

# Keeping interval-based functional dependencies up-to-date <sup>\*</sup>

Carlo Combi and Pietro Sala

Department of Computer Science, University of Verona  
{[carlo.combi](mailto:carlo.combi@univr.it)|[pietro.sala](mailto:pietro.sala@univr.it)}@univr.it

**Abstract.** In the temporal database literature, every fact stored in a database may be equipped with two temporal dimensions: the valid time, which describes the time when the fact is true in the modeled reality, and the transaction time, which describes the time when the fact is current in the database and can be retrieved. Temporal functional dependencies (TFDs) add valid time to classical functional dependencies (FDs) in order to express database integrity constraints over the flow of time. Currently, proposals dealing with TFDs adopt a point-based approach, where tuples hold at specific time points, to express integrity constraints such as “*for each month, the salary of an employee depends only on his role*”. To the best of our knowledge, there are no proposals dealing with interval-based temporal functional dependencies (ITFDs), where the associated valid time is represented by an interval and there is the need of representing both point-based and interval-based data dependencies. In this paper, we propose ITFDs based on Allen’s interval relations and discuss their expressive power with respect to other TFDs proposed in the literature: ITFDs allow us to express interval-based data dependencies, which cannot be expressed through the existing point-based TFDs. ITFDs allow one to express constraints such as “*employees starting to work the same day with the same role get the same salary*” or “*employees with a given role working on a project cannot start to work with the same role on another project that will end before the first one*”. Furthermore, we propose new algorithms based on B-trees to efficiently verify the satisfaction of ITFDs in a temporal database. These algorithms guarantee that, starting from a relation satisfying a set of ITFDs, the updated relation still satisfies the given ITFDs.

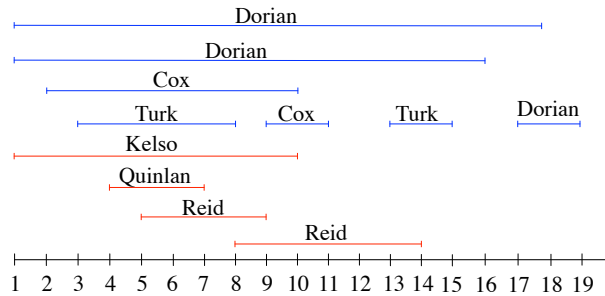
## 1 An example of interval-based constraints

Most health care institutions collect a large quantity of clinical information about patient and physician actions, such as therapies and surgeries, as well as about health care processes, such as admissions, discharges, and exam requests. All these pieces of information are temporal in nature and the associated temporal dimension needs to be carefully considered in order to be able to properly represent clinical data and to reason about them [2]. In this section, we briefly

---

<sup>\*</sup> A short summary of the results published in [3] and [4].

#	TherType	PatId	Phys	DrugCode	Qty	B	E
1	antiviral	1	Dorian	0458	300	1	16
2	analgesics	1	Cox	0976	200	2	10
3	cardiovascular	1	Turk	0118	100	3	8
4	antipyretics	1	Cox	0976	100	9	11
5	sedative	1	Turk	0345	10	13	15
6	anxiolytic	1	Dorian	0345	10	17	19
7	antiviral	2	Kelso	0458	200	1	10
8	cardiovascular	2	Quinlan	0118	100	4	7
9	analgesics	2	Reid	0976	150	5	9
10	antiviral	2	Reid	0458	300	8	14
11	antiviral	1	Dorian	0789	200	1	18



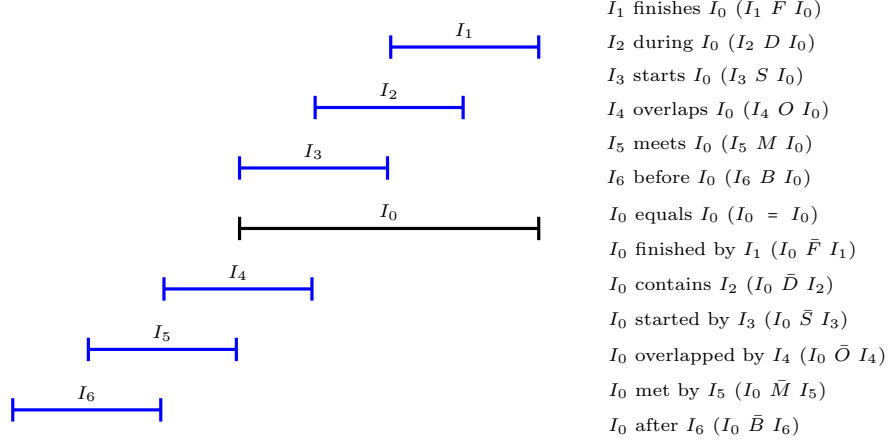
**Fig. 1.** An instance of relation *PatTherapies*, storing data about patient therapies and its representation on the time line with values for attribute *Phys*

introduce a real-world example taken from clinical medicine, namely that of patient therapies.

Suppose we have patients who undergo several different therapies: each therapy can be supervised by a physician, and consists of the administration of some drug to the patient. Information about patients and therapies is stored in a relation according to the schema  $PatTherapies(TherType, PatId, DrugCode, Qty, Phys, B, E)$ , where *TherType* identifies a type of pharmacological therapy, *PatId* represents a patient ID, *DrugCode* and *Qty* the prescribed drug and its quantity, respectively, and *Phys* the physician who made the prescription (and is responsible for the therapy). Finally, attributes *B* and *E* represent the beginning and end time points of the tuple valid interval, respectively: they represent the bounds of the interval specified by the physician for each therapy. An instance of *PatTherapies* is provided in Fig. 1.

*Example 1.* A policy of the hospital may be described as follows:

Every patient may receive several therapies at the same time from different physicians, but overlapping therapies for the same patient must be prescribed by the same physician. In other words, if a patient during a therapy needs another therapy which lasts beyond the end of the current therapy, then this therapy must be prescribed by the same physician who prescribed the other one;



**Fig. 2.** The thirteen Allen relations between intervals

It is easy to see that in order to verify these policies through the acquired data, both the start points and the end points of every pair of tuples come into play.

## 2 Interval-based functional dependencies

Given a totally ordered set  $\mathbb{O} = \langle O, \leq \rangle$ , an interval  $I$  over  $\mathbb{O}$  is a pair  $I = [b, e]$  where  $b, e \in O$  and  $b \leq e$ . For any interval  $I = [b, e]$  over  $\mathbb{O}$  let  $points(I)$  denote the set of points in  $O$  between  $b$  and  $e$ :  $points(I) = \{p \mid p \in O \text{ and } b \leq p \leq e\}$ . While the possible distinct relations between two points considering only the linear order are reduced to three (equality, successor, and predecessor), considering the order among the two endpoints of two intervals leads us to have thirteen possible relations. These relations are depicted in Fig. 2 according to the notation proposed by Allen in [1]. It is worth noting that every relation has its dual obtained by switching the position of the two intervals. Consider, for example, two intervals  $I_1 = [b_1, e_1]$  and  $I_2 = [b_2, e_2]$ : we have that  $I_1 D I_2$  ( $I_1$  during  $I_2$ ), if and only if  $b_2 < b_1 < e_1 < e_2$ . By reverting the arguments, we have that  $I_2 \bar{D} I_1$  ( $I_2$  contains  $I_1$ ), if and only if  $b_2 < b_1 < e_1 < e_2$ , which is equivalent to  $I_1 D I_2$ . More precisely, given two intervals  $I = [b, e]$  and  $I' = [b', e']$  we say that:

- (1)  $I = I'$  iff  $b = b'$  and  $e = e'$ ;
- (2)  $I M I'$  iff  $e = b'$ ;
- (3)  $I S I'$  iff  $b = b'$  and  $e < e'$ ;
- (4)  $I F I'$  iff  $b > b'$  and  $e = e'$ ;
- (5)  $I O I'$  iff  $b < b'$  and  $b' < e < e'$ ;
- (6)  $I D I'$  iff  $b' < b$  and  $e < e'$ ;
- (7)  $I B I'$  iff  $e < b'$ .

In discussing our new functional dependencies based on intervals within a relational framework, we use a simple temporal (relational) data model based on the concept of temporal relation. A temporal relation  $r$  is a relation on a temporal relation schema  $\mathcal{R}$  defined on attributes  $U \cup \{B, E\}$ , where  $U$  represents a

set of atemporal attributes and  $B, E$  are the temporal attributes describing the valid interval of a tuple. We assume that the domain of both attributes  $B$  and  $E$  is a totally ordered set  $\mathbb{O}$ . Clearly, a tuple  $t \in r$  satisfies  $t[B] \leq t[E]$ . We recall that, assuming the underlying domain for attributes  $A_1$  and  $A_2$  has a total order, atomic formulas for comparing tuples are either of the form  $t[A_1] \theta t'[A_2]$  or of the form  $t[A_1] \theta c$ , with  $\theta \in \{=, \neq, <, \leq, >, \geq\}$ ,  $A_1, A_2$  being attribute names,  $c$  a constant value and  $t, t'$  tuples of relation  $r$ . To avoid ambiguities in the terminology employed, in the following we will use *(temporal) instance* for “(temporal) relation” and will let *relation* refer to Allen’s interval relations.

## 2.1 Interval-based temporal functional dependencies

Let us now consider the basic definition of an *Interval-based Temporal Functional Dependency* (ITFD). In the following, we will only deal with interval relations in the set  $\mathcal{A} = \{S, F, B, M, D, O, =\}$ . Indeed, in this case it is not meaningful to distinguish between a relation and its dual, as it will be clear from the following definition of interval-based temporal functional dependency.

**Definition 1.** *Let  $X$  and  $Y$  be sets of atemporal attributes of a temporal relation schema  $\mathcal{R} = R(U, B, E)$  and  $\sim$  an Allen’s interval relation. An instance  $r$  of  $\mathcal{R}$  satisfies an ITFD  $X \rightarrow_{\sim} Y$  if for each pair of tuples  $t_1$  and  $t_2$  such that  $[t_1[B], t_1[E]] \sim [t_2[B], t_2[E]]$  and  $t_1[X] = t_2[X]$ , it is also true that  $t_1[Y] = t_2[Y]$ .*

Basically, ITFDs group tuples whose  $B$  and  $E$  attribute values satisfy the interval relation  $\sim$ . In the above definition, all the possible tuples having as valid interval either  $[b, e]$  or  $[e', b']$ , where  $[b, e] \sim [e', b']$  are considered together. If there exist two tuples having their valid intervals related through the considered relation  $\sim$ , respectively, and both tuples agree on (the tuple of) values of atemporal attributes  $X$ , then the ITFD imposes that both tuples must agree on (the tuple of) values of atemporal attributes  $Y$ .

As already mentioned, we focus only on (sub) set  $\mathcal{A}$  of Allen’s interval relations, without considering the dual ones. Indeed, dual relations are not needed for the specification and verification of ITFDs, because ITFDs are based on the equality of the considered (atemporal) values. Thus, each (ordered) pair of tuples satisfying an interval relation will satisfy also the dual one, where tuples will be considered in the pair with the opposite order. In other words, any ITFD with a given interval relation implies also the corresponding ITFD with the dual relation (and vice versa).

Let us now consider the first requirement expressed in Example 1 of Sect. 1: it can be rephrased as “*overlapping drug administrations for a given patient must have the same physician*”. This constraint can be expressed by the ITFD

$$PatId \rightarrow_O Phys.$$

A time-oriented graphical account of tuples of relation *PatTherapies* is provided in the lower part of Fig. 1. As we may notice, the instance satisfies ITFD

$PatId \rightarrow_O Phys$  only for tuples related to the patient with  $PatId = 1$ . Dr. Cox added a therapy *antipyretics*, but the related valid interval is contained in the interval of therapy *antiviral* prescribed by Dr. Dorian. Tuples related to therapies of patient with  $PatId = 2$  instead do not satisfy ITFD  $PatId \rightarrow_O Phys$ , as both intervals of therapies prescribed by Dr. Reid overlap a therapy prescribed by another physician. This kind of property cannot be expressed with point-based TFDs.

Verifying the satisfaction of  $X \overset{\sim}{\rightarrow} Y$  may be considered in two different but intertwined ways: i) given an instance  $r$  of  $R$ , check whether or not  $r$  satisfies  $X \overset{\sim}{\rightarrow} Y$ , ii) given an instance  $r$  of  $R$  satisfying  $X \overset{\sim}{\rightarrow} Y$  and a tuple  $t$ , verify whether  $r \cup \{t\}$  still satisfies  $X \overset{\sim}{\rightarrow} Y$ . We call the first problem *checking ITFD satisfaction*, while the second one is called *incremental ITFD verification*. It is not difficult to see that these two problems are closely related. In fact, checking ITFD satisfaction reduces to the incremental ITFD verification by adopting the algorithm developed for this problem and, starting from  $i = 0$  with instance  $r_0 = \emptyset$  with schema  $R$ , incrementally verifying whether  $r_i \cup \{t_i\}$  with  $t_i \in r \setminus r_i$  satisfies ITFD  $X \overset{\sim}{\rightarrow} Y$ . If the update of  $r_i$  with  $t_i$  still verifies  $X \overset{\sim}{\rightarrow} Y$ , then  $r_{i+1} = r_i \cup \{t_i\}$ ,  $i = i + 1$  and the algorithm is applied again. If  $r$  satisfies  $X \overset{\sim}{\rightarrow} Y$ , after  $|r|$  iterations we can determine ITFD satisfaction. Some complexity improvements to this naive approach can be done as shown in Table 1.

ITFD	tuple insertion	tuple deletion	ITFD satisfaction checking
$X \rightarrow_S Y$	$\mathcal{O}(\log( r ))$	$\mathcal{O}(\log( r ))$	$\mathcal{O}( r  \cdot \log( r ))$
$X \rightarrow_F Y$	$\mathcal{O}(\log( r ))$	$\mathcal{O}(\log( r ))$	$\mathcal{O}( r  \cdot \log( r ))$
$X \rightarrow_B Y$	$\mathcal{O}(\log( r ))$	$\mathcal{O}(\log( r ))$	$\mathcal{O}( r  \cdot \log( r ))$
$X \rightarrow_M Y$	$\mathcal{O}(\log( r ))$	$\mathcal{O}(\log( r ))$	$\mathcal{O}( r  \cdot \log( r ))$
$X \rightarrow_D Y$	$\mathcal{O}(\log( r ))$	$\mathcal{O}( r )$	$\mathcal{O}( r  \cdot \log( r ))$
$X \rightarrow_O Y$	$\mathcal{O}(\log( r ))$	$\mathcal{O}( r )$	$\mathcal{O}( r  \cdot \log( r ))$

**Table 1.** The complexities for the tuple insertion, deletion, and ITFD satisfaction checking, by our proposed incremental verification algorithm of ITFDs

## References

- [1] James F. Allen. Maintaining knowledge about temporal intervals. *Commun. ACM*, 26(11):832–843, 1983.
- [2] Carlo Combi, Elpida Keravnou-Papailiou, and Yuval Shahar. *Temporal Information Systems in Medicine*. Springer-Verlag New York, Inc., New York, NY, USA, 2010.
- [3] Carlo Combi and Pietro Sala. Temporal functional dependencies based on interval relations. In Carlo Combi, Martin Leucker, and Frank Wolter, editors, *TIME*, pages 23–30. IEEE, 2011.
- [4] Carlo Combi and Pietro Sala. Interval-based temporal functional dependencies: specification and verification. *Annals of Mathematics and Artificial Intelligence*, pages 1–46, 2013.