Temporal graphs, from snapshots to optimal paths

Michelle Döring Hendrik Molter

1 One graph, three representations

Definition 1. Fix a finite vertex set and a lifetime T. A temporal graph is a sequence of loopless snapshots indexed by $0, \dots, T-1$. In the directed setting each snapshot records arcs. In the undirected setting the two orientations represent the same physical edge. The *footprint* contains the edges that appear at least once. Give each such edge the set of times at which it appears. Alternatively, list the arc occurrences (t, u, v) in chronological order, with ties in any order. All three descriptions retain the vertex set and lifetime, including isolated vertices and empty final snapshots.

Lemma 2. *Converting snapshots to a labelled footprint and back recovers the original temporal graph.*

Proof. Fix a time t and vertices u, v. Reconstructing the snapshot puts the arc (u, v) at t exactly when its label set contains t. That is precisely the condition for including t when the label set was formed. Thus every snapshot is recovered, and the lifetime was copied by both conversions. □

Definition 3. A labelling is *simple* if each footprint edge has exactly one time, *proper* if arcs (u, v) and (v, w) with $w \neq u$ always have distinct labels, and *happy* if both hold. Opposite directed arcs may appear simultaneously. In symmetric snapshots the condition says that each snapshot is a matching.

Definition 4. A temporal walk follows edges at their appearance times. The departure times increase strictly in a *strict* walk and weakly in a *nonstrict* one. A path is a walk whose vertices are pairwise distinct. A vertex reaches another if a path joins them in the chosen mode. A path with no edges reaches its own starting vertex.

Definition 5. The *reachability graph* in a chosen mode is the static digraph on the same vertices with an arc $u \to v$ exactly when $u \neq v$ and a temporal path in that mode joins u to v. The zero-edge path gives reflexive reachability, but the digraph omits loops.

2 When do strict and nonstrict paths agree?

Lemma 6. *On a proper temporal graph, every nonstrict path already has strictly increasing departure times.*

Proof. Consecutive edges share their middle vertex. Their outer vertices are distinct because a path never repeats a vertex. Properness therefore rules out equal departure times on consecutive edges. Weak increase at every step becomes strict increase. □

Theorem 7. *On a proper directed or undirected temporal graph, u reaches v strictly if and only if u reaches v nonstrictly.*

Proof. A strict path satisfies the nonstrict rule. In the other direction, the preceding lemma shows that the very same nonstrict path, with its original departure times, is a strict witness. □

Example 8. The corresponding assertion about every nonstrict *walk* fails even for a single undirected edge.

Proof. Let the edge appear at times 0 and 1. The graph is proper because it has only one physical edge. At time 0 a walk can cross that edge and return immediately. The two departures have the same time. □

Definition 9. Give each edge occurrence a nonnegative integer traversal time. A departure at a with traversal time c arrives at $a+c$. A subsequent step may depart upon arrival or after waiting. The lifetime bounds departures, and the final arrival may follow the last snapshot.

Lemma 10. *For fixed vertices and departure times, uniform traversal time zero gives exactly the nonstrict paths, and uniform time one gives exactly the strict paths.*

Proof. Availability and vertex distinctness are the same in both descriptions. For departures a, b in traversal order, feasibility means $a + c \leq b$. For $c = 0$ this says $a \leq b$. For $c = 1$ it says $a < b$, since the times are integers. □

3 What makes a path optimal?

Definition 11. Fix endpoints, a release time and an arrival deadline. Among feasible paths, a *shortest* path minimizes hops, a *fastest* path minimizes elapsed time, and a *foremost* path minimizes arrival time. A *latest-departure* path maximizes its first departure time. A *minimum-waiting* path minimizes the sum of waits between successive traversals. Waiting before the first departure contributes nothing.

Theorem 12. *If each traversal takes c units, a path with k hops and total internal waiting time w has elapsed time $w + kc$.*

Proof. Partition the interval from first departure to final arrival at the successive arrivals and departures. The traversal intervals contribute kc in total. The remaining gaps are exactly the internal waits, of total length w. On a zero-hop path departure and arrival coincide and both terms vanish. □

Lemma 13. *At uniform traversal time zero, a feasible path is fastest exactly when it has minimum total waiting time.*

Proof. Set $c = 0$ in the preceding identity. Every feasible path then has the same value under both objectives, so the two sets of minimizers agree. □

Example 14. Take a directed triangle with arcs $0 \to 1$, $1 \to 2$, $0 \to 2$, each available at times 0 through 3. Let each traversal take $c \in \{0, 1\}$ units, with release time 0 and deadline $3 + c$. The two-hop detour departing at $0, c$ minimizes waiting, uses more hops than the direct arc, and is fastest exactly when $c = 0$.

Proof. The second departure coincides with the first arrival, so the detour waits zero time. Its two traversals take $2c$ units in total. At $c = 0$ this is the smallest possible elapsed time. At $c = 1$ the direct arc needs only one unit and arrives sooner when both paths start at time 0. □