Holographic Case-Based Reasoning

Devi Ganesan(✉) and Sutanu Chakraborti

Indian Institute of Technology Madras, Chennai 600036, TN, India
{gdevi,sutanuc}@cse.iitm.ac.in

Abstract. In this paper, we present a novel extension of CBR that allows cases to be more proactive at problem solving, by enriching case representations and facilitating richer interconnectedness between cases. We empirically study the improvements resulting from a holographic realization on experimental datasets. In addition to making CBR more cognitively appealing, the idea has the potential to lend itself as an elegant general CBR formalism of which diverse realizations of CBR can be viewed as instances.

Keywords: Case-based reasoning · Case base maintenance · Holonic cases · Holographic reasoner · Cognitive CBR · CBR formalism

1 Introduction

Case-Based Reasoning (CBR) is founded on the central premise of reusing past experiences to solve problems, and this is particularly effective in ill-defined domains, where sufficiently rich logical or mathematical models of the domain are unavailable. In a help desk domain where the goal is to answer user queries on malfunctioning of a software, no domain model of the software is available for model-based diagnosis, but logs of past episodes of problems solving can be exploited to build a CBR diagnosis system. Thus, one appeal of CBR is in its ability to reduce human (expert) effort needed to engineer rich top down domain knowledge. In this respect, CBR seems, on the surface, to share some commonalities with Machine Learning (ML) which uses bottom up methods, largely driven by induction, to acquire knowledge. Unsurprisingly, there is a growing trend in the CBR community to embrace state-of-the-art ML techniques, for instance those from the field of Deep Learning, to CBR. In reality, however, CBR is a problem solving paradigm, broad enough in its scope, to elegantly embrace both top down and bottom up approaches effectively to solve a problem in a given domain. We hold the view that to bring back CBR to the centre stage of AI, it is imperative to appreciate CBR as a paradigm closely driven by the problem specific to the domain under consideration, rather than as a toolkit (like a set of ML algorithms) that can be easily adapted to suit diverse problem needs but is distanced from the nuances of the actual problems being solved.

Dijkstra had once remarked: "Computer science should be called computing science for the same reason why surgery is not called knife science". In saying

I. Watson and R. Weber (Eds.): ICCBR 2020, LNAI 12311, pp. 144–159, 2020.
https://doi.org/10.1007/978-3-030-58342-2_10

so, he intended to point out the futility in trying to understand the solution technique (the knife) without a keen appreciation of the problem at hand (the patient anatomy). Machine Learning methods are analogues of knives that can dissect a wide range of problems, starting from very simple ones (apples) to very complex ones (a human patient). Their effectiveness depends on the extent to which its user is aware of the problem complexities. CBR, on the other hand is a paradigm for problem solving, for performing a surgery, which may use the knife of Machine Learning when appropriate, but may need several other tools as well. In particular, CBR critically relies on a top down model of the domain, that decides the representation of cases, and in particular, the knowledge containers required by the reasoner, viz. cases, vocabulary, similarity and adaptation knowledge [23].

Bottom up methods, such as Machine Learners that induce similarity knowledge from data accumulated over time, can feed into these knowledge containers and can be used effectively in many situations to alleviate knowledge engineering bottleneck. The way top down knowledge is traded off for bottom up knowledge, or vice versa, is a key design choice that differentiates CBR systems deployed till date in diverse domains. Knowledge rich domains (i.e. domains where the domain knowledge is readily available) may rely more on top down knowledge, while knowledge light domains rely more on bottom up learners to compensate for absence of rich domain knowledge [9]. Irrespective of the nature of domains, however, the design choice is critically guided by the need to minimize what we call the "representation gap": the information loss incurred by an expert in the process of recording his problem solving experiences in the CBR knowledge containers. The effectiveness of a CBR system in a given domain is critically dependent on how well this representation gap is bridged.

In this paper, we propose the concept of holographic CBR, that aims at bridging this representation gap by breaking free of certain presuppositions implicit in conventional CBR systems. One such presupposition is that cases are passive knowledge containers, and hence case addition or deletion does not affect the rest of the case base. Clearly, human memories are more interesting; the experience of encountering a new problem and solving it, not only adds this experience passively to our storehouse of experience, but can lead to a re-organization of the remaining set of experiences, as well. Holographic CBR is founded on the philosophy that cases can be made more proactive in problem solving by embedding in them a richer model of how they relate to the CBR system as a whole. In practical terms, it involves enriching the representation of cases; in particular, each case can have its own local similarity, adaptation and vocabulary knowledge, which it can use, in addition to shared knowledge containers, to refer to other cases and collaborate in order to arrive at a solution to the problem. We show that holographic CBR not only leads to more cognitive realizations of CBR, but also offers us a fresh perspective that allows us to picture conventional CBR, and a large class of CBR realizations reported in literature in specific domains, as special instances of holographic CBR systems.

In Sect. 2, we discuss the inspiration behind the holographic conception from disciplines as diverse as physics, biology and organization structures. In Sect. 3, we discuss basic ideas of holographic CBR. Section 4 illustrates the essential idea by way of two realizations of holographic CBR. In Sect. 5, we discuss how our work relates to other work in literature, and how it can be further extended. Section 6 summarizes our main conclusions.

2 Holographic Systems

The conventional view in neuroscience in the earlier part of the last century was that specific memories were confined to specific locations in the brain. This viewpoint was advocated, for example by Wilder Penfield, a Canadian neurosurgeon [17], who experimented by electrically stimulating various brain regions of epileptic patients. In the mid-nineties, there was a surprise in store for the neuroscience community, when Karl Lashley's three decades of research culminated in evidences contrary to Penfield's findings. Lashley had trained rats to run a maze, and then surgically removed portions of their brains, with the aim of completely removing the regions in their brains responsible for their maze running abilities [13]. Interestingly, he discovered that irrespective of the brain region that was removed, their memories refused to perish. Lashley was joined by Karl Pribram, who hypothesized that the only explanation of Lashley's findings would be that memories, instead of being localized at specific brain regions, were distributed throughout the brain [20]. Whatever was true with rats was also true with humans, in that patients with portions of the brain selectively removed did not have specific memories wiped out; rather they could hazily reconstruct most of what was known before the surgery. To quote Talbott [28], who provides an engaging account of Pribram's findings, "Individuals who had received head injuries in car collisions and other accidents never forgot half of their family, or half of a novel they had read". This phenomenon can be attributed to non-localized or holographic memories, where each component contains an imprint of the whole. The name "holographic" pervades study of complexity in diverse areas such as biology, physics and organizational systems. For example, holism [26] is a method of study which believes that the whole is greater than the sum of the parts; the term 'holon' [12] refers to a system that is both a whole and a part; a hierarchy of such self-regulating holons is called a holarchy [12].

We were tempted to explore if ideas of holographic systems can inspire the engineering of systems more adept at simulating aspects of cognition. The traditional view of CBR is analogous to that of Penfield's in neuroscience, in that the cases are treated as isolated pieces of knowledge that do not interact with each other. One fallout of such an assumption is in case base maintenance, where cases can be deleted from the case base, or fresh cases can be added, without affecting the rest of the case base. This is clearly inconsistent with cognitive findings on human memory, where a new experience is known to affect related memories in interesting ways that facilitate the creation of abstractions. Similarly, forgetting may not be localized to just one specific experience, but may result in the blurring out of a class of memories associated with the experience being lost. These observations gave rise to the design hypothesis that in a holographic model of CBR, the cases need not only be isolated passive pieces of knowledge but can be proactively interconnected with other cases in ways more interesting that explored by conventional CBR systems. It is through the interconnectedness of cases, that a model where the whole is greater than the sum of parts, can be realized.

3 Holographic CBR

Let us use an analogy to convey the essential idea behind holographic CBR. Consider three different settings.

Setting 1: Let us consider the case of a person X who attempts to float an organization to address requirements from client Y. X hires a set of employees with diverse skillsets to address the client needs. X also hires a project manager who acts as a mediator between the team members and Y. Y issues a query to the mediator, who facilitates interaction between the project members, and responds back to Y with a solution. The mediator has some coarse knowledge about the skills of the team members, which helps in directing client queries to one or more of them. The fine-grained knowledge of how best to get the problem solved, by collaborating with each other, rests with each team member. So a team member may receive a query from the mediator and choose to solve it; alternately, she may direct it to another member whom she reckons to be more appropriate for the job. In certain cases, the team member may like to get more clarity from the mediator regarding the client query, and in case the mediator is not sure herself, she may approach the client to get a clarification. In the course of interaction with team members, the mediator may update her knowledge of skills of the team members, and the team members keep enriching their knowledge of the organization as a whole. Such an evolution of the mediator, along with the team members, renders the system more competent in addressing subsequent queries.

Setting 2: This is a hypothetical variant of Setting 1, where X hires a mediator who has a complete knowledge of the skillsets of each team member. Given a client query, the mediator solves it by assembling inputs from her members. In comparison to Setting 1, the team members are passive, in that their role is limited to answering queries from the mediator. They do not collaborate with others, and have no knowledge either of the client query, or of the skillsets of others in their project team.

Setting 3: This is yet another variant of Setting 2, in which the mediator attempts to answer the client query, all by herself, on behalf of X. In case she is not able to do so, she requests X to hire another employee having certain skills. Employees thus progressively are added on demand, provided X agrees, given his budget constraints and his level of confidence in the mediator. Each employee, while being distinct to each other in terms of skills and competencies, is fully aware of his or her role in the broader context of the problem being solved.

In the context of a CBR system, the client Y is analogous to the user who presents a query to a CBR system, X is the designer of the CBR system, the

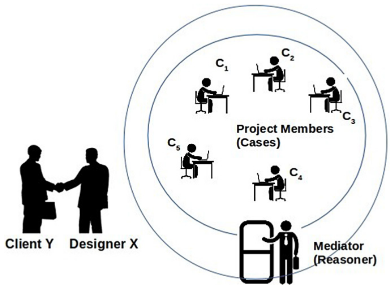

Fig. 1. Mediator analogy

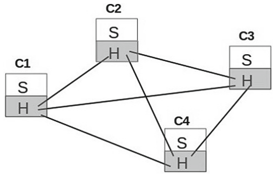

Fig. 2. Cases with Solo (S) and Holo (H) components

mediator is the case-based reasoner put in place by X to address the needs of Y, and the project members are the cases $C_1, C_2, .., C_5$. This analogy is shown in the schematic shown in Fig. 1. Setting 2 is the case of traditional CBR, where the mediator (the reasoner) has full access to knowledge of cases. The reasoner uses the knowledge of similarity to identify cases that may be useful, gets solutions from them, and combines these solutions using adaptation knowledge to answer the query posed by Y. Both similarity and adaptation knowledge are centralized and available exclusively to the reasoner. In contrast, Setting 3 is holographic, and Setting 1 is semi-holographic. We refer to the CBR systems in Settings 1, 2 and 3 as SH (for semi-holographic), TR (for traditional) and HG (for holographic) respectively.

In HG, the case base is grown on demand. Each case, in addition to storing a representation of the specific problem it solves, has knowledge of the reasoning goals as well as knowledge of how it relates to the other cases. SH can be conceived of striking a middle ground between the extremes of TR and HG, where the cases are richer than those in traditional case bases. Since cases only have local models of related cases, but are not equipped with the model of the case base as a whole, they are critically reliant on the reasoner (mediator) to dictate the retrieval process.

In Sect. 1, we had discussed that effectiveness of a CBR system can be improved by minimizing the information loss incurred by an expert in the process of recording his experiences of problem solving in the knowledge containers provided by CBR. It is clear that the loss is maximal in TR, and minimal in HG, with SH striking a middle ground. In HG, the cases have the highest autonomy in that each case has a reasonably good model of the goals of the CBR system, and also of the knowledge contained in every other case. We can visualize a spectrum of CBR applications ranging from TR to HG, through SH. As we move from TR to HG, the cases start having a richer representation of knowledge contained in other cases, as well as of the overall goals of problems solving. Henceforth, we shall use the term holographic to refer to systems that are either SH or HG. In a holographic setting, each case has two components which we call the solo component (referred to as the S component henceforth) and the holo component (referred to as the H component henceforth). The S component is the traditional problem-solution part and represents the individual experience that the case stands for. The H component, on the other hand embodies the essence of the proposed holographic setting, in that it defines the role of the case in relation to the case base and the underlying domain knowledge as a whole. We can picture the cases interacting with each other via their H components (see Fig. 2). Interestingly, such a holographic realization entails a change in our perspective of knowledge containers in CBR. The H component in cases facilitates localization of adaptation and similarity knowledge within each case; in other words, unlike in traditional CBR where knowledge containers other than cases are centralized, in a holographic setting, adaptation and similarity knowledge get distributed across the case base in holographic CBR. It may be noted that the scheme still allows for capturing aspects of domain knowledge that are shared by all cases, outside those in H components via the global knowledge resources possessed by the reasoner. Secondly, the H component of each case can capture diverse forms of relationships of a case with other cases in the case base. We envisage that the H component of each case can be used to capture how a case has been used, and its direct associations with other cases as well, so that any case maintenance operation would, no longer, be agnostic to this more general notion of ensuring case base competence. A schematic representation of a holographic reasoner for the problem of predicting animal names is given in Fig. 3. It is interesting to observe that both the reasoner and the cases have the same structure in a holographic reasoner. The reasoner holds the global knowledge containers and uses them to solve the larger problem of predicting the animal name given its representation. Each of the cases also hold the same kind of knowledge containers locally and, hence, can be called 'holonic'. These holonic cases use their local knowledge containers to solve the problem that they individually stand for. The problem part is pictorially depicted in the schematic diagram and it can correspond to any type of representation chosen by the case based designer for the problem part of experiences.

4 Realization of a Holographic Reasoner

In this section, we discuss the realization of a holographic reasoner in two settings: knowledge-rich and knowledge-light domains. A knowledge-rich domain is one where the domain knowledge is readily available. In practice, there are many domains where domain knowledge is not available readily or is costly to acquire in terms of time. We refer to such domains as knowledge-light domains.

Knowledge-Rich Domain. A key difference between a conventional and a holographic reasoner is with respect to the case addition process. In conventional settings, the reasoner is fully responsible for adding new cases to the case base. Whereas, in a holographic reasoner, this responsibility is shared among the cases. In the following pseudocode, the function `ADD_CASE` of `HOLOGRAPHIC_REASONER` describes how the case addition process varies between a knowledge-rich and knowledge-light settings. In a knowledge-rich setting, the holographic reasoner

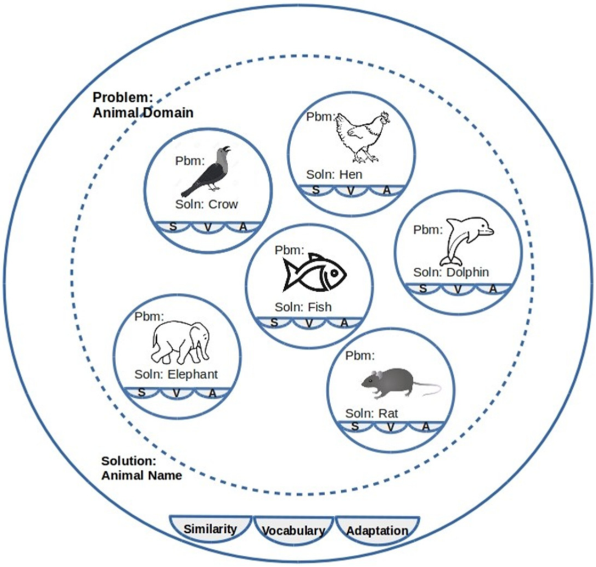

Fig. 3. Holographic Reasoner - A Schematic Diagram; S, V, A (in each case) stand for the local Similarity, Vocabulary and Adaptation knowledge containers respectively.

(mediator) uses the global similarity knowledge to direct an incoming case with its problem and solution components to its most similar case (a team member) in the case base. The global similarity knowledge, which can be shallow (coarse) compared to the local similarity knowledge in cases, enables the reasoner to quickly reach the relevant area of the problem space. Next, the most similar case spawns a case addition process that tries to predict a solution for the incoming problem, that is, it forms an expectation. This is explained by the functions `ADD_CASE, PREDICT` in the pseudocode for class `HOLONIC_CASE_KRICH`. If it faces an expectation failure, then it engages in a conversation with the domain expert. The expert feedback, together with a pointer to the new case, is stored as part of the local vocabulary as explained in the function `GET_EXPERT_FEEDBACK`. It is important to note that addition of a new case is performed by an existing case itself when there is an expectation failure. Thus, the responsibility for case addition lies not only with the reasoner but is also shared among the cases. On the other hand, if a case does not face an expectation failure, then it may choose not to do anything further or continue to add the new case to case base. This depends on constraints such as case base size, response time, etc. as known to the case base designer. In the `PREDICT` function, it is possible that the query gets redirected multiple number of times and it terminates only when a case finds itself to be the most similar one to the query.

Knowledge-Light Domain. In knowledge-light domains, a holographic reasoner does not have a domain expert to interact with. The knowledge-light setting is more like a conventional reasoner where the new cases are added to the case base as they arrive. Hence, the case acquisition process does not include any interaction with the domain expert. Instead, a holonic case could spawn a process to learn bottom up abstractions from their neighbourhood as shown by the function `INTROSPECT` in the class `HOLONIC_CASE_KLIGHT`. The H component of cases will now contain the parameters corresponding to the local model of abstraction, which could be any machine learning model such as Logistic Regression or Bayesian Classifier. This is like a human problem solver trying to learn something by observing their fellow problem solvers rather than asking the domain expert directly.

```
Class HOLOGRAPHIC_REASONER
 Vocabulary, Similarity, Adaptation, Case
      Base // Global Knowledge Containers
 Function ADD_CASE (newProblem, newSolution)
   If the domain is knowledge-rich:
     MostSimilarCase = RETRIEVE (newProblem)
     If MostSimilarCase is not null:
          MostSimilarCase.ADD_CASE (
               newProblem, newSolution)
     Else: //Adds the first case
          newCase = new HOLONIC_CASE_KRICH()
          newCase.Problem = newProblem
          newCase.Solution = newSolution
          GET_EXPERT_FEEDBACK (newCase)
          Store newCase in the CaseBase

   Else If the domain is knowledge-light:
        newCase = new HOLONIC_CASE_KLIGHT()
        newCase.Problem = newProblem
        newCase.Solution = newSolution
        Store newCase in the CaseBase

 Function RETRIEVE (incomingProblem)
   If CaseBase contains zero cases: Return
        null
   Else: Return the case in the CaseBase that
         is most similar to the
        incomingProblem according to the
        global Similarity knowledge

 Function PREDICT (incomingProblem)
   //The prediction process of the
        holographic reasoner   invokes the
        prediction process of the most
        similar case
   MostSimilarCase = RETRIEVE (
        incomingProblem)
   Return MostSimilarCase.PREDICT (
        incomingProblem)

------------------------------------------

Class HOLONIC_CASE_KRICH
 Problem, Solution //Solo Components
 Local Vocabulary, Local Similarity, Local
      Adaptation //Holo Components
 CaseBase // pointer to reasoner's case base
 Function ADD_CASE(newProblem, newSolution)
   If PREDICT(newProblem) matches Solution:
       //No Expectation Failure; No Case
            Addition
       Return null
   Else:
       newCase = new HOLONIC_CASE_KRICH()
       newCase.Problem = newProblem
       newCase.Solution = newSolution
       GET_EXPERT_FEEDBACK (newCase)
       Store newCase in the CaseBase

 Function PREDICT (incomingProblem)
   //MostSimilarCase is that case in the
        local neighbourhood (including self)
        which is most similar to the
        incomingProblem and is determined
        using the Local Vocabulary and Local
        Similarity knowledge.
   If MostSimilarCase is this holonic case
        itself: Return Solution
   Else: //Invokes the prediction process of
        the most similar case
       Return MostSimilarCase. PREDICT (
            incomingProblem)

 Function GET_EXPERT_FEEDBACK (newCase)
   Get feedback from a domain expert as to
        why the new case is being added, what
         feature-value pairs differentiate
        the new case from itself, etc. Update
         the Local Vocabulary to include the
        feedback and pointers to locally
        added cases.

------------------------------------------

Class HOLONIC_CASE_KLIGHT
 Problem, Solution //Solo Components
 Local Vocabulary, Local Similarity, Local
      Adaptation //Holo Components
 CaseBase //pointer to reasoner's case base
 Model // to store model parameters
 Function ADD_CASE(newProblem, newSolution)
   newCase = new HOLONIC_CASE_KLIGHT()
   newCase.Problem = newProblem
   newCase.Solution = newSolution
   Store newCase in the CaseBase

 Function PREDICT (incomingProblem)
   Predict a solution for the incomingProblem
         using Model
   Return the above prediction

 Function INTROSPECT ()
   //Invoked by reasoner (say after the case
        base reaches a certain size)
   Model = learn a model over the local
        neighbourhood , for example, a
        logistic regression model over the
        ten nearest neighbours
```

Observations from Experimental Datasets. Next, we present our observations on the characteristics of a holographic reasoner in the light of its realization on experimental datasets.

Knowledge-Rich Domain. The zoo case base from UCI repository [7] is an instance of a knowledge-rich domain and the nature of its domain (viz. animals) facilitates the authors themselves to play the role of a domain expert. It is a simple database containing 17 Boolean-valued attributes, 7 classes of animals and 101 data instances. On this case base, we realized both a holographic reasoner and a conventional case-based reasoner. In both the reasoners, global similarity knowledge was represented using the following two weight vectors: $S0$, a uniform weight vector and $S1$ emphasizing the attributes *feathers, aquatic, backbone, legs* three times over the rest. We did not employ any global or local adaptation knowledge. In the holographic setting, the expert gives her feedback using a list of entries where each entry is of the form `{feature_id:feature_value}`. For example, suppose the reasoner is adding a case *dolphin* to its case base and the most similar case is *dogfish*. Then, *dogfish* would face an expectation failure when it tries to predict the class of *dolphin* (as dolphin is a mammal). Expert feedback in this example could be `{milk_feeding:True}`. The holonic cases store the expert feedback together with a pointer to the newly created cases. The local vocabulary of a holonic case corresponds to those attributes used by an expert for giving feedback. Jaccard coefficient was used for estimating local similarity. In the conventional reasoner, we also found the footprint set [27], which is a minimal set of cases that has the same competence (problem-solving ability) as the entire case base. Competence based maintenance algorithms, such as the footprint algorithm, compress the case base in a post-facto way i.e. compression happens only after the experiences are stored. In terms of the representation gap, the damage is already done. In contrast, a holographic reasoner is capable of doing pre-facto compression i.e. it can compress the case base while adding the experience itself. While the post-facto compression relies purely on the cases to reduce the case base size, the pre-facto approach is able to acquire the knowledge enabling compression from the domain expert herself. This can facilitate the

Table 1. Observations on a knowledge-rich domain: zoo case base; $S0$: uniform weight vector and $S1$: weight vector that emphasizes *feathers, aquatic, backbone, legs* thrice over others. Results are based on 3-fold cross validation.

Reasoner type	Global similarity	No. of cases added	Case base size	Test accuracy %
Conventional (full CB)	$S1$	67.3	67.3	96.2
Conventional (footprint)	$S1$		13.6	94.3
Holographic	$S1$		**13.3**	**97.0**
Holographic	$S0$	67.3	11.3	94.3

reducing of knowledge gap between a reasoner and the domain expert. Table 1 shows the total number of cases added by the reasoner to its case base, the resulting case base size and the prediction accuracy. It can be observed from the table that the holographic reasoner performs best both in terms of case base compression and performance when the global similarity knowledge is $S1$. This can be attributed to the impact of domain knowledge acquired in the form of expert feedback. It can also be observed that when the global similarity knowledge is coarse ($S0$), the holographic reasoner is still able to achieve better compression and performance comparable to the footprint set. Hence, it makes it suitable for domains where one could not easily get a rich global similarity measure and may prefer to begin with a simple global similarity measure, progressively learning local similarities based on expert feedback.

Knowledge-Light Domains. The datasets used are CPU from OpenML [29] and Wine from UCI repository [7]. The CPU dataset contains 209 instances with 7 attributes and the task is to predict the relative cpu performance (regression). The Wine dataset contains a total of 178 cases with 13 attributes and 3 classes. The task is to predict the quality of wine given its attribute values (classification). For regression, the case-based reasoner (traditional as well as holographic) uses the distance-weighted average of the 3-nearest neighbours' predictions. Each holonic case learns a Locally Weighted Linear Regression (LWLR) [5] model over its neighbourhood. For classification, the holonic cases learn a naive Bayes classifier to model their local neighbourhood. The independence assumption in naive Bayes has an advantage for small-sized case bases because the algorithm is known to predict well even with small-sized training data. As elaborated in the previous sections, the H components of cases store the LWLR parameters and conditional probabilities in the regression and classification settings respectively. In our experiments, the size of the local neighbourhood is fixed empirically to be 10. It is important to note that knowledge-light holographic realizations in practice can be far more sophisticated in terms of richness of holonic case representations and processes they can spawn. The examples above use relatively simplistic ML tools to illustrate the essential idea. In particular, it is easy to see that the holographic perspective can accommodate richness in both top down and bottom up knowledge, hence most existing CBR systems can be viewed as instances of the general holographic CBR conception (see Sect. 5).

Here, we are interested in studying whether the global competence of cases increase in a holographic setting. In all our experiments in knowledge-light setting, the reasoner combines the solutions of the three nearest neighbours to solve the query problem. This process of combining the solutions of multiple cases in some appropriate way to solve the target problem is called compositional adaptation. Retention score [15] is a global competence measure suited for such scenarios and estimates the retention quality of a case based on its ability to cover highly retainable cases with the support of a few but highly retainable cases. This is achieved by a recursive formulation in the lines of PageRank [16]. We do not go into the details of this formulation but would like to emphasize the following fact: *retention scores can be used to order the cases in descending*

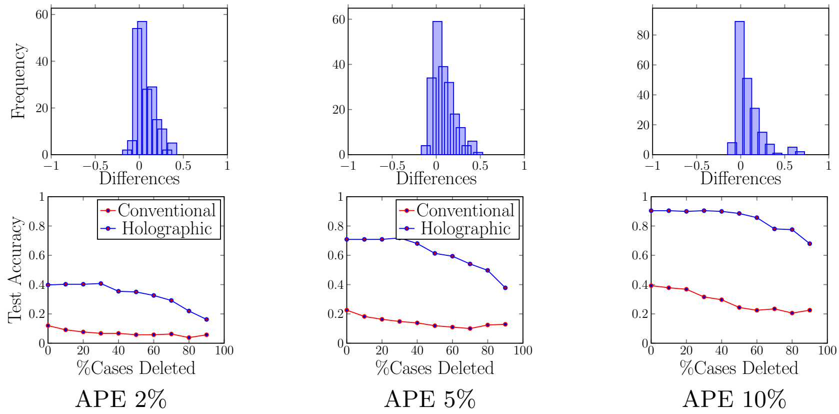

Fig. 4. Results on regression dataset (CPU); The top row shows the histograms of differences in retention scores (holographic − traditional) corresponding to the different settings of Acceptable Prediction Error (APE) shown in the bottom row.

Fig. 5. Results on classification dataset (Wine).

order of their global competence. In our experiments, we have used a variation of retention scores called *weighted retention scores* in which every set of cases that solves a target problem is weighed by its problem solving ability. After measuring the retention scores of cases in conventional and holographic settings, we plotted a *histogram of their differences* (holographic − traditional) to see if the differences are more skewed towards the positive side. This would indicate that a holographic design has resulted in an increase of competence for many cases. We also tested the effectiveness of the increased competence by progressively deleting the case base and observing its impact on the performance of reasoner on test data. We would expect a holographic reasoner to perform better than a conventional one even as the case base is progressively shrunk in size.

Figure 4 shows the results on the regression dataset. The top row shows the histograms for different settings of Acceptable Prediction Error (APE). APE is the percentage error allowed in the reasoner's predictions and is typically fixed by the user for the regression task. The more the right-skewedness, the better is the holographic design in terms of case competence. In the CPU dataset, as the histograms are skewed towards the right, it can be inferred that there is an increase in the case competence under the holographic design. Holographic design is consistently better than the conventional ones with increase in the progressive reduction of case base size. In Fig. 5, the histogram is skewed towards the right, hence, increased competence of cases in this holographic design becomes evident.

5 Discussion and Related Work

In this paper, we have restricted our scope to demonstrating the effect of holographic realizations on case addition, though in practice we need to have a mechanism for case deletion as well. We envisage two kinds of deletions: soft and hard. It is easy to see that holonic cases carry information about their local neighbourhood even after the neighbouring cases are deleted. We call this soft deletion. Though this increases the robustness of reasoner, in cases where we deliberately want to delete a (noisy) case, this may be undesirable, and a hard deletion is called for. In soft deletion, the H components of neighbours are retained, and is analogous to employees taking leave in a holarchic organization. In hard deletion, the H components of neighbours are updated before a case is deleted; this is analogous to handover-takeover processes in an organization, when an employee leaves the organization (is fired). Another interesting aspect not discussed in the paper is the impact of the order in which cases are acquired by a holographic reasoner. We can draw inspiration from how a child progressively acquires a storehouse of experience she encounters when systematically guided by an adult. Educational material for children aims at presenting experiences in an order that facilitates highest compression thereby improving the learning experience, where lessons are not merely recorded as facts, but are richly connected to each other. Reorganization of case interconnections over time to facilitate more effective retrieval is out of scope of this paper, though it opens up interesting area for future work.

In his work on Dynamic Memory [24], Roger Schank had emphasized the role of expectation failures in triggering the need for explanations and consequent generalization of memory structures. An event of visiting a restaurant like McDonald's where one has to pay before one eats, may lead to expectation failure for someone used to paying after eating in a restaurant. She would then attempt to find an explanation, generalize her memory structures and accommodate the new experience. This may involve creating a specific dimension (an attribute) that discriminates between the two categories of restaurants. Thus, while specific details of most restaurant trips are forgotten and abstracted out ("mushed up", to use Schank's terminology), some restaurant trips (like the McDonald's trip) are thus more influential than others in effecting changes to our memory structures. In the holographic setting, these changes that a case causes should be recorded in its H component during insertion, so that the influence of the case is preserved even when the case is deleted. Ideally, active processes must be spawned by the H component of cases as new cases are inserted, deleted or updated, to make changes to similarity and adaptation knowledge of related cases, facilitate case-to-case direct connections, or record and preserve influence of the case on the underlying representations. In the context of maintenance, H components can also potentially carry explanations pertaining to poorly aligned cases. The holographic setting can also accommodate bottom up knowledge induced from data in local similarity and adaptation knowledge containers in holonic cases to complement top down knowledge, thereby alleviating the knowledge acquisition bottleneck that plagued Schank's conceptualization limiting its practical use.

Several classical CBR systems can be thought of as instances of the more general holographic CBR framework. Aspects of it are ideologically close to the proposed holographic design for a knowledge rich domain. PROTOS [4] is a case-based reasoner built to serve as a learning apprentice system for heuristic based classification. It is interesting to see that many ideas in PROTOS such as difference links, efficient retrieval, expert feedback were aimed at overcoming limits of traditional CBR systems. In the early days of CBR, knowledge-rich reasoners such as CYRUS [25] and CELIA [21] were built to demonstrate the cognitive aspects of CBR. In CYRUS, which is an attempt to model the reconstructive model of memory, the cases are stored as hierarchically indexed facts. CELIA aims at modelling the passage from a novice to expert; the cases are composed of interconnected case snippets. Knowledge-intensive CBR systems like CREEK [1] reinforce the importance of integrating general domain knowledge with CBR systems and having rich knowledge representations. Some other interesting works to explore in this direction include the CREEK-based knowledge-intensive conversational CBR system [10] and Bayesian-Network powered CBR system [2]. The holographic perspective shows these as instantiations of the same umbrella framework, and is also suggestive of more proactivity on the part of cases that can be realized if the full potential of holographic CBR is exploited, by realizing richly interconnected cases that spawn active processes, and are empowered to influence H components of related cases, and generate explanations for failures.

Distributed CBR [19] is a terminology used in the CBR community to indicate research efforts towards organising knowledge in single versus multiple case bases and processing knowledge using single versus multiple agents. There are also many agent-based CBR approaches where knowledge is distributed such as [3,14,18,22] where the focus is on knowledge modelling, architecture and building of CBR based systems. Unlike domain specific engineering realizations such as distributed CBR, holographic systems are inspired differently: they are aimed at repositioning a broad spectrum of CBR applications (including distributed CBR systems) based on how they attempt to reduce the representation gap: all that is lost of the intent with which a case is being recorded, in the process of its representation. Such a repositioning has an essential cognitive appeal in that it helps us get to the heart of appreciating discrepancies in system effectiveness with respect to a human expert who solves problems using experiential reasoning. In future, it would be interesting to accommodate the study of analogical reasoning in a comprehensive way into the fold of holographic systems.

A related perspective is from the very recent work by Susan Craw et al. [6] where the authors present connections of CBR to cognitive models. In particular, the authors refer to the dichotomy between two modes of thought as identified by Kahneman [11]. While fast thinking relies on instinctive, unconscious, frequent and stereotypical decision making, slow thinking is more deliberative, conscious, logical and calculating. Slow thinking can correct errors made by fast thinking. In the CBR context, Craw et al. [6] suggest that simple retrieve/reuse may fall in the realm of fast thinking and this is appropriate when case base alignment is high, i.e. similar problems do indeed have similar solutions. On the other hand, in the face of poor alignment deliberate slow processes (say, complicated adaptation or multiple redirections) should intervene. It is compelling to picture the S and H components as facilitating fast and slow thinking respectively. Finally, we note that there are some recent claims that Deep Neural Networks (DNN) exhibit holographic behaviour [8]. However, there has been no understanding of the equivalents of holons and the organisational structure inside a DNN. DNNs do not facilitate the integration of top down knowledge about the domain, thus restricting their scope of applications in the context of CBR, where a problem-centric view, that allows for flexible integration of top down and bottom up is called for.

6 Conclusion

The historical roots of CBR can be traced to the seminal work by Roger Schank on dynamic memory [24] where he proposed mechanisms for creation and update of memory structures to account for abstraction, generalization, and goal based reminding (as in analogical reminding) which play a central role in modelling cognition. However, the cognitive emphasis in memory based reasoning waned over time. On occasions, machine learning techniques appeared to present easier alternatives to a principled mix of top down and bottom up knowledge that the CBR paradigm would ideally exploit reasoning based on representations, that are rich, and yet not too difficult to acquire to facilitate experiential problem solving. The concept of holographic reasoner is an attempt to bring back to perspective a wider set of possibilities than conventional CBR systems can offer, while showing its ability to position diverse CBR realizations in a unifying framework.

In living systems, every cell has in its nucleus (analogous to the H component) an imprint of the design of the organism as a whole. Not unlike the organism it is part of, every cell has a digestive, respiratory, nervous an immune system. This is remarkably different from a brick which is perhaps barely aware of the design of the building, of which it is a part. The design almost wholly resides in the mind of the designer. The difference between the ideal holonic case and the traditional case in CBR is one of that between the cell and the brick. As we foray into the ambitious realms of Artificial General Intelligence (AGI), we speculate holographic systems may well hold clues, if not answers, to design of computational models of cognition that can address certain limitations of traditional approaches.

References

1. Aamodt, A.: Knowledge-intensive case-based reasoning in CREEK. In: Funk, P., González Calero, P.A. (eds.) ECCBR 2004. LNCS (LNAI), vol. 3155, pp. 1–15. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-28631-8_1
2. Aamodt, A., Langseth, H.: Integrating Bayesian networks into knowledge-intensive CBR. In: AAAI Workshop on Case-Based Reasoning Integrations, pp. 1–6 (1998)
3. Bach, K., Reichle, M., Reichle-Schmehl, A., Althoff, K.D.: Implementing a coordination agent for modularised case bases. In: Proceedings of 13th UKCBR@ AI, pp. 1–12 (2008)
4. Bareiss, E.R., Porter, B.W., Wier, C.C.: Protos: an exemplar-based learning apprentice. In: Machine Learning, pp. 112–127. Elsevier (1990)
5. Cleveland, W.S., Devlin, S.J.: Locally weighted regression: an approach to regression analysis by local fitting. J. Am. Stat. Assoc. **83**(403), 596–610 (1988)
6. Craw, S., Aamodt, A.: Case based reasoning as a model for cognitive artificial intelligence. In: Cox, M.T., Funk, P., Begum, S. (eds.) ICCBR 2018. LNCS (LNAI), vol. 11156, pp. 62–77. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01081-2_5
7. Dheeru, D., Karra Taniskidou, E.: UCI machine learning repository (2017). http://archive.ics.uci.edu/ml
8. Gan, W.C., Shu, F.W.: Holography as deep learning. Int. J. Mod. Phys. D **26**(12), 1743020 (2017)
9. Ganesan, D., Chakraborti, S.: An empirical study of knowledge tradeoffs in case-based reasoning. In: Proceedings of IJCAI, pp. 1817–1823 (2018)
10. Gu, M., Aamodt, A.: A knowledge-intensive method for conversational CBR. In: Muñoz-Ávila, H., Ricci, F. (eds.) ICCBR 2005. LNCS (LNAI), vol. 3620, pp. 296–311. Springer, Heidelberg (2005). https://doi.org/10.1007/11536406_24
11. Kahneman, D.: Thinking, Fast and Slow. Macmillan, New York (2011)
12. Koestler, A.: The Ghost in the Machine. Macmillan, New York (1968)
13. Lashley, K.S.: In search of the engram. In: Physiological Mechanisms, p. 454 (1950)
14. Leake, D.B., Sooriamurthi, R.: When two case bases are better than one: exploiting multiple case bases. In: Aha, D.W., Watson, I. (eds.) ICCBR 2001. LNCS (LNAI), vol. 2080, pp. 321–335. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-44593-5_23
15. Mathew, D., Chakraborti, S.: A generalized case competence model for casebase maintenance. AI Commun. **30**(3–4), 295–309 (2017)
16. Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank citation ranking: bringing order to the web. Technical report, Stanford InfoLab (1999)
17. Penfield, W., Gage, L.: Cerebral localization of epileptic manifestations. Arch. Neurol. Psychiatry **30**(4), 709–727 (1933)
18. Pla, A., LóPez, B., Gay, P., Pous, C.: eXiT* CBR. v2: distributed case-based reasoning tool for medical prognosis. Decis. Supp. Syst. **54**(3), 1499–1510 (2013)
19. Plaza, E., McGinty, L.: Distributed case-based reasoning. Knowl. Eng. Rev. **20**(3), 261–265 (2005)
20. Pribram, K.H.: Brain and Perception: Holonomy and Structure in Figural Processing. Psychology Press, London (2013)
21. Redmond, M.: Distributed cases for case-based reasoning: facilitating use of multiple cases. In: Proceedings of AAAI, vol. 90, pp. 304–309 (1990)
22. Reichle, M., Bach, K., Althoff, K.D.: Knowledge engineering within the application-independent architecture seasalt. Int. J. Knowl. Eng. Data Mining **1**(3), 202–215 (2011)

23. Richter, M.M.: Knowledge Containers. Readings in Case-Based Reasoning. Morgan Kaufmann Publishers, Burlington (2003)
24. Schank, R.C.: Dynamic Memory: A Theory of Reminding and Learning in Computers and People, vol. 240. Cambridge University Press, Cambridge (1982)
25. Schank, R.C., Kolodner, J.L.: Retrieving information from an episodic memory or why computers' memories should be more like people's. In: Proceedings of IJCAI, vol. 2, pp. 766–768 (1979)
26. Smuts, J.C.: Holism and Evolution. Ripoll Classic, Moscow (1926)
27. Smyth, B., McKenna, E.: Competence models and the maintenance problem. Comput. Intell. **17**(2), 235–249 (2001)
28. Talbot, M.: The Holographic Universe. HarperCollins, New York (1991)
29. Vanschoren, J., Van Rijn, J.N., Bischl, B., Torgo, L.: OpenML: networked science in machine learning. ACM SIGKDD Explor. Newslett. **15**(2), 49–60 (2014)