Report on:
Expressive power of linear algebra query languages

I would like to remark that this is not a blind review. The conference paper from PODS 2021, of which this submission is an extended version, is well-known among researchers interested in the connections between linear algebra, database query languages, arithmetic circuits, and logic, and this reviewer has read that paper and has also seen a presentation of this work.

This is a fine paper. It extends the matrix query language MATLANG by a specific form of recursion, namely for-loops whose iterations are bounded by the dimensions of the given matrices. An important point of the precise definition of this iteration construct is that the iteration proceeds along the canonical unit vectors, which induces order information. Many of the results of the paper heavily rely on this order. It is worth to disuss this point this explicitly — as the authors indeed do — since the relation between logics and query languages (and their expressive power) on one side, and algorithmic models such as circuits and Turing machines (and their complexity) on the other side heavily depends on the presence or absence of an order. On ordered domains, this relationship is quite tight, since logical queries may then process their inputs along the given order and are thus are very closely related to algorithmic constructions. For instance, in descriptive complexity, it is well known how to design logical formalisms or query languages that precisely capture given complexity levels on ordered domains. In the absence of an order however, this is not possible. Abstract unordered structures may have symmetries, and logics treat symmetric (isomorphic) objects in the same way, and may not choose one of a collection of indistinguishable objects. This is a main obstacle for finding logics or query languages that capture specific complexity levels, such as polynomial time, on structures that do not come with a (definable) linear order. Thus, the construction of for-MATLANG, as proposed here, places this in the ordered domain. Indeed the authors succeed to define expressions in for-MATLANG for many familiar linear-algebraic algorithms, thus showing that this extension indeed provides a matrix query language that is able to deal with many important algorithmic tasks of linear algebra.

The main mathematical contribution, however, of this paper is a tight relationship between for-MATLANG and uniform families of arithmetic circuits of polynomial degree for functions operating on matrices. This is not completely surprising, given that for-MATLANG makes use of order information, but it is certainly a very interesting result.

The paper also contains a number of further results, on restrictions and variations of for-MATLANG, which give detailed insights in the expressive power of such query languages. The paper thus provides a number non-trivial and significant contributions to the database query languages in a linear algebraic setting.

The paper is also carefully presented, and it can essentially be published in its present form. I would however suggest to expand a bit the paragraph on **Related work** at the bottom of page three, and to give more details; in particular this concerns the remark that classical logics with aggregation and fixed-point logic with counting (FPC) can also be used for linear algebra. This point deserves a bit more attention and more references to the literature (for instance to the work of Anuj Dawar and his co-authors). It is known that FPC can express a large collection of linear-algebraic algorithms over fields of characteristic 0 whereas there are severe obstacles in fields with finite characteristics. Incidentally, it is a pity that the authors exclusively consider linear-algebraic problems over the field of reals. I would welcome if the authors could, at least briefly, discuss the power of for-MATLANG for other fields.