A PROOF

A.1 Proof of Proposition 1

Referring to Definition 1, a pattern p_0 is declared over a set of events, say $\mathcal{E}_0$. It applies to the (at least two) sub-patterns $p_1, \ldots, p_k$ embedded in SEQ or AND of p_0, denoted by $\mathcal{E}_1, \ldots, \mathcal{E}_k$. Referring to the sequence/concurrent relationships among $p_1, \ldots, p_k$, we have $\mathcal{E}_i \cap \mathcal{E}_j = \emptyset, i, j \in 1, \ldots, k$, i.e., the sub-patterns $p_1, \ldots, p_k$ do not overlap on events. It follows that the number of sub-patterns embedded in p_0 is not greater than $|\mathcal{E}_0|$. As presented in Definition 2, the [ATLEAST a] [WITHIN b] conditions in each pattern can be checked in $O(k)$ time. The time cost of determination of whether $t \vDash p_0$ is thus $O(|\mathcal{E}_0|^2)$, where $|\mathcal{E}_0|$ is the number of events declared in p_0.

A.2 Proof of Theorem 2

The problem is clearly in NP. For any trace t, whether it matches each pattern can be recursively checked, referring to Proposition 1.

To show the hardness of the pattern consistency problem, we present a reduction from the 3SAT problem, which is one of Karp's 21 NP-complete problems [24]. Consider a CNF formula $C = c_1 \wedge \cdots \wedge c_m$, where each clause $c_i = l_{i1} \vee l_{i2} \vee l_{i3}$ have three literals, $i = 1, \ldots, m$, and each literal l_{ij} is either x or $\neg x$ for some variable $x \in X, n = |X|$. The 3SAT problem is to determine whether there exists an assignment to make all the clauses satisfied.

The transformation first introduces a set of events

$$\mathcal{E} = \{C_0, C_1, \ldots, C_m, X_1, \ldots, X_n, \neg X_1, \ldots, \neg X_n\},$$

where each C_i corresponds to a clause $c_i, i = 1, \ldots, m$, and $X_j, \neg X_j$ represent $x_j, \neg x_j$, respectively, $j = 1, \ldots, n$. A set $\mathcal{P}$ of event patterns over $\mathcal{E}$ are constructed as follows:

(1) For each $x_j, \neg x_j$, we introduce an event pattern

$$p_j : \texttt{SEQ}(C_0, \texttt{AND}(X_j, \neg X_j)\ \texttt{ATLEAST 1 WITHIN 1})\ \texttt{ATLEAST 3 WITHIN 3}.$$

(2) For each clause $c_i = l_{i1} \vee l_{i2} \vee l_{i3}$, we add an event pattern

$$p_{n+i} : \texttt{SEQ}\big(C_i, \texttt{AND}(X_{i1}, X_{i2}, X_{i3})\big)\ \texttt{ATLEAST 2 WITHIN 2}.$$

(3) For each C_i, we consider

$$\texttt{SEQ}(C_0, C_i)\ \texttt{WITHIN 1}.$$

Event pattern p_j requires that the timestamp distance between X_j and $\neg X_j$ is exactly 1, while p_{n+i} requires that at least one of X_{i1}, X_{i2}, X_{i3} should have timestamp distance 2 compared to C_i. (Figure 14 illustrates the interval conditions in the complex temporal networks corresponding to the aforesaid patterns, see Definition 5 for details.)

We show that the CNF formula in transformation is satisfiable if and only if there exists a trace t of events such that $t[C_0] = s, t[C_i] = s + 1, t[X_{ik}] = s + 3$ for some $X_{ik}, k = 1, 2, 3, i = 1, \ldots, n$, s is any timestamp in the domain, i.e., $t \vDash \mathcal{P}$ the event patterns are consistent.

First, if the CNF formula in transformation is satisfiable, we set the corresponding timestamp assignment $t[C_i] = s + 1$ and $t[C_0] = s$, where s is any timestamp in the domain. For each clause $c_i = l_{i1} \vee l_{i2} \vee l_{i3}$, there must exist some l_{ik} = true, $k = 1, 2, 3$, while others equal 0. For each l_{ik} = true, we assign $t[X_{ik}] = s + 3$; otherwise,

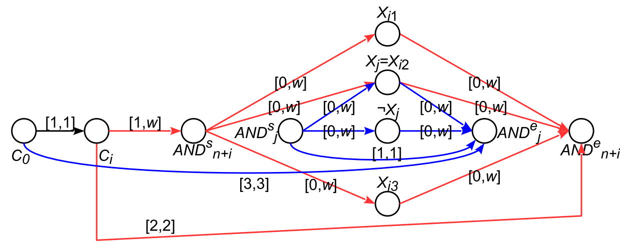

Figure 14: Reduction from 3SAT problem

$t[X_{ik}] = s + 2$ for l_{ik} = false. It is easy to see $t \vDash p_{n+i}$ in $\mathcal{P}$ in the transformation. Moreover, for each X_j = true, we have $t[X_j] = s+3$ and $t[\neg X_j] = s + 2$; otherwise, $t[X_j] = s + 2$ and $t[\neg X_j] = s + 3$ for X_j = false. The CNF assignment guarantees that $t \vDash p_j$ in $\mathcal{P}$ in the transformation. That is, we have $t \vDash \mathcal{P}$.

Conversely, suppose that the event patterns are consistent, i.e., there exists a timestamp assignment t having $t \vDash \mathcal{P}$. Let $t[C_0] = s$ the timestamp in the domain. It follows $t[C_i] = s + 1, i = 1, \ldots, m$. Pattern p_j in $\mathcal{P}$ in the transformation ensures that only one of $X_j, \neg X_j$ will have π label $s + 3$, i.e., true in the CNF assignment; the other will have t label $s + 2$ and false assignment in the CNF. Referring to $t \vDash p_{n+i}$ in $\mathcal{P}$ in the transformation, for each $i = 1, \ldots, n$, there must exists some $X_{ik}, k = 1, 2, 3$ such that $t[X_{ik}] = s + 3$. According to the aforesaid assignment, we have the corresponding literal l_{ik} = true. For others with $t[X_{ik}] = s + 2$, it has l_{ik} = false. That is, the CNF formula is satisfiable.

A.3 Proof of Theorem 3

The problem is clearly in NP. Given a trace t' with modified timestamp, whether it matches each pattern can be recursively checked, referring to Definition 2, and the modification cost can be calculated in $O(n)$ time by Formula 1.

To illustrate the hardness of the timestamp modification problem, we show a reduction from the SET COVER problem, which is one of Karp's 21 NP-complete problems [24].

Given a set of m elements $\mathcal{U} = \{u_1, \ldots, u_m\}$ and n sets $\mathcal{S} = \{s_1, \ldots, s_n\}$ such that $s_i \subseteq \mathcal{U}$ and $\cup_i s_i = \mathcal{U}$. A set cover is a $\mathcal{C} \subseteq \mathcal{S}$ of sets whose union is still $\mathcal{U}$. The minimum set cover problem is to identify the smallest number of sets whose union still contains all elements in $\mathcal{U}$.

Consider a set of events

$$\mathcal{E} = \{S_1, \ldots, S_i, \ldots, S_n, S'_1, \ldots, S'_n, U_1, \ldots, U_j, \ldots, U_m\}$$

where each S_i, S'_i represents a set s_i, and U_j corresponds to an element u_j. Let $\mathcal{E}_j = \{S_i \in \mathcal{E} \mid u_j \in s_i\}$, i.e., all the events S_i whose corresponding s_i contains u_j.

First, the event pattern set $\mathcal{P}$ is constructed as follows.

(1) For each U_j, we add into $\mathcal{P}$ a pattern

$$p_j : \texttt{SEQ}(U_j, \texttt{AND}(S_{i1}, \ldots, S_{ik}))\ \texttt{ATLEAST 2 WITHIN 2},$$

where $S_{i1}, \ldots, S_{ik}$ correspond to all the sets $s_{i1}, \ldots, s_{ik} \subseteq \mathcal{S}$ containing element u_j.

(2) For each U_j, S'_i, we introduce a pattern

$$\texttt{SEQ}(S'_i, U_j)\ \texttt{WITHIN 1}.$$

(Figure 15 illustrates the interval conditions in the complex temporal networks corresponding to the aforesaid patterns, see Definition 5 for details.)

Next, we build a trace t of events where $t[S'_i] = 0, t[U_j] = 1, t[S_i] = 2, i = 1, \dots, n, j = 1, \dots, m$. The patterns require that there must exist a $S_i \in \mathcal{E}_j$ such that $t[S_i] - t[U_j] = 2$, which is not observed in the given t.

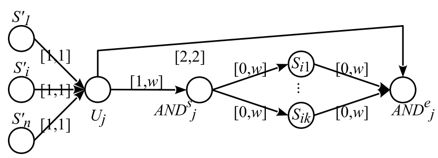

Figure 15: Reduction from SET COVER problem

To prove NP-hardness we show that there is a set cover C of size k if and only if there exists a trace t' matches the event patterns in the transformation and $\Delta(t, t') = k$.

First, if there exists a set cover C with size k, for each $s_i \in C$, we assign the corresponding $t'[S_i] = 3$. Referring to the repair cost in Formula 1, we have $\delta(t', t) = k$. Moreover, the set cover requires that for each u_j, there is at least one set s_i in C that contains u_j, i.e., having the corresponding $t'[S_i] = 3$. It leads to $t' \vDash \mathcal{P}$.

Suppose that there is a trace t' with $t' \vDash \mathcal{P}$ and $\Delta(t, t') = k$. Referring to the transformation of each pattern p_j in $\mathcal{P}$, there must have some $S_i \in \mathcal{E}_j$ with repaired $t'[S_i] = 3$. By collecting all the corresponding s_i, since all the elements u_j are covered, it forms a set cover C with size k.

A.4 Proof of Theorem 4

To show the hardness of approximation, we illustrate that the reduction in Theorem 3 from the SET COVER problem, which is known NP-hard to approximate to within any constant factor [10], is gap-preserving.

Let C^* denote a minimum set cover, and t^* be a modified trace with the minimum modification cost. It is sufficient to conclude

- if $|C^*| \le k$, then $\Delta(t, t^*) \le k$,
- if $|C^*| > \alpha k$, then $\Delta(t, t^*) > \alpha k$,

where $\alpha > 1$.

A.5 Proof of Proposition 5

The conclusion can be proved by induction referring to the recursive Definitions 2 and 5.

(1) For $p = E$, we have $t[p^s] = t[p^e] = t[E]$ for any t in Definition 2, and $\Phi_p = \emptyset, \Gamma_p = \emptyset$, referring to item 1 in Definition 5.

(2) For $p = \texttt{SEQ}(p_1, p_2, \dots, p_k)$ [ATLEAST a] [WITHIN b], referring to item 2 in Definition 5, it has $\Gamma_p \setminus (\Gamma_{p_1} \cup \dots \cup \Gamma_{p_k}) = \emptyset$. First, suppose that $t \vDash p$. Referring to $t[p_i^e] < t[p_{i+1}^s]$ in item 2 in Definition 2, we have $t \vDash \phi(E^e_{p_i}, E^s_{p_{i+1}})$:[1, w], $i = 1, \dots, k-1$, in item 2 in Definition 5. Moreover, the ATLEAST a WITHIN b condition is equivalent to the interval condition $\phi(E^s_p, E^e_p)$:[a, b]. It leads to $t \vDash (\Phi_p, \Gamma_p)$. Conversely, we show $t \vDash (\Phi_p, \Gamma_p)$ implies $t \vDash p$ in a similar way.

(3) For $p = \texttt{AND}(p_1, p_2, \dots, p_k)$ [ATLEAST a] [WITHIN b], the binding condition $\gamma(E^s_p, \{E^s_{p_1}, \dots, E^s_{p_k}\})$:*min* and the interval conditions $\phi(E^s_p, E^s_{p_i})$:[0, w], $i = 1, \dots, k$ in item 3 in Definition 5 ensure $t[p^s] = \min(\pi(p^s_1), \pi(p^s_2), \dots, \pi(p^s_k))$ in item 3 in Definition 2. Similar equivalence on max binding condition can also be observed. Again, the ATLEAST a WITHIN b condition is equivalently captured by the interval condition $\phi(E^s_p, E^e_p)$:[a, b].

A.6 Proof of Corollary 6

Similar to the proof of Proposition 5, the conclusion can be proved by induction referring to the recursive Definitions 2 and 6. In particular, since there is no binding condition, the proof w.r.t. binding conditions in step (3) in the proof of Proposition 5 is not necessary.

A.7 Proof of Proposition 7

The binding conditions Γ appear only w.r.t. AND event patterns, together with a list of interval conditions, referring to item 3 in Definition 5.

First, suppose that there exists a $\Phi_k \in \aleph_\Gamma$ such that $\Phi \cup \Phi_k$ is consistent, i.e., having some $t \vDash \Phi \cup \Phi_k$. For each binding condition (say $\gamma(E_i, \mathcal{E}_i)$:*min*) in Γ, we have some $E_j \in \mathcal{E}_i$ with $t[E_i] = t[E_j]$, according to the full binding in Definition 7. Since $t \vDash \Phi$, we have $t[E_i] = t[E_j] \le t[E_k], \forall E_k \in \mathcal{E}_i$, referring to the aforesaid interval conditions in item 3 in Definition 5. That is, it has $t \vDash \Gamma$ as well.

Next, suppose that (Φ, Γ) is consistent, having some $t \vDash (\Phi, \Gamma)$. For each binding condition (say $\gamma(E_i, \mathcal{E}_i)$:*min*) in Γ, we have some $E_j \in \mathcal{E}_i$ with $t[E_i] = t[E_j]$, according to the binding condition Definition 4. That is, it satisfies $t \vDash \phi(E_i, E_j)$:[0, 0] in Φ_γ. Considering the aforesaid interval condition for each binding condition γ in Γ, we obtain a $\Phi_k \in \aleph_\Gamma$ having $t \vDash \Phi_k$.

A.8 Proof of Proposition 8

The conclusion is clear by considering all the pairwise shortest paths, over the distance graph with $2 * (n-1)$ edges in two paths, respectively.

A.9 Proof of Proposition 9

Firstly, we prove $\Phi_c \vDash \Phi$. For any $t \vDash \Phi_c$, according to Formula 2, we have $t[E_j] - t[E_i] = t[E_j] - t[E_{j-1}] + t[E_{j-1}] - \dots - t[E_i] \ge -d_{i+1,i} - d_{i+2,i+1} - \dots - d_{j,j-1} = -d_{ji} \ge a$ for any $\phi(E_i, E_j)$:[a, b] $\in \Phi, i < j$. Similar derivation applies to $t[E_j] - t[E_i] \le d_{ij} \le b$. That is, $t \vDash \Phi_c$ implies $t \vDash \Phi$.

Next, we prove $\Phi \vDash \Phi_c$. For any $t \vDash \Phi$, referring to the shortest path, we have $-d_{i+1,i} \le t[E_{i+1}] - t[E_i] \le d_{i,i+1}, i = 1, \dots, n-1$. Given the definition $\phi(E_i, E_{i+1})$:[$-d_{i+1,i}, d_{i,i+1}$] in Φ_c, it follows $t \vDash \Phi_c$.

A.10 Proof of Proposition 10

To capture the bound of optimal timestamp modification, we first show that a trace timestamp explanation with optimal timestamp modification t^* always exists, such that for each E_i, it either has $t^*(E_i) = t[E_i]$ or $\exists E_j \in \mathcal{E}$ that $t^*(E_j) = t[E_j], t^*(E_i) - t^*(E_j) = d_{ji}$ or $-d_{ij}$.

For each E_i, let $\mathcal{E}_i = \{E_i\} \cup \{E_j \mid t^*(E_i) - t^*(E_j) = d_{ji} \text{ or } -d_{ij}\}$. According to Definition 8 of chain temporal networks, we have $\forall E_k \in \mathcal{E} \setminus \mathcal{E}_i, E_j \in \mathcal{E}_i, t^*(E_k) - t^*(E_j) \ne d_{jk}, t^*(E_k) - t^*(E_j) \ne -d_{kj}$.

If there exists some $E_j \in \mathcal{E}_i$ such that $t^*(E_j) = t[E_j]$, the conclusion is proved. Otherwise, i.e., $t^*(E_j) \neq t[E_j], \forall E_j \in \mathcal{E}_i$, we can generate a new trace t' by slightly modifying a same small amount c for all the timestamp of eventes $E_j \in \mathcal{E}$. There are two cases:

(1) If $|\{E_j \in \mathcal{E}_i \mid t^*(E_j) > t[E_j]\}| \neq |\{E_j \in \mathcal{E}_i \mid t^*(E_j) < t[E_j]\}|$, the total modification cost of t' reduces, which is a contradiction to the optimal t^*.

(2) If $|\{E_j \in \mathcal{E}_i \mid t^*(E_j) > t[E_j]\}| = |\{E_j \in \mathcal{E}_i \mid t^*(E_j) < t[E_j]\}|$, where the total modification cost of t' is the same to t^*, there are two sub-cases:

(2a) We have $t'[E_j] = t[E_j]$ for some $E_j \in \mathcal{E}_i$. The conclusion is prove.

(2b) We have $t'[E_k] - t'[E_j] = d_{jk}$ or $-d_{kj}$ for some $E_k \in \mathcal{E} \setminus \mathcal{E}_i, E_j \in \mathcal{E}_i$. E_k is then added into $\mathcal{E}_i$ w.r.t. t' which has the same modification cost to t^*, i.e., optimal. We iteratively apply the aforesaid cases (1) and (2) for t'.

Finally, if all the events are added into $\mathcal{E}_i$, it belongs to either case (1) or (2a).

Referring to the aforesaid proved property of optimal solution, for any $E_i \in \mathcal{E}$, we have an optimal timestamp modification with $t^*(E_i)$ no greater than $t[E_j]+d_{ji}$ for all $E_j \in \mathcal{E}$. Similar lower bound applies to $t[E_j] - d_{ij}$. The conclusion is proved.

A.11 Proof of Proposition 11

For the events $\mathcal{E} = \{E_1, \ldots, E_n\}$ in the pattern, we denote $E_{\max} = \arg\max_{E\in\mathcal{E}} t[E]$ and $E_{\min} = \arg\min_{E\in\mathcal{E}} t[E]$. Referring to the single binding in Definition 9, we have

$$\Phi_1 = \{\phi(\mathsf{AND}^s, E_{\min}){:}[0,0], \phi(\mathsf{AND}^e, E_{\max}){:}[0,0]\}.$$

Let t^* be the optimal timestamp modification w.r.t. full binding, having $E^*_{\max} = \arg\max_{E\in\mathcal{E}} t^*(E)$, and $E^*_{\min} = \arg\min_{E\in\mathcal{E}} t^*(E)$. We show that t^* is also a timestamp modification w.r.t. the single binding, in all the possible cases.

(1) If $t[E_{\min}] \leq t^*[E^*_{\min}]$, it has $t^*[E] = t^*[E^*_{\min}]$ for all $E \in \mathcal{E} \setminus \{E_{\max}\}$ with $t[E] \leq t^*[E^*_{\min}]$. Any other possible timestamp modification on E greater than $t^*[E^*_{\min}]$ must have modification cost no less than π^*. It indicates $t^*[E_{\min}] = t^*[E^*_{\min}]$.

(2) If $t[E_{\min}] > t^*[E^*_{\min}]$, it has $t^*[E_{\min}] = t^*[E^*_{\min}]$. Any other possible modification with a different event having the minimum timestamp $t^*[E^*_{\min}]$ must have modification cost no less than t^*. It also indicates $t^*[E_{\min}] = t^*[E^*_{\min}]$.

(3) If $t[E_{\max}] \geq t^*[E^*_{\max}]$, it has $t^*[E] = t^*[E^*_{\max}]$ for all $E \in \mathcal{E} \setminus \{E_{\min}\}$ with $t[E] \geq t^*[E^*_{\max}]$. Any other possible timestamp modification on E less than $t^*[E^*_{\max}]$ must have modification cost no less than t^*. It indicates $t^*[E_{\max}] = t^*[E^*_{\max}]$.

(4) If $t[E_{\max}] < t^*[E^*_{\max}]$, it has $t^*[E_{\max}] = t^*[E^*_{\max}]$. Any other possible modification with a different event having the maximum timestamp $t^*[E^*_{\max}]$ must have modification cost no less than t^*. It also indicates $t^*[E_{\max}] = t^*[E^*_{\max}]$.

Since we have $t^*[E_{\max}] = t^*[E^*_{\max}]$ and $t^*[E_{\min}] = t^*[E^*_{\min}]$ in all the cases, it is sufficient to show that t^* is also a timestamp modification w.r.t. the single binding Φ_1, and could be returned by Algorithm 2 with single binding.