Chronological Networks in Archaeology: a Formalised Scheme

Eythan Levy*[1], Gilles Geeraerts[2], Frédéric Pluquet[3], Eli Piasetzky[4], and Alexander Fantalkin[1]

[1]Tel-Aviv University, Department of Archaeology and Ancient Near Eastern Cultures, Tel-Aviv, Israel, eythan.levy@gmail.com
[2]Université libre de Bruxelles, Computer Science Department, Brussels, Belgium, gigeerae@ulb.ac.be
[3]Haute École Louvain en Hainaut, Tournai, Belgium, pluquetf@helha.be
[4]Tel-Aviv University, School of Physics and Astronomy, Tel-Aviv, Israel, eip@tauphy.tau.ac.il

Abstract

This paper proposes a new methodology for modelling chronological data in archaeology. We introduce the concept of "chronological network", a flexible model for representing chronological entities, synchronisms between them, and other chronological constraints such as *termini post/ante quem* and duration bounds. We propose a procedure for checking the consistency of a chronological network and for refining dating estimates from the available synchronisms and constraints. We introduce CHRONOLOG, a chronology software application that allows users to build a chronological network interactively. The software automatically checks the consistency of the network and computes the tightest possible chronological range for each entity, within seconds. CHRONOLOG is freely available online at http://chrono.ulb.be[1].

Keywords: Chronology, Archaeology, Archaeological dating, Computational archaeology, Algorithms, Formal methods, Quantitative methods.

1 Introduction

Our understanding of the ancient past often takes the shape of a network. Synchronisms between kings, historical eras, archaeological strata and ceramic types induce a complex web of interconnected chronological objects. An important aspect of such a web is the strong dependency among its components: a change at one end of the network can directly impact dates anywhere along the network. Changing a king's regnal dates, for example, can directly affect the dating of an archaeological stratum containing objects bearing that king's name. This can in turn affect the dating of ceramic types found in that stratum, and so on. Although chronological networks are often informally described in archaeological literature, they are often not explicitly recognised as such and, as a result, have never been fully formalised. This paper presents a formalised framework of chronological networks in archaeology. We first describe a conceptual model of chronological networks, featuring chronological sequences, upper/lower bounds on dates and durations, and several types of synchronisms (Section 2). This leads to a detailed mathematical model of chronological networks (Section 3). Based on this mathematical formalism, we introduce efficient algorithms for solving

*Corresponding author.

[1]The CHRONOLOG website is password-protected for the duration of the reviewing process. The password is "JAS" (without the quotes).

basic chronological problems. Two particular problems – *consistency checking* (i.e., verifying that the network features no contradictions) and *tightening* (i.e. computing the tightest possible chronological ranges for each date and duration) are of paramount importance. Readers not concerned with the details of mathematical modelling can skip Section 3, and move on to Section 4, which describes CHRONOLOG – software that facilitates construction of chronological networks, checks their consistency and provides tightened estimates of each boundary and duration, quickly and interactively. We illustrate the use of CHRONOLOG with a case study related to the Egyptian 26th dynasty (Section 5). Finally, Section 6 discusses future perspectives for both the model and the software implementation.

Related works

The question of representing and manipulating information about time has been long studied in the field of artificial intelligence, see for example the seminal works of Allen [All84, All91]. While some of the techniques we develop in Section 3 are related to these works (like the graph-based representation of the chronological constraints), the latter are very general and do not focus on needs related to archaeological data. Moreover, these earlier works are mainly concerned with the *representation* of the data, while we also present algorithms and software that directly address archaeological problems.

Allen's early work ([All84]) characterised 13 basic relations among temporal intervals. These relations were originally defined in the framework of temporal logic, but were later apllied to archaeology by Holst [Hol04]. The characterisation of chronological relations presented in this paper (Section 2.1) expands on Allen and Holst.

An interesting related work is that of Kromholz [Kro87], who proposed in 1987 to use off-the-shelf business-oriented computer programs to formalise archaeological chronology problems. These programs use typical models from the business world (PERT and Gantt charts) and rely on classical algorithmic methods (the co-called "Critical Path Method") to analyse them and test different chronological hypotheses. Kromholz rightfully asked "how to deal with the immense quantity of data offered by every spadeful of earth we disturb" ([Kro87], p. 119) and we fully concur with his pioneering approach. His model differs from ours in several ways. To begin with, the data models are different. Ours allows us to model more diverse types of chronological constraints (see Section 2.1). Furthermore, the two approaches do not address exactly the same questions and rely on totally different algorithmic techniques. Finally, the technique proposed by Kromholz uses commercially-produced business-oriented software not originally intended for archaeology, which requires the user to shuttle between the terminologies of two widely different disciplines. The solution proposed in this paper is aimed at archaeologists' needs, with a data model consisting of more archaeologically-meaningful basic elements.

Our work can also be compared to more traditional formal approaches for stratigraphic analysis, such as those of the frequently-used Harris matrix ([Har79]) or the partial order scalogram analysis of relations by Ilan Sharon ([Sha95]). These approaches, however, deal only with relative chronology, while our approach considers both relative and absolute chronology aspects in a unified model. As such it comes close to the approach of Bruno Desachy ([Des16]), who augments the traditional Harris matrix approach by adding to it, as in our model (see Section 2 below), upper and lower bounds on the start date, end date, and duration of each stratigraphic unit. Our approach features an additional set of possible synchronisms, a more powerful algorithmic tool for detecting inconsistencies and new algorithms for computing tight time and duration ranges (see Section 3).

The work closest to ours is that of David Falk, who implemented a chronological tool called Groundhog ([Fal20], see `http://www.lagomorph-rampant.com/chronology/index.html`), which allows building of chronological networks and testing them for internal contradictions. His approach differs from ours in several aspects. First, our model allows for more diverse types of chronological constraints (see Section 2).

Second, Falk's approach relies on exhaustive search, by generating all possible combinations of dates, thus yielding exponential-time algorithms, whereas we employ a more efficient approach, using polynomial-time algorithms (see Section 3); this means that Falk's approach is unlikely to be able to handle networks of large sizes in short processing time. Our technique can scale and handle networks with several hundred chronological constraints in less than a second, allowing for a truly interactive experience for the user (see Section 4.3.2).

Other formal approaches to archaeological chronology, not directly related to ours, rely on fuzzy logics ([NH15]), aoristic analysis ([Cre12]), and evidence density estimation ([DD16]). For the Bayesian approach in radiocarbon, and its relation to CHRONOLOG, see Section 4.3.1.

2 Chronological Networks

First, a comment about our notation. In the discussion that follows, terms that receive a formal definition are capitalised, e.g., such as Chronological Networks, Time-periods, Sequences, and Chronological Relations.

We start by introducing our formalised model of Chronological Networks. The model allows representation of basic chronological units termed ”Time-periods”, grouped in ”Sequences” and related to each other through ”Chronological Relations”. We also discuss the advantages of relying on Chronological Networks for formalising archaeological data and queries.

2.1 Modelling the network

Our model of Chronological Networks features three types of objects: “Time-periods”, “Sequences”, and “Chronological Relations”.

2.1.1 Time-periods and Sequences

Time-periods. A *Time-period* represents a continuous interval of time, such as a king's reign, a historical era, or the time-span of an archaeological stratum (see Figure 1). It is characterised by a *start date*, an *end date*, and a *duration*. Our model allows for the following types of chronological constraints on dates: a start/end date can be *unknown*, *known* (e.g. 1984 CE), *lower bounded* (not earlier than 1984 CE), *upper bounded* (not later than 1984 CE) or known within a *range* (e.g. between 1984 and 1990). In the same way, durations can also be *unknown*, *known* (e.g. 5 years), *lower bounded* (at least 5 years), *upper bounded* (at most 5 years) or known within a *range* (e.g. between 5 and 10 years). A Time-period is thus represented by at most six numbers: minimum duration, maximum duration, earliest start date, latest start date, earliest end date, latest end date. Clearly, dates and durations are related since the duration of a Time-period is defined as the difference between its end and start dates. However, dates and durations are modelled separately since this allows constraints to be set independently on dates and durations. We use the following graphical notation: a Time-period is represented as a rectangle with the Time-period's name on top, its duration in the centre, its start date at the bottom left corner and its end date at the bottom right corner. Ranges are represented with square brackets (e.g. “[1984, 1990]”), upper bounds with the smaller-or-equal “$\leq$” sign (“$\leq$ 1984”), lower bounds with the greater-or-equal “$\geq$” sign (“$\geq$ 1984”) and unknown dates or duration with a question mark (see Figure 1). All the examples presented in this paper assume that the unit of time is the year (our model of Chronological Networks does however work in the same way for any other unit of time).

Sequences. A *Sequence* represents a set of consecutive Time-periods, with no gaps between them (see Figure 2). Hence, the end date of a Time-period always equals the start date of the next Time-period in the

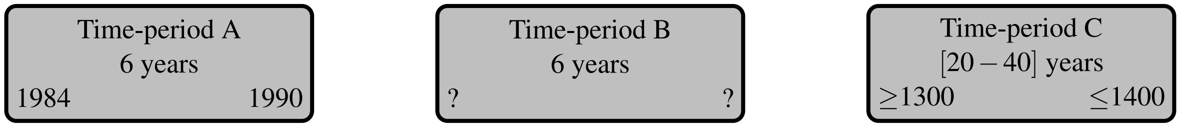


Figure 1: Three examples of Time-periods: Time-period A is fully known, Time-period B has unknown start and end dates, but a known duration, Time-period C has partial knowledge of its start/end dates and duration.

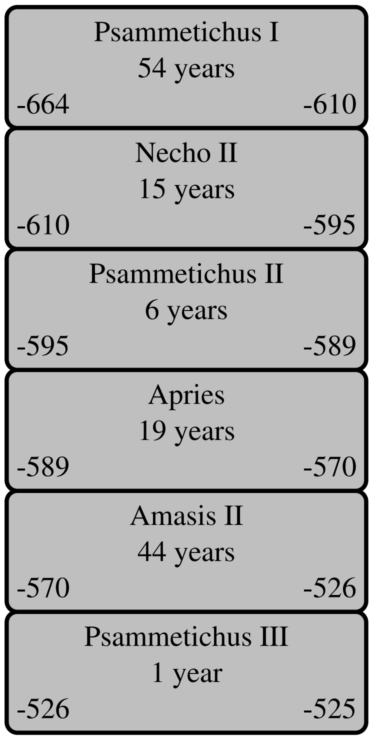
Sequence. In case one needs a Sequence that does feature a gap, an additional Time-period representing the gap must be inserted in the Sequence. Our model allows for the definition of Sequences having absolute chronology (known dates and duration, Figure 2a), floating Sequences (known durations but unknown dates, Figure 2b) or Sequences partially anchored in time (with partial knowledge of the start/end dates).

2.1.2 Chronological Relations

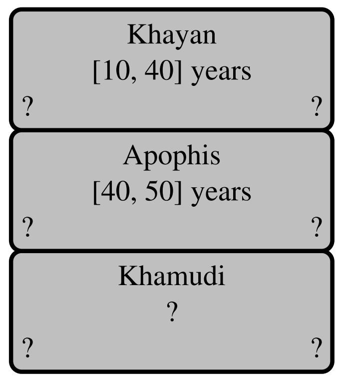
Chronological Relations express diverse types of relationships between two Time-periods. This section presents a wide set of Chronological Relations relevant for archaeological modelling. Chronological Relations are often referred to as "Synchronisms" in archaeological literature, though not all are strictly synchronic (see below).

Synchronic relations. A *contemporaneity synchronism* between two Time-periods A and B imposes that A and B have at least one unit of time in common. More precisely, it imposes that A cannot start after the end of B and that B cannot start after the end of A (see for example [Hol04, p. 136]). We define synchronic relations as the contemporaneity synchronism and special cases thereof (see below). Table 1 presents the contemporaneity synchronism, with a suggested notation, a graphic view of its four base cases, and a mathematical expression of its semantics. The contemporaneity synchronism is archaeologically relevant for cases of contemporaneity between kings, historical/archaeological eras, ceramic types, or archaeological strata. It is the most general type of synchronism, as it only imposes the presence of at least one common unit of time between the Time-periods, without any additional constraints. Table 2 presents more precise types of synchronisms, each of which is a special case of the contemporaneity synchronism:

- *Inclusion synchronisms:* A Time-period is entirely contained inside another. An example is an archaeological stratum that belongs solely to a given archaeological era (e.g. "Stratum V is included in the Iron Age II").
- *Overlap synchronisms:* Two Time-periods, besides sharing an intersection, each feature an extent of time not included in the other Time-period. An example is ceramic types that are consecutive, yet have a time of common production.
- *Start Period synchronisms:* The start of a Time-period is contained in another Time-period. An example is an archaeological stratum that starts during a given king's reign.
- *End Period synchronisms:* The end of a Time-period is contained in another Time-period. An example is an archaeological stratum that ends during a given king's reign.
- *Synchronised boundaries:* The start or end (or both) of two Time-periods are equal. For example, cases of several archaeological strata that were destroyed during the same event.



(a) Absolute chronology of the Egyptian 26th dynasty ([Kit00], p. 50)



(b) Relative chronology of the late Egyptian 15th dynasty, adapted from Ryholt's reconstruction of the Turin King List ([Ryh97], p. 119, Table 22). The bounds on Khayan derive from a preserved figure for decades equal to 10, 20 or 30, and those of Apophis from a preserved figure for decades equal to 40. In both cases, the number of years, months and days is lost in a lacuna, as is Khamudi's entire reign duration.

Figure 2: Two examples of Sequences, one (Egyptian 26th dynasty) with full knowledge of dates and durations and the other (Egyptian 15th dynasty) with only duration bounds.

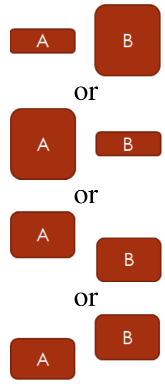
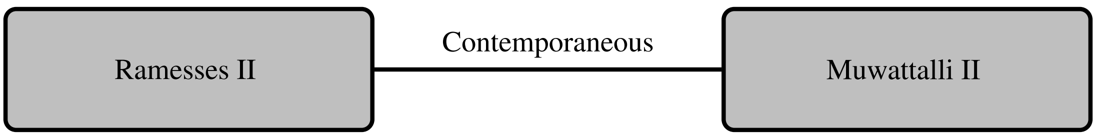
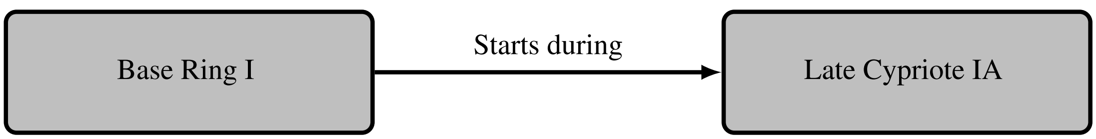
Name	Image	Notation	Semantics
Contemporaneity synchronism		$A \sim B$	$\mathsf{end}(A) \geq \mathsf{start}(B)$ **and** $\mathsf{end}(B) \geq \mathsf{start}(A)$

Table 1: The contemporaneity synchronism, with its four base cases, suggested notation, and formal semantics. In the images, time is assumed to flow from above to below.



(a) Contemporaneity synchronism between Pharaoh Ramesses II and Hittite King Muwattalli II ([Bry05], p. 377).



(b) "Starts during" synchronism between Cypriot Base Ring I ware and the Late Cypriote IA period ([Joh16], p. 253).

Figure 3: Graphical notations for Chronological Relations (dates and durations are omitted).

Asynchronic relations. An *asynchronism* is defined as a Chronological Relation between two Time-periods that have no unit of time in common. The asynchronisms included in our conceptual model are "A ends before the start of B" and "A starts after the end of B". Table 3 presents these asynchronisms, with their formal semantics and suggested notations.

Ordered boundaries. Table 4 presents *ordered boundaries*, Chronological Relations that represent an order between start and end dates (boundaries). These Relations are not necessarily synchronic or asynchronic.

Delay synchronisms. Table 5 presents *delay synchronisms*, a customisable type of Chronological Relation which expresses an exact, minimum or maximum delay between two boundaries.

Graphical notations. Chronological Relations are represented by a line (for symmetric relations) or an arrow (for non-symmetric relations) connecting two Time-periods (see Figure 3). The synchronism's name is written above the line or arrow.

In the sequel, we will refer to the data model of chronological networks presented here as the ChronoLog data model, named after the software application presented in Section 4.

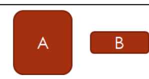
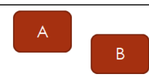
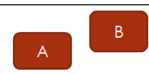
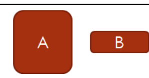
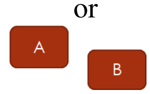
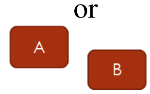
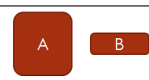
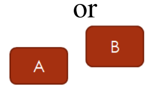
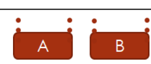
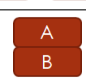
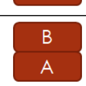
Name	Image	Notation	Semantics
Inclusion synchronisms			
A is included in B		$A \subseteq B$	$\mathsf{start}(A) \geq \mathsf{start}(B)$ **and** $\mathsf{end}(A) \leq \mathsf{end}(B)$
A includes B		$A \supseteq B$	$\mathsf{start}(A) \leq \mathsf{start}(B)$ **and** $\mathsf{end}(B) \leq \mathsf{end}(A)$
Overlap synchronisms			
A overlaps with succeeding B		$A \leq B$	$\mathsf{start}(A) \leq \mathsf{start}(B) \leq \mathsf{end}(A) \leq \mathsf{end}(B)$
A overlaps with preceding B		$A \geq B$	$\mathsf{start}(B) \leq \mathsf{start}(A) \leq \mathsf{end}(B) \leq \mathsf{end}(A)$
Start Period synchronisms			
A starts during B	 or 	$A \leftharpoonup B$	$\mathsf{start}(B) \leq \mathsf{start}(A) \leq \mathsf{end}(B)$
A includes the start of B	 or 	$A \rightharpoonup B$	$\mathsf{start}(A) \leq \mathsf{start}(B) \leq \mathsf{end}(A)$
End Period synchronisms			
A ends during B	 or 	$A \leftharpoondown B$	$\mathsf{start}(B) \leq \mathsf{end}(A) \leq \mathsf{end}(B)$
A includes the end of B	 or 	$A \rightharpoondown B$	$\mathsf{start}(A) \leq \mathsf{end}(B) \leq \mathsf{end}(A)$
Synchronised boundaries			
Synchronous start		$A \top B$	$\mathsf{start}(A) = \mathsf{start}(B)$
Synchronous end		$A \perp B$	$\mathsf{end}(A) = \mathsf{end}(B)$
Equality		$A = B$	$\mathsf{start}(A) = \mathsf{start}(B)$ **and** $\mathsf{end}(A) = \mathsf{end}(B)$
A precedes immediately B		$A \urcorner\!\llcorner B$	$\mathsf{end}(A) = \mathsf{start}(B)$
A follows immediately B		$A \lrcorner\!\ulcorner B$	$\mathsf{end}(B) = \mathsf{start}(A)$

Table 2: List of specialised cases of the contemporaneity synchronism, with suggested notations and formal semantics. Synchronisms have been paired with their inverse relation, except for synchronised boundaries, which have no inverse relations. In the images, time is assumed to flow from above to below.

Name	Image	Notation	Semantics
Ordered asynchronisms			
A ends before the start of B		$A \ll B$	$\mathsf{end}(A) < \mathsf{start}(B)$
A starts after the end of B		$A \gg B$	$\mathsf{start}(A) > \mathsf{end}(B)$

Table 3: List of asynchronisms, with suggested notations and formal semantics. In the images, time is assumed to flow from above to below.

Name	Image	Notation	Semantics
Ordered start			
A starts before the start of B		$\overline{A} < \overline{B}$	$\mathsf{start}(A) < \mathsf{start}(B)$
A starts after the start of B		$\overline{A} > \overline{B}$	$\mathsf{start}(A) > \mathsf{start}(B)$
Ordered end			
A ends before the end of B		$\underline{A} < \underline{B}$	$\mathsf{end}(A) < \mathsf{end}(B)$
A ends after the end of B		$\underline{A} > \underline{B}$	$\mathsf{end}(A) > \mathsf{end}(B)$
Ordered start/end			
A starts before the end of B		$\overline{A} < \underline{B}$	$\mathsf{start}(A) < \mathsf{end}(B)$
A ends after the start of B		$\underline{A} > \overline{B}$	$\mathsf{end}(A) > \mathsf{start}(B)$

Table 4: List of ordered boundaries, with suggested notations and formal semantics. Synchronisms have been paired with their inverse relation. In the images, time is assumed to flow from above to below.

Delay synchronisms

$$A \left\{ \begin{array}{l} \text{starts} \\ \text{ends} \end{array} \right. \left\{ \begin{array}{l} \text{exactly} \\ \text{at least} \\ \text{at most} \end{array} \right. X \text{ years} \left\{ \begin{array}{l} \text{before} \\ \text{after} \end{array} \right. \left\{ \begin{array}{l} \text{start of} \\ \text{end of} \end{array} \right. B$$

Table 5: Delay synchronisms.

2.1.3 Expressiveness of the model

Expressiveness. The model presented above allows us to represent most sorts of relevant archaeological knowledge. It works for absolute chronologies (known start and end dates) but also for relative chronologies (unknown, or partially known, start and end dates). The model can also deal with gaps in a stratigraphic sequence (as after a major destruction in a site) by inserting gap Time-periods between Time-periods representing strata. Co-regencies can be handled by inserting a co-regency Time-period between two "sole reign" Time-periods of the same Sequence. Partial co-existence of two succeeding pottery types (or cultural phases) can be represented in the same way, by creating a third Time-period between the two pottery Time-periods, within the same Sequence. Alternatively, one can also create two single-Period ceramic sequences, one for each pottery type, and link them with an Overlap synchronism. Discrete historical events (say the Fall of Constantinople) can be represented by a Time-period having a zero duration. In the same way, single-burial tombs will also be allotted a zero duration, and multi-burial tombs a non-zero duration.

Limitations. Our model also presents a number of limitations. For example, it cannot model a reign of "5 *or* 15" years, although such constraints do occasionally occur in archaeology, due to badly preserved numerals on inscriptions. In such a case, we would need to use a weaker constraint, namely the range "[5-15]" years. The same limitation also applies to start/end dates. Furthermore, Chronological Relations that necessitate an *or*-operator also fall outside of the model (note that all the Chronological Relationships presented above feature only *and*-relationships). An example of such a Chronological Relation is the *General Asynchronism*, defined as "*A* ends before the start of *B or A* starts after the end of *B*". The reason for limiting ourselves to *and*-relationships is in order to be able to analyse the network using fast algorithms (see Section 3 below and [GLP17]).

2.1.4 Facing archaeological complexity

This section discusses how the ChronoLog data model can be applied to real-life archaeological data. As formal modelling objects, ChronoLog Time-periods have a unique start and end date, and ChronoLog Sequences contain Time-periods in direct succession, without gaps or overlaps. Such simplified definitions directly fit only specific types of archaeological data, such as strata delimited by destruction layers, and kings reigning in direct succession. Archaeological periods however (representing cultural phases, say Late Bronze I, or Iron Age II), as modern abstractions of ancient material culture, do not have a single start and end date, since given material traits appear gradually, and can start at different times in different regions. One archaeological context can already exhibit, say, Iron Age II material characteristics, while another contemporary context still exhibits Iron Age I characteristics. Furthermore, consecutive cultural phases always feature a certain amount of overlap with each other, as given material traits do not disappear overnight, but coexist with new ones, even in the same region. We show here that the ChronoLog data model has the required flexibility to describe even such complex cases.

First, although archaeological periods do not have a single start and end date, archaeologists do routinely grant them approximate dates ("*We must therefore place the boundary between [Corinthian] LG and EPC very near 720 B.C.*" [Col08, p. 316]), absolute bounds ("*This would place the start of Middle Cypriote III earlier than 1700 B.C.*" [Mer02, p. 6]) or relative bounds ("*there can be no doubt that LC [Late Cypriot] IA started before the end of the Second Intermediate Period.*" [Mer92, p. 50]). Such cases can be modelled within the ChronoLog data model by using ranges, bounds and Chronological Relations, respectively. The problem of an overlap between two archaeological periods can be handled either by inserting an overlap Time-period between the two archaeological periods, or by splitting them into two ChronoLog Sequences and adding an *overlap synchronism* between them. Finally, the problem of regional changes can be dealt with by building several regional sequences, instead of one master sequence.

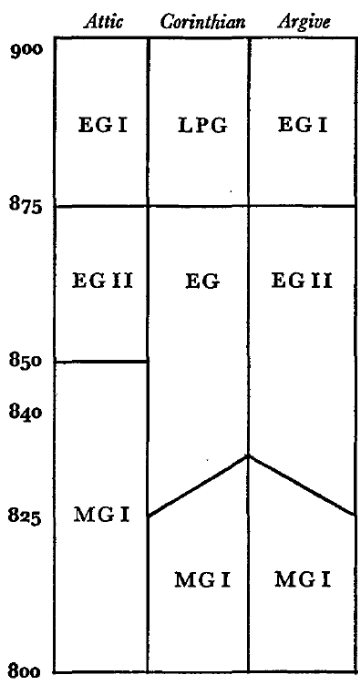
We illustrate these techniques with an example from Greek archaeology ([Col08], p. 327-330). Figure 4a presents an excerpt of Coldstream's chronological chart of Geometric ceramic styles, featuring three regional sequences. The Attic sequence is approximated in Coldstream's chart as a pure sequence, with no overlaps. The Corinthian and Argive sequences, however, do show an overlap between the EG and MG phases, justified thus by Coldstream: "*In Corinthian and Argive, the grave groups show that the transition from EG to MG was more gradual than in Attica.*" ([Col08], p. 328). The chart does give precise figures for most transitions (900, 875, 850, 800) but the accompanying text explicitly notes that these figures are approximate ([Col08], p. 227-229). Figure 4b presents a simple version of Coldstream's chart using the CHRONOLOG data model. The Attic sequence is modelled *as is*, using one CHRONOLOG sequence, without overlaps. The Corinthian sequence was modelled using an extra Time-period representing the EG-MG I overlap, which, in Coldstream's chart, starts after 840 and finishes in 825. The Argive EG II-MG I overlap was modelled in the same way. Figure 5 presents alternative modelling options. We first show an alternative modelling for the Corinthian EG-MG I overlap, where the EG and MG I are connected by an overlap synchronism, rather than using an overlap Time-period (Figure 5b). CHRONOLOG also allows to explicitly model the approximate aspect of Coldstream's transition figures, for example by widening them to 20-year ranges (Figure 5c).

In short, the CHRONOLOG data model allows to express complex archaeological realities by building models of increasing size and complexity. A fully flexible model would ideally feature regional sequences (rather than one master sequence), ranges for every transition, and overlaps between every pair of successive phases. For modelling specific ceramic types (say Cypriot Base-Ring I), one could even use two separate Time-periods, representing the type's *production* and *use*, respectively. Both would share a common start, but use would end later than production. One must remember however that a model is, by definition, a conventional and approximated description of reality. It is up to the user to decide on the model's degree of precision, according to the needs of his research. The CHRONOLOG data model cannot solve the inherent difficulties of defining archaeological phases, which are purely a matter of archaeological judgement. It rather aims at providing archaeologists with a practical tool for building chronological models and deriving chronological information from them (see Sec. 2.2).

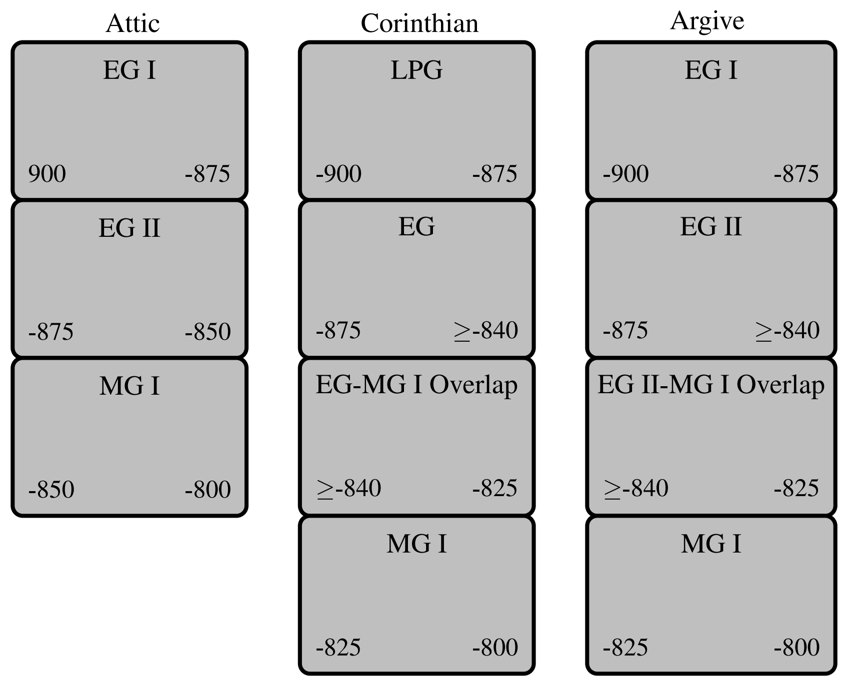
Finally, CHRONOLOG enables archaeologists to *explicitly* model their definition of an archaeological period (say "Late Bronze IIA"), by using a dateless (i.e. floating) Time-period, and appropriate synchronisms between that period and its defining artifacts (*fossiles directeurs*) and strata. In this way, the given archaeological period is not a vaguely-defined "black box", nor an *input* of the chronological model, but rather an *output* of the model, inheriting its absolute chronology from the more concrete Time-periods representing strata and artifact types. Should new data later modify our understanding of that period, we could directly update its definition in the model (by adding or removing synchronisms with specific artifact types and strata), in order to assess how the change affects the period's chronology.

2.1.5 Example: the Kingdom of ChronoLand

We close this section by introducing a "toy" example, dubbed *ChronoLand*, that we will use as a running example throughout the rest of this paper. In the Kingdom of ChronoLand, Kings K_1 and K_2 reigned in succession. We do not know their precise reign dates, but both reigns are known to have occurred between 1200 and 1300 CE. We also know from ancient annals that King K_1's reign did not exceed 10 years, and we know from epigraphic sources that King K_2's reigned at least 35 years. Recent excavations at *ChronoCity*, the capital city of ChronoLand, have unearthed two archaeological strata: S_1 and S_2. The earlier stratum, S_1, was built on bedrock, and contained an in-situ stela of King K_1, claiming he built ChronoCity. The latest stratum, S_2, was destroyed by fire in a heavy conflagration. According to ancient annals, the city was destroyed during the reign the reign of King K_2 and was never reoccupied. Finally, we assume that each of our strata has a

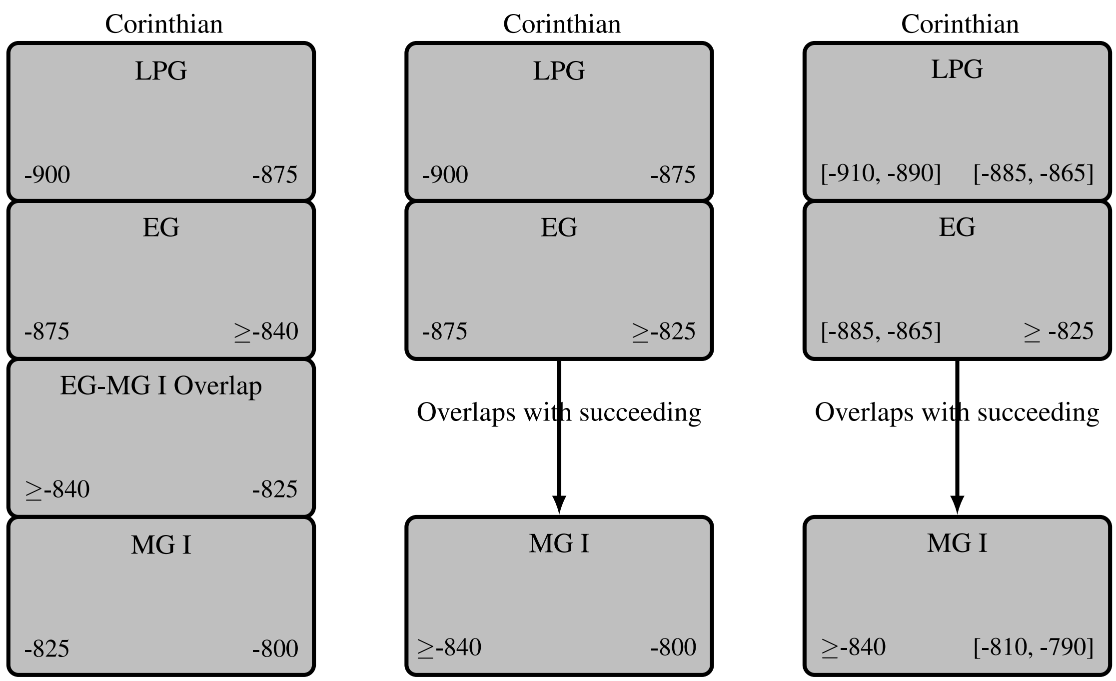


(a) Coldstream's chronological chart of the Greek Geometric period, showing the Attic, Corinthian and Argive regional sequences ([Col08], p. 330, partial view).



(b) Equivalent representation of Coldstream's Attic, Corinthian and Argive chronological sequences, using the CHRONOLOG data model.

Figure 4: Modelling regional archaeological periods, with and without overlaps.



(a) Simple modelling (fixed dates), with an overlap Time-period.

(b) Simple modelling (fixed dates), with an overlap synchronism.

(c) Modelling with date-ranges and an overlap synchronism.

Figure 5: Three different options for modelling Coldstream's sequence of Corinthian Late Protogeometric to Middle Geometric I (see Figure 4a). Figure 5a and Figure 5b show two equivalent ways to represent an overlap: with a Time-period or an overlap synchronism. Figure 5c shows how to add uncertainty on the boundary dates, by using 20-year ranges instead of fixed dates.

duration of at least 20 years and at most 100 years. The modelling of these data as a Chronological Network is shown in Figure 6a. The Chronological Network synthesises in a clear and unambiguous way the data derived from all our sources. We discuss below the computational operations possible on this network and the conclusions that can be drawn from them.

2.2 Querying the network

We now wish to define two basic operations that a user might want to perform on a Chronological Network.

2.2.1 Consistency check

The *consistency check* operation verifies whether the encoded chronological data feature a contradiction. As an example, a slight variant in the previous ChronoLand example (see Figure 7) yields a non-consistent network. In this variant, King K_2's duration is not set to at least 35 years, but rather to at most 25 years. Why is such a model not consistent? The two upper bounds on K_1 and K_2 yield a 35 years upper bound on the dynasty's duration. However, the two strata S_1 and S_2 have a combined duration of at least 40 years and therefore cannot be included within the duration of the ChronoLand dynasty, which is at most 35 years (10+25). In the simple case of ChronoLand, this contradiction can be detected by the "naked eye". In larger networks, featuring dozens of Time-periods and synchronisms, only an automated consistency check is capable of detecting all possible faults. A formal algorithm for consistency check of Chronological Networks will be presented in Section 3.

2.2.2 Tightening

A Chronological Network as defined above gathers all the chronological information known by the researcher. Based on this input information, more knowledge can be deduced regarding the dates and durations of the Time-periods. The *tightening* operation is the search for the tightest possible bounds for each start date, end date, and duration. These bounds are *optimal*, in the sense that they characterise *exactly* the set of allowed values for the start/end dates and durations. Any value outside these bounds violates a constraint of the network. And any further restriction of a bound would imply rejecting an allowed value, i.e. one that *does not* violate any constraint. In practical terms, the tightening operation makes all upper bounds as small as possible (e.g. a 1280 upper bound for a date is more precise than 1300), and all lower bounds as large as possible (e.g. 1220 is more precise than 1200). The result of the tightening operation applied to ChronoLand is shown in Figure 6b. Some of the new bounds are straightforward (e.g. the 1200 latest start of K_1 derives from the 1200 earliest start of K_1) while others are more complex (see below). Where do those improved bounds come from? Typically, they come from some given input data that propagates along the Network from one Time-period to another, following a trail of Chronological Relations. As an example, let us look at the 1240 earliest end of K_2. It derives from the following considerations:

1. K_1 starts after 1200

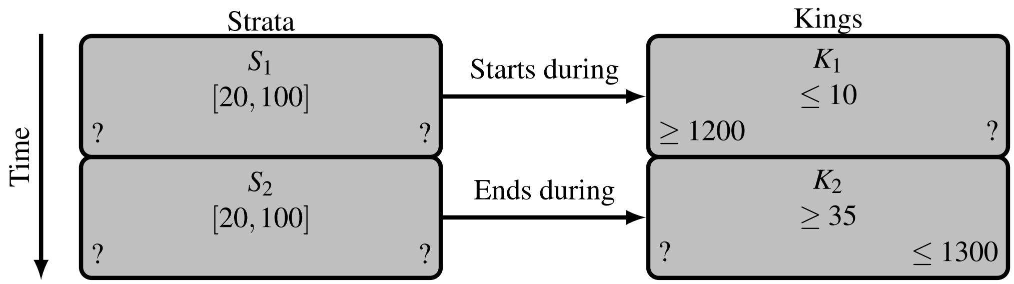
2. S_1 starts during K_1, hence it also starts after 1200

3. S_1 lasts at least 20 years, hence it ends after 1220

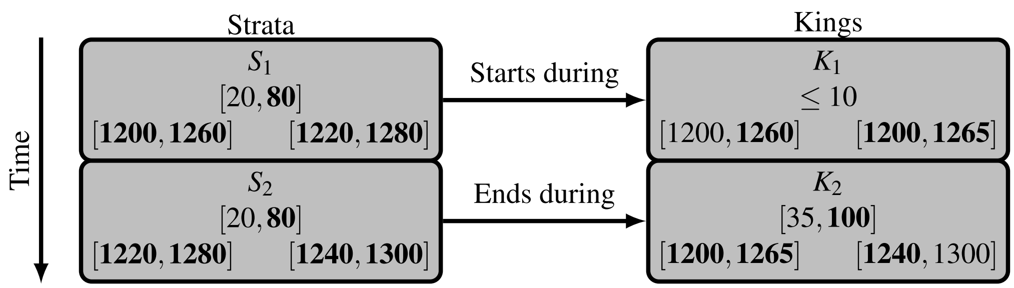
4. S_2 starts when S_1 ends, hence it also starts after 1220

5. S_2 lasts at least 20 years, hence it ends after 1240

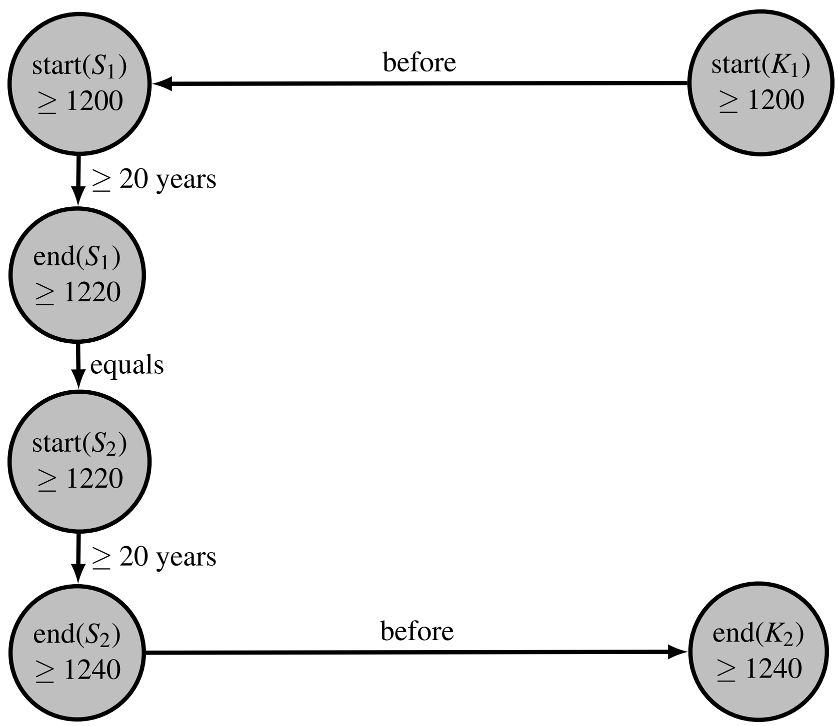
6. S_2 ends during K_2 hence K_2 ends after S_2 does, hence after 1240.



(a) Chronological Network showing the input constraints of ChronoLand.



(b) Result of the tightening procedure. Enhanced upper and lower bounds are shown in bold.



(c) Trace for the 1240 earliest end of K_2.

Figure 6: The ChronoLand example: input constraints, tightened ranges, and example of a trace.

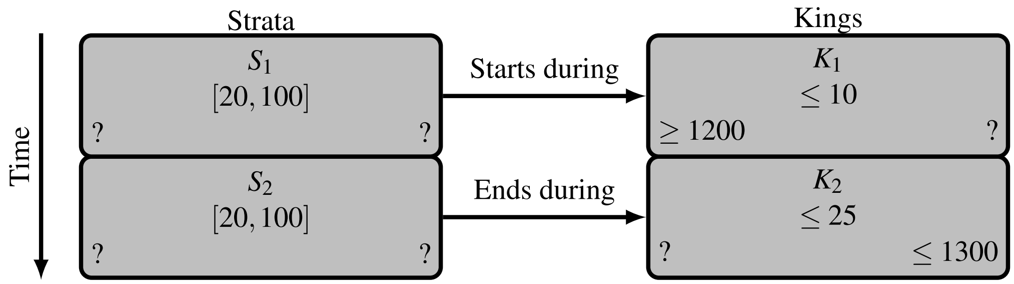


Figure 7: Example of a non-consistent Chronological Network. The ChronoLand dynasty has at most 35 years, while the two strata have a combined duration of at least 40 years. This yields a contradiction since the strata are known to be included in the time-span of the dynasty (by the "Starts during" and "Ends during" relations), which is too short to accommodate 40 years.

Such an explanation for a tightened bound is called a *trace*. It consists of a path along the network, starting (in this example) from one given *source* (the 1200 earliest start of K_1) and propagating along the network from K_1 to K_2, via S_1 and S_2, following the two given synchronisms. Figure 6c provides a graphical view of this trace.

2.2.3 Discussion

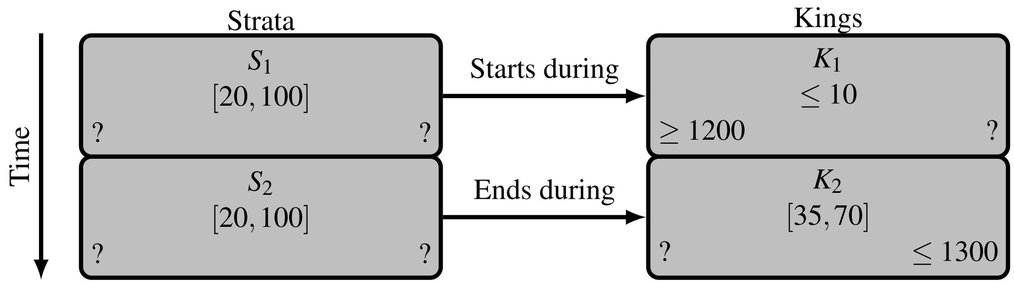
The ChronoLand example shows that searching for the tightest range without the help of a computer is not easy, even in a simple example, let alone in a real archaeological case featuring hundreds of Time-periods and many constraints. One can also easily miss the optimal propagation path. In the above example (earliest end of K_2), one could easily have taken an alternative path, starting from K_1 and going directly to K_2, resulting in a 1235 earliest end of K_2 (through K_2's 35 years minimum duration) rather than the optimal 1240. Note also that in some cases, the tightening process features unexpected phenomena, as in the above example, where a Sequence having no absolute chronology of its own (S_1 and S_2) helped tighten the date-ranges of Time-periods that do have an absolute chronological estimate as input (K_1 and K_2, included between 1200 and 1300). In archaeological research, failing to apply the tightening procedure fully and correctly will often result in sub-optimal chronologies. Indeed, whereas chronological papers often do provide the sources of their absolute chronology (though often not in a full and formal way), seldom do they present the full consequences of this prior knowledge. In the sequel of this paper, the bounds encoded in the network before the tightening procedure will be called *input* bounds. They represent chronological information established (known or hypothesised) *a priori* by the researcher. The bounds resulting from the tightening procedure will then be called *computed* bounds, since they need to be calculated (see Section 3).

2.3 Using the network

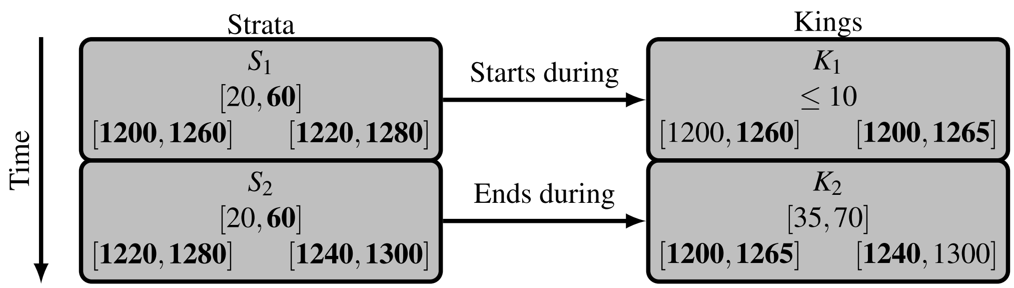
2.3.1 Practical usage

The operations described above allow to *check the global impact of local changes* to the network and to *test chronological hypotheses*, as described below.

Checking the impact of local changes. What if we added a 70 years upper bound to King K_1's reign (in addition to the 35 years lower bound)? Surely such an upper bound is quite realistic, since seldom in History has a king reigned more than 70 years. How would this new constraint affect our network? Will it affect any



(a) The ChronoLand example with an upper bound of 70 years added to the specification of King K_2's duration.



(b) Result of the tightening operation

Figure 8: The ChronoLand example with an additional 70 years upper bound on King K_2's duration.

of the tightened ranges, and how? The answer to this question is shown in Figure 8: the maximum durations of S_1 and S_2 have been reduced from 80 years to 60 years. This result is not easy to observe manually. We have a 60 years upper bound because the full length of the ChronoLand dynasty is now at most 80 years (10+70), while each stratum has at least 20 years. Since the stratigraphic sequence is included in the dynasty's length (via the "starts during" and "ends during" synchronisms), each stratum can have at most 60 years (80-20). More generally, a local change (addition, removal, or update of a constraint) can have three outcomes: (1) no computed range is affected, (2) at least one computed range is affected, (3) a contradiction is created.

Testing chronological hypotheses. We already know from our inputs that Stratum S_1 was built by King K_1 . Is it possible that he also built Stratum S_2? Note that the answer does not pop up immediately by simply looking at the computed bounds (Figure 6b): the computed start date of S_2 ($[1220,1280]$) could apparently fit both in K_1's reign (start date in $[1200,1260]$, end date in $[1200,1265]$) and in K_2's reign (start date in $[1200,1265]$, end date in $[1240,1300]$). To check the hypothesis that King K_1 built Stratum S_2, we add a synchronism "S_2 starts during K_1" and check the feasibility of the network. The resulting network is not consistent. If S_2 started during K_1's reign, his reign would have needed to encompass the whole duration of Stratum S_1 (which was built during his reign, see above), hence it would have at least 20 years, which contradicts the 10 years maximum duration of K_1. Hence, although it was not obvious at first sight, our set of input data do in fact imply that only King K_2 could have built Stratum S_2. In other words, a chronological network hides much more knowledge than appears at first sight. This shows that in many practical archaeological cases, important chronological conclusions might have been overlooked by the researchers, through lack of a computational tool.

2.3.2 Advantages

We end this section by listing the advantages of the formal approach to Chronological Networks:

1. *Clear disclosure of all ground hypotheses.* All the hypotheses from which the final chronology derives are explicitly laid out as inputs in the Chronological Network. Each range of the final chronology can therefore be entirely justified in terms of these inputs, by using traces. There are thus no hidden assumptions, common knowledge, circular reasoning, or "rules of thumb" involved in the process of chronology-building. Ideally, the ground hypotheses should be undisputed facts of chronology, but one might also want to take specific (debated) hypotheses as ground inputs, in order to test whether these hypotheses are valid or what their precise consequences would be on the overall network. Furthermore, the full disclosure of inputs allows contenders of a given chronology to simply change the inputs they do not agree with, and recompute the tightneded ranges in order to obtain an alternative chronology.

2. *Separation between the combinatorial structure of the network and its absolute chronology.* In our approach, the combinatorial structure of the network (given by the Sequences and Chronological Relations) is clearly separated from the aspects of absolute dating. The latter are represented by the adjunction of chronological estimates only at some specific points in the network (in our case, the 1200 earliest start of K_1 and the 1300 latest end of K_2) and the rest of the absolute chronology, for all Time-periods, is then computed automatically by the tightening operation, as the few input absolute dating estimates propagate along the network. The structure of the network thus remains unchanged, even if the input absolute chronological estimates are later changed (if say, K_1 and K_2 are to be re-dated to the range $[1300-1400]$ instead of $[1200-1300]$).

3. *Optimality.* The ChronoLand example demonstrated that computing the tightened ranges for dates and durations is difficult, even for small networks. On larger, life-size Chronological Networks, in addition to being tedious and error-prone, such computation is virtually impossible without the help of a computer. Algorithmic computation of the tightened ranges (see Section 3) guarantees the optimal, i.e. tightest possible, ranges for each date and duration.

4. *Knowledge discovery.* As seen in the ChronoLand example, some interesting chronological knowledge sometimes lies hidden within the network's structure, as the fact that only King K_2 could have built Stratum S_2. The formal approach opens the way to the use of algorithms to automatically discover such relations.

5. *Tagged chronologies.* The computational approach has the potential to produce several alternative chronologies for the same network, based on inclusion or exclusion of given sets of constaints. More precisely, every constraint in the network could be tagged with labels describing their type, such as "literary data", "stratigraphic data", "epigraphic data", or "astronomical dates". This is especially interesting for complex case studies, involving many different types of basic data. As an example, for the chronology of ancient Egypt, one could be interested in the effect that an exclusion of astronomical dates would have on the overall chronology.

6. *Classification of constraints.* The computational approach also allows to classify chronological constraints to distinguish between those that do or do not impact the global network. For example, the date bounds of 1200 and 1300 on the ChronoLand kings have a strong impact on the network, since they provide the source of absolute chronology for all the computed dates. On the other hand, the 100 years upper bounds on Strata S_1 and S_2 of ChronoLand have no impact on any of the computed

bounds of the network. They can be removed from the model without impacting the results. Spotting such "low-impact" constraints is of great chronological interest, though not easy to do without a computational tool.

In conclusion, the proposed approach to chronology makes it possible to study complex Chronological Networks in a more rigorous, rational, and scientific way.

3 Mathematical modelling

This section presents a mathematical formalisation of Chronological Networks and shows how to solve the tightening and consistency problems algorithmically. We have tried to avoid an excess of mathematical formalism, and presented the results with limited mathematical notations and no formal proofs. Examples are used in order to help the reader grasp the notions at work. Grasping the mathematical model is not necessary in order to use the CHRONOLOG software. Readers with no interest in the mathematical modelling can thus skip directly to Section 4. The reader interested in a more formal treatment of these results is referred to our previous publication [GLP17].

We first show how a complete Chronological Network can be expressed as a set of inequalities between the boundaries of the Time-periods (i.e. start and end dates) (Section 3.1). We then show how this set of inequalities can be represented as a graph (Section 3.2) and finally show how the tightening and consistency check problems can be solved using graph algorithms (Section 3.3).

It is worth noting that the techniques presented in this section consist in manipulating and analysing simple constraints on the start and end dates of the Time-periods. Such techniques have been introduced in the field of optimisation and linear programming [Sho81], and of formal verification [Dil89]. They have been widely applied in several fields of computer science, including computer aided verification [AD94, BDM$^+$98, BDL04] and artificial intelligence [All84] (to name a few), but never, as far as we are aware of, in the field of archaeology. The underlying algorithms for analysing these constraints are standard graph algorithms which have been well-studied for several decades (see for instance[Flo62]).

3.1 The Chronological Network as a set of inequalities

Let us define B as the set of boundaries (start dates and end dates) of all Time-periods of a given Chronological Network. For example, in the case of ChronoLand, the Time-periods are S_1, S_2, K_1 and K_2, so we have:

$$B = \{\mathsf{start}(S_1), \mathsf{end}(S_1), \mathsf{start}(S_2), \mathsf{end}(S_2), \mathsf{start}(K_1), \mathsf{end}(K_1), \mathsf{start}(K_2), \mathsf{end}(K_2)\}$$

where $\mathsf{start}(p)$ and $\mathsf{end}(p)$ represent respectively the beginning and the end of the Time-period p. Our goal is to represent a Chronological Network as a set of logical constraints involving only boundaries and constants. The rules of this representation are given now.

Time-periods. For each Time-period, we need to encode constraints on boundaries and durations. For a boundary b, the absolute time constraints can have the shape $b \geq k$ (Lower bound), $b \leq k$ (Upper bound), $b \geq k_1$ **and** $b \leq k_2$ (Range) or $b = k$ (Exact date), with constant values k, k_1, k_2. For the Time-period p, its duration is represented as $\mathsf{end}(p) - \mathsf{start}(p)$. The duration constraints can thus have the shape $\mathsf{end}(p) - \mathsf{start}(p) \geq k$ (Lower bound), $\mathsf{end}(p) - \mathsf{start}(p) \leq k$ (Upper bound), $\mathsf{end}(p) - \mathsf{start}(p) \geq k_1$ **and** $\mathsf{end}(p) - \mathsf{start}(p) \leq k_2$ (Range), $\mathsf{end}(p) - \mathsf{start}(p) = k$ (Exact duration), $\mathsf{end}(p) - \mathsf{start}(p) \geq 0$ (Unknown duration), with constant values k, k_1, k_2.

$\mathsf{start}(K_1) \geq 1200$	(Earliest start of K_1)
and $\mathsf{end}(K_2) \leq 1300$	(Latest end of K_2)
and $\mathsf{end}(S_1) - \mathsf{start}(S_1) \geq 20$ **and** $\mathsf{end}(S_1) - \mathsf{start}(S_1) \leq 100$	(Duration of S_1)
and $\mathsf{end}(S_2) - \mathsf{start}(S_2) \geq 20$ **and** $\mathsf{end}(S_2) - \mathsf{start}(S_2) \leq 100$	(Duration of S_2)
and $\mathsf{end}(K_1) - \mathsf{start}(K_1) \geq 0$ **and** $\mathsf{end}(K_1) - \mathsf{start}(K_1) \leq 10$	(Duration of K_1)
and $\mathsf{end}(K_2) - \mathsf{start}(K_2) \geq 35$	(Duration of K_2)
and $\mathsf{end}(K_1) = \mathsf{start}(K_2)$	(Sequence of kings)
and $\mathsf{end}(S_1) = \mathsf{start}(S_2)$	(Sequence of strata)
and $\mathsf{start}(S_1) \geq \mathsf{start}(K_1)$ **and** $\mathsf{start}(S_1) \leq \mathsf{end}(K_1)$	(S_1 starts during K_1)
and $\mathsf{end}(S_2) \geq \mathsf{start}(K_2)$ **and** $\mathsf{end}(S_2) \leq \mathsf{end}(K_2)$	(S_2 ends during K_2)

Figure 9: The ChronoLand example presented as a set of logical constraints.

Sequences. For each Sequence, we need to encode the fact that the end of a Time-period equals the start of the next one. Hence, for each two consecutive Time-periods p_1 and p_2 of a Sequence, we have the constraint: $\mathsf{end}(p_1) = \mathsf{start}(p_2)$.

Chronological Relations. Each Chronological Relation defined in Section 3 has been formally defined using equations and inequalities in Tables 1, 2, 3 and 4.

All the information from a Chronological Network can be encoded by means of constraints on the boundaries. For example, Figure 9 provides the full encoding of the ChronoLand example as a set of inequality constraints. The above-defined constraints for Time-periods, Sequences, and Chronological Relations need to be combined with " **and** " logical connectors (conjunction) since we need all of them to hold true. This yields a large *global constraint*, as shown in Figure 9, which exhibits the following aspects: (1) the constraint is a *conjunction* of inequalities, in the sense that it features only the " **and** " logical connector. All other operators, including "**or**" and "**not**" are disallowed. This will be crucial in the sequel of the paper, in order to obtain efficient algorithms to analyse Chronological Networks; (2) all the elements that are combined by means of the " **and** " operator are inequalities or equalities comparing either a single boundary or a difference of two boundaries to a constant value (for example, $\mathsf{start}(p) \leq k$ or $\mathsf{end}(p) - \mathsf{start}(p) \geq k$).

3.2 The Chronological Network as a graph

In order to analyse Chronological Networks expressed as constraints, we will translate these constraints into *graphs* and rely on standard graph algorithms. This section explains how this is being done.

Normalising the constraints. The objective of the normalisation procedure is to rewrite the constraints as a conjunction of simple constraints having all the same basic shape:

$$b_1 - b_2 \leq k,$$

where b_1 and b_2 are boundaries in B and k is a constant. Each of these simple constraints will be called an *atomic constraint*. Note that the only comparison allowed in those simple constraints is $\leq$, i.e. all of the following are disallowed: $\geq$, $<$, $>$ and $=$. To achieve this normalisation, equalities such as $\mathsf{end}(K_1) = \mathsf{start}(K_2)$ are being rewritten as a conjunction of two inequalities: $\mathsf{end}(K_1) \leq \mathsf{start}(K_2)$ **and** $\mathsf{end}(K_1) \geq \mathsf{start}(K_2)$,

$$
\begin{array}{lr}
\quad z_0 - \mathsf{start}(K_1) \leq -1200 & \text{(Earliest start of } K_1\text{)} \\
\textbf{and } \mathsf{end}(K_2) - z_0 \leq 1300 & \text{(Latest end of } K_2\text{)} \\
\textbf{and } \mathsf{start}(S_1) - \mathsf{end}(S_1) \leq -20 \textbf{ and } \mathsf{end}(S_1) - \mathsf{start}(S_1) \leq 100 & \text{(Duration of } S_1\text{)} \\
\textbf{and } \mathsf{start}(S_1) - \mathsf{end}(S_2) \leq -20 \textbf{ and } \mathsf{end}(S_2) - \mathsf{start}(S_2) \leq 100 & \text{(Duration of } S_2\text{)} \\
\textbf{and } \mathsf{start}(K_1) - \mathsf{end}(K_1) \leq 0 \textbf{ and } \mathsf{end}(K_1) - \mathsf{start}(K_1) \leq 10 & \text{(Duration of } K_1\text{)} \\
\textbf{and } \mathsf{start}(K_2) - \mathsf{end}(K_2) \leq -35 & \text{(Duration of } K_2\text{)} \\
\textbf{and } \mathsf{end}(K_1) - \mathsf{start}(K_2) \leq 0 \textbf{ and } \mathsf{start}(K_2) - \mathsf{end}(K_1) \leq 0 & \text{(Sequence of kings)} \\
\textbf{and } \mathsf{end}(S_1) - \mathsf{start}(S_2) \leq 0 \textbf{ and } \mathsf{start}(S_2) - \mathsf{end}(S_1) \leq 0 & \text{(Sequence of strata)} \\
\textbf{and } \mathsf{start}(K_1) - \mathsf{start}(S_1) \leq 0 \textbf{ and } \mathsf{start}(S_1) - \mathsf{end}(K_1) \leq 0 & \text{(} S_1 \text{ starts during } K_1\text{)} \\
\textbf{and } \mathsf{start}(K_2) - \mathsf{end}(S_2) \leq 0 \textbf{ and } \mathsf{end}(S_2) - \mathsf{end}(K_2) \leq 0 & \text{(} S_2 \text{ ends during } K_2\text{)} \\
\textbf{and } z_0 - \mathsf{end}(K_1) \leq 0 & \text{(Earliest end of } K_1\text{)} \\
\textbf{and } z_0 - \mathsf{start}(K_2) \leq 0 & \text{(Earliest start of } K_2\text{)} \\
\textbf{and } z_0 - \mathsf{end}(K_2) \leq 0 & \text{(Earliest end of } K_2\text{)} \\
\textbf{and } z_0 - \mathsf{start}(S_1) \leq 0 & \text{(Earliest start of } S_1\text{)} \\
\textbf{and } z_0 - \mathsf{end}(S_1) \leq 0 & \text{(Earliest end of } S_1\text{)} \\
\textbf{and } z_0 - \mathsf{start}(S_2) \leq 0 & \text{(Earliest start of } S_2\text{)} \\
\textbf{and } z_0 - \mathsf{end}(S_2) \leq 0 & \text{(Earliest end of } S_2\text{)}
\end{array}
$$

Figure 10: The normalised constraint for the ChronoLand example.

which further rewrites to $\mathsf{end}(K_1) - \mathsf{start}(K_2) \leq 0$ **and** $\mathsf{start}(K_2) - \mathsf{end}(K_1) \leq 0$. Similarly, strict inequalities such as $\mathsf{end}(K_1) - \mathsf{start}(K_2) < k$ are expressed as non-strict inequalities, i.e. $\mathsf{end}(K_1) - \mathsf{start}(K_2) \leq k - 1$. In order to normalise absolute date bounds, like $\mathsf{end}(K_2) \leq 1300$, we need to add a new boundary to B, called z_0, and corresponding to the pre-defined origin of time. This origin of time is our reference point, i.e. our "date 0", and needs to be chosen according to the dates that will be manipulated in the example. For example, if all our dates fall within the 26th Dynasty of Egypt (664 BC to 525 BC, [Kit00]), we could safely choose z_0 to correspond to 700 BC. In this case, the year 664 BC would be encoded as 700-664=36, and the year 525 BC as 700-525=175. The point of setting this reference date is to ensure that all the dates that our algorithms will need to consider are not negative. We assume, for the rest of the paper, that z_0 corresponds to date 0. In this case, the upper bound $\mathsf{end}(K_2) \leq 1300$ becomes $\mathsf{end}(K_2) - z_0 \leq 1300$, and the lower bound $\mathsf{start}(K_1) \geq 1200$ becomes $-\mathsf{start}(K_1) \leq -1200$ which rewrites to $z_0 - \mathsf{start}(K_1) \leq -1200$. Finally, for each boundary b that does not already have a lower bound, we add the constraint that it occurs after the origin of time, hence after z_0, thus: $b \geq z_0$, which normalises to $z_0 - b \leq 0$.

As an example, the normalised constraint for the ChronoLand example is given in Figure 10. Note that the normalisation procedure produces a constraint which is *equivalent* to the original one in the sense that the possible values for the boundaries that satisfy the original constraint are the same that satisfy the normalised constraint. Clearly, all constraints resulting from a Chronological Network can be turned into such an equivalent normalised constraint, using the procedure sketched in this section. From now on, we will thus assume that all constraints are normalised, i.e. are a conjunction of atomic constraints of the form $b_1 - b_2 \leq k$, where b_1 and b_2 are boundaries (including the special boundary z_0) and k is a constant.

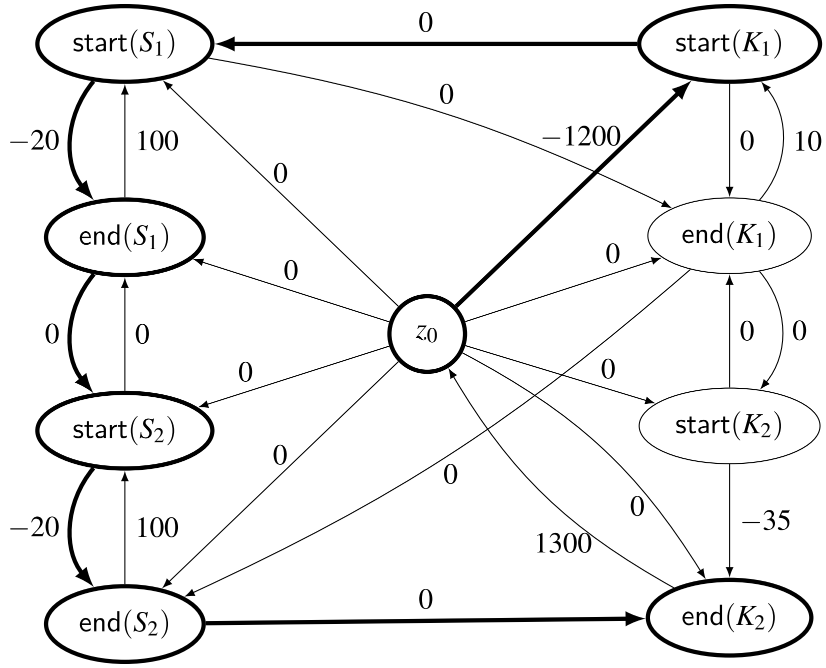


Figure 11: Graph representation of the constraint from Figure 10. The bold path shows the shortest path between z_0 and $\mathsf{end}(K_2)$, which allows us to infer that the reign of K_2 ends after 1240 (because $z_0 - \mathsf{end}(K_2) \leq -1240$). This bold part corresponds to the trace given in Figure 6c.

Graph representation of the constraints. Once the constraint is normalised, we can easily represent it as a *directed weighted graph* (that we henceforth simply call a *graph*). Intuitively, a graph is a diagram (see Figure 11) made up of two kinds of elements: *nodes*, represented as ellipses; and *edges*, which are arrows from one node to another, bearing a label called the *weight* of the edge. In our case, the graph corresponding to a normalised constraint contains:

1. one node per boundary in B; and
2. for each atomic constraint $b_1 - b_2 \leq k$, an edge from b_1 to b_2 with weight k.

In the case of the ChronoLand example, the graph corresponding to the normalised constraint of Figure 10 is given in Figure 11.

3.3 Algorithms for tightening and consistency check

Let us now explain how the graph representing a given Chronological Network helps us solve the tightening and consistency check problems defined in Section 2.2.

Tightening. Let us consider again the ChronoLand example (which is consistent), and let us focus on the inputs regarding king K_2 (see Figure 6a). We know that the reign of K_2: (i) lasts at least 35 years; and (ii) ends before 1300. Clearly, these two pieces of information allow us to infer that the reign of K_2 must start *before* 1265. Let us now explain how we can extract this information from the graph (Figure 11) corresponding to the ChronoLand Chronological Network. First, in terms of constraints, we can express the inputs as:

$$\mathsf{end}(K_2) - \mathsf{start}(K_2) \geq 35 \textbf{ and } \mathsf{end}(K_2) \leq 1300,$$

which is equivalent to the normalised constraint:

$$\mathsf{start}(K_2) - \mathsf{end}(K_2) \leq -35 \textbf{ and } \mathsf{end}(K_2) - z_0 \leq 1300.$$

Now, observe that, *if two inequalities $A_1 \leq B_1$ and $A_2 \leq B_2$ hold, then $A_1 + A_2 \leq B_1 + B_2$ holds as well*. We can thus sum inequalities and deduce new information from this sum. In our example, this sum is shown in Figure 12 (bottom left), where we deduce that $\mathsf{start}(K_2) - \mathsf{end}(K_2) + \mathsf{end}(K_2) - z_0 \leq -35 + 1300$, i.e. $\mathsf{start}(K_2) - z_0 \leq 1265$, or, in words, that the start of the reign of K_2 must occur before 1265.

Now, let us consider the graph equivalent to this constraint: it is displayed in Figure 12, bottom right (considering the solid edges only for the moment). The combination of the two atomic constraints $\mathsf{start}(K_2) - \mathsf{end}(K_2) \leq -35$ and $\mathsf{end}(K_2) - z_0 \leq 1300$ corresponds to a *path* visiting successively nodes $\mathsf{start}(K_2)$, $\mathsf{end}(K_2)$ and finally z_0 in the graph. Let us defined the *weight* of a path as the sum of the weights of the traversed edges. Then, the weight of the $\mathsf{start}(K_2) \to \mathsf{end}(K_2) \to z_0$ path is $-35 + 1300 = 1265$, which is exactly the information that we have obtained by combining the atomic constraints. We can thus modify our graph by adding a new edge from $\mathsf{start}(K_2)$ (first node of the path) to z_0 (last node of the path) with weight 1265. We can thus see that *each path in the graph from some boundary b_1 to some boundary b_2 corresponds to a combination of atomic constraints (from the inputs), involving all the boundaries traversed by the path. Such a path can thus be used to infer an upper bound on $b_1 - b_2$*. Then, let us assume for example that there exists *another path* from $\mathsf{start}(K_2)$ to z_0 which is *shorter* (for example, 1000 instead of 1265). Since this path corresponds to another combination of atomic constraints from the inputs, it provides a *tighter bound* on $\mathsf{start}(K_2) - z_0$. Thus, looking for tighter bounds amounts to looking for shorter paths between given pairs of nodes.

The main takeaway message of this example is that there is a *correspondence* between paths in the graph and sets of atomic constraints. More precisely:

1. Every time we have a path with weight w from b_1 to b_2, this path corresponds to a set of atomic constraints that sum up to $b_1 - b_2 \leq w$.

2. Symmetrically, a set of input atomic constraints that sum up to $b_1 - b_2 \leq w$ means there is a path of weight w from b_1 to b_2 in the graph. The information we can extract from the graph is thus *complete*: all the information given by the atomic constraints is indeed present in the graph, and the most precise information on $b_1 - b_2$ can be obtained by looking for *the shortest path between b_1 and b_2*.

Thus, *tightening* a Chronological Network amounts to *finding all the shortest paths between each pairs of nodes (or boundaries)* in the corresponding graph, which is a problem that has been thoroughly studied in computer science [MAR$^+$17] and for which many efficient algorithms exist. This outlines our procedure for tightening:

Theorem 1 (Adapted from [Dil89])**.** *Let C be a Chronological Network (with a set of boundaries B), and let G be the graph obtained from the normalised constraint extracted from C. Let b_1 and b_2 be two boundaries from B. Then, the tightest atomic constraint on $b_1 - b_2$ that one can infer from C is:*

$$b_1 - b_2 \leq SP(b_1, b_2),$$

where $SP(b_1, b_2)$ is the weight of the shortest path from node b_1 to node b_2 in the graph G.

As an example, the set of all tightest atomic constraints that can be extracted from the ChronoLand inputs (Figure 6) is shown in Figure 13 as a matrix. We have chosen a matrix representation here as drawing the graph will all edges, including the ones computed during tightening, would make the figure unreadable. For each pair of boundaries b_1 and b_2, the value $SP(b_1, b_2)$ is presented in row b_1, column b_2 of the matrix.

$$\mathsf{start}(K_2) - \mathsf{end}(K_2) \leq -35 \textbf{ and } \mathsf{end}(K_2) - z_0 \leq 1300$$

$$\begin{array}{rrcr} & \mathsf{start}(K_2) - \mathsf{end}(K_2) & \leq & -35 \\ + & \mathsf{end}(K2) - z_0 & \leq & 1300 \\ \hline & \mathsf{start}(K_2) - z_0 & \leq & 1265 \end{array}$$

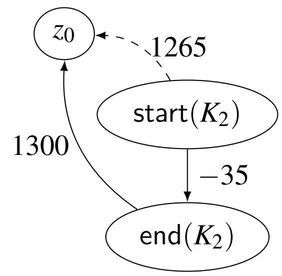


Figure 12: From constraints to graphs. The constraints $\mathsf{start}(K_2) - \mathsf{end}(K_2) \leq -35$ **and** $\mathsf{end}(K_2) - z_0 \leq 1300$ are translated into the given graph (solid edges only), which is an excerpt of the graph in Figure 11. This information allows us to deduce that $\mathsf{start}(K_2) \leq 1265$, where the value 1265 is the weight $-35 + 1300$ of the shortest path from $\mathsf{start}(K_2)$ to z_0. We can reflect this new piece of information in the graph by adding a direct (dashed) edge from $\mathsf{start}(K_2)$ to z_0, with weight 1265.

	z_0	start(S_1)	end(S_1)	start(S_2)	end(S_2)	start(K_1)	end(K_1)	start(K_2)	end(K_2)
z_0	0	−1200	−1220	−1220	−1240	−1200	−1200	−1200	−1240
start($S1$)	1260	0	−20	−20	−40	10	0	0	−40
end($S1$)	1280	80	0	0	−20	80	80	80	−20
start($S2$)	1280	80	0	0	−20	80	80	80	−20
end($S2$)	1300	100	80	80	0	100	100	100	0
start($K1$)	1260	0	−20	−20	−40	0	0	0	−40
end($K1$)	1265	10	−10	−10	−30	10	0	0	−35
start($K2$)	1265	10	−10	−10	−30	10	0	0	−35
end($K2$)	1300	100	80	80	60	100	100	100	0

Figure 13: The matrix SP of all-pairs shortest paths for the ChronoLand example. The entry $SP(b_1, b_2)$ in row b_1 column b_2 gives the weight of the shortest path from b_1 to b_2, i.e. the constraint $b_1 - b_2 \leq SP(b_1, b_2)$, which is the tightest atomic constrain on $b_1 - b_2$ that one can infer from the given inputs.

The most relevant results extracted from this matrix are shown in the computed ranges of Figure 6b. For example, the value -1240 in row z_0, column $\mathsf{end}(K_2)$ indicates that $z_0 - \mathsf{end}(K_2) \leq -1240$, i.e. $\mathsf{end}(K_2) \geq 1240$, or, in words, that the reign of K_2 must end after 1240 (as discussed in Section 2.2.2). This is the tightest lower bound on $\mathsf{end}(K_2)$ that we can infer from the inputs. It has been obtained thanks to the path highlighted in bold in Figure 11, which is the shortest path from z_0 to $\mathsf{end}(K_2)$, and also corresponds to the trace from Figure 6c. Observe, however that the matrix contains more information than what is presented in Figure 6b. For example, it tells us that $\mathsf{start}(K_2) - \mathsf{end}(S_2) \leq -30$, i.e. that $\mathsf{end}(S_2) \geq \mathsf{start}(K_2) + 30$, or, in words, that the end of stratum S_2 occurs at least 30 years after the start of K_2's reign.

Consistency check. In the discussion so far, we have assumed that the Chronological Network under consideration is consistent. We explain how to check this. Let us consider again the example of non-consistent network from Figure 7, and let us understand why it is non-consistent using the techniques we have discussed so far. The atomic constraints that yield non-consistency are as follows (as discussed in Section 2.2.1):

$$\begin{array}{lr} \mathsf{start}(K_1) - \mathsf{start}(S_1) \leq 0 & (S_1 \text{ starts during } K_1) \\ \textbf{and } \mathsf{start}(S_1) - \mathsf{end}(S_1) \leq -20 \textbf{ and } \mathsf{end}(S_1) - \mathsf{start}(S_2) \leq 0 \textbf{ and } \mathsf{start}(S_2) - \mathsf{end}(S_2) \leq -20 & \text{(Strata duration)} \\ \textbf{and } \mathsf{end}(S_2) - \mathsf{end}(K_2) \leq 0 & (S_2 \text{ ends during } K_2) \\ \textbf{and } \mathsf{end}(K_2) - \mathsf{start}(K_2) \leq 25 \textbf{ and } \mathsf{start}(S_2) - \mathsf{end}(K_1) \leq 0 \textbf{ and } \mathsf{end}(K_1) - \mathsf{start}(K_1) \leq 10 & \text{(Dynasty duration).} \end{array}$$

Summing all these atomic constraints (as we did previously) yields the conclusion that $0 \leq -5$, a clear impossibility. This is witnessed by a cycle (i.e. a path starting and ending in the same node) of *negative weight* -5 in the corresponding graph:

$$\mathsf{start}(K_1) \xrightarrow{0} \mathsf{start}(S_1) \xrightarrow{-20} \mathsf{end}(S_1) \xrightarrow{0} \mathsf{start}(S_2) \xrightarrow{-20} \mathsf{end}(S_2) \xrightarrow{0} \mathsf{end}(K_2) \xrightarrow{25} \mathsf{start}(K2) \xrightarrow{0} \mathsf{end}(K_1) \xrightarrow{10} \mathsf{start}(K_1).$$

In the ChronoLand example, which is consistent, the graph contains no negative cycle, see Figure 11.

These examples highlight the technique to check consistency, as introduced by Shostak [Sho81]: a constraint is consistent *if and only if* its corresponding graphs has no negative cycle.

Theorem 2 (Adapted from [Sho81])**.** *Let C be a Chronological Network, and let G be the graph corresponding to the constraint encoding C. Then, C is consistent if and only if G contains no cycle of negative weight.*

Algorithms for all-pairs shortest paths. Now that we have shown that we can solve both the consistency and the tightening problems by computing the shortest paths between all possible pairs of nodes in a graph (*all-pairs shortest paths* for short), let us briefly discuss algorithms to do so. First, note that most algorithms to compute all-pairs shortest paths include a test to detect negative cycle. That is, the output of such algorithms is either:

1. "fail", when the graph contains a cycle of negative weight. This cycle can then be used to provide a trace of non-consistency, i.e. a set of constraints that yield a contradiction.

2. or the length of the shortest paths between all pairs of nodes, given, for instance, under the form of a matrix as in Figure 13. The algorithm also returns the actual shortest paths, which can be used to obtain *traces* (Figure 6c) for the new computed results.

Many efficient algorithms to compute all-pairs shortest paths exists, see [MAR+17] for a survey. Here, "efficient" means that these algorithms run in *polynomial time* with respect to the size (number of boundaries and number of atomic constraints) of the Chronological Network, as opposed to exhaustive search

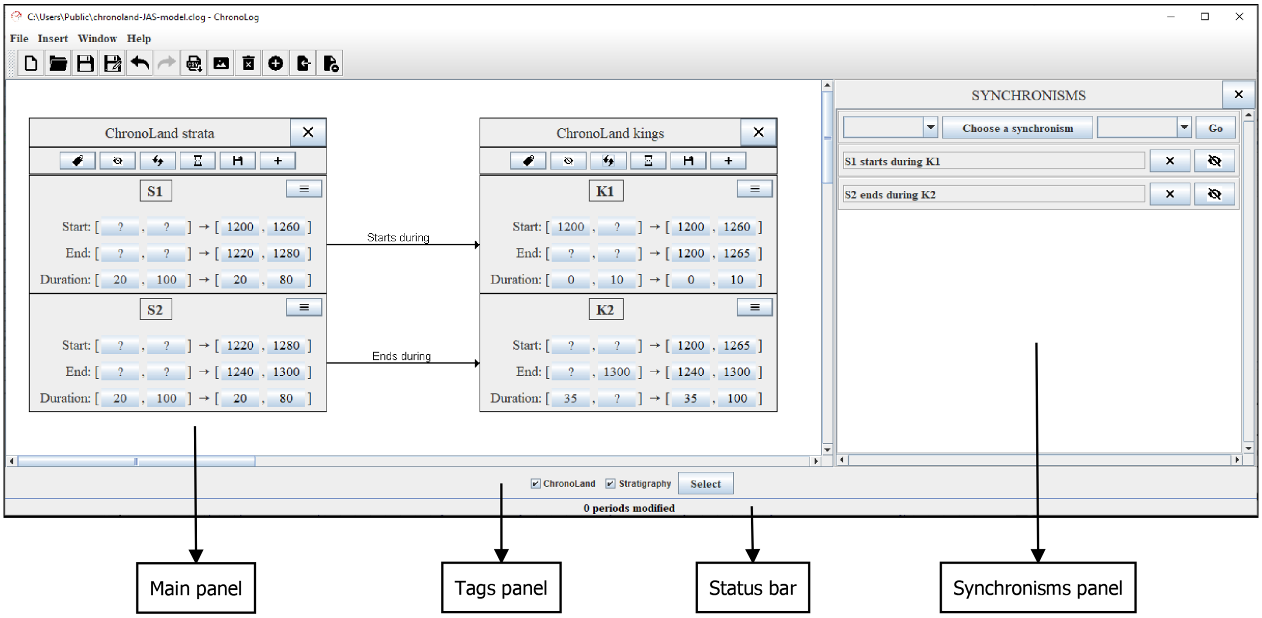


Figure 14: The CHRONOLOG software display, showing the ChronoLand example.

algorithms, which run in exponential time and are often impractical. For example, in our setting, the classical Floyd-Warshall algorithm [Flo62] runs in time proportional to $|B|^3$, where $|B|$ is the number of boundaries in the Chronological Network A more efficient algorithm is Johnson's algorithm [Joh77], which runs in time proportional to $|B|^2 \log(|B|) + |B||A|$, where $|A|$ is the number of atomic constraints in the Chronological Network. In practice, these measures of efficiency indicate that it is possible to handle Chronological Networks with thousands of boundaries and atomic constraints (see Section 4 below). The next section presents our software implementation of the algorithms presented here.

4 The CHRONOLOG software

CHRONOLOG is a software utility that allows users to create Chronological Networks (as defined in Section 2) and to modify them. The software automatically tests the consistency of the network, and computes the tightened ranges of each start date, end date, and duration. Figure 14 shows a general overview of the CHRONOLOG interface, consisting of a main panel depicting the Chronological Network (ChronoLand in this case), a "Synchronisms" panel displaying all the Chronological Relations of the network, a "Tags" Panel showing the tags associated to each Sequence (see Section 2.3.2 above), and a status bar (more on this below). The main aspects of CHRONOLOG are briefly presented below.

4.1 Representation of the Network

Time-periods. Figure 15 provides an example of a Time-period as represented in CHRONOLOG. The Time-period features four lines, representing the Time-period's name, duration, start date, and end date. The latter three lines have a common structure: the input range, an arrow, the computed range. Note that the chronological data are always represented by ranges: known dates/durations are represented by ranges with equal lower and higher bound, unknown lower/higher bounds are represented by question marks. The bounds appear on clickable buttons. Clicking on a bound launches a simple dialog enabling to enter a custom value, or the "Unknown" value, for the bound (see Figure 15). Dates B.C.E. are input with a minus sign (thus -1200

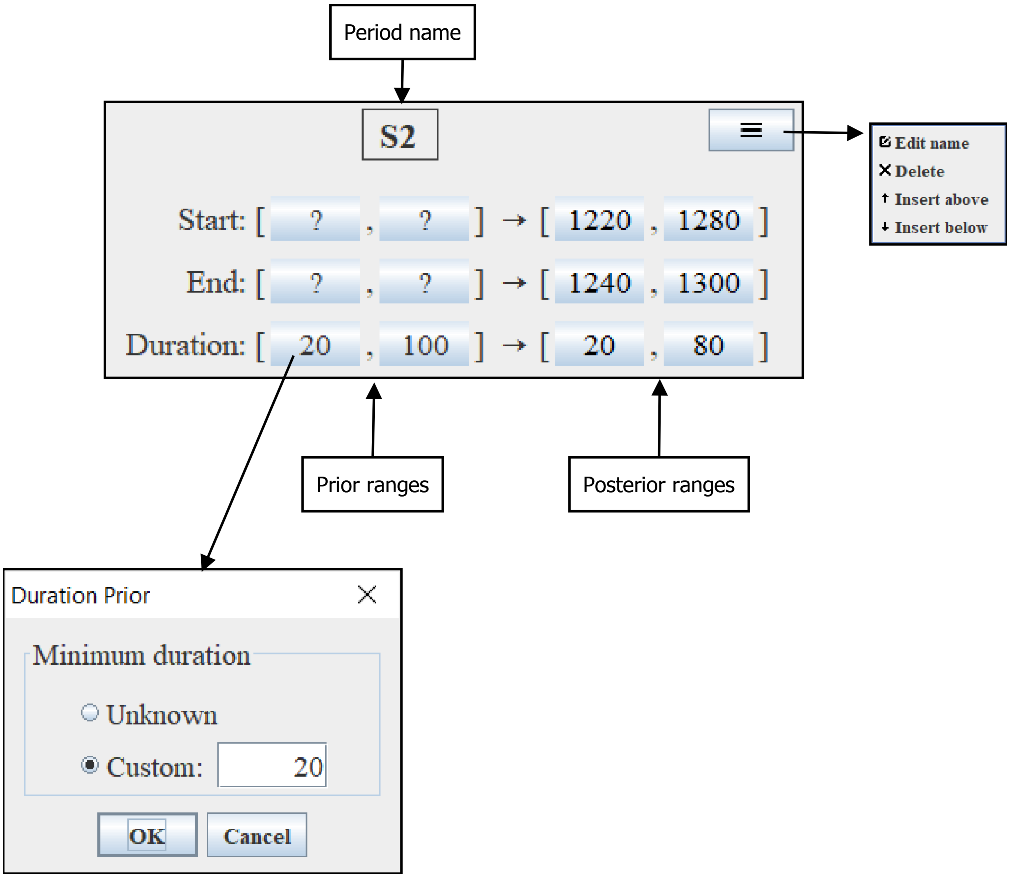


Figure 15: Representation of a Time-period in CHRONOLOG.

for 1200 B.C.E.). The button in the upper right corner of a Time-period allows to rename the Time-period, delete it, or insert a new Time-period in the same Sequence.

Sequences. Sequences are represented by Time-periods stacked on top of each other, with time flowing from above to below (see Figure 16). The top row of the Sequence contains the Sequence name, followed by a button to delete the Sequence. The second row features a set of buttons allowing diverse actions: tagging the Sequence (see below), hiding it, reloading the default values of each bound, setting duration bounds for each Time-period at once, saving the Sequence to a file (see below), and adding a Time-period at the end of the Sequence. Sequences can be added to the current network from the menu bar, either by selecting one from the CHRONOLOG library ("Insert → Insert from library") or by creating a new one interactively ("Insert → New sequence").

Chronological Relations. By language abuse, and following common practice, we refer to all Chronological Relations in CHRONOLOG as "synchronisms". The addition of a Chronological Relation to the network is done through the "Synchronisms" panel on the right side of the Chronolog window. This panel features the list of current Chronological Relations (see Figure 17) and allows to add a new Relation by choosing two Time-periods in the associated combo boxes and clicking on "Choose a synchronism" to choose the type of Relation. This displays a dialog featuring all the types of Chronological Relations defined in Section 2 above, with both a graphical depiction and the formal definition of the Relation. The Relation is then added

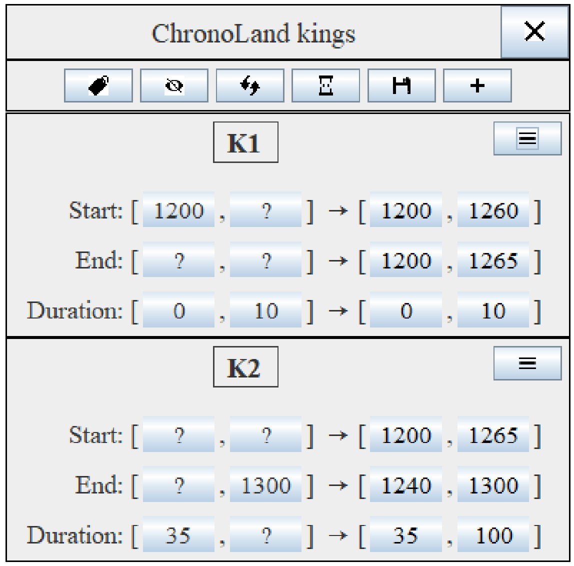


Figure 16: Representation of a Sequence in CHRONOLOG.

to the network via the “Go” Button. Relations can also be deleted or temporarily hidden from the network via the “Delete” and “Hide/Show” buttons. Relations can also be added directly between two Time-periods by joining the two Time-periods with the mouse, which will will automatically draw a line between them and display the Chronological Relations choice dialog. Finally, a Relation can be clicked on, which displays a dialog allowing to modify, hide or delete it.

Input/output. The current network can be saved to a file (“File → Save”) and later reloaded into CHRONOLOG (“File → Open”) as a new model. Furthermore, components of a network (i.e. sets of Sequences and Chronological Relations) can also be loaded from a file or from the CHRONOLOG library, in order to add them to the current model (“Insert → Insert from file” or “Insert → Insert from library”). These files have a JSON format, and a “.clog” extension. The chronology computed by CHRONOLOG can also be exported to a CSV (comma-separated-values) file (“File → Export (CSV)”), or to an image file representing the whole chronological network (“File → Export as image”).

4.2 Main functionalities

Consistency. At each modification of the network (removal/addition of a bound or a Chronological Relation), CHRONOLOG automatically checks the consistency of the network. In case of a non-consistent network, an error message is displayed, as well as a trace providing a list of conflicting constraints.

Tightening. At each modification of the network (removal/addition of a bound or a Chronological Relation), if the consistency check has been successful, the tightening procedure is launched automatically, and the computed bounds are updated for each Time-period. The bounds that now have a different value than before are shown in red, and the number of modified Time-periods is displayed in the status bar. Figure 18 provides the example of ChronoLand with an updated input of maximum 70 years for K_2 (see Section 2.2.1). The updated computed strata durations (upper bound of 60 instead of 80) are shown in red, and the status bar indicates “3 periods modified” (including the updated reign of K_2).

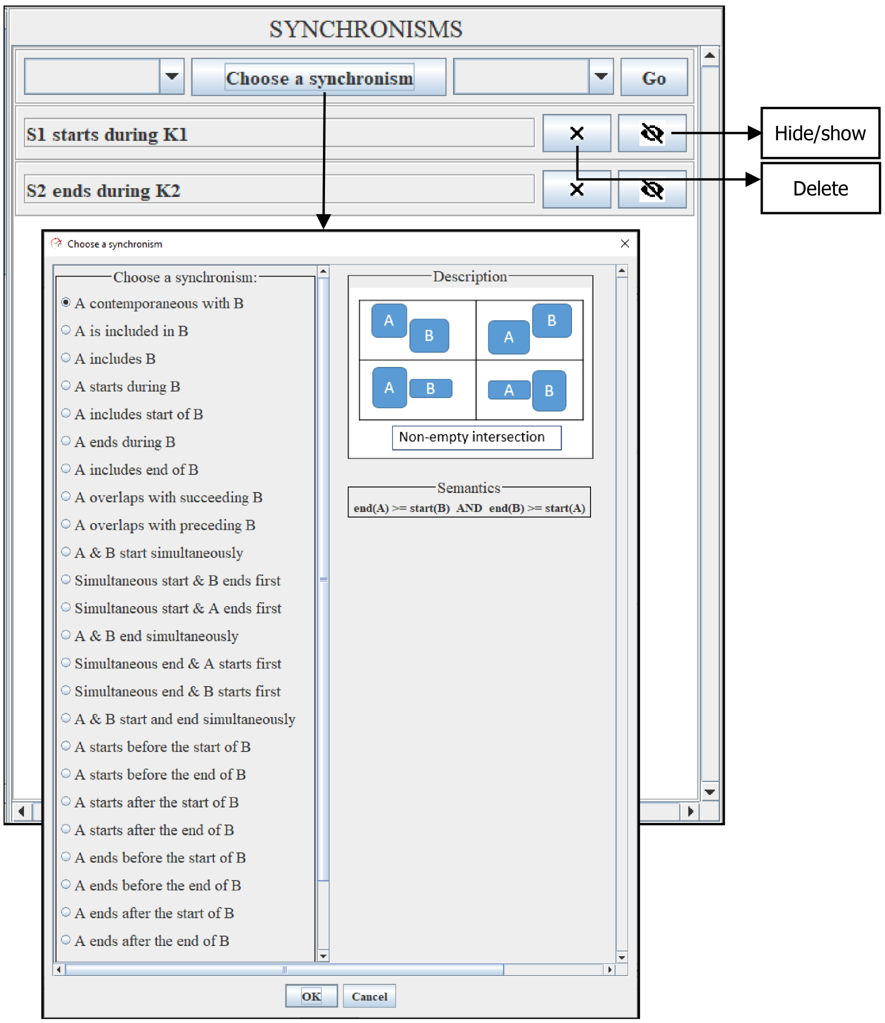


Figure 17: Choice of a Chronological Relation in CHRONOLOG.

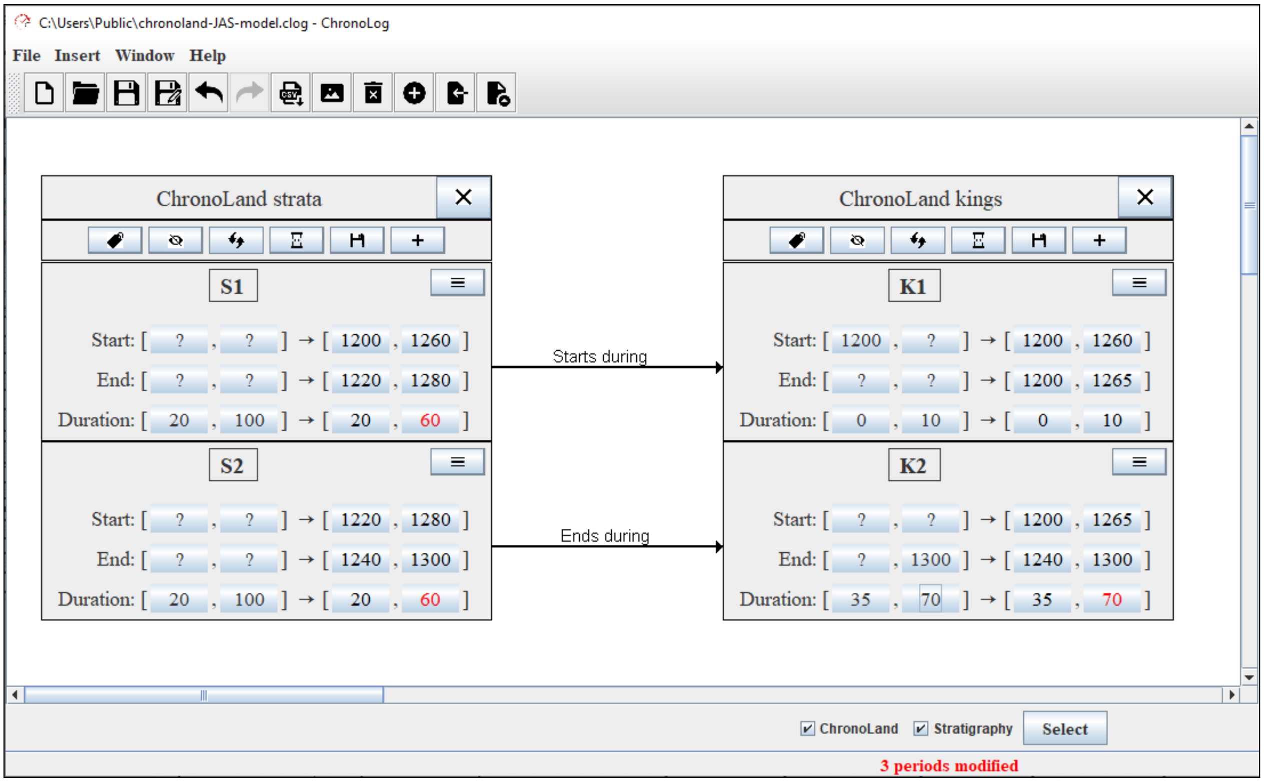


Figure 18: Result of re-tightening the network after updating the maximum duration of K_2 to 70 years.

Traces. Each computed bound is clickable, in which case the full trace for the bound is displayed. Figure 19 provides the full trace for the 1240 lower bound for the end date of K_2. This trace conforms to the one provided in Section 2.2.2.

Tagging. In addition to the above-described features, ChronoLog also implements a powerful *tagging* mechanism that allows to associate several keywords (or *tags*) to each Sequence, and to activate or deactivate all Sequences bearing a given tag at any moment. This allows the user to consider, in a single Chronological Network, several sources of prior knowledge and to test the potential implications of these different hypotheses. This is realised through the "Tags" panel, located at the bottom of the ChronoLog window, where each tag can be checked or unchecked, resulting in hiding/showing the associated Sequences and rerunning the consistency check and tightening process.

4.3 Discussion

4.3.1 ChronoLog and radiocarbon dating

Radiocarbon measurement are the main source of absolute dating used by archaeologists today. We discuss here how to incorporate radiocarbon data into ChronoLog models.

The laboratory results of radiocarbon measurements need calibration to be expressed as absolute calendar dates. Since ChronoLog deals with calendar dates, its input should consist of calibrated radiocarbon readings. The calibration can be done using standard tools like OxCal ([Ram95]). Following the radiocarbon procedure, the radiocarbon result of a measured sample is expressed as a full probability distribution (usually

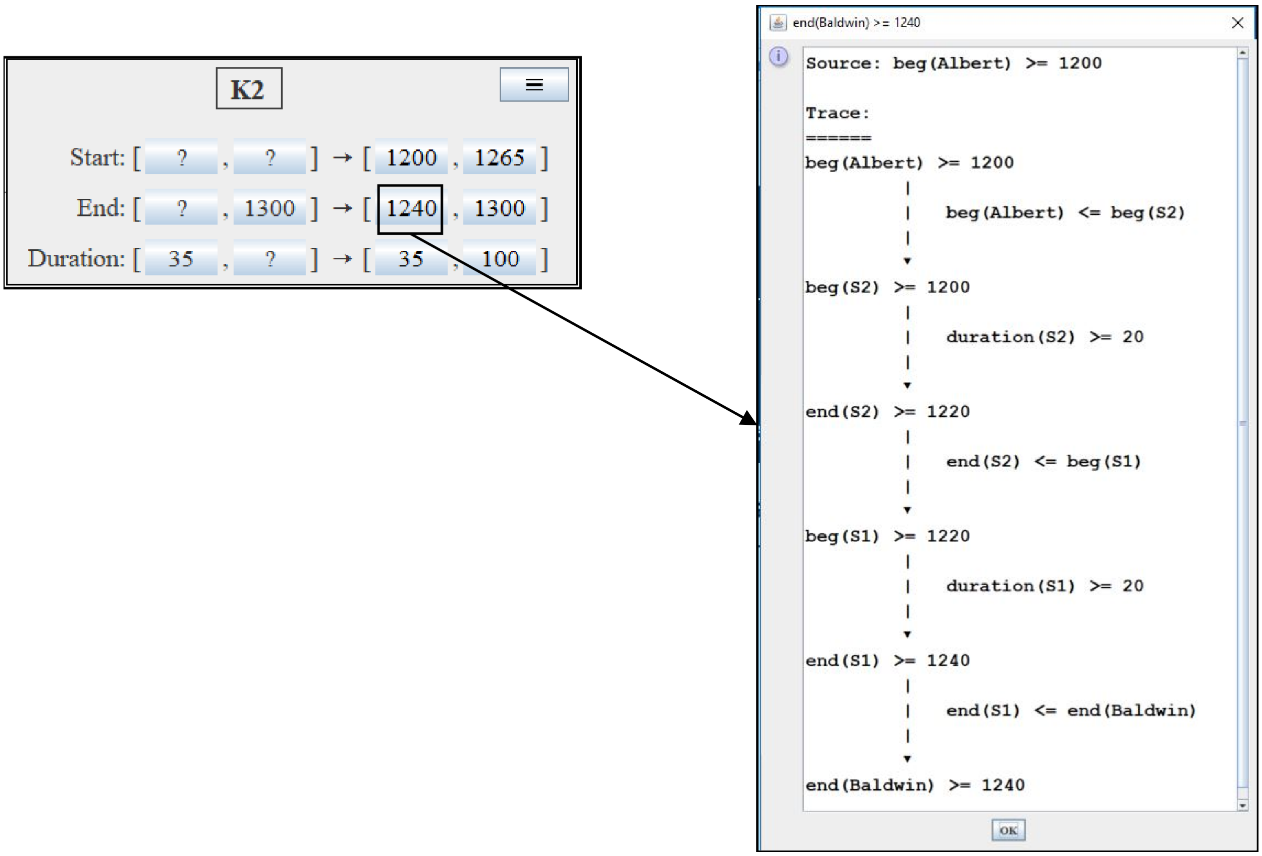


Figure 19: Trace for the 1240 lower bound for the end date of K_2

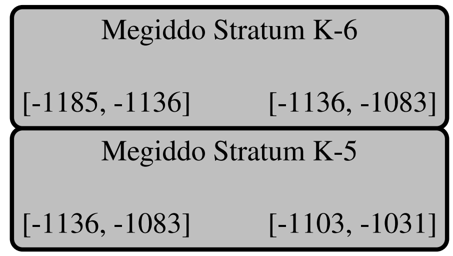


Figure 20: Archaeological strata with radiocarbon ranges (without duration constraints): Strata K-6 and K-5 at Megiddo, Israel ([FAMP17], p. 274).

not a normal distribution) or by the 68% or 95% confidence level limits for the date. The calibrated radiocarbon results for the boundaries of strata and archaeological periods consist of TPQs, TAQs and date-ranges. These data can be inserted *as is* into CHRONOLOG (see Figure 20). Yet, the CHRONOLOG consistency check and tightening operations are not probabilistic (see Section 2.2), hence these bounds are considered as deterministic input, without an associated probability. The final computed ranges should be seen as an "if-result", that is: "*if* the radiocarbon bounds are correct, as well as the other constraints of the model, *then* the computed ranges are the tightest possible ones satisfying all the input constraints".

Often, Bayesian modelling is used to determine the dates of samples, using priors that take into account historical constraints, order of layers, synchronisation of time between strata, etc... Such constraints can be modelled in CHRONOLOG as well. In case such modelled radiocarbon dates are used, it is mandatory to make sure that the CHRONOLOG constraints do not contradict any of the Bayesian prior assumptions. A safer approach would be to include only unmodelled radiocarbon dates (68% or 94% confidence level limits) into CHRONOLOG. A possible exception to this rule could be the inclusion of some fixed identical priors in both OxCal and CHRONOLOG, like succession of strata, which are not meant to be changed in the CHRONOLOG model.

4.3.2 Notes

Units. CHRONOLOG currently uses year-precision, meaning that only whole years can be encoded (no months, days or fractional years). The algorithms presented in this paper can also be used to attain day-precision (with fractional years and support for leap years), by using the day as a the computational unit in the algorithms, a feature we leave for future work.

Efficiency. CHRONOLOG has been shown to run fast even on large Chronological Networks. An experiment on a large network, featuring over 75 Time-periods, 100 Chronological Relations and 100 duration/date constraints, had the consistency check and tightening operations run in less than one second, on a simple laptop computer running an Intel Core M-5Y10c processor at 0.80 GHz.

Availability. CHRONOLOG is freely available for non-commercial use. It has been written using the Java programming language, and runs on any platform (Windows, MacOS, Linux, ...) having a Java installation (`https://www.java.com/en/download/`). The base distribution of CHRONOLOG includes a library of standard chronological Sequences for pharaonic Egypt and the Ancient Near East. A webpage for CHRONOLOG is available at `http://chrono.ulb.be/`[2], from which the software can be downloaded at no cost.

[2] The CHRONOLOG website is password-protected for the duration of the reviewing process. The password is "JAS" (without the quotes).

If you use CHRONOLOG, or publish chronological results obtained with the help of CHRONOLOG, please include a link to the utility's web page, and a reference to this article.

5 Case study: Egyptian 26th dynasty

We present a case study related to the Egyptian 26th dynasty (Table 6). We choose a well-known chronology to demonstrate how CHRONOLOG can be used to reconstruct a chronology from primary data, and to assess the impact of specific data on the chronology.

King	Dates	Duration
Psammetichus I	664-610	54 years
Necho II	610-595	15 years
Psammetichus II	595-589	6 years
Apries	589-570	19 years
Amasis	570-526	44 years
Psammetichus III	526-525	1 year

Table 6: Standard chronology of the Egyptian 26th dynasty ([Kit00, p. 50], [HKW06, p. 494]).

5.1 Data

Egyptologists have established the chronology of Dyn. 26 based on the historical fixed point of 525 B.C.E. for the dynasty's end, combined with a reconstruction of the reign durations ([HKW06], p. 267-268)[3]. These durations have been deduced from a combination of sources:

1. **Highest attested regnal years.** Ancient Egyptian dates start with the regnal year of the current king (starting at Year 1), followed by a month and a day. Only for Psammetichus II does an ancient inscription provide the exact reign-length. For other kings, the highest attested year provides only a minimum reign-length. For example, Amasis's highest attested year 44 implies a reign of at least 43 full years.

2. **Funerary stelae.** Funerary stelae[4] sometimes mention the deceased's birth date, death date, and lifespan. When the birth and death occurred during different reigns, this information can help fix the duration of the reigns. Egyptologists used this technique to deduce the precise reign-lengths of Psammetichus I, Necho II, and Apries.

3. **Herodotus and Manetho.** The ancient historians Herodotus and Manetho[5] provide the full sequence of Dyn. 26 kings, as well as alleged reign-lengths. Egyptologists have relied on this source for fixing the reign-lengths of Amasis and Psammetichus III.

Table 7 summarises all the relevant data (see Appendix for full details).

[3] The date of 525 B.C.E. for the end of Dyn. 26, marked by the Persian invasion of Egypt, is the prevalent view. See [Dep96] for a slightly earlier dating (527-525 B.C.E.) and [Bec02] for a rebuttal of this view.

[4] The relevant stelae (see Table 7) concern individuals and *Apis bulls*, sacred bulls mummified and buried with full honors, including funerary stelae.

[5] Egyptologists use here Africanus's version of Manetho's *epitome* rather than Eusebius's. Note also that the latter does not feature King Psammetichus III ([Man40, p. 171-173]).

Name	Source	CHRONOLOG constraint
Psammetichus I	Highest attested year	At least 54 years
Necho II		At least 15 years
Psammetichus II		6 years
Apries		At least 19 years
Amasis		At least 43 years
Amasis	Herodotus & Manetho	44 years
Psammetichus III		0-1 years
Apis bull III	Funerary stelae	Starts 52 years after the start of Psammetichus I Ends 15 years after the start of Necho II Duration = 17 years
Apis bull IV		Starts 15 years after the start of Necho II Ends 11 years after the start of Apries Duration = 17 years
Priest Psammetichus		Starts 0 years after the start of Necho II Ends 26 years after the start of Amasis Duration = 66 years
Other Psammetichus		Starts 2 years after the start of Necho II Ends 34 years after the start of Amasis Duration = 72 years
Besmaut		Starts 17 years after the start of Psammetichus I Ends 22 years after the start of Amasis Duration = 99-100 years

Table 7: Set of chronological constraints used to reconstruct the chronology of the Egyptian 26th dynasty (see Appendix for full details).

5.2 Reconstructing the chronology

We built a CHRONOLOG model containing all the above-described constraints (see Appendix, Figure 21). The results are shown in Table 8. CHRONOLOG computed a precise duration for each king, except Psammetichus III (set to 0-1 years). The resulting chronology has a one-year uncertainty, with the dynasty beginning in 664 or 663 B.C.E. The lower date (663 B.C.E.) was the standard date for the start of the dynasty until the late 1950s[6] (see for example [Kie53], p. 157). It was later abandoned in favour of 664 B.C.E. based on an astronomical argument by Parker ([Par57]). This higher date implies a one-year duration[7] for Psammetichus III. Adding this constraint to our model now provides the current standard chronology for the dynasty (see Appendix, Figure 22).

[6]The then-standard date of 663 B.C.E. was based on slightly different data: a 43-year reign of Amasis and a one-year reign of Psammetichus III (see [Kie53], p. 156-157). The latter was based on papyri allegedly mentionning a Year 2 of Psammetichus III, but now reattributed to the later king Psammetichus IV ([CU80], [Vle91, p. 3-4]).

[7]Parker used an astronomical argument to show that Amasis's reign started in 570 B.C.E. rather than 569 B.C.E., resulting in 664 B.C.E. for the start of the dynasty. He worked on the basis of 43-44 years for Amasis and one year for Psammetichus III. The latter duration is now outdated (see note 6) but Parker's astronomical argument still applies here, since our framework (44 years for Amasis and 0-1 years for Psammetichus III) implies the same uncertainty as before (570-569 B.C.E.) for the start of Amasis.

King	CHRONOLOG result		
	Start	**End**	**Duration**
Psammetichus I	664-663	610-609	54 y.
Necho II	610-609	595-594	15 y.
Psammetichus II	595-594	589-588	6 y.
Apries	589-588	570-569	19 y.
Amasis	570-569	526-525	44 y.
Psammetichus III	526-525	525	0-1 y.

Table 8: Chronology computed by CHRONOLOG (see Appendix, Figure 21). All the reign-lengths have been precisely computed, except Pammetichus III (0-1 years). The resulting chronology floats by only one year, with the dynasty beginning in 664 or 663 B.C.E.

5.3 Testing hypotheses

CHRONOLOG allows to test the precise impact of each piece of data. For example, which funerary stela determines the duration of which king? Are all stelae truly necessary? If not, which ones are indispensable? CHRONOLOG can easily answer such questions by excluding specific data from the model (see Section 4.1). A simple experiment yields the following insights:

1. Apis Bull III is indispensable for establishing the precise duration of Psammetichus I. That is, hiding Bull III makes us loose the precise 54-year duration of the king.

2. One of the two stelae among the Priest Psammetich and the other Psammetich is necessary in order to fix the duration of Apries. That is, removing one of them from the model has no effect, but removing both makes us loose the precise 19-year duration of Apries.

3. The complete chronology can be reconstructed using only 2 out of the 5 funerary stelae, namely Apis Bull III and the Priest Psammetichus (see Appendix, Figure 23). In other words, the other stelae offer only redundant information (but are nevertheless useful for providing greater robustness to the model).

4. Hiding the contributions of Herodotus and Manetho makes us loose the precise 44-year duration for Amasis and strips us of a lower bound for the start date of the dynasty (see Appendix, Figure 24). In other words, the funerary stelae are not enough for setting Amasis's precise duration.

5.4 Discussion

The chronology reconstructed here with CHRONOLOG was historically obtained by manual computation (see [Gar45, p. 17-18], [Kie53, p. 153-157], and [HKW06, p. 466] for concrete examples). Also, the impact of specific chronological data was formerly only manually assessed (see for example [Kie53, p. 155-156]). CHRONOLOG enabled us to perform both kinds of operations in a simpler and automated way. Note that the example of Dyn. 26 is small and hence still manually computable. Yet, it illustrates the full potential of CHRONOLOG for building and assessing chronologies, especially for larger data sets, where manual treatment would be impracticable.

It is also interesting to notice the coherence of the raw Egyptological data: a change of dates or duration of even one year in most of our funerary stelae would render the model inconsistent. This pleads in favour of the trustworthiness of the chronological information provided by these stelae. CHRONOLOG can thus also be used to check the consistence of epigraphic sources, and to detect any incorrect chronological claim found therein. The full Dyn. 26 model is available on the ChronoLog web site (`http://chrono.ulb.be/`), enabling readers to run the above-described experiments by themselves.

6 Conclusion

This paper introduced the notion of *Chronological Network*, a powerful formalism for representing chronological data organised as a set of Sequences, composed of *Time-periods* sharing *Chronological Relations* with each other. The simplest such relation is that of *contemporaneity*, where two Time-periods have at least one unit of time in common. Our model allows to specify many other types of Chronological Relations, both synchronic and asynchronic (see Table 1, 2, 3 and 4). The Chronological Networks model further allows one to specify constraints on the start date, end date, and duration of the Time-periods, expressed as exact values, bounds, or ranges. The model enables archaeologists to present their data and ground hypotheses in a clear, rigorous and complete fashion.

Moreover, we have shown how to formally and automatically analyse Chronological Networks, by defining two basic and important operations, namely *consistency check* and *tightening*. The consistency check operation checks whether the model features a contradiction, and the tightening operation allows one to obtain the most precise possible chronological estimate for each boundary and duration, expressed as a range.

We have shown how a chronological network can be encoded as a mathematical object called a *directed weighted graph*, and how graph algorithms can be used to solve the tightening and consistency check problems efficiently. This approach builds an important link between the field of archaeological chronology and the field of computer ccience, where the sub-fields of artificial intelligence, combinatorial optimisation and formal methods have developed a rich set of models and algorithms for the study of time. The applicability of such tools for archaeological problems has still been insufficiently addressed, and this paper is intended as a step in this direction.

We have implemented our techniques in a tool called ChronoLog, which is freely available to the archaeological community. This tool implements the tightening and consistency check operations, and thus allows one to compute the most precise chronological information that can be inferred from a given Chronological Network. To the best of our knowledge, no efficient and complete model or software solution to this end has been introduced before.

Finally, we have applied our methodology to a practical-case study, showing how the absolute chronology of the Egyptian 26th dynasty can be reconstructed from primary data using ChronoLog, and how the tool can be used to assess the precise impact of each piece of input data.

In future works, we intend to investigate other kinds of information that could be automatically extracted from Chronological Networks. For example, one could be interested in discovering automatically all the constraints and relations that have no impact on the final tightened ranges and to automatically remove them from the network in order to keep a minimal "core" set of chronological constraints. Another interesting problem is the definition of a *robustness* index, which expresses the strength of a given bound. This index can be defined as a function of the number of different paths in the network that ensure the given bound. The computation of such robustness indexes can add a significant quantitative aspect to the results, enabling to differentiate between "stronger" and "weaker" results. A third important application would to be to query the model directly in order to ask which precise Chronological Relations hold true between two given Time-periods. A final interesting trail would be to investigate how our deterministic approach could be be combined with probabilistic knowledge, in order to add a further layer of uncertainty on the data, in addition to the one currently represented by deterministic ranges. We intend to address these questions in future papers, both within our theoretical framework of Chronological Networks, and also as part of the ChronoLog software.

Acknowledgements

Eythan Levy was supported by the Center for Absorption in Science (Israel Ministry of Absorption), by the Dan David Foundation and by a Rotenstreich Fellowship for Outstanding Doctoral Students in the Humanities. The authors warmly thank Prof. Israel Finkelstein and Dr Alfred Kromholz for their valuable comments on this paper. We also thank Mr Itamar Ben-Ezra for designing the CHRONOLOG logo.

A Appendix: details of the Dyn. 26 case study

A.1 Dataset

King	Highest attested regnal year	CHRONOLOG constraint
Psammetichus I	55	at least 54 years
Necho II	16	at least 15 years
Psammetichus II	7 (*year of death*)	6 years
Apries	20	at least 19 years
Amasis	44	at least 43 years
Psammetichus III	None	None

Table 9: Highest attested regnal years from contemporary Dyn. 26 inscriptions (see [HW82, 1166], [HKW06, p. 281-282] and [Dep96, p. 186]). All kings are assigned a minimum duration, except Psammetichus II, who has an exact duration, since an inscription provides his exact date of death. Note that Egyptian regnal years corresponded to civil calendar years, ranging from one New Year's Day to the next. The *predating* system was used in that period, meaning that when a king died in a given year, the remaining months until the next New Year's Day were counted as Year 1 of the new king ([Gar45], [HKW06, p. 461-463]). When counting reign durations using whole years, that last year of the deceased king was attributed to the new king. Thus Psammetichus II, who died in the course of his seventh year, is attributed 6 years of reign (rather than 7).

Name	Birth	Death	Duration	CHRONOLOG constraints
Apis bull III	Year 53 of Psam. I (Month 6, Day 19)	Year 16 of Necho II (Month 2, Day 6)	16 years, 7 months, 17 days	Starts 52 years after the start of Psam. I
				Ends 15 years after the start of Necho II
				Duration = 17 years
Apis bull IV	Year 16 of Necho II (Month 2, Day 7)	Year 12 of Apries (Month 8, Day 12)	17 years, 6 months, 5 days	Starts 15 years after the start of Necho II
				Ends 11 years after the start of Apries
				Duration = 17 years
Priest Psammetichus	Year 1 of Necho II (Month 11, Day 1)	Year 27 of Amasis (Month 8, Day 28)	65 years, 10 months, 2 days	Starts 0 years after the start of Necho II
				Ends 26 years after the start of Amasis
				Duration = 66 years
Other Psammetichus	Year 3 of Necho II (Month 10, Day 1 or 2)	Year 35 of Amasis (Month 2, Day 6)	71 years, 4 months, 6 days	Starts 2 years after the start of Necho II
				Ends 34 years after the start of Amasis
				Duration = 72 years
Besmaut	Year 18 of Psam. I (no months or days given)	Year 23 of Amasis (no months or days given)	99 years (no months or days given)	Starts 17 years the after the start of Psam. I
				Ends 22 years after the start of Amasis
				Duration = 99-100 years

Table 10: Funerary stelae of Apis Bulls and individuals spanning several reigns (adapted from [Kie53], p. 153-157). These inscriptions help set the precise duration of kings Psammetichus I, Necho II and Apries. The dates and durations in the stelae are given in day precision (except for the stela of Besmaut). The following example illustrates how they were converted to whole years in the CHRONOLOG constraints. Apis Bull III was born in Year 53 of Psammetichus I (Month 6, Day 19), died in Year 16 of Necho II (Month 2, Day 6) and lived 6 years, 7 months and 17 days. In the CHRONOLOG constraint, he is assigned 17 years of life, because adding 7 months and 17 days to his birth date (Month 6, Day 19) yields an additional complete year, after counting the initial 16 years. Such is also the case for the priest Psammetichus and the other Psammetichus, but not for Apis Bull IV. Regarding Besmaut, the absence of months and days in the dates and duration obliges us to set a range of 99-100 years for his reign, as we do not know if the sum of the fractional parts of his birth date and duration exceeded a year. Finally, note that the CHRONOLOG constraint for the start of Apis Bull III is "starts 52 years after the start of Psammetichus I" (rather than 53 years), since regnal years start at 1 rather than 0. The same rule holds for the other start and end years.

King	Herodotus	Manetho	ChronoLog constraint
Psammetichus I	54 years	54 years	Not used
Necho II	16 years	6 years	
Psammetichus II	6 years	6 years	
Apries	25 years	19 years	
Amasis	44 years	44 years	44 years
Psammetichus III	6 months	6 months	0-1 years

Table 11: Dynasty 26 reign durations by Herodotus (II.157-161, III.10-14) and Manetho (Africanus) [Man40, p. 169-171]. Only the durations of Amasis and Psammetichus III are used in the ChronoLog models. Psammetichus III's reign of 6 months is set to 0-1 years in the ChronoLog constraint, as a 6 months reign can count for either 0 or 1 year in the Egyptian predating system, depending on whether the reign started in the first half or the second half of the year (see caption to Table 9).

A.2 CHRONOLOG models

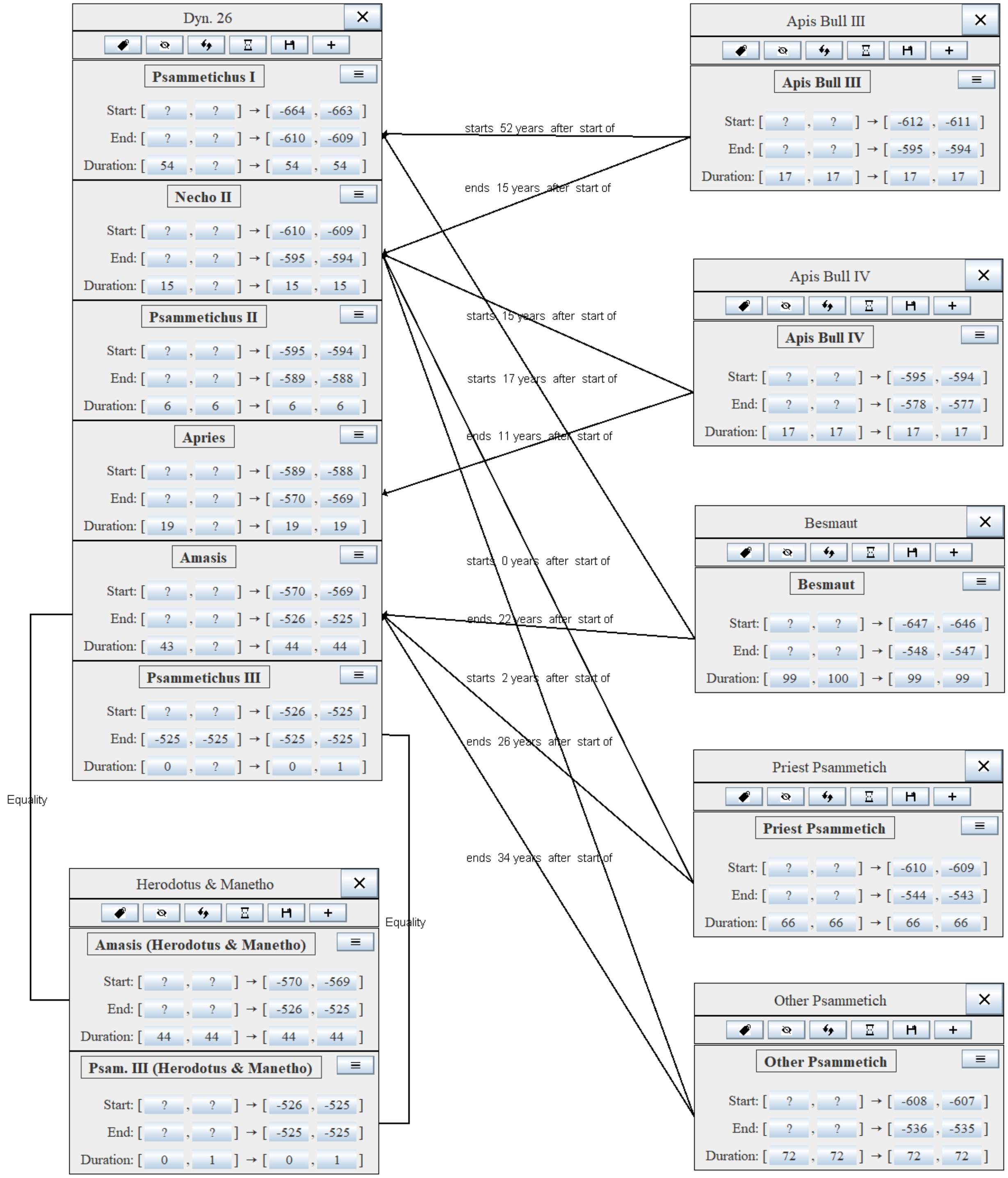


Figure 21: CHRONOLOG model for Dyn. 26 (with 0-1 years for Psammetichus III).

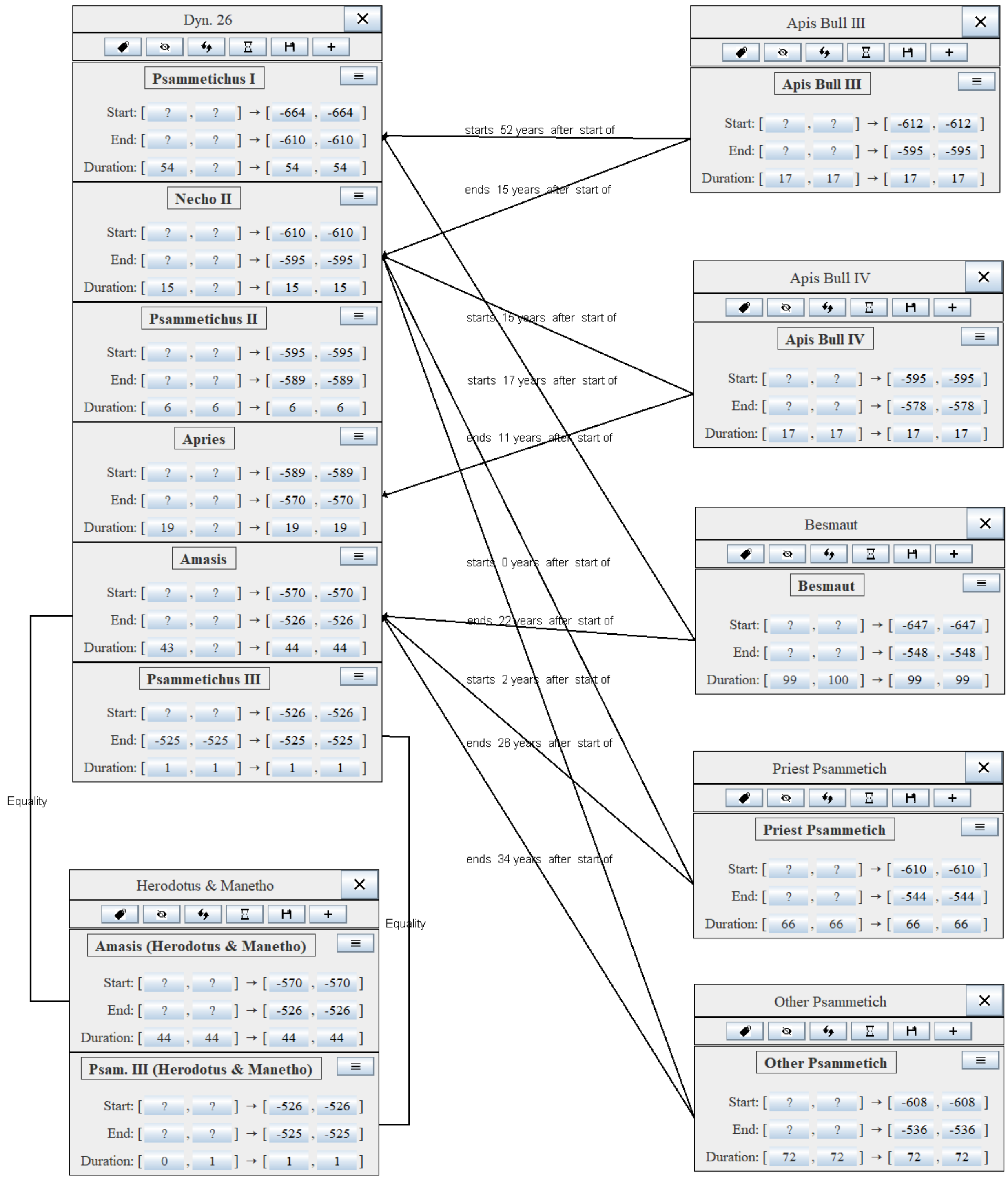


Figure 22: CHRONOLOG model for Dyn. 26 (with 1 year for Psammetichus III).

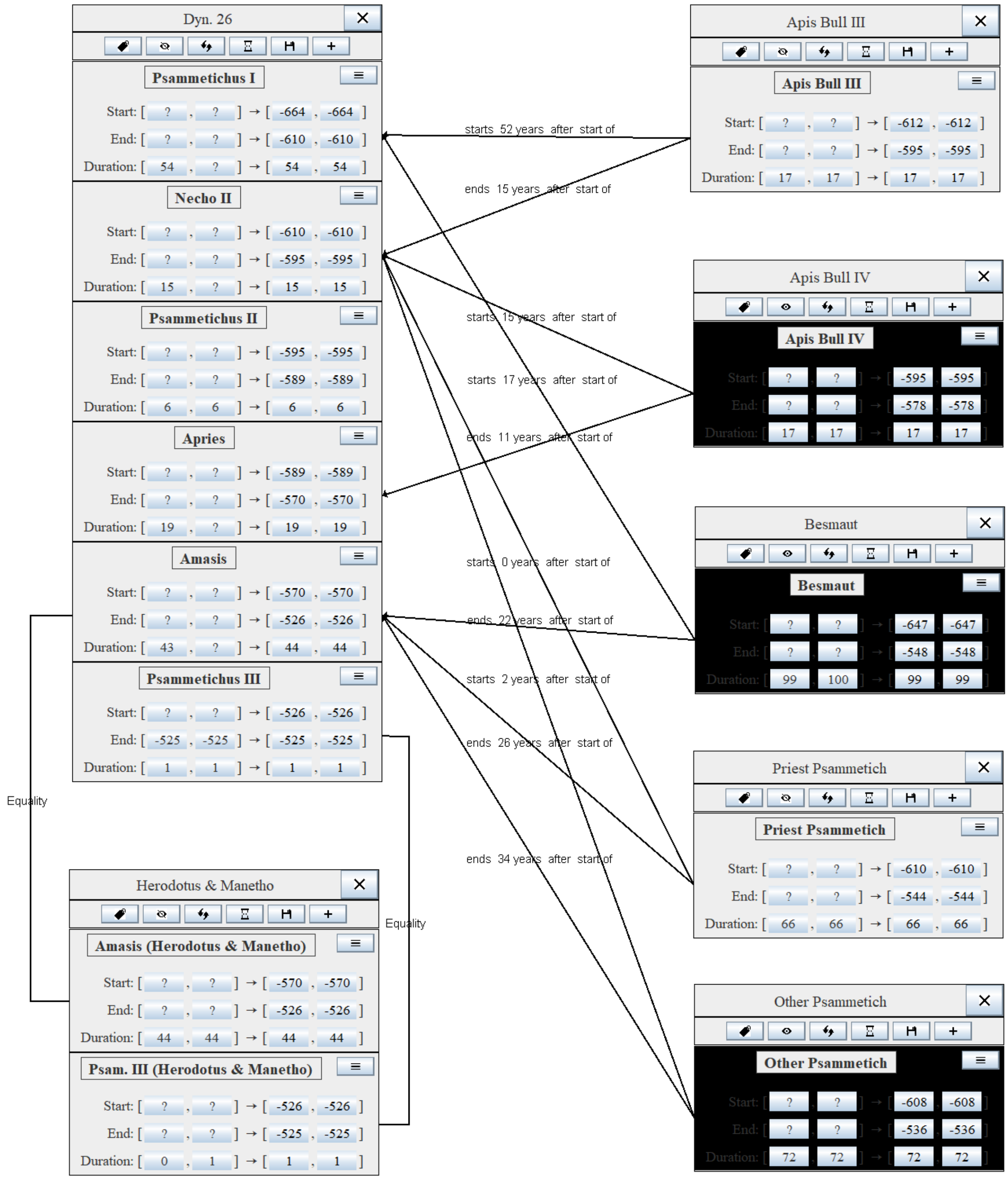


Figure 23: Same model as Figure 22, but without Apis Bull IV, Besmaut and the “other” Psammetich. The resulting chronology is not affected.

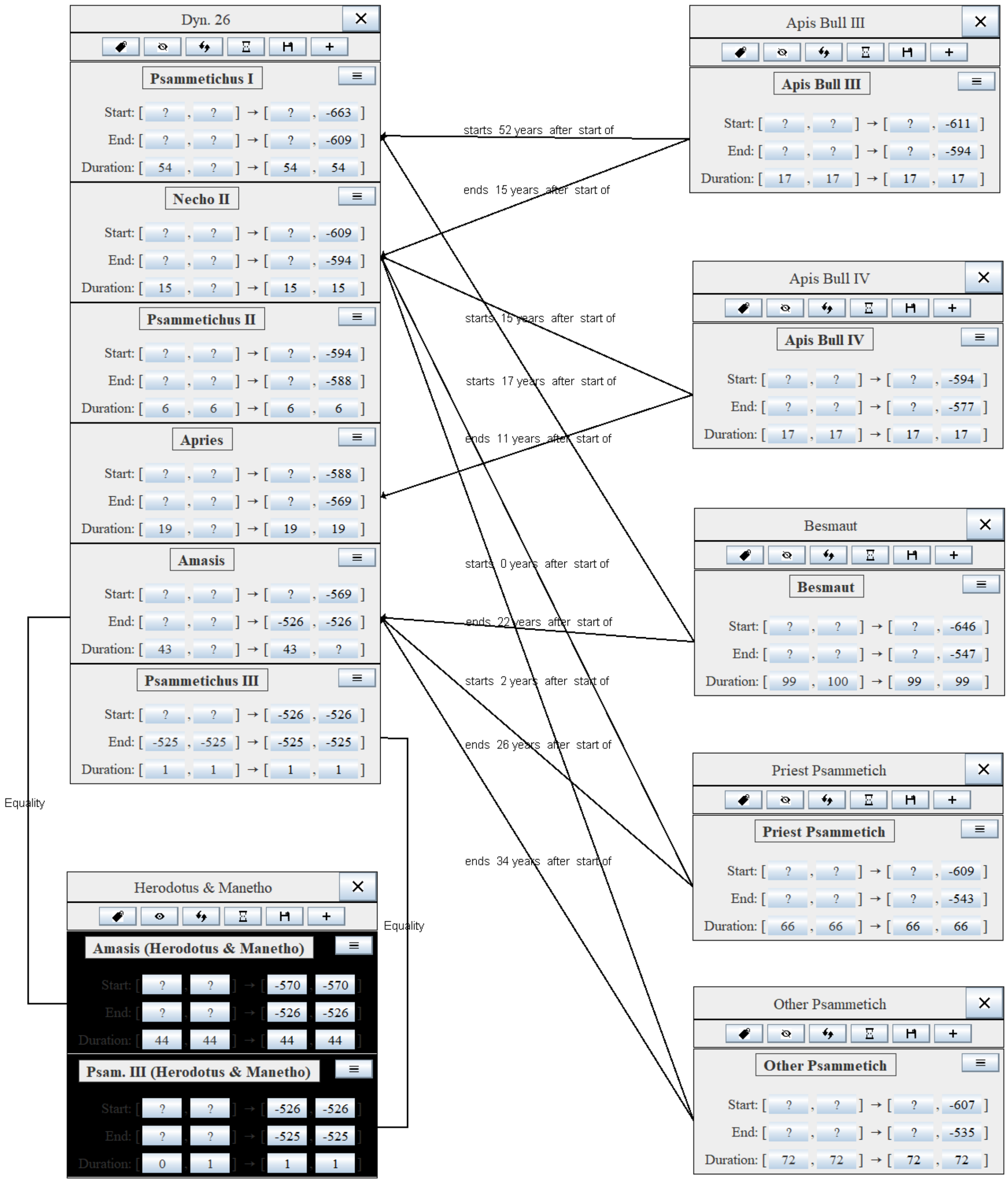


Figure 24: Same model as Figure 22, but without Herodotus's and Manetho's reign durations. The resulting chronology has no maximum duration for Amasis and no lower bounds for the dates of most pharaohs.

References

[AD94] Rajeev Alur and David L. Dill. A theory of timed automata. *Theoretical Computer Science*, 126(2):183–235, 1994.

[All84] James F. Allen. Towards a general theory of action and time. *Artificial Intelligence*, 23:123–154, 1984.

[All91] James F. Allen. Time and time again: The many ways to represent time. *International Journal of Intelligent Systems*, 6(4):341–355, 1991.

[BDL04] Gerd Behrmann, Alexandre David, and Kim G. Larsen. A tutorial on UPPAAL. In *Formal Methods for the Design of Real-Time Systems: 4th International School on Formal Methods for the Design of Computer, Communication, and Software Systems, SFM-RT 2004*, volume 3185 of *Lecture Notes in Computer Science*, pages 200–236. Springer, September 2004.

[BDM+98] Marius Bozga, Conrado Daws, Oded Maler, Alfredo Olivero, Stavros Tripakis, and Sergio Yovine. KRONOS: A model-checking tool for real-time systems (tool-presentation). In *Formal Techniques in Real-Time and Fault-Tolerant Systems, 5th International Symposium, FTRTFT'98, Lyngby, Denmark, September 14-18, 1998, Proceedings*, volume 1486 of *Lecture Notes in Computer Science*, pages 298–302. Springer, 1998.

[Bec02] Jürgen von Beckerath. Nochmals die Eroberung Ägyptens durch Kambyses. *Zeitung der Ägyptischen Sprache und Altertumskunde*, 129:1–5, 2002.

[Bry05] Trevor Bryce. *The Kingdom of the Hittites*. Oxford University Press, Oxford, 2nd edition, 2005.

[Col08] John N. Coldstream. *Greek Geometric Pottery: A Survey of Ten Local Styles and Their Chronology*. Bristol Phoenix, 2nd edition, 2008.

[Cre12] Enrico R. Crema. Modelling temporal uncertainty in archaeological analysis. *Journal of Archaeological Method and Theory*, 19:440–461, 2012.

[CU80] Eugene Cruz-Uribe. On the existence of Psammetichus IV. *Serapis*, 5(2):35–39, 1980.

[DD16] Peter Demján and Dagmar Dreslerová. Modelling distribution of archaeological settlement evidence based on heterogeneous spatial and temporal data. *Journal of Archaeological Science*, 69:100–109, 2016.

[Dep96] Leo Depuydt. Egyptian regnal dating under Cambyses and the date of the Persian conquest. In *Studies in Honor of William Kelly Simpson*, pages 179–190. Boston Museum of Fine Arts, Boston, 1996.

[Des16] Bruno Desachy. From observed successions to quantified time: formalizing the basic steps of chronological reasoning. *Acta Imeko*, 5(2):4–13, 2016.

[Dil89] David L. Dill. Timing assumptions and verification of finite-state concurrent systems. In *Automatic Verification Methods for Finite State Systems, International Workshop, Grenoble, France, June 12-14, 1989, Proceedings*, volume 407 of *Lecture Notes in Computer Science*, pages 197–212. Springer Verlag, 1989.

[Fal20] David A. Falk. Computer analytics in chronology testing and its implications for the date of the Exodus. In Richard E. Averbeck and K. Lawson Younger Jr., editors, *"An Excellent Fortress for His Armies, a Refuge for the People", Egyptological, Archaeological, and Biblical Studies in Honor of James K. Hoffmeier*. Pennsylvania State University Press, University Park, PA, 2020. In press.

[FAMP17] Israel Finkelstein, Eran Arie, Mario A.S. Martin, and Eliezer Piasetzky. New evidence on the Late Bronze/Iron I transition at Megiddo: Implications for the end of the Egyptian rule and the appearance of Philistine pottery. *Egypt and the Levant*, 27:261–280, 2017.

[Flo62] Robert W. Floyd. Algorithm 97: Shortest path. *Communications of the ACM*, 5(6):345–, June 1962.

[Gar45] Alan H. Gardiner. Regnal years and civil calendar in Pharaonic Egypt. *The Journal of Egyptian Archaeology*, 31:11–28, 1945.

[GLP17] Gilles Geeraerts, Eythan Levy, and Frédéric Pluquet. Models and Algorithms for Chronology. In Sven Schewe, Thomas Schneider, and Jef Wijsen, editors, *24th International Symposium on Temporal Representation and Reasoning (TIME 2017)*, volume 90 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 13:1–13:18, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[Har79] Edward C. Harris. *Principles of Archaeological Stratigraphy*. Academic Press, New York. NY, 1979.

[HKW06] Erik Hornung, Rolf Krauss, and David A. Warburton. *Ancient Egyptian Chronology*. Brill, 2006.

[Hol04] Mads Kähler Holst. Complicated relations and blind dating: Formal analysis of relative chronological structures. In Caitlin E. Buck and Andrew R. Millard, editors, *Tools for Constructing Chronologies*, pages 129–147. Springer, London, 2004.

[HW82] Wolfgang Helck and Wolfhart Westendorf, editors. *Lexikon der Ägyptologie, Band IV*. Harrassowitz, Wiesbaden, 1982.

[Joh77] Donald B. Johnson. Efficient algorithms for shortest paths in sparse networks. *Journal of the ACM*, 24(1):1–13, 1977.

[Joh16] Christine Leigh Johnston. *Networks and Intermediaries: Ceramic Exchange Systems in the Late Bronze Age Mediterranean*. PhD thesis, University of California, Los Angeles, 2016.

[Kie53] Friedrich Karl Kienitz. *Die politische Geschichte Ägyptens vom 7.bis zum 4.Jahrhundert vor der Zeitwende*. Akademie Verlag, Berlin, 1953.

[Kit00] Kenneth A. Kitchen. Regnal and genealogical data of ancient Egypt. In Manfred Bietak, editor, *The Synchronisation of Civilizations in the Eastern Mediterranean in the Second Millenium B.C.*, pages 39–52. Austrian Academy of Sciences Press, Vienna, 2000.

[Kro87] Alfred H. Kromholz. Business and industry in archaeology. In Paul Ahström, editor, *High, Middle or Low? (Part 1)*, pages 119–137. Paul Ahströms Förlag, Gothenburg, 1987.

[Man40] Manetho. *History of Egypt and Other Works*. Harvard University Press, Cambridge, MA, 1940. Translated by W. G. Waddell.

[MAR+17] Amgad Madkour, Walid G. Aref, Faizan Ur Rehman, Mohamed Abdur Rahman, and Saleh Basalamah. A Survey of Shortest-Path Algorithms. Technical Report CoRR abs/1705.02044, Cornell University Library, arXiv.org, 2017.

[Mer92] Robert S. Merrillees. The absolute chronology of the Bronze Age in Cyprus: A revision. *Bulletin of the American Schools of Oriental Research*, (288):47–52, 1992.

[Mer02] Robert S. Merrillees. The relative and absolute chronology of the Cypriote White Painted Line Style. *Bulletin of the American Schools of Oriental Research*, (326):1–9, 2002.

[NH15] Franco Niccolucci and Sorin Hermon. Time, chronology and classification. In Juan A. Barcelo and Igor Bogdanovic, editors, *Mathematics and Archaeology*, pages 257–271. CRC Press, Boca Raton, London, New York, 2015.

[Par57] Richard A. Parker. The length of reign of Amasis and the beginning of the Twenty-sixth Dynasty. *Mitteilungen des Deutschen Archäologischen Instituts, Abteilung Kairo*, 15:208–212, 1957.

[Ram95] Christopher Bronk Ramsey. Radiocarbon calibration and analysis of stratigraphy: the OxCal program. *Radiocarbon*, 37(2):425–430, 1995.

[Ryh97] Kim S.B. Ryholt. *The Political Situation in Egypt during the Second Intermediate Period, 1800–1550 BC*. CNI Publications, Copenhagen, 1997.

[Sha95] Ilan Sharon. Partial order scalogram analysis of relations - a mathematical approach to the analysis of stratigraphy. *Journal of Archaeological Science*, 22:751–767, 1995.

[Sho81] Robert E. Shostak. Deciding linear inequalities by computing loop residues. *Journal of the ACM*, 28(4):769–779, 1981.

[Vle91] Sven P. Vleeming. *The Gooseherds of Hou (Pap. Hou)*. Peeters, Leuven, 1991.