Compact Answers to Temporal Path Queries (extended version)

No Institute Given

Abstract. We study path-based graph queries that in addition to navigation over edges also perform navigation over time. This allows one to ask questions about the dynamics of networks, like traffic movement, cause effect relationships, or spread of diseases. In this setting, graphs consist of triples annotated with validity intervals while a query produces pairs of nodes where each pair is associated with a binary relation over time. For instance, such a pair could be two airports and the relation could map potential departure times to possible arrival times. An open question is how to represent such a relation in a compact form and how to maintain such properties during query evaluation. We investigate four compact representations of answers to a such a query that rely on alternative ways of encoding sets of intervals over both discrete and dense time. We discuss their respective advantages and drawbacks, in terms of conciseness, uniqueness, and computational cost. Notably, the most refined encoding guarantees that query answers over dense time can be finitely represented.

Table of Contents

2

1 Introduction

Temporal databases [5] have traditionally focused on computing relations where each tuple is annotated with a *single* time point (or single interval) for validity. In comparison, little attention has been paid to relations over time points, which indicate how different events are related temporally. This is the focus of this paper, more precisely binary relations over time points.

One example is linking potential departure times of a road trip to their corresponding arrival times, considering uncertainties like traffic. Such a relation matches each departure time with a range of possible arrival times, or equivalently each potential arrival time back to possible departures. Another example is modelling cause-effect scenarios with uncertain delays. For instance, one may compute the relation that associates the potential malfunction a component in an airplane to the subsequent malfunction of another component. A third application of binary temporal relations, which we will use as a running example, is modeling the spread of phenomena such as messages or diseases. For instance, in epidemiology, the latency and infectiousness periods of a virus induce a temporal relation that maps potential infection times to possible subsequent transmissions, which allows modeling disease propagation within a population.

These examples suggest at least three elementary operations that one may want to perform over such relations: *(i) filtering* a temporal relation (based on some knowledge about events that took place), *(ii) combining* two temporal relations, and *(iii) composing* them (e.g. to model transitive cause-effect or spread). Or in database terms, selection, union and join respectively. In particular, a fundamental question is how one can *represent* a temporal binary relation to start with, in a compact way (or simply in a finite way over dense time), and whether compactness (resp. finiteness) is preserved by selection, union and/or join. This is the main problem addressed in this paper.
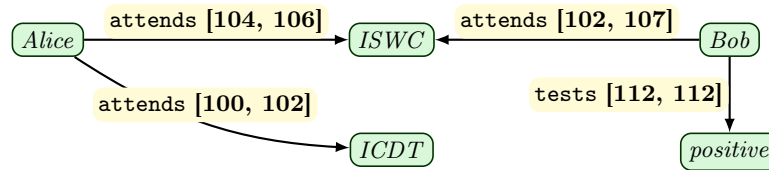

Fig. 1: A temporal graph

In this work, we consider the query language recently introduced in [2], which computes and performs operations on binary temporal relations, along with a graph data model used to filter such relations. For presentation purposes, we simplify the language and data model to a fragment that is essentially Regular Path Queries (RPQs) over time-labeled graphs,[1] used for navigation over the graph and equipped with a so-called *temporal operator*, which enables navigation in time by a certain *range*. We note that, in principle, our results are not limited to this language and could potentially apply to other (graph) query languages, as soon as they support edge concatenation (join) and allow for temporal navigation.

An RPQ q is a regular expression, and a pair $\langle n_1, n_2 \rangle$ of nodes is an answer to q over a graph G if there exists a path from n_1 to n_2 in G whose concatenated labels match this regular expression. The language of [2], called *Temporal RPQs* (TRPQs), extends RPQs with a temporal operator that allows navigation from one node at a certain time to the same node in past or future moments. Accordingly, an answer is a pair $\langle \langle n_1, t_1 \rangle, \langle n_2, t_2 \rangle \rangle$, where n_1 and n_2 are each associated with a time point, which mark the starting and arrival time of the temporal navigation respectively.

More precisely, this temporal operator, which we denote as $\mathbf{T}_\delta$ here, allows navigation in time within the range specified as the interval δ. For instance, the latency and infectiousness periods of a virus may imply that a person can only become infectious at least three days after the infection time t, and remain infectious at most five days after t. If $\mathcal{T}$ is our temporal domain, this induces a binary temporal relation

$$R = \{(t, t + d) \mid t \in \mathcal{T} \text{ and } d \in [3, 5]\}.$$

Accordingly, if $\mathcal{N}_G$ is the set of nodes in the queried graph G, then the query $\mathbf{T}_{[3,5]}$ outputs all pairs $\langle \langle n, t_1 \rangle, \langle n, t_2 \rangle \rangle$ such that $n \in \mathcal{N}_G$ and $(t_1, t_2) \in R$. This output can be joined and filtered based on events registered in G, effectively restricting the initial temporal relation R, and associating each returned time point with a meaningful node. For instance, the graph of Figure 1 represents data about attendance at a conference, where each fact is a labelled edge, annotated with an interval for validity. We can extend our query accordingly, as follows

$$q_1 = \mathbf{T}_{[3,5]}/\texttt{attends}/\texttt{attends}^-.$$

This query outputs all pairs $\langle \langle n_1, t_1 \rangle, \langle n_2, t_2 \rangle \rangle$ such that n_1 and n_2 attended a same event, and n_1 may have transmitted the virus to n_2 (during this event) at time t_2 if n_1 got infected at time t_1. These include the two pairs $\langle \langle Alice, 100 \rangle, \langle Bob, 104 \rangle \rangle$, and $\langle \langle Alice, 100 \rangle, \langle Bob, 105 \rangle \rangle$, meaning that if Alice contracted the virus on day 100, then she may have transmitted it to Bob during day 104 or day 105 (among other possibilities). In this query, the "$/$" operator is a join, and the subqueries $\texttt{attends}$ and $\texttt{attends}^-$ act as filters on our initial temporal

---

[1] RPQs are a central building block of navigational graph query languages such as Cypher [15] and SPARQL [17].

relation R, restricting it based on the time intervals contained in the graph. We can further restrict this relation by requiring that person n_2 tested positive at most a week after the possible transmission, with

$$q_2 = q_1/?\Big(\mathbf{T}_{[0,7]}/\texttt{tests}/(=\textit{positive})\Big).$$

In this query, the operator $?q$ acts as an existential quantifier: it only requires the existence of a match for the subquery q, so that the output of q_2 is a subset of the output of q_1.

As we have seen above, our initial temporal relation R can be easily represented in a compact way, using the temporal domain $\mathcal{T}$ for the set of initial infection points, and a single interval of temporal distances $[3, 5]$ that indicates by how much each initial time point can be shifted. This holds regardless of whether $\mathcal{T}$ is discrete or continuous.

However, this does not hold anymore for the outputs of our queries. For instance, the temporal relation induced by the answers to q_2 (which all have *Alice* and *Bob* as first and second node respectively) is uneven: over discrete time (and with a granularity of one day), 100 is associated with a single day 105, whereas 101 is associated with both 105 and 106. Moreover, without a compact representation of query answers, altering the time granularity (e.g. from days to hours) may significantly increase the number of output tuples (exponentially in the size of the input, assuming that interval boundaries are encoded in binary). Besides, even in the case where the output of a query can be represented compactly, computing all answers before summarizing them may be prohibitive, because the number of intermediate results may impact the performance of (worst-case quadratic) join operations. Hence the need to not only represent answers in a compact way, but also maintain compactness during query evaluation. This is even a necessity over dense time, where the lack of a finite representation may forbid query evaluation. To our knowledge, these are still open questions (in particular, left open in [2]), which effectively rule out arbitrary selection, union and joins over binary temporal relations.

Contributions and Organization. In this work, we propose four alternative compact representations of answers to TRPQs. In Section 2, we provide an informal overview of our four representations. In Section 3, we define the basic notation, followed by the syntax and semantics of the query language. In Section 4, we introduce our four representations: first, in Section 4.1, we present two simpler ones that aggregate tuples along a single time dimension (e.g., distance); then, in Section 4.2, we present two more complex ones that aggregate tuples along two temporal dimensions. We analyze the respective advantages and drawbacks of each representation in terms of finiteness (over dense time), compactness (when finite), uniqueness, and the computational cost of query answering and minimizing a set of tuples. Notably, the fourth (more complex) representation guarantees that compactness and finiteness are maintained throughout query evaluation. In Section 5, we discuss related work, and conclude in Section 6.

2   Overview

This section provides a high-level overview of the four compact representations of answers to TRPQs defined and studied in this article. As a running example, the following query q_3 outputs all pairs $\langle\langle n_1, t_1\rangle, \langle n_2, t_2\rangle\rangle$ such that, if Alice got infected at time t_1 while attending event n_1, then she may have transmitted the virus at time t_2 while attending event n_2:

$$q_3 = \texttt{attends}^-/(= Alice)/\mathbf{T}_{[3,5]}/\texttt{attends}.$$

Over discrete time, with days as temporal granularity, evaluating q_3 over the graph G of Figure 1 outputs a set $[\![q_3]\!]_G$ of 7 pairs, each of the form $\langle\langle ICDT, t_1\rangle, \langle ISWC, t_2\rangle\rangle$. Since the two nodes ($ICDT$ and $ISWC$) are identical for all answers, we can focus on the binary temporal relation induced by these, i.e. the relation $R = \{(t_1, t_2) \mid \langle\langle ICDT, t_1\rangle, \langle ISWC, t_2\rangle\rangle \in [\![q_3]\!]_G\}$. Each pair $(t_1, t_2) \in R$ can equivalently be represented as the pair $(t_1, t_2 - t_1)$, where the second component is the temporal distance between t_1 and t_2. Accordingly, Figure 2a displays these 7 pairs over the Euclidean plane of (discrete) time per distance.
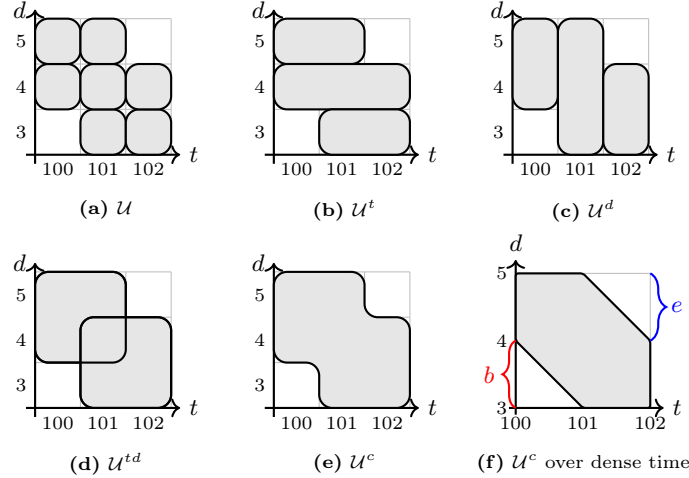

Fig. 2: Alternative representations of the answers to Query q_3, in the plane of time per distance

Each of the four compact representations that we study in this article is an alternative way to reduce the number of such pairs. The first representation, which we call $\mathcal{U}^t$, groups these pairs by distance (i.e. horizontally in our figure), while aggregating time points as intervals. In this example, this yields three compact answers, namely $([100, 101], 5)$, $([100, 102], 4)$, and $([101, 102], 3)$. Each of these is a rectangle with height 1 in our plane, as illustrated with Figure 2b, and their union is indeed the area of interest. The second representation, called

$\mathcal{U}^d$, is symmetric, in the sense that it groups results by time points and aggregates distances, so that compact answers are now rectangle with width 1, as illustrated with Figure 2c. The first representation $\mathcal{U}^t$ ensures that the number of compact answers is independent of the size of the intervals present in the graph. However, this number is linear in the size of the intervals used in the query (via the temporal navigation operator). And conversely for $\mathcal{U}^d$.

A third natural attempt (which we call $\mathcal{U}^{td}$) consists in aggregating tuples along both dimensions, representing our binary relation R as a set of pairs of intervals, one of time points, the other of distances. These correspond to arbitrary rectangles in our plane. As can be seen on Figure 2d, two such rectangles are still needed to cover the whole area (or three if we forbid overlapping rectangles). Besides, some desirable properties of the first two representations are lost: the most compact representation (i.e. smallest set of rectangles needed to cover the area) is not unique anymore (e.g. if the area is an "L"-shaped polygon), and minimizing the number of answers (i.e. finding a minimal set of such rectangles) becomes intractable (if we allow overlapping rectangle).

Moreover, for all three representations, the number of tuples needed to aggregate all answers to q_3 may grow with a change of time granularity. This can once again be seen in Figure 2a: if we adopt hours rather than days, then the number of "steps" (bottom left and top right of the area) gets multiplied by 24. At the limit (i.e. when granularity approaches 0), we reach dense time, and the area to cover becomes a rectangle cropped by two lines with slope -1 (drawn in Figure 2f). Clearly, such an area cannot be covered by finitely many rectangles.

This is why we introduce a fourth representation, which we call $\mathcal{U}^c$. Over dense time, a tuple in $\mathcal{U}^c$ precisely stands for an area like the one of Figure 2f, by means of two intervals (one for times, the other for distance), which intuitively represent the rectangle to be cropped, together with two values b and e that specify where the two cropping lines intersect respectively with this rectangle. This representation is also useful over discrete time: in our example, the whole area can be captured with a single tuple that consists of the interval $[100, 102]$ for times, as illustrated with Figure 2e, the interval $[3, 5]$ for distance, and the values 1 and 1 for b and e. Our most technical result is that binary relations that can be represented as such geometric shapes are closed under composition. As a consequence, a TRPQ can only produce a union of such shapes. So this last format $\mathcal{U}^c$ overcomes the limitations of the previous ones, in the sense that answers to a TRPQ can always be represented in a finite way. This is also the most compact of these four representations. The price to pay however is an arguably less readable format. Besides, minimizing a set of tuples under this view is still intractable.

Our main findings (in terms of finiteness, compactness, uniqueness of representation, cost of minimization and query answering) about each of the four representations are summarized at the end of the dedicated section, in Figure 6.

3 Preliminaries

Temporal Graphs. For maximal generality (and readability), we use a very simple graph data model, where each fact is a labelled edge, annotated with a set of time intervals for validity. Formally, we assume two infinite sets $\mathcal{N}$ of nodes and $\mathcal{E}$ of edge labels. As is conventional (e.g. in RDF), we represent a fact as a triple $(s, p, o) \in \mathcal{N} \times \mathcal{E} \times \mathcal{N}$ (where s, p and o intuitively stand for "subject", "property" and "object" respectively). In addition, for the temporal dimension of our data, we assume an underlying temporal domain $\mathcal{T}$ that may be either discrete or dense. For simplicity, we will use $\mathbb{Z}$ in the former case, and $\mathbb{Q}$ in the latter. We also use $\mathsf{intv}(\mathcal{T})$ for the set of nonempty intervals over $\mathcal{T}$.

In our model, a database instance simply assigns (finitely many) intervals of validity to (finitely many) triples. Precisely, a *temporal graph* $G = \langle \mathcal{T}_G, \mathcal{F}_G, \mathsf{val}_G \rangle$ consists of a bounded *effective temporal domain* $\mathcal{T}_G \in \mathsf{intv}(\mathcal{T})$, a finite set of triples $\mathcal{F}_G \subseteq \mathcal{N} \times \mathcal{E} \times \mathcal{N}$, and a function $\mathsf{val}_G \colon \mathcal{F}_G \to 2^{\mathsf{intv}(\mathcal{T}_G)}$ that maps a triple to a finite set of intervals over $\mathcal{T}_G$. For instance, in Figure 1, val_G assigns the (singleton) set of intervals $\{[104, 106]\}$ to the triple $(Alice, \mathtt{attends}, ISWC)$.

Temporal Regular Path Queries. We adopt the query language introduced in [2], but in a simpler form, made possible by the simplified data model defined above. We emphasise that this is without loss of expressivity.

A *Temporal Regular Path Query* (TRPQ) is an expression for the symbol "trpq" in the following grammar:

$$\mathsf{trpq} ::= \mathsf{edge} \mid \mathsf{node} \mid \mathbf{T}_\delta \mid (\mathsf{trpq}/\mathsf{trpq}) \mid (\mathsf{trpq} + \mathsf{trpq}) \mid \mathsf{trpq}[m, n] \mid \mathsf{trpq}[m, \_]$$

$$\mathsf{edge} ::= label \mid \mathsf{edge}^-$$

$$\mathsf{node} ::= pred \mid \leq k \mid (?\mathsf{trpq}) \mid \neg \, \mathsf{node}$$

with $label \in \mathcal{E}$, $k \in \mathcal{T}, \delta \in \mathsf{intv}(\mathcal{T})$, $m, n \in \mathbb{N}^+$, and $m \leq n$.

Here, edge and node are filters on edges and nodes respectively. The terminal symbol *pred* stands for a Boolean predicate that can be evaluated locally for one node n, which we write $n \models pred$ (for instance, in Figure 1, the node *positive* satisfies the predicate ($= positive$)). Similarly, the Boolean predicate $\leq k$ evaluates whether a time point is less than or equal to k. The expression $(?\mathsf{trpq})$ filters the nodes that satisfy trpq, and $\neg$ represents logical negation.

Finally, the temporal navigation operator $\mathbf{T}_\delta$ stands for navigation in time by any distance in the interval δ. The other operators on trpq are regular path query (RPQ) operators: / for join, + for union, and $\mathsf{trpq}[m, n]$ denotes repetition of trpq from m to n times. In particular, $\mathsf{trpq}[0, \_]$ represents Kleene closure (equivalent to the $*$ operator in regular expressions).

The formal semantics of TRPQs is provided in Figure 3, where $[\![q]\!]_G$ is the evaluation of a TRPQ q over a temporal graph G. In this definition, we use q^i for the TRPQ defined inductively by $q^1 = q$ and $q^{i+1} = q^i/q$. For convenience, we represent (w.l.o.g.) an answer as two nodes together with a time point and a distance, rather than two nodes and two time points, i.e. we use tuples of the form $\langle n_1, n_2, t, d \rangle$ rather than $\langle \langle n_1, t \rangle, \langle n_2, t + d \rangle \rangle$. We also use $\mathcal{N}_G$ to denote the

$$\begin{aligned}
\llbracket label \rrbracket_G &= \{\langle n_1, n_2, t, 0 \rangle \mid t \in \tau \text{ for some } \tau \in \mathsf{val}_G(n_1, label, n_2)\} \\
\llbracket \mathsf{edge}^- \rrbracket_G &= \{\langle n_2, n_1, t, 0 \rangle \mid \langle n_1, n_2, t, 0 \rangle \in \llbracket \mathsf{edge} \rrbracket_G\} \\
\llbracket pred \rrbracket_G &= \{\langle n, n, t, 0 \rangle \mid n \models pred \text{ and } t \in \mathcal{T}_G\} \\
\llbracket \leq k \rrbracket_G &= \{\langle n, n, t, 0 \rangle \mid n \in \mathcal{N}_G, t \in \mathcal{T}_G \text{ and } t \leq k\} \\
\llbracket \mathbf{T}_\delta \rrbracket_G &= \{\langle n, n, t, d \rangle \mid n \in \mathcal{N}_G, t \in \mathcal{T}_G, d \in \delta \text{ and } t + d \in \mathcal{T}_G\} \\
\llbracket ?\mathsf{trpq} \rrbracket_G &= \{\langle n, n, t, 0 \rangle \mid \langle n, n', t, d \rangle \in \llbracket \mathsf{trpq} \rrbracket_G \text{ for some } n' \in \mathcal{N} \text{ and } d \in \mathcal{T}\} \\
\llbracket \neg \mathsf{node} \rrbracket_G &= \left(\{\langle n, n \rangle \mid n \in \mathcal{N}_G\} \times \mathcal{T}_G \times \{0\}\right) \setminus \llbracket \mathsf{node} \rrbracket_G \\
\llbracket \mathsf{trpq}_1/\mathsf{trpq}_2 \rrbracket_G &= \{\langle n_1, n_3, t, d_1 + d_2 \rangle \mid \\
&\qquad \exists n_2 \colon \langle n_1, n_2, t, d_1 \rangle \in \llbracket \mathsf{trpq}_1 \rrbracket_G \text{ and } \langle n_2, n_3, t + d_1, d_2 \rangle \in \llbracket \mathsf{trpq}_2 \rrbracket_G\} \\
\llbracket \mathsf{trpq}_1 + \mathsf{trpq}_2 \rrbracket_G &= \llbracket \mathsf{trpq}_1 \rrbracket_G \cup \llbracket \mathsf{trpq}_2 \rrbracket_G \\
\llbracket \mathsf{trpq}[m, n] \rrbracket_G &= \bigcup_{k=m}^{n} \llbracket \mathsf{trpq}^k \rrbracket_G \\
\llbracket \mathsf{trpq}[m, \_] \rrbracket_G &= \bigcup_{k \geq m} \llbracket \mathsf{trpq}^k \rrbracket_G
\end{aligned}$$

Fig. 3: Semantics of TRPQs

set of nodes that intuitively appear in G, i.e. all n such that triple (n, p, o) or (s, p, n) is in $\mathcal{F}_G$ for some s, p and o.

Operations on intervals. If $\alpha \in \mathsf{intv}(\mathbb{Z})$ (resp. $\mathsf{intv}(\mathbb{Q})$) and $d \in \mathbb{Z}$ (resp. $\mathbb{Q}$), then we use $\alpha + d$ (resp. $\alpha - d$) for the interval $\{t + d \mid t \in \alpha\}$ (resp. $\{t - d \mid t \in \alpha\}$). Similarly, if α and β are intervals, then we use $\alpha + \beta$ for the interval $\{t + d \mid t \in \alpha, d \in \beta\}$. We also use $\alpha \ominus \beta$ for the interval $\alpha + \{-t \mid t \in \beta\}$.

4   Compact answers

In this section, we define and study the four compact representations of answers to a TRPQ sketched in Section 2. Let $\mathcal{U}$ denote the universe of all tuples that may be output by TRPQs, i.e. $\mathcal{U} = \mathcal{N} \times \mathcal{N} \times \mathcal{T} \times \mathcal{T}$. Then each of our four compact representations can be viewed as an alternative format to encode subsets of $\mathcal{U}$.

We specify each of these four formats as a set of admissible tuples, denoted $\mathcal{U}^t, \mathcal{U}^d, \mathcal{U}^{td}$ and $\mathcal{U}^c$ respectively. Let $\mathcal{U}^x$ be any of these four sets. A tuple $\mathbf{u}$ in $\mathcal{U}^x$ represents a subset of $\mathcal{U}$, which we call the *unfolding* of $\mathbf{u}$. And the unfolding of a *set* $U \subseteq \mathcal{U}^x$ of such tuples is the union of the unfoldings of the elements of U. We say that U is *compact* if it is finite and if no strictly smaller (w.r.t. cardinality) subset of $\mathcal{U}^x$ has the same unfolding. A set $V \subseteq \mathcal{U}$ can be *finitely represented* (in $\mathcal{U}^x$) if there is a finite $U \subseteq \mathcal{U}^x$ with unfolding V.

4.1   Folding time points ($\mathcal{U}^t$) or distances ($\mathcal{U}^d$)

Each of our two first compact representations aggregates tuples in $\mathcal{U}$ along one dimension: either the time point associated to the first node, or the distance between times points associated to each node. The corresponding universes $\mathcal{U}^t$ and $\mathcal{U}^d$ of tuples are $\mathcal{N} \times \mathcal{N} \times \mathsf{intv}(\mathcal{T}) \times \mathcal{T}$ and $\mathcal{N} \times \mathcal{N} \times \mathcal{T} \times \mathsf{intv}(\mathcal{T})$ respectively. The

unfolding of a tuple $\langle n_1, n_2, \tau, d \rangle \in \mathcal{U}^t$ is $\{\langle n_1, n_2, t, d \rangle \mid t \in \tau\}$, and analogously $\{\langle n_1, n_2, t, d \rangle \mid d \in \delta\}$ for a tuple $\langle n_1, n_2, t, \delta \rangle \in \mathcal{U}^d$.

Inductive representation. In order to understand when the answers $[\![q]\!]_G$ to a TRPQ q over a temporal graph G can be finitely represented in $\mathcal{U}^t$ or $\mathcal{U}^d$, and what the size of such representations may be, we define by structural induction on q two (not necessarily compact) representation of $[\![q]\!]_G$ in $\mathcal{U}^t$ and $\mathcal{U}^t$, noted $(\!|q|\!)^t_G$ and $(\!|q|\!)^d_G$ respectively (these representations also pave the way for implementing query evaluation). For instance, in the case where q is of the form *pred*, we define $(\!|pred|\!)^t_G$ as $\{\langle n, n, \mathcal{T}_G, 0 \rangle \mid n \models pred\}$.

Due to space limitations, we only provide the full definition of $(\!|q|\!)^t_G$ and $(\!|q|\!)^d_G$ in the appendix (together with proofs of correctness). We highlight here the least obvious operators. The first one is the temporal join $\mathsf{trpq}_1/\mathsf{trpq}_2$, which intuitively composes temporal relations. If we assume (by induction) that the answers to each operand (trpq_1 and trpq_2) are represented in $\mathcal{U}^t$, then the representation of $\mathsf{trpq}_1/\mathsf{trpq}_2$ in $\mathcal{U}^t$ can be computed as a regular join together with simple arithmetic operations on interval boundaries, as follows:

$$(\!|\mathsf{trpq}_1/\mathsf{trpq}_2|\!)^t_G = \Big\{ \langle n_1, n_3, ((\tau_1 + d_1) \cap \tau_2) - d_1, d_1 + d_2 \rangle \mid \exists n_2 :$$

$$\langle n_1, n_2, \tau_1, d_1 \rangle \in (\!|\mathsf{trpq}_1|\!)^t_G, \langle n_2, n_3, \tau_2, d_2 \rangle \in (\!|\mathsf{trpq}_2|\!)^t_G \text{ and } (\tau_1 + d_1) \cap \tau_2 \neq \emptyset \Big\}.$$

This observation also holds for $\mathcal{U}^d$, but with different operations on interval boundaries:

$$(\!|\mathsf{trpq}_1/\mathsf{trpq}_2|\!)^d_G = \Big\{ \langle n_1, n_3, t_1, t_2 - t_1 + \delta_2 \rangle \mid \exists n_2 : \langle n_1, n_2, t_1, \delta_1 \rangle \in (\!|\mathsf{trpq}_1|\!)^d_G,$$

$$\langle n_2, n_3, t_2, \delta_2 \rangle \in (\!|\mathsf{trpq}_2|\!)^d_G \text{ and } t_2 - t_1 \in \delta_1 \Big\}.$$

The less obvious of these two operations is $(\!|\mathsf{trpq}_1/\mathsf{trpq}_2|\!)^t_G$, which we illustrate with Figure 4).

Another case of interest is the temporal navigation operator $\mathbf{T}_\delta$, for the second representation $\mathcal{U}^d$ (the case of $\mathcal{U}^t$ is trivial). Consider a query of the form $q/\mathbf{T}_\delta$ (or symmetrically $\mathbf{T}_\delta/q$). If the subquery $\mathbf{T}_\delta$ is evaluated independently, then the output of this subquery may be infinite over dense time, which rules out in practice an inductive evaluation:

$$(\!|\mathbf{T}_\delta|\!)^d_G = \{\langle n, n, t, ((\delta + t) \cap \mathcal{T}_G) - t \rangle \mid n \in \mathcal{N}_G, t \in \mathcal{T}_G, (\delta + t) \cap \mathcal{T}_G \neq \emptyset\}.$$

However, if $(\!|q|\!)^d_G$ is finite, then the output of the whole query $q/\mathbf{T}_\delta$ can be represented finitely, and effectively computed as follows:

$$(\!|q/\mathbf{T}_\delta|\!)^d_G = \{\langle n_1, n_2, t, (\delta' + \delta) \cap \mathcal{T}_G \rangle \mid \langle n_1, n_2, t, \delta' \rangle \in (\!|q|\!)^d_G, (t + (\delta' + \delta)) \cap \mathcal{T}_G \neq \emptyset\}.$$

Finiteness over dense time. Over discrete time, trivially, $[\![q]\!]_G$ can be finitely represented in $\mathcal{U}$, [2] therefore as well in any of our four compact representations.

---

[2] Recall that we assume the effective temporal domain $\mathcal{T}_G$ of G to be bounded.
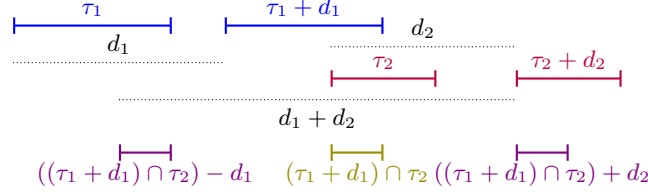
Fig. 4: Join of two tuples $\langle n_1, n_2, \tau_1, d_1 \rangle$ and $\langle n_2, n_3, \tau_2, d_2 \rangle$ in $\mathcal{U}^t$. Each tuple is depicted as two intervals τ_i and $\tau_i + d_i$. The two corresponding intervals τ_o and $\tau_o + d_o$ for the output tuple $\langle n_1, n_3, \tau_o, d_o \rangle$ are in violet.

But over dense time, this is not always possible. For instance, if q_t is the query $(= positive)/\texttt{tests}^-/\mathbf{T}_{[-7,0]}$, and if the graph G of Figure 1 is interpreted over dense time, then $\langle positive, Bob, 112, d \rangle \in [\![q_t]\!]_G$ for every rational number d in $[-7, 0]$. And no tuple in $\mathcal{U}^t$ can represent more than one of these tuples. From the definition of $(\!|q|\!)^t_G$, the only possible source of non-finiteness for $\mathcal{U}^t$ is the temporal navigation operator $\mathbf{T}_\delta$, and only if δ is not a singleton interval (i.e., if it specifies a certain *range* rather than a fixed distance).

Similarly, if q_d is the query $(= Bob)/\texttt{attends}$, then $\langle Bob, ISWC, t, 0 \rangle \in [\![q]\!]_G$, for every rational number t in $[102, 107]$, and no tuple in $\mathcal{U}^d$ can represent more than one of these.

Compactness. If $[\![q]\!]_G$ can be finitely represented in $\mathcal{U}^t$ or $\mathcal{U}^d$, then a natural requirement on these representations is conciseness. It is easy to see that a finite set $U \subseteq \mathcal{U}^t$ is compact iff all time intervals for the same n_1, n_2 and d within U are coalesced. Formally, let $\sim$ denote the binary relation over $\mathcal{U}^t$ defined by $\langle n_1, n_2, \tau_1, d_1 \rangle \sim \langle n_3, n_4, \tau_2, d_2 \rangle$ iff $\langle n_1, n_2, d_1 \rangle = \langle n_3, n_4, d_2 \rangle$ and $\tau_1 \cup \tau_2 \in \mathsf{intv}(\mathcal{T})$. Then U is compact iff $\mathbf{u}_1 \not\sim \mathbf{u}_2$ for all $\mathbf{u}_1, \mathbf{u}_2 \in U$ s.t. $\mathbf{u}_1 \neq \mathbf{u}_2$. More, there is a unique way to coalesce a finite set of intervals. Therefore if $V \subseteq \mathcal{U}$ can be finitely represented in $\mathcal{U}^t$, then V also has a unique compact representation in $\mathcal{U}^t$. A symmetric observation holds for a set $U \subseteq \mathcal{U}^d$, where we coalesce distance intervals rather than time intervals.

Coalescing a set of intervals is known to be in $O(n \log n)$, and efficient implementations have been devised (see Section 5). For this reason, coalescing intermediate results in $(\!|q|\!)^t_G$ or $(\!|q|\!)^d_G$ may be an interesting query evaluation strategy. For instance, this could reduce the size of the operands of a (worst-case quadratic) temporal join.

Size of compact answers. Our two representations $\mathcal{U}^t$ and $\mathcal{U}^d$ exhibit an interesting symmetry when it comes to the size of compact answers to a query, intuitively growing with the size of the intervals present in q in the case of $\mathcal{U}^t$, and the intervals present in G in the case of $\mathcal{U}^d$. This suggests that $\mathcal{U}^t$ is be better suited to large graphs intervals and small temporal navigation intervals (and conversely, that $\mathcal{U}^d$ is better suited to large query intervals).

Unfortunately, this does not hold for arbitrary queries. Indeed, the size of a compact representation of $[\![q]\!]_G$ may be affected by the size of the effective

temporal domain $\mathcal{T}_G$ alone, even if all time intervals in the query and graph are singletons, as soon as q contains an occurrence of the closure operator $[m, \_]$, as illustrated with the following example:

Example 1. Consider a temporal graph G s.t. $\mathsf{val}_G(n_1, e, n_2) = \{[0,0]\}$ and consider the query $q = e/(\mathrm{T}_{[2,2]})[1, \_]$. The compact representation of $[\![q]\!]_G$ is $\{\langle n_1, n_2, [0,0], d\rangle \mid d \in D\}$ in $\mathcal{U}^t$, and $\{\langle n_1, n_2, 0, [d,d]\rangle \mid d \in D\}$ in $\mathcal{U}^d$, where $D = \{d \in \mathcal{T}_G \cap \mathbb{N}^+ \mid d \bmod 2 = 0\}$.   □

However, for queries without closure operator, which we call *star-free*, the symmetry sketched above holds. To formalize this, we introduce a notation that we will reuse for the other two representations. Let $\mathcal{U}^x$ be one of $\mathcal{U}^t$, $\mathcal{U}^d$, $\mathcal{U}^{td}$ or $\mathcal{U}^c$. Consider a temporal graph G and a star-free query q, such that $[\![q]\!]_G$ can be finitely represented in $\mathcal{U}^x$. We fix G and q, with the exception of $\mathcal{T}_G$, and either *(i)* the intervals present in G, or *(ii)* the intervals present in q (with the requirement that $[\![q]\!]_G$ can still be finitely represented in $\mathcal{U}^x$). In case *(i)*, let n be the cumulated length of the intervals in G, and let V be a compact representation of $[\![q]\!]_G$ in $\mathcal{U}^x$. We use $\#\mathsf{answers}^\tau(\mathcal{U}^x)$ for the function that maps n to the the number of tuples in V. In case *(ii)*, we use $\#\mathsf{answers}^\delta(\mathcal{U}^x)$ with an analogous meaning, but where n is the cumulated size of the intervals in q.

The following results says that the size of the compact representation of $[\![q]\!]_G$ in $\mathcal{U}^t$ (when it exists) may be affected by the size of the intervals present in q, but not the ones used to label triples in G, and conversely for $\mathcal{U}^d$:

Proposition 1. *In the worst case,*

$$\#\mathsf{answers}^\tau(\mathcal{U}^t) = O(1) \qquad\qquad \#\mathsf{answers}^\delta(\mathcal{U}^t) = \Omega(n)$$
$$\#\mathsf{answers}^\tau(\mathcal{U}^d) = \Omega(n) \qquad\qquad \#\mathsf{answers}^\delta(\mathcal{U}^d) = O(1)$$

Complexity of query answering. Let $\mathcal{U}^x$ be one of $\mathcal{U}^t$, $\mathcal{U}^d$, $\mathcal{U}^{td}$ or $\mathcal{U}^c$. We formulate a decision problem analogous to the classical boolean query answering problem (for atemporal databases), in such a way that it remains defined even if $[\![q]\!]_G$ does not admit a finite representation in $\mathcal{U}^x$. We say that a tuple $\mathbf{u}$ in $\mathcal{U}^x$ is a *compact answer* to q over G if its unfolding is a subset of $[\![q]\!]_G$ and is maximal among the tuples in $\mathcal{U}^x$ that satisfy this condition. We can now define our (four) problems (where x is either t, d, td or c):

COMPACTANSWER^x
Input: temporal graph G, TRPQ q, tuple $\mathbf{u} \in \mathcal{U}^x$
Decide: $\mathbf{u}$ is a compact answer to q over G

Complexity for these problems is (partly) driven by the size of the input time intervals, and there is no reason a priori to assume that intervals in the graphs are larger than the ones in the query. This is why we do not focus on data complexity (where the query is fixed), but instead on combined complexity, where the size of the query and data may vary (we leave for future work a finer-grained analysis). We show in the appendix that the results proven in [3] for

answering TRPQs in $\mathcal{U}$ immediately transfer to $\mathcal{U}^t$ (resp. $\mathcal{U}^d$), even in the case where $[\![q]\!]_G$ cannot be finitely represented in $\mathcal{U}^t$ (resp. $\mathcal{U}^d$):

Proposition 2.
COMPACTANSWER^t and COMPACTANSWER^d are PSPACE-complete.

We also observe that hardness can be proven with a graph of fixed size, with the exception of the effective temporal domain $\mathcal{T}_G$. However, the number of operators of the query used in this reduction is not fixed.

4.2 Folding time points and distances ($\mathcal{U}^{td}$ and $\mathcal{U}^c$)

Each of the two compact representations $\mathcal{U}^t$ and $\mathcal{U}^d$ defined above aggregates tuples in $\mathcal{U}$ along one dimension (either time points or distances), and both may fail to represent the answers to a query compactly (or even finitely over dense time). So a first natural attempt to address this limitation consists in aggregating tuples along both dimensions. Accordingly, we define the universe $\mathcal{U}^{td}$ as $\mathcal{N} \times \mathcal{N} \times \mathsf{intv}(\mathcal{T}) \times \mathsf{intv}(\mathcal{T})$, and the unfolding of $\langle n_1, n_2, \tau, \delta \rangle \in \mathcal{U}^{td}$ as $\{\langle n_1, n_2, t, d \rangle \mid t \in \tau, d \in \delta\}$.

This representation is more compact than the two previous ones, since unfolding a tuple in $\mathcal{U}^t$ or $\mathcal{U}^d$ yields a subset of some unfolded tuple of $\mathcal{U}^{td}$. However, as we will see below, minimizing the cardinality of a set of query answers becomes intractable. Besides, maybe surprisingly, this new format may also fails to represent query answers in a compact fashion (or finitely over dense time). To see this, observe that for two fixed nodes, a tuple in $\mathcal{U}^{td}$ has a natural representation as a rectangle in the Euclidean plane P of times per distances. The following example shows a query whose output, depicted in Figure 5b, cannot be represented (over dense time) as a finite set of rectangles in P.

Example 2. Consider a temporal graph G with two edges such that $\mathsf{val}_G(n_1, e_1, n_2) = \{[0,2]\}$ and $\mathsf{val}_G(n_2, e_2, n_3) = \{[1,3]\}$ Let q be the query $e_1/\mathbf{T}_{[0,2]}/e_2$. Then $[\![q]\!]_G = \{\langle n_1, n_2, t, d \rangle \mid t \in [0,2], d \in [0,2] \text{ and } t + d \in [1,3]\}$. So for an answer $\langle n_1, n_2, t, d \rangle \in [\![q]\!]_G$, we have

$$1 \leq t + d \leq 3, \quad \text{that is,} \quad 1 - t \leq d \leq 3 - t.$$

Besides, from the query q, we have that $0 \leq d \leq 2$. Therefore

$$\max(1 - t, 0) \leq d \leq \min(3 - t, 2).$$

This observation gives us an interval

$$\delta_t = [\max(1 - t, 0), \min(3 - t, 2)]$$

of admissible distances for each $t \in [0,2]$, so that $[\![q]\!]_G = \{\langle n_1, n_2, t, d \rangle \mid t \in [0,2] \text{ and } d \in \delta_t\}$. Extending this kind of reasoning to the general case, we derive the formula (1) presented below for δ_t. □

As shown in Figure 5b, the area covered by the output of this query can be viewed as a rectangle cropped by two parallel lines, each with slope -1. These cropped rectangles turn out to have an interesting property: if two temporal relations have such a shape, then their composition also does.

Associating each tuple with such a cropped rectangle (as opposed to a regular rectangle in $\mathcal{U}^{td}$) allows us to solve our initial problem, i.e., to compute inductively the output of a TRPQ q over a graph G in such a way that the number of output tuples remains independent of the size of the intervals present in either q or G. To represent these cropped rectangles, we introduce a fourth format, which we call $\mathcal{U}^c$ (where "c" stands for "cropped"), where each tuple carries, in addition to a time interval τ and a distance interval δ (a.k.a. a rectangle), the two values $b, e \in \mathcal{T}$ depicted in Figure 5b, which intuitively indicate where the cropping lines intersect the rectangle induced by τ and δ. For each time point $t \in \mathcal{T}$, these define a specific range of distances $\delta_t \subseteq \delta$, as

$$\delta_t = {}_\delta\lfloor\ b_\delta + \max(0, b - t)\ ,\ e_\delta - \max(0, t - e)\ \rfloor_\delta \tag{1}$$

where ${}_\delta\lfloor$ and $\rfloor_\delta$ stand for the left and right delimiters of δ (for instance, if δ is closed-open, then ${}_\delta\lfloor$ is "[" and then $\rfloor_\delta$ is ")").

Accordingly, a tuple $\langle n_1, n_2, \tau, \delta, b, e \rangle \in \mathcal{N} \times \mathcal{N} \times \mathsf{intv}(\mathcal{T}) \times \mathsf{intv}(\mathcal{T}) \times \mathcal{T} \times \mathcal{T}$ is in $\mathcal{U}^c$ iff δ_t is nonempty for every $t \in \tau$. And the unfolding of this tuple is $\{\langle n_1, n_2, t, d \rangle \mid t \in \tau, d \in \delta_t\}$.

Inductive representation. As we did for our first two representations, we define (in appendix) by structural induction on q two representations $(\!|q|\!)_G^{td}$ and $(\!|q|\!)_G^c$ of $[\![q]\!]_G$ in $\mathcal{U}^{td}$ and $\mathcal{U}^c$ respectively. Our most technical result is correctness of this definition for the join operator in $\mathcal{U}^c$ (and to a lesser extent the temporal navigation operator).

We reproduce these definitions here to make apparent the fact these can be computed by means of simple arithmetic operations on time points and interval boundaries (together with a regular join on nodes for the join operator). The inductive evaluation $(\!|\mathsf{trpq}_1/\mathsf{trpq}_2|\!)_G^c$ in $\mathcal{U}^c$ of the query $\mathsf{trpq}_1/\mathsf{trpq}_2$ is defined as:

$$\{\mathbf{u}_1 \bowtie \mathbf{u}_2 \mid \mathbf{u}_1 \in (\!|\mathsf{trpq}_1|\!)_G^c, \mathbf{u}_2 \in (\!|\mathsf{trpq}_2|\!)_G^c, \mathbf{u}_1 \sim \mathbf{u}_2\}$$

where $\mathbf{u}_1 \sim \mathbf{u}_2$ and $\mathbf{u}_1 \bowtie \mathbf{u}_2$ are defined as follows.
For $\mathbf{u}_1 = \langle n_1, n_2, \tau_1, \delta_1, b_1, e_1 \rangle$ and $\mathbf{u}_2 = \langle n_3, n_4, \tau_2, \delta_2, b_2, e_2 \rangle$, let

$$\begin{aligned}\delta_1' &= {}_{\delta_1}\lfloor\ b_{\delta_1} + \max(0, b_1 - b_{\tau_1}), e_{\delta_1} - \max(0, e_{\tau_1} - e_1)\ \rfloor_{\delta_1}, \text{ and} \\ \tau &= (((\tau_1 + \delta_1') \cap \tau_2) \ominus \delta_1') \cap \tau_1.\end{aligned}$$

Then the relation $\sim\ \subseteq \mathcal{U}^c \times \mathcal{U}^c$ is defined as $\mathbf{u}_1 \sim \mathbf{u}_2$ iff $n_2 = n_3$ and $\tau \neq \emptyset$. And if $\mathbf{u}_1 \sim \mathbf{u}_2$, then $\mathbf{u}_1 \bowtie \mathbf{u}_2$ is defined as $\langle n_1, n_4, \tau, \delta_1 + \delta_2, b, e \rangle$, with $b = \max(b_1, b_2 - b_{\delta_1})$ and $e = \min(e_1, e_2 - e_{\delta_1})$.

Similarly, if $b_{\mathcal{T}_G}$ and $e_{\mathcal{T}_G}$ are the boundaries of the interval $\mathcal{T}_G$, then $(\!|\mathbf{T}_\delta|\!)_G^c$ is defined as

$$\{\langle n, n, \mathcal{T}_G, \delta, b_{\mathcal{T}_G}, e_{\mathcal{T}_G} \rangle \bowtie \langle n, n, \mathcal{T}_G, [0,0], b_{\mathcal{T}_G}, e_{\mathcal{T}_G} \rangle \mid n \in \mathcal{N}_G\}.$$
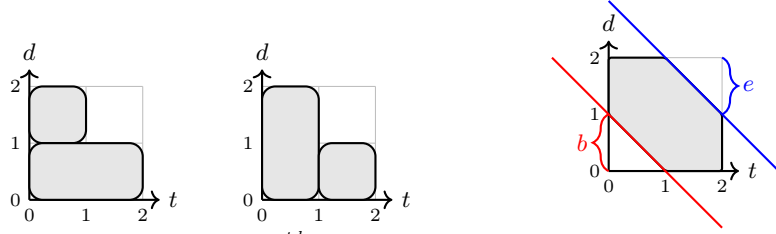
Fig. 5: (a) Two minimal covers in $\mathcal{U}^{td}$ **(b)** Answers to the query of Example 2

Finiteness over dense time. We already illustrated in Figure 5b why $\llbracket q \rrbracket_G$ may not be finitely representable in $\mathcal{U}^{td}$. In contrast, $\llbracket q \rrbracket_G$ can be finitely represented in $\mathcal{U}^c$, as a direct consequence of the (correctness of the) inductive definition of $(\!|q|\!)_G^c$, which produces finitely many tuples (notably, the operator $\mathsf{trpq}_1/\mathsf{trpq}_2$ produces at most one tuple per pair $(\mathbf{u}_1, \mathbf{u}_2) \in (\!|\mathsf{trpq}_1|\!)_G^c \times (\!|\mathsf{trpq}_2|\!)_G^c)$.

Compactness. Let $U \subseteq \mathcal{U}^{td}$ and $V \subseteq \mathcal{U}$ be two sets of tuples that share the same nodes n_1 and n_2. Then U unfolds as V iff they intuitively cover the same area in the Euclidean plane of times per distances, i.e. if

$$\bigcup \{\tau \times \delta \mid \langle n_1, n_2, \tau, \delta \rangle \in U\} = \{(t, d) \mid \langle n_1, n_2, t, d \rangle \in V\}.$$

There may be several compact representations in $\mathcal{U}^{td}$ for the same $V \subseteq \mathcal{U}$. For instance, the minimal number of rectangle needed to cover an "L"-shaped polygon is two, and there are several such covers, as illustrated with Figure 5a. This argument easily generalizes to discrete time.

Besides, minimizing the representation of $\llbracket q \rrbracket_G$ in $\mathcal{U}^{td}$ (i.e. computing a minimal set of tuples in $\mathcal{U}^{td}$ with unfolding $\llbracket q \rrbracket_G$) is intractable. To see this, we first observe that computing such a set is harder than deciding whether there exists one with size k (for a given k). Next, the following problem is known to be NP-complete: given a rectilinear polygon r and a number k, decide whether there is a set of at most k (possibly overlapping) rectangles that exactly cover r [9,4]. This problem reduces to ours, observing that for any rectilinear polygon r, a query q (with only unions) and graph G can be constructed in polynomial time so that $\llbracket q \rrbracket_G$ covers exactly r. This hardness result also immediately translates to $\mathcal{U}^c$: a rectangle is a specific case of a cropped rectangle, therefore the same reduction can be used, and if there is a minimization with at most k tuples in $\mathcal{U}^{td}$, then there is also one in $\mathcal{U}^c$.

An important difference though between these two representations and the two previous ones ($\mathcal{U}^t$ and $\mathcal{U}^d$) is that a compact set U of answers in $\mathcal{U}^{td}$ or $\mathcal{U}^c$ may be *redundant*, meaning that two tuples $\mathbf{u}_1$ and $\mathbf{u}_2$ in U may have overlapping unfoldings. This can for instance be seen in Figure 2d, for $\mathcal{U}^{td}$. If we require tuples to be non-redundant, then the representations of answers to a query is in general less concise. But it may be better suited for downstream tasks, such as aggregation. Besides, tractability of minimization in $\mathcal{U}^{td}$ is regained, because finding a cover with a minimal number of non-overlapping rectangles is tractable [20,14]. However, uniqueness is not regained, as shown in Fig. 5a.

	Finite (dense time)	Unique	Size (star-free q)		Minimization	Query answering
			graph intervals	query intervals		
$\mathcal{U}^t$	no	yes	$O(1)$	$\Omega(n)$	$O(n \log n)$	PSPACE-c
$\mathcal{U}^d$	no	yes	$\Omega(n)$	$O(1)$	$O(n \log n)$	PSPACE-c
$\mathcal{U}^{td}$	no	no	$O(1)$	$\Omega(n)$	NP-h / $O(n^{2.5})$	PSPACE-c
$\mathcal{U}^c$	yes	no	$O(1)$	$O(1)$	NP-h	PSPACE-h

Fig. 6: Summary of results.

Size of compact answers. If $V \subseteq \mathcal{U}$ can be finitely represented in $\mathcal{U}^{td}$, then trivially, a compact representation of V in $\mathcal{U}^{td}$ must be smaller than the compact representation of V in $\mathcal{U}^t$ (resp. $\mathcal{U}^d$), if the latter exists. So compact answers under this representation must be smaller than under the two previous ones. However, the size of a compact representation may still be affected by the size of time intervals in q (for a star-free q already). In contrast, for $\mathcal{U}^c$, immediately from the definition of $(\!|q|\!)_G^c$, the number of tuples in $(\!|q|\!)_G^c$ (for a star-free q) is independent of the size of the intervals in G or q (even though $(\!|q|\!)_G^c$ is not necessarily compact).

Proposition 3. *In the worst case,*

$$\#\mathsf{answers}^\tau(\mathcal{U}^{td}) = O(1) \qquad \#\mathsf{answers}^\delta(\mathcal{U}^{td}) = \Omega(n)$$
$$\#\mathsf{answers}^\tau(\mathcal{U}^c) = O(1) \qquad \#\mathsf{answers}^\delta(\mathcal{U}^c) = O(1)$$

Complexity of query answering. The hardness results of [3] over $\mathcal{U}$ can also be lifted to $\mathcal{U}^{td}$ and $\mathcal{U}^c$, and this bound is tight for $\mathcal{U}^{td}$:

Proposition 4. COMPACTANSWER^{td} is PSPACE-complete, COMPACTANSWER^c is PSPACE-hard.

5 Related work

In temporal relational databases, tuples (or attributes) are most commonly associated with a *single* time interval that represents either validity or transaction time [5]. Intervals are commonly used instead of time points as a compact representation. To maintain the compact and unique representation through operations, the *coalescing operator*, which merges value-equivalent tuples over consecutive or overlapping time intervals, has received a lot of attention. Böhlen et al. [6] showed that coalescing can be implemented in SQL, and provided a comprehensive analysis of various coalescing algorithms. Al-Kateb et al. [1] investigated coalescing in the attribute timestamped CME temporal relational model and the work in [12] defines coalescing for temporal multi-set relations. Zhou et al. [23] exploited SQL:2003's analytical functions for the computation of coalescing, which to date, is the state-of-the-art technique and can be computed efficiently in $\mathcal{O}(n \log n)$. In two of our four models that use a single interval, i.e.,

$\mathcal{U}^t$ and $\mathcal{U}^d$, coalescing can be used to achieve a compact representation, but it is not applicable for $\mathcal{U}^{td}$ and $\mathcal{U}^c$. Bitemporal databases [18] use two intervals (or rectangles) as a compact representation for two time points, typically one for validity and one for transaction times. Even though we are not modeling transaction times, our representation $\mathcal{U}^{td}$ also uses two intervals, In bitemporal databases, coalescing does not provide a unique representation [22] and, since it is designed as a non-redundant operation, may not provide the most compact representation.

Also relevant to our work is the efficient computation of temporal joins over intervals. There has been a long line of research on temporal joins [16], ranging from partition-based [10,8], index-based [13,19], and sorting based [21,7] techniques. Recently, in [11] it has been shown that a temporal join with the overlap predicate can be transformed into a sequence of two range joins. Our inductive representations of answers require temporal joins and range joins (e.g. for $\mathcal{U}^t$ and $\mathcal{U}^d$ respectively).

Finally, our work builds upon the original proposal made in [2] to extend regular path queries over property graphs with a temporal navigation operator—effectively allowing selection, join, and union of binary temporal relations. Our work focuses on producing compact answers that are finite even in the case of continuous time, a problem that was left open in [2].

6   Conclusions

We investigated how to compactly represent answers to queries over binary temporal relations in the TRPQ setting, where both data and queries may include time intervals for validity and temporal navigation, respectively. To our knowledge, this question was previously open. We defined and analyzed four alternative representations of compact answers to TRPQs, varying in conciseness and potential use. Notably, the fourth ensures that query answers are always finitely representable and independent of input time interval lengths. We see this as a step toward integrating temporal navigation into database systems.

An open question is whether non-redundancy is tractable under this fourth representation, in particular whether tractability for minimizing compact answers is regained.

References

1. Al-Kateb, M., Mansour, E., El-Sharkawi, M.E.: CME: A temporal relational model for efficient coalescing. In: Proc. of the 12th Int. Symp. on Temporal Representation and Reasoning (TIME). pp. 83–90. IEEE Computer Society (2005)
2. Arenas, M., Bahamondes, P., Aghasadeghi, A., Stoyanovich, J.: Temporal regular path queries. In: Proc. of the 38th IEEE Int. Conf. on Data Engineering (ICDE). pp. 2412–2425. IEEE Computer Society (2022)
3. Arenas, M., Bahamondes, P., Stoyanovich, J.: Temporal regular path queries: Syntax, semantics, and complexity. CoRR Technical Report arXiv:2107.01241, arXiv.org e-Print archive (2021), https://arxiv.org/abs/2107.01241, available at https://arxiv.org/abs/2107.01241
4. Aupperle, L., Conn, H.E., Keil, J.M., O'Rourke, J.: Covering Orthogonal Polygons with Squares. Johns Hopkins University, Department of Computer Science (1988)
5. Böhlen, M.H., Dignös, A., Gamper, J., Jensen, C.S.: Temporal data management – An overview. In: Tutorial Lectures of the 7th European Summer School on Business Intelligence and Big Data (eBISS). Lecture Notes in Business Information Processing, vol. 324, pp. 51–83. Springer (2017)
6. Böhlen, M.H., Snodgrass, R.T., Soo, M.D.: Coalescing in temporal databases. In: Proc. of the 22nd Int. Conf. on Very Large Data Bases (VLDB). pp. 180–191 (1996)
7. Bouros, P., Mamoulis, N., Tsitsigkos, D., Terrovitis, M.: In-memory interval joins. Very Large Database J. **30**(4), 667–691 (2021)
8. Cafagna, F., Böhlen, M.H.: Disjoint interval partitioning. Very Large Database J. **26**(3), 447–466 (2017)
9. Culberson, J.C., Reckhow, R.A.: Covering polygons is hard. J. of Algorithms **17**(1), 2–44 (1994)
10. Dignös, A., Böhlen, M.H., Gamper, J.: Overlap interval partition join. In: Proc. of the 35th ACM Int. Conf. on Management of Data (SIGMOD). pp. 1459–1470 (2014)
11. Dignös, A., Böhlen, M.H., Gamper, J., Jensen, C.S., Moser, P.: Leveraging range joins for the computation of overlap joins. Very Large Database J. **31**(1), 75–99 (2022)
12. Dignös, A., Glavic, B., Niu, X., Gamper, J., Böhlen, M.H.: Snapshot semantics for temporal multiset relations. Proc. of the VLDB Endowment **12**(6), 639–652 (2019)
13. Enderle, J., Hampel, M., Seidl, T.: Joining interval data in relational databases. In: Proc. of the 25th ACM Int. Conf. on Management of Data (SIGMOD). pp. 683–694 (2004)
14. Eppstein, D.: Graph-theoretic solutions to computational geometry problems. In: Revised Papers the 35th Int. Workshop on Graph-Theoretic Concepts in Computer Science (WG). Lecture Notes in Computer Science, vol. 5911, pp. 1–16 (2009)
15. Francis, N., Green, A., Guagliardo, P., Libkin, L., Lindaaker, T., Marsault, V., Plantikow, S., Rydberg, M., Selmer, P., Taylor, A.: Cypher: An evolving query language for property graphs. In: Proc. of the 39th ACM Int. Conf. on Management of Data (SIGMOD). pp. 1433–1445 (2018)
16. Gao, D., Jensen, C.S., Snodgrass, R.T., Soo, M.D.: Join operations in temporal databases. Very Large Database J. **14**(1), 2–29 (2005)
17. Harris, S., Seaborne, A.: SPARQL 1.1 query language. W3C Recommendation, World Wide Web Consortium (Mar 2013), available at http://www.w3.org/TR/sparql11-query

18

18. Jensen, C.S., Snodgrass, R.T.: Bitemporal relation. In: Encyclopedia of Database Systems. Springer, 2nd edn. (2018)

19. Kaufmann, M., Manjili, A.A., Vagenas, P., Fischer, P.M., Kossmann, D., Färber, F., May, N.: Timeline index: a unified data structure for processing queries on temporal data in SAP HANA. In: Proc. of the 34th ACM Int. Conf. on Management of Data (SIGMOD). pp. 1173–1184 (2013)

20. Keil, J.M.: Minimally covering a horizontally convex orthogonal polygon. In: Proc. of the 2nd Annual ACM SIGACT/SIGGRAPH Symposium on Computational Geometry (SCG). pp. 43–51 (1986)

21. Piatov, D., Helmer, S., Dignös, A.: An interval join optimized for modern hardware. In: Proc. of the 32th IEEE Int. Conf. on Data Engineering (ICDE). pp. 1098–1109. IEEE Computer Society (2016)

22. Toman, D.: Point-based temporal extensions of SQL and their efficient implementation. In: Temporal Databases, Dagstuhl. Lecture Notes in Computer Science, vol. 1399, pp. 211–237. Springer (1997)

23. Zhou, X., Wang, F., Zaniolo, C.: Efficient temporal coalescing query support in relational database systems. In: Proc. of the 17th Int. Conf. on Database and Expert Systems Applications (DEXA). Lecture Notes in Computer Science, vol. 4080, pp. 676–686. Springer (2006)

A Appendix

The structure adopted in this appendix differs from one followed in the article. Results here are grouped by topic (inductive representation, complexity, etc.), rather than by format ($\mathcal{U}^t$, $\mathcal{U}^d$, etc.). This allows us to factorize some proofs, and emphasize what differs from one case to the other.

The most technical results are the correctness of the inductive representation of compact answers in $\mathcal{U}^c$ (in particular for the join operator), proven in Section A.2, and to a lesser extent the analogous result for $\mathcal{U}^{td}$, proven in Section A.2.

On the other hand, complexity proofs (in Section A.3) leverage results already proven in [3].

Results pertaining to the size of compact answers (in Section A.4), follow either from the corresponding inductive representations (for the upper bounds), or from simple examples (for the lower bounds), similar to the ones already provided in the body of the article.

Almost all arguments that pertain to compactness and cost of coalescing answers are already provided in Section 4, so we complete these for a single case, in Section A.5.

For finiteness over dense time, all negative results (i.e. non-finiteness) are illustrated in the article, and all positive results (i.e. finiteness) follow from the definitions of the inductive representations (and the fact that these are correct).

A.1 Notation

Sets, relations, order. We use $\mathsf{dom}(R)$ and $\mathsf{range}(R)$ for the domain and range of a binary relation, respectively. For a set S and a (possibly partial) order $\preceq$ over S, we denote with $\max_{\preceq} S$ the set of maximal elements in S w.r.t. $\preceq$, i.e., $\{s \in S \mid s \preceq s' \text{ implies } s = s' \text{ for all } s' \in S\}$.

Complement of a set of intervals. Let α be a bounded interval in $\mathsf{intv}(\mathcal{T}_G)$, and let S be a finite set of intervals s.t. $\bigcup S \subseteq \alpha$. We us $\mathsf{compl}\,(S, \alpha)$ to denote the complement of $\bigcup S$ in α represented as maximal intervals, i.e. :

$$\mathsf{compl}\,(S, \alpha) = \max_{\subseteq}\{I \subseteq \bigcup S \setminus \alpha \mid I \in \mathsf{intv}(\mathcal{T})\}$$

Interval of distances for a given time point t. Let $\langle n_1, n_2, \tau, \delta, b, e \rangle \in \mathcal{U}^c$, and let $t \in \tau$.
In the article, we defined the interval δ_t for each t as

$$_{\delta}\lfloor\ b_\delta + \max(0, b - t)\ ,\ e_{\delta_i} - \max(0, t - e)\ \rfloor_\delta$$

In this appendix, we will use $\delta(t)$ instead of δ_t.
This notation will allow us to write $\delta_1(t)$ when several tuples are involved.
Note that the time points b and e in this notaton are still omitted, for conciseness, because they are clear from the context.

A.2 Inductive characterizations

Let q be a TRPQ and G a temporal graph.

Then $[\![q]\!]_G$ is the set of anwers to q over G (represented as tuples in $\mathcal{U}$).

In this section, we provide the full definition of the four inductive representations of $[\![q]\!]_G$ discussed in the article, in $\mathcal{U}^t$, $\mathcal{U}^d$, $\mathcal{U}^{td}$ and $\mathcal{U}^c$ respectively, and prove that they are correct.

These representations are denoted as $(\!|q|\!)^t_G$, $(\!|q|\!)^d_G$, $(\!|q|\!)^{td}_G$ and $(\!|q|\!)^c_G$ respectively.

In $\mathcal{U}^t$

Definition. If q is a TRPQ and $G = \langle \mathcal{T}_G, \mathcal{F}_G, \mathsf{val}_G \rangle$ a temporal graph, then the representation $(\!|q|\!)^t_G$ of $[\![q]\!]_G$ in $\mathcal{U}^t$ is defined inductively as follows:

$$(\!|label|\!)^t_G = \{\langle n_1, n_2, \tau, 0\rangle \mid \tau \in \mathsf{val}_G(n_1, label, n_2)\}$$

$$[\![\mathsf{edge}^-]\!]_G = \{\langle n_2, n_1, \tau, 0\rangle \mid \langle n_1, n_2, \tau, 0\rangle \in [\![\mathsf{edge}]\!]_G\}$$

$$(\!|pred|\!)^t_G = \{\langle n, n, \mathcal{T}_G, 0\rangle \mid n \models pred\}$$

$$(\!|\leq k|\!)^t_G = \begin{cases} \{\langle n, n, _{\mathcal{T}_G}\lfloor b_{\mathcal{T}_G}, k\rfloor, 0\rangle \mid n \in \mathcal{N}_G\} & \text{if } k \in \mathcal{T}_G, \text{ and} \\ \emptyset & \text{otherwise} \end{cases}$$

$$(\!|\mathbf{T}_\delta|\!)^t_G = \{\langle n, n, [t_1, t_1], t_2 - t_1\rangle \mid n \in \mathcal{N}_G, t_1 \in \mathcal{T}_G \text{ and } t_2 \in (\delta + t_1) \cap \mathcal{T}_G\}$$

$$(\!|(?\mathsf{trpq})|\!)^t_G = \{\langle n, n, \tau, 0\rangle \mid \langle n, n', \tau, d\rangle \in (\!|\mathsf{trpq}|\!)^t_G \text{ for some } n' \in \mathcal{N}_G, d \in \mathcal{T}\}$$

$$(\!|\neg\mathsf{node}|\!)^t_G = \bigcup_{n \in \mathcal{N}_G} \left\{\langle n, n, \tau, 0\rangle \mid \tau \in \mathsf{compl}\left(\{\tau' \mid \langle n, n, \tau', 0\rangle \in (\!|\mathsf{node}|\!)^t_G\}, \mathcal{T}_G\right)\right\}$$

$$(\!|\mathsf{trpq}_1/\mathsf{trpq}_2|\!)^t_G = \Big\{\langle n_1, n_3, ((\tau_1 + d_1) \cap \tau_2) - d_1, d_1 + d_2\rangle \mid \exists n_2 :$$
$$\langle n_1, n_2, \tau_1, d_1\rangle \in (\!|\mathsf{trpq}_1|\!)^t_G, \langle n_2, n_3, \tau_2, d_2\rangle \in (\!|\mathsf{trpq}_2|\!)^t_G \text{ and } (\tau_1 + d_1) \cap \tau_2 \neq \emptyset \Big\}.$$

$$(\!|\mathsf{trpq}_1 + \mathsf{trpq}_2|\!)^t_G = (\!|\mathsf{trpq}_1|\!)^t_G \cup (\!|\mathsf{trpq}_2|\!)^t_G$$

$$(\!|\mathsf{trpq}[m, n]|\!)^t_G = \bigcup_{k=m}^{n} (\!|\mathsf{trpq}^k|\!)^t_G$$

$$(\!|\mathsf{trpq}[m, \_]|\!)^t_G = \bigcup_{k \geq m} (\!|\mathsf{trpq}^k|\!)^t_G$$

We observe that when q is of the form $(\mathsf{trpq}_1 + \mathsf{trpq}_2)$, $(\mathsf{trpq}[m, \_])$ and $(\mathsf{trpq}[m, n])$, the defnition of $(\!|q|\!)^t_G$ is nearly identical to the one of $[\![q]\!]_G$. This also holds for the three representations below.

Correctness. We start with a (trivial) lemma:

Lemma 1. *Let $G = \langle \mathcal{T}_G, \mathcal{F}_G, \mathsf{val}_G \rangle$ be a temporal graph and let q be an expression for the symbol* node *or label in the grammar of Section 3.*
Then:

- *each tuples in $[\![q]\!]_G$ is of the form $\langle n_1, n_2, t, 0\rangle$ for some n_1, n_2 and t,*
- *each tuples in $(\!|q|\!)^t_G$ is of the form $\langle n_1, n_2, \tau, 0\rangle$ for some n_1, n_2 and τ.*

Proof. Immediate from the definitions of $[\![q]\!]_G$ and $(\![q]\!)_G^t$.

The following result states that the representation $(\![q]\!)_G^t$ is correct:

Proposition 5. *For a temporal graph* $G = \langle \mathcal{T}_G, \mathcal{F}_G, \mathsf{val}_G \rangle$ *and a TRPQ* q, *the unfolding of* $(\![q]\!)_G^t$ *is* $[\![q]\!]_G$.

Proof.
Let $G = \langle \mathcal{T}_G \mathcal{F}_G, \mathsf{val}_G \rangle$ be a temporal graph, and let q be a TRPQ.
We show below that:

(I) for any $\langle n_1, n_2, t, d \rangle \in [\![q]\!]_G$, there is a $\tau \in \mathsf{intv}(\mathcal{T})$ such that
  (a) $\langle n_1, n_2, \tau, d \rangle \in (\![q]\!)_G^t$, and
  (b) $t \in \tau$,
(II) for any $\langle n_1, n_2, \tau, d \rangle \in (\![q]\!)_G^t$ for any $t \in \tau$,
  $\langle n_1, n_2, t, d \rangle$ is in $[\![q]\!]_G$.

We proceed by induction on the structure of q.
If q is of the form *label*, edge^-, *pred*, $\leq k$, $(\mathsf{trpq} + \mathsf{trpq})$, $\mathsf{trpq}[m, n]$ or $\mathsf{trpq}[m, \_]$,
then I and II immediately follow from the definitions of $[\![q]\!]_G$ and $(\![q]\!)_G^t$.
So we focus below on the four remaining cases:

- $q = \mathbf{T}_\delta$.
  From the above definitions, we have:

  $$[\![q]\!]_G = \{\langle n, n, t, d \rangle \mid n \in \mathcal{N}_G, t \in \mathcal{T}_G, d \in \delta \text{ and } t + d \in \mathcal{T}_G\}$$
  $$(\![q]\!)_G^t = \{\langle n, n, [t_1, t_1], t_2 - t_1 \rangle \mid n \in \mathcal{N}_G, t_1 \in \mathcal{T}_G \text{ and } t_2 \in (\delta + t_1) \cap \mathcal{T}_G\}$$

  - For I, let $\mathbf{v} = \langle n, n, t, t + d \rangle \in [\![q]\!]_G$.
    And let $\mathbf{u} = \langle n, n, [t, t], d \rangle$ in $\mathcal{U}^t$.
    For Ia we show that $\mathbf{u} \in (\![q]\!)_G^t$.
    From $\mathbf{v} \in [\![q]\!]_G$, we get $n \in \mathcal{N}_G$ and $t \in \mathcal{T}_G$.
    Besides, because $\mathbf{v} \in [\![q]\!]_G$ still,

    $$t + d \in \mathcal{T}_G \tag{2}$$

    and

    $$d \in \delta \tag{3}$$
    $$t + d \in t + \delta \tag{4}$$

    So from (2) and (4)
    $$t + d \in (\delta + t) \cap \mathcal{T}_G \tag{5}$$

    So there is a t_2 (namely $t + d$) such that $d = t_2 - t$ and $t_2 \in t + \delta \cap \mathcal{T}_G$.
    Together with the definition of $(\![q]\!)_G^t$, this implies $\mathbf{u} \in (\![q]\!)_G^t$, which concludes the proof for Ia.
    And trivially, $t \in [t, t]$, so Ib is verified as well.

- For II, let $\mathbf{u} = \langle n, n, [t, t], d \rangle \in (\!|q|\!)_G^t$.
  From $\mathbf{u} \in (\!|q|\!)_G^t G$, we get $n \in \mathcal{N}_G$ and $t \in \mathcal{T}_G$.
  So to conclude the proof, it is sufficient to show that *(i)* $d \in \delta$ and *(ii)* $t + d \in \mathcal{T}_G$.
  Because $\mathbf{u} \in (\!|q|\!)_G^t G$ still, we have

$$d = t_2 - t \text{ for some } t_2 \in (\delta + t) \cap \mathcal{T}_G \tag{6}$$

  From (6), we get $t_2 = t + d$.
  Therefore from (6) still,

$$t + d \in (\delta + t) \cap \mathcal{T}_G \tag{7}$$

  which proves *(ii)* .
  And from (7), we also get

$$t + d \in \delta + t$$
$$t + d - t \in (\delta + t) - t$$
$$d \in \delta$$

  which proves *(i)* .

- $q = (?\mathsf{trpq})$.
  From the above definitions, we have:

$$[\![q]\!]_G = \{\langle n, n, t, 0 \rangle \mid \langle n, n', t, t + d \rangle \in [\![\mathsf{trpq}]\!]_G \text{ for some } n' \in \mathcal{N}_G, d \in \mathcal{T}\}$$
$$(\!|q|\!)_G^t = \{\langle n, n, \tau, 0 \rangle \mid \langle n, n', \tau, d \rangle \in (\!|\mathsf{trpq}|\!)_G^t \text{ for some } n' \in \mathcal{N}_G, d \in \mathcal{T}\}$$

  - For I, let $\langle n, n, t, 0 \rangle \in [\![q]\!]_G$.
    From the definition of $[\![q]\!]_G$, there are n' and d such that $\langle n, n', t, t+d \rangle \in [\![\mathsf{trpq}]\!]_G$.
    So by IH, there is a τ s.t. $t \in \tau$ and $\langle n, n', \tau, d \rangle \in (\!|\mathsf{trpq}|\!)_G^t$.
    Therefore $\langle n, n, \tau, 0 \rangle \in (\!|q|\!)_G^t$, from the definition of $(\!|q|\!)_G^t$.
  - For II, let $\langle n, n, \tau, 0 \rangle \in (\!|q|\!)_G^t$.
    From the definition of $(\!|q|\!)_G^t$, there are n' and d s.t. $\langle n, n', \tau, d \rangle \in (\!|\mathsf{trpq}|\!)_G^t$.
    Now take any $t \in \tau$.
    By IH, $\langle n, n', t, t + d \rangle \in [\![\mathsf{trpq}]\!]_G$.
    Therefore $\langle n, n, t, 0 \rangle \in [\![q]\!]_G$, from the definition of $[\![q]\!]_G$.

- $q = \neg\mathsf{node}$.
  From the above definitions, we have:

$$[\![q]\!]_G = (\{\langle n, n \rangle \mid n \in \mathcal{N}_G\} \times \mathcal{T}_G \times \{0\}) \setminus [\![\mathsf{node}]\!]_G$$
$$(\!|q|\!)_G^t = \bigcup_{n \in \mathcal{N}_G} \left\{ \langle n, n, \tau, 0 \rangle \mid \tau \in \mathsf{compl}\left(\{\tau' \mid \langle n, n, \tau', 0 \rangle \in (\!|\mathsf{node}|\!)_G^t\}, \mathcal{T}_G\right) \right\}$$

- For I, let $\mathbf{v} = \langle n, n, t, 0 \rangle \in [\![q]\!]_G$.
  From the definition of $[\![q]\!]_G$, $\mathbf{v} \notin [\![\mathsf{node}]\!]_G$.
  So
  $$t \notin \{t' \mid \langle n, n, t', 0 \rangle \in [\![\mathsf{node}]\!]_G\} \tag{8}$$

  Now by IH, together with Lemma 1, we get:

  $$\langle n, n, t', 0 \rangle \in [\![\mathsf{node}]\!]_G \text{ iff } t' \in \tau' \text{ for some } \tau' \text{ s.t. } \langle n, n, \tau', 0 \rangle \in (\!|\mathsf{node}|\!)^t_G \tag{9}$$

  So from (8) and (9):

  $$t \notin \bigcup \{\tau' \mid \langle n, n, \tau', 0 \rangle \in (\!|\mathsf{node}|\!)^t_G\}$$

  Therefore
  $$t \in \mathcal{T}_G \setminus \bigcup \{\tau' \mid \langle n, n, \tau', 0 \rangle \in (\!|\mathsf{node}|\!)^t_G\} \tag{10}$$

  So $t \in \tau$ for some $\tau \in \mathsf{compl}\left(\bigcup\{\tau' \mid \langle n, n, \tau', 0 \rangle \in (\!|\mathsf{node}|\!)^t_G\}, \mathcal{T}_G\right)$.
  And $\langle n, n, \tau, 0 \rangle \in (\!|q|\!)^t_G$, from the definition of $(\!|q|\!)^t_G$.
- For II, let $\langle n, n, \tau, 0 \rangle \in (\!|q|\!)^t_G$.
  And take any $t \in \tau$.
  From the definition of $(\!|q|\!)^t_G$:

  $$t \in \mathcal{T}_G \setminus \bigcup \{\tau' \mid \langle n, n, \tau', 0 \rangle \in (\!|\mathsf{node}|\!)^t_G\}$$

  Together with (9), this implies

  $$\langle n, n, t, 0 \rangle \notin [\![\mathsf{node}]\!]_G$$

  Therefore $\langle n, n, t, 0 \rangle \in [\![q]\!]_G$, from the definition of $[\![q]\!]_G$.

- $q = \mathsf{trpq}_1 / \mathsf{trpq}_2$.
  From the above definitions, we have:

  $$[\![q]\!]_G = \{\langle n_1, n_3, t, d_1 + d_2 \rangle \mid \exists n_2 \colon \langle n_1, n_2, t, d_1 \rangle \in [\![\mathsf{trpq}_1]\!]_G \text{ and } \langle n_2, n_3, t + d_1, d_2 \rangle \in [\![\mathsf{trpq}_2]\!]_G\}$$
  $$(\!|q|\!)^t_G = \Big\{\langle n_1, n_3, ((\tau_1 + d_1) \cap \tau_2) - d_1, d_1 + d_2 \rangle \mid$$
  $$\exists n_2 \colon \langle n_1, n_2, \tau_1, d_1 \rangle \in (\!|\mathsf{trpq}_1|\!)^t_G, \langle n_2, n_3, \tau_2, d_2 \rangle \in (\!|\mathsf{trpq}_2|\!)^t_G \text{ and } (\tau_1 + d_1) \cap \tau_2 \neq \emptyset \Big\}$$

  - For I, let $\mathbf{v} = \langle n_1, n_3, t, d \rangle \in [\![q]\!]_G$.
    Fom the definition of $[\![q]\!]_G$, there are n_2, d_1 and d_2 such that
    $\langle n_1, n_2, t, d_1 \rangle \in [\![\mathsf{trpq}_1]\!]_G$, $\langle n_2, n_3, t + d_1, d_2 \rangle \in [\![\mathsf{trpq}_2]\!]_G$ and $d = d_1 + d_2$.
    By IH, because $\langle n_1, n_2, t, d_1 \rangle \in [\![\mathsf{trpq}_1]\!]_G$, there is a τ_1 such that $t \in \tau_1$
    and
    $$\langle n_1, n_2, \tau_1, d_1 \rangle \in (\!|\mathsf{trpq}_1|\!)^t_G \tag{11}$$

    And similarly, because $\langle n_2, n_3, t + d_1, d_2 \rangle \in [\![\mathsf{trpq}_2]\!]_G$, there is a τ_2 such
    that $t + d_1 \in \tau_2$ and

    $$\langle n_2, n_3, \tau_2, d_2 \rangle \in (\!|\mathsf{trpq}_2|\!)^t_G \tag{12}$$

From $t \in \tau_1$, we get

$$t + d_1 \in \tau_1 + d_1 \tag{13}$$

Together with the fact that $t + d_1 \in \tau_2$, this implies

$$\tau_1 + d_1 \cap \tau_2 \neq \emptyset \tag{14}$$

So from (11), (12), (14) and the definition of $(\!|q|\!)_G^t$,

$$\langle n_1, n_2, ((\tau_1 + d_1) \cap \tau_2) - d_1, d_1 + d_2 \rangle \in (\!|q|\!)_G^t$$

which proves Ia.
And in order to prove Ib, we only need to show that

$$t \in ((\tau_1 + d_1) \cap \tau_2) - d_1$$

We know that $t \in \tau_1$, therefore

$$t + d_1 \in \tau_1 + d_1$$

Together with the fact that $t + d_1 \in \tau_2$, this yields

$$t + d_1 \in (\tau_1 + d_1) \cap \tau_2$$
$$t \in ((\tau_1 + d_1) \cap \tau_2) - d_1$$

- For II, let $\mathbf{u} = \langle n_1, n_3, \tau, d \rangle \in (\!|q|\!)_G^t$, and let $t \in \tau$.
  We show that $\langle n_1, n_3, t, t + d \rangle \in [\![q]\!]_G$.
  Because $\mathbf{u} \in (\!|q|\!)_G^t$, from the definition of $(\!|q|\!)_G^t$, there are τ_1, τ_2, d_1, d_2 and n_2 s.t.:
  (i) $d = d_1 + d_2$
  (ii) $\tau = ((\tau_1 + d_1) \cap \tau_2) - d_1$
  (iii) $\langle n_1, n_2, \tau_1, d_1 \rangle \in (\!|\mathsf{trpq}_1|\!)_G^t$
  (iv) $\langle n_2, n_3, \tau_2, d_2 \rangle \in (\!|\mathsf{trpq}_2|\!)_G^t$
  Since $t \in \tau$, from ii, we have

$$t \in ((\tau_1 + d_1 \cap \tau_2) - d_1 \tag{15}$$
$$t + d_1 \in (((\tau_1 + d_1 \cap \tau_2) - d_1) + d_1 \tag{16}$$
$$t + d_1 \in (\tau_1 + d_1) \cap \tau_2 \tag{17}$$
$$t + d_1 \in \tau_1 + d_1 \tag{18}$$
$$t \in \tau_1 \tag{19}$$

From iii, by IH, for any $t' \in \tau_1$

$$\langle n_1, n_2, t' + d_1 \rangle \in [\![q]\!]_G$$

In particular, from (19)

$$\langle n_1, n_2, t, t + d_1 \rangle \in [\![q]\!]_G \tag{20}$$

And from iv, by IH, for any $t'' \in \tau_2$

$$\langle n_2, n_3, t'', t'' + d_2 \rangle \in [\![q]\!]_G$$

In particular, from (17)

$$\langle n_2, n_3, t + d_1, (t + d_1) + d_2 \rangle \in [\![q]\!]_G \qquad (21)$$

So from (20), (21) and the definition of $[\![q]\!]_G$

$$\langle n_1, n_3, t, t + d_1 + d_2 \rangle \in [\![q]\!]_G$$

In $\mathcal{U}^d$

Definition. We start with the case where q is an expression for the symbol node or edge in the grammar of Section 3.

$$(\![label]\!)_G^d = \{\langle n_1, n_2, t, [0,0] \rangle \mid t \in \tau \text{ for some } \tau \in \mathsf{val}_G(n_1, label, n_2)\}$$

$$(\![edge^-]\!)_G^d = \{\langle n_2, n_1, t, [0,0] \rangle \mid \langle n_1, n_2, t, [0,0] \rangle \in (\![edge]\!)_G^d\}$$

$$(\![pred]\!)_G^d = \{\langle n, n, t, [0,0] \rangle \mid n \models pred \text{ and } t \in \mathcal{T}_G\}$$

$$(\![\leq k]\!)_G^d = \{\langle n, n, t, [0,0] \rangle \mid n \in \mathcal{N}_G, t \in \mathcal{T}_G \text{ and } t \leq k\}$$

$$(\![(?trpq)]\!)_G^d = \{\langle n, n, t, [0,0] \rangle \mid \delta \colon \langle n, n', t, \delta \rangle \in (\![trpq]\!)_G^d \text{ for some } n' \in \mathcal{N} \text{ and } \delta \in \mathsf{intv}(\mathcal{T})\}$$

$$(\![\neg node]\!)_G^d = \Big\{\langle n, n, t, [0,0] \rangle \mid n \in \mathcal{N}_G \text{ and } t \in \mathcal{T}_G \setminus \{t' \mid \langle n, n, t', [0,0] \rangle \in (\![node]\!)_G^d\}\Big\}$$

Next, we consider the operators $(trpq_1 + trpq_2)$, $(trpq[m, \_])$ and $(trpq[m, n])$. For these cases, $(\![q]\!)_G^{td}$ is once again defined analogously to $[\![q]\!]_G$, in terms of temporal join (a.k.a. $trpq_1/trpq_2$) and set union.
We only write the definitions here for the sake of completeness:

$$(\![trpq_1 + trpq_2]\!)_G^{td} = (\![trpq_1]\!)_G^{td} \cup (\![trpq_2]\!)_G^{td}$$

$$[\![trpq[m, n]]\!]_G = \bigcup_{k=m}^{n} (\![trpq^k]\!)_G^{td}$$

$$[\![trpq[m, \_]]\!]_G = \bigcup_{k \geq m} (\![trpq^k]\!)_G^{td}$$

The only remaining operators are temporal join ($trpq_1/trpq_2$) and temporal navigation ($\mathbf{T}_\delta$), already defined in the article.
We reproduce here these two definition for convenience:

$$(\![trpq_1/trpq_2]\!)_G^d = \Big\{\langle n_1, n_3, t_1, t_2 - t_1 + \delta_2 \rangle \mid \exists n_2 \colon \langle n_1, n_2, t_1, \delta_1 \rangle \in (\![trpq_1]\!)_G^d,$$
$$\langle n_2, n_3, t_2, \delta_2 \rangle \in (\![trpq_2]\!)_G^d \text{ and } t_2 - t_1 \in \delta_1\Big\}$$

$$(\![\mathbf{T}_\delta]\!)_G^d = \Big\{\langle n, n, t, ((\delta + t) \cap \mathcal{T}_G) - t \rangle \mid n \in \mathcal{N}_G, t \in \mathcal{T}_G \text{ and } (\delta + t) \cap \mathcal{T}_G \neq \emptyset\Big\}$$

We also reproduce the alternative characterization of $(\![\mathbf{T}_\delta]\!)_G^d$ provided in the article, as a unary operator:

$$(\![q/\mathbf{T}_\delta]\!)_G^d = \Big\{\langle n_1, n_2, t, (\delta' + \delta) \cap \mathcal{T}_G \rangle \mid \langle n_1, n_2, t, \delta' \rangle \in (\![q]\!)_G^d \text{ and } (t + (\delta' + \delta)) \cap \mathcal{T}_G \neq \emptyset\Big\}$$

Correctness. The following result states that the representation $(\![q]\!)_G^d$ is correct:

Proposition 6. *Let $G = \langle \mathcal{T}_G, \mathcal{F}_G, \mathsf{val}_G \rangle$ be a temporal graph and q a TRPQ. Then the unfolding of $(\![q]\!)_G^d$ is $[\![q]\!]_G$.*

Proof.
Let $G = \langle \mathcal{T}_G \mathcal{F}_G, \mathsf{val}_G \rangle$ be a temporal graph, and let q be a TRPQ.
We show below that:

(I) for any $\langle n_1, n_2, t, d \rangle \in [\![q]\!]_G$, there is a $\delta \in \mathsf{intv}(\mathcal{T})$ such that
    (a) $\langle n_1, n_2, t, \delta \rangle \in (\![q]\!)_G^d$, and
    (b) $d \in \delta$,
(II) for any $\langle n_1, n_2, t, \delta \rangle \in (\![q]\!)_G^d$ for any $d \in \delta$,
    $\langle n_1, n_2, t, d \rangle$ is in $[\![q]\!]_G$.

We proceed once again by induction on the structure of q.
If q is of the form *label*, edge^-, *pred*, $\leq k$, $\neg\mathsf{node}$, $(\mathsf{trpq} + \mathsf{trpq})$, $\mathsf{trpq}[m, n]$ or $\mathsf{trpq}[m, \_]$, then I and II immediately follow from the definitions of $[\![q]\!]_G$ and $(\![q]\!)_G^d$.
If q is of the form $(?\mathsf{trpq})$, then the proof is nearly identical to the one already provided for $(\![(?\mathsf{trpq})]\!)_G^t$.
So we focus below on the two remaining cases:

− $q = \mathbf{T}_\delta$.
    From the definitions above, we have:

$$[\![q]\!]_G = \{\langle n, n, t, d \rangle \mid n \in \mathcal{N}_G, t \in \mathcal{T}_G, d \in \delta \text{ and } t + d \in \mathcal{T}_G\}$$
$$(\![q]\!)_G^d = \{\langle n, n, t, ((\delta + t) \cap \mathcal{T}_G) - t \rangle \mid n \in \mathcal{N}_G, t \in \mathcal{T}_G \text{ and } (\delta + t) \cap \mathcal{T}_G \neq \emptyset\}$$

    • For I, let $\mathbf{v} = \langle n, n, t, d \rangle \in [\![q]\!]_G$.
    And let $\mathbf{u} = \langle n, n, t, ((\delta + t) \cap \mathcal{T}_G) - t \rangle$ in $\mathcal{U}^d$.
    For Ia we show that $\mathbf{u} \in (\![q]\!)_G^d$.
    From $\mathbf{v} \in [\![q]\!]_G$, we get $n \in \mathcal{N}_G$ and $t \in \mathcal{T}_G$.
    Besides, because $\mathbf{v} \in [\![q]\!]_G$ still,

$$t + d \in \mathcal{T}_G \tag{22}$$

    and

$$d \in \delta \tag{23}$$
$$t + d \in t + \delta \tag{24}$$

    So from (22) and (24)

$$t + d \in (\delta + t) \cap \mathcal{T}_G \tag{25}$$
$$(\delta + t) \cap \mathcal{T}_G \neq \emptyset \tag{26}$$

Together with the definition of $(\!|q|\!)^d_G$, this implies $\mathbf{u} \in (\!|q|\!)^d_G$, which concludes the proof for Ia.

Finally, from (25), we get

$$t + d - t \in ((\delta + t) \cap \mathcal{T}_G) - t \tag{27}$$

$$d \in ((\delta + t) \cap \mathcal{T}_G) - t \tag{28}$$

which proves Ib.

- For II, let $\mathbf{u} = \langle n, n, t, ((\delta+t)\cap\mathcal{T}_G)-t \rangle \in (\!|q|\!)^d_G$, and let $d \in ((\delta+t)\cap\mathcal{T}_G)-t$. From $\mathbf{u} \in (\!|q|\!)^d_G G$, we get $n \in \mathcal{N}_G$ and $t \in \mathcal{T}_G$.

  So to conclude the proof, it is sufficient to show that *(i)* $d \in \delta$ and *(ii)* $t + d \in \mathcal{T}_G$.

  By assumption, we have

  $$d \in ((\delta + t) \cap \mathcal{T}_G) - t \tag{29}$$

  $$d + t \in (\delta + t) \cap \mathcal{T}_G \tag{30}$$

  $$d + t \in \mathcal{T}_G \tag{31}$$

  which proves *(ii)* .

  And from (30), we also get

  $$d + t \in \delta + t$$

  $$d + t - t \in (\delta + t) - t$$

  $$d \in \delta$$

  which proves *(i)* .

- $q = \mathsf{trpq}_1/\mathsf{trpq}_2$.

  From the definitions above, we have:

  $$[\![q]\!]_G = \{\langle n_1, n_3, t, d_1 + d_2 \rangle \mid \exists n_2 \colon \langle n_1, n_2, t, d_1 \rangle \in [\![\mathsf{trpq}_1]\!]_G \text{ and } \langle n_2, n_3, t + d_1, d_2 \rangle \in [\![\mathsf{trpq}_2]\!]_G\}$$
  $$(\!|q|\!)^d_G = \Big\{ \langle n_1, n_3, t_1, \delta_2 + t_2 - t_1 \rangle \mid \exists n_2 \colon$$
  $$\langle n_1, n_2, t_1, \delta_1 \rangle \in (\!|\mathsf{trpq}_1|\!)^d_G, \langle n_2, n_3, t_2, \delta_2 \rangle \in (\!|\mathsf{trpq}_2|\!)^d_G \text{ and } t_2 - t_1 \in \delta_1 \Big\}$$

  - For I, let $\mathbf{v} = \langle n_1, n_3, t, d \rangle \in [\![q]\!]_G$.

    Fom the definition of $[\![q]\!]_G$, there are n_2, d_1 and d_2 such that $\langle n_1, n_2, t, d_1 \rangle \in [\![\mathsf{trpq}_1]\!]_G$, $\langle n_2, n_3, t + d_1, d_2 \rangle \in [\![\mathsf{trpq}_2]\!]_G$ and $d = d_1 + d_2$. By IH, because $\langle n_1, n_2, t, d_1 \rangle \in [\![\mathsf{trpq}_1]\!]_G$, there is a δ_1 such that $d_1 \in \delta_1$ and

    $$\langle n_1, n_2, t, \delta_1 \rangle \in (\!|\mathsf{trpq}_1|\!)^d_G \tag{32}$$

    And similarly, because $\langle n_2, n_3, t + d_1, d_2 \rangle \in [\![\mathsf{trpq}_2]\!]_G$, there is a δ_2 such that $d_2 \in \delta_2$ and

    $$\langle n_2, n_3, t + d_1, \delta_2 \rangle \in (\!|\mathsf{trpq}_2|\!)^d_G \tag{33}$$

    Next, since $d \in \delta_1$

    $$t + d_1 \in t + \delta_1 \tag{34}$$

So from (32), (33), (34) and the definition of $(\!|q|\!)_G^d$ (replacing t_1 with t and t_2 with $t + d_1$), we get

$$\langle n_1, n_2, t, \delta_2 + (t + d_1) - t \rangle \in (\!|q|\!)_G^d$$

which proves Ia.

And in order to prove Ib, we only need to show that $d \in \delta_2 + (t + d_1) - t$, or in other words that

$$d \in \delta_2 + d_1$$

We know that

$$d_2 \in \delta_2 \tag{35}$$

$$d_2 + d_1 \in \delta_2 + d_1 \tag{36}$$

Together with the fact that $d = d_1 + d_2$, this concludes the proof for Ib.

- For II, let $\mathbf{u} = \langle n_1, n_3, t_1, \delta \rangle \in (\!|q|\!)_G^d$, and let $d \in \delta$.

Because $\mathbf{u} \in (\!|q|\!)_G^d$, from the definition of $(\!|q|\!)_G^d$, there are δ_1, δ_2, t_2 and n_2 s.t.:

(i) $\delta = \delta_2 + t_2 - t_1$
(ii) $t_2 \in t_1 + \delta_1$
(iii) $\langle n_1, n_2, t_1, \delta_1 \rangle \in (\!|\mathsf{trpq}_1|\!)_G^d$
(iv) $\langle n_2, n_3, t_2, \delta_2 \rangle \in (\!|\mathsf{trpq}_2|\!)_G^d$

From i and ii, we get

$$\delta = \delta_2 + (t_1 + \delta_1) - t_1$$
$$= \delta_2 + \delta_1$$

Together with $d \in \delta$, this implies that there are $d_1 \in \delta_1$ and $d_2 \in \delta_2$ such that $d = d_1 + d_2$.

Next, because $d_1 \in \delta_1$, from iii, by IH

$$\langle n_1, n_2, t_1, t_1 + d_1 \rangle \in [\![q]\!]_G \tag{37}$$

And similarly, because $d_2 \in \delta_2$, from iv

$$\langle n_2, n_3, t_2, t_2 + d_2 \rangle \in [\![q]\!]_G \tag{38}$$

So from (37), (38) and the definition of $[\![q]\!]_G$

$$\langle n_1, n_3, t_1, d_1 + d_2 \rangle \in [\![q]\!]_G \tag{39}$$

Together with the fact that $d = d_1 + d_2$, this concludes the proof for II.

In $\mathcal{U}^{td}$

Definition. We start with the case where q is an expression for the symbol edge or node in the grammar of Section 3.

As a consequence of Lemma 1, $(\!|q|\!)_G^{td}$ can be trivially defined out of $(\!|q|\!)_G^t$ by replacing the distance 0 with the interval $[0,0]$, i.e.

$$(\!|\mathsf{edge}|\!)_G^{td} = \{\langle n_1, n_2, \tau, [0,0]\rangle \mid \{\langle n_1, n_1, \tau, 0\rangle \in (\!|\mathsf{edge}|\!)_G^t\}$$
$$(\!|\mathsf{node}|\!)_G^{td} = \{\langle n, n, \tau, [0,0]\rangle \mid \{\langle n, n, \tau, 0\rangle \in (\!|\mathsf{node}|\!)_G^t\}$$

Next, if q is of the form $(\mathsf{trpq}_1 + \mathsf{trpq}_2)$, $(\mathsf{trpq}[m, \_])$ or $(\mathsf{trpq}[m, n])$, then the definition of $(\!|q|\!)_G^{td}$ is once again nearly identical to the one of $[\![q]\!]_G$:

$$(\!|\mathsf{trpq}_1 + \mathsf{trpq}_2|\!)_G^{td} = (\!|\mathsf{trpq}_1|\!)_G^{td} \cup (\!|\mathsf{trpq}_2|\!)_G^{td}$$
$$[\![\mathsf{trpq}[m, n]]\!]_G = \bigcup_{k=m}^{n} (\!|\mathsf{trpq}^k|\!)_G^{td}$$
$$[\![\mathsf{trpq}[m, \_]]\!]_G = \bigcup_{k \geq m} (\!|\mathsf{trpq}^k|\!)_G^{td}$$

The only remaining operators are temporal join $(\mathsf{trpq}_1/\mathsf{trpq}_2)$ and temporal navigation $(\mathbf{T}_\delta)$:

$$(\!|\mathsf{trpq}_1/\mathsf{trpq}_2|\!)_G^{td} = \bigcup \{\mathbf{u}_1 \bowtie \mathbf{u}_2 \mid \mathbf{u}_1 \in (\!|\mathsf{trpq}_1|\!)_G^{td}, \mathbf{u}_2 \in (\!|\mathsf{trpq}_2|\!)_G^{td}\}$$
$$(\!|\mathbf{T}_\delta|\!)_G^{td} = \bigcup_{n \in \mathcal{N}_G} \{\langle n, n, \mathcal{T}_G, \delta\rangle \bowtie \langle n, n, \mathcal{T}_G, [0,0]\rangle\}$$

where $\mathbf{u}_1 \bowtie \mathbf{u}_2$ is defined as follows.

Let $\mathbf{u}_1 = \langle n_1, n_2, \tau_1, \delta_1\rangle$ and $\mathbf{u}_2 = \langle n_3, n_4, \tau_2, \delta_2\rangle$.

Define τ_2' as

$$\tau_2' = (\tau_1 + \delta_1) \cap \tau_2$$

If $n_2 \neq n_3$ or $\tau_2' = \emptyset$, then $\mathbf{u}_1 \bowtie \mathbf{u}_2 = \emptyset$.

Otherwise, let:

$$\tau = (\tau_2' \ominus \delta_1) \cap \tau_1$$
$$b = b_{\tau_2'} - b_{\delta_1}$$
$$e = e_{\tau_2'} - e_{\delta_1}$$

And for every $t \in \tau$, let

$$\delta(t) = {}_{\delta_1}\lfloor\ b_{\delta_1} + \max(0, b - t),\ e_{\delta_1} - \max(0, t - e)\ \rfloor_{\delta_1}$$

Then

$$\mathbf{u}_1 \bowtie \mathbf{u}_2 = \{\langle n_1, n_4, [t, t], \delta(t) + \delta_2\rangle \mid t \in \tau\}$$

Correctness. We start with a lemma:

Lemma 2. *Let* $\alpha, \beta \in \mathsf{intv}(\mathcal{T})$. *Then*

$$\beta \ominus \alpha = \{t \mid (t + \alpha) \cap \beta \neq \emptyset\}$$

Next, if $\mathbf{u} = \langle n_1, n_2, \tau, \delta \rangle \in \mathcal{U}^{td}$, we call *temporal relation induced by* $\mathbf{u}$ the set

$$\{(t, t+d) \mid t \in \tau, d \in \delta\}$$

We also use the operator $\bowtie : (\mathcal{T} \times \mathcal{T}) \times (\mathcal{T} \times \mathcal{T}) \rightarrow (\mathcal{T} \times \mathcal{T})$ for the composition of two binary relations, i.e.

$$R_1 \bowtie R_2 = \{t_1, t_3 \mid (t_1, t_2) \in R_1 \text{ and } (t_2, t_3) \in R_2 \text{ for some } t_2\}$$

We can now formulate the following lemma:

Lemma 3. *Let* $\mathbf{u}_1 = \langle n_1, n_2, \tau_1, \delta_1 \rangle$ *and* $\mathbf{u}_2 = \langle n_3, n_4, \tau_2, \delta_2 \rangle$ *be two tuples in* $\mathcal{U}^{td}$ *such that* $n_2 = n_3$. *And for* $i \in \{1, 2\}$, *let* R_i *denote the temporal relation induced by* $\mathbf{u}_i$. *Then*

$$R_1 \bowtie R_2 = \bigcup_{\langle n_1, n_4, \tau, \delta \rangle \in \mathbf{u}_1 \bowtie \mathbf{u}_2} \{(t, t+d) \mid t \in \tau, d \in \delta\}$$

Proof. $\mathbf{u}_1 = \langle n_1, n_2, \tau_1, \delta_1 \rangle$ and $\mathbf{u}_2 = \langle n_3, n_4, \tau_2, \delta_2 \rangle$ be two tuples in $\mathcal{U}^{td}$ such that $n_2 = n_3$. And for $i \in \{1, 2\}$, let R_i denote the temporal relation induced by $\mathbf{u}_i$.

We show that:

(I) (a) If $\tau_2' = \emptyset$, then $\mathsf{dom}(R_1 \bowtie R_2) = \emptyset$,
    (b) otherwise $\tau = \mathsf{dom}(R_1 \bowtie R_2)$,
(II) for each $t \in \tau$,

$$t + \delta(t) + \delta_2 = \{t' \mid (t, t') \in R_1 \bowtie R_2\}$$

We start with I.
From the defnition of "+" (applied to two intervals):

$$\tau_1 + \delta_1 = \{t + d \mid t \in \tau_1, d_1 \in \delta_1\} \tag{40}$$

So from the definition of R_1

$$\tau_1 + \delta_1 = \mathsf{range}(R_1) \tag{41}$$

Since $\tau_2 = \mathsf{dom}(R_2)$, this implies

$$(\tau_1 + \delta_1) \cap \tau_2 = \mathsf{range}(R_1) \cap \mathsf{dom}(R_2) \tag{42}$$

$$\tau_2' = \mathsf{range}(R_1) \cap \mathsf{dom}(R_2) \tag{43}$$

If $\mathsf{range}(R_1) \cap \mathsf{dom}(R_2) = \emptyset$, then $\mathsf{dom}(R_1 \bowtie R_2) = \emptyset$, immediately from the definition of $\bowtie$, which concludes the proof of Ia.
Otherwise, from Lemma 2,

$$\tau_2' \ominus \delta_1 = \{t \mid (t + \delta_1) \cap \tau_2' \neq \emptyset\} \tag{44}$$

So from (43)

$$\tau_2' \ominus \delta_1 = \{t \mid (t + \delta_1) \cap \mathsf{range}(R_1) \cap \mathsf{dom}(R_2) \neq \emptyset\}$$
$$(\tau_2' \ominus \delta_1) \cap \tau_1 = \{t \in \tau_1 \mid (t + \delta_1) \cap \mathsf{range}(R_1) \cap \mathsf{dom}(R_2) \neq \emptyset\}$$
$$(\tau_2' \ominus \delta_1) \cap \tau_1 = \mathsf{dom}(R_1 \bowtie R_2)$$
$$\tau = \mathsf{dom}(R_1 \bowtie R_2)$$

which proves Ib.

Now for II, let $t \in \tau$.
We show below that *(i)* $t + \delta(t) = \{t' \mid (t, t') \in R_1 \text{ and } t' \in \mathsf{range}(R_1) \cap \mathsf{dom}(R_2)\}$. Together with the definition of $\bowtie$ (and the fact that $t + \delta(t)$ is an interval), this proves II.

We only prove the result for the case where τ, τ_2' and δ_1 are closed-closed intervals (the proof for the other 63 cases is symmetric).
First, from Ib and the assumption that $t \in \tau$, we have $t \in \tau_1$. So from the definition of R_1,

$$t + \delta_1 = \{t' \mid (t, t') \in R_1\} \tag{45}$$

Together with (43), this means that *(i)* is equivalent to *(ii)* $t + \delta(t) = \{(t + \delta_1) \cap \tau_2'\}$.
So in order to prove II (and conclude our proof), it is sufficient to prove *(ii)* .

Now since $t \in \tau$, from Ib and the definition of τ_2', we have $(t + \delta(t)) \cap \tau_2' \neq \emptyset$.
And since $\delta(t)$ and τ_2' are intervals, $(t + \delta(t)) \cap \tau_2'$ is an also an interval.
So in order to prove *(ii)* , it is sufficient to show that $t + b_{\delta(t)}$ (resp. $t + e_{\delta(t)}$) is the smallest (resp. greatest) value in $(t + \delta_1) \cap \tau_2'$.
We only prove the result for $t + b_{\delta(t)}$ (the proof for $t + e_{\delta(t)}$) is symmetric.
We consider two cases.

- If $b \le t$, then

$$b_{\tau_2'} - b_{\delta_1} \le t \qquad\qquad \text{from the definition of } b \qquad (46)$$
$$b_{\tau_2'} - b_{\delta_1} + b_{\delta_1} \le t + b_{\delta_1} \qquad\qquad\qquad\qquad\qquad (47)$$
$$b_{\tau_2'} \le t + b_{\delta_1} \qquad\qquad\qquad\qquad\qquad (48)$$

And because $t \in \tau$

$$t \le e_\tau \qquad\qquad\qquad\qquad\qquad (49)$$
$$t \le e_{\tau_2'} - b_{\delta_1} \qquad\qquad \text{from the definition of } \tau \qquad (50)$$
$$t + b_{\delta_1} \le e_{\tau_2'} - b_{\delta_1} + b_{\delta_1} \qquad\qquad\qquad\qquad\qquad (51)$$
$$t + b_{\delta_1} \le e_{\tau_2'} \qquad\qquad\qquad\qquad\qquad (52)$$

So from (48) and (52)

$$t + b_{\delta_1} \in \tau_2' \tag{53}$$

Next, since $b \leq t$ (by assumption), we have

$$b - t \leq 0$$
$$\max(0, b - t) = 0$$

So from the definition of $\delta(t)$

$$b_{\delta(t)} = b_{\delta_1} \tag{54}$$

Therefore $t + b_{\delta(t)}$ is the smallest value in $t + \delta_1$.
So from (53), it is also the smallest value in $t + \delta_1 \cap \tau_2'$, which concludes the proof for this case.
– If $b > t$, then

$$b - t > 0 \tag{55}$$
$$\max(0, b - t) = b - t \tag{56}$$

So from the definition of $\delta(t)$

$$b_{\delta(t)} = b_{\delta_1} + b - t \tag{57}$$

Besides, from (55)
$$b - t + b_{\delta_1} > b_{\delta_1} \tag{58}$$

So from (57) and (58)
$$b_{\delta(t)} > b_{\delta_1} \tag{59}$$

Next, since $t \in \tau$
$$b_\tau \leq t \tag{60}$$

And from the definition of τ

$$b_{\tau_2'} - e_{\delta_1} \leq b_\tau \tag{61}$$

So from (60) and (61)

$$b_{\tau_2'} - e_{\delta_1} \leq t \tag{62}$$
$$b_{\tau_2'} - t \leq e_{\delta_1} \tag{63}$$
$$b_{\tau_2'} - t + b_{\delta_1} - b_{\delta_1} \leq e_{\delta_1} \tag{64}$$
$$b_{\delta_1} + (b_{\tau_2'} - b_{\delta_1}) - t \leq e_{\delta_1} \tag{65}$$
$$b_{\delta_1} + b - t \leq e_{\delta_1} \qquad \text{from the definition of } b \tag{66}$$
$$b_{\delta(t)} \leq e_{\delta_1} \qquad\qquad \text{from (57)} \tag{67}$$

Therefore from (59) and (67)

$$b_{\delta(t)} \in \delta_1 \tag{68}$$
$$t + b_{\delta(t)} \in t + \delta_1 \tag{69}$$

Finally, from (57) still,

$$t + b_{\delta(t)} = t + b_{\delta_1} + b - t \tag{70}$$

$$= t + b_{\delta_1} + b_{\tau_2'} - b_{\delta_1} - t \qquad \text{from the definition of } b \tag{71}$$

$$= b_{\tau_2'} \tag{72}$$

So $t + b_{\delta(t)}$ is the smallest value in τ_2'.
Together with (69), this concludes the proof for this case.

The following result states that the representation $(q)_G^{td}$ is correct:

Proposition 7. *Let* $G = \langle \mathcal{T}_G, \mathcal{F}_G, \mathsf{val}_G \rangle$ *be a temporal graph and* q *a TRPQ. Then the unfolding of* $(q)_G^{td}$ *is* $[\![q]\!]_G$.

Proof.
Let $G = \langle \mathcal{T}_G \mathcal{F}_G, \mathsf{val}_G \rangle$ be a temporal graph, and let q be a TRPQ.
We show below that:

(I) for any $\langle n_1, n_2, t, d \rangle \in [\![q]\!]_G$, there are $\tau, \delta \in \mathsf{intv}(\mathcal{T})$ such that
  (a) $\langle n_1, n_2, \tau, \delta \rangle \in (q)_G^{td}$,
  (b) $t \in \tau$, and
  (c) $d \in \delta$.
(II) for any $\langle n_1, n_2, \tau, \delta \rangle \in (q)_G^{td}$ for any $(t, d) \in \tau \times \delta$,
  $\langle n_1, n_2, t, d \rangle$ is in $[\![q]\!]_G$.

We proceed once again by induction on the structure of q.
If q is of the form *label*, edge^-, *pred*, $(\mathsf{trpq} + \mathsf{trpq})$, $\mathsf{trpq}[m, n]$ or $\mathsf{trpq}[m, \_]$, then I and II immediately follow from the definitions of $[\![q]\!]_G$ and $(q)_G^{td}$.
If q is of the form $\neg \mathsf{node}$ or $(?\mathsf{trpq})$, then the proof is nearly identical to the one already provided for $(q)_G^t$.
So we focus below on the two remaining cases:

− $q = \mathsf{trpq}_1 / \mathsf{trpq}_2$.
  From the above definitions, we have:

$$[\![q]\!]_G = \Big\{ \langle n_1, n_3, t, d_1 + d_2 \rangle \mid \exists n_2 :$$
$$\langle n_1, n_2, t, d_1 \rangle \in [\![\mathsf{trpq}_1]\!]_G \text{ and } \langle n_2, n_3, t + d_1, d_2 \rangle \in [\![\mathsf{trpq}_2]\!]_G \Big\}$$
$$(\mathsf{trpq}_1 / \mathsf{trpq}_2)_G^{td} = \bigcup \{ \mathbf{u}_1 \bowtie \mathbf{u}_2 \mid \mathbf{u}_1 \in (\mathsf{trpq}_1)_G^{td}, \mathbf{u}_2 \in (\mathsf{trpq}_2)_G^{td} \}$$

  • For I, let $\mathbf{v} = \langle n_1, n_3, t, d \rangle \in [\![q]\!]_G$.
    Fom the definition of $[\![q]\!]_G$, there are n_2, d_1 and d_2 such that $\langle n_1, n_2, t, d_1 \rangle \in [\![\mathsf{trpq}_1]\!]_G$, $\langle n_2, n_3, t + d_1, d_2 \rangle \in [\![\mathsf{trpq}_2]\!]_G$ and $d = d_1 + d_2$.
    By IH, because $\langle n_1, n_2, t, d_1 \rangle \in [\![\mathsf{trpq}_1]\!]_G$, there are τ_1 and δ_1 such that $t \in \tau_1$, $d_1 \in \delta_1$ and

$$\langle n_1, n_2, \tau_1, \delta_1 \rangle \in (\mathsf{trpq}_1)_G^{td} \tag{73}$$

    Let R_1 be the temporal relation induced by this tuple $\langle n_1, n_2, \tau_1, \delta_1 \rangle$.
    Since $t \in \tau_1$ and $d_1 \in \delta_1$, we have

$$(t, t + d_1) \in R_1 \tag{74}$$

Similarly, because $\langle n_2, n_3, t+d_1, d_2\rangle \in [\![\mathsf{trpq_2}]\!]_G$, there are τ_2 and δ_2 such that $t + d_1 \in \tau_2$, $d_2 \in \delta_2$ and

$$\langle n_2, n_3, \tau_2, \delta_2\rangle \in (\!|\mathsf{trpq_2}|\!)_G^{td} \tag{75}$$

Let R_2 be the temporal relation induced by this tuple $\langle n_2, n_3, \tau_2, \delta_2\rangle$. Since $t + d_1 \in \tau_2$ and $d_2 \in \delta_2$, we have

$$(t + d_1, t + d_1 + d_2) \in R_2 \tag{76}$$

So from (74), (76) and Lemma 3, there are τ and δ such that $\langle n_1, n_3, \tau, \delta\rangle \in u_1 \bowtie u_2$, $t \in \tau$ and $d_1 + d_2 = d \in \delta$, which concludes the proof for I.

- For II, let $\mathbf{u} = \langle n_1, n_3, t_1, \delta\rangle \in (\!|q|\!)_G^{td}$, and let $(t, d) \in \tau \times \delta$.
  Because $\mathbf{u} \in (\!|q|\!)_G^{td}$, from the definition of $(\!|q|\!)_G^{td}$, there are $\mathbf{u}_1$ and $\mathbf{u}_2$ s.t.:
  (i) $\mathbf{u} \in \mathbf{u}_1 \bowtie \mathbf{u}_2$
  (ii) $\mathbf{u}_1 \in (\!|\mathsf{trpq_1}|\!)_G^{td}$
  (iii) $\mathbf{u}_2 \in (\!|\mathsf{trpq_2}|\!)_G^{td}$
  Let R_i be the temporal relation induced by u_i for $i \in \{1, 2\}$.
  From i, and Lemma 3,

$$(t, t + d) \in R_1 \bowtie R_2 \tag{77}$$

Now let $\mathbf{u}_1 = \langle n_1, n_2, \tau_1, \delta_1\rangle$ and $\mathbf{u}_2 = \langle n_2, n_3, \tau_2, \delta_2\rangle$ for some $n_2, \tau_1, \tau_2, \delta_1$ and δ_2.
From (77) and the definition of $\bowtie$, there must be d_1 and d_2 s.t. $d = d_1 + d_2$, $t \in \tau_1$, $d_1 \in \delta_1$, $t + d_1 \in \tau_2$ and $d_2 \in \delta_2$.
So from ii, and iii, by IH

$$\langle n_1, n_2, t, d_1\rangle \in [\![\mathsf{trpq_1}]\!]_G \tag{78}$$
$$\langle n_2, n_3, t + d_1, d_2\rangle \in [\![\mathsf{trpq_2}]\!]_G \tag{79}$$

So from (78), (79) and the definition of $[\![q]\!]_G$

$$\langle n_1, n_3, t, d_1 + d_2\rangle \in [\![q]\!]_G,$$

which concludes the proof for II.

In $\mathcal{U}^c$

Definition. If q is an expression for the symbol edge or node in the grammar of Section 3, then the definition of $(\!|q|\!)_G^c$ is nearly identical to the one of $(\!|q|\!)_G^{td}$, extending each tuple $\{\langle n, n, \tau, [0,0]\rangle$ with b_τ and e_τ, i.e.

$$(\!|\mathsf{edge}|\!)_G^c = \{\langle n_1, n_2, \tau, [0,0], b_\tau, e_\tau\rangle \mid \{\langle n_1, n_2, \tau, [0,0]\rangle \in (\!|\mathsf{edge}|\!)_G^{td}\}$$
$$(\!|\mathsf{node}|\!)_G^c = \{\langle n, n, \tau, [0,0], b_\tau, e_\tau\rangle \mid \{\langle n, n, \tau, [0,0]\rangle \in (\!|\mathsf{node}|\!)_G^{td}\}$$

Next, if q is of the form $(\mathsf{trpq}_1 + \mathsf{trpq}_2)$, $(\mathsf{trpq}[m, \_])$ or $(\mathsf{trpq}[m, n])$, then the definition of $(\!|q|\!)_G^{td}$ is once again nearly identical to the one of $[\![q]\!]_G$:

$$(\!|\mathsf{trpq}_1 + \mathsf{trpq}_2|\!)_G^c = (\!|\mathsf{trpq}_1|\!)_G^c \cup (\!|\mathsf{trpq}_2|\!)_G^c$$

$$[\![\mathsf{trpq}[m, n]]\!]_G = \bigcup_{k=m}^{n} (\!|\mathsf{trpq}^k|\!)_G^c$$

$$[\![\mathsf{trpq}[m, \_]]\!]_G = \bigcup_{k \geq m} (\!|\mathsf{trpq}^k|\!)_G^c$$

So the only remaining operator are temporal join $(\mathsf{trpq}_1/\mathsf{trpq}_2)$ and temporal navigation $(\mathbf{T}_\delta)$, already defined in the article. We reproduce these two definition for convenience:

$$(\!|\mathsf{trpq}_1/\mathsf{trpq}_2|\!)_G^c = \{\mathbf{u}_1 \bowtie \mathbf{u}_2 \mid \mathbf{u}_1 \in (\!|\mathsf{trpq}_1|\!)_G^c, \mathbf{u}_2 \in (\!|\mathsf{trpq}_2|\!)_G^c \text{ and } \mathbf{u}_1 \sim \mathbf{u}_2\}$$
$$(\!|\mathbf{T}_\delta|\!)_G^c = \{\langle n, n, \mathcal{T}_G, \delta, b_{\mathcal{T}_G}, e_{\mathcal{T}_G} \rangle \bowtie \langle n, n, \mathcal{T}_G, [0, 0], b_{\mathcal{T}_G}, e_{\mathcal{T}_G} \rangle \mid n \in \mathcal{N}_G\}$$

where $\mathbf{u}_1 \sim \mathbf{u}_2$ and $\mathbf{u}_1 \bowtie \mathbf{u}_2$ are defined as follows.
Let $\mathbf{u}_1 = \langle n_1, n_2, \tau_1, \delta_1, b_1, e_1 \rangle$ and $\mathbf{u}_2 = \langle n_3, n_4, \tau_2, \delta_2, e_2, b_2 \rangle$.
Define

$$\delta_1' = {}_{\delta_1}\lfloor\, b_{\delta_1} + \max(0, b_1 - b_{\tau_1}), e_{\delta_1} - \max(0, e_{\tau_1} - e_1)\, \rfloor_{\delta_1}$$

and

$$\tau = (((\tau_1 + \delta_1') \cap \tau_2) \ominus \delta_1') \cap \tau_1$$

Then $\mathbf{u}_1 \sim \mathbf{u}_2$ iff $n_2 = n_3$ and $\tau \neq \emptyset$.
If $\mathbf{u}_1 \sim \mathbf{u}_2$, then $\mathbf{u}_1 \bowtie \mathbf{u}_2 = \langle n_1, n_4, \tau, \delta_1 + \delta_2, b, e \rangle$, with

$$b = \max(b_1, b_2 - b_{\delta_1})$$
$$e = \min(e_1, e_2 - e_{\delta_1})$$

Correctness. We start with two lemmas:

Lemma 4. *Let* $\mathbf{u} = \langle n_1, n_2, \tau, \delta, b, e \rangle \in \mathcal{U}^c$. *Then for any* $t_1, t_2 \in \tau$ *s.t.* $t_1 \leq t_2$:

$$t_1 + b_{\delta(t_1)} \leq t_2 + b_{\delta(t_2)} \text{ and}$$
$$t_1 + e_{\delta(t_1)} \leq t_2 + e_{\delta(t_2)}$$

Lemma 5. *Let* $\mathbf{u} = \langle n_1, n_2, \tau, \delta, b, e \rangle \in \mathcal{U}^c$. *And let* τ' *denote the interval* $(b_\tau + b_{\delta(b_\tau)}, e_\tau + e_{\delta(e_\tau)})$. *Then for any* $t' \in \tau'$, *there is a* $t \in \tau$ *s.t.* $t' \in t + \delta(t)$.

Next, similarly to what we did above for $\mathcal{U}^{td}$, if $\mathbf{u} = \langle n_1, n_2, \tau, \delta, b, e \rangle \in \mathcal{U}^c$, we call *temporal relation induced by* $\mathbf{u}$ the set

$$\{(t, t + d) \mid t \in \tau, d \in \delta(t)\}$$

We can now formulate a result analogous to Lemma 3:

Lemma 6. *Let* $\mathbf{u}_1, \mathbf{u}_2 \in \mathcal{U}^c$, *and for* $i \in \{1, 2\}$, *let* R_i *denote the temporal relation induced by* $\mathbf{u}_i$. *If* $\mathbf{u}_1 \sim \mathbf{u}_2$ *and* $\mathbf{u}_1 \bowtie \mathbf{u}_2 = \langle n_1, n_3, \tau, \delta, b, e \rangle$, *then*

$$R_1 \bowtie R_2 = \{(t, t + d) \mid t \in \tau, d \in \delta(t)\}$$

Proof. Let $\mathbf{u}_1 = \langle n_1, n_2, \tau_1, \delta_1 \rangle$ and $\mathbf{u}_2 = \langle n_2, n_3, \tau_2, \delta_2 \rangle$.

As explained in Section A.1, for $i \in \{1, 2\}$ and $t \in \tau_i$, we use $\delta_i(t)$ for the interval

$$\delta_i \lfloor\, b_{\delta_i} + \max(0, b_i - t) \;,\; e_{\delta_i} - \max(0, t - e_i)\, \rfloor_{\delta_i}$$

We need to prove that *(i)* $\tau = \mathsf{dom}(R_1 \bowtie R_2)$ and that *(ii)* for each $t \in \tau$,

$$t + \delta(t) = \{t' \mid (t, t') \in R_1 \bowtie R_2\}$$

The proof of *(i)* is nearly identical to the one provided above for Lemma 3.

For *(ii)* , let $t \in \tau$.

We only provide a proof for the case where τ, δ_1 and δ_2 are closed-closed intervals (the proof for the other 63 cases is symmetric).

Since $t \in \tau$, from the defnition of τ, $t \in \tau_1$.

Therefore from the definition of R_1,

$$t + \delta_1(t) = \{t' \mid (t, t') \in R_1\} \tag{80}$$

So from *(i)* and the fact that $t \in \tau$

$$t + \delta_1(t) \cap \mathsf{dom}(R_2) \neq \emptyset \tag{81}$$

Now let a (resp. z) denote the smallest (resp. largest) value in $t + \delta_1(t) \cap \mathsf{dom}(R_2)$. Then from (80), a (resp. z) is also the smallest value s.t. $(t, a) \in R_1$ and $a \in \mathsf{dom}(R_2)$ (resp. the largest value s.t. $(t, z) \in R_1$ and $z \in \mathsf{dom}(R_2)$).

Next, from Lemma 4, for any $x \in [a, z]$, we have

$$a + b_{\delta_2(a)} \leq x + b_{\delta_2(x)} \tag{82}$$

and

$$x + e_{\delta_2(x)} \leq z + e_{\delta_2(z)} \tag{83}$$

Now let a' and z' denote $a + b_{\delta_2(a)}$ and $z + e_{\delta_2(z)}$ respectively.

From (82) and the definition of R_2, a' is the smallest value s.t. $(x, a') \in R_2$ for some $x \in [a, b]$.

And similarly, from (83) and the definition of R_2, z' is the largest value s.t. $(x, z') \in R_2$ for some $x \in [a, b]$.

Together with the defnition of a (resp. of z), this implies that a' (resp. z') is also the smallest (resp. largest) value s.t. $(t, a') \in R_1 \bowtie R_2$ (resp. $(t, z') \in R_1 \bowtie R_2$).

To conclude the proof, we show that:

1. $(t, x) \in R_1 \bowtie R_2$ for each $x \in [a', z']$, and
2. $t + \delta(t) = [a', z']$.

We start with 1.

Consider the tuple $\mathbf{u}' = \langle n_2, n_3, [a, b], \delta_2, b_2, e_2 \rangle \in \mathcal{U}^c$, and let R' be the temporal relation induced by $\mathbf{u}'$.

Then from he definitions of u' and $\mathbf{u}_2$:

$$R' \subseteq R_2 \tag{84}$$

Now take any $x \in [a', z']$.

From Lemma 5 and the definitions of a' and z', there is a $w \in [a, b]$ such that $x \in \delta_2(w)$.

Therefore

$$(w, x) \in R'$$

So from (84)

$$(w, x) \in R_2 \tag{85}$$

Finally, since $[a, b] = t + \delta_1(t)$ and $w \in [a, b]$,

$$(t, w) \in R_1 \tag{86}$$

Together with (85), this implies

$$(t, x) \in R_1 \bowtie R_2$$

which concludes the proof for 1.

For 2, we only prove that $t + b_{\delta_t} = a'$ (the proof that $t + e_{\delta_t} = z'$ is symmetric). Following the definition of b, we consider 2 cases:

1. $b_1 < b_2 - b_{\delta_1}$
2. $b_1 \geq b_2 - b_{\delta_1}$

In Case 1, we have

$$
\begin{aligned}
b_1 &< & b_2 - b_{\delta_1} & \tag{87} \\
\max(b_1, b_2 - b_{\delta_1}) &= & b_2 - b_{\delta_1} & \tag{88} \\
b &= & b_2 - b_{\delta_1} & \quad \text{from the definition of } b \tag{89}
\end{aligned}
$$

And (in Case 1 still):

$$
\begin{aligned}
b_1 &< & b_2 - b_{\delta_1} & \tag{90} \\
0 &< & b_2 - b_{\delta_1} - b_1 & \tag{91} \\
\max(0, b_2 - b_{\delta_1} - b_1) &= & b_2 - b_{\delta_1} - b_1 & \tag{92}
\end{aligned}
$$

Then we consider two subcases:

(i) $t < b_2 - b_{\delta_1}$
(ii) $t \geq b_2 - b_{\delta_1}$

In Case i, we get

$$
\begin{aligned}
t &< & b_2 - b_{\delta_1} & \tag{93} \\
0 &< & b_2 - b_{\delta_1} - t & \tag{94} \\
\max(0, b_2 - b_{\delta_1} - t) &= & b_2 - b_{\delta_1} - t & \tag{95}
\end{aligned}
$$

Now from the definition of δ_t,

$$
\begin{align}
b_{\delta_t} &= b_{\delta_1} + b_{\delta_2} + \max(0, b - t) \tag{96}\\
&= b_{\delta_1} + b_{\delta_2} + \max(0, b_2 - b_{\delta_1} - t) \qquad \text{from (89)} \tag{97}\\
&= b_{\delta_1} + b_{\delta_2} + b_2 - b_{\delta_1} - t \qquad \text{from (95)} \tag{98}\\
&= b_{\delta_2} + b_2 - t \tag{99}\\
b_{\delta_t} + t &= b_{\delta_2} + b_2 - t + t \tag{100}\\
&= b_{\delta_2} + b_2 \tag{101}
\end{align}
$$

Next, from the definition of a'

$$
\begin{align}
a' &= b_{\delta_2(a)} + a \tag{102}\\
&= b_{\delta_2} + \max(0, b_2 - a) + a \tag{103}
\end{align}
$$

And, from the definition of a

$$
\begin{align}
a &= b_{\delta_1(t)} + t \tag{104}\\
&= b_{\delta_1} + \max(0, b_1 - t) + t \tag{105}
\end{align}
$$

Then we have two further subcases:

(I) $t \geq b_1$, or
(II) $t < b_1$

In case I:

$$
\begin{align}
t &\geq b_1 \tag{106}\\
0 &\geq b_1 - t \tag{107}\\
\max(0, b_1 - t) &= 0 \tag{108}\\
a &= b_{\delta_1} + t \qquad \text{from (105)} \tag{109}\\
\max(0, b_2 - a) &= \max(0, b_2 - b_{\delta_1} - t) \tag{110}\\
&= b_2 - b_{\delta_1} - t \qquad \text{from (95)} \tag{111}\\
&= b_2 - a \qquad \text{from (109)} \tag{112}
\end{align}
$$

In case II:

$$
\begin{align}
t &< b_1 \tag{113}\\
0 &< b_1 - t \tag{114}\\
\max(0, b_1 - t) &= b_1 - t \tag{115}\\
a &= b_{\delta_1} + b_1 - t + t \qquad \text{from (105)} \tag{116}\\
&= b_{\delta_1} + b_1 \tag{117}\\
\max(0, b_2 - a) &= \max(0, b_2 - b_{\delta_1} - b_1) \tag{118}\\
&= b_2 - b_{\delta_1} - b_1 \qquad \text{from (92)} \tag{119}\\
&= b_2 - a \qquad \text{from (117)} \tag{120}\\
& \tag{121}
\end{align}
$$

So in both cases I and II, we get

$$\max(0, b_2 - a) = b_2 - a$$

Thefore from (103)

$$
\begin{align}
a' &= b_{\delta_2} + b_2 - a + a \tag{122} \\
&= b_{\delta_2} + b_2 \tag{123} \\
&= t + b_{\delta_t} \qquad \text{from (101)} \tag{124}
\end{align}
$$

which concludes the proof for Case 1 i.

We continue with Case 1 ii.
From ii:

$$
\begin{align}
t &\geq b_2 - b_{\delta_1} \tag{125} \\
0 &\geq b_2 - b_{\delta_1} - t \tag{126} \\
\max(0, b_2 - b_{\delta_1} - t) &= 0 \tag{127}
\end{align}
$$

Now from the definition of δ_t:

$$
\begin{align}
b_{\delta_t} &= b_{\delta_1} + b_{\delta_2} + \max(0, b - t) \tag{128} \\
&= b_{\delta_1} + b_{\delta_2} + \max(0, b_2 - b_1 - t) \qquad \text{from (89)} \tag{129} \\
&= b_{\delta_1} + b_{\delta_2} \qquad \text{from (127)} \tag{130} \\
b_{\delta_t} + t &= b_{\delta_1} + b_{\delta_2} + t \tag{131}
\end{align}
$$

Next, from 1 and ii, by transitivity, we get

$$
\begin{align}
b_1 &\leq t \tag{132} \\
\max(0, b_1 - t) &= 0 \tag{133}
\end{align}
$$

And from the definition of a

$$
\begin{align}
a &= b_{\delta_1(t)} + t \tag{134} \\
&= b_{\delta_1} + \max(0, b_1 - t) + t \tag{135} \\
&= b_{\delta_1} + t \qquad \text{from (133)} \tag{136} \\
&\geq b_{\delta_1} + b_2 - b_{\delta_1} \qquad \text{from Case ii} \tag{137} \\
&\geq b_2 \tag{138} \\
0 &\geq b_2 - a \tag{139} \\
\max(0, b_2 - a) &= 0 \tag{140}
\end{align}
$$

Therefore from (103) and (140)

$$
\begin{align}
a' &= b_{\delta_2} + a \tag{141} \\
&= b_{\delta_2} + b_{\delta_1} + t \qquad \text{from (136)} \tag{142} \\
&= b_{\delta_t} + t \qquad \text{from (101)} \tag{143}
\end{align}
$$

which concludes the proof for Case 1 ii.

We continute with Case 2.
In this case, we get

$$b_1 \geq b_2 - b_{\delta_1} \tag{144}$$
$$\max(b_1, b_2 - b_{\delta_1}) = b_1 \tag{145}$$
$$b = b_1 \qquad \text{from the definition of } b \tag{146}$$

And from 2 still, we derive

$$b_1 \geq b_2 - b_{\delta_1} \tag{147}$$
$$0 \geq b_2 - b_{\delta_1} - b_1 \tag{148}$$
$$\max(0, -b_{\delta_1} - b_1) = 0 \tag{149}$$

As well as

$$b_1 \geq b_2 - b_{\delta_1} \tag{150}$$
$$b_1 + b_{\delta_1} \geq b_2 \tag{151}$$

Next, we distinguish two subcases, namely

(a) $t < b_1$ and
(b) $t \geq b_1$

We start with Case a.
In this case,

$$t < b_1 \tag{152}$$
$$0 < b_1 - t \tag{153}$$
$$\max(0, b_1 - t) = b_1 - t \tag{154}$$

And from the definition of δ_t:

$$b_{\delta_t} = b_{\delta_1} + b_{\delta_2} + \max(0, b - t) \tag{155}$$
$$= b_{\delta_1} + b_{\delta_2} + \max(0, b_1 - t) \qquad \text{from (146)} \tag{156}$$
$$= b_{\delta_1} + b_{\delta_2} + b_1 - t \qquad \text{from (154)} \tag{157}$$
$$b_{\delta_t} + t = b_{\delta_1} + b_{\delta_2} + b_1 - t + t \tag{158}$$
$$= b_{\delta_1} + b_{\delta_2} + b_1 \tag{159}$$

Next, from the definition of a

$$a = b_{\delta_1(t)} + t \tag{160}$$
$$= \max(0, b_1 - t) + b_{\delta_1} + t \tag{161}$$
$$= b_1 - t + b_{\delta_1} + t \qquad \text{from (154)} \tag{162}$$
$$= b_1 + b_{\delta_1} \tag{163}$$

So from (151)

$$
\begin{aligned}
a &\geq && b_2 & &&&\text{(164)}\\
0 &\geq && b_2 - a & &&&\text{(165)}\\
\max(0, b_2 - a) &= && 0 & &&&\text{(166)}\\
b_{\delta_2} + \max(0, b_2 - a) &= && b_{\delta_2} & &&&\text{(167)}\\
b_{\delta_2(a)} &= && b_{\delta_2} & &&&\text{(168)}\\
b_{\delta_2(a)} + a &= && b_{\delta_2} + a & &&&\text{(169)}\\
a' &= && b_{\delta_2} + a & &\text{from the definition of } a' &&\text{(170)}\\
a' &= && b_{\delta_2} + b_1 + b_{\delta_1} & &\text{from (163)} &&\text{(171)}\\
a' &= && b_{\delta_t} + t & &\text{from (159)} &&\text{(172)}
\end{aligned}
$$

which concludes the proof for Case 2 a.

We end with Case 2 b. In this case,

$$
\begin{aligned}
t &\geq && b_1 & &&&\text{(173)}\\
0 &\geq && b_1 - t & &&&\text{(174)}\\
\max(0, b_1 - t) &= && 0 & &&&\text{(175)}
\end{aligned}
$$

And from the definition of δ_t:

$$
\begin{aligned}
b_{\delta_t} &= && b_{\delta_1} + b_{\delta_2} + \max(0, b - t) & &&&\text{(176)}\\
&= && b_{\delta_1} + b_{\delta_2} + \max(0, b_1 - t) & &\text{from (146)} &&\text{(177)}\\
&= && b_{\delta_1} + b_{\delta_2} & &\text{from (175)} &&\text{(178)}\\
b_{\delta_t} + t &= && b_{\delta_1} + b_{\delta_2} + t & &&&\text{(179)}
\end{aligned}
$$

Next, from the definition of a

$$
\begin{aligned}
a &= && b_{\delta_1(t)} + t & &&&\text{(180)}\\
&= && \max(0, b_1 - t) + b_{\delta_1} + t & &&&\text{(181)}\\
&= && b_{\delta_1} + t & &\text{from (175)} &&\text{(182)}
\end{aligned}
$$

Now from b

$$
\begin{align}
b_1 + \;\leq\;\; & t \tag{183} \\
b_1 + b_{\delta_1} \;\leq\;\; & t + b_{\delta_1} \tag{184} \\
b_1 + b_{\delta_1} \;\leq\;\; & a && \text{from (182)} \tag{185} \\
b_2 \;\leq\;\; & a && \text{from (151), by transitivity} \tag{186} \\
b_2 - a \;\leq\;\; & 0 \tag{187} \\
\max(0, b_2 - a) \;=\;\; & 0 \tag{188} \\
b_{\delta_2} + \max(0, b_2 - a) \;=\;\; & b_{\delta_2} \tag{189} \\
b_{\delta_2(a)} \;=\;\; & b_{\delta_2} \tag{190} \\
b_{\delta_2(a)} + a \;=\;\; & b_{\delta_2} + a \tag{191} \\
a' \;=\;\; & b_{\delta_2} + a && \text{from the definition of } a' \tag{192} \\
\;=\;\; & b_{\delta_2} + b_{\delta_1} + t && \text{from (182)} \tag{193} \\
\;=\;\; & b_{\delta_t} + t && \text{from (179)} \tag{194}
\end{align}
$$

which concludes the proof for Case 2 b.

The following result states that the representation $(\!|q|\!)_G^c$ is correct:

Proposition 8. *Let* $G = \langle \mathcal{T}_G, \mathcal{F}_G, \mathsf{val}_G \rangle$ *be a temporal graph and* q *a TRPQ. Then the unfolding of* $(\!|q|\!)_G^c$ *is* $[\![q]\!]_G$.

Proof. Let $G = \langle \mathcal{T}_G \mathcal{F}_G, \mathsf{val}_G \rangle$ be a temporal graph, and let q be a TRPQ. To prove the result, it is sufficient to show that:

  (I) for any $\langle n_1, n_2, t, d \rangle \in [\![q]\!]_G$, there are $\tau, \delta \in \mathsf{intv}(\mathcal{T})$ and $b, e \in \mathcal{T}$ such that
      (a) $\langle n_1, n_2, \tau, \delta, b, e \rangle \in (\!|q|\!)_G^c$,
      (b) $t \in \tau$, and
      (c) $d \in \delta(t)$ (where $\delta(t)$ is defined in terms of t, δ, b and e, as explained above).
  (II) for any $\langle n_1, n_2, \tau, \delta, b, e \rangle \in (\!|q|\!)_G^c$ for any $t \in \tau$ and $d \in \delta(t)$,
       $\langle n_1, n_2, t, d \rangle$ is in $[\![q]\!]_G$.

Again, the proof is by induction on the structure of q.
If q is of the form *label*, edge^-, *pred*, $\leq k$, $(\mathsf{trpq} + \mathsf{trpq})$, $\mathsf{trpq}[m, n]$ or $\mathsf{trpq}[m, \_]$, then I and II immediately follow from the definitions of $[\![q]\!]_G$ and $(\!|q|\!)_G^c$.
If q is of the form $\neg\mathsf{node}$ or $(?\mathsf{trpq})$, then the proof is nearly identical to the one already provided for $(\!|q|\!)_G^t$.
And if q is of the form $\mathbf{T}_\delta$ or $\mathsf{trpq}_1/\mathsf{trpq}_2$, then the proof is nearly identical to the one already provided for $(\!|q|\!)_G^{td}$, using Lemma 6 instead of 3.

A.3 Complexity of query answering

We provide in this section complexity results for query answering under the different compact representations studied in the article. The proofs leverage results proven in [2] for non-compact answers.

We start by reproducing the decision problem investigated in [2], which will be intrumental:

> **ANSWER**
> **Input**:   temporal graph G over discrete time, TRPQ q, tuple $\mathbf{u} \in \mathcal{U}$
> **Decide**: $\mathbf{u} \in [\![q]\!]_G$

Our proofs are structured as follows:

- For membership, we leverage the fact that ANSWER is in PSPACE, which was proven in [2]: more precisely, we show in Section A.3 that COMPACTANSWER^t, COMPACTANSWER^d, and COMPACTANSWER^{td} can each be reduced to a finite number of independent calls to an oracle for ANSWER.
- For hardness, the results trivially immediately from the fact that ANSWER is PSPACE-hard, which was also proven in [2]. We show this in Section A.3, with a basic reduction from ANSWER to each of the 4 other problems. We also note that the reduction (from QBF to ANSWER) provided in [2] uses a graph G of fixed size, with the only exception of the temporal domain $\mathcal{T}_G$, and that the graph G' that we use in our reductions only extends the size of G by a constant factor. So this property is preserved for our four problems.

Membership If $G = \langle \mathcal{T}_G \mathcal{F}_G, \mathsf{val}_G \rangle$ is a temporal graph and q a TRPQ, we use $\mathsf{boundaries}(G, q)$ for the set of all interval boundaries that appear in G and q, i.e.

$$\mathsf{boundaries}(G, q) = \bigcup \Big\{ \{b_\delta, e_\delta\} \mid \mathbf{T}_\delta \text{ appears in } q \Big\} \cup \{b_{\mathcal{T}_G}, e_{\mathcal{T}_G}\} \cup$$
$$\bigcup \Big\{ \{b_\tau, e_\tau\} \mid \tau \in \mathsf{val}_G f \text{ for some triple } f \in \mathcal{F}_G \Big\}$$

Note that $\mathsf{boundaries}(G, q)$ is finite.

Next, if $Q \subseteq \mathbb{Q}$, we use Q^{+-} to denote the least superset of Q that is closed under addition and subtraction.
We can now make the two following observations:

Lemma 7. *Let G be a temporal graph, let q be a TRPQ, let $\mathbf{u} = \langle n_1, n_2, t, d \rangle \in [\![q]\!]_G$, let $Q = \mathsf{boundaries}(G, q) \cup \{d\}$, and let τ be the largest interval s.t. $t \in \tau$ and $\langle n_1, n_2, t', d \rangle \in [\![q]\!]_G$ for all $t' \in \tau$. Then*

$$b_\tau \in Q^{+-} \ \text{ and } e_\tau \in Q^{+-}$$

Lemma 8. *Let G be a temporal graph, let q be a TRPQ, let $\mathbf{u} = \langle n_1, n_2, t, d \rangle \in [\![q]\!]_G$, let $Q = \mathsf{boundaries}(G, q) \cup \{t\}$ and let δ be the largest interval s.t. $d \in \delta$ and $\langle n_1, n_2, t, d' \rangle \in [\![q]\!]_G$ for all $d' \in \delta$. Then*

$$b_\delta \in Q^{+-} \ \text{ and } e_\delta \in Q^{+-}$$

Next, let $\sqsubseteq$ denote set inclusion lifted to pairs of intervals, i.e.

$$(\tau_1, \delta_1) \sqsubseteq (\tau_2, \delta_2) \text{ iff } \tau_1 \subseteq \tau_2 \text{ and } \delta_1 \subseteq \delta_2$$

The following is an immediate consequence of Lemmas 7 and 8:

Corollary 1. *Let G be a temporal graph, let q be a TRPQ, let $\mathbf{u} = \langle n_1, n_2, t, d \rangle \in [\![q]\!]_G$, let $Q = \mathsf{boundaries}(G, q) \cup \{t, d\}$, let $P = \{(t, d) \mid \langle n_1, n_2, t, d \rangle \in [\![q]\!]_G\}$, and let $(\tau, \delta) \in \max_{\sqsubseteq}\{(\tau', \delta') \in \mathsf{intv}(\mathcal{T}) \times \mathsf{intv}(\mathcal{T}) \mid t \in \tau \text{ and } d \in \delta\}$. Then*

$$\{b_\tau, e_\tau, b_\delta, e_\delta\} \subseteq Q^{+-}$$

We can now prove our membership results:

Proposition 9. COMPACTANSWER^t *is in PSPACE.*

Proof.
Let G be a temporal graph, let q be a TRPQ, and let $\mathbf{u} = \langle n_1, n_2, \tau, d \rangle \in \mathcal{U}^t$.
We use Q for $\mathsf{boundaries}(G, q) \cup \{d\}$, and T for the set defined by

$$T = \{t \mid \langle n_1, n_2, t, d \rangle \in [\![q]\!]_G\}$$

We also use k to denote the product of the denominators of all numbers in Q, i.e.

$$k = \Pi\{j \mid \frac{i}{j} \in Q \text{ for some i } \in \mathbb{Z}\}$$

Note that k (encoded in binary) can be computed in time polynomial (therefore using space polynomial) in the cumulated sizes of G, q and $\mathbf{u}$.
We also use $\frac{1}{k}\mathbb{Z}$ (resp. $\frac{1}{2k}\mathbb{Z}$) for the set of all multiples of $\frac{1}{k}$ (resp. $\frac{1}{2k}$), i.e

$$\frac{1}{k}\mathbb{Z} = \{\frac{i}{k} \mid i \in \mathbb{Z}\}$$

and

$$\frac{1}{2k}\mathbb{Z} = \{\frac{i}{2k} \mid i \in \mathbb{Z}\}$$

Note that

$$Q^{+-} \subset \frac{1}{k}\mathbb{Z} \subset \frac{1}{2k}\mathbb{Z}$$

Now let τ' be the largest interval such that $\tau \subseteq \tau' \subseteq T$.
Recall that by assumption, $\tau \neq \emptyset$.
Under this assumption, $\langle G, q, u \rangle$ is an instance of COMPACTANSWER^t iff $\tau = \tau'$.
We show that $\tau = \tau'$ can be decided using space polynomial in the cumulated size of (the encodings of) G, q and u.

First, fom Lemma 7, we observe that $b_\tau \notin \frac{1}{k}\mathbb{Z}$ or $e_\tau \notin \frac{1}{k}\mathbb{Z}$ implies $\tau \neq \tau'$.
And $b_\tau \in \frac{1}{k}\mathbb{Z}$ (resp. $e_\tau \in \frac{1}{k}\mathbb{Z}$) can be decided in time polynomial in the encoding of b_τ (resp. e_τ).
So we can focus on the case where $b_\tau \in \frac{1}{k}\mathbb{Z}$ and $e_\tau \in \frac{1}{k}\mathbb{Z}$.

We use b_{inf} for the largest element of $(\frac{1}{2k}\mathbb{Z}) \setminus \tau$ that satisfies $b_{\mathsf{inf}} \leq b_\tau$.
And similarly we use e_{sup} for the smallest element of $(\frac{1}{2k}\mathbb{Z}) \setminus \tau$ that satisfies $e_\tau \leq e_{\mathsf{inf}}$.
Observe that b_{inf} and e_{sup} can be computed using space polynomial in (the encoding of) τ.

We show below that for any (nonempty) interval α with boundaries in $\frac{1}{k}\mathbb{Z}$,

$$\alpha \subseteq T \text{ iff } \alpha \cap \frac{1}{2k}\mathbb{Z} \subseteq T \tag{195}$$

Therefore in order to decide whether $\tau = \tau'$, it is sufficient to decide whether

(I) $\tau \cap \frac{1}{2k}\mathbb{Z} \subseteq T$, and
(II) $\{b_{\mathsf{inf}}, e_{\mathsf{sup}}\} \cap T = \emptyset$

Now observe that:

- I can be decided with a finite number of independent calls to a procedure for ANSWER, and
- II can be decided with two calls to such a procedure.

And it was shown in [2] that ANSWER is in PSPACE.

To complete the proof, we show that (195) holds.
The right direction ($\alpha \subseteq T$ implies $\alpha \cap \frac{1}{2k}\mathbb{Z} \subseteq T$) is trivial.
For the left direction, assume by contradiction that $\alpha \cap \frac{1}{2k}\mathbb{Z} \subseteq T$ but $\alpha \not\subseteq T$.
Take any $t \in \alpha \setminus T$.
Since $\alpha \cap \frac{1}{2k}\mathbb{Z} \subseteq T$ and $t \notin T$, we have

$$t \notin \frac{1}{2k}\mathbb{Z} \tag{196}$$

Next, since α has boundaries in $\frac{1}{k}\mathbb{Z}$,

$$\alpha \cap \frac{1}{2k}\mathbb{Z} \neq \emptyset \tag{197}$$

(for instance, $b_\alpha + \frac{1}{2k} \in \alpha \cap \frac{1}{2k}\mathbb{Z}$).
Together with (196), this implies that there is a t' in $\alpha \cap \frac{1}{2k}\mathbb{Z}$ s.t. either $t' < t$ or $t < t'$.
Let us assume w.l.o.g. that the former holds (the proof for the latter case is symmetric).
And let t_{inf} be the largest value that satisfies $t_{\mathsf{inf}} \in \alpha \cap \frac{1}{2k}\mathbb{Z}$ and $t_{\mathsf{inf}} < t$.
Then

$$t - t_{\mathsf{inf}} < \frac{1}{2k} \tag{198}$$

Now recall that by assumption, $\alpha \cap \frac{1}{2k}\mathbb{Z} \subseteq T$.
Therefore $t_{\mathsf{inf}} \in T$.
So from Lemma 7, there is a β with boundaries in $\frac{1}{k}\mathbb{Z}$ s.t. $\beta \subseteq T$ and $t_{\mathsf{inf}} \in \beta$.
Then we have two cases, either $e_\beta \neq t_{\mathsf{inf}}$ or $e_\beta = t_{\mathsf{inf}}$:

– $(e_\beta \neq t_{\inf})$.
In this case, since $e_\beta \in \frac{1}{k}\mathbb{Z}$, and $t_{\inf} \in \frac{1}{2k}\mathbb{Z}$,

$$\frac{1}{2k} \leq e_\beta - t_{\inf}$$

Together with (198), this yields (by transitivity)

$$t - t_{\inf} < e_\beta - t_{\inf} \tag{199}$$
$$t < e_\beta \tag{200}$$

Now since $t_{\inf} \in \beta$,
$$b_\beta \leq t_{\inf} \tag{201}$$

Together with $t_{\inf} < t$, this implies

$$b_\beta < t \tag{202}$$

Together with (200), this yields

$$t \in \beta$$

Since $\beta \subseteq T$, this implies $t \in T$, which contradicts the definition of t.

– $(e_\beta = t_{\inf})$.
In this case, since β has boundaries in $\frac{1}{k}\mathbb{Z}$,

$$t_{\inf} \in \frac{1}{k}\mathbb{Z} \tag{203}$$

And because $t \in \alpha$ and $t_{\inf} < t$

$$t_{\inf} < t \leq e_\alpha \tag{204}$$
$$t_{\inf} < e_\alpha \tag{205}$$

Together with (203) and $e_\alpha \in \frac{1}{k}\mathbb{Z}$, this implies

$$\frac{1}{k} \leq e_\alpha - t_{\inf} \tag{206}$$

Now let $t_{\sup} = t_{\inf} + \frac{1}{2k}$.
From (206), we get
$$t_{\sup} < e_\alpha \tag{207}$$

Next, since $t_{\inf} \in \alpha$ and $t_{\inf} < t_{\sup}$

$$b_\alpha < t_{\sup} \tag{208}$$

So from (207) and (208)
$$t_{\sup} \in \alpha$$

So from Lemma 7, there is a β' with boundaries in $\frac{1}{k}\mathbb{Z}$ s.t. $\beta' \subseteq T$ and $t_{\mathsf{sup}} \in \beta'$.

Next, from (198) and the definition of t_{sup}

$$t_{\mathsf{sup}} - t < \frac{1}{2k} \tag{209}$$

So with an argument symmetric to the one used above to show $t \in \beta$, we get $t \in \beta'$, which once again contradicts $t \notin T$.

Proposition 10. $\textsc{CompactAnswer}^d$ *is in* $\textsc{PSpace}$

Proof.

The proof is symmetric to the one provided above for Proposition 9, using Lemma 8 instead of Lemma 7.

Proposition 11. $\textsc{CompactAnswer}^{td}$ *is in* $\textsc{PSpace}$.

Proof.

The proof is analogous to the one provided above for Proposition 9, using Corollary 1 instead of Lemma 7.

More precisely, let G be a temporal graph, let q be a TRPQ, and let $\mathbf{u} = \langle n_1, n_2, \tau, \delta \rangle \in \mathcal{U}^{td}$.

We use Q for $\mathsf{boundaries}(G, q) \cup \{t, d\}$, and P for the set defined by

$$P = \{(t, d) \mid \langle n_1, n_2, t, d \rangle \in [\![q]\!]_G\}$$

We also define k, $\frac{1}{k}\mathbb{Z}$ and $\frac{1}{2k}\mathbb{Z}$ identically as in the proof of Proposition 9.

Then analogously to what we showed in this proof, for any pair of intervals (α_1, α_2) with boundaries in $\frac{1}{k}\mathbb{Z}$,

$$\alpha_1 \times \alpha_2 \subseteq P \text{ iff } \left(\alpha_1 \cap \frac{1}{2k}\mathbb{Z}\right) \times \left(\alpha_2 \cap \frac{1}{2k}\mathbb{Z}\right) \subseteq P$$

So with a similar argument, deciding whether $\langle n_1, n_2, \tau, \delta \rangle$ is a compact answer to q over G can be reduced to deciding

- $\{b_\tau, e_\tau, b_\delta, e_\delta\} \subseteq \frac{1}{k}\mathbb{Z}$,
- $(\tau \cap \frac{1}{2k}\mathbb{Z}) \times (\delta \cap \frac{1}{2k}\mathbb{Z}) \subseteq P$,
- $\left(\{b_{\mathsf{inf}}^\tau, e_{\mathsf{sup}}^\tau\} \times (\delta \cap \frac{1}{2k}\mathbb{Z})\right) \cap P = \emptyset$ and
- $\left((\tau \cap \frac{1}{2k}\mathbb{Z}) \times \{b_{\mathsf{inf}}^\delta, e_{\mathsf{sup}}^\delta\}\right) \cap P = \emptyset$

where b_{inf}^τ is the largest element in $(\frac{1}{2k}\mathbb{Z}) \setminus \tau$ that satisfies $b_{\mathsf{inf}}^\tau \leq b_\tau$, e_{sup}^τ is the smallest element in $(\frac{1}{2k}\mathbb{Z}) \setminus \tau$ that satisfies $e_\tau \leq e_{\mathsf{sup}}^\tau$, and b_{inf}^δ and e_{sup}^δ are defined analogously.

Hardness

Proposition 12. $\textsc{CompactAnswer}^t$ *is PSpace-hard*

Proof. The proof is a straightforward reduction from $\textsc{Answer}$.
Let $G = \langle \mathcal{T}_G \mathcal{F}_G, \mathsf{val}_G \rangle$ be a temporal graph, let q be a TRPQ and let $\mathbf{u} = \{n_1, n_2, t, d\} \in \mathcal{U}$.
W.l.o.g., let us assume that $\{n_1, n_2\} \subseteq \mathcal{N}_G$ (the proof for the 3 other cases is symmetric).

Now consider two fresh edge labels $e_1, e_2 \in \mathcal{E}$ that do not appear in $\mathcal{F}_G$. And let $G' = \langle \mathcal{T}_G, \mathcal{F}_{G'}, \mathsf{val}_{G'} \rangle$ be the graph nearly identical to G, but extended with two triples, as follows:

- $\mathsf{val}_{G'}(f) = \mathsf{val}_G(f)$ for each triple $f \in \mathcal{F}_G$,
- $\mathsf{val}_{G'}(n_1, e_1, n_1) = \{[t, t]\}$, and
- $\mathsf{val}_{G'}(n_2, e_2, n_2) = \{[t + d, t + d]\}$.

Let q' be the following TRPQ:

$$q' = e_1/q/e_2$$

Then immediately from the semantics of TRPQs:

$$\mathbf{u} \in [\![q]\!]_G \text{ iff } [\![q']\!]_{G'} = \{\langle n_1, n_2, t, d \rangle\} \tag{210}$$

Now consider the tuple $\mathbf{u}' = \{n_1, n_2, [t, t], d\} \in \mathcal{U}^t$.
Then from (210), $\mathbf{u} \in [\![q]\!]_G$ iff $\mathbf{u}'$ is a compact answer to q over G' in $\mathcal{U}^t$.

Clearly, this input $\langle G', q', \mathbf{u}' \rangle$ for $\textsc{CompactAnswer}^t$ can be computed in time polynomial in the size of (the encodings of) G, q and $\mathbf{u}$.
And it was shown in [2] that $\textsc{Answer}$ is PSpace-complete.

Proposition 13. $\textsc{CompactAnswer}^d$, $\qquad\textsc{CompactAnswer}^{td}$ $\qquad$ *and* $\textsc{CompactAnswer}^c$ *are PSpace-hard.*

Proof. The proofs are nearly identical to the one provided above for $\textsc{CompactAnswer}$.
The graph G' is defined identically in all cases, so that the reductions only differ w.r.t. to the tuple $\mathbf{u}'$.
This tuple is defined as follows:

- $\langle n_1, n_2, t, [d, d] \rangle$ for $\textsc{CompactAnswer}^d$,
- $\langle n_1, n_2, [t, t], [d, d] \rangle$ for $\textsc{CompactAnswer}^{td}$,
- $\langle n_1, n_2, [t, t], [d, d], t, t \rangle$ for $\textsc{CompactAnswer}^c$.

A.4   Size of compact answers

Proposition 1. *In the worst case,*

$$\#\mathsf{answers}^\tau(\mathcal{U}^t) = O(1) \qquad\qquad \#\mathsf{answers}^\delta(\mathcal{U}^t) = \Omega(n)$$
$$\#\mathsf{answers}^\tau(\mathcal{U}^d) = \Omega(n) \qquad\qquad \#\mathsf{answers}^\delta(\mathcal{U}^d) = O(1)$$

Proof.

- $\#\mathsf{answers}^\tau(\mathcal{U}^t) = O(1)$ follows from the definition of $(\!(q)\!)_G^t$ (in Section A.2) and Proposition 5, by induction on the structure of q:
  - If q is of the form *label*, edge^-, *pred*, $\leq k, \mathbf{T}_\delta$ or ?trpq, the number of tuples in $(\!(q)\!)_G^t$ is independent of the size of the intervals present in G.
  - For the binary operators, (i.e. when q is of the form $\mathsf{trpq}_1/\mathsf{trpq}_2$ or $\mathsf{trpq}_1+\mathsf{trpq}_2$), the cardinality of $(\!(q)\!)_G^t$ is bounded by some function of the size of the operands.

    Precisely, if k_1 (resp. k_2) is the cardinality of $(\!(q_1)\!)_G^t$ (resp. $(\!(q_2)\!)_G^t$), then the cardinality of $(\!(q_1/q_2)\!)_G^t$ (resp. $(\!(q_1+q_2)\!)_G^t$) is bounded by $k_1 \cdot k_2$ (resp. $k_1 + k_2$). And by IH, the cardinality of $(\!(q_i)\!)_G^t$ is independent of the size of intervals in G.

    This argument also applies to the case where q is of the form $\mathsf{trpq}[m,n]$, since it is equivalent in this case to a finite union of joins.
  - If q is of the form $\neg q'$, then the cardinality of $(\!(q)\!)_G^t$ is bounded by $2k$, where k is the cardinality of $(\!(q')\!)_G^t$, since the complement in $\mathcal{T}_G$ of an interval can always be represented with at most two intervals. And once again, by IH, k is independent of the size of intervals in G.

- To show that $\#\mathsf{answers}^\delta(\mathcal{U}^t) = \Omega(n)$, for each $i \in \mathbb{N}$, consider the TRPQ $q_i = \mathrm{T}_{[0,i]}$, and the graph G_i with (discrete) effective temporal domain $[0, i]$, a single node n and no edge. Then

$$(\!(q_i)\!)_{G_i}^t = \{\langle n, n, [0,0], d\rangle \mid d \in [0,i]\}$$

  Each tuple in this set has a different value for the distance d, therefore $(\!(q_i)\!)_{G_i}^t$ is the compact representation of $[\![q_i]\!]_{G_i}$ in $\mathcal{U}^t$.
  And this set has cardinality i.

- $\#\mathsf{answers}^\delta(\mathcal{U}^d) = O(1)$ follows from the definition of $(\!(q)\!)_G^d$ (in Section A.2) and Proposition 6, with an argument nearly identical to the one that we provided above to show that $\#\mathsf{answers}^\tau(\mathcal{U}^t) = O(1)$.

- To show that $\#\mathsf{answers}^\tau(\mathcal{U}^d) = \Omega(n)$, we use a fixed query $q = e$, with $e \in \mathcal{E}$. For each $i \in \mathbb{N}$, consider the graph G_i with (discrete) effective temporal domain $[0, i]$, and a single triple $f = (n_1, e, n_2)$ such that $\mathsf{val}_{G_i}(f) = \{[0, i]\}$. Then

$$(\!(q)\!)_{G_i}^d = \{\langle n_1, n_2, t, [0,0]\rangle \mid t \in [0, i]\}$$

  Each tuple in this set has a different value for the distance d, therefore $(\!(q)\!)_{G_i}^d$ is the compact representation of $[\![q]\!]_{G_i}$ in $\mathcal{U}^d$.
  And this set has cardinality i.

Proposition 3. *In the worst case,*

$$\#\mathsf{answers}^\tau(\mathcal{U}^{td}) = O(1) \qquad\qquad \#\mathsf{answers}^\delta(\mathcal{U}^{td}) = \Omega(n)$$

$$\#\mathsf{answers}^\tau(\mathcal{U}^c) = O(1) \qquad\qquad \#\mathsf{answers}^\delta(\mathcal{U}^c) = O(1)$$

Proof.

- To show that $\#\mathsf{answers}^\delta(\mathcal{U}^d) = \Omega(n)$, we use an example analogous to the one used to show non-finiteness (illustrated with Figure 5b).
  For each $i \in \mathbb{N}$, consider the TRPQ $q_i = \mathrm{T}_{[0,i]}$, and the graph G_i with (discrete) effective temporal domain $[0,i]$, a single node n and no edge. Then

  $$[\![q_i]\!]_{G_i} = \{\langle n, n, t, d\rangle \mid t \in [0,i] \text{ and } d \in [0, i-t]\}$$

  For instance, if $i = 2$, then

  $$[\![q_i]\!]_{G_i} = \{\langle n,n,0,0\rangle, \langle n,n,0,1\rangle, \langle n,n,0,2\rangle, \langle n,n,1,1\rangle, \langle n,n,1,2\rangle, \langle n,n,2,2\rangle\}$$

  Now consider again the Euclidean plane $\mathbb{Z} \times \mathbb{Z}$ of time per distance, and the polygon defined by the points $(0,0), (0,n), (n,0)$ and $\{(t, i-t+1) \mid t \in [1,i]\}$. Intuitively, this is a "near triangle" (visually, analogous to an upper approximation of the integral of a linear function).
  The minimal number of tuples in $\mathcal{U}^{td}$ needed to represent $[\![q_i]\!]_{G_i}$ is the minimal number of rectangles needed to cover this polygon, and this number is i.

- To show that $\#\mathsf{answers}^\tau(\mathcal{U}^{td}) = O(1)$ we use an argument similar to the one that we provided to show that $\#\mathsf{answers}^\tau(\mathcal{U}^t) = O(1)$, in the proof of Proposition 1.
  We proceed once again again by induction on the structure of q. If q is an expression for the symbols edge or node in the grammar of Section 3, then the argument is identical to the one provided for $\mathcal{U}^t$. This is is also the case if q is of the form $\mathsf{trpq} + \mathsf{trpq}$ or $\mathsf{trpq}[m,n]$.
  So we focus here on the two remaining operators, namely the cases where q is of the form $\mathbf{T}_\delta$ or $\mathsf{trpq}_1/\mathsf{trpq}_2$.
  Recall that

  $$(\!|\mathsf{trpq}_1/\mathsf{trpq}_2|\!)_G^{td} \bigcup \{\mathbf{u}_1 \bowtie \mathbf{u}_2 \mid \mathbf{u}_1 \in (\!|\mathsf{trpq}_1|\!)_G^{td}, \mathbf{u}_2 \in (\!|\mathsf{trpq}_2|\!)_G^{td}\}$$

  and

  $$(\!|\mathbf{T}_\delta|\!)_G^{td} = \bigcup_{n \in \mathcal{N}_G} \{\langle n, n, \mathcal{T}_G, \delta\rangle \bowtie \langle n, n, \mathcal{T}_G, [0,0]\rangle\}$$

  Let $\mathbf{u}_1 = \langle n_1, n_2, \tau_1, \delta_1\rangle$ and $\mathbf{u}_2 = \langle n_1, n_2, \tau_2, \delta_2\rangle$ be two tuples in $\mathcal{U}^{td}$. For simplicity, we assume that all intervals here are closed-closed, and we focus on the discrete case (but the argument can be easily adapted to the other cases).
  We show that the cardinality of $\mathbf{u}_1 \bowtie \mathbf{u}_2$ is bounded by a function of the

cardinality of δ_1. Together with the definitions of $(\!|\mathsf{trpq}_1/\mathsf{trpq}_2|\!)_G^{td}$ and $(\!|\mathbf{T}_\delta|\!)_G^{td}$, this proves our claim.

We use R_1, R_2, $R_1 \bowtie R_2$, b and e here with the same meaning as in the proof of Lemma 3. We already saw that the cardinality of $\mathbf{u}_1 \overline{\bowtie} \mathbf{u}_2$ is the cardinality of the set $\mathsf{dom}(R_1 \bowtie R_2) \setminus [b, e] + 1$ if $b \leq e$, and the cardinality of $\mathsf{dom}(R_1 \bowtie R_2)$ otherwise.

In the former case, the set $\mathsf{dom}(R_1 \bowtie R_2) \setminus [b, e]$ is the union of the two intervals $[b_{\mathsf{dom}(R_1 \bowtie R_2)}, b]$ and $[e, e_{\mathsf{dom}(R_1 \bowtie R_2)}]$, and the cardinality of each of these is bounded by $e_{\delta_1} - b_{\delta_1}$, which is indeed the cardinality of δ_1.

For the case where $e < b$, $\mathsf{dom}(R_1 \bowtie R_2)$ is also the union of these two intervals (which now overlap), and $e_{\delta_1} - b_{\delta_1}$ is still an upper bound on the cardinality of each of them.

– #$\mathsf{answers}^\tau(\mathcal{U}^{td}) = O(1)$ and #$\mathsf{answers}^\tau(\mathcal{U}^{td}) = O(1)$ both follow from the definition of $(\!|q|\!)_G^c$ (in Section A.2) and Proposition 8, with arguments nearly identical to the one that we provided to show that #$\mathsf{answers}^\tau(\mathcal{U}^t) = O(1)$, in the proof of Proposition 1.

A.5   Compactness

Regarding the cost of coalescing, all our results are already justified in the body of the article.

Regarding (non-)unicity of compact representations, all arguments are also provided, with the exception of non-unicity for the fourth representations (i.e. in $\mathcal{U}^c$). For this case, we show that a set V of tuples in $\mathcal{U}$ that share the same nodes n_1 and n_2 may have several compact representations in $\mathcal{U}^c$.

Over dense time, consider the Euclidean plane $\mathbb{Q} \times \mathbb{Q}$. Observe that:

(i) any rectangle $\tau \times \delta$ in this plane is exactly covered by some tuple in $\mathcal{U}^c$ (namely $\langle n_1, n_2, \tau, \delta, b_\tau, e_\tau \rangle$), and

(ii) the area covered by a tuple in $\mathcal{U}^c$ forms either a rectangle, or a polygon with some non-square angles.

Now assume that the area covered by V forms an L-shaped polygon. From ii, this area cannot be exactly covered by a single tuple in $\mathcal{U}^c$, and from i there is more than one pair of tuples that exactly cover it (as illustrated with Figure 5a).

The same argument can easily be adapted to discrete time.

B   Experiments

We conducted experiments to investigate *(i)* how compact query answers can be in $\mathcal{U}$, $\mathcal{U}^t$ and $\mathcal{U}^d$ and, for the two latter representations, *(ii)* how the size of input intervals in graph and query affect the size of compact answers, and *(iii)* to what extent coalescing $(\!|q|\!)_G^t$ and $(\!|q|\!)_G^d$ affects compactness.

We used the dataset provided in [2], which represents a graph of people and rooms with meetings for contact tracing. The temporal graph G consists

of $24,990$ nodes and $2,638,623$ edges over a domain of 52 time points. The minimum, average, and maximum interval duration in nodes (resp., edges) are 1, 19.7, and 52 (resp., 1, 2.2, and 5). We used SQL to implement the various approaches, and PostgreSQL as a backend. As a query, we retrieve all people that had a positive contact up to a certain time period in the past, with duration x, i.e.,

$$q_4 := (= positive)/\mathbf{T}_{[-x,0]}/\texttt{meets}$$

In our experiments we used two parameters: x, which increases the distance interval $[-x,0]$ in q_4, and a scaling factor k that multiplies the size of all intervals in G. In our first experiment, we compared the size of $[\![q_4]\!]_G$ to its *compact* representations in $\mathcal{U}^t$ and $\mathcal{U}^d$, denoted with $[q_4]_G^t$ and $[q_4]_G^d$ below. The results for varying distances (i.e., values for x) are shown in Figure 7 (left). We see that $[q_5]_G^t$ and $[q_5]_G^d$ provide a compression ratio of about 2.2 for this dataset, which has a very small temporal domain (the curve becomes flat for $x \geq 52$ because the interval $[-x,0]$ is longer than the whole temporal domain).
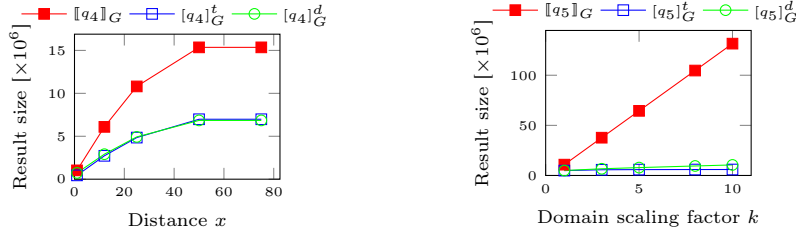

Fig. 7: Result sizes for varying x and fixed $k = 1$ (left), or varying k and fixed $x = 25$ (right)

In the next experiment, we use k to scale up intervals in G, while fixing $x = 25$. Figure 7 (right) shows the results. With more time points in the graph, the differences between $[q_5]_G^t$ and the compact representations increases dramatically, with a compression factor of 22 for $[q_5]_G^t$ and 12.6 for $[q_5]_G^d$ when $k = 10$. From both plots, we see that the difference between the sizes of $[q_5]_G^t$ and $[q_5]_G^d$ is relatively small. However, for smaller values of x, $[q_5]_G^t$ is more compact, while $[q_5]_G^d$ is more compact for a larger x. To illustrate this behavior, we plot the ratio $|[q_5]_G^d|/|[q_5]_G^t|$, in Figure 8 (left), for a fixed $k = 5$ and varying x, and in Figure 8 (right) for a fixed $x = 25$ and varying k. We also performed experiments to show the impact of coalescing on compactness. Figure 9 shows result sizes with and without coalescing, for a fixed $k = 5$ and different values for x. We see that $(\!|q_5|\!)_G^t$ and $[q_5]_G^t$ have comparable sizes, whereas $(\!|q_5|\!)_G^d$ achieves a very low compression ratio, being similar in size to $[\![q_5]\!]_G$.

In summary, we can see from the experiments that both $\mathcal{U}^t$ and $\mathcal{U}^d$ provide a much more compact representation than $\mathcal{U}$. Indeed, $\mathcal{U}$ suffers from large intervals, either in the input graph or in the input query When intervals in the query are smaller than in the graph, $\mathcal{U}^t$ is the most compact representation, while $\mathcal{U}^d$ is the most compact in the opposite case. Concluding, coalesced representations
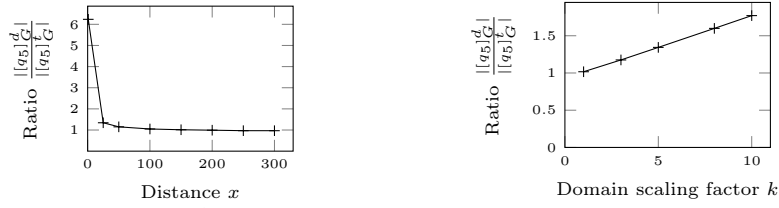
Fig. 8: Ratio $|[q_5]_G^d|$ over $|[q_5]_G^t|$ for varying x and fixed $k = 5$ (left), or varying k and fixed $x = 25$ (right)

are substantially more compact as compared to their uncoalesced counterparts, which one one hand provides advantages for data storage, but may also provide advantages in query processing, since less data has to be processed.

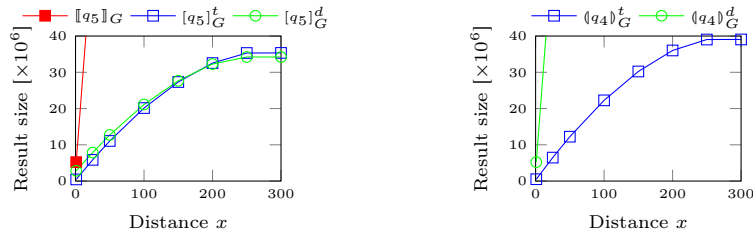Guidelines to reproduce our experiments are available at https://anonymous.4open.science/r/CompactTRPQ-B703/experiments.

Fig. 9: Result sizes for varying x and fixed $k = 5$, with coalescing (left) and without (right)