

On Computing the Minimal Generator Family for Concept Lattices and Icebergs

Kamal Nehmé¹, Petko Valtchev¹, Mohamed H. Rouane¹, and Robert Godin²

¹ DIRO, Université de Montréal, Montréal (Qc), Canada

² Département d'informatique, UQAM, Montréal (Qc), Canada

Abstract. Minimal generators (or *mingen*) constitute a remarkable part of the closure space landscape since they are the antipodes of the closures, i.e., minimal sets in the underlying equivalence relation over the powerset of the ground set. As such, they appear in both theoretical and practical problem settings related to closures that stem from fields as diverging as graph theory, database design and data mining. In FCA, though, they have been almost ignored, a fact that has motivated our long-term study of the underlying structures under different perspectives. This paper is a two-fold contribution to the study of mingen families associated to a context or, equivalently, a closure space. On the one hand, it sheds light on the evolution of the family upon increases in the context attribute set (e.g., for purposes of interactive data exploration). On the other hand, it proposes a novel method for computing the mingen family that, although based on incremental lattice construction, is intended to be run in a batch mode. Theoretical and empirical evidence witnessing the potential of our approach is provided.

1 Introduction

Within the closure operators/systems framework, *minimal generators*, or, as we shall call them for short, *mingen*, are, beside closed and pseudo-closed elements, key elements of the landscape. In some sense they are the antipodes of the closed elements: a mingen lays at the bottom of its class in the closure-induced equivalence relation over the ground set, whereas the respective closure is the unique top of the class. This is the reason for mingen to appear in almost every context where closures are used, e.g., in fields as diverging as the database design (as *key* sets [7]), graph theory (as *minimal transversals* [2]), data analysis (as *lacunes irréductibles*¹, the name given to them in French in [6]) and data mining (as minimal premises of association rules [8]). In FCA, mingen have been used for computational reasons, e.g., in TITANIC [11], where they appear explicitly, as opposed to their implicit use in NextClosure [3] as canonical representations (prefixes) of concept intents.

Despite the important role played by mingen, they have been paid little attention so far in the FCA literature. In particular, many computational problems

¹ Irreducible gaps, translation is ours.

related to the mingen family are not well understood, let alone efficiently solved. This observation has motivated an ongoing study focusing on the mingen sets in a formal context that considers them from different standpoints including batch and incremental computation, links to other remarkable members of the closure framework such as pseudo-closed, etc. Recently, we proposed an efficient method for maintaining the mingen family of a context upon increases in the context object set [16]. The extension of the method to lattice merge has been briefly sketched as well. Moreover, the mingen-related part of the lattice maintenance method from [16] was proved to easily fit the iceberg lattice maintenance task as in [10].

In this paper, we study the mingen maintenance problem in dual settings, i.e., upon increases in the attribute set of the context. The study has a two-fold motivation and hence contributes in two different ways to the FCA field. Thus, on the one hand, the evolution of the mingen is given a characterization, in particular, with respect to the sets of stable/vanishing/newly forming mingen. To assess the impact of the provided results, it is noteworthy that although in lattice maintenance the attribute/object cases admit dual resolution, this does not hold for mingen maintenance, hence the necessity to study the attribute case separately. On the other hand, the resulting structure characterizations are embedded into an efficient maintenance method that can, as all other incremental algorithms, be run in a batch mode. The practical performances of the new method as batch iceberg-plus-mingen constructor have been compared to the performances of TITANIC, the algorithm which is reportedly the most efficient one producing the mingen family and the frequent part of the closure family². The results of the comparison proved very encouraging: although our algorithm produces the lattice precedence relation beside concepts and mingen, it outperformed TITANIC when run on a sparse data set. We tend to see this as a clear indication of the potential the incremental paradigm has for mingen computation.

The paper starts with a recall of basic results about lattices, mingen, and incremental lattice update (Section 2). The results of the investigation on the evolution of the mingen family are presented in Section 3 while the proposed maintenance algorithm, INCA-GEN, is described in Section 4. In Section 5, we design a straightforward adaptation of INCA-GEN to iceberg concept lattice maintenance. Section 6 discusses preliminary results of the practical performance study that compared the algorithm to TITANIC.

2 Background on Concept Lattices

In the following, we recall basic results from FCA [18] that will be used in later paragraphs.

² Other algorithms include CLOSE and A-CLOSE [9].