TPLP **20** (6): 942–957, 2020.
doi:10.1017/S1471068420000411

Reasoning about Cardinal Directions between 3-Dimensional Extended Objects using Answer Set Programming

Yusuf Izmirlioglu and Esra Erdem
Sabanci University, Faculty of Engineering and Natural Sciences, 34956 Istanbul, Turkey
{yizmirlioglu,esra.erdem}@sabanciuniv.edu

submitted 10 August 2020; accepted 11 August 2020

Abstract

We propose a novel formal framework (called 3D-nCDC-ASP) to represent and reason about cardinal directions between extended objects in 3-dimensional (3D) space, using Answer Set Programming (ASP). 3D-nCDC-ASP extends Cardinal Directional Calculus (CDC) with a new type of default constraints, and nCDC-ASP to 3D. 3D-nCDC-ASP provides a flexible platform offering different types of reasoning: Nonmonotonic reasoning with defaults, checking consistency of a set of constraints on 3D cardinal directions between objects, explaining inconsistencies, and inferring missing CDC relations. We prove the soundness of 3D-nCDC-ASP, and illustrate its usefulness with applications.

KEYWORDS: Qualitative Spatial Reasoning, Answer Set Programming, Cardinal Directional Calculus, 3D Space, Consistency Checking, Marine Exploration, Building Design, Digital Forensics

1 Introduction

Qualitative spatial reasoning studies representation and reasoning with different aspects of space, such as direction, distance, size using parts of natural language rather than quantitative data. Qualitative models are useful in contexts where quantitative data is not available due to incomplete knowledge or uncertainty. Examples are exploration of an unknown territory such as disaster rescue, marine habitat discovery and underwater archeology.

Qualitative reasoning is also relevant for contexts with complete information and quantitative data because human agents tend to express spatial relation or configuration by means of qualitative terms for the sake of sociable and convenient communication. For instance, while designing a building, it is more intuitive and understandable to describe the location of the transformer room as follows: "The transformer room must be at the rear side of the building, near the electric panel. It should be located on a lower level than the entrance." In this case, a quantitative description may be too complicated or even not possible.

Most qualitative calculi and reasoning mechanisms have been developed for objects in 1-dimensional (1D) or 2-dimensional (2D) space, as described in the surveys (Chen et al. 2015; Dylla et al. 2017). However, in real environments, agents move and explore in all 3 dimensions or deal with complex 3-dimensional (3D) objects. In this paper, we study representation of and reasoning about qualitative directions in 3D space. We consider 3D cardinal directions (e.g., to the north and above, the east and below, to the southwest and on the same level) as in the related studies (Chen et al. 2007; Hou et al. 2016), which extend Cardinal Directional Calculus (CDC) (Goyal and Egenhofer 1997; Skiadopoulos and Koubarakis 2004; Skiadopoulos and Koubarakis 2005) to 3D space. Different from these studies, instead of blocks (rectangular prism shape objects), we consider 3D objects of arbitrary shapes, that may be disconnected.

In CDC, cardinal directions between objects are represented by formulas called CDC constraints. In our study, to incorporate commonsense knowledge into reasoning, we introduce a new type of constraint (called *default 3D constraint*) to represent default relations (e.g., the garage is by default below and to the north of the entrance in a building). We call this extended version of 3D-CDC as *3-dimensional nonmonotonic CDC (3D-nCDC)*.

One of the central problems in 3D CDC literature is the consistency checking of a set of 3D CDC constraints. Informally, this problem is concerned about the existence of a possible configuration of objects with respect to the given CDC constraints. We study consistency checking in 3D-nCDC, and provide a general solution that is not restricted to tractable cases as in related work. In addition to consistency checking, we consider other forms of reasoning important for various real-world applications: nonmonotonic reasoning, explaining inconsistencies, and inferring missing CDC relations between objects.

We propose a formal framework to represent 3D-nCDC constraints and to reason about them, using the logic programming paradigm Answer Set Programming (ASP) (Marek and Truszczyński 1999; Niemelä 1999; Lifschitz 2002), based on the answer set semantics (Gelfond and Lifschitz 1988; Gelfond and Lifschitz 1991). For that reason, we call this framework as 3D-nCDC-ASP. We show the soundness and completeness of 3D-nCDC-ASP, implement it using the ASP language ASP-Core-2 (Calimeri et al. 2020) and the ASP solver Clingo (Gebser et al. 2011), and show interesting applications in marine exploration using underwater robots, building design and regulation, and evidence-based digital forensics. Proofs are provided in Appendix of the extended version of this paper (Izmirlioglu and Erdem 2020).

2 Related Work

Cardinal directions in 3D have been studied in the literature for blocks, by directly extending CDC to 3D space (3D CDC) (Chen et al. 2007; Hou et al. 2016), by utilizing projections of objects into 1D (Pais and Pinto-Ferreira 2000) or 2D (Li et al. 2009), or in terms of the 13 relations of Interval Algebra (Allen 1983) as in the block algebra (Balbiani et al. 2002). We understand 3D cardinal directions as in 3D CDC, instead of combinations of lower-dimensional relations that may not be directional. Different from these studies: (i) instead of blocks, we consider 3D objects of arbitrary shapes, that may be disconnected, (ii) to incorporate commonsense knowledge into reasoning, we introduce default 3D constraints to represent default relations. An example that illustrates the strengths of adopting directly a 3D calculus instead of projecting it to lower dimensions is available in the long version of this paper (Izmirlioglu and Erdem 2020).

One of the central problems studied in 3D CDC is the consistency checking of a set of 3D CDC constraints. Polynomial time algorithms have been introduced by Chen et al. (2007) and Hou et al. (2016) for consistency checking in 3D CDC under the condition that constraints are basic (i.e., not disjunctive). Different from these studies: (iii) we study the consistency checking problem in 3D-nCDC and provide a general solution, but without restricting it to the tractable cases, (iv) we also consider other forms of reasoning important for various real-world applications: nonmonotonic reasoning, explaining inconsistencies, and inferring missing CDC relations between objects, and (v) we propose a formal framework (called 3D-nCDC-ASP) to represent 3D-nCDC constraints and to reason about these constraints, using ASP.

ASP has been applied to solve the consistency checking problem in 1D and 2D qualitative calculi. For instance, Brenton et al. (2016) represent Region Connection Calculus with eight base relations (RCC-8) (Cohn et al. 1997), Walega et al. (2017) represent RCC-5 (David A. Randell and Cohn 1992), and Baryannis et al. (2018) represent Trajectory Calculus (Baryannis et al. 2018) in ASP. Like (Brenton et al. 2016) and (Baryannis et al. 2018), we utilize the ASP language ASP-Core-2 and the ASP solver CLINGO ; (Walega et al. 2017) utilizes ASPMT language, and the SMT solver Z3 (de Moura and Bjørner 2008). Different from these studies: (a) we consider a 3D qualitative calculus, and extend it with new types of default constraints whose semantics is provided by means of the nonmonotonic constructs of ASP. Furthermore, (b) we consider not only consistency checking but also other reasoning problems mentioned above.

3D-NCDC-ASP extends our earlier work NCDC-ASP (Izmirlioglu and Erdem 2018), which investigates nonmonotonic CDC in 2D using ASP, to 3D. We represent 3D cardinal directions between 3D extended objects, perform consistency checks of 3D-nCDC constraints, and generate missing 3D cardinal direction relations between objects. Our representation of 3D-nCDC constraints is (c) methodologically different, (d) enables generation of explanations for inconsistencies, and (e) enables a more general definition of default CDC constraints.

Qualitative directional relations in 3D are used in robotics. For instance, Zampogiannis et al. (2015) define six directional relations (i.e., *left*, *right*, *front*, *behind*, *below*, *above*) between point clouds in 3D by utilizing cones, for the purpose of grounding. However, such related work in robotics do not study reasoning problems, like consistency checking or inference of (missing) relations (e.g., compositions or inverses), in the spirit of the well-studied qualitative spatial reasoning calculi. The lack of formal studies on such reasoning problems might lead to incorrect conclusions. For instance, based on Zampogiannis et al. (2015)'s directional relations, Mota and Sridharan (2018) further define *above* as an inverse of *below* by an ASP rule and rely on it for further inferences. However, according to the definitions of directional relations in these studies, it is not always correct that, for every two objects A and B, A is *below* B iff B is *above* A (see the extended version (Izmirlioglu and Erdem 2020) for a counter example). On the other hand, 3D-NCDC-ASP (1) stems from a qualitative spatial calculus of 3D CDC, where computational aspects are well-studied, (2) extends 3D CDC further to 3D-nCDC with nonmonotonic constructs and considering other automated reasoning problems (like inferring missing relations and explanation generation), (3) is sound and complete (Corollary 1), and (4) provides a computational tool to automate reasoning about 3D cardinal directions. In that sense, 3D-NCDC-ASP provides a provably correct method and tool that robotics studies can benefit from.

We have summarized the similarities and differences of our contributions above in comparison with the closely related work in qualitative spatial reasoning about 3D cardinal directional relations (i)–(v), and in applications of ASP to qualitative spatial reasoning, including our earlier studies (a)–(d). We have also discussed related studies about qualitative spatial relations in robotics, and the further needs in robotics for qualitative spatial reasoning by emphasizing the significance of our contributions (1)–(4).

Further differences from the related work and our earlier work will be pointed out as we provide details about 3D-NCDC-ASP.

3 3D-nCDC: Nonmonotonic Cardinal Direction Calculus in 3-Dimensional Space

Cardinal Directional Calculus (CDC) (Goyal and Egenhofer 1997; Skiadopoulos and Koubarakis 2004; Liu et al. 2010) describes qualitative direction of an extended spatial object a (the primary or target object) with respect to another object b (the reference object) on a plane, in terms of

https://doi.org/10.1017/S1471068420000411 Published online by Cambridge University Press

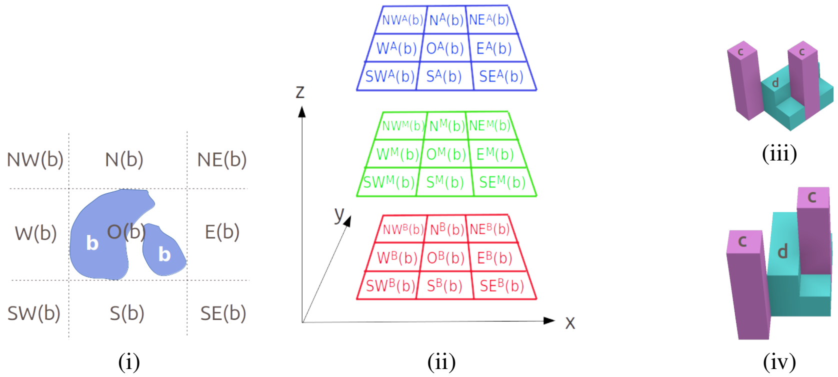

Fig. 1: (i) The minimum bounding rectangle of a region b, and the 9 single-tiles on the plane relative to b. (ii) The 27 single-tiles in 3D relative to object b. (iii) Two spatial objects c and d. (iv) The spatial objects of (iii) are axes-aligned. The direction of c with respect to d in 3D is represented by the multi-tile 3D nCDC relation $O^M : O^A : SW^M : SW^A$.

cardinal directions as follows. The minimum bounding rectangle of a region b, denoted $mbr(b)$, is the smallest rectangle that contains b and has sides parallel to the x and y axes. The minimum bounding rectangle of the reference object b divides the plane into nine regions (called tiles) and these tiles define the nine cardinal directions relative to b: *north* (N), *south* (S), *east* (E), *west* (W), *northeast* (NE), *northwest* (NW), *southeast* (SE), *southwest* (SW), *on* (O), as illustrated in Fig. 1(i). After identifying the unique tiles $R_1(b),\ldots,R_k(b)$ $(1 \le k \le 9)$ occupied by the primary object a, the direction of a with respect to b is expressed by the basic CDC relation R_1:R_2:...:R_k.

Spatial objects and relations in 3D-nCDC Our study relies on two extensions of CDC to 3D space: Three-dimensional Cardinal Direction (TCD) calculus (Chen et al. 2007), and Block Cardinal Direction (BCD) calculus (Hou et al. 2016). TCD and BCD consider spatial objects that are blocks in 3D space. Different from TCD and BCD, we consider *spatial objects* as nonempty, regular, compact volumes in $\mathbb{R}^3$. Spatial objects have positive volume, so lower dimensional entities such as points, lines, surfaces are not considered in 3D nCDC. A subset of $\mathbb{R}^3$ is *regular* if it is equal to closure of its interior; regular objects do not have isolated singular points or emanating lines or planes. A set is connected if it cannot be stated as union of two disjoint nonempty closed sets. An object is *connected* if its interior is a connected set (so trivial cases where an object has two separate components which touch on a mere single point or line are excluded); connected objects might have holes inside. An object that is not connected is called *disconnected*. A *possibly disconnected object* is a union of finite number of connected objects. Let ***Reg*** and ***Reg**** denote the set of connected and possibly disconnected objects in $\mathbb{R}^3$, respectively.

Since we consider spatial objects of arbitrary shapes, we describe the direction of a target object a with respect to a reference object b, by identifying the minimum bounding box of b. Let $\inf_x(b)$ and $\sup_x(b)$ denote the infimum and supremum of the projection of object b on the x-axis. Similarly, the projections of b on the y and z axes are described by $\inf_y(b)$, $\sup_y(b)$, $\inf_z(b)$, $\sup_z(b)$. We define the *minimum bounding box (mbb)* of an object b as a prism whose sides are described by six planes: $x = \inf_x(b)$, $x = \sup_x(b)$, $y = \inf_y(b)$, $y = \sup_y(b)$, $z = \inf_z(b)$, and $z = \sup_z(b)$. Therefore, the $mbb(b)$ of an object b divides the space into 27 tiles: $NW^A(b),\ldots,SE^A(b),NW^M(b),\ldots,SE^M(b),NW^B(b),\ldots,SE^B(b)$ as illustrated in Fig. 1(ii). Here, the superscripts A, M and B denote three levels on the z-axis: *above*, *middle*, *below*. For example, $N^B(b)$ is the tile below and to the north of b, and consists of the coordinates $(x,y,z) \in \mathbb{R}^3$ where $\inf_x(b) < x < \sup_x(b)$, $y > \sup_y(b)$, $z < \inf_z(b)$. Note that the tiles are open sets and do not include their boundary points. In TCD and BCD, the objects are already blocks, so $mbb(b) = b$.

https://doi.org/10.1017/S1471068420000411 Published online by Cambridge University Press

As in TCD and BCD, a *basic 3D-nCDC relation* $a\ R_1{:}R_2{:}\ldots{:}R_k\ b$ holds if and only if $a \cap R_i(b) \neq \emptyset$ for every $1 \leq i \leq k$. For example, in Fig. 1(iii) (that is axes-aligned in (iv)), $c\ O^M : O^A : SW^M : SW^A\ d$. If $k = 1$, this basic CDC relation is called a *single-tile relation*; if $k \geq 2$, it is called a *multi-tile relation*. Let us denote by $\mathscr{R}^s$ the set of single-tile relations, and by $\mathscr{R}$ the set of basic 3D-nCDC relations over ***Reg****.

As in BCD, a *disjunctive 3D-nCDC relation* is a finite set $\delta = \{\delta_1, \ldots, \delta_o\}$, $(o > 1)$ of basic 3D-nCDC relations, intuitively describing their exclusive disjunction. TCD does not consider disjunctive relations. A *3D-nCDC relation* can be basic or disjunctive.

Basic/disjunctive 3D-nCDC constraints A formula of the form $u\ \delta\ v$, where u and v are spatial variables and δ is a 3D-nCDC relation, is called a *3D-nCDC constraint*.

A *3D-nCDC constraint network* C is a set of 3D-nCDC constraints $v_i\ \delta\ v_j$, $(v_i \neq v_j)$ defined by a set V of spatial variables $(v_1, \ldots, v_l)$ where variables range over a domain D of spatial objects in $\mathbb{R}^3$, and a set Q of 3D-nCDC relations δ, such that, for every pair (u_i, u_j) of variables in V, at most one 3D-nCDC constraint is included in C.

A *basic 3D-nCDC (constraint) network* consists of solely basic 3D-nCDC constraints. A basic 3D-nCDC network C is *complete* if it includes a unique 3D-nCDC constraint for every pair (v_i, v_j), $i \neq j$ of variables in V; otherwise, C is *incomplete*.

Consistency checking A pair (a,b) of spatial objects *satisfies* a basic 3D-nCDC constraint $u\ \delta\ v$ if $a\ \delta\ b$ holds. A pair (a,b) of spatial objects *satisfies* a disjunctive 3D-nCDC constraint $u\ \delta\ v$ where $\delta = \{\delta_1, \ldots, \delta_o\}$, if $a\ \delta_i\ b$ holds for exactly one $\delta_i \in \delta$.

Let C be a 3D-nCDC network that consists of basic or disjunctive 3D-nCDC constraints specified by variables in $V = \{v_1, \ldots, v_l\}$. A *solution* for C is a set of l-tuples $(a_1, a_2, \ldots, a_l)$ of spatial objects in D such that every constraint $v_i\ \delta\ v_j$ in C is satisfied by the corresponding pair (a_i, a_j) of spatial objects. If C has a solution then it is called *consistent*.

The *consistency checking problem* $I = (C, V, D, Q)$ in 3D-nCDC, decides the consistency of C.

Theorem 1

If C is an incomplete basic 3D-nCDC network, or C is a 3D-nCDC network that includes disjunctive 3D-nCDC constraints over D=***Reg****, then $I = (C, V, D, Q)$ is an NP-complete problem.

Default constraints of 3D-nCDC To enable defaults for commonsense reasoning, we introduce *default 3D-nCDC constraints*, which are expressions of the form

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

where u and v are variables in V and δ is a basic 3D-nCDC relation in Q. The meaning of default 3D-nCDC constraints is provided in ASP over a discretized space.

4 Discretized Consistency Checking in 3D-nCDC

Let $\Lambda_{m,n,p}$ denote the set of unit cubes (called cells) in a prism of size $m \times n \times p$, aligned with x, y, z axes. Every cell is identified by its x, y, z coordinates, relative to the origin $(1,1,1)$. Every spatial object a is described by a nonempty subset $\Lambda_{m,n,p}(a)$ of cells in $\Lambda_{m,n,p}$ occupied by a.

A cell (x_1, y_1, z_1) is a *neighbor* of another cell (x_2, y_2, z_2) if $|x_1 - x_2| + |y_1 - y_2| + |z_1 - z_2| = 1$. A cell (x_1, y_1, z_1) is connected to another cell (x_2, y_2, z_2) if (x_1, y_1, z_1) is a neighbor of (x_2, y_2, z_2) or (x_1, y_1, z_1) is connected to a neighbor (x_3, y_3, z_3) of (x_2, y_2, z_2). A spatial object a is *connected* in the grid if there exists a (stem) cell in $\Lambda_{m,n,p}(a)$ that is connected to every other cell in a.

The projection of an object b on the x-axis is defined by x-coordinates of all cells of b in $\Lambda_{m,n,p}$. Let $\inf_x^{m,n,p}(b)$ and $\sup_x^{m,n,p}(b)$ denote the infimum and supremum of the projection of b on the

https://doi.org/10.1017/S1471068420000411 Published online by Cambridge University Press

x-axis. Similarly, the projections of b on the y and z axes are denoted by $\inf_y^{m,n,p}(b)$, $\sup_y^{m,n,p}(b)$, $\inf_z^{m,n,p}(b)$, $\sup_z^{m,n,p}(b)$. The minimum bounding box $mbb^{m,n,p}(b)$ of a spatial object b in $\Lambda_{m,n,p}$ is the smallest prism in $\Lambda_{m,n,p}$ that contains b, that has the following sides parallel to the x, y or z axes: $\inf_x^{m,n,p}(b)$, $\sup_x^{m,n,p}(b)$, $\inf_y^{m,n,p}(b)$, $\sup_y^{m,n,p}(b)$, $\inf_z^{m,n,p}(b)$, $\sup_z^{m,n,p}(b)$.

The prism is partitioned into a set $R_{m,n,p}(b)$ of 27 tiles with respect to minimum bounding box of a reference object b. For example, $N^B_{m,n,p}(b)$ is the tile below and to the north of b, and consists of the cells $(x,y,z) \in \Lambda_{m,n,p}$ where $\inf_x^{m,n,p}(b) \le x \le \sup_x^{m,n,p}(b)$, $y > \sup_y^{m,n,p}(b)$, $z < \inf_z^{m,n,p}(b)$.

Let $D_{m,n,p}$ denote the set of all spatial objects in $\Lambda_{m,n,p}$. A pair (a,b) of spatial objects in $D_{m,n,p}$ *satisfies* a basic 3D-nCDC constraint $u\ \delta\ v$ if

(C1) $a \cap R_{m,n,p}(b) \neq \emptyset$ for every single-tile relation R in δ, and
(C2) $a \cap R_{m,n,p}(b) = \emptyset$ for every single-tile relation R that is not included in δ.

A pair (a,b) of spatial objects in $D_{m,n,p}$ *satisfies* a disjunctive 3D-nCDC constraint $u\ \delta\ v$ where $\delta = \{\delta_1, ..., \delta_o\}$, if $a\ \delta_i\ b$ holds for exactly one $\delta_i \in \delta$.

Let C be a 3D-nCDC network that consists of basic or disjunctive 3D-nCDC constraints specified by variables in $V = \{v_1, ..., v_l\}$. A *solution* for C is a set of l-tuples $(a_1, a_2, ..., a_l)$ of spatial objects in $D_{m,n,p}$ such that every constraint $v_i\ \delta\ v_j$ in C is satisfied by the corresponding pair (a_i, a_j) of spatial objects. If C has a solution then it is called *consistent*.

The *discretized consistency checking problem* $I_{m,n,p} = (C, V, D_{m,n,p}, Q)$ in 3D-nCDC, decides the consistency of C. The following theorem allows us to solve I by declaratively solving $I_{m,n,p}$.

Theorem 2
The consistency checking problem $I = (C, V, D, Q)$ over $D = \textbf{\textit{Reg*}}$ and the discretized consistency checking problem $I_{m,n,p} = (C, V, D_{m,n,p}, Q)$ where $m, n, p \geq 2|V| - 1$ have the same answers.

It is important to emphasize here that we discretize the consistency checking problem, not the environment. For example, given a consistency checking problem with a set of qualitative spatial constraints about a building design (as mentioned in the introduction), we do not discretize the building itself; rather we try to solve the discretized consistency checking problem over a 3D grid of appropriate size. We do not process grounded numerical spatial data or instantiate cardinal directions over real numbers either.

5 Discretized Consistency Checking in 3D-nCDC using ASP

Let $I_{m,n,p} = (C, V, D_{m,n,p}, Q)$ be a discretized 3D-nCDC consistency checking problem, where C consists of 3D-nCDC constraints and might be incomplete, and $D_{m,n,p}$ is the set of all spatial objects in $\Lambda_{m,n,p}$ that may be disconnected and have holes. In the following, we incrementally describe an ASP program to solve $I_{m,n,p}$. A brief review of ASP is provided in Appendix of the extended version (Izmirlioglu and Erdem 2020).

5.1 Basic 3D-nCDC Networks

Suppose that C contains basic 3D-nCDC constraints only. Let us describe the ASP program $\Pi^1_{m,n,p}$ that solves $I_{m,n,p}$.

1) We describe every basic 3D-nCDC constraint $u\ \delta\ v$ in C, by atoms of the form $rel(u,v,r)$ for each single-tile relation r in δ. Then, C can be represented by a set F_B of facts:

$$rel(u,v,r) \leftarrow \quad (r \in \delta,\ u\ \delta\ v \in C). \tag{1}$$

https://doi.org/10.1017/S1471068420000411 Published online by Cambridge University Press

For example, a basic 3D-nCDC constraint $a\ N^A : NW^M\ b$ is represented in ASP by the facts:

$$rel(a,b,NA).\quad rel(a,b,NWM).$$

2) A $mbb^{m,n,p}(u)$ is generated for every spatial object u, by nondeterministically identifying the infimum/supremum of its projection on the x axis with the choice rules:

$$\begin{array}{ll}\{inf_x(u,\underline{x}):\ 1\le \underline{x}\le m\}=1\leftarrow & (u\in V)\\ \{sup_x(u,\overline{x}):\ 1\le \overline{x}\le m\}=1\leftarrow & (u\in V)\end{array}\tag{2}$$

ensuring that the infimum is less than or equal to the supremum:

$$\leftarrow inf_x(u,\underline{x}),\ sup_x(u,\overline{x})\quad (\underline{x}>\overline{x},\ u\in V).\tag{3}$$

Similar rules are added for the infimum/supremum of its projection on y and z axes.

3) We instantiate every variable $u\in V$ by a spatial object in $D_{m,n,p}$, by nondeterministically assigning some cells (x,y,z) in $\Lambda_{m,n,p}$ to u so that (i) the minimum bounding box of this object is exactly $mbb^{m,n,p}(u)$ generated by rules $(2)\cup(3)$, and (ii) the 3D-nCDC constraints in C are satisfied.

3)(i) An assignment of cells (x,y,z) to a variable u is described by atoms of the form $occ(u,x,y,z)$, nondeterministically generated by the choice rules:

$$\{occ(u,x,y,z):\ (x,y,z)\in\Lambda_{m,n,p}\}\ge 1\leftarrow\quad (u\in V).\tag{4}$$

Projection of this spatial object onto x axis are defined by the rules:

$$xocc(u,x)\leftarrow occ(u,x,y,z)\quad ((x,y,z)\in\Lambda_{m,n,p},\ u\in V).\tag{5}$$

Similar rules are added for its projection on the y and z axes.

We ensure that, for x axis, the projected coordinates lie between the infimum and supremum,

$$\begin{array}{ll}\leftarrow inf_x(u,\underline{x}),\ xocc(u,x') & (x'<\underline{x},\ 1\le x'\le m,\ u\in V)\\ \leftarrow sup_x(u,\overline{x}),\ xocc(u,x') & (x'>\overline{x},\ 1\le x'\le m,\ u\in V)\end{array}\tag{6}$$

at least one of the cells assigned to u is on the infimum, and another one on the supremum.

$$\begin{array}{ll}\leftarrow not\ xocc(u,\underline{x}),inf_x(u,\underline{x}) & (u\in V)\\ \leftarrow not\ xocc(u,\overline{x}),sup_x(u,\overline{x}) & (u\in V).\end{array}\tag{7}$$

Similar constraints are added for its projection on the y and z axes.

3(ii) We ensure that the instantiations of objects (by assignment of cells (x,y,z) to variables $u\in V$) satisfies every basic 3D-nCDC constraint $u\ \delta\ v$ in C. For that, we add constraints to ensure that conditions (C1) and (C2) are not violated.

For example, if δ contains the single tile relation N^B then we add the following to describe when condition (C1) for N^B is violated (i.e., when u does not occupy any cells to the north of and below $mbb^{m,n,p}(v)$).

$$\begin{array}{l}violated(u,v)\leftarrow rel(u,v,NB),\ inf_x(v,\underline{x}),\ sup_x(v,\overline{x}),\ sup_y(v,\overline{y}),\ inf_z(v,\underline{z}),\\ \quad \#count\{x,y,z\!:\ occ(u,x,y,z),\underline{x}\le x\le\overline{x},\ y>\overline{y},\ z<\underline{z},\ (x,y,z)\in\Lambda_{m,n,p}\}\le 0\quad (u\in V).\end{array}\tag{8}$$

If δ does not contain N^B, then the following rules describe when condition (C2) is violated (i.e., u occupies some cells to the north of and below $mbb^{m,n,p}(v)$).

$$\begin{array}{l}violated(u,v)\leftarrow \#count\{x,y,z\!:\ occ(u,x,y,z),\underline{x}\le x\le\overline{x},\ y>\overline{y},\ z<\underline{z},\ (x,y,z)\in\Lambda_{m,n,p}\}\ge 1,\\ \quad not\ rel(u,v,NB),\ existrel(u,v),\ inf_x(v,\underline{x}),\ sup_x(v,\overline{x}),\ sup_y(v,\overline{y}),\ inf_z(v,\underline{z})\quad (u\in V).\end{array}\tag{9}$$

https://doi.org/10.1017/S1471068420000411 Published online by Cambridge University Press

Here, since the network C might be incomplete, $\mathit{existrel}(u,v)$ atoms identify which pair of variables have a constraint in the network C:

$$\mathit{existrel}(u,v) \leftarrow \mathit{rel}(u,v,r) \quad (r \in \mathscr{R}^s,\ u,v \in V). \tag{10}$$

For every one of 26 other single tile relations, we add rules similar to (8) and (9). After that, we eliminate such violations:

$$\leftarrow \mathit{violated}(u,v), \mathit{existrel}(u,v) \quad (u,v \in V). \tag{11}$$

The ASP program $\Pi^1_{m,n,p}$ described above (including the ASP description F_B of C) for checking the consistency of a basic 3D-nCDC network C over $D_{m,n,p}$ is sound and complete. Let $\mathscr{O}_{m,n,p}$ denote the set of atoms of the form $\mathit{occ}(u,x,y,z)$ where $u \in V$ and x, y, z are positive integers such that $1 \leq x \leq m$, $1 \leq y \leq n$, $1 \leq z \leq p$.

Theorem 3
Let $I_{m,n,p} = (C,V,D_{m,n,p},Q)$ be a discretized consistency checking problem, where C is a basic 3D-nCDC network. For an assignment X of spatial objects in $D_{m,n,p}$ to variables in V, X is a solution of $I_{m,n,p}$ if and only if X can be represented in the form of $X = Z \cap \mathscr{O}_{m,n,p}$ for some answer set Z of $\Pi^1_{m,n,p}$. Moreover, every solution of $I_{m,n,p}$ can be represented in this form in only one way.

From Theorems 2 and 3:

Corollary 1
The consistency checking problem $I = (C,V,D,Q)$ has a solution if and only if the program $\Pi^1_{m,n,p}$ $(m,n,p \geq 2|V|-1)$ has an answer set.

5.2 Disjunctive 3D-nCDC Constraints

Suppose that C contains basic or disjunctive 3D-nCDC constraints only. Let us describe the ASP program $\Pi^2_{m,n,p}$ that solves $I_{m,n,p}$. The program $\Pi^2_{m,n,p}$ is obtained from $\Pi^1_{m,n,p}$, by adding new rules for each disjunctive 3D-nCDC constraint as follows.

1) Every disjunctive 3D-nCDC constraint $u\ \{\delta_1,\dots,\delta_o\}\ v$ in C is represented in ASP by a set F_V of facts:

$$\mathit{disjrel}(u,v,i,r) \leftarrow \quad (r \in \delta_i,\ 1 \leq i \leq o). \tag{12}$$

2) Recall that a pair (a,b) of spatial objects satisfies $u\ \delta\ v$ where $\delta = \{\delta_1,\dots,\delta_o\}$, if $a\ \delta_i\ b$ holds for exactly one $\delta_i \in \delta$. Therefore, for every disjunctive 3D-nCDC constraint $u\ \delta\ v$, we nondeterministically choose $\delta_i \in \delta$, and represent the basic 3D-nCDC constraint $u\ \delta_i\ v$:

$$\{\mathit{chosen}(u,v,i) : 1 \leq i \leq o\} = 1 \leftarrow \tag{13}$$

$$\mathit{rel}(u,v,R) \leftarrow \mathit{chosen}(u,v,i),\ \mathit{disjrel}(u,v,i,R). \tag{14}$$

The ASP program $\Pi^2_{m,n,p}$ is sound and complete.

Theorem 4
Let $I_{m,n,p} = (C,V,D_{m,n,p},Q)$ be a discretized consistency checking problem, where C contains basic or disjunctive 3D-nCDC constraints. For an assignment X of spatial objects in $D_{m,n,p}$ to variables in V, X is a solution of $I_{m,n,p}$ if and only if X can be represented in the form of $X = Z \cap \mathscr{O}_{m,n,p}$ for some answer set Z of $\Pi^2_{m,n,p}$. Moreover, every solution of $I_{m,n,p}$ can be represented in this form in only one way.

https://doi.org/10.1017/S1471068420000411 Published online by Cambridge University Press

5.3 Default 3D-nCDC Constraints

Suppose that C also contains default 3D-nCDC constraints. Let us describe the ASP program $\Pi^3_{m,n,p}$ that solves $I_{m,n,p}$. The program $\Pi^3_{m,n,p}$ is obtained from $\Pi^2_{m,n,p}$, by adding new rules for each default 3D-nCDC constraint as follows.

1) We represent every default 3D-nCDC constraint *default u δ v* (where δ is a basic relation) by a set F_D of facts:

$$\mathit{defaultrel}(u,v,r) \leftarrow \quad (r \in \delta). \tag{15}$$

2) The default 3D-nCDC constraint *default u δ v* applies if there is no evidence against it:

$$\mathit{drel}(u,v) \leftarrow \mathit{not}\ \neg \mathit{drel}(u,v), \mathit{defaultrel}(u,v,r) \quad (r \in \delta). \tag{16}$$

3) The evidence against a default constraint *default u δ v* can be due to violations of conditions (C1) and (C2), which are defined by atoms of the form $\mathit{violatedDef}(u,v)$ similar to atoms $\mathit{violated}(u,v)$: use *defaultrel* instead of *rel*. For example, if δ contains the single-tile relation N^B then we add the following rules to describe when condition (C1) for N^B is violated.

$$\begin{array}{l} \mathit{violatedDef}(u,v) \leftarrow \mathit{defaultrel}(u,v,NB),\ \mathit{inf}_x(v,\underline{x}),\ \mathit{sup}_x(v,\bar{x}),\ \mathit{sup}_y(v,\bar{y}),\ \mathit{inf}_z(v,\underline{z}), \\ \quad \#\mathit{count}\{x,y,z\colon \mathit{occ}(u,x,y,z), \underline{x} \le x \le \bar{x},\ y > \bar{y},\ z < \underline{z},\ (x,y,z) \in \Lambda_{m,n,p}\} \le 0 \quad (u \in V). \end{array} \tag{17}$$

If δ does not contain N^B, then the following rules describe when condition (C2) is violated.

$$\begin{array}{l} \mathit{violatedDef}(u,v) \leftarrow \mathit{not}\ \mathit{defaultrel}(u,v,NB),\ \mathit{existDefRel}(u,v), \\ \quad \#\mathit{count}\{x,y,z\colon \mathit{occ}(u,x,y,z), \underline{x} \le x \le \bar{x},\ y > \bar{y},\ z < \underline{z},\ (x,y,z) \in \Lambda_{m,n,p}\} \ge 1, \\ \quad \mathit{inf}_x(v,\underline{x}),\ \mathit{sup}_x(v,\bar{x}),\ \mathit{sup}_y(v,\bar{y}),\ \mathit{inf}_z(v,\underline{z}) \quad (u \in V). \end{array} \tag{18}$$

For every one of 26 other single tile relations, we add rules similar to (17) and (18).

4) Then, the evidence against a default 3D-nCDC constraint *default u δ v* via such violations can be defined as follows:

$$\begin{array}{l} \neg \mathit{drel}(u,v) \leftarrow \mathit{violatedDef}(u,v), \mathit{existDefRel}(u,v) \\ \stackrel{\sim}{\leftarrow} \neg \mathit{drel}(u,v), \mathit{existDefRel}(u,v) \quad [1@1,u,v] \end{array} \tag{19}$$

where $\mathit{existDefRel}(u,v)$ is defined as follows:

$$\mathit{existDefRel}(u,v) \leftarrow \mathit{defaultrel}(u,v,r) \quad (r \in \mathscr{R}^s,\ u,v \in V). \tag{20}$$

The weak constraint above minimizes the evidences provided by abductive inferences of occupied cells. The rule aims to satisfy as many default 3D-nCDC constraints as possible, so as not to conflict with the other 3D-nCDC constraints in C.

5) The evidence (or abnormal cases) against a default 3D-nCDC constraint can be provided by the user. Consider, for instance, a building whose entrance is from its ceiling; then, the abnormal entrance provides an exception to a default constraint that expresses that the "normally, the terrace is above the entrance". This exception can be expressed as follows:

$$\begin{array}{l} \neg \mathit{drel}(u,v) \leftarrow \mathit{ab}(v), \mathit{existDefRel}(u,v) \\ \neg \mathit{drel}(u,v) \leftarrow \mathit{ab}(u), \mathit{existDefRel}(u,v) \\ \mathit{ab}(\mathit{Entrance}) \leftarrow . \end{array}$$

For every answer set Z for $\Pi^3_{m,n,p}$, the assumption expressed by a default 3D-nCDC constraint *default u δ v applies* if there is no exception $\mathit{drel}(u,v)$ in Z against the default.

https://doi.org/10.1017/S1471068420000411 Published online by Cambridge University Press

6 Connected Spatial Objects

Until now, we have assumed that objects belong to ***Reg****, and they can be disconnected. In many real-world applications, spatial objects are connected (and thus belong to ***Reg***). We ensure connectedness of these objects, by adding the following rules to $\Pi^3_{m,n,p}$.

For each spatial object, we formulate the concept of connectedness by incrementally defining its connected cells starting from one cell (called the stem cell), and then enforce all the cells of the object to be reachable from this stem cell. Note that it is sufficient to check the connectedness only for objects which act as target variables in some constraint in C. The connectedness of other objects can be accomplished by freely constructing them inside their minimum bounding boxes.

1) Let $Trg_C \subseteq V$ be the set of variables that appear as a target object in some constraint in C. We define the stem cell for each target spatial object $u \in Trg_C$, as the left bottom below corner cell of the object. First, we find the cells with minimum x coordinate:

$$\begin{array}{ll} \textit{left-side}(u,y,z) \leftarrow \textit{inf}_x(u,\underline{x}),\, \textit{occ}(u,\underline{x},y,z) & (1 \le y \le n,\ 1 \le z \le p,\ u \in Trg_C) \\ \textit{left-border}(u,y) \leftarrow \textit{inf}_x(u,\underline{x}),\, \textit{occ}(u,\underline{x},y,z) & (1 \le y \le n,\ 1 \le z \le p,\ u \in Trg_C). \end{array} \tag{21}$$

Then, among these cells, we find the cells with the minimum y coordinate

$$\textit{ymin}(u,y_m) \leftarrow \#\textit{min}\,\{y:\ \textit{left-border}(u,y)\,\} = y_m \quad (u \in Trg_C). \tag{22}$$

Then, among these cells, we pick the cell with minimum z coordinate:

$$\begin{array}{ll} \textit{zborder}(u,z) \leftarrow \textit{left-side}(u,y_m,z),\, \textit{ymin}(u,y_m) & (u \in Trg_C) \\ \textit{zmin}(u,z_m) \leftarrow \#\textit{min}\,\{z:\ \textit{zborder}(u,z)\,\} = z_m & (u \in Trg_C). \end{array} \tag{23}$$

Then, we define the stem cell as follows:

$$\textit{stem}(u,\underline{x},y_m,z_m) \leftarrow \textit{inf}_x(u,\underline{x}),\, \textit{ymin}(u,y_m),\, \textit{zmin}(u,z_m) \quad (u \in Trg_C). \tag{24}$$

2) For every target spatial object $u \in Trg_C$), we define a set of connected cells starting from the stem cell:

$$\begin{array}{l} \textit{connset}(u,x,y,z) \leftarrow \textit{stem}(u,x,y,z) \quad (u \in Trg_C). \\ \textit{connset}(u,x_2,y_2,z_2) \leftarrow \textit{connset}(u,x_1,y_1,z_1),\, \textit{occ}(u,x_2,y_2,z_2) \\ \quad (|x_2-x_1|+|y_2-y_1|+|z_2-z_1|=1,\ u \in Trg_C). \end{array} \tag{25}$$

3) We ensure that every cell of u belongs to the connected set:

$$\leftarrow \textit{not connset}(u,x,y,z),\, \textit{occ}(u,x,y,z) \quad (u \in Trg_C). \tag{26}$$

7 Inferring Missing 3D-nCDC Relations

Let Z be an answer set for $\Pi^3_{m,n,p}$. For every pair of different spatial objects a and b, we say that a and b are *related* by a 3D-nCDC relation in Z if there exists an atom $\textit{rel}(a,b,r)$ for some single-tile relation $r \in \mathscr{R}^s$ in Z, or a $\textit{drel}(a,b)$ atom in Z. Otherwise, we say that there is a missing relation between a and b. In such cases (e.g., to explain the relative direction between two objects), it is beneficial to infer the missing relations.

1) Suppose that the user specifies which missing relations (u,v) shall be inferred, by a set F_I of facts of the form $\textit{toinfer}(u,v)$.

2) To infer a missing relating between two different spatial objects u and v, we nondetermin-

https://doi.org/10.1017/S1471068420000411 Published online by Cambridge University Press

istically generate a basic 3D-nCDC relation δ that consists of single-tile relations r:

$$\begin{array}{l} known(u,v) \leftarrow existrel(u,v). \\ known(u,v) \leftarrow drel(u,v). \\ \{infer(u,v,r) : r \in \mathscr{R}^s\} \geq 1 \leftarrow not\ known(u,v), toinfer(u,v). \end{array} \tag{27}$$

3) We add rules similar to (8), (9) and (11), using *infer* atoms instead of *rel* atoms, *inferViolated* atoms instead of *violated* atoms, and *existInfer* atoms instead of *existrel* atoms, to ensure the conditions (C1) and (C2) for each inferred single-tile relation.

Let $\Pi^{3,f}_{m,n,p}$ be the program obtained from $\Pi^3_{m,n,p}$ as described above (including F_I). The atoms of the form $infer(u,v,r)$ in an answer set for $\Pi^{3,f}_{m,n,p}$ describe *inferred 3D-nCDC relations*.

8 Explaining Inconsistencies in 3D-nCDC

If the constraint network C is inconsistent, constraints are not satisfiable all together. However, when we exclude some constraints, the network may become consistent. In that sense, the set of excluded constraints are a source of inconsistency in the original network C.

To find a source of inconsistency, we replace constraints (11) with the weak constraints:

$$\stackrel{\sim}{\leftarrow} violated(u,v), existrel(u,v)\ [1@2,u,v] \quad (u,v \in V). \tag{28}$$

According to this weak constraint, each violated 3D-nCDC constraint has a cost of 1, and the number of violated constraints are optimized with priority 2.

Let Z be an answer set for the program obtained from $\Pi^3_{m,n,p}$ by replacing (11) with (28). Then, the set E_Z atoms of the form $violated(u,v)$ that appear in Z describes the basic/disjunctive constraints $u\ \delta\ v$ in C that are violated; furthermore if these constraints are excluded, then C would be consistent. Therefore, we say that E_Z provides an *explanation* for the inconsistency of the network C.

Note that the inconsistency might be due to the violation of mandatory constraints or users' requests/preferences. Since the mandatory constraints cannot be changed, it might be better to explain inconsistencies in terms of the violations of user's requests/preferences, by replacing (11) with the following weak constraints (instead of (28)):

$$\begin{array}{l} \leftarrow violated(u,v),\ mandatory(u,v),\ existrel(u,v) \quad (u,v \in V) \\ \stackrel{\sim}{\leftarrow} violated(u,v),\ not\ mandatory(u,v),\ existrel(u,v)\ [1@2,u,v] \quad (u,v \in V). \end{array} \tag{29}$$

Such an explanation is illustrated with an example in Section 9.2.

Note that the weak constraint above allows us to find minimal explanations. The priority of the weak constraints in (28), (29) is higher than the priority of the weak constraints utilized by the default constraints (19), since consistency checking is prioritized.

Since the explanations are provided in terms of violations of constraints/preferences specified by the user, they can be presented to the user in an understandable format in the same way as constraints/preferences are specified. For instance, if the 3D-nCDC constraint *Director* O^A *Entrance* is specified by the user as a request in natural language as follows "The director's office is placed above the entrance," and the answer set Z includes the atom $violated(Director, Entrance)$, then an explanation for the inconsistency of the network (i.e., that the design of the building with respect to the given constraints and preferences is not possible) can be presented to the user also in natural language matching with his/her own specification: "... because the director's office cannot be placed above the entrance." If the user specifies his/her requests via a graphical user interface, then the requests that cannot be fulfilled could instead be highlighted by red color.

9 Applications of 3D-nCDC-ASP

We discuss the usefulness of 3D-nCDC-ASP by three interesting real-world applications: marine explorations with an underwater human-robot team, building design and regulation in architecture, and evidence-based digital forensics.

9.1 Marine Exploration with Underwater Robots

The application presented in this section is motivated by the challenges of 3D localization and natural human-robot communication in underwater robotics and marine exploration (Zereik et al. 2018). Below a certain depth, GPS does not function and sunlight cannot penetrate, so obtaining exact and absolute locations of objects is not possible. Topographical entities may be discontinuous and precise boundaries are often not clear, so agents need to describe rough positions of the entities in the fauna relative to one another.

Suppose that a group of researchers and underwater robots are in a mission to discover a biological habitat in ocean basin. The environment is unknown to them. During this exploration, Researcher 1 is investigating the sedimentary rock, Robot 1 is checking the fragmented marsh, which is below the sedimentary rock to its southwest and southeast, and Robot 2 is at the thermal zone, which is above the sedimentary rock to its east and southeast. Robot 2 reports the existence of a semi-active volcanic vent, located above the marsh to its northeast. Researcher 2 finds a kelp forest with two separated parts: one part is located to the north and the other part is located to the southeast of the volcanic vent, both parts are located at a lower depth. Robot 3 discovers a fungi culture to the south of the kelp forest on the same level, and to the east and below of the marsh. The fungi culture is of interest to Researcher 1 but which direction should he proceed to reach it?

The qualitative spatial information provided by the four agents can be encoded as a 3D-nCDC constraint network as follows:

$$\begin{array}{lll} Marsh\ SW^B{:}SE^B\ SedRock & Volcano\ E^A{:}SE^A\ SedRock & Volcano\ NE^A\ Marsh \\ Kelp\ N^B{:}SE^B\ Volcano & Fungi\ S^M\ Kelp & Fungi\ E^B\ Marsh. \end{array}$$

The goal is to infer the relation of the fungi culture with respect to the location of Researcher 1, the sedimentary rock. For that purpose, we consider the program $\Pi^3_{m,n,p}$, including a set F_B of facts (1) describing the basic 3D-nCDC constraints above, and the fact $toinfer(Fungi, SedRock)$. In every answer set for this program, atoms of the form $infer(Fungi, SedRock, r)$ reveal a possible location of the fungi culture with respect to the sedimentary rock. For instance, one of these answer sets computed by CLINGO includes $infer(Fungi, SedRock, SEB)$, leading to the inferred 3D-nCDC constraint $Fungi\ SE^B\ SedRock$. Then, Researcher 1 can be guided towards southeast and below, to find the fungi culture.

9.2 Building Design and Regulation

The application presented in this section is motivated by the challenges of building design and regulations in architecture. As argued in (Borrmann and Beetz 2010), legal requirements and official regulations together with client demands about housing, rooms and equipment inside the building are usually documented using qualitative words of daily language rather than mathematical formulas. For this reason, qualitative spatial reasoning is required.

Suppose that an architect is designing a multi-floor library building. The entrance corridor and the door are in the ground floor, and south (middle front) of the building. The regulations impose the electric panel to be on the same floor or a lower level than the entrance. The electric

https://doi.org/10.1017/S1471068420000411 Published online by Cambridge University Press

panel must also be situated next to the main cable, which is at the north side of the building. The system room can be on another floor, however, for ease of cabling along the shaft, it must be vertically aligned with the electric panel. The heating unit is normally instituted on a lower level, and southwest to the entrance. Moreover, the library director requests her office to be on and above the entrance corridor, for convenience of monitoring. She also requests that the system room be located to the left of her office on the same floor. The presumed location of the secretary is to the right of the director's office. Is it possible to come up with a design of this library to respect all these constraints, requests, and assumptions?

The spatial requirements of the building design describe above can be specified by the following 3D-nCDC constraint network:

$$\begin{array}{lll} \textit{Panel}\ \{N^M, N^B\}\ \textit{Entrance} & \textit{System}\ \{O^B, O^M, O^A\}\ \textit{Panel} & \textit{Director}\ O^A\ \textit{Entrance} \\ \textit{System}\ W^M\ \textit{Director} & \textit{default Heating}\ SW^B\ \textit{Entrance} & \textit{default Secretary}\ E^M\ \textit{Director}. \end{array}$$

With the program $\Pi^3_{m,n,p}$, including a set $F_B \cup F_V \cup F_D$ of facts describing the 3D-nCDC constraints above, this constraint network is found inconsistent by CLINGO. To explain this inconsistency, we utilize the method explained in Section 8: replace the constraints (11) in $\Pi^3_{m,n,p}$ with the weak constraints (29), where $\mathit{mandatory}(\mathit{Panel}, \mathit{Entrance})$ given in the input represents an official regulation. An answer set computed for this program by CLINGO includes the atom $\mathit{violated}(\mathit{Director}, \mathit{Entrance})$, and thus provides the following explanation: the director's request about the location of her office (i.e., the 3D-nCDC constraint *Director* O^A *Entrance*) cannot be fulfilled with respect to the other desired features of the library.

9.3 Evidence-Based Digital Forensics

The application presented in this section is motivated by the challenges of evidence-based digital forensics (Costantini et al. 2019), that goes beyond data analysis. We consider a fictional crime story inspired by Agatha Christie's novel "Hercule Poirot's Christmas". Suppose that the grandfather of the Lee family is murdered.

The police obtains some images of the crime scene from the cameras located in the house. The images yield the following information at the moment of the crime:

$$\begin{array}{llll} \textit{Body}\ S^M : SE^M\ \textit{Table} & \textit{Teapoy}\ E^M\ \textit{Sofa} & \textit{Suitcase}\ \{S^M, SW^M\}\ \textit{Table} & \\ \textit{Body}\ N^M : NE^M\ \textit{Teapoy} & \textit{Phone}\ O^A\ \textit{Table} & \textit{Sofa}\ SE^M\ \textit{Bed} & \textit{Coat}\ O^M\ \textit{Hanger}. \end{array} \tag{30}$$

Notice that, since some images are not clear, there is some uncertainty regarding the position of the suitcase. Meanwhile, the detective Poirot interviews the two suspects of the crime.

Suspect 1 (Pilar): "... Suddenly, some noise and a scream came from upstairs. I immediately went to my grandfather's bedroom and found him dead on the floor. His body was lying in front of the table, a bit to the right. There was a rope hanging up on the window that is behind the body, which is strange. There was a muffler on top of the drawer, which probably belongs to my grandfather. The phone book on the table was open. Also, I saw a whistle and toy balloon on the floor, next to the body to its right, that is somehow peculiar... "

Suspect 2 (Alfred): "... I was sitting in the guest room with Stephan. I heard a noise and then ran upstairs to my father's bedroom. The room was untidy. Probably someone else had visited him before because I noticed a suitcase in front of the table. I saw some drugs on the teapoy. There was a knife on the floor next to the body, to its right. It was to the front and underneath the phone..."

Table 1: *Experimental evaluations*

Instance	$\|V\|$	$\|C\|$	Grid Size	Grounding&Total Time (sec)		Memory (GB)	#Rules
$M1$	5	7	9×9×9	0.30	0.34	<0.01	241853
$M2$	10	14	19×19×19	7.98	10.71	0.77	5050676
$M3$	15	21	29×29×29	48.82	68.11	4.18	27826869
$M4$	20	28	39×39×39	175.33	227.19	13.79	91678832
$B1$	6	6	11×11×11	0.66	477.48	0.13	796379
$B1'$	6	5	11×11×11	0.55	3.30	0.07	714772
$B2$	12	12	23×23×23	16.27	>10000	2.57	15445966
$B2'$	12	10	23×23×23	13.85	2174.47	1.48	13884200
$D1$	16	15	31×31×31	282.64	4401.02	3.87	30577147
$D2$	13	13	25×25×25	82.40	>10000	1.71	13253185

From Suspect 1's statement, the following 3D-nCDC constraints are obtained:

$$\begin{array}{lll} Body\ S^M : SE^M\ Table & Rope\ N^A\ Body & Muffler\ O^A\ Drawer \\ PhoneBook\ O^A\ Table & Whistle\ E^M\ Body & Balloon\ E^M\ Body. \end{array} \quad (31)$$

From Suspect 2's statement, the following 3D-nCDC constraints are obtained:

$$Suitcase\ S^M\ Table \quad Drug\ O^A\ Teapoy \quad Knife\ E^M\ Body \quad Knife\ S^B\ Phone. \quad (32)$$

Considering also the following commonsense knowledge about locations of objects:

$$default\ Phone\ O^A\ Table \quad default\ Umbrella\ O^M\ Hanger \quad default\ Coat\ O^M\ Hanger. \quad (33)$$

the detective concludes that Suspect 1 is truthful whereas Suspect 2 is not.

The 3D-nCDC constraint network obtained from Suspect 2's statements (32), the digital evidence (30) and the commonsense knowledge (33) is found inconsistent by CLINGO, using the program $\Pi^3_{m,n,p}$. An explanation for this inconsistency is found by replacing the constraints (11) in $\Pi^3_{m,n,p}$ with the weak constraints (28): the atom $violated(Knife, Phone)$ in the answer set indicates that the knife cannot be to the front and below of the phone.

9.4 Discussion

We have presented three scenarios from different real-world applications. In each scenario, the 3D-nCDC constraints are obtained from the qualitative directional constraints specified by agents. The number of objects and constraints are reasonable from the perspectives of the relevant real-world applications. Yet, for the purpose of investigating the scalability of our method, we have constructed larger scenarios with more number of objects and constraints by "replicating" the scenarios above multiple times. Instance $M1$ denotes the marine exploration scenario presented in Section 9.1, with 5 spatial objects and 7 3D-nCDC constraints. Instances $M2$–$M4$ replicate this instance twice, three times, and four times, respectively.

We have also constructed some instances to investigate how computational performance changes when the instance is inconsistent. Instance $B1$ denotes the building design scenario presented in Section 9.2, with 6 spatial objects and 6 3D-nCDC constraints; it is inconsistent. Instance $B1'$ is a consistent instance obtained from $B1$ by dropping the violated 3D-nCDC constraint. Instances $B2$ and $B2'$ replicate Instances $B1$ and $B1'$ twice, respectively. In addition, we have considered instances, $D1$ and $D2$, that describe the digital forensics scenarios presented in Section 9.3, where the consistency of statements of Suspect 1 and 2 are checked, respectively.

We have measured the time and memory consumption for these consistency checking problem

https://doi.org/10.1017/S1471068420000411 Published online by Cambridge University Press

instances, on a workstation with 3.3GHz Intel Xeon W-2155 CPU and 32GB memory, using CLINGO 5.3.0. Note that, due to Corollary 1, the solutions found by 3D-nCDC-ASP to these instances are correct. The results are shown in Table 1. We can observe from these results that, as the number of objects and constraints increase, the computation time and memory increase.

For instance, when the number of spatial variables and the number of 3D-nCDC constraints double, and the grid size increases more than 2^3 times (from $M1$ to $M2$, $B1$ to $B2$, $B1'$ to $B2'$), the number of rules (as reported by CLINGO) increases by almost 20 times. Such increase in the program size also causes an increase in the computation time and the memory consumption.

We also observe from Instances $B1$, $B2$ and $D2$ that the inconsistency of a network is determined in a longer time, since the search space gets larger.

10 Conclusion

We have introduced a general and provably correct framework (3D-nCDC-ASP) for representing the cardinal directions between (dis)connected extended objects in 3D space, by means of 3D-nCDC constraints (including default 3D-nCDC constraints), and for reasoning about these relations using Answer Set Programming, based on a discretization of the space (preserving the meaning of cardinal directions in continuous space).

3D-nCDC-ASP can be used to check the consistency of a set of 3D-nCDC constraints, infer unknown cardinal direction relations, and explain source of inconsistency. It can deal with the challenges of incomplete or uncertain knowledge as well as defaults about cardinal directions between objects, as often encountered in applications.

Allowing combinations of reasoning capabilities, 3D-nCDC-ASP provides a flexible environment and a computational tool for various real-world applications, as illustrated by some realistic scenarios in marine explorations with an underwater human-robot team, building design and regulation in architecture, and evidence-based digital forensics.

Acknowledgments We have benefited from useful discussions with Philippe Balbiani (on the use of ASP for qualitative reasoning about cardinal directions), Anthony Cohn, Volkan Patoglu and Subramanian Ramamoorthy (on applications of 3D-nCDC in robotics), Stefania Costantini (on applications of 3D-nCDC in digital forensics), and Mehdi Nourbakhsh (on applications of 3D-nCDC in building design). This work is partially supported by Cost Action CA17124.

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. 2002. Tractability results in the block algebra. *J. Log. Comput. 12,* 5, 885–909.

BARYANNIS, G., TACHMAZIDIS, I., BATSAKIS, S., ANTONIOU, G., ALVIANO, M., SELLIS, T., AND TSAI, P.-W. 2018. A trajectory calculus for qualitative spatial reasoning using answer set programming. *Theory and Practice of Logic Programming 18,* 3-4, 355–371.

BORRMANN, A. AND BEETZ, J. 2010. Towards spatial reasoning on building information models. In *Proceedings of the 8th European Conference on Product and Process Modeling (ECPPM)*. 61–67.

BRENTON, C., FABER, W., AND BATSAKIS, S. 2016. Answer set programming for qualitative spatio-temporal reasoning: Methods and experiments. In *OASIcs-OpenAccess Series in Informatics*. Vol. 52.

CALIMERI, F., FABER, W., GEBSER, M., IANNI, G., KAMINSKI, R., KRENNWALLNER, T., LEONE, N., MARATEA, M., RICCA, F., AND SCHAUB, T. 2020. Asp-Core-2 input language format. *Theory and Practice of Logic Programming 20,* 2, 294–309.

CHEN, J., COHN, A. G., LIU, D., WANG, S., OUYANG, J., AND YU, Q. 2015. A survey of qualitative spatial representations. *The Knowledge Engineering Review 30,* 1, 106–136.

CHEN, J., LIU, D., JIA, H., AND ZHANG, C. 2007. Cardinal direction relations in 3D space. In *Proc. of International Conference on Knowledge Science, Engineering and Management*. 623–629.

COHN, A. G., BENNETT, B., GOODAY, J., AND GOTTS, N. M. 1997. Qualitative spatial representation and reasoning with the region connection calculus. *GeoInformatica 1,* 3, 275–316.

COSTANTINI, S., GASPERIS, G. D., AND OLIVIERI, R. 2019. Digital forensics and investigations meet artificial intelligence. *Ann. Math. Artif. Intell. 86,* 1-3, 193–229.

DAVID A. RANDELL, Z. C. AND COHN, A. G. 1992. A spatial logic based on regions and connection. In *Proc. of KR*.

DE MOURA, L. AND BJØRNER, N. 2008. Z3: An efficient smt solver. In *Tools and Algorithms for the Construction and Analysis of Systems*. 337–340.

DYLLA, F., LEE, J. H., MOSSAKOWSKI, T., SCHNEIDER, T., DELDEN, A. V., VEN, J. V. D., AND WOLTER, D. 2017. A survey of qualitative spatial and temporal calculi: algebraic and computational properties. *ACM Computing Surveys (CSUR) 50,* 1, 7.

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. 1988. The stable model semantics for logic programming. In *Proc. of ICLP*. MIT Press, 1070–1080.

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.

HOU, R., WU, T., AND YANG, J. 2016. Reasoning with cardinal directions in 3D space based on block algebra. *Proc. of ICEITI*.

IZMIRLIOGLU, Y. AND ERDEM, E. 2018. Qualitative reasoning about cardinal directions using answer set programming. In *Proc. of AAAI*.

IZMIRLIOGLU, Y. AND ERDEM, E. 2020. Reasoning about Cardinal Directions between 3-Dimensional Extended Objects using Answer Set Programming (arXiv).

LI, C., LU, J., YIN, C., AND MA, L. 2009. Qualitative spatial representation and reasoning in 3D space. In *Proc. of ICICTA*. 653–657.

LIFSCHITZ, V. 2002. Answer set programming and plan generation. *Artificial Intelligence 138*, 39–54.

LIU, W., ZHANG, X., LI, S., AND YING, M. 2010. Reasoning about cardinal directions between extended objects. *Artificial Intelligence 174,* 12-13, 951–983.

MAREK, V. AND TRUSZCZYŃSKI, M. 1999. Stable models and an alternative logic programming paradigm. In *The Logic Programming Paradigm: a 25-Year Perspective*. Springer Verlag, 375–398.

MOTA, T. AND SRIDHARAN, M. 2018. Incrementally grounding expressions for spatial relations between objects. In *Proc. of IJCAI*. 1928–1934.

NIEMELÄ, I. 1999. Logic programs with stable model semantics as a constraint programming paradigm. *Annals of Mathematics and Artificial Intelligence 25*, 241–273.

PAIS, J. AND PINTO-FERREIRA, C. 2000. Spatial representation and reasoning using the n-dimensional projective approach. In *Tech. Report, AAAI Workshop on Spatial and Temporal Granularity*. 79–82.

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., SCHULTZ, C., AND BHATT, M. 2017. Non-monotonic spatial reasoning with answer set programming modulo theories. *Theory and Practice of Logic Programming 17,* 2, 205–225.

ZAMPOGIANNIS, K., YANG, Y., FERMÜLLER, C., AND ALOIMONOS, Y. 2015. Learning the spatial semantics of manipulation actions through preposition grounding. In *Proc. of ICRA*. 1389–1396.

ZEREIK, E., BIBULI, M., MISKOVIC, N., RIDAO, P., AND PASCOAL, A. 2018. Challenges and future trends in marine robotics. *Annual Reviews in Control*.