On the Propagation Strength of SAT Encodings for Qualitative Temporal Reasoning

Matthias Westphal, Julien Hué, Stefan Wölfl
Department of Computer Science, University of Freiburg
Georges-Köhler-Allee, 79110 Freiburg, Germany
{westpham, hue, woelfl}@informatik.uni-freiburg.de

***Abstract*—Several studies in Qualitative Spatial and Temporal Reasoning discuss translations of the satisfiability problem on qualitative constraint languages into propositional SAT. Most of these encodings focus on compactness, while propagation strength is seldom discussed.**

In this work, we focus on temporal reasoning with the Point Algebra and Allen's Interval Algebra. We understand all encodings as a combination of propagation and search. We first give a systematic analysis of existing propagation approaches for these constraint languages. They are studied and ordered with respect to their propagation strength and refutation completeness for classes of input instances. Secondly, we discuss how existing encodings can be derived from such propagation approaches. We conclude our work with an empirical evaluation which shows that the older ORD-encoding by Nebel and Bürckert performs better than more recently suggested encodings.

I. Introduction

Formulating combinatorial search problems in propositional SAT has become a thriving research direction in Artificial Intelligence. Among the modeled problems, satisfiability problems of Qualitative Spatial and Temporal Reasoning (QSTR) have attracted interest. The key idea of QSTR is the abstraction of spatial or temporal data from concrete numerical information to employ qualitative descriptions of relations for reasoning. Examples of such qualitative formalisms include temporal reasoning, in particular, the Point Algebra (PA) [21] and Allen's Interval Algebra (IA) [1].

The formalisms in QSTR are often understood as constraint languages, which are relational fragments of first-order (FO) logic. The satisfiability problem over such formalisms provides the fundamentals for applications and further reasoning tasks such as logic entailment or minimal descriptions. Therefore a lot of work is directed towards satisfiability both from a theoretical point of view, e.g., by identifying tractable classes [18], [4] and a more applied point of view, e.g., by proposing algorithms [12], [5] or new SAT encodings [10], [19]. However, combining theory and practice is sometimes not easy; theoretical results might turn out to be hard to efficiently use in practice and conversely practical suggestions might not take into account theoretical aspects.

Recent SAT encodings of such qualitative formalisms are syntactic reformulations of a given problem instance that aim at a concise and compact encoding that preserves satisfiability. As we discuss in this paper preserving satisfiability is usually done by making the propositional models correspond to a small tractable class as this provides for a simple and concise encoding that seems more suitable for practical use. However, this ignores the utility of larger classes, and in particular does not take into account existing theoretical work.

Interestingly, recent works in this direction show that researchers are divided on the question whether there exist propositional SAT encodings preserving larger tractable classes. The question is however equivocal and can have a broad range of interpretations. Some options are to give Unit Propagation (UP) the ability to decide satisfiability of large tractable classes, to have propositional models corresponding to large tractable classes, or to ensure UP establishes strong k-consistency and a SAT solver mimics a search strategy for tractable classes. In this paper, we focus on presenting the satisfiability problem as a matter of constraint propagation and search. To this end, we study different propagation techniques and combine recent trends and past ideas to build propagation approaches weaker than strong 3-consistency which preserve specific tractable classes.

We start from a general constraint language formalization and utilize ideas and results by Bodirsky and Chen [3] who consider FO interpretations of the Interval Algebra, and Bodirsky and Dalmau [4] who propose to use Datalog programs to define and study k-consistency for qualitative constraint languages. We first sketch different propagation approaches for the Point Algebra and Interval Algebra through Datalog programs, discuss their properties, and make comparisons between them. We then consider different ways of grounding these programs on problem instances into propositional CNF. This two-step process gives insights into the theoretical properties of UP for constraint propagation on these encodings. In particular, it sheds light on the theoretical properties of existing SAT encodings. Choosing the Datalog abstraction has the benefit that we defer the actual formulation (with a wealth of possible optimizations) in propositional CNF, and also make it clear that the programs could be used as propagation algorithms within Constraint Programming or written as table constraints.

In the next section, we briefly introduce the Point Algebra and Interval Algebra as temporal formalisms. Section II provides the necessary background on QSTR, constraint languages, FO interpretations, Datalog, and comparisons between constraint propagations. In Section III we discuss the Point Algebra as a constraint language and provide simple Datalog programs for propagation and their theoretical comparison. Section IV follows the same idea applied to the Interval Algebra. In Section V we derive propositional CNF from input combined with Datalog programs, and compare this with existing approaches. We present results of an empirical evaluation in Section VI, which complements our theoretical results. Finally, Section VII discusses related work and Section VIII finishes the paper with a summary and outlook on future work.

II. Preliminaries

A. Qualitative temporal reasoning

Qualitative Temporal Reasoning deals with relations that hold between entities in an infinite domain D, e.g., the set $\mathbb{Q}$ of rational numbers for temporal point-like events. In QSTR these relations are usually assumed to induce a *relation algebra*. For this, we define the set $B = \{R_1, \dots, R_m\}$ of base relations which forms a partition of D^2, i.e., a jointly exhaustive and pairwise disjoint set. Relations in B must include the equality on D and be closed under converse. This gives rise to a relation algebra, by first closing the set B under intersection, union, and complement to form a Boolean algebra (denoted by 2^B), and expanding this algebra (conceived of as an algebraic structure) by the set-theoretically defined composition and converse functions. Usually, a crucial question is whether the resulting set of relations is closed under composition, but this is well-known to be true for the formalisms we consider here. In what follows we write $\cup, \cap, \bar{\cdot}, \cdot^{-1}, \circ$ to denote the mentioned operations on 2^B.

We briefly sketch two temporal formalisms which are the main focus in this work.

The Point Algebra (PA) [21] is the most simple formalism for temporal reasoning with 3 base relations between time points represented by rational numbers: "precedes" $<$, "equals" $=$, and "succeeds" $>$. *Allen's Interval Algebra* (IA) [1] describes the possible relations between time intervals. We obtain 13 base relations between intervals (such as "during", "overlaps", etc.) and their disjunctions.

Example 1. *In the PA, we can use relations such as $\{<,=\}$ denoting "precedes or equals"—the union of two base relations. Given information: u $\{<,=\}$ v, v $\{<\}$ w, we can conclude by composition u $(\{<,=\} \circ \{<\})$ w that u $\{<\}$ w.*

Fig. 1 shows the base relations (without the converse relations) of both these formalisms, which are introduced later in a more general setting as relational FO structures.

B. Relational structures

We define standard concepts of FO logic. The notation we use is close to [4], [3].

A *relational signature* σ is a countable set of distinct relation symbols $\{R_1, \dots\}$, each R_i with an associated arity $n_i \in \mathbb{N}$. Given a relational signature σ, a *relational σ-structure* Γ is given by a domain D^Γ and an interpretation of each relation symbol R_i over D^Γ, $R_i^\Gamma \subseteq \left(D^\Gamma\right)^{n_i}$. We say that Γ is a *finite* structure if D^Γ is finite. Further, for $\sigma' \subseteq \sigma$ the σ'-*reduct* of a relational σ-structure Γ is the σ'-structure obtained

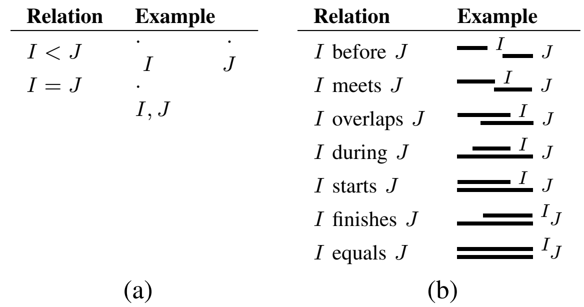

Fig. 1. Base relations (without converses) of the (a) Point Algebra and (b) Allen's Interval Algebra.

from Γ by forgetting the interpretation of all symbols not in σ'. Conversely, a σ-structure Γ is called an *expansion* of a σ'-structure Γ' if Γ' is the σ'-reduct of Γ.

A *constraint language* is essentially a relational structure. One usually refers to constraint languages when relational structures are used in the context of finding solutions to the constraint satisfaction problem (CSP). Consider as an example the simple signature $\sigma_< := \{<\}$ with one binary relational symbol. The $\sigma_<$-structure over $\mathbb{Q}$ with the usual interpretation of $<$ over $\mathbb{Q}$ is a constraint language. All constraint languages used herein have a natural interpretation of their symbols over a domain D, and we simply write $\Gamma^{D,\sigma}$ to denote the constraint language for σ with this natural interpretation.

C. Constraint satisfaction problems

Let Γ be a relational σ-structure. The *constraint satisfaction problem for* Γ is the decision problem whether there exists a homomorphism from some given finite σ-structure I to Γ. That is, a map $h\colon D^I \to D^\Gamma$ that preserves each relation $R_i \in \sigma$, i.e., for an n_i-ary R_i it holds for all $v_1, \dots, v_{n_i} \in D^I$:

$$(v_1, \dots, v_{n_i}) \in R_i^I \Rightarrow \big(h(v_1), \dots, h(v_{n_i})\big) \in R_i^\Gamma \ .$$

An instance I of CSP(Γ) is denoted by $I \in$ CSP(Γ) and we refer to h as a *solution* of I.

If h is a homomorphism defined on a subset of D^I, we say that h is a *partial solution*. A problem instance is said to be k-*consistent* if every partial solution defined on $k-1$ variables can be extended to any other variable to form a partial solution on k variables. A problem instance is *strongly k-consistent* if it is l-consistent for every $l \leq k$.

Example 2. *An instance I of CSP($\Gamma^{\mathbb{Q},<}$) is given by $D^I := \{a, b, c\}$ with $<^I := \{(a, b), (b, c)\}$. The instance is satisfiable, due to the solution h with $h(a) := 0$, $h(b) := 1, h(c) := 2$, as $(0, 1) \in <^{\Gamma^{\mathbb{Q},<}}$, and $(1, 2) \in <^{\Gamma^{\mathbb{Q},<}}$.*

The *primal constraint graph* of an instance I over D^I is the undirected binary graph (D^I, E) where a set $\{v, w\}$ is included in E if there is a k-ary relation R such that $(v_1, \dots, v_k) \in R^I$ and $v, w \in \{v_1, \dots, v_k\}$.

We here chose the relational structures formulation to reinforce the view that reasoning approaches in QSTR typically rewrite finite input structures. Further, Bodirsky and Dalmau [4] stated the utility of Datalog for constraint satisfaction with infinite domains and introduced strong k-consistency on ω-categorical constraint languages for arbitrary k.

D. FO interpretations

We use the definition of interpretation by Hodges [13] which was also used in [3], but emphasize the *syntactic* part.

Let σ, σ' be two relational signatures and d a positive natural number. A d-dimensional *syntactic interpretation* π of σ' in σ is defined by (i) a FO σ-formula (called *domain formula*) $\partial_\pi(v_1, \ldots, v_d)$ with d free variables and (ii) a map that assigns to each relation symbol $R \in \sigma'$ of some arity n a FO σ-formula (the *defining formula*) $\varphi_\pi(R)(\overline{v}_1, \ldots, \overline{v}_n)$ where the $\overline{v}_i$ are disjoint d-tuples of distinct variables.

Let Γ' be a relational σ'-structure and Γ be a relational σ-structure. We say, Γ' has a d-dimensional *FO interpretation* in Γ if there is a d-dimensional syntactic interpretation π of σ' in σ and a surjective map $f_\pi \colon \partial_\pi(\Gamma^d) \to D^{\Gamma'}$ (the *coordinate map*) such that for each $R \in \sigma$ of some arity n and all d-tuples $\overline{a}_i \in \partial_\pi(\Gamma^d)$ it holds:

$$\Gamma' \models R\big(f_\pi(\overline{a}_1), \ldots, f_\pi(\overline{a}_n)\big) \Leftrightarrow \Gamma \models \varphi_\pi(R)(\overline{a}_1, \ldots, \overline{a}_n).$$

Note, there always exists the trivial 1-dimensional FO interpretation (π_{id}, f_{id}) of Γ in Γ.

A syntactic interpretation is said to be *primitive positive* if ∂_π and all φ_π are purely conjunctive formulas with at most existential quantification. This is the case when problem instances of Γ' can easily be conceived of as problem instances of Γ. Further, in the context of a d-dimensional interpretation, we use as notation for d-tuples $\overline{v} = (v^{(1)}, \ldots, v^{(d)})$ and write $\overline{V} = \{\, v^{(i)} \mid v \in V, 1 \leq i \leq d \,\}$ for sets of items.

In the following we only work with syntactic interpretations where ∂_π and all φ_π are quantifier-free and in CNF. Further, we consider expansions to a countable signature $\widehat{\sigma} \supseteq \sigma$ in which arbitrary clauses built on σ are denoted by a relation symbol with the appropriate arity. Thus, all interpretations we use are quantifier-free and primitive positive (qfpp) for such a signature $\widehat{\sigma}$. It allows us to translate every finite σ'-structure I into a $\widehat{\sigma}$-structure $\pi(I)$. Assuming a d-dimensional syntactic interpretation π of σ' in $\widehat{\sigma}$, this $\pi(I)$ is obtained as follows. We say we add a conjunct $R(v_1, \ldots, v_n)$ for an n-ary $R \in \widehat{\sigma}$ to $\pi(I)$ if we add $(v_1, \ldots, v_n)$ to $R^{\pi(I)}$. The full translation procedure is then as follows:

1) set $D^{\pi(I)} := \overline{D^I}$,
2) for each $v \in D^I$ add all conjuncts of $\partial_\pi(v^{(1)}, \ldots, v^{(d)})$ to $\pi(I)$, and
3) for each $R \in \sigma'$ of some arity n and tuple $(v_1, \ldots, v_n) \in R^I$, add each conjunct of $\varphi_\pi(R)(\overline{v}_1, \ldots, \overline{v}_n)$ to $\pi(I)$.

E. Datalog

We give a very brief description of Datalog. In particular, we decided to introduce it without constants nor distinguishing intentional and extensional database predicates; for a general introduction we refer to [8].

We denote by false a distinguished 0-ary relation symbol representing inconsistency (cf. [4]). A *σ-Datalog program* Π is a set of rules of the form $L_0 :\!- L_1, \ldots, L_n$ where each L_i is an atomic formula $R_i(v_1, \ldots, v_{n_i})$ with an n_i-ary relation symbol R_i, each v_i is a variable, $R_0 \in \sigma \cup \{\mathsf{false}\}$, and $R_i \in \sigma$ for $0 < i \leq n$. Further, each variable occurring in L_0 must occur in some $L_i, 1 \leq i \leq n$. L_0 is referred to as the head and $L_1, \ldots, L_n$ as the body of a rule. A rule is a *k-rule* if it has k distinct variables.

The operational semantics of Datalog can here be given as follows. Our input is a finite σ'-structure I. The evaluation of the program is an "extended" finite $\sigma \cup \sigma'$-structure $\Pi(I)$. We initialize $\Pi(I)$ with $D^{\Pi(I)} = D^I$, $R^{\Pi(I)} \leftarrow R^I$ for $R \in \sigma'$, $R^{\Pi(I)} \leftarrow \emptyset$ for $R \in \sigma \setminus \sigma'$, and then repeatedly apply all rules until a fixpoint is reached. Rule application works as follows. Let $w_1, \ldots, w_n$ be the variables appearing in the rule's head and α be any map of the variables appearing in the rule to $D^{\Pi(I)}$. If for each $L_i = R_i(v_1, \ldots, v_{n_i})$ in the body it holds $\big(\alpha(v_1), \ldots, \alpha(v_{n_i})\big) \in R_i^{\Pi(I)}$, add $\big(\alpha(w_1), \ldots, \alpha(w_n)\big)$ to $R_0^{\Pi(I)}$. In case $R_0 = \mathsf{false}$, we say Π has derived false, which we understand as a contradiction. This will be made explicit later. Note this procedure is polynomial time.

Datalog is here used to sketch different types of constraint propagation, in particular ones weaker than strong 3-consistency. For this, we also consider a limited evaluation of Datalog programs on *chordal graphs*. A binary undirected graph $G = (V, E)$ is *chordal* if each cycle in G of length at least four has an edge between two non-adjacent nodes. We denote by $\widehat{G} = (V, \widehat{E})$ some chordal graph with $\widehat{E} \supseteq E$. Such restricted Datalog programs on chordal graphs will only derive information on edges in $\widehat{E}$. We make this explicit in the following definition.

Definition 1. *For signatures σ, σ', let Π be a σ-Datalog program, I be a finite relational σ'-structure, and $\widehat{G} = (D^I, \widehat{E})$ be some chordal supergraph of the primal constraint graph of I. We denote by $\Pi|_{\widehat{G}}$ the Datalog program that is the* chordal variant *of Π on symbols $\sigma \cup \{E\}$ where E is a distinguished binary relation symbol. We expand I to $\sigma' \cup \{E\}$, set $E^I := \{\, (v, w) \mid \{v, w\} \in \widehat{E} \,\}$, and append to the body of each rule $E(v, w)$ for all pairs of variables v, w that appear together in an atomic formula in the rule.*

By $\Pi|_{\widehat{G}}(I)$ we denote the $\sigma \cup \sigma'$-reduct of the result of the program application, i.e., we forget E after computation.

For our purposes it is irrelevant how $\widehat{G}$ is computed. However, we do require for comparisons that it is uniquely defined for a fixed instance I. In particular, for translated instances $\pi(I)$ we assume that $\widehat{G}$ is derived from I and as such for $\{v, w\} \in E^I$ we can infer information between all components of $\overline{v}, \overline{w}$, i.e., we have $E^{\pi(I)} := \{\, (v^{(i)}, w^{(j)}) \mid (v, w) \in E^I, 1 \leq i, j \leq d \,\}$.

F. Comparing Datalog programs on different representations

In the following, we write $\Pi \circ \pi$ to denote the combination of a syntactic interpretation (the representation) and a Datalog program (the constraint propagation). Note, the considered syntactic interpretations are linear time translations and also include the trivial interpretation π_{id}.

Definition 2 (cf. [4])**.** *Let Γ be a constraint language, π be a syntactic interpretation of the relation symbols of Γ, and Π be a Datalog program. The program given interpretation $\Pi \circ \pi$* solves *an instance $I \in CSP(\Gamma)$ if Π derives* false *on $\pi(I)$ if and only if I is unsatisfiable. We say $\Pi \circ \pi$ is* refutation-complete *for* CSP(Γ) *if it solves all instances of CSP(Γ).*

In the following we define a partial order on Datalog programs with interpretations which conveys a notion of propagation strength. Our definition takes into account the considered syntactic interpretations.

Definition 3. *Let I, Δ be relational τ-structures with $I \in CSP(\Delta)$. We denote by $\mathrm{PSol}_\Delta(I)$ the* set of all partial solutions *of I in Δ. For a τ-Datalog program Π, $\mathrm{PSol}_\Delta\big(\Pi(I)\big)$ is the set of partial solutions of the τ-structure $\Pi(I)$ except if Π derives* false *in which case we define $\mathrm{PSol}_\Delta\big(\Pi(I)\big)$ to be empty.*

Let Γ be a relational σ-structure such that Δ has a qfpp interpretation (π, f_π) in Γ of dimension d. For $\pi(I)$, let I' be a σ-structure defined on the same domain, such that for each $R \in \sigma$ it holds $R^{\pi(I)} \subseteq R^{I'}$. Every partial solution $h' \in \mathrm{PSol}_\Gamma(I')$ defined on $\overline{V'}$ for some $V' \subseteq D^I$ induces a map $h\colon V' \to D^\Delta$ by $h(v) := f_\pi\big(h'(v^{(1)}), \ldots, h'(v^{(d)})\big)$. We denote the set of these maps by $\mathrm{PSol}_\Delta(I')$, as they are partial solutions of $I \in CSP(\Delta)$.

Definition 4. *Let Δ be a relational structure. Further, let Γ be a relational σ-structure with a qfpp interpretation of Δ in Γ using the syntactic part π and Π be a σ-Datalog program. Similarly, consider a second relational σ'-structure Γ' with an analogous setup. We say $\Pi \circ \pi$ is* strictly stronger *than $\Pi' \circ \pi'$ wrt. Δ, denoted by $\Pi \circ \pi \prec_\Delta \Pi' \circ \pi'$, if and only if for every $I \in CSP(\Delta)$ it holds $\mathrm{PSol}_\Delta\big(\Pi(\pi(I))\big) \subseteq \mathrm{PSol}_\Delta\big(\Pi'(\pi'(I))\big)$, and there exists an $I' \in CSP(\Delta)$ such that $\mathrm{PSol}_\Delta\big(\Pi(\pi(I'))\big) \subsetneq \mathrm{PSol}_\Delta\big(\Pi'(\pi'(I'))\big)$.*

III. The Point Algebra

The signature of the PA is $\sigma_{\mathrm{PA}} := \{=, <, \leq, \neq, \geq, >, \mathbb{Q}^2\}$ where each symbol is binary (we here exclude $\emptyset$ to simplify matters). Then, the PA is simply $\Gamma^{\mathbb{Q},\sigma_{\mathrm{PA}}}$. It is well-known that strong 3-consistency implies satisfiability (Th. 2 in [15]) and instances of CSP$(\Gamma^{\mathbb{Q},\sigma_{\mathrm{PA}}})$ can be solved in polynomial time.

We are further interested in two reducts of $\Gamma^{\mathbb{Q},\sigma_{\mathrm{PA}}}$ induced by the following signatures: (i) $\sigma_{\mathrm{PAB}} := \{<, =, >\}$, and (ii) $\sigma_{ORD} := \{\leq, =, \neq\}$. The base relation reduct σ_{PAB} is strictly less expressive than the PA as it cannot express $\leq$ [3]. However, the σ_{ORD}-reduct can express the entire PA. More precisely, there is a qfpp interpretation of $\Gamma^{\mathbb{Q},\sigma_{\mathrm{PA}}}$ in $\Gamma^{\mathbb{Q},\sigma_{ORD}}$ which we denote by π_{ORD}: the dimension is 1, the coordinate map is the identity, and the defining formulas are straightforward, e.g., $\varphi_{\pi_{ORD}}(v < w) = v \leq w \wedge v \neq w$.

A. Propagation for the PA and reducts

For $\Gamma^{\mathbb{Q},\sigma_{\mathrm{PA}}}$ the following Datalog program $\Pi^{\mathrm{PA}}_{\leq 3}$ establishes strong 3-consistency using composition, converse, and intersection as provided by the traditional relation-algebraic view.

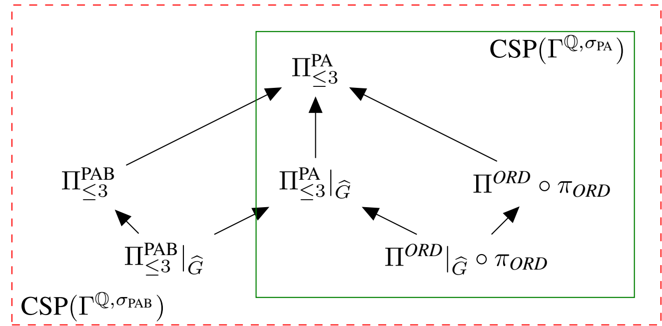

Fig. 2. Lattice of propagation strength of Datalog programs for PA. Frames indicate refutation completeness for reducts.

For all pairs $A, B \in \sigma_{\mathrm{PA}}$, we have as rules

$$x\;C\;y \;\;\text{:-}\;\; x\;A\;z, z\;B\;y. \qquad x\;I\;y \;\text{:-}\; x\;A\;y, x\;B\;y.$$
$$y\;R\;x \;\;\text{:-}\;\; x\;A\;y.$$

where $C \in \sigma_{\mathrm{PA}}$ denotes the composition $A \circ B$, $I \in \sigma_{\mathrm{PA}}$ denotes the intersection $A \cap B$ (where we replace the head with false if the intersection is empty), and $R \in \sigma_{\mathrm{PA}}$ denotes A^{-1}. Additionally, we add false :- $x\;D\;x$ for each $D \in \{<, \neq, >\}$ to establish 1-consistency.

Besides $\Pi^{\mathrm{PA}}_{\leq 3}$ we consider three variants: (i) $\Pi^{\mathrm{PA}}_{\leq 3}|_{\widehat{G}}$, (ii) $\Pi^{\mathrm{PAB}}_{\leq 3}$ which is $\Pi^{\mathrm{PA}}_{\leq 3}$ restricted to those rules whose bodies have symbols in σ_{PAB}, and (iii) $\Pi^{\mathrm{PAB}}_{\leq 3}|_{\widehat{G}}$.

We further consider the application of the interpretation π_{ORD}. Following Nebel and Bürckert [18], we consider the theory of weak orders here formulated as Π^{ORD}:

$$\begin{array}{lll@{\qquad}lll} x \leq y & \text{:-} & x \leq z, z \leq y. & x \leq y & \text{:-} & x = y. \\ x = y & \text{:-} & x \leq y, y \leq x. & y \leq x & \text{:-} & x = y. \\ \text{false} & \text{:-} & x = y, x \neq y. & \text{false} & \text{:-} & x \neq x. \end{array}$$

We also obtain the chordal variant $\Pi^{ORD}|_{\widehat{G}}$.

The propagation strength of these programs given by $\prec_{\Gamma^{\mathbb{Q},\sigma_{\mathrm{PA}}}}$ is shown in Fig. 2, where arrows symbolize the partial order and a program is refutation-complete for a particular reduct if it appears in the corresponding frame.

Proposition 1. *The partial order shown in Fig. 2 is correct. It is further complete given the transitivity of $\prec_{\Gamma^{\mathbb{Q},\sigma_{PA}}}$.*

Proof idea: Compare the rules of the programs: $\Pi^{\mathrm{PA}}_{\leq 3}$ is a strict superset of $\Pi^{\mathrm{PAB}}_{\leq 3}$, Π^{ORD} does not propagate $<$, and chordal variants only propagate on edges in $\widehat{G}$. ■

Proposition 2. *1) $\Pi^{PAB}_{\leq 3}|_{\widehat{G}}$ is refutation-complete for $\Gamma^{\mathbb{Q},\sigma_{PAB}}$, 2) $\Pi^{ORD}|_{\widehat{G}} \circ \pi_{ORD}$ is refutation-complete for $\Gamma^{\mathbb{Q},\sigma_{PA}}$, 3) $\Pi^{PAB}_{\leq 3}$ is* not *refutation-complete for $\Gamma^{\mathbb{Q},\sigma_{PA}}$, 4) refutation completeness of other programs as indicated in Fig. 2 follows from the transitivity of $\prec_{\Gamma^{\mathbb{Q},\sigma_{PA}}}$.*

Proof idea: 1), 2) Satisfiability coincides with conditions on cycles [20], [18], [12]. Chordal graphs express the necessary chords on induced cycles in G such that the programs recognize cycles. 3) $\Pi^{\mathrm{PAB}}_{\leq 3}$ does not infer from symbols $\sigma_{\mathrm{PA}} \setminus \sigma_{\mathrm{PAB}}$, e.g. $\leq, \neq$. ■

IV. The Interval Algebra

We have a signature for the 13 binary base relations of the IA denoted by σ_{IAB}. The IA signature $\sigma_{\text{IA}} := 2^{\sigma_{\text{IAB}}} \setminus \{\emptyset\}$ is rather large as it has a cardinality of $|\,\sigma_{\text{IA}}\,| = 2^{13} - 1 = 8\,191$.

It is clear that intervals and points are closely related. We can construct the intended semantics of the IA based on a syntactic interpretation π of σ_{IAB} in σ_{PAB} (cf. [3]) which defines the base relations between intervals based on the relations between their start and endpoints as seen in Fig. 1. The dimension of this interpretation is 2 and the domain formula $\partial_\pi(v, w) := v < w$ describes well-formed intervals. It induces the interpretation of σ_{IAB} on the domain of intervals $int(\mathbb{Q}) := \partial_\pi(\Gamma^{\mathbb{Q},\sigma_{\text{PAB}}}) = \{\ \overline{a} \in \mathbb{Q}^2 \mid a^{(1)} < a^{(2)}\ \}$. Because all other symbols in σ_{IA} denote unions of σ_{IAB} symbols, we extend π to σ_{IA} by using defining formulas equivalent to disjunctions of those on base relations. This defines the entire IA constraint language $\Gamma^{int(\mathbb{Q}),\sigma_{\text{IA}}}$ based on $\Gamma^{\mathbb{Q},\widehat{\sigma_{\text{PA}}}}$.

In contrast to the tractable PA, solving instances of $\text{CSP}(\Gamma^{int(\mathbb{Q}),\sigma_{\text{IA}}})$ is NP-complete [1]. There are however several important reducts of the IA corresponding to relations that are definable with special qfpp $\widehat{\sigma_{\text{PA}}}$-formulas. Here the most important ones are the following.

1) The $\sigma_{\mathcal{P}\text{-PAB}}$-reduct of relations that are *pointizable over base relations*; those which are definable by qfpp σ_{PAB}-formulas.
2) The $\sigma_{\mathcal{P}}$-reduct of *pointizable* relations [15]; those which are definable by qfpp σ_{PA}-formulas.
3) The $\sigma_{\textit{ORD-H}}$-reduct of ORD-Horn relations [18]; those which are definable by qfpp $\widehat{\sigma_{\textit{ORD}}}$-formulas that only consist of negation-free clauses that contain at most one symbol of $\{\leq, =\}$.

It is easy to see that $\sigma_{\mathcal{P}}$ is also the set of relations that are definable by qfpp $\sigma_{\textit{ORD}}$-formulas.

Example 3. *Consider the following relations and defining formulas given by Nebel and Bürckert [18]:*

- $\{\textit{before}, \textit{meets}, \textit{overlaps}\}$ *is pointizable:* $\varphi_\pi(\{\textit{before}, \textit{meets}, \textit{overlaps}\})(\overline{v}, \overline{w}) = v^{(2)} < w^{(2)}$
- $\{\textit{overlaps}, \textit{starts}, \textit{finishes}^{-1}\}$ *is an ORD-Horn relation:* $\varphi_\pi(\{\textit{overlaps}, \textit{starts}, \textit{finishes}^{-1}\})(\overline{v}, \overline{w}) = v^{(1)} \leq w^{(1)} \wedge w^{(1)} \leq v^{(2)} \wedge w^{(1)} \neq v^{(2)} \wedge v^{(2)} \leq w^{(2)} \wedge (v^{(1)} \neq w^{(1)} \vee v^{(2)} \neq w^{(2)})$
- $\{\textit{before}, \textit{after}\}$ *is neither pointizable nor an ORD-Horn relation:* $\varphi_\pi(\{\textit{before}, \textit{after}\})(\overline{v}, \overline{w}) = (v^{(2)} < w^{(1)} \vee w^{(2)} < v^{(1)})$

The well-known relation between these signatures is

$$\sigma_{\text{IAB}} \subsetneq \sigma_{\mathcal{P}\text{-PAB}} \subsetneq \sigma_{\mathcal{P}} \subsetneq \sigma_{\textit{ORD-H}} \subsetneq \sigma_{\text{IA}}\ ,$$
$$|\,\sigma_{\text{IAB}}\,| = 13 < |\,\sigma_{\mathcal{P}\text{-PAB}}\,| = 29 < |\,\sigma_{\mathcal{P}}\,| = 187 < |\,\sigma_{\textit{ORD-H}}\,| = 867\ ,$$

see, e.g., [18]. Numbers are given for σ_{IA}, i.e., excluding the empty relation, but including the universal relation.

Instances of all these reducts can be solved in polynomial time using strong 3-consistency and the $\sigma_{\textit{ORD-H}}$-reduct is further the largest tractable reduct containing all base relations [18]. This of course underlines the interest in having algorithms that can take advantage of it.

A. Used syntactic interpretations

We define three distinct syntactic interpretations of σ_{IA}. These interpretations constitute FO interpretations and we later show that their "grounding" to propositional SAT matches existing SAT encodings of the IA.

The ORD-Horn representation $\pi_{\textit{ORD-H}}$ follows [18] and interprets σ_{IA} in $\widehat{\sigma_{\textit{ORD}}}$. The domain formula is here translated accordingly to $\partial_{\pi_{\textit{ORD-H}}}(v, w) := v \leq w \wedge v \neq w$. Each $\varphi_{\pi_{\textit{ORD-H}}}$ is a negation-free CNF $\widehat{\sigma_{\textit{ORD}}}$-formula that consists of prime implicates necessary to build the relation [18]. This map gives the right clauses for $\sigma_{\textit{ORD-H}}$ as defined above.

The syntactic interpretation $\pi_{\mathcal{P}}$ interprets σ_{IA} in $\widehat{\sigma_{\text{PAB}}}$, with the usual domain formula ∂_π. It follows the same approach as before; each defining formula is a qfpp $\widehat{\sigma_{\text{PA}}}$-formula that consists of prime implicates. This map gives defining formulas that are qfpp σ_{PA}-formulas for relation symbols in $\sigma_{\mathcal{P}}$.

The representation by Pham et al. [19] π_{Ph} interprets σ_{IA} in $\widehat{\sigma_{\text{PAB}}}$ with the usual domain formula ∂_π. For $R \in \sigma_{\mathcal{P}}$, let $\mu(R)$ be the corresponding CNF $\widehat{\sigma_{\text{PAB}}}$-formula consisting of *all* 4-conjuncts on $\{\ (v^{(i)}, w^{(j)}) \mid 1 \leq i \leq j \leq 2\ \}$. This gives, e.g., $\mu(\text{before})(\overline{v}, \overline{w}) = \bigwedge_{1 \leq i \leq j \leq 2} v^{(i)} < w^{(j)}$. To define $\varphi_{\pi_{\text{Ph}}}$ for arbitrary $R \in \sigma_{\text{IA}}$, let $\widehat{R} \in \sigma_{\mathcal{P}}$ be the smallest upper approximation of R in $\sigma_{\mathcal{P}}$, i.e., the relation symbol with smallest set $\widehat{R}^{\Gamma^{int(\mathbb{Q}),\sigma_{\text{IA}}}}$ such that $R^{\Gamma^{int(\mathbb{Q}),\sigma_{\text{IA}}}} \subseteq \widehat{R}^{\Gamma^{int(\mathbb{Q}),\sigma_{\text{IA}}}}$. Further, for such $R, \widehat{R}$, let $\psi(R)$ be the set of all symbols $r \in \sigma_{\text{IAB}}$ such that $r^{\Gamma^{int(\mathbb{Q}),\sigma_{\text{IA}}}} \subseteq (\widehat{R}^{\Gamma^{int(\mathbb{Q}),\sigma_{\text{IA}}}} \setminus R^{\Gamma^{int(\mathbb{Q}),\sigma_{\text{IA}}}})$. Then, $\varphi_{\pi_{\text{Ph}}}(R) := \mu(\widehat{R}) \wedge \bigwedge_{r \in \psi(R)} \neg\mu(r)$. Note, this map does not yield qfpp σ_{PAB}-formulas for relations in $\sigma_{\mathcal{P}\text{-PAB}}$, however the only non-unit clauses represent the universal relation.

B. Propagation for the IA and reducts

For σ_{IA} we define $\Pi^{\text{IA}}_{\leq 3}$ for strong 3-consistency similar as with the PA, and obtain the restricted programs $\Pi^{\text{IA}}_{\leq 3}|_{\widehat{G}}$, $\Pi^{\text{IAB}}_{\leq 3}$, and $\Pi^{\text{IAB}}_{\leq 3}|_{\widehat{G}}$ as before. We consider applying the already introduced PA Datalog programs to the defined translations. Further, we consider an extended version of the $\Pi^{\textit{ORD}}$ program, $\Pi^{\textit{ORD}}_{\textit{RES}}$ (and $\Pi^{\textit{ORD}}_{\textit{RES}}|_{\widehat{G}}$), which emulates positive unit resolution on all Horn clauses C in defining formulas of $\varphi_{\pi_{\textit{ORD-H}}}$ equivalent to the form $P(v, w) \vee \bigvee_i (v_i \neq w_i)$ via rules $P(v, w)$:- $C(v, w, \ldots), = (v_1, w_1), \ldots,$ with $P \in \{\leq, =\}$ or the head as false if the clause has no symbol $\{\leq, =\}$.

The propagation strength and refutation completeness of the programs with translations is given in Fig. 3.

Proposition 3. *The partial order shown in Fig. 3 is correct. It is further complete given the transitivity of* $\prec_{\Gamma^{int(\mathbb{Q}),\sigma_{IA}}}$.

Proof idea: Use the well-known reduction theorem [13]; formulas on intervals translate into equivalent formulas on points. This way rules of the programs can be compared. ■

Proposition 4. *1)* $\Pi^{\textit{PAB}}_{\leq 3}|_{\widehat{G}} \circ \pi_{\textit{Ph}}$ *is refutation-complete for* $\Gamma^{int(\mathbb{Q}),\sigma_{\mathcal{P}\text{-}PAB}}$*, 2)* $\Pi^{\textit{ORD}}|_{\widehat{G}} \circ \pi_{\textit{ORD-H}}$ *is refutation-complete for* $\Gamma^{int(\mathbb{Q}),\sigma_{\mathcal{P}}}$*, 3)* $\Pi^{\textit{ORD}}_{\textit{RES}}|_{\widehat{G}} \circ \pi_{\textit{ORD-H}}$ *is refutation-complete for* $\Gamma^{int(\mathbb{Q}),\sigma_{\textit{ORD-H}}}$*, 4)* $\Pi^{\textit{IAB}}_{\leq 3}$ *is* not *refutation-complete for* $\Gamma^{int(\mathbb{Q}),\sigma_{\textit{IAB}}}$*, 5)* $\Pi^{\textit{PAB}}_{\leq 3} \circ \pi_{\textit{Ph}}$ *is* not *refutation-complete for* $\Gamma^{int(\mathbb{Q}),\sigma_{\mathcal{P}}}$*, 6)* $\Pi^{\textit{PA}}_{\leq 3} \circ \pi_{\mathcal{P}}$ *is* not *refutation-complete for*

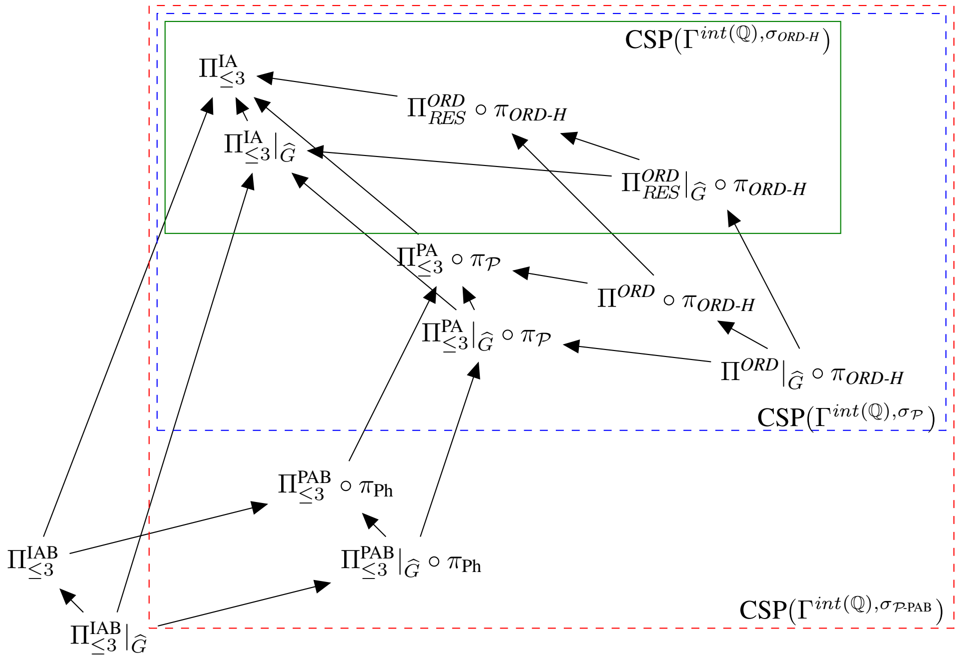

Fig. 3. Lattice of propagation strength of Datalog programs for IA. Frames indicate refutation completeness for reducts.

$\Gamma^{int(\mathbb{Q}),\sigma_{ORD\text{-}H}}$, *7) refutation completeness of other programs as indicated in Fig. 3 follows from the transitivity of* $\prec_{\Gamma^{int(\mathbb{Q}),\sigma_{IA}}}$.

Proof idea: 1), 2) from qfpp interpretations and Prop. 2. 3) combine argument of [18] (positive unit resolution) and Prop. 2. 4) $\Pi^{IAB}_{\leq 3}$ has disjunctive symbols in heads of some 3-rules, but is incapable of establishing 2-consistency on them. 5) from qfpp interpretations and Prop. 2. 6) $\Pi^{PA}_{\leq 3} \circ \pi_{\mathcal{P}}$ only works on symbols in $\sigma_{\mathcal{P}}$, but input instances have symbols in $\sigma_{ORD\text{-}H} \supsetneq \sigma_{\mathcal{P}}$. ■

V. Grounding to Propositional CNF

We show how to write Datalog programs with syntactic interpretations in propositional CNF. Alternatively, all programs can be used as propagators in Constraint Programming (CP) with the interpretations as modeling. We first sketch the classic reasoning approach using strong 3-consistency that forms the basis of so-called qualitative constraint-based solvers.

A. The classic QSTR approach

For a binary constraint language Γ with signature σ and a σ-Datalog program $\Pi_{\leq 3}$ that establishes strong 3-consistency, we model a given instance $I \in$ CSP(Γ) as a *finite* constraint problem where the constraint variables are pairs of elements from D^I, which take as values relation symbols of σ. We only need to express the most constraining relation on a pair, thus if $\sigma = 2^{\sigma_B} \setminus \{\emptyset\}$ for some σ_B, it is sufficient to define the domain of constraint variables as σ_B, i.e., constraint variables get assigned sets of base relations.

Given refutation completeness of $\Pi_{\leq 3}$ on CSP instances of some σ'-reduct Γ' of Γ, we can use domain splitting in any search strategy to restrict domains of constraint variables to take values in σ'. Thus, $\Pi_{\leq 3}$ immediately proves satisfiability or unsatisfiability once all constraint variables have been restricted to σ'. It is well-known that if Γ, Π stem from a relation algebra, maintaining strong 3-consistency is equivalent to maintaining generalized arc-consistency on table constraints [7], [24].

B. Straightforward translations to propositional CNF

Let σ' be the signature of a given instance I, π a syntactic interpretation of σ' in $\widehat{\sigma}$ and Π a $\widehat{\sigma}$-Datalog program. We consider effectively the Herbrand-expansion of the translated input $\pi(I)$ (existential quantification) and the Datalog rules (universal quantification). This gives propositional atoms: $\{ \, p_{R(v_1,\ldots,v_n)} \mid R \; n\text{-ary in } \sigma \text{ and } v_1,\ldots,v_n \in D^{\pi(I)} \, \}$. The instance $\pi(I)$ is written as propositional CNF by writing the conjunction of its $\widehat{\sigma}$ symbols on their interpretations, where the symbols themselves are written as the clauses they represent. We further instantiate the rules of Π on this set of propositional atoms, where we identify false with $\bot$ and for chordal variants immediately evaluate the special relation E. Assuming the signature σ and interpretation π are of fixed size, the size of the resulting CNF is polynomial in the size of the input.

If all heads of Π are symbols in σ, the rules can be written as Horn clauses and UP has at least the propagation strength of Π. In particular, this means UP retains refutation completeness. We note that there are trivial SAT encodings establishing strong 3-consistency, but they are costly in terms of CNF size. For $n := |\, D^{\pi(I)} \,|$ there are $O(n^k)$ distinct instantiations of a k-rule and without reductions $\Pi^{IA}_{\leq 3}$ contains ≈ 67 million 3-rules (cf. Π^{ORD} has a single 3-rule).

Given properties of relations and programs we can reduce the number of atoms and clauses without violating theoretical properties, e.g., relations R with $R^\Gamma = D^\Gamma$ can be identified with $\top$, 1-rules can be dropped if checking 1-consistency is done upon grounding. Atoms that are provably false given I can be identified with $\bot$. Further, easy examples are the symmetry of the equality relation $=$ and concurrence of relations and their converse relations on swapped arguments, e.g., $v < w$ if and only if $w > v$. We can easily reduce the set of propositional atoms in these cases. Additionally, for relations that constitute complements of each other we can assign a negated atom to one of them, e.g., in the case of σ_{ORD} where we have both $=$ and $\neq$. Several rules of Π^{ORD} and Π^{ORD}_{RES} become tautological statements in propositional logic and can be removed. Rules for positive unit resolution in Π^{ORD}_{RES} are thus removed and both Π^{ORD} and Π^{ORD}_{RES} have the same grounding.

C. The support encoding of Pham et al.

The perhaps most popular encodings were introduced by Pham et al. [19] who considered the PA and IA, but used the classic QSTR approach as the starting point and accordingly used existing techniques for mapping finite CSPs to SAT. We can modify our grounding strategy to sketch the *support* encoding defined by them. Note, their encoding requires properties of relation algebras and thus we limit ourselves to the programs $\Pi^{\mathrm{PAB}}_{\leq 3}$ and $\Pi^{\mathrm{IAB}}_{\leq 3}$. Pham et al. found their support encoding of $\Pi^{\mathrm{PAB}}_{\leq 3} \circ \pi_{\mathrm{Ph}}$ to be the best IA encoding compared to other encodings they proposed (not include here).

To obtain the support encoding we: 1) do not identify the universal relation with $\top$, instead we assume during translation that we have the universal relation on all pairs, 2) restrict 3-rules to instantiations on u, v, w where it holds $u \lhd v \lhd w$ for $\lhd$ a total order on $D^{\pi}(I)$, 3) identify all propositional atoms $R(v, w)$ for which $(v, w) \in R^{\pi(I)}$ would violate 2-consistency with $\bot$ to simplify the formula.

Note, 2) does not invalidate correctness of the encoding, because the first modification guarantees that between each pair of variables some atom denoting a relation must hold in a propositional model. Given that 3-rules stem from a relation algebra this guarantees that these models are valid.

We conclude with refutation properties of the support encoding for the considered programs. For this, maps $\pi_{\widehat{\mathrm{IAB}}}$, $\pi_{\widehat{\mathrm{PAB}}}$ write relation symbols as disjunctions of base relation symbols.

Proposition 5. *UP on the support encoding of: 1)* $\Pi^{PAB}_{\leq 3}|_{\widehat{G}}$ *is refutation-complete for* $\Gamma^{\mathbb{Q},\sigma_{PAB}}$*, 2)* $\Pi^{PAB}_{\leq 3} \circ \pi_{\widehat{\sigma_{PAB}}}$ *is* not *refutation-complete for* $\Gamma^{\mathbb{Q},\sigma_{PA}}$*, 3)* $\Pi^{IAB}_{\leq 3} \circ \pi_{\widehat{\sigma_{IAB}}}$ *is* not *refutation-complete for* $\Gamma^{\mathbb{Q},\sigma_{IAB}}$.

Proof idea: 1) Negative unit resolution substitutes the missing rules. 2) Several rules on $\widehat{\sigma_{\mathrm{PAB}}}$ are missing, thus refutations are not guaranteed. 3) Similar to 2). ■

As a consequence UP on the support encoding of $\Pi^{\mathrm{PAB}}_{\leq 3} \circ \pi_{\mathrm{Ph}}$ is refutation-complete for $\Gamma^{int(\mathbb{Q}),\sigma_{\mathcal{P}\text{-PAB}}}$, but not $\Gamma^{int(\mathbb{Q}),\sigma_{\mathcal{P}}}$.

We briefly summarize selected IA reasoning approaches built on Datalog programs in Table I and also list the CNF size of the first instance from the $A(150, 10.5, 6.5)$ set of benchmarks that is used in the evaluation below. Our encoding using $\pi_{ORD\text{-}H}$ is the smallest one as there is only one 3-rule.

VI. Empirical Evaluation

To conclude our study, we consider the empirical performance of the most promising SAT encodings: the Pham et al. grounding of $\Pi^{\mathrm{IAB}}_{\leq 3}|_{\widehat{G}} \circ \pi_{\widehat{\sigma_{\mathrm{IAB}}}}$, $\Pi^{\mathrm{PAB}}_{\leq 3}|_{\widehat{G}} \circ \pi_{\mathrm{Ph}}$, and our grounding with optimized encoding of $\Pi^{ORD}_{RES}|_{\widehat{G}} \circ \pi_{ORD\text{-}H}$. As benchmarks we consider random graphs as often used in this area. We use two different models, where the first model gives random instances around the phase transition that range from under-constrained to over-constrained problems, and the second model gives non-hard instances where chordal supergraphs have very low density. This should suffice to demonstrate the performance of the encodings on easy and hard problems.

The first random generator is commonly known as the A-model [17]. It has as parameters $n \in \mathbb{N}$ the number of nodes in the constraint graph, $d \in [0, n-1]$ its average degree, and $l \in [1, 13]$ the average label size. We initialize a graph with n nodes and randomly add edges from the complete graph with n variables until we have reached the average degree d. Finally, we assign randomly drawn labels from $\sigma_{\mathrm{IA}} \setminus \{int(\mathbb{Q})\}$ with an average size of l. We simply write $A(n, d, l)$ to denote a corresponding set of instances. In particular, we choose d to obtain instances from the phase transition.

The second random model we use builds constraint graphs according to the well-known Watts-Strogatz model [22] to which we assign random labels as before. We here denote the model by $W(n, d, \beta, l)$. Like the A-model it builds graphs with n nodes and an average degree of d. However, the graphs are initially constructed as local clusters and then each edge is randomly reassigned according to the probability β. For $\beta \ll 1.0$ this model has small-world properties, i.e., many clusters that are well connected and on average short length of paths. Resulting networks have sparser chordal supergraphs than those of the A-model. We use this model to provide larger but easy instances. Instances we generated from this model were solved with few failed decisions.

The used SAT solver is Glucose-2.2 [2]. As suggested by Pham et al. we preprocess input with strong 3-consistency, but our triangulations are based on the original input. Triangulation is done with the GreedyFillIn method [6] as suggested by Chmeiss and Condotta [9]. Runtime does not include the time necessary for this preprocessing nor the translation via syntactic interpretation as this overhead is insignificant given the 2 hour time limit. The tool we developed to compute the SAT encodings is available.[1]

GQR-1500 [23] is a generic qualitative constraint reasoner written for arbitrary relation algebras that maintains strong 3-consistency for propagation. It here provides the results of $\Pi^{\mathrm{IA}}_{\leq 3}$ with the ORD-Horn relations for domain splitting. These are here only listed to put all results in the context of a classic strong 3-consistency CP approach.

All experiments were run on an Intel Xeon CPU with 2.66 GHz, 4 GB memory, and a CPU time limit of 2 hours.

[1] See the publication's entry at http://www.informatik.uni-freiburg.de/~ki/publications/

TABLE I
OVERVIEW OF SELECTED IA REASONING APPROACHES.

Approach (references)	Syntactic interpretation	Propagation	CNF size of first instance in $A(150, 10.5, 6.5)$ variables	clauses
Classic CP approach	π_{id}	$\Pi^{\mathrm{IA}}_{\leq 3}$	NA	
CP with "partial weak consistency" ([9])	π_{id}	$\Pi^{\mathrm{IA}}_{\leq 3}\vert_{\widehat{G}}$	NA	
Pham-IAB ([19])	$\pi_{\widehat{\sigma_{\mathrm{IAB}}}}$	Modified grounding $\Pi^{\mathrm{IAB}}_{\leq 3}$	138 239	85 221 614
Pham-IAB-chordal ([19], [4], [16])	$\pi_{\widehat{\sigma_{\mathrm{IAB}}}}$	Modified grounding $\Pi^{\mathrm{IAB}}_{\leq 3}\vert_{\widehat{G}}$	47 599	13 462 761
Pham-PAB ([19])	π_{Ph}	Modified grounding $\Pi^{\mathrm{PAB}}_{\leq 3}$	134 250	38 207 793
Pham-PAB-chordal ([19], [4], [16])	π_{Ph}	Modified grounding $\Pi^{\mathrm{PAB}}_{\leq 3}\vert_{\widehat{G}}$	49 626	6 174 476
ORD-Horn ([18])	$\pi_{\mathit{ORD\text{-}H}}$	Grounding $\Pi^{\mathit{ORD}}_{\mathit{RES}}$	134 550	26 870 352
ORD-Horn-chordal ([18], here)	$\pi_{\mathit{ORD\text{-}H}}$	Grounding $\Pi^{\mathit{ORD}}_{\mathit{RES}}\vert_{\widehat{G}}$	49 926	4 474 722

TABLE II
RUNTIME IN SECONDS FOR $A(150, d, 6.5)$

d	name	percentile 25th	50th	75th	90th
10.0	GQR-1500	1.64	6.06	30.27	179.38
	ORD-Horn-chordal	24.69	53.58	130.73	291.77
	Pham-PAB-chordal	112.45	185.64	330.83	587.48
	Pham-IAB-chordal	566.03	965.00	1 557.33	2 692.75
10.5	GQR-1500	9.02	78.43	663.80	4 107.85
	ORD-Horn-chordal	63.68	227.02	743.84	1 968.07
	Pham-PAB-chordal	151.87	423.19	1 066.77	2 739.81
	Pham-IAB-chordal	301.41	1 445.11	3 607.48	-
11.0	GQR-1500	1.82	35.19	578.80	6 633.01
	ORD-Horn-chordal	14.99	103.02	615.40	2 373.00
	Pham-PAB-chordal	34.71	182.19	958.14	3 094.41
	Pham-IAB-chordal	49.86	295.69	1 919.98	-
11.5	GQR-1500	0.27	2.53	28.40	285.62
	ORD-Horn-chordal	5.27	18.27	86.66	352.47
	Pham-PAB-chordal	9.75	37.21	159.74	550.22
	Pham-IAB-chordal	14.82	51.68	230.27	948.78

TABLE III
RUNTIME IN SECONDS FOR $W(n, 10.0, 0.1, 7.5)$

n	name	percentile 25th	50th	75th	90th
500	GQR-1500	3.49	3.66	3.90	6.47
	ORD-Horn-chordal	18.09	21.41	24.89	28.42
	Pham-PAB-chordal	94.88	128.70	180.69	220.14
750	GQR-1500	11.58	12.58	21.30	28.16
	ORD-Horn-chordal	62.97	75.13	89.67	103.89

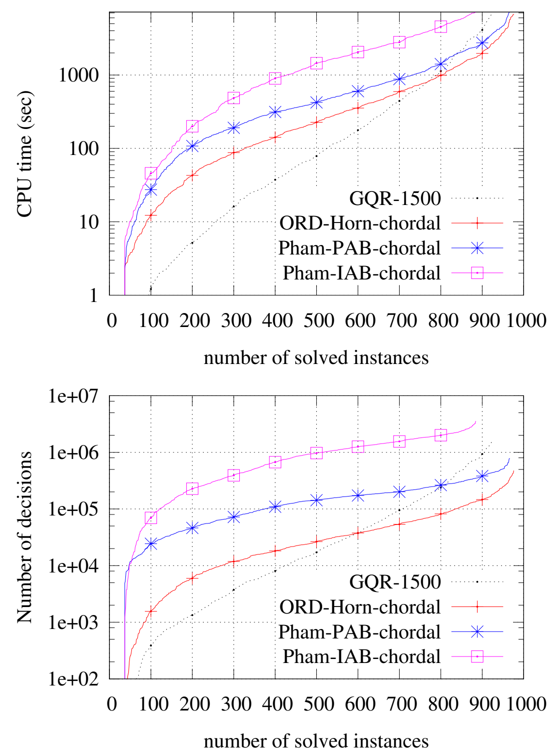

Fig. 4. Runtime and decisions for $A(150, 10.5, 6.5)$ (both logscale).

For each considered parameter setting of the A-model we generated $1\,000$ problem instances. Table II summarizes the runtime results. Fig. 4 gives more detailed results for the set with the overall hard instances. One can notice the stronger propagation of $\Pi^{\mathit{ORD}}_{\mathit{RES}}\vert_{\widehat{G}}$ pays off in terms of both runtime and decisions. Further, we can see that Glucose spends more time on propagation with $\Pi^{\mathit{ORD}}_{\mathit{RES}}\vert_{\widehat{G}}$ than with the Pham encoding of $\Pi^{\mathrm{PAB}}_{\leq 3}\vert_{\widehat{G}}$. The runtime results of the ORD-Horn-chordal encoding are better than those for Pham-PAB-chordal, and for most instances are an improvement of factor 2.

For the W-model, we generated 100 instances for each considered parameter setting. Table III lists the results for these instances. Here, using ORD-Horn is overall better than the Pham encoding by a factor of 5 to 8. Note, Glucose exceeded the available memory with the Pham-IAB-chordal encoding on both sets, with Pham-PAB-chordal on the last set, and with ORD-Horn-chordal on five instances of the last set.

VII. RELATED WORK

The explicit use of FO interpretations was presented in [3] to analyze pointizable relations. Less explicitly it has been applied in many works to study the properties of pointizable and also ORD-Horn relations. We have used syntactic interpretations to derive propositional encodings. We have further used FO interpretations to study the differences between distinct constraint propagation approaches. The Datalog approach ties in with important theoretic results on reasoning with relational structures on infinite domains, in particular k-consistencies [4]. We have here aimed at refutation completeness with constraint propagation weaker than strong 3-consistency as this is more suited for SAT encodings.

The propositional SAT encoding of $\Pi^{ORD}_{RES} \circ \pi_{ORD}$ has been used almost 20 years ago in [18] to prove the tractability of $\sigma_{ORD\text{-}H}$. However, it does seem to not have attracted any interest for practical use as a SAT encoding. We have optimized and combined this encoding with restricted evaluations on chordal graphs. The encoding remains refutation-complete for the tractable class of ORD-Horn relations. We can also mention an encoding proposed in [10] which takes advantage of convex relations – a strict subset of ORD-Horn relations.

Chordal variants of encodings trace back to [16] where the support encoding of Pham et al. on points was combined with a chordal restriction; there referred to as "divide-and-conquer". It was introduced as tree decompositions and chordal graphs later in [11]. Structural properties have however been considered before: Bodirsky and Dalmau already discussed tree decompositions on ω-categorical constraint languages [4], van Beek [20] proposed finding cycles using the strongly connected components for the PA. Further, temporally labeled graphs defined by Gerevini and Schubert [12] make use of both the σ_{ORD}-reduct of the PA and sparse graphs by not necessarily deriving relations on all pairs of D^I but rather following longer chains of orderings for cycle checking. On easy sparse instances propagation with such specialized algorithms [20], [12], [5] can be expected to be more beneficial than (full) constraint propagation.

VIII. Conclusion and Future Work

In this work we have studied existing constraint-based approaches for reasoning with the Point Algebra and Allen's Interval Algebra. These approaches are based on at most strong 3-consistency and we provided lattices showing propagation strength and refutation completeness for different fragments of both formalisms. Although all these approaches have been applied for reasoning in the literature, it seems that no explicit statement on propagation strength has been provided before. In particular, we used previous results on FO interpretations to make the comparison formally explicit.

We further analyzed and improved existing SAT encodings of these formalisms. One of the oldest propositional SAT encodings seems to have gone unnoticed when other encodings were put forward, but its strong theoretical and empirical results seem superior to newer encodings. We have combined this encoding with propagation on chordal graphs which turns out to be the best encoding in our evaluation.

Our results are easily applicable to modeling of qualitative reasoning problems in Constraint Programming. Further, other point-based formalisms could be reduced to these encodings using syntactic interpretations.

For future work it should be useful to combine the ideas presented here with earlier work on purely syntactical interpretations of arbitrary qualitative relation algebra formalisms [14]. It should further be worthwhile to consider the existing SAT encodings of other formalisms, e.g., the Region Connection Calculus in the same way as the temporal formalisms in here.

Acknowledgment

We thank the anonymous reviewers for helpful comments. This work was supported by DFG (Transregional Collaborative Research Center *SFB/TR 8 Spatial Cognition*, project R4-[LogoSpace]).

References

[1] James F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983.
[2] Gilles Audemard and Laurent Simon. Predicting learnt clauses quality in modern SAT solvers. In *IJCAI Proceedings*, pages 399–404, 2009.
[3] Manuel Bodirsky and Hubie Chen. Qualitative temporal and spatial reasoning revisited. *Journal of Logic and Computation*, 19(6):1359–1383, 2009.
[4] Manuel Bodirsky and Víctor Dalmau. Datalog and constraint satisfaction with infinite templates. In *STACS Proceedings (LNCS 3884)*, pages 646–659, 2006.
[5] Manuel Bodirsky and Jan Kára. A fast algorithm and Datalog inexpressibility for temporal reasoning. *ACM Transactions on Computational Logic*, 11(3), 2010.
[6] Hans L. Bodlaender and Arie M. C. A. Koster. Treewidth computations I. Upper bounds. *Information and Computation*, 208(3):259–275, 2010.
[7] Sebastian Brand. Relation variables in qualitative spatial reasoning. In *KI Proceedings (LNCS 3238)*, pages 337–350, 2004.
[8] Stefano Ceri, Georg Gottlob, and Letizia Tanca. What you always wanted to know about Datalog (and never dared to ask). *IEEE Transactions on Knowledge and Data Engineering*, 1(1):146–166, 1989.
[9] Assef Chmeiss and Jean-François Condotta. Consistency of triangulated temporal qualitative constraint networks. In *ICTAI Proceedings*, pages 799–802, 2011.
[10] Jean-François Condotta and Dominique D'Almeida. Qualitative constraints representation for the time and space in SAT. In *ICTAI Proceedings*, pages 74–77, 2007.
[11] Jean-François Condotta and Dominique D'Almeida. Consistency of qualitative constraint networks from tree decompositions. In *TIME Proceedings*, pages 149–156, 2011.
[12] Alfonso Gerevini and Lenhart K. Schubert. Efficient algorithms for qualitative reasoning about time. *Artificial Intelligence*, 74(2):207–248, 1995.
[13] Wilfrid Hodges. *Model Theory*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1993.
[14] Julien Hué, Matthias Westphal, and Stefan Wölfl. An automatic decomposition method for qualitative spatial and temporal reasoning. In *ICTAI Proceedings*, pages 588–595, 2012.
[15] Peter B. Ladkin and Roger D. Maddux. On binary constraint networks. Technical report, Kestrel Institute, Palo Alto, California, 1988.
[16] Jason Jingshi Li, Jinbo Huang, and Jochen Renz. A divide-and-conquer approach for solving interval algebra networks. In *IJCAI Proceedings*, pages 572–577, 2009.
[17] Bernhard Nebel. Solving hard qualitative temporal reasoning problems: Evaluating the efficiency of using the ORD-Horn class. *Constraints*, 1(3):175–190, 1997.
[18] Bernhard Nebel and Hans-Jürgen Bürckert. Reasoning about temporal relations: A maximal tractable subclass of Allen's Interval Algebra. *Journal of the ACM*, 42(1):43–66, 1995.
[19] Duc Nghia Pham, John Thornton, and Abdul Sattar. Towards an efficient SAT encoding for temporal reasoning. In *CP Proceedings (LNCS 4204)*, pages 421–436, 2006.
[20] Peter van Beek. Reasoning about qualitative temporal information. *Artificial Intelligence*, 58(1-3):297–326, 1992.
[21] Marc Vilain, Henry Kautz, and Peter van Beek. Constraint propagation algorithms for temporal reasoning. In *Readings in Qualitative Reasoning about Physical Systems*, pages 377–382. Morgan Kaufmann, 1986.
[22] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, June 1998.
[23] Matthias Westphal and Julien Hué. Nogoods in qualitative constraint-based reasoning. In *KI Proceedings (LNCS 7526)*, pages 180–192, 2012.
[24] Matthias Westphal and Stefan Wölfl. Qualitative CSP, finite CSP, and SAT: Comparing methods for qualitative constraint-based reasoning. In *IJCAI Proceedings*, pages 628–633, 2009.