Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence

# Case Adaptation with Qualitative Algebras *

**Valmi Dufour-Lussier**
LORIA
Université de Lorraine
CNRS UMR 7503, Inria
F-54506 Vandœuvre
*valmi.dufour@loria.fr*

**Florence Le Ber**
ICUBE
Université de Strasbourg
ENGEES, CNRS
F-67000 Strasbourg
*florence.leber@engees.unistra.fr*

**Jean Lieber**
LORIA
Université de Lorraine
CNRS UMR 7503, Inria
F-54506 Vandœuvre
*jean.lieber@loria.fr*

**Laura Martin**
INRA
UR055
ASTER–Mirecourt
F-88500 Mirecourt
*laura.martin@mirecourt.inra.fr*

## Abstract

This paper proposes an approach for the adaptation of spatial or temporal cases in a case-based reasoning system. Qualitative algebras are used as spatial and temporal knowledge representation languages. The intuition behind this adaptation approach is to apply a substitution and then repair potential inconsistencies, thanks to belief revision on qualitative algebras. A temporal example from the cooking domain is given.

## 1 Introduction

Case-based reasoning (CBR) [Riesbeck and Schank, 1989] is a framework in which a new problem (the target case) is solved by first retrieving an older, similar problem to which the solution is known (the source case), and then adapting this solution to fit the new problem. While the retrieval stage has been thoroughly studied by the CBR community, the adaptation stage has received less attention until recently. One proposal to address the adaptation problem is to apply a belief revision operator, revising source knowledge by target knowledge [Lieber, 2007]. In this paper, we apply Lieber's proposal to case knowledge represented using a qualitative algebra, such as Allen's calculus [Allen, 1983] or RCC8 [Randell *et al.*, 1992].

Qualitative spatial and temporal reasoning (QSTR) as a research domain has been active since the beginning of the 1980s. The paradigm has been exploited to help solve planning and constraint satisfaction problems, but rarely within CBR. Nevertheless, many domains in which QSTR is used could be addressed with CBR because the knowledge involved is usually contextual and incompletely formalised. This is the case in the domain of landscape agronomy, in which knowledge can be acquired from schematic descriptions of the spatial organisation of farmlands. Another example is the cooking domain, in which some knowledge is of a temporal nature.

In section 2, our approach is illustrated informally using a cooking example. Section 3 then introduces the formal notions required for the approach, namely in terms of CBR, revision-based adaptation, and QSTR. The approach is then defined in details in section 4, and an algorithm is described in section 5. Section 6 illustrates those formal notions and the results of the algorithm using the example introduced in section 2. Related work is discussed in section 7.

*The paper on which this extended abstract is based was the recipient of the best paper award of the 2012 International Conference on Case-Based Reasoning [Dufour-Lussier *et al.*, 2012].

## 2 A cooking example

To illustrate temporal case adaptation, we use TAAABLE [Cojan *et al.*, 2011], a CBR application for cooking. TAAABLE answers user queries, for instance: "I want a recipe for a carrot risotto." If no matching recipe is found in the cookbook (the case base), a recipe of the same type with similar ingredients will be retrieved, for instance a mushroom risotto. TAAABLE will then suggest the user replaces mushrooms with carrots. On the other hand, in its current form, it will not be able to help the user in adapting the recipe.

Suppose the mushrooms were added to the rice 2 minutes before the end, but the cooking domain knowledge indicates that carrots must be cooked for 25 minutes in order to be done, whereas the rice must be cooked for 18 minutes. A proper adaptation would require not only the lengthening of the cooking time of the vegetables, but also a reordering of the actions in the recipe. Therefore we expect the approach we will now introduce, given a retrieved recipe and a requested substitution, will be able to reorder the actions of the recipe in order to present TAAABLE users with a usable procedure.

## 3 Background

### 3.1 Case-based reasoning and case adaptation

In this paper, `Source`, `Target` and `DK` respectively denote the case to be adapted, the target case and the domain knowledge. `Source` and `Target` are required to be consistent with `DK`. Given `Source` and `Target`, the adaptation aims at building a new case, `AdaptedCase`. This case is built by adding some information to the target case (intuitively, `Target` specifies only the "problem part" of the query), and it has to be consistent with `DK`.

It is assumed that a matching step precedes the adaptation process, providing links between `Source` and `Target`. It is represented by a substitution $\sigma$, mapping descriptors of `Source` to descriptors of `Target`. As an example, in the system TAAABLE, matching is performed during retrieval. This process, applied to the cooking example of the previous section, would have returned $\sigma$ = `mushroom` $\leadsto$ `carrot`. In the following, this preprocessing step of adaptation is considered to be given and, thus, $\sigma$ is an input of the adaptation process described in section 4.

### 3.2 Belief revision and revision-based adaptation

In a given representation formalism, a revision operator $\dot{+}$ maps two knowledge bases $\psi$ and $\mu$ to a knowledge base $\psi \dot{+} \mu$, the revision of $\psi$ by $\mu$. Intuitively, $\psi \dot{+} \mu$ is obtained by making a minimal change of $\psi$ into $\psi'$, so that the conjunction of $\psi'$ and $\mu$, $\psi' \wedge \mu$, is consistent. Then, $\psi \dot{+} \mu$ is this conjunction.

The notion of minimal change can be modelled in various ways, so there are various revision operators. However, postulates have been proposed for such an operator, such as the AGM postulates [Alchourrón *et al.*, 1985]. These postulates have been applied to propositional logic [Katsuno and Mendelzon, 1991] and well studied in this formalism. Given a distance `dist` on the set $\mathcal{U}$ of the interpretations, an operator $\dot{+}_{\texttt{dist}}$ can be uniquely defined (up to logical equivalence) as: the set of models of $\psi \dot{+}_{\texttt{dist}} \mu$ is the set of models of $\mu$ that have a minimal distance to the set of models of $\psi$.

Given a revision operator $\dot{+}$, $\dot{+}$-adaptation consists simply in using this revision operator to perform adaptation, taking into account the domain knowledge:

$$\texttt{AdaptedCase} = (\texttt{DK} \wedge \texttt{Source}) \dot{+} (\texttt{DK} \wedge \texttt{Target}) \quad (1)$$

The intuition behind revision-based adaptation is to reduce adaptation to an inconsistency repair.

### 3.3 Qualitative representation of temporal knowledge

#### Definitions

A qualitative algebra is a relation algebra that defines a set $\mathfrak{B}$ of binary relations applicable between two variables, usually representing points, intervals or regions. Allen interval algebra [Allen, 1983], for instance, introduces 13 basic relations between intervals, corresponding to the 13 possible arrangements of their lower and upper bounds. 7 relations are illustrated in figure 1. The 6 others are the inverse of the first 6 (eq is symmetric).

$\mathcal{INDU}$ [Pujari *et al.*, 1999] extends the set of Allen relations by combining them with relations over the interval durations. For 7 Allen relations, there is only one possible duration relation (e.g. $i$ {d} $j$ implies that the duration of $i$ is shorter than the duration of $j$). For the other 6, all three duration relations $<$, $=$ and $>$ are possible. This yields a total of 25 basic relations. They are written as $r^s$, where $r$ is an Allen relation and $s$ is a duration relation.

Qualitative knowledge can be represented as qualitative constraint networks (QCNs). A QCN is a pair $(V, C)$, where $V$ is a set of variables, and $C$ is a set of constraints of the form $V_i\ C_{ij}\ V_j$ with $V_i, V_j \in V$, and $C_{ij}$ is a set of the basic relations defined by the algebra ($C_{ij}$ is a relation that is a disjunction of the basic relations, i.e. $i\ \{r_1, r_2\}\ j$ means that $i$ is related to $j$ with either $r_1$ or $r_2$). In $\mathcal{INDU}$, shortcut notations $r^?$ and $?^s$ respectively represent the Cartesian product of $r$ and all possible duration relations and the product of $s$ and all possible Allen relations (e.g., $\{\text{m}\}^? = \{\text{m}^<, \text{m}^=, \text{m}^>\}$; $\{\text{d}\}^? = \{\text{d}^<\}$; $\{?\}^= = \{\text{b}^=, \text{m}^=, \text{o}^=, \text{eq}^=, \text{oi}^=, \text{mi}^=, \text{bi}^=\}$).

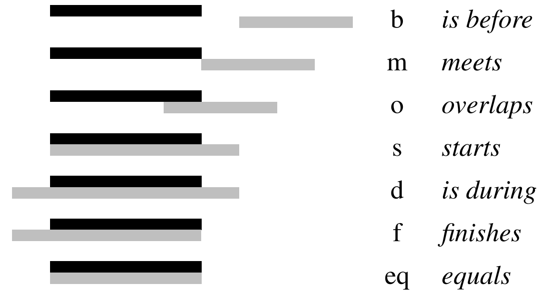


Figure 1: Allen interval algebra basic relations.

A *scenario* is a QCN $\mathcal{S} = (V_\mathcal{S}, C_\mathcal{S})$ such that for each $V_i, V_j \in V_\mathcal{S}$, there exists one constraint $V_i\ \{r\}\ V_j \in C_\mathcal{S}$. $\mathcal{S}$ satisfies the QCN $\mathcal{N} = (V_\mathcal{N}, C_\mathcal{N})$ if $\mathcal{S}$ and $\mathcal{N}$ have the same set of variables and each constraint relation in $\mathcal{S}$ is a subset of the corresponding constraint relation in $\mathcal{N}$. A scenario is consistent if a valuation can be provided for the variables such that all constraints are observed, and a QCN is consistent if it has a consistent scenario. Two QCNs are said to be equivalent if every scenario of the former is a scenario of the latter and vice-versa.

#### Revision of QCNs

A QCN is a knowledge base and thus, the issue of revising a QCN $\psi$ by a QCN $\mu$ can be addressed. Building on the work of [Condotta *et al.*, 2008], we defined a revision operator for QCNs, following the idea of an operator $\dot{+}_{\texttt{dist}}$ (cf. section 3.2), where an interpretation is a scenario, a model of a QCN is a scenario that satisfies it, and a distance `dist` between scenarios/interpretations is defined as follows.

First, a distance $d$ between basic relations of the considered algebra is defined. Formally, a *neighbourhood graph* whose vertices are the relations of the algebra is given, and $d(r, s)$ is the distance between $r$ and $s$ in the graph. It represents closeness between relations. For instance, b and m are close ($d(\text{b}, \text{m}) = 1$) since they express similar conditions on the boundaries of the intervals (for the lower bounds: $=$ for both; for the upper bounds: $<$ for b and $=$ for m). $d$ makes it possible to define `dist`, a distance between two scenarios $\mathcal{S} = (V, C_\mathcal{S})$ and $\mathcal{T} = (V, C_\mathcal{T})$ based on the same set of variables $V$, as:

$$\texttt{dist}(\mathcal{S}, \mathcal{T}) = \sum_{V_i, V_j \in V, i \neq j} d(r_\mathcal{S}(V_i, V_j), r_\mathcal{T}(V_i, V_j)) \quad (2)$$

where $r_\mathcal{S}(V_i, V_j)$ is the relation $r$ such that $V_i\ \{r\}\ V_j \in C_\mathcal{S}$.

Given two QCNs $\psi$ and $\mu$, the revision of $\psi$ by $\mu$ returns the set $R$ of scenarios satisfying $\mu$ that are the closest ones to the set of scenarios satisfying $\psi$.[1]

[1] This slightly differs from the definition of revision given in section 3.2 where $\psi \dot{+} \mu$ is a knowledge base, not a set of models.

# 4 Formalisation

## 4.1 Representation of the adaptation problem

**Parametrised QCNs**

It is assumed that the variables of the considered QCNs can be parametrised by elements of a given set $\mathcal{P}$. A parameter $p \in \mathcal{P}$ is either a *concrete parameter*, $p \in \mathcal{CP}$, or an *abstract parameter*, $p \in \mathcal{AP}$: $\mathcal{P} = \mathcal{CP} \cup \mathcal{AP}$, $\mathcal{CP} \cap \mathcal{AP} = \emptyset$. A concrete parameter denotes a concept of the application domain, e.g. $\texttt{mushroom} \in \mathcal{CP}$ for the cooking example. In this example, the formal interval $\texttt{cooking}(\texttt{mushroom})$ represents the temporal interval of the mushroom cooking. The domain knowledge $\texttt{DK} = (V_{\text{DK}}, C_{\text{DK}})$ is a set of constraints, for example:

$$C_{\text{DK}} = \left\{ \begin{array}{l} \texttt{cooking(rice)} \overset{?}{=} \texttt{18\_min} \\ \texttt{cooking}(x)\ \{\text{m}\}^{?}\ \texttt{cooked}(x) \\ \texttt{18\_min}\ ?^{<}\ \texttt{25\_min} \end{array} \right\}$$

where $\texttt{rice} \in \mathcal{CP}$ and $x \in \mathcal{AP}$, represents the facts that rice requires 18 minutes of cooking, that $x$ is cooked as soon as the action of cooking $x$ is finished, and that 18 minutes are shorter than 25 minutes. An abstract parameter must be understood with a universal quantification over the concrete parameters; e.g. $\texttt{cooking}(x)\ \{\text{m}\}^{?}\ \texttt{cooked}(x)$ entails $\texttt{cooking(mushroom)}\ \{\text{m}\}^{?}\ \texttt{cooked(mushroom)}$.

Let $\mathcal{N}_1$ and $\mathcal{N}_2$ be two QCNs. $\mathcal{N}_1 \wedge \mathcal{N}_2$ is the QCN $\mathcal{N} = (V, C)$ such that $V = V_1 \cup V_2$ and $C$ contains the constraints of $C_1$, the constraints of $C_2$, and the constraints that are deduced by instantiation of the abstract parameters by concrete parameters appearing in $\mathcal{N}_1$ and $\mathcal{N}_2$ For example, if $\mathcal{N}_1 = C_{\text{DK}}$ defined by equation (4.1) and $\mathcal{N}_2 = (\{\texttt{cooking(tomato)}, \texttt{cooked(tomato)}\}, \emptyset)$, then $\mathcal{N}_1 \wedge \mathcal{N}_2 = (V, C)$ with $C = C_{\text{DK}} \cup \{\texttt{cooking(tomato)}\ \{\text{m}\}^{?}\ \texttt{cooked(tomato)}\}$.

**Substitutions**

The *atomic substitution* $\sigma = p \rightsquigarrow q$, where $p, q \in \mathcal{P}$, is the function from $\mathcal{P}$ to $\mathcal{P}$ defined by $\sigma(a) = \begin{cases} q & \text{if } a = p \\ a & \text{otherwise} \end{cases}$. A *substitution* is a composition $\sigma_1 ; \ldots ; \sigma_n$ of atomic substitutions $\sigma_i$.

Let $\sigma = p \rightsquigarrow q$ be an atomic substitution. $\sigma$ is *concrete* if $p, q \in \mathcal{CP}$. $\sigma$ is an *atomic abstraction* if $p \in \mathcal{CP}$ and $q \in \mathcal{AP}$. $\sigma$ is an *atomic refinement* if $p \in \mathcal{AP}$ and $q \in \mathcal{CP}$. A *concrete substitution* (resp., an abstraction, a refinement) is a composition of concrete atomic substitutions (resp., of atomic abstractions, of atomic refinements). Any concrete substitution $\sigma$ can be written $\sigma = \alpha ; \varrho$ where $\alpha$ is an abstraction and $\varrho$ is a refinement, as the following equation illustrates:

$$\texttt{mushroom} \rightsquigarrow \texttt{carrot} = \texttt{mushroom} \rightsquigarrow x ; x \rightsquigarrow \texttt{carrot}$$

where $\texttt{mushroom}, \texttt{carrot} \in \mathcal{CP}$ and $x \in \mathcal{AP}$. This can be shown as follows. First, $\sigma$ can be written $p_1 \rightsquigarrow q_1 ; \ldots ; p_n \rightsquigarrow q_n$ with $p_i, q_i \in \mathcal{CP}$ and $p_i \neq p_j$ if $i \neq j$. Let $x_1, \ldots, x_n$ be $n$ abstract parameters, let $\alpha_i = p_i \rightsquigarrow x_i$, let $\varrho_i = x_i \rightsquigarrow q_i$, let $\alpha = \alpha_1 ; \ldots ; \alpha_n$, and let $\varrho = \varrho_1 ; \ldots ; \varrho_n$. $\alpha$ is an abstraction, $\varrho$ is a refinement and $\sigma = \alpha ; \varrho$.

Let $\sigma$ be a substitution. $\sigma$ is extended on qualitative variables by applying it to their parameters. For example, if $\sigma = \texttt{mushroom} \rightsquigarrow \texttt{carrot}$ then $\sigma(\texttt{cooking(mushroom)}) = \texttt{cooking(carrot)}$. Then, $\sigma$ is extended to a constraint $c = (V_i\ C_{ij}\ V_j)$ by $\sigma(c) = (\sigma(V_i)\ C_{ij}\ \sigma(V_j))$. Finally, $\sigma$ is extended on a QCN by applying it to its variables and constraints: $\sigma((V, C)) = (\sigma(V), \sigma(C))$ where $\sigma(V) = \{\sigma(V_i) \mid V_i \in V\}$ and $\sigma(C) = \{\sigma(c) \mid c \in C\}$.

**Adaptation problem**

An adaptation problem is given by a tuple $(\texttt{Source}, \texttt{Target}, \texttt{DK}, \sigma)$. $\texttt{Source}$ and $\texttt{Target}$ are the representations of the source and target cases by QCNs with concrete variables (i.e. not parametrised by any abstract parameter). $\texttt{DK}$ is a QCN representing the domain knowledge. $\sigma = p_1 \rightsquigarrow q_1 ; \ldots ; p_n \rightsquigarrow q_n$ is a concrete substitution such that each $p_i$ (resp., $q_i$) parametrises a variable of $\texttt{Source}$ (resp., $\texttt{Target}$). $\texttt{DK} \wedge \texttt{Source}$ and $\texttt{DK} \wedge \texttt{Target}$ are assumed to be consistent (cf. section 3.1). The goal of adaptation is to build a consistent QCN $\texttt{AdaptedCase}$ that entails $\texttt{DK} \wedge \texttt{Target}$, whose qualitative variables are obtained by applying $\sigma$ on the qualitative variables of $\texttt{Source}$, and that is obtained thanks to minimal modification of $\texttt{DK} \wedge \texttt{Source}$.

## 4.2 Principles of revision-based adaptation of a QCN

A first idea to perform the adaptation, given a tuple $(\texttt{Source}, \texttt{Target}, \texttt{DK}, \sigma)$, is to apply $\sigma$ on $\texttt{Source}$, thus obtaining a QCN $\texttt{DK} \wedge \sigma(\texttt{Source})$ that may be inconsistent, and then restoring consistency. Although this gives a good intuition of the revision-based adaptation of a QCN, it is not consistent with the irrelevance of syntax principle. (Indeed, any two inconsistent knowledge bases are equivalent: their sets of models are both empty.) Thus, at a semantic level, repairing an inconsistent knowledge base is meaningless. By contrast, revision aims at modifying a *consistent* knowledge base with another *consistent* one, the conjunction of which may be inconsistent.

Therefore, the revision-based adaptation consists first in decomposing $\sigma$ in an abstraction $\alpha$ and a refinement $\varrho$: $\sigma = \alpha ; \varrho$. Then, $\alpha$ is applied to $\texttt{Source}$: a QCN $\texttt{DK} \wedge \alpha(\texttt{Source})$ is built that is necessarily consistent since $\texttt{DK} \wedge \texttt{Source}$ is consistent and every constraint of $\texttt{DK} \wedge \alpha(\texttt{Source})$ corresponds to a constraint of $\texttt{DK} \wedge \texttt{Source}$. In other words, $\texttt{DK} \wedge \texttt{Source}$ is consistent and is more or equally constrained as $\texttt{DK} \wedge \alpha(\texttt{Source})$, so $\texttt{DK} \wedge \alpha(\texttt{Source})$ is consistent.

The third step involves revision. The idea is to make a revision of $\psi$ by $\mu$ where $\psi = \texttt{DK} \wedge \alpha(\texttt{Source})$ and $\mu = \texttt{DK} \wedge \texttt{Target} \wedge \mathcal{N}_\varrho$ where $\mathcal{N}_\varrho$ represents the following statement: "Each qualitative variable $V_i$ of $\alpha(\texttt{Source})$ is constrained to be equal to its refinement $\varrho(V_i)$." For this purpose, the relation *eq* for equality is used (eq$^{=}$ in $\mathcal{INDU}$): $V_i\ eq\ \varrho(V_i)$. Therefore, $\mathcal{N}_\varrho = (V_\varrho, C_\varrho)$ where

$$V_\varrho = \alpha(V) \cup \sigma(V) \quad ; \quad C_\varrho = \{V_i\ eq\ \varrho(V_i) \mid V_i \in \alpha(V)\}$$

$\mu$ is consistent since $\texttt{DK} \wedge \texttt{Target}$ is and since each constraint $V_i\ eq\ \varrho(V_i)$ of $\mathcal{N}_\varrho$ either is a tautology (when $V_i$ does not contain any abstract parameter refined by $\varrho$) or links a variable $V_i$ that does not appear in $\texttt{DK} \wedge \texttt{Target}$ with $\varrho(V_i)$.

Then, $\psi \dotplus \mu$ gives a set of scenarios and AdaptedCase is chosen among them.

## 5 Algorithm and implementation

The revision algorithm takes as input $\psi = \texttt{DK} \wedge \alpha(\texttt{Source})$, $\mu = \texttt{DK} \wedge \texttt{Target} \wedge \mathcal{N}_\varrho$, as well as a relation neighbourhood graph and a transitivity table for the algebra used. The neighbourhood graph enables to define a distance $d$ between relations and the transitivity table defines a relation composition function $\circ : \mathfrak{B} \times \mathfrak{B} \to 2^{\mathfrak{B}}$, for example, $\text{m} \circ \text{mi} = \{\text{eq, f, fi}\}$ in Allen algebra.

First, it is necessary to ensure that all variables in either QCN are present in the other QCN as well. All pairs of variables that have no relation associated to them are given the relation $\mathfrak{B}$–the unspecified relation.

The algorithm must then search within the scenarios of $\mu$ the ones that minimise the distance to $\psi$. The distance between the QCNs $\psi$ and $\mu$ is the smallest distance between any scenario of $\psi$ and a scenario $\mu$, computed using equation (2). Considering that the minimum of sums is never less than the sum of minimums, a lower bound on the distance between two QCNs can be obtained in time $O(|V|^2 \cdot |\mathfrak{B}|^2)$ by computing the pair-wise minimal distance for each constraint and summing those. This defines an admissible heuristic which is used to instantiate an A$^*$search. The initial state is $\mu$ and a goal state is a scenario of $\mu$. A successor state is obtained by selecting one constraint and keeping only one relation on this constraint. The QCN $\psi$ is used in the cost in the heuristic functions.

The amount of scenarios for a QCN is of the order of $O\left(|\mathfrak{B}|^{\frac{|V|\cdot(|V|-1)}{2}}\right)$.

## 6 Result

This section revisits the example from section 2.

Most temporal aspects of recipes can be represented in $\mathcal{INDU}$ by reifying cooking actions, ingredient states, and durations as intervals. For instance, the following could be included in the domain knowledge: $\texttt{cooking(carrot)}\ \{\text{m}\}^?\ \texttt{cooked(carrot)}$ and $\texttt{cooking(carrot)}\ ?^=\ \texttt{25\_min}$, with the provision that, e.g. $\texttt{18\_min}\ ?^<\ \texttt{25\_min}$.

To limit the amount of variables shown, we simplify the representation by replacing duration intervals with duration relations between the relevant action intervals. In this representation, $\psi$ contains

$$C_{\texttt{DK}} = \left\{ \begin{array}{l} \texttt{cooking(rice)}\ ?^<\ \texttt{cooking(carrot)} \\ \texttt{cooking(rice)}\ \{\text{m}\}^?\ \texttt{serve} \\ \texttt{cooking(carrot)}\ \{\text{m}\}^?\ \texttt{serve} \end{array} \right\}$$
$$C_{\alpha(\texttt{Source})} = \left\{ \texttt{cooking}(x)\{\text{f}^<\}\texttt{cooking(rice)} \right\}$$

In TAAABLE, there is no firm adaptation constraint from Target ($C_{\texttt{Target}} = \emptyset$) therefore $\mu$ contains simply the constraints

$$C_{\texttt{DK}} = \left\{ \begin{array}{l} \texttt{cooking(rice)}\ ?^<\ \texttt{cooking(carrot)} \\ \texttt{cooking(rice)}\ \{\text{m}\}^?\ \texttt{serve} \\ \texttt{cooking(carrot)}\ \{\text{m}\}^?\ \texttt{serve} \end{array} \right\}$$
$$C_\varrho = \{\texttt{cooking}(x)\ ?^=\ \texttt{cooking(carrot)}\}$$

The revision algorithm returns two scenarios which are predictably distinguished only by the duration relation between serve and the other actions, since this relation is defined as being unimportant in the domain knowledge. One scenario $\mathcal{T} = (V_\mathcal{T}, C_\mathcal{T})$ is such that $C_\mathcal{T}$ is

$$\left\{ \begin{array}{l} \texttt{cooking}(x)\{\text{m}^>\}\texttt{serve} \\ \texttt{cooking(carrot)}\{\text{m}^>\}\texttt{serve}, \\ \texttt{cooking(rice)}\{\text{m}^>\}\texttt{serve} \\ \texttt{cooking}(x)\{\text{eq}^=\}\texttt{cooking(carrot)} \\ \texttt{cooking}(x)\{\text{fi}^>\}\texttt{cooking(rice)} \\ \texttt{cooking(carrot)}\{\text{fi}^>\}\texttt{cooking(rice)} \end{array} \right\}$$

In both scenarios, the lengthening of the vegetable cooking is associated with the inversion of the relation between the vegetable and the rice, i.e. $\text{f}^<$ becomes $\text{fi}^>$, which corresponds to the expected order inversion between the start of both actions. Therefore, the adaptation is successful.

## 7 Related work

Several research work focused on the representation of time within the CBR framework. Most were interested in the analysis or in the prediction of temporal processes (e.g. breakdown or disease diagnosis starting from regular observations or successive events). The temporal aspect is generally taken into account from sequences of events or sometimes from relative or absolute time stamps [Dojat *et al.*, 1998; Ma and Knight, 2003; Sánchez-Marré *et al.*, 2005]. Particularly, the problem of temporal adaptation has been given much attention in CBR with a workflow representation [Minor *et al.*, 2010]. Only a few work [Jaczynski and Trousse, 1998; Dørum Jære *et al.*, 2002] adopted a qualitative representation. In [Dørum Jære *et al.*, 2002], cases are represented by temporal graphs and the retrieval step is based on graph matching. In [Jaczynski and Trousse, 1998], cases are indexed by chronicles and temporal constraints, which are represented with a subset of Allen relations.

Some recent work also dealt with a combination of CBR and spatial reasoning, for instance in order to improve web services for spatial information [Osman *et al.*, 2006], or for spatial event prediction in hostile territories [Li *et al.*, 2009].

## 8 Conclusion

Qualitative algebras are important to the field of knowledge representation and are especially useful for qualitative reasoning on space and on time, but their use in CBR has received very little attention so far. This paper focuses on the adaptation of cases represented in a qualitative algebra. A cooking example uses the temporal algebra $\mathcal{INDU}$. This adaptation uses the principles of revision-based adaptation and combines it with a matching between the source and target cases.

A prototype for adaptation of cases represented in a qualitative algebra has been implemented in Perl[2] and applied to the examples of this paper, but it is time-consuming and still requires improvements in order to be integrated into an operational system like TAAABLE.

[2]The Perl library and Java bindings for this and other revision tools are available at http://revisor.loria.fr.

## References


[Alchourrón *et al.*, 1985] C. E. Alchourrón, P. Gärdenfors, and D. Makinson. On the Logic of Theory Change: partial meet functions for contraction and revision. *Journal of Symbolic Logic*, 50:510–530, 1985.

[Allen, 1983] James F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, November 1983.

[Cojan *et al.*, 2011] J. Cojan, V. Dufour-Lussier, E. Gaillard, J. Lieber, E. Nauer, and Y. Toussaint. Knowledge extraction for improving case retrieval and recipe adaptation. In *Computer Cooking Contest Workshop*, 2011.

[Condotta *et al.*, 2008] Jean-François Condotta, Souhila Kaci, and Nicolas Schwind. A framework for merging qualitative constraints networks. In David Wilson and H. Chad Lane, editors, *FLAIRS Conference*, pages 586–591. AAAI Press, 2008.

[Dojat *et al.*, 1998] M. Dojat, N. Ramaux, and D. Fontaine. Scenario recognition for temporal reasoning in medical domains. *Artificial Intelligence in Medicine*, 14:139–155, 1998.

[Dørum Jære *et al.*, 2002] M. Dørum Jære, A. Aamodt, and P. Skalle. Representing temporal knowledge for case-based prediction. In *Advances in Case-Based Reasoning, Proceedings of ECCBR 2002*, volume 2416 of *LNAI*, pages 174–188. Springer, 2002.

[Dufour-Lussier *et al.*, 2012] Valmi Dufour-Lussier, Jean Lieber, Florence Le Ber, and Laura Martin. Adapting spatial and temporal cases. In *Case-Based Reasoning Research and Development (ICCBR 2012)*, volume 7466 of *LNAI*, pages 77–91. Springer, 2012.

[Jaczynski and Trousse, 1998] M. Jaczynski and B. Trousse. WWW assisted browsing by reusing past navigations of a group of users. In *Advances in Case-Based Reasoning*, pages 160–171. Springer, 1998.

[Katsuno and Mendelzon, 1991] H. Katsuno and A. Mendelzon. Propositional knowledge base revision and minimal change. *Artificial Intelligence*, 52(3):263–294, 1991.

[Li *et al.*, 2009] H. Li, H. Muñoz-Avila, D. Bransen, C. Hogg, and R. Alonso. Spatial event prediction by combining value function approximation and case-based reasoning. In *Case-Based Reasoning Research and Development (ICCBR 2009)*, volume 5650 of *LNAI*, pages 465–478. Springer, 2009.

[Lieber, 2007] Jean Lieber. Application of the revision theory to adaptation in case-based reasoning: The conservative adaptation. In Rosina Weber and Michael Richter, editors, *Case-Based Reasoning Research and Development*, volume 4626 of *LNCS*, pages 239–253. Springer, 2007.

[Ma and Knight, 2003] J. Ma and B. Knight. A Framework for Historical Case-Based Reasoning. In *Case-Based Reasoning Research and Development*, pages 246–260. Springer, 2003.

[Minor *et al.*, 2010] Mirjam Minor, Ralph Bergmann, Sebastian Görg, and Kirstin Walter. Towards case-based adaptation of workflows. In *Case-Based Reasoning Research and Development (ICCBR 2010)*, volume 6176 of *LNAI*, pages 421–435. Springer, 2010.

[Osman *et al.*, 2006] T. Osman, D. Thakker, Y. Yang, and C. Claramunt. Semantic spatial web services with case-based reasoning. In *Web and Wireless Geographical Information Systems, Proceedings of the 6th International Symposium, W2GIS 2006*, volume 4295 of *LNCS*, pages 247–258. Springer, 2006.

[Pujari *et al.*, 1999] Arun K. Pujari, G. Vijaya Kumari, and Abdul Sattar. INDU: An interval & duration network. *Advanced Topics in Artificial Intelligence*, pages 291–303, 1999.

[Randell *et al.*, 1992] David A. Randell, Zhan Cui, and Anthony G. Cohn. A spatial logic based on regions and connection. In *Knowledge Representation*, pages 165–176, 1992.

[Riesbeck and Schank, 1989] Christopher K. Riesbeck and Roger C. Schank. *Inside Case-based Reasoning*. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1989.

[Sánchez-Marré *et al.*, 2005] M. Sánchez-Marré, U. Cortés, M. Martínez, J. Comas, and I. Rodríguez-Roda. An approach for temporal case-based reasoning: Episode-based reasoning. In *International Conference on Case-Based Reasoning (ICCBR 2005)*, volume 3620 of *LNAI*, pages 465–476, 2005.