of creating paper machine AI are no less domain-specific than the paper machine operators' skills and knowledge themselves. Indeed, part of the whole point of cognitive mimetics is to understand the specificities that get introduced by the design context, the constraints of computational systems, and human thinking. Thus, the design process should not be too rigidly defined normatively, since actual design (or research, or any creative thinking) processes will not follow those patterns anyway. Too little specification on the other hand, and it becomes impossible to build the mimetic bridge and common understanding between different actors.

3.4 Design Problems and Representations

For Cross [38] a successful design process involves the skillful management of the *problem space* and the *solution space.* CM is intended to make contributions to both. We follow Visser [47] in noting that while design involves problem-solving, it not only problem-solving. Equally important are problem-finding, problem-specification, problem-structuring and re-structuring. Furthermore, some problems can be solved by routine methods, and this applies especially to AI and to software development more broadly. For instance, software patterns are a subfield of software development on solving recurring issues by applying known patterns [48]. Here is where intentional management of the problem space and the solution space becomes important. If a problem found in research has a known solution pattern in AI, there may be no need to invent something new by mimetic means. This is a likely scenario in many real-world settings. Nevertheless, it is fully possible that we may *want* to attempt to solve it by a new way for the sake of discovering a new pattern for solutions.

From another perspective, identifying and analyzing the *system of regular actions* [10] that is the source (and in a way the eventual target) of the mimetic process can uncover the structure of the problem space in many ways. First, we will by necessity uncover an expert-based understanding of the problem domain. It uncovers the real problems. This already scopes and ties the intended design to actual human action and a naturalistic picture of the domain. Second, by in-depth analysis, the empirical problem spaces (as they complete their tasks) *of the operators* become a direction towards possible technical solutions in AI. Just as there are patterns of software solutions [48], so there are (context-specific) patterns of solutions in human action. Typically, intelligence demanding tasks are such that they can be achieved in many different ways and the intelligent way is dictated partly by contextual issues. The variety of strategies for succeeding in tasks give direct inputs for possible technical solutions to the problems, which can be evaluated on normative *or* pragmatic grounds. One can even imagine collecting the patterns in human action to libraries for a kind of context-specific ontology of action which when abstracted (and connected to software solutions) could be employed across contexts.

Design processes, including problem and solution spaces, can be understood in terms of representations [47]. Namely, the design process is cognitively speaking a series of evolving representations (mental and concrete) that iteratively seek to specify the requirements, functions, objectives, and constraints for the artefact in an increasingly specific manner. This “arc” from abstract to concrete is, in mimetics, an arc from concrete to abstract and back to concrete, following a gradient from research subjects to design to embodiment in a computer system – and iterating over this arc as necessary. This “arc” presents some issues in terms of establishing shared and meaningful representations across the mimetic design process, which is in the end by necessity a joint venture among various fields of expertise, from human cognitive research to design and to programming.

Perhaps most clearly this danger will manifest if there is only a superficial understanding of what research is uncovering. There may be no silver bullet for this problem, as we do bring forth concepts from cognitive *science* that provide a framework for our thinking. But the thinking and the background of, for example, programmers, is different. Thus, it will be an interesting question to see how the mapping relation succeeds and what factors may cause or hinder success. The assumption is that the more deeply the programmer understands the source, more deep and interesting the solutions will be. For the broader development of AI, we can hope that the questions marks can be put deeper by this method. Put more pragmatically, there will be a "user" of the method, and in the very least, a user of the results of the research who is likely to come from a different field than cognitive science. Floridi [7,49] has illustrated this by his idea of Level of Abstraction (LoA). It can be interpreted simply by saying that individuals with different backgrounds "see" different things when looking at the same object or phenomenon. In our case, it means that one LoA (of a LoA) is to be iteratively mapped onto another LoA. Put another way, there is the LoA that the source has (the operator in the paper mill), there is LoA which the mimetic research takes on the operators' LoA, which needs to be translated to the LoA of a programmer-designer who, in addition, creates a *machine*-LoA that structures the program. Establishing these translations and mappings is a key part of finding success with CM. Essentially, the method will fail if the mapping relation fails. Thus, it is important to consider, as a methodological design question, the kinds of mental activities that mimetics invokes (or should invoke) in practitioners, and the foremost is thinking by analogy [50,51]. Indeed, CM can be thought of as a relative of *design by analogy* [52-54]. Moreno and colleagues' [54] study is promising, in terms of the benefits of analogical thinking on design results may have, but much more thinking will have to go into if and how this benefit can be realized in cognitive mimetics.

Finally, the question of multiple realizability and the possible ways in which cognitive processes can be implemented on machines remains an important theoretical issue for CM. Research, design, and practice have a common ground in this respect.

4 Conclusion and future directions

The impetus for this article was the need to specify and answer methodological questions in cognitive mimetics. One may read this as a report on a work in progress. The idea of the mapping relation seems to be an apt tool for surfacing major issues in the methodology. First, we discussed some historical antecedents and attempted to fence our idea with respect to some major ideas in cognitive research and AI. As noted, on the loosest interpretation, cognitive mimetics is simply about giving a name to an existing phenomenon, the fact that taking inspiration from human thinking has played a major role in AI and computer science history. From this simple premise, however, many questions emerge, and the purpose of this paper was to first state some of the questions and where possible answer them from our perspective. The point is that to that to make progress, these questions must be answered in some way. They provide compass points on an issue that involves enormous complexity if approached without the right questions. The way we have answered the questions shows the outline of a particular path towards AI solutions based on human cognition. This is not the only path – but it is a path we feel makes sense and gives a fresh perspective on the issues. It is also not necessarily the easiest path, indeed the specificities that are introduced by the mapping of human thinking to computational systems introduce a host of issues, the least of which is not the ability for different experts to achieve mutual understanding. Many of the questions are such that they can't be decided *a priori* by speculating on them in a reflective manner. Much work remains to tackle these issues and they can only be done by iteration in practice by combining research and design.

References

1. Licklider, J. C.: Man-computer symbiosis. IRE Transactions on Human Factors in Electronics **1**, 4–11(1960).
2. Hintikka, J., Halonen, I., Mutanen, A.: Interrogative logic as a general theory of reasoning. In Studies in Logic and Practical Reasoning, vol. 1, pp. 295–337. Elsevier, Amsterdam (2002).
3. Karvonen, A., Kujala, T., Saariluoma, P.: Types of mimetics for the design of intelligent technologies. In International Conference on Human Systems Engineering and Design: Future Trends and Applications, pp. 40–46. Springer, Cham (2019).
4. Shannon, C. E.: A mathematical theory of communication. The Bell System Technical Journal *27*(3), 379-423 (1948).
5. Jaworski, W. The logic of how-questions. *Synthese* **166**(1), 133–155 (2009).
6. Van Fraassen, B. C.: *The Scientific Image*. Oxford University Press, Oxford (1980).
7. Floridi, L.: (2013). What is a philosophical question?. Metaphilosophy **44**(3), 195–221.
8. von Wright, G. H.: Explanation and Understanding. Cornell University Press, Ithaca (2004).
9. Saariluoma, P.: Foundational Analysis: Presuppositions in Experimental Psychology, vol. 2. Psychology Press, Hove (1997).
10. Saariluoma, P., Cañas, J. J., Leikas, J.: Designing for Life: A Human Perspective on Technology Development. Springer, Cham (2016).
11. Turing, A. M.: On computable numbers, with an application to the Entscheidungsproblem. Proceedings of the London Mathematical Society **2**(1), 230–265 (1937).
12. Petzold, C.: The Annotated Turing: A Guided Tour Through Alan Turing's Historic Paper on Computability and the Turing Machine. Wiley Publishing, New York (2008).
13. Turing, A. M.: Computing machinery and intelligence. Mind **59**(236), 433–460 (1950).
14. Fortnow, L., & Homer, S.: A short history of computational complexity. Bulletin of the EATCS **80**(01), 2003 (2003).
15. Kujala, T., Saariluoma, P.: Cognitive mimetics for designing intelligent technologies. Adv. Hum. Comput. Interact. (2018).
16. Saariluoma, P., Kujala, T., Karvonen, A., Ahonen M.: Cognitive mimetics - main ideas. In: Proceedings on the International Conference on Artificial Intelligence (ICAI), pp. 202–206. The Steering Committee of the World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp) (2018).
17. McCarthy, J., Minsky, M. L., Rochester, N., Shannon, C. E.: A proposal for the Dartmouth summer research project on artificial intelligence, august 31, 1955. *AI magazine*, *27*(4), 12-12 (2003).
18. Haugeland, J. (ed.): Mind Design II: Philosophy, Psychology, Artificial Intelligence. MIT Press, Cambridge (1997).
19. Haugeland, J.: The nature and plausibility of cognitivism. Behavioral and Brain Sciences **1**(2), 215–226 (1978).

20. Thagard, P.: Mind: Introduction to Cognitive Science. MIT Press, Cambridge, MA (2005).
21. Waldrop, M. M.: The Dream Machine: J.C.R. Licklider and the Revolution That Made Computing Personal. Viking Penguin, New York (2001).
22. Newell, A., Simon, H. A.: Human problem solving. Prentice-hall, Englewood Cliffs, NJ (1972).
23. Ohlsson, S.: The problems with problem solving: Reflections on the rise, current status, and possible future of a cognitive research paradigm. The Journal of Problem Solving 5(1) (2012).
24. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach. Pearson Education, London (2016).
25. Simon, H. A.: The Sciences of the Artificial. MIT Press, Cambridge, MA (2019).
26. Russell, S.: Rationality and intelligence: a brief update. In: Fundamental Issues of Artificial Intelligence, pp. 7–28. Springer, Cham (2016).
27. Goertzel, B., Pennachin, C.: Artificial General Intelligence. Springer-Verlag, Berlin (2007).
28. Craver, C. F.: When mechanistic models explain. Synthese 153(3), 355-376 (2006).
29. Bunge, M.: Matter and Mind: A Philosophical Inquiry (Vol. 287). Springer Science & Business Media, Cham (2010).
30. Wells, A. J.: Turing's analysis of computation and theories of cognitive architecture. Cognitive Science **22**(3), 269–294 (1998).
31. Shagrir, O.: The rise and fall of computational functionalism. In Ben-Menahem, Y. (ed.), Hilary Putnam, pp. 220–250. Cambridge University Press, Cambridge, UK (2005).
32. Putnam, H.: Mind and machines. Journal of Symbolic Logic **9**(1), 57–80 (1960).
33. Putnam, H.: The nature of mental states. In: W. H. Capitan, & D. Merrill, Art, Mind and Religion, pp. 1–223. Pittsburg University Press, Pittsburg (1967).
34. Polk, T. A., & Seifert, C. M.: Cognitive modeling. MIT Press (2002).
35. Dijkstra, E. W. On the cruelty of really teaching computing science. Communications of the ACM **32**(12), 1398–1404 (1989).
36. Dym, C. L., Brown, D. C.: Engineering Design: Representation and Reasoning. Cambridge University Press, Cambridge, UK (2012).
37. Eppinger, S., Ulrich, K.: Product Design and Development. McGraw-Hill, Singapore (1995).
38. Cross, N.: Engineering Design Methods: Strategies for Product Design. John Wiley & Sons, Hoboken (2021).
39. Pahl, G., Beitz, W.: Engineering Design: A Systematic Approach. Springer Science & Business Media, Cham (2013).
40. Gregor, S.: The nature of theory in information systems. MIS Quarterly **30**(3), 611–642 (2006).
41. Hanson, N. R.: Patterns of discovery: An inquiry into the conceptual foundations of science. Cambridge University Press, New York (1981).
42. Franssen, M.: The normativity of artefacts. Studies in History and Philosophy of Science Part A **37**(1), 42–57 (2006).
43. McCulloch, W. S., & Pitts, W.: A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics 5(4), 115-133 (1943).
44. Bush, V. As we may think. The Atlantic Monthly **176**(1), 101–108 (1945).
45. Engelbart, D. C.: Augmenting Human Intellect: A Conceptual Framework. Air Force Office of Scientific Research, Washington, DC (1962).
46. Hollnagel, E., Woods, D. D.: Joint cognitive systems: Foundations of cognitive systems engineering. CRC press (2005).

47. Visser, W.: Designing as construction of representations: A dynamic viewpoint in cognitive design research. Human–Computer Interaction 21(1), 103-152 (2006).
48. Coplien, J. O.: Software design patterns: Common questions and answers. The patterns handbook: Techniques, strategies, and applications 13, 311 (1998).
49. Floridi, L.: The method of levels of abstraction. Minds and machines 18(3), 303-329 (2008).
50. Gentner, D., Holyoak, K. J., Kokinov, B. N. (Eds.): The Analogical Mind: Perspectives from Cognitive Science. MIT Press, Cambridge, MA (2001).
51. Gentner, D.: Structure-mapping: a theoretical framework for analogy. Cognitive Science **7**, 155–170 (1983). doi:10.1207/s15516709cog0702_3
52. Visser, W.: Two functions of analogical reasoning in design: a cognitive-psychology approach. Design Studies **17**(4), 417–434 (1996).
53. Qian, L., Gero, J.: Function-behavior-structure paths and their role in analogy-based design. AIEDAM **10**(4), 289–312 (1996).
54. Moreno, D. P., Hernández, A. A., Yang, M. C., Otto, K. N., Hölttä-Otto, K., Linsey, J. S., Wood, K. L., Linden, A.: Fundamental studies in design-by-analogy: a focus on domain-knowledge experts and applications to transactional design problems. Design Studies **35**(3), 232–272 (2014).