The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Qualitative Reasoning about Cardinal Directions Using Answer Set Programming*

Yusuf Izmirlioglu, Esra Erdem
Sabanci University
Istanbul, Turkey

Abstract

We propose a novel method for representing and reasoning about an incomplete set of constraints about basic/disjunctive qualitative direction relations over simple/connected/disconnected regions, using Answer Set Programming, and prove its correctness with respect to cardinal direction calculus. We extend this method further with default qualitative direction constraints, and discuss its usefulness with some sample scenarios.

Introduction

Various tasks, like navigating to a destination or describing the location of an object, involve dealing with spatial properties and relations of objects. For higher precision of solutions, if data is available, quantitative approaches can be employed to find metric solutions for these tasks. On the other hand, for some applications (e.g., exploration of an unknown environment), quantitative data may not always be available due to incomplete knowledge about the environment; and, for some applications (e.g., that involve human-robot interactions) sociable and understandable interactions and acceptable explanations are often more desirable than high precision (Kuipers 1983). For these applications, qualitative spatial relations seem more suitable. They can deal with describing imprecise data about spatial relations in environments, and their verbal descriptions are sufficient and understandable for describing a way to some destination or the location of an entity.

Consider, for instance, an agent helping a parent to find her missing child in a shopping mall that is not completely known to the agent nor to the parents. If the agent receives some sightings of the child (e.g., "to the south of Store A"), it will be useful (i) if the agent can understand the relative location of the child described qualitatively, and (ii) if the agent can find out where the child might be, based on such qualitative direction constraints, and (iii) describe qualitatively in which direction (e.g., "to north") the parents might go to find their child. In another scenario, a robot may not have quantitative description of an environment (e.g., after a disaster, as in search and rescue), and should be able to understand a human if she provides information by directions, reason about the qualitative constraints provided by the humans, and describe qualitatively the outcome of its reasoning.

We consider a particular sort of qualitative spatial relations, cardinal directions (e.g., west, south, northeast, southwest, and their combinations), between spatial entities, as in Cardinal Direction Calculus (CDC) (Skiadopoulos and Koubarakis 2004; 2005). In CDC, qualitative directional relations between spatial entities are described by means of constraints, like "the missing child is to the south of Store A". In real world, the regions occupied by these entities may have holes (e.g., Store A may have a small garden in the middle) or may be disconnected (e.g., Store A may consist of two parts across a small street). Moreover, the given set of constraints may be incomplete (i.e., qualitative spatial relations between some spatial objects are not known) or some constraints may involve disjunctions (e.g., missing child is to the south of Store A or to the north of Store B). In such cases, with uncertainty or incomplete knowledge, checking the consistency of a given set of constraints is NP-complete (Table 1).

Furthermore, we consider qualitative constraints that necessitate commonsense knowledge like defaults (e.g., food truck is normally to the west of Store X). Such constraints are not considered as part of CDC, so we introduce a new sort of constraints, called default CDC constraints.

We propose a novel formal method to represent and reason about cardinal directions with respect to this extended set of CDC constraints, using Answer Set Programming (ASP) (Brewka, Eiter, and Truszczynski 2016). ASP is a knowledge representation and reasoning framework that can be used to declaratively solve NP-complete problems (e.g., CDC consistency checking). Its expressive languages (like ASP-Core-2 (Calimeri et al. 2013)) and solvers (like CLINGO (Gebser et al. 2011)) support nonmonotonic constructs (e.g., to express default qualitative spatial relations), aggregates (e.g., to identify infimum/supremum of projections of spatial regions), and disjunctions (e.g., to express uncertainty of qualitative directional relations).

In particular, we model consistency checking as a set of formulas in ASP, so that an ASP solver can be used with this general formulation to decide for the consistency of a given

*This work is partially supported by TUBITAK Grant 114E491 and Chist-Era COACHES.

set of domain-specific CDC constraints, which are also modeled as ASP formulas. This method is applicable to all NP-complete cases of consistency checking reported in Table 1.

We prove that our ASP formulation is correct with respect to the definition of Skiadopoulos and Koubarakis: The given set of CDC constraints is consistent if and only if the corresponding ASP formulation is satisfiable.

We discuss the useful aspects of our ASP-based method for reasoning over cardinal directions with some examples, and provide experimental evaluations.

Related Work

There are various qualitative approaches to studying directional spatial relations (Frank 1991; Ligozat 1998; Balbiani, Condotta, and del Cerro 1999; Freksa 1992; Goyal and Egenhofer 1997; Goyal 2000; Renz and Mitra 2004; Skiadopoulos and Koubarakis 2004; 2005; Navarrete, Morales, and Sciavicco 2007; Liu et al. 2010; Liu and Li 2011; Cohn, Renz, and Sridhar 2012; Lee, Renz, and Wolter 2013; Liu 2013). Like many recent theoretical studies (Skiadopoulos and Koubarakis 2005; Navarrete, Morales, and Sciavicco 2007; Liu et al. 2010; Liu and Li 2011; Liu 2013), our work is based on CDC introduced by Skiadopoulos and Koubarakis (Skiadopoulos and Koubarakis 2004) (which is based on (Goyal and Egenhofer 1997; Goyal 2000)) for representing cardinal direction relations between connected plane regions. Our choice of CDC is mainly due to two reasons: cardinal directions (e.g., north, east, west, south, northeast, northwest, southeast, southwest, and their combinations) are more convenient for verbal descriptions, and it is expressive enough to consider spatial objects as themselves rather than points or boxes. Indeed, in a point-based approximation, Spain is northeast of Portugal; in a box-based model Portugal is contained in Spain; according to Skiadopoulos and Koubarakis' model, Spain lies partially at the northwest, at the north, at the northeast, at the east and at the southeast of Portugal.

Reasoning about cardinal directions has been studied, sometimes augmenting them with other sorts of spatial properties and relations, like topological relations, distance and size information (Cohn, Renz, and Sridhar 2012). Computational complexity analysis of the core problems (like consistency checking) has been studied under different conditions, as summarized in Table 1. For NP-complete problems, most of the proposed approaches are monotonic and rely on methods like constraint satisfaction or model checking. However, in some applications, for more acceptable explanations or descriptions of qualitative spatial relations, commonsense knowledge and nonmonotonic reasoning are needed. Therefore, it is not surprising to see formulations of qualitative spatial reasoning using ASP. Interval Algebra (Allen 1983) and variants of Regional Connection Calculus (Randell, Cui, and Cohn 1992) are formalized in ASP (Walega, Bhatt, and Schultz 2015; Li 2012; Brenton, Faber, and Batsakis 2016).

CDC, as defined in (Frank 1991), is represented in ASP in (Walega, Bhatt, and Schultz 2015). However, this approach views objects as points or boxes, and leads to anomalies as discussed by Skiadopoulos and Koubarakis (Skiadopoulos and Koubarakis 2004) and as illustrated by the example above. In this model, spatial objects are of certain shape, like circle or square, which are identified by parameters, like center or radius. Then, for consistency checking, regions are generated by the choice of these parameters. Also, directional relations between objects are expressed in polynomial inequalities and algebraic constraints. Our ASP-based method for qualitative reasoning over cardinal directions is more general: spatial objects do not have to be considered as points or boxes, they can have arbitrary shapes; spatial objects can be disconnected; and spatial relations can be disjunctive. Due to such a general setting, for consistency checking, regions are generated directly from qualitative descriptions, if the network is consistent.

Answer Set Programming

Answer Set Programming (ASP) (Brewka, Eiter, and Truszczynski 2016), based on answer set semantics (Gelfond and Lifschitz 1991), is a knowledge representation and reasoning paradigm that provides a formal framework for modeling intractable problems, like consistency checking in CDC, by logical formulas, called rules, of the form

$$\mathit{Head} \leftarrow A_1, \ldots, A_k, \mathit{not}\ A_{k+1}, \ldots, \mathit{not}\ A_l$$

where $l \geq k \geq 0$, *Head* is an atom or $\perp$, and each A_i is an atom. A rule is called a *constraint* if *Head* is $\perp$. A set of rules is called a *program*.

ASP provides special constructs to express default assumptions, nondeterministic choices, and aggregates. For instance, thanks to default negation *not*, the following rule expresses that normally the elevator works fine (*works*) unless stated or observed otherwise that it does not work ($\neg$*works*):

$$\mathit{works} \leftarrow \mathit{not}\ \neg \mathit{works}.$$

The following choice rule allows nondeterministically selecting at least 1 and at most 3 numbers x for every set u:

$$1\{\mathit{select}(u,x) : \mathit{num}(x)\}3 \leftarrow \mathit{set}(u).$$

The following rule defines the smallest number, N, selected so far using the aggregate *min*:

$$\mathit{smallest}(N) \leftarrow N = \#\mathit{min}\ \{x : \mathit{select}(u,x), \mathit{set}(u)\}.$$

Cardinal Direction Calculus

Cardinal direction calculus (CDC) describes orientation of spatial objects with respect to one another in terms of cardinal direction relations. We briefly describe some terminology and notation relevant to the rest of the paper, in the spirit of (Skiadopoulos and Koubarakis 2004; Liu et al. 2010).

In CDC, spatial objects are regions on a plane (i.e., nonempty regularly closed subsets of $\mathbb{R}^2$). A region is connected if its interior is connected; note that connected regions may have holes inside. A connected region is called simple if it is topologically equivalent to a closed disk (i.e., no holes). A (possibly disconnected) region can be viewed as a finite union of connected regions. We denote by ***Simp***, ***Reg*** and ***Reg****, the sets of simple, connected, and finite unions of connected regions, respectively (Fig. 1(i)).

The minimum bounding box of a region b, denoted $mbr(b)$, is the smallest rectangle which contains b and has

Table 1: Computational Complexity Analysis of Consistency Checking Problems in Cardinal Direction Calculus

	Basic CDC Relations		Disjunctive CDC Relations
	Complete	**Incomplete**	
Simp	P (Liu et al. 2010, Thm 8)	P (Navarrete, Morales, and Sciavicco 2007, Thm 3)	NP-complete (Navarrete, Morales, and Sciavicco 2007, Thm 4)
Reg	P (Liu 2013, Thm 5.4)	NP-complete (Liu et al. 2010, Thm 5)	–
Reg*	P (Liu 2013, Thm 5.7)	NP-complete (Liu 2013, Thm 5.8)	NP-complete (Skiadopoulos and Koubarakis 2005, Thm 6)

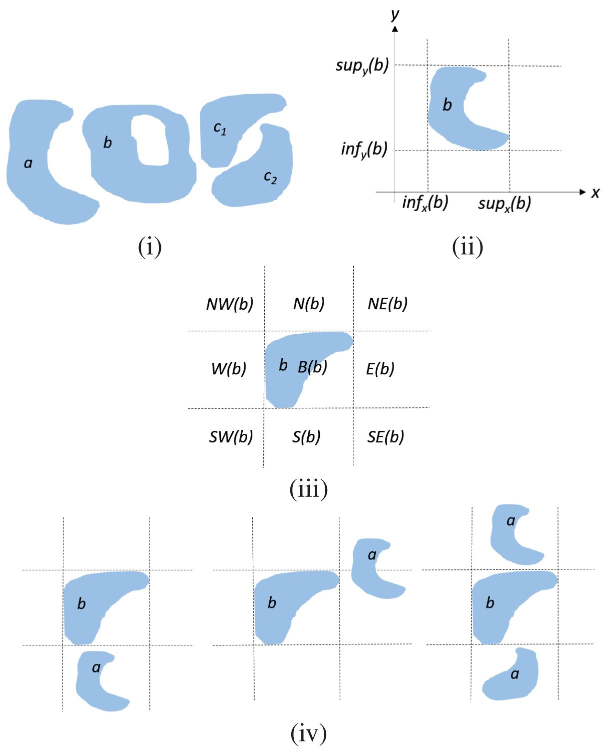

Figure 1: (i) Regions: a, b, c_1, c_2 are in ***Reg***, where $c = c_1 \cup c_2$ is in ***Reg****. (ii) A region and its bounding box. (iii) Reference tiles. (iv) Sample relations (orientation of a with respect to b): $a\ S\ b$, $a\ NE{:}E\ b$, $a\ N : S\ b$.

sides parallel to the axes (Fig. 1(ii)). The sides of the box are essentially defined by the infimum (i.e., $\inf_x(b)$, $\inf_y(b)$) and supremum ($\sup_x(b)$, $\sup_y(b)$) of the projection of b on x-axis and y-axis.

The orientation of a spatial object a (called the primary object) with respect to another spatial object b (called the reference object) is defined by means of cardinal direction relations. For that, we extend the sides of the minimum bounding box $mbr(b)$ of the reference object along the axes, dividing the plane into nine regions (called tiles): $O(b)$, $S(b)$, $SW(b)$, $W(b)$, $NW(b)$, $N(b)$, $NE(b)$, $E(b)$, $SE(b)$ (Fig. 1(iii)). Each tile specifies an orientation with respect to b: onto b, south of b, southwest of b, etc. Then, by identifying the tiles $R_1(b), ..., R_k(b)$ $(1 \leq k \leq 9)$ occupied by the primary object a, we denote the basic CDC relation of a with respect to b by the expression $R_1{:}R_2{:}...{:}R_k$: according to the second figure of Fig. 1(iv), $a\ E : NE\ b$. Note that $a\ R_1{:}R_2{:}...{:}R_k\ b$ iff $a \cap R_i(b) \neq \emptyset$ for every $1 \leq i \leq k$. If $k = 1$ then this basic CDC relation is called a single-tile relation; otherwise, it is called a multi-tile relation. A CDC relation is a finite set $\{\delta_1, ..., \delta_n\}$ of basic CDC relations, intuitively describing their exclusive disjunction.

A formula of the form $u\ \delta\ v$, where u and v are variables ranging over spatial objects in ***Reg**** and δ is a CDC relation, is called a CDC constraint. A CDC constraint network C is a set of CDC constraints defined by a set $V = \{v_1, ..., v_n\}$ of variables ranging over a set D of spatial objects in ***Reg****, and a set Q of CDC relations:

$$C = \{v_i\ \delta_{ij}\ v_j \mid \delta_{ij} \in Q,\ v_i,\ v_j \in V\}.$$

A CDC network C is complete if it specifies a unique CDC constraint for every pair (v_i, v_j) of variables in V $(i \neq j)$; otherwise, it is called incomplete.

A pair (a, b) of spatial objects in ***Reg**** satisfies a CDC constraint $u\ \delta\ v$ if $(a, b) \in \delta$. A solution for a CDC network C with $V = \{v_1, ..., v_n\}$ is a set of n-tuples $(a_1, a_2, ..., a_n) \in D^n$ such that every CDC constraint $v_i\ \delta_{ij}\ v_j$ in C is satisfied by the corresponding pair (a_i, a_j) of spatial objects in D. A CDC network that has a nonempty solution is consistent.

Deciding the consistency of a CDC network is one of the main problems studied in literature about CDC. The complexity analysis of this problem is summarized in Table 1.

Consistency Checking using ASP

Let C be a CDC constraint network defined by a set V of variables ranging over the set D of all spatial objects in ***Reg****, and a set Q of CDC relations. Let us denote by $I = (C, V, D, Q)$ the problem of checking the consistency of this network. Note that checking the consistency of C is defined in continuous space since $D \subseteq 2^{\mathbb{R}^2}$. This problem can be discretized in the spirit of (Liu et al. 2010) by viewing the plane as a sufficiently fine grid so that the regions occupied by spatial objects can be specified by a set of grid cells.

Discretized Consistency Checking

Let Λ_m represent a grid of size $m \times m$. Every spatial object $a \in D$ then can be viewed as a nonempty set $\Lambda_m(a)$ of grid cells (i.e., possibly disconnected regions) in Λ_m. Let D_m denote the set of all such $\Lambda_m(a)$ for every $a \in D$. A pair (a, b) of spatial objects in D satisfies a basic CDC constraint $u\ \delta\ v$ in C if $(\Lambda_m(a), \Lambda_m(b)) \in \delta$. In other words, if

(C1) $\Lambda_m(a) \cap \bigcup\{\Lambda_m(e) \in D_m | \Lambda_m(e)\ R\ b\} \neq \emptyset$ for every single-tile relation R in δ, and

(C2) $\Lambda_m(a) \cap \bigcup\{\Lambda_m(e) \in D_m | \Lambda_m(e)\ R\ b\} = \emptyset$ for every single-tile relation R that is not included in δ,

then a solution $(a_1, a_2, ..., a_n) \in D^n$ for a CDC network C with $V = \{v_1, ..., v_n\}$ can be characterized by a set of n-tuples $(\Lambda_m(a_1), \Lambda_m(a_2), ..., \Lambda_m(a_n)) \in D_m^n$. Thus, if the grid is fine enough, the discretized version $I_m = (C, V, D_m, Q)$ of the consistency checking problem and I have the same answer. If $m \geq 2|V| - 1$ then the grid is fine enough:

Theorem 1. *If $m \geq 2|V| - 1$ then the consistency checking problems $I = (C, V, D, Q)$ and $I_m = (C, V, D_m, Q)$ have the same answers.*

Proof. Every solution of I_m is trivially a solution of I (i.e., if I_m is consistent so is I). Suppose that I is consistent. Take any solution $(a_1, a_2, ..., a_n) \in D^n$ of I. We first show that I can be characterized by a digital solution. In CDC, regions are closed but borders of spatial objects are not important for CDC relations, so borders of each a_i can be removed to obtain an open region $\hat{a}_i$. According to Theorem 1.11 of (Wheeden 2015), each $\hat{a}_i$ can be rewritten as a countable union of non-overlapping closed squares $s_{i,j}$ (j is a positive integer) in $\mathbb{R}^2$. These squares can be divided into smaller squares and normalized if necessary. Now, for every spatial object a_i, consider a finite union $\bar{a}_i$ of N_i squares $s_{i,j}$. Due to Theorem 3.14 and 3.19 of (Wheeden 2015), $\hat{a}_i$, $\bar{a}_i$, and $\hat{a}_i \setminus \bar{a}_i$ are Lebesque measurable sets. Moreover, the measure $|\hat{a}_i \setminus \bar{a}_i| = |\hat{a}_i| - |\bar{a}_i|$ approaches to 0, as N_i tends to infinity. Since zero measure or infinitesimal regions do not change CDC relations, $(\bar{a}_1, \bar{a}_2, ..., \bar{a}_n)$ satisfy C as well. This suggests that every region $\hat{a}_i$ can be approximated by a finite union of non-overlapping closed squares in $\mathbb{R}^2$. Note that such an instantiation preserves the CDC relations in C.

Let us denote by O_x the ordered list that consists of elements of $\{\inf_x(a), \sup_x(a) | a \in D\}$, and by O_y the ordered list that consists of elements of $\{\inf_y(a), \sup_y(a) | a \in D\}$. Now we prove that, there exists an instantiation of a_i such that every two consecutive items in O_x (resp. O_y) differ by at most 1 cell. Suppose otherwise: without loss of generality, there exists an instantiation of a_i such that some consecutive items $o_x(i)$ and $o_x(i+1)$ in O_x differ by 2 cells or more. Then the columns between $o_x(i)$ and $o_x(i+1)$ can be combined into one column. This contraction can be repeated for other consecutive items in O_x, and the order of elements in O_x will not change and the CDC constraints will not be violated. Similarly, the rows can be combined with respect to the ordering in O_y so that the consecutive items differ by at most 1 cell. Then, since $|O_x| = |O_y| = 2|V|$, the solution of I can be represented on a normalized grid of size $m \times m$ where $m = 2|V| - 1$. □

Liu et al. (2010) have the same result for complete networks in ***Reg****. Theorem 1 extends this result to possibly incomplete networks. An alternative proof of Theorem 1 is possible using Liu et al.'s definitions/results.

Theorem 1 allows us to formulate the consistency checking of a CDC network in ASP.

Basic CDC Consistency Checking using ASP

Let us first consider basic CDC costraints. Let $I_m = (C, V, D_m, Q)$ be a discretized version of a consistency checking problem ($m = 2|V| - 1$), where C contains nondisjunctive CDC constraints and may be incomplete. Note that since $D \subseteq$ ***Reg****, spatial objects may be disconnected regions and have holes. We define the corresponding ASP program Π_{I_m} as follows.

We represent the given constraint network in ASP by a set of facts. In particular, we describe a basic CDC constraint $u\ \delta\ v$ by the following facts:

$$\textit{rel}(u, v, R) \leftarrow \qquad (R \in \delta). \tag{1}$$

First, an assignment of a nonempty set $\Lambda_m(u)$ of cells $(x, y) \in \Lambda_m$ to variables $u \in V$ is generated by the choice rules:

$$1\{\textit{occ}_u(x, y) : (x, y) \in \Lambda_m\} \leftarrow \tag{2}$$

where atoms $\textit{occ}_u(x, y)$ express that the grid cells (x, y) are occupied by a spatial object in D_m denoted by the spatial variable u. Note that these choice rules are augmented by a cardinality constraint to ensure that the assignment is nonempty.

Next, it is guaranteed that this assignment satisfies every basic CDC constraint $u\ \delta\ v$ in C. For that, first we identify the minimum bounding box $mbr(v)$ of the spatial object denoted by v, by means of the infimums and supremums of the projections of the object on x and y axes. The following rules define the infimum and supremum on x-axis.

$$\begin{array}{l} \textit{inf}_x(v, \underline{x}) \leftarrow \underline{x} = \min\{x : \textit{occ}_v(x, y),\ (x, y) \in \Lambda_m\} \\ \textit{sup}_x(v, \overline{x}) \leftarrow \overline{x} = \max\{x : \textit{occ}_v(x, y),\ (x, y) \in \Lambda_m\}. \end{array}$$

Note that these definitions use aggregates min and max supported by ASP. Similar rules are added for y axis.

Then, for each single tile relation that δ contains (resp. does not contain), we add constraints for ensuring (C1) (resp. (C2)). For instance, if δ contains the single tile relation N (north) then the following constraint ensures condition (C1) for N: if u is north of v then there should be some cells to the north of $mbr(v)$ occupied by u.

$$\begin{array}{l} \leftarrow \{\textit{occ}_u(x, y) : \underline{x} < x < \overline{x}, y > \overline{y}, (x, y) \in \Lambda_m\}0, \\ \quad \textit{rel}(u, v, N), \textit{inf}_x(v, \underline{x}), \textit{sup}_x(v, \overline{x}), \textit{sup}_y(v, \overline{y}) \end{array} \tag{3}$$

If δ does not contain N then the following constraint ensures condition (C2) for N: if u is not north of v then there should not be any cells to the north of $mbr(v)$ occupied by u.

$$\begin{array}{l} \leftarrow 1\{\textit{occ}_u(x, y) : \underline{x} < x < \overline{x}, y > \overline{y}, (x, y) \in \Lambda_m\}, \\ \quad \textit{not rel}(u, v, N), \textit{inf}_x(v, \underline{x}), \textit{sup}_x(v, \overline{x}), \textit{sup}_y(v, \overline{y}) \end{array} \tag{4}$$

Similar rules are added for other single tile relations.

This ASP program is correct: we can decide for the consistency of a basic CDC network using this ASP program.

Theorem 2. *For a discretized version $I_m = (C, V, D_m, Q)$ of a consistency checking problem, where C consists of basic CDC constraints, I_m has a solution iff the corresponding ASP program Π_{I_m} has an answer set.*

The proof of Theorem 2 follows from Lemmas 1 and 2 below.

Let X be an answer set for Π_{I_m}. For every variable $v \in V$, let us denote by $X(v)$ the assignment of grid cells (x, y) to v obtained from $occ_v(x, y)$ in X. These grid cells essentially form $\Lambda_m(v)$ in D_m.

Lemma 1. *For a discretized version $I_m = (C, V, D_m, Q)$ of a consistency checking problem with $V = \{v_1, ..., v_n\}$, where C consists of basic CDC constraints, let X be an answer set for the ASP program Π_{I_m}. Then the n-tuple $(X(v_1), X(v_2), ..., X(v_n))$ is a solution for I_m.*

Proof. Let Π'_{I_m} be the program obtained from Π_{I_m} by dropping the constraints like (3) and (4). We apply the splitting set theorem (Erdogan and Lifschitz 2004) to Π'_{I_m}: an answer set Y_1 for the top part (1)∪(2) describes the CDC constraints C and possible assignments $\Lambda_m(u)$ of regions to variables $u \in V$, whereas the answer set Y_2 for the bottom part evaluated with respect to Y_1 defines $mbr(u)$ for these variables; and $Y_1 \cup Y_2$ is an answer set for Π'_{I_m}. With Prop. 2 of (Erdogan and Lifschitz 2004), by adding constraints like (3) and (4) for each CDC relation in δ, the answer sets for Π'_{I_m} that do not satisfy (C1) and (C2) are eliminated. Then the answer sets X for Π_{I_m} characterize assignments $X(v)$ of regions to every variable in $v \in V$ that satisfy (C1) and (C2). Thus the n-tuples $(X(v_1), X(v_2), ..., X(v_n))$ are solutions for I_m. □

Let $(\Lambda_m(v_1), \Lambda_m(v_2), ..., \Lambda_m(v_n)) \in D_m^n$ be a solution for $I_m = (C, V, D_m, Q)$. We denote by $Occ_m(v_i)$ the set of atoms of the form $occ_{v_i}(x, y)$ where (x, y) is in $\Lambda_m(v_i)$.

Lemma 2. *If a discretized version $I_m = (C, V, D_m, Q)$ of a consistency checking problem with $V = \{v_1, ..., v_n\}$ and basic CDC constraints in C, has a solution $(\Lambda_m(v_1), \Lambda_m(v_2), ..., \Lambda_m(v_n)) \in D_m^n$, then the ASP program Π_{I_m} has an answer set that contains $\cup_{i=1}^n Occ_m(v_i)$.*

Proof. Every solution $(\Lambda_m(v_1), \Lambda_m(v_2), ..., \Lambda_m(v_n))$ for $I_m = (C, V, D_m, Q)$ describes possible assignments of grid cells to variables $v_i \in V$. Then $\cup_{i=1}^n Occ_m(v_i)$ is included in some answer set Y for the program Π'_{I_m} obtained from Π_{I_m} by dropping constraints like (3) and (4). Every $\Lambda_m(v_i)$ satisfies conditions (C1) and (C2). Then, no $Occ_m(v_i)$ violates constraints like (3) and (4). Then, by Prop. 2 of (Erdogan and Lifschitz 2004), Y is an answer set for Π_{I_m} as well. □

From Theorems 1 and 2, the following corollary follows:

Corollary 1. *For a consistency checking problem $I = (C, V, D, Q)$, where C consists of basic CDC constraints, I has a solution iff the corresponding ASP program $\Pi_{I_{2|V|-1}}$ has an answer set.*

Solving Variations of Consistency Checking

Thanks to the expressive formalism of ASP (e.g., recursive definitions, nondeterministic choice, defaults), we can solve variations of consistency checking by ensuring that the spatial objects are connected, by allowing disjunctive constraints, and/or by expressing default qualitative relations.

Connected Regions

Suppose that $I = (C, V, D, Q)$ is a consistency checking problem, where C contains nondisjunctive CDC constraints and may be incomplete, but the spatial objects are connected (i.e., $D \subseteq$ ***Reg***). We solve this problem by adding the following rules to the ASP program Π_{I_m}.

First, we recursively define (4-)connectedness of grid cells that are occupied by the same spatial object u:

$$\begin{array}{l} conn_u(x_1, y_1, x_2, y_2) \leftarrow occ_u(x_1, y_1), \\ \quad occ_u(x_2, y_2) \quad (|x_1 - x_2| + |y_1 - y_2| = 1) \\ conn_u(x_1, y_1, x_3, y_3) \leftarrow conn_u(x_1, y_1, x_2, y_2), \\ \quad conn_u(x_2, y_2, x_3, y_3) \quad ((x_3, y_3) \in \Lambda_m). \end{array} \tag{5}$$

Note that $conn_u$ expresses the reflexive transitive closure of the adjacency relation of cells occupied by u.

Next, we guarantee that every two grid cells (x_1, y_1) and (x_2, y_2) in Λ_m that are occupied by the same spatial object u are connected indeed:

$$\begin{array}{l} \leftarrow not\, conn_u(x_1, y_1, x_2, y_2), \\ \quad occ_u(x_1, y_1), occ_u(x_2, y_2). \end{array} \tag{6}$$

Theorem 3. *For a discretized version $I_m = (C, V, D_m, Q)$ of a consistency checking problem, where C consists of basic CDC constraints and $D_m \subseteq$ **Reg**, I_m has a solution iff the corresponding ASP program Π_{I_m} combined with (5) ∪ (6) for every variable $u \in V$ has an answer set.*

Proof. The proof follows from an application of the splitting set theorem (Erdogan and Lifschitz 2004) with the atoms of Π_{I_m}, Theorem 2, Prop. 4 of (Erdem and Lifschitz 2003) for extending the answer sets for Π_{I_m} by connectedness definition, and Prop. 2 of (Erdogan and Lifschitz 2004) for ensuring connectedness of cells occupied by the same object. □

From Theorems 1 and 3, we can obtain a correctness result for $I = (C, V, D, Q)$ as a corollary.

Disjunctive CDC Constraints

Suppose that $I = (C, V, D, Q)$ is a consistency checking problem, where C contains disjunctive CDC constraints and may be incomplete, and $D \subseteq$ ***Reg****. We solve this problem by the ASP program Π_{I_m} where $m = 2|V| - 1$ described above, provided that the disjunctive CDC constraints $u \{\delta_1, \delta_2, ..., \delta_z\}\ v$ are represented as follows:

$$\begin{array}{l} 1\{delta(u, v, i) : 1 \leq i \leq z\}1 \leftarrow \\ rel(u, v, R) \leftarrow delta(u, v, i) \quad (R \in \delta_i). \end{array} \tag{7}$$

These rules essentially nondeterministically pick one of the basic CDC constraints in $\{\delta_1, \delta_2, ..., \delta_z\}$.

Theorem 4. *For a discretized version $I_m = (C, V, D_m, Q)$ of a consistency checking problem, where C consists of CDC constraints and $D_m \subseteq$ **Reg***, I_m has a solution iff the ASP program $\Pi^d_{I_m}$ obtained from Π_{I_m} by replacing (1) with (7) for disjunctive CDC constraints has an answer set.*

Proof. The proof follows from an application of the splitting set theorem (Erdogan and Lifschitz 2004) with the splitting set that consists of the atoms of the form $delta(u, v, i)$, and Theorem 2. □

From Theorems 1 and 4, we can obtain a correctness result for $I = (C, V, D, Q)$ as a corollary.

Inferring Cardinal Directions

For a consistency checking problem $I = (C, V, D, Q)$, where C consists of basic CDC constraints and $D \subseteq \textbf{\textit{Reg*}}$, if the given CDC network C is incomplete, it may be useful to infer cardinal directions between two spatial objects u and v whose CDC relation is not known at all (i.e., there is no CDC constraint $u\ \delta\ v$ in C).

The following choice rules infer a basic CDC relation from Q, when added to the relevant ASP program for discretized version of the consistency checking problem $I_m = (C, V, D_m, Q)$:

$$1\{rel(u, v, R) : R \in Q\}1 \leftarrow .$$

Default CDC Constraints

In the variations of consistency checking of CDC constraints, it is assumed that the CDC constraints are monotonic. In various applications, due to dynamic domains with human presence, qualitative spatial relations may have exceptions. For instance, the food truck may change its location from time to time, but it may have a default location, say to the south of the toy store.

Such examples suggest extending a CDC network with a set C_d of default qualitative directional constraints that we denote as follows

$$default\ u\ \delta\ v$$

and necessitate a formalism to express these new form of CDC constraints.

This is possible thanks to nonmonotonic construct *not* and aggregates supported by ASP. For instance, the following rules express *default* $u\ \delta\ v$ when δ is a basic CDC relation:

$$rel(u, v, R) \leftarrow not\ 1\{rel(u, v, R') : R' \in Q^s\}\ (R \in \delta)$$

where Q^s denotes the set of all single-tile relations.

The following rules express *default* $u\ \delta\ v$ when $\delta = \{\delta_1, \delta_2, ..., \delta_z\}$ is a disjunctive CDC relation:

$$\begin{aligned} &1\{delta(u, v, i) : 1 \le i \le z\}1 \leftarrow \\ &rel(u, v, R) \leftarrow delta(u, v, i), \\ &\qquad not\ 1\{rel(u, v, R') : R' \in Q^s\} \quad (R \in \delta_i). \end{aligned}$$

Examples

Consider an assisting agent in a shopping mall, who has incomplete knowledge about relative locations of stores.

Scenario 1: Meeting Suppose that a girl wants to meet her father in the shopping mall, but she has not been to this mall before. She knows that her father is waiting somewhere southwest of the cafeteria and northwest of the boutique. The girl asks an assisting agent for help, who is to the north of the coffee store. With the constraints conveyed by the girl, the agent knows the following: *CoffeeShop* $S{:}SE{:}SW$ *Girl*, *CoffeeShop* $O{:}S$ *Cafeteria*, *Cafeteria* $O{:}N{:}E{:}NE$ *BookStore*, *BookStore* $W{:}NW$ *CoffeeShop*, *Boutique* $W{:}SW$ *BookStore*, *Father* SW *Cafeteria*, *Father* NW *Boutique*. These CDC constraints can be depicted as in Figure 2.

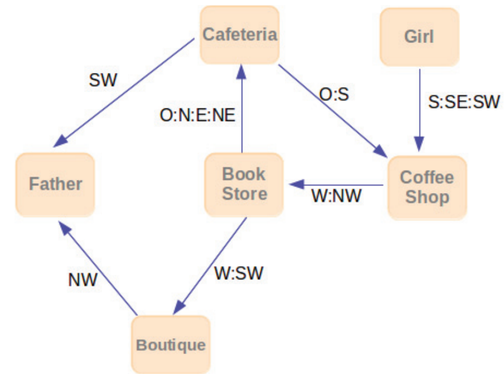

Figure 2: Meeting scenario: Basic CDC constraints.

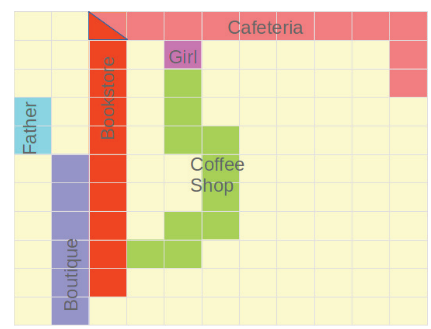

Figure 3: Meeting scenario: Layout.

Using the ASP program for consistency check, the agent can check the consistency of this CDC network and identifies possible relative directions of stores as depicted in Figure 3. Furthermore, as described in the previous section, by extending the ASP program with the rules

$$1\{rel(Father, Girl, R) : R \in Q^s\}1 \leftarrow$$

where Q^s denotes the single-tile relations, the agent infers a new directional relation between the girl and her father: *Father* SW *Girl*. Then, the agent guides the girl towards the direction of her father to the southwest.

Scenario 2: Missing Child Suppose that two parents are looking for their missing child in a shopping mall and request help from an agent in the food court. Suppose also that the parents do not know the exact locations of stores.

Suppose that the agent receives sightings of the child at the west or northwest of the pool, and meanwhile knows that children typically like playgrounds. Then the CDC network that the agent knows contains the disjunctive CDC constraint (e.g., *Child* $\{W, NW\}$ *Pool*), the default CDC constraint (i.e., *default Child* O *PlayGround*), and basic constraints *Parents* O *FoodCourt*, *Bank* $O{:}S$ *FoodCourt*, *Bank* $W{:}NW$ *Pool*, *PlayGround* $S{:}SE$ *Bank*, *PlayGround* $S{:}SW$ *PetStore*, *Pool* $N{:}NE{:}SE$ *PetStore* as depicted in Figure 4.

Using the ASP program for consistency check, extended

Table 2: A comparison of the original ASP formulation (CDC-ASP-1) with the alternative ASP formulation incremental assignment (CDC-ASP-2): CPU times in seconds.

	Connected				Possibly Disconnected			
	Consistent		Inconsistent		Consistent		Inconsistent	
n	CDC-ASP-1	CDC-ASP-2	CDC-ASP-1	CDC-ASP-2	CDC-ASP-1	CDC-ASP-2	CDC-ASP-1	CDC-ASP-2
4	1.98	1.31	2.70	0.97	1.56	0.72	1.61	0.73
6	69.93	13.20	278.70	19.30	65.30	11.25	90.91	11.64
8	>1000	740.63	>1000	404.77	>1000	155.15	>1000	179.73

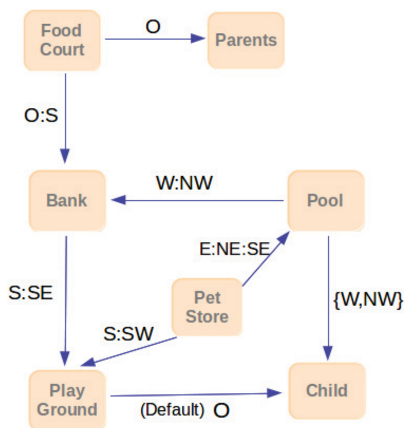

Figure 4: Missing child scenario: CDC constraints.

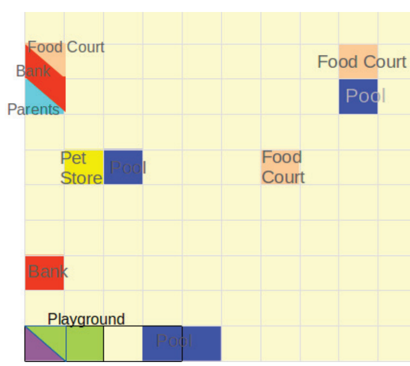

Figure 5: Missing child scenario: Layout.

with disjunctive constraints

$$1\{rel(Child, Pool, W), rel(Child, Pool, NW)\}1 \leftarrow$$

and default constraints

$$rel(Child, PlayGround, O) \leftarrow$$
$$not\ 1\{rel(Child, PlayGround, R) : R \in Q^s\}$$

where Q^s denotes the single-tile relations, the agent can check the consistency of this network and identify possible relative directions of spatial objects as depicted in Figure 3. Furthermore, by extending the ASP program with the rules

$$1\{rel(Child, Parents, R) : R \in Q^s\}1 \leftarrow$$

the agent can infer a new directional relation between the parents and the child: *Child* S *Parents*. Then, the agent guides the parents towards the direction of their child to the south.

Further Improvements and Evaluations

Our ASP formulation of CDC consistency checking generates an assignment of cells to spatial objects at once, by the choice rule (2). In an alternative approach, this assignment can be done incrementally with respect to the given CDC constraints. For instance, if we are given a CDC constraint like $u\ N\ v$ in ASP:

$$rel(u, v, North) \leftarrow$$

then cells to the north of the $\sup_y(v)$ can be used to generate an assignment to the spatial object u:

$$1\{occ_u(x, y) : (x, y) \in \Lambda_m, y > y_2, x \le x_2, x \ge x_1\} \leftarrow$$
$$rel(u, v, North), \sup_y(v, y_2), \sup_x(v, x_2), \inf_x(v, x_1).$$

We have evaluated both formulations over 12 scenarios where directional relations are described by basic CDC relations with incomplete networks specified over $n = 4, 6, 8$ connected/disconnected spatial objects. Six of these instances are consistent, and the others are not. The experiments are performed on a Linux server with Intel E5-2665 CPU with 2.4GHz and 64GB memory, using the ASP solver CLINGO 4.5.4.

According to these results (shown in Table 2), the alternative approach with incremental assignment of cells performs better in terms of computation time, increasing the scalability of our approach. Adding connectedness constraints increases the computation times for both ASP approaches. Further improvements of the ASP formulation, as well as experimental evaluations of our approach with instances that involve default CDC constraints and disjunctive CDC constraints, are part of our ongoing work.

Although we target NP-complete problems of Table 1 with our ASP-based approach, we have also experimented with complete CDC networks (i.e., problems in P), and observed that Liu et al. (2010)'s polytime algorithm performs better for consistency checking of complete CDC networks.

Discussions and Conclusion

Considering cardinal direction calculus (CDC) of Skiadopoulos and Koubarakis, we have introduced a provably correct and generic method for representing constraints about basic/disjunctive qualitative cardinal direction relations over connected/disconnected regions on a plane, using answer set programming (ASP), so that existing state-of-the-art ASP solvers can be used to check the consistency of these constraints and infer new qualitative direction relations when the constraints are incomplete. No existing CDC reasoner can handle uncertainty (represented by disjunctive constraints) or incomplete knowledge.

Note that, in most of the cases, consistency checking of CDC constraints is NP-complete (Table 1), and our method provides solutions for all of them. In that sense, it is more general than the proposed solutions in the literature (including the ASP-based methods).

Furthermore, we have extended CDC with a new sort of constraints, default qualitative direction constraints, that allow us to utilize commonsense knowledge (e.g., children normally like playgrounds) and assumptions (e.g., food truck is normally seen to the south of Store X) about directional relations between spatial objects. These constraints can be formalized in ASP, thanks to nonmonotonic negation and aggregates.

We have illustrated possible uses and usefulness of our methods by sample scenarios in a dynamic environment that involve incomplete knowledge, disjunctive CDC relations, and default CDC constraints. These methods can be applied to various applications, like patrolling/exploration of an unknown environment, without having to change the ASP formulation for consistency checking. Possibility of reasoning over CDC constraints in such environments is important, e.g., for human-robot interactions as well, so that a robot can understand qualitative descriptions of directional relations provided by humans, can reason about these possible uncertain/incomplete qualitative knowledge, and provide guidance to humans by means of qualitative descriptions.

References

Allen, J. F. 1983. Maintaining knowledge about temporal intervals. *Commun. ACM* 26(11):832–843.

Balbiani, P.; Condotta, J.; and del Cerro, L. F. 1999. A new tractable subclass of the rectangle algebra. In *Proc. of IJCAI*, 442–447.

Brenton, C.; Faber, W.; and Batsakis, S. 2016. Answer set programming for qualitative spatio-temporal reasoning: Methods and experiments. In *Tech. Comm. of ICLP*.

Brewka, G.; Eiter, T.; and Truszczynski, M. 2016. Answer set programming: An introduction to the special issue. *AI Magazine* 37(3):5–6.

Calimeri, F.; Faber, W.; Gebser, M.; Ianni, G.; Kaminski, R.; Krennwallner, T.; Leone, N.; Ricca, F.; and Schaub, T. 2013. ASP-Core-2 input language format. https://www.mat.unical.it/aspcomp2013/files/ASP-CORE-2.03c.pdf.

Cohn, A. G.; Renz, J.; and Sridhar, M. 2012. Thinking inside the box: A comprehensive spatial representation for video analysis. In *Proc. of KR*.

Erdem, E., and Lifschitz, V. 2003. Tight logic programs. *Theory and Practice of Logic Prog.* 3(4-5):499–518.

Erdogan, S. T., and Lifschitz, V. 2004. Definitions in answer set programming. In *Proc. of LPNMR*, 114–126.

Frank, A. U. 1991. Qualitative spatial reasoning about cardinal directions. In *Proc. of Auto-Carto 10*.

Freksa, C. 1992. Using orientation information for qualitative spatial reasoning. In *Proc. of International Conference on Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, 162–178.

Gebser, M.; Kaufmann, B.; Kaminski, R.; Ostrowski, M.; Schaub, T.; and Schneider, M. T. 2011. Potassco: The Potsdam answer set solving collection. *AI Commun.* 24(2):107–124.

Gelfond, M., and Lifschitz, V. 1991. Classical negation in logic programs and disjunctive databases. *New Generation Computing* 9:365–385.

Goyal, R., and Egenhofer, M. J. 1997. The direction-relation matrix: A representation for directions relations between extended spatial objects. *The annual assembly and the summer retreat of University Consortium for Geographic Information Systems Science* 3:95–102.

Goyal, R. 2000. *Similarity assessment for cardinal directions between extended spatial objects*. Ph.D. Dissertation, The University of Maine.

Kuipers, B. 1983. *The Cognitive Map: Could It Have Been Any Other Way?* Springer US. 345–359.

Lee, J. H.; Renz, J.; and Wolter, D. 2013. Starvars - effective reasoning about relative directions. In *Proc. of IJCAI*, 976–982.

Li, J. J. 2012. Qualitative spatial and temporal reasoning with answer set programming. In *Proc. of ICTAI*, 603–609.

Ligozat, G. 1998. Reasoning about cardinal directions. *J. Vis. Lang. Comput.* 9(1):23–44.

Liu, W., and Li, S. 2011. Reasoning about cardinal directions between extended objects: The NP-hardness result. *Artificial Intelligence* 175(18):2155–2169.

Liu, W.; Zhang, X.; Li, S.; and Ying, M. 2010. Reasoning about cardinal directions between extended objects. *Artificial Intelligence* 174(12-13):951–983.

Liu, W. 2013. *Qualitative constraint satisfaction problems: algorithms, computational complexity, and extended framework*. Ph.D. Dissertation, University of Technology, Sydney.

Navarrete, I.; Morales, A.; and Sciavicco, G. 2007. Consistency checking of basic cardinal constraints over connected regions. In *Proc. of IJCAI*, 495–500.

Randell, D. A.; Cui, Z.; and Cohn, A. G. 1992. A spatial logic based on regions and connection. In *Proc. of KR*, 165–176.

Renz, J., and Mitra, D. 2004. Qualitative direction calculi with arbitrary granularity. In *Proc. of PRICAI*, 65–74.

Skiadopoulos, S., and Koubarakis, M. 2004. Composing cardinal direction relations. *Artificial Intelligence* 152(2):143–171.

Skiadopoulos, S., and Koubarakis, M. 2005. On the consistency of cardinal direction constraints. *Artificial Intelligence* 163(1):91–135.

Walega, P. A.; Bhatt, M.; and Schultz, C. P. L. 2015. ASPMT(QS): non-monotonic spatial reasoning with answer set programming modulo theories. In *Proc. of LPNMR*, 488–501.

Wheeden, R. L. 2015. *Measure and integral: an introduction to real analysis*, volume 308. CRC press.