(Abschnitt: BfArM-Daten – Umsteiger-Algorithmen)

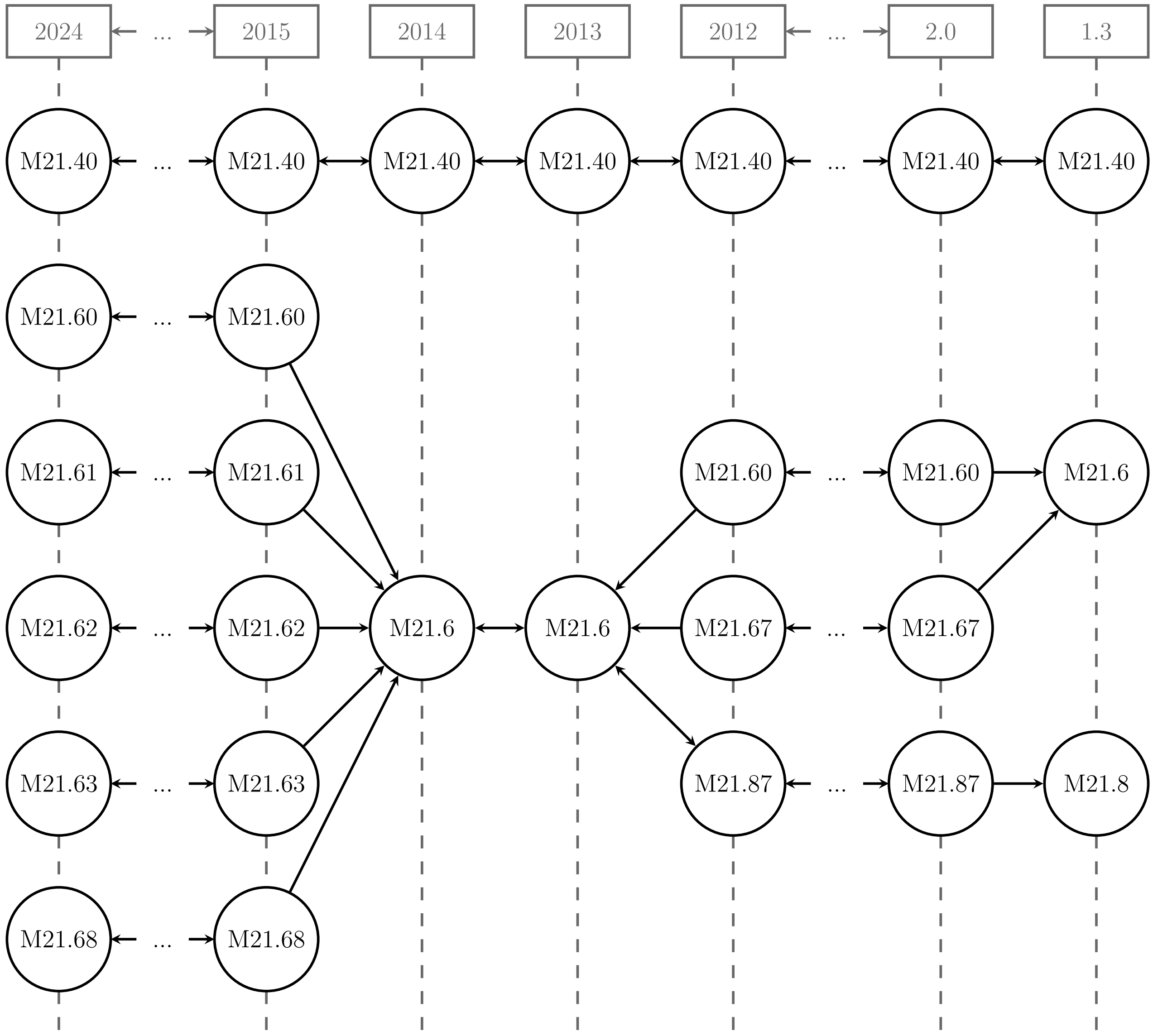

Abbildung 1: caption

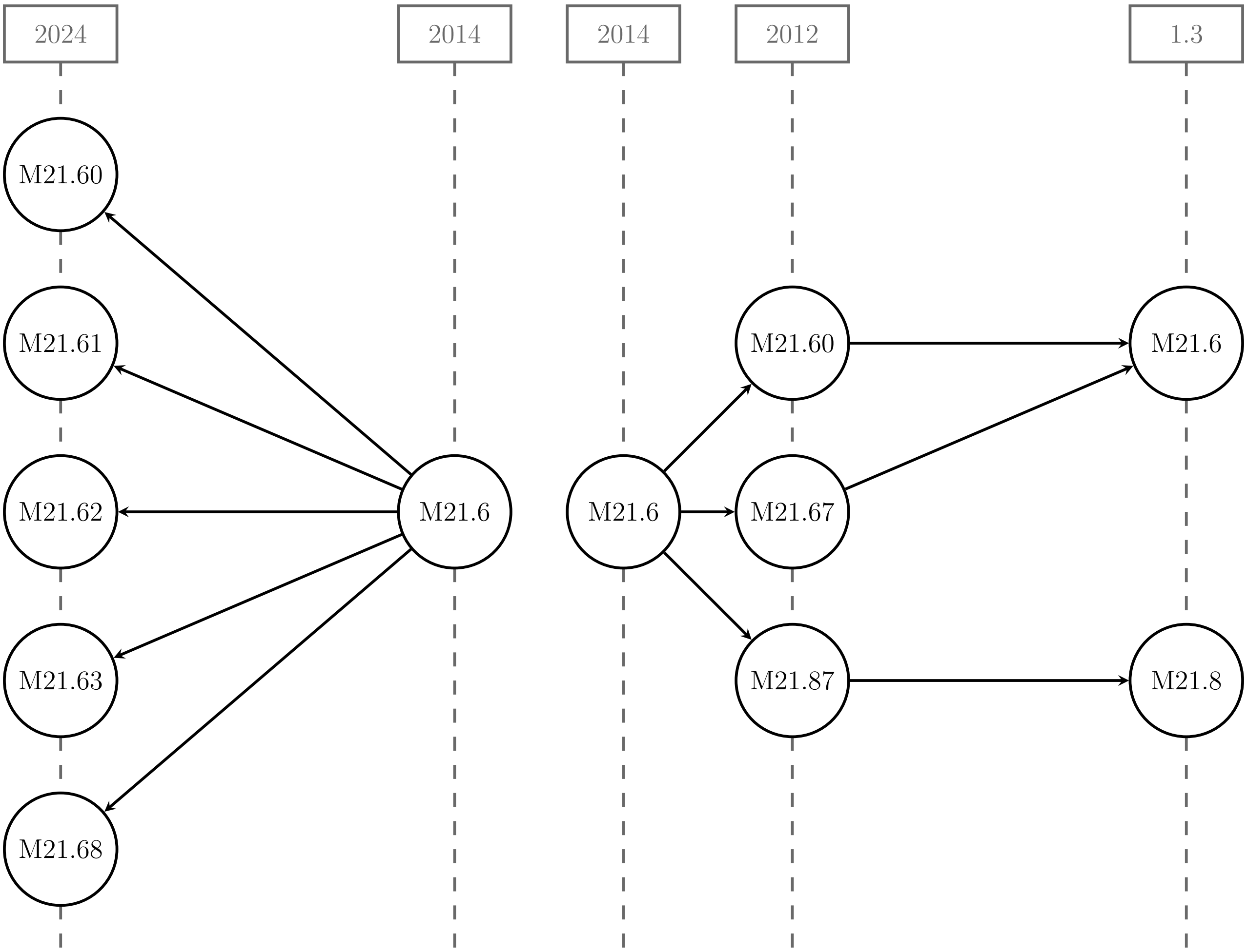

Abbildung 2: caption

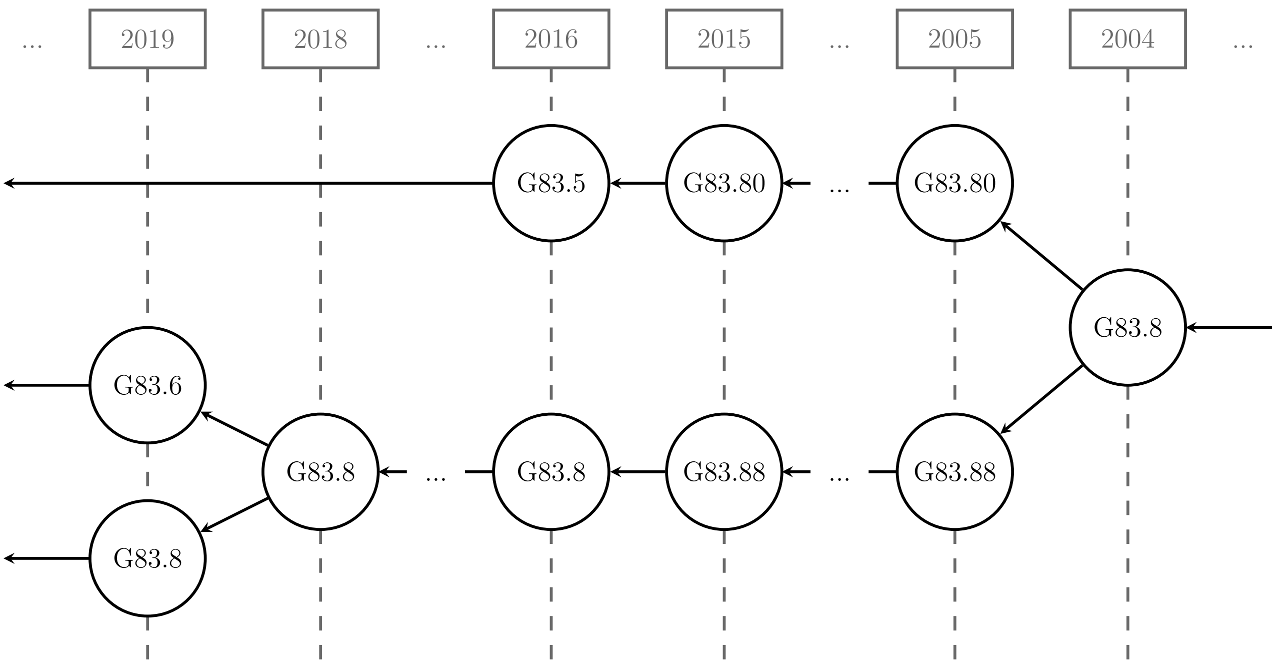

Abbildung 3: caption

D26: The ***transitive closure*** R^* of a binary relation R is the relation R^* defined by $(x, y) \in R^*$ if and only if there exists a sequence $x = v_0, v_1, v_2, \ldots, v_k = y$ such that $k \geq 1$ and $(v_i, v_{i+1}) \in R$, for $i = 0, 1, \ldots, k-1$. Equivalently, the transitive closure R^* of the relation R is the smallest transitive relation that contains R.

D27: Let G be the digraph representing a relation R. Then the digraph G^* representing the transitive closure R^* of R is called the ***transitive closure of the digraph*** G. Thus, an arc (x, y), $x \neq y$, is in the transitive closure G^* if and only if there is a directed x-y path in G. Similarly, there is a self-loop in digraph D^* at vertex x if and only if there is a directed cycle in digraph G that contains x.

Abbildung 4: aus (Gross et al., 2013, Seite 172)

E7: Suppose a relation R on the set $S = \{a, b, c, d\}$ is given by

$$\{(a, a), (a, b), (b, c), (c, b), (c, d)\}$$

Then the digraph G representing the relation R and the transitive closure G^* are as shown in Figure 3.1.7.

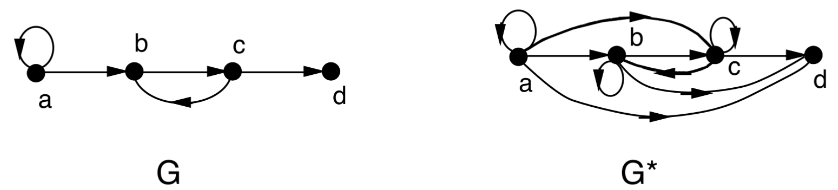

Abbildung 5: aus (Gross et al., 2013, Seite 172)

5.3.1 The Purdom Algorithm

Purdom, in [Pur70], made two key observations:

1. During the computation of transitive closure of a directed acyclic graph, if node $s \prec_t t$, then additions to the successor list of node s cannot affect the successor list of node t. One should therefore compute the successor list of t first and then that of s. By processing nodes in reverse topological order, one need add to a node only the successor lists of its immediate successors since the latter would already have been fully expanded. We call this idea the *immediate successor* optimization [AgJ90].

2. All nodes within a strongly connected component (SCC) in a graph have identical reachability properties, and the condensation graph obtained by collapsing all the nodes in each strongly connected component into a single node is acyclic.

Abbildung 6: aus (Dar, 1993, Seite 76f)

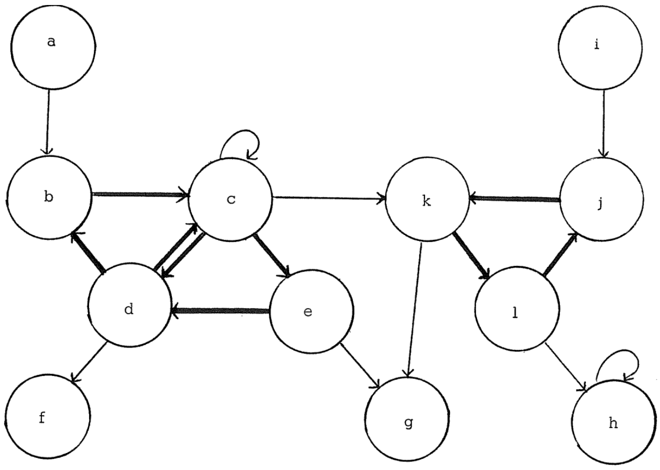

Abbildung 7: aus (Purdom, 1970, Seite 78)

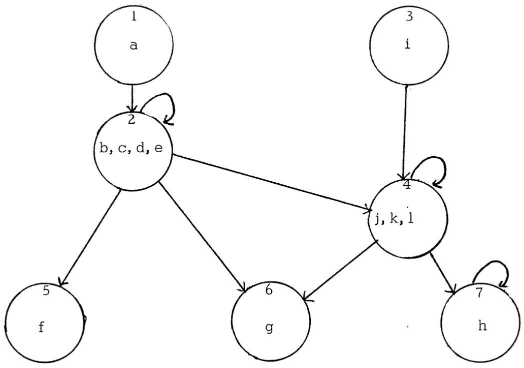

Abbildung 8: aus (Purdom, 1970, Seite 78)

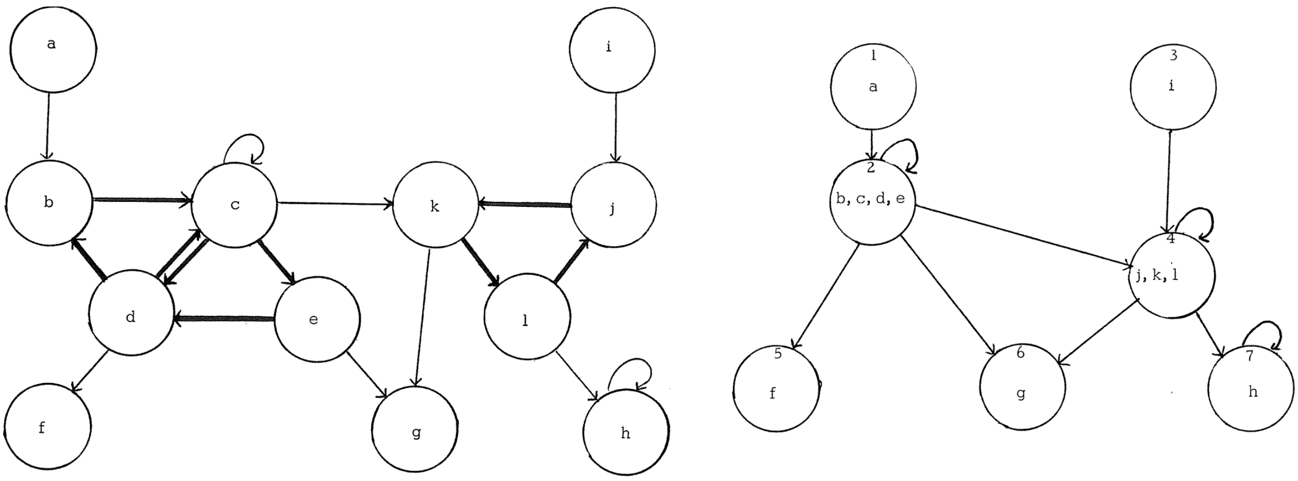

Abbildung 9: aus (Purdom, 1970, Seite 78)

2.2 Search

A second approach for computing the transitive closure is to search the graph n times, each time starting from a different node, thereby determining what can be reached from that node. In this approach, we completely disregard any parts of the TC-relation that have already been computed for other starting nodes. Instead, we search the entire part of the graph that is reachable from the starting node by following every edge we can get to.

Abbildung 10: aus (Jakobsson, 1991, Seite 200)

Literatur

S. Dar, *Augmenting Databases with Generalized Transitive Closure*, ser. Computer sciences technical report. University of Wisconsin–Madison, 1993.

J. Gross, J. Yellen, and P. Zhang, *Handbook of Graph Theory, Second Edition*, ser. Discrete Mathematics and Its Applications. Taylor & Francis, 2013.

H. Jakobsson, “Mixed-approach algorithms for transitive closure,” in *Proceedings of the tenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, 1991, pp. 199–205. [Online]. Available: https://dl.acm.org/doi/pdf/10.1145/113413.113431

P. Purdom, “A transitive closure algorithm,” *BIT Numerical Mathematics*, vol. 10, no. 1, pp. 76–94, 1970.