EMBEDDING FINITE LATTICES INTO THE COMPUTABLY ENUMERABLE DEGREES — A STATUS SURVEY

STEFFEN LEMPP, MANUEL LERMAN, AND REED SOLOMON

Abstract. We survey the current status of an old open question in classical computability theory: Which finite lattices can be embedded into the degree structure of the computably enumerable degrees? Does the collection of embeddable finite lattices even form a computable set?

Two recent papers by the second author show that for a large subclass of the finite lattices, the so-called join-semidistributive lattices (or lattices without so-called "critical triple"), the collection of embeddable lattices forms a Π^0_2-set.

This paper surveys recent joint work by the authors, concentrating on restricting the number of meets by considering "quasilattices", i.e., finite upper semilattices in which only some meets of incomparable elements are specified. In particular, we note that all finite quasilattices with one meet specified are embeddable; and that the class of embeddable finite quasilattices with two meets specified, while nontrivial, forms a computable set. On the other hand, more sophisticated techniques may be necessary for finite quasilattices with three meets specified.

1. Introduction

One of the longstanding open questions in classical computability theory is the characterization of all finite lattices embeddable into the computably enumerable (c.e.) degrees. This problem was first raised in the late 1960's but has up to now defied many attempts at a solution. At this point, it is even unclear whether a "reasonable" (e.g., decidable, or "purely lattice-theoretic") characterization exists. Progress has been steady over the past decade but very slow. In this paper, we will try to point out an approach to solving the problem which we consider

1991 *Mathematics Subject Classification.* Primary: 03D25.

Key words and phrases. lattice embedding, computably enumerable degrees.

The first author's research was partially supported by NSF grant DMS-9732526 and by the Vilas Foundation of the University of Wisconsin. The second author's research was supported by the University of Connecticut Research Foundation. The third author's research was partially supported by an NSF Postdoctoral Fellowship.

hopeful and which has led to some further, yet unpublished, partial results by the authors.

We note here that the lattice embeddings problem is currently the primary remaining obstacle toward showing the decidability of the ∀∃-theory of the c.e. degrees (in the language of partial ordering) as the former obviously forms a subproblem of the latter. If the lattice embeddings problem can be shown to have a decidable (and "reasonable") solution, then one would hope to show the remainder of the ∀∃-theory also to be decidable using the techniques of Slaman and Soare [14] in their solution of the extension of embeddings problem (i.e., given two finite partial orders $\mathcal{P} \subset \mathcal{Q}$, deciding whether any embedding of $\mathcal{P}$ into the c.e. degrees can be extended to an embedding of $\mathcal{Q}$) and of Ambos-Spies, Jockusch, Shore, and Soare [1] in their work on the promptly simple degrees (i.e., those c.e. degrees not forming half of a minimal pair).

To very briefly recap the history of the lattice embeddings problem up to this point, the first lattice embeddings result is contained in the minimal pair theorem of Lachlan [6] and Yates [17], which implies that the four-element diamond lattice can be embedded into the c.e. degrees. Lerman (unpublished) and Thomason [16] extended this by showing that all finite (indeed, all countable) distributive lattices can be so embedded. Lachlan [7] found the first two examples of finite embeddable nondistributive lattices, namely the five-element lattices M_3 and N_5. Lachlan and Soare [8], following a suggestion of Lerman, exhibited the first nonembeddable finite lattice, S_8 (see Figure 1). The best possible results obtainable by the techniques of the late 1980's were presented in Ambos-Spies and Lerman [2, 3], which isolated a Nonembeddability Condition (NEC) and an Embeddability Condition

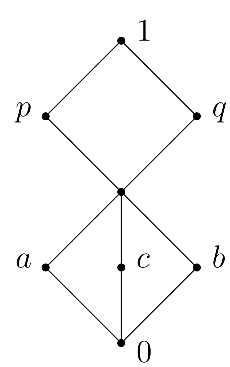

FIGURE 1. Lattice S_8

(EC), respectively. The latter condition, ensuring embeddability, is very complicated and formulated in terms of trees used to carry out the construction. The former condition, however, ensuring nonembeddability, is a simple lattice-theoretic condition.

To formulate NEC, we introduce the following

Definition 1. Let $\mathcal{L} = \langle L, \leq, \vee, \wedge \rangle$ be a finite lattice.

(1) Elements $a, b, c \in L$ form a *critical triple* if they are pairwise incomparable; $a \vee c = b \vee c$; and $a \wedge b \leq c$.

(2) $\mathcal{L}$ satisfies the *Nonembeddability Condition (NEC)* if there are a critical triple $a, b, c \in L$ and two additional incomparable elements $p, q \in L$ such that

$$a \leq p \wedge q \leq a \vee c \leq q. \tag{1}$$

(3) $\mathcal{L}$ is *principally decomposable* if for any two elements $a > b$ in L such that a is minimal over b, the set $[0, a] - [0, b]$ has a least element (where 0 is the least element of $\mathcal{L}$).

(4) $\mathcal{L}$ is *join-semidistributive* if for all $a, b, c \in L$,

$$a \vee c = b \vee c \text{ implies } a \vee c = (a \wedge b) \vee c. \tag{2}$$

These notions are closely connected by the following easy

Lemma 2. *A finite lattice $\mathcal{L}$ is join-semidistributive iff it is has no critical triple iff it is principally decomposable.*

From now on, we will use the term "join-semidistributive" in place of "principally decomposable" since the former is the one used by lattice theorists.

Remark 3. (1) There is a dual, but distinct notion called "meet-semidistributive". Figure 2 shows a finite lattice which is join-semidistributive but not meet-semidistributive.

(2) Join-semidistributive lattices form in some sense a class of finite lattices complementary to the modular lattices: Any finite join-semidistributive modular lattice is distributive. For more information on the lattice theory of semidistributive lattices, see Gorbunov [5].

The condition NEC thus pointed to the fact that the first step in solving the lattice embeddings problem was to consider the case of finite join-semidistributive lattices. When Downey [4] showed that there is an initial segment of c.e. degrees into which no finite non-join-semidistributive lattice can be embedded, he conjectured that on the other hand, every finite join-semidistributive lattice can be embedded into any nontrivial interval of the c.e. degrees.

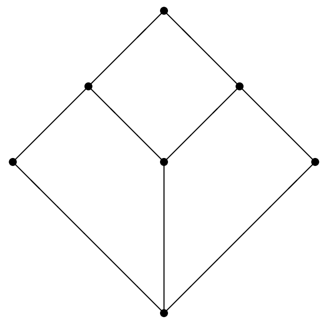

FIGURE 2. A join-semidistributive non-meet-semidistributive lattice

This conjecture was refuted by Lempp and Lerman [9], who exhibited a finite join-semidistributive lattice, L_{20} (see Figure 3), which cannot be embedded into the c.e. degrees. Further developing the techniques used for L_{20}, Lerman [11, 12] subsequently isolated a necessary and sufficient Π^0_2-criterion for the embeddability of finite join-semidistributive lattices

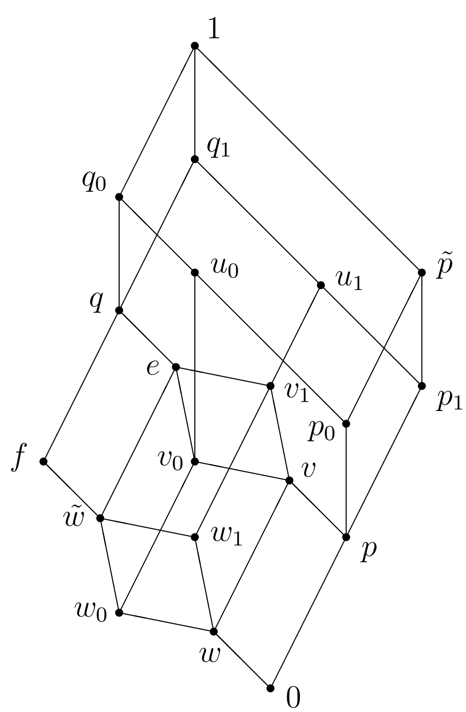

FIGURE 3. Lattice L_{20}

into the c.e. degrees. (Lerman [11] gives the partial result for finite so-called "ranked" lattices, which is then extended in Lerman [12] to the case of all finite join-semidistributive lattices.)

We defer the precise definition of Lerman's Embeddability Criterion to section 3.5 since we need to first introduce a number of additional definitions and also provide some intuition explaining the various conditions of Lerman's criterion.

First of all, however, we would like to point out the approach we have taken over the past several years in attacking the lattice embeddings problem. As will become clearer in the intuitive discussion of lattice embedding techniques in the next section, the hardest part is to ensure that meets are preserved under the embedding. It is therefore natural to take an inductive approach towards the lattice embedding problem by restricting the number of meets to be considered, motivating our definition of quasilattices. We first remark that any finite upper semilattice carries a natural lattice structure.

Remark 4. Any finite upper semilattice $\mathcal{L} = \langle L, \leq, \vee, 0, 1\rangle$ (with least element 0 and greatest element 1) can be made into a lattice by defining the meet function by

$$a \wedge b = \bigvee\{c \in L \mid c \leq a, b\}. \tag{3}$$

(The existence of 0 in L ensures that the set on the right-hand side above is always nonempty.)

We can now make the following

Definition 5. A *quasilattice* $\mathcal{L} = \langle L, \leq, \vee, \wedge, 0, 1\rangle$ (with least element 0 and greatest element 1) is an upper semilattice $\langle L, \leq, \vee\rangle$ together with a partial meet function $\wedge$ defined on some (but not necessarily all) unordered pairs of incomparable elements of L, where $a \wedge b$ (if defined) equals the meet defined by Remark 4.

Both a finite upper semilattice and a finite lattice are thus examples of quasilattices (where the meet is never or always defined, respectively). However, we will be most interested in examples where the meet is defined for a limited number of unordered pairs of incomparable elements, say, n many; we will call such a quasilattice a *quasilattice with n meets specified*.

We note that Lerman [11, 12] used closely related structures called pseudolattices. Lerman uses, in place of the partial meet functions, $(n+1)$-ary meet relations $M_n(a_1, \ldots, a_n, b)$, denoting that $c \leq a_i$ for all i implies $c \leq b$.

The current status of the lattice embeddings problem can thus be summarized in the following

Main Statement. (1) (Folklore) *Any finite quasilattice with no meets specified (i.e., any finite upper semilattice) is embeddable into the c.e. degrees.*

(2) *Any finite quasilattice with one meet specified is embeddable into the c.e. degrees.*

(3) *It is decidable which finite quasilattices with two meets specified are embeddable into the c.e. degrees; and not every finite quasilattice is so embeddable.*

(4) *It is currently unknown whether the characterization of the embeddable finite quasilattices with three meets specified is decidable. Our techniques developed for only two meets are not known to suffice to solve this problem.*

The rest of this paper is devoted to explaining, at least on an intuitive level, why we believe that our approach to the lattice embeddings problem via quasilattices is the most promising one; and to discuss the various clauses of our Main Statement. Specifically, we will address clause (1) of our Main Statement in section 2.4; clause (2) in section 3.4; clause (3) in section 3.7; and clause (4) in section 3.8.

We conclude this section by remarking that a large number of variations of the lattice embeddings problem into the c.e. degrees have been studied, too numerous to cover in detail here. Firstly, one can study lattice embeddings *preserving* 0 *and* 1, i.e., mapping 0 to the degree $\mathbf{0}$ and 1 to the degree $\mathbf{0}'$, respectively. (Any currently known "plain" lattice embedding into the c.e. degrees also preserves 0.) Next, one can study lattice embeddings into initial segments or intervals of c.e. degrees. Finally, lattice embeddings into the lattice of ideals of c.e. degrees have been studied.

2. Embedding Finite Upper Semilattices

We start with an intuitive description of the basic lattice embedding construction and the conflicts between the various strategies involved. Since the meet requirements are the most complicated ones, we first concentrate on the other requirements, i.e., we will present the argument for embedding finite upper semilattices (viewed as quasilattices with no meet specified). Fix a finite quasilattice $\mathcal{L}$ and its set $J_{\mathcal{L}}$ of join-irreducible elements. (Here 0 is not considered a join-irreducible element.)

Convention 6. In order to simplify notation from now on, we denote the c.e. degree which is the image of a lattice element a, say, by the corresponding bold-face letter $\mathbf{a}$, and the c.e. set representing the degree $\mathbf{a}$ by the corresponding upper-case letter A.

There are now four types of requirements to be satisfied for an embedding of $\mathcal{L}$ into the c.e. degrees:

$$\mathcal{C}^{a,b} : A \leq_T B \qquad \text{(for } a < b \text{ in } \mathcal{L}) \quad (4)$$

$$\mathcal{I}^{a,b}_{\Phi} : A \neq \Phi(B) \qquad \text{(for } a \not\leq b \text{ in } \mathcal{L};\ a \in J_{\mathcal{L}}) \quad (5)$$

$$\mathcal{J}^{d,e,f} : \exists\, \Gamma^{d,e} \left(F = \Gamma^{d,e}(D \oplus E) \right) \qquad \text{(for } f = d \vee e \text{ in } \mathcal{L}) \quad (6)$$

$$\mathcal{M}^{p,q,r}_{\Psi} : \Psi(P) = \Psi(Q) \text{ is total } \implies \exists\, \Delta^{p,q} \left(\Psi(P) = \Delta^{p,q}(R) \right) \qquad \text{(for } r = p \wedge q \text{ in } \mathcal{L}) \quad (7)$$

Here Φ and Ψ range over all possible Turing functionals. Clearly, the $\mathcal{C}$- and $\mathcal{J}$-requirements are "global", each building a single reduction (whose names we will suppress, except in the initial discussion of the $\mathcal{J}$-strategies below), whereas the $\mathcal{I}$- and $\mathcal{M}$-requirements are "local", each strategy on the tree of strategies working with a separate diagonalization witness or a separate functional $\Delta^{p,q}$, respectively.

We now gradually introduce the strategies for the four types of requirements and each time sketch the conflicts with the strategies discussed before. The discussion of the meet requirements will be deferred to the next section.

2.1. **Comparability Requirements $\mathcal{C}^{a,b}$.** For this requirement, we simply ensure that any number x targeted for a c.e. set A, say, is first chosen at a stage $< x$, and that when (if ever) x enters A, it simultaneously enters all sets B with $b > a$. Clearly, this simple strategy ensures the comparability requirements: Given $a < b$ in $\mathcal{L}$ and a number x, we first check if x is chosen with target C (for some $c \leq a$) by stage x. If not, then $x \notin A$; otherwise, $x \in A$ iff $x \in B$.

2.2. **Incomparability Requirements $\mathcal{I}^{a,b}_{\Phi}$.** The strategy for this requirement is simply the Friedberg-Muchnik strategy: We choose a "big" diagonalization witness x (i.e., larger than any number previously mentioned in the construction) and keep x out of A for now. We then wait for a computation $\Phi(B; x) = 0$. When (and if) such a computation appears, we enumerate x into A and preserve $\Phi(B; x)$ by restraining B up to its use.

The above two types of strategies present no serious conflict and can easily be combined to show that any finite partial order can be embedded into the c.e. degrees.

2.3. **Join Requirements $\mathcal{J}^{d,e,f}$.** When, at a stage s, a "big" number $x > s$ is targeted to be enumerated into a set C (with $c \leq f$, and so x may also be enumerated into F), then we also define a computation $\Gamma(D \oplus E; x)$ (for the functional $\Gamma = \Gamma^{d,e}$) with "big" use $\gamma_s(x)$. We now agree that

(1) the current use $\gamma(x)$ must enter D or E by the stage at which x enters C and thus must enter F;
(2) if the current use $\gamma(x)$ enters D or E at a stage s', say, before x enters C, then, at stage s', we redefine the computation $\Gamma(D \oplus E; x)$ with new "big" use $\gamma(x)$; and
(3) the use $\gamma(x)$ of x is increased at most finitely often.

Clearly, this will ensure the join requirement since we can define $\Gamma(D \oplus E; x) = 0$ for all x which are not targeted for some set C with $c \leq f$ by stage x. In this latter case, x cannot enter F, and so $\Gamma(D \oplus E)$ is correct on those x. If x is chosen as a target by stage x, then by (3) above, $\gamma(x)$ will eventually stabilize, and $D \oplus E$ can compute the stage when this happens by (2). Now $x \in F$ iff $\gamma(x) \in D \cup E$ by (1) for the final value of the use $\gamma(x)$.

Instead of being so explicit about the join functionals, however, we will now present an alternative way to deal with the comparability and join requirements simultaneously, which will also be useful in dealing with the meet requirements later on.

2.4. **Co-Principal Filters.** We start with the following

Definition 7. Let $\mathcal{L}$ be a finite quasilattice.

(1) A *filter* of $\mathcal{L}$ is any upward closed subset $F \neq L, \emptyset$ of L.
(2) The *filter generated by* a set $S \subset L$ (where $S \neq \emptyset$ and $0 \notin S$) is the upward closure of S in $\mathcal{L}$ and is denoted by (S). If $S = \{a_0, \dots, a_n\}$, we abbreviate $(\{a_0, \dots, a_k\})$ by $(a_0, \dots, a_k)$ or simply by $(\overrightarrow{a})$.
(3) A *co-principal filter* of $\mathcal{L}$ is a filter F such that $L - F$ is closed under join, or equivalently, such that $L - F$ is of the form $[0, b]$ for some $b \in L - \{1\}$. For any $b \in L - \{1\}$, we denote the co-principal filter $L - [0, b]$ by $F(b)$.

Note that as long as we enumerate numbers of "roughly equal" size into all sets C (for all c in some co-principal filter F) then we can build the join functionals Γ for all possible joins as above. This observation allows one to establish clause (1) of our Main Statement, namely, to show that all finite upper semilattices can be embedded into the c.e. degrees by combining the above three types of strategies as follows: When an $\mathcal{I}^{a,b}$-strategy chooses a witness x targeted for A, it will also target the same number x for all sets C with $c \not\leq b$. Now since $F(b) = L - [0, b]$ forms a co-principal filter in $\mathcal{L}$, it is easy to check that if the $\mathcal{I}^{a,b}$-strategy enumerates x into all sets C with $c \in F(b)$ when x enters A, then each $\mathcal{J}^{d,e,f}$-requirement is satisfied, since $f \in F(b)$ iff $d \in F(b)$ or $e \in F(b)$, so x is targeted, and possibly later enters, F iff x enters D or E.

2.5. **Covering Sequences and Covering Arrays.** Since we will also have to consider meet requirements later on, we now restrict our attention to embedding finite join-semidistributive lattices. (This will allow us to present the Embeddability Criterion of Lerman [11, 12] in a somewhat modified and simplified form.)

First of all, when taking into consideration meet requirements, it will not always be possible to enumerate diagonalization witnesses in one step as sketched two paragraphs above; rather, we will have to "retarget" a number of times as indicated in the general description of the join functional Γ above. To simultaneously deal with all join requirements, and to simplify the description, we introduce two key notions in the following

Definition 8. Let $\mathcal{L}$ be a finite lattice.

A *covering sequence* is an ordered sequence $\overrightarrow{a} = \langle a_0, \dots, a_l \rangle$ of elements of L such that $(a_0, \dots, a_i)$ is a co-principal filter for all $i \leq l$. (We allow $l = -1$, i.e., a covering sequence may be empty.)

Covering sequences provide an easy method to ensure the satisfaction of all comparability and join requirements intuitively as follows: When an $\mathcal{I}_\Phi^{a,b}$-strategy chooses a diagonalization witness x, then

(I) the strategy chooses a covering sequence $\langle a_0, \dots, a_l \rangle$ with $a = a_l$ and $b \notin (a_0, \dots, a_l)$, and chooses associated "big" numbers x_i for all $i \leq l$ where $x_l = x$;

(II) if the strategy enumerates any associated number before enumerating x into A, then, for some $l'' <$ the current l, it enumerates, for all $i \leq l''$, the currently associated numbers x_i into all sets C with $c \geq$ the current a_i, and chooses a new covering sequence $\langle a'_0, \dots, a'_{l'} \rangle$ (which contains the tail $\langle a_{l''+1}, \dots, a_l \rangle$ of the current covering sequence as a subsequence) as well as "big" associated numbers x'_i for all "new" elements a'_i of the new covering sequence (whereas for all "old" elements a'_i of the new covering sequence, the currently associated number remains associated with a'_i);

(III) if the strategy enumerates x into A then it enumerates, for all $i \leq$ the current l, the currently associated number x_i into all sets C with $c \geq$ the current a_i; and

(IV) it chooses a new covering sequence (as in (II) above) at most finitely often.

It is now not hard to see that the above (I)–(IV) will ensure the incomparability and join requirements since, by the definition of covering sequences, the associated numbers can be viewed as uses of join functionals Γ.

We make this more precise in the following

Definition 9. A *covering array* consists of a sequence

$$\overrightarrow{A} = \langle \overrightarrow{a_0}, \dots, \overrightarrow{a_m} \rangle \tag{8}$$

of target sequences $\overrightarrow{a_j} = \langle a_{0,j}, \dots, a_{l_j,j} \rangle$ together with a sequence of *transition maps* $\langle T_0, \dots, T_{m-1} \rangle$ such that for each $j < m$, T_j is a map from $(l'_j, l_j]$ into $[0, l_{j+1}]$ (for some $l'_j < l_j$) satisfying, for all $j < m$,

$$l'_j < i < i' \leq l_j \implies T(i) < T(i'), \text{ and} \tag{9}$$

$$\forall i \in (l'_j, l_j] \Big(a_{i,j} = a_{T_j(i),j+1} \Big). \tag{10}$$

(So each T_j is an order-preserving map from a final segment of (indices of) $\overrightarrow{a_j}$ to (indices of) $\overrightarrow{a_{j+1}}$, preserving the lattice element $a_{i,j}$. Here we allow $l_j = -1$ only if $j = m$, i.e., only the last covering sequence of the covering array may be empty. We abbreviate the composition $T_{j_1-1} \cdots T_{j_0}$ (for $0 \leq j_0 \leq j_1 \leq m$) by T_{j_0,j_1}.)

We say that $\overrightarrow{A}$ is a *covering array for an $\mathcal{I}^{a,b}_\Phi$-requirement* if furthermore $a = a_{l_0,0}$ and $b \notin (a_{0,0}, \dots, a_{l_0,0})$. (I.e., a is the last element of the first covering sequence in $\overrightarrow{A}$, and the set B is not initially targeted, namely, not before a computation $\Phi(B; x) = 0$ has been found. Of course, once such a computation has been found, we can target new numbers y for sets C with $c \leq b$ since such y can be chosen above the use of the computation $\Phi(B; x)$.)

The above clauses (I)–(IV) for satisfying all comparability and join requirements can now be phrased as follows: All enumerations of an $\mathcal{I}^{a,b}$-strategy, once it chooses a diagonalization witness x, correspond to a covering array $\overrightarrow{A} = \langle \overrightarrow{a_0}, \dots, \overrightarrow{a_m} \rangle$ for this requirement in the following sense: When the $\mathcal{I}$-strategy chooses a new "big" diagonalization witness x targeted for A at a stage s_0, say, it also chooses new "big" numbers $x_{i,0}$ targeted for $A_{i,0}$. Once a computation $\Phi(B; x) = 0$ has been found, the $\mathcal{I}$-strategy enumerates into sets and chooses new numbers in m many steps: At step $j > 0$ of this process, say, at a stage s_j, any number $x_{i,j-1}$ such that $i \notin \operatorname{dom} T_{j-1}$ is enumerated into its target $A_{i,j-1}$ (and thus into all sets C with $c \geq a_{i,j-1}$). Also, for any $i \leq l_j$, if $i \in \operatorname{ran} T_{j-1}$ then we set $x_{i,j} = x_{T^{-1}_{j-1}(i),j-1}$; if $i \notin \operatorname{ran} T_{j-1}$ then we choose a new "big" number $x_{i,j}$ targeted for $A_{i,j}$.

It is now easy to verify that the use of the covering array for the $\mathcal{I}^{a,b}_{\Phi}$-requirement will also satisfy any $\mathcal{J}^{d,e,f}$-requirement: Suppose a number $y = x_{i,j_0}$ is targeted for some set C with $c \leq f$ at some stage $s_{j_0} < y$, say. Consider the sequence

$$\langle x_{i,j_0},\ x_{T_{j_0}(i),j_0+1},\ x_{T_{j_0,j_0+2}(i),j_0+2},\ \ldots,\ x_{T_{j_0,j_1}(i),j_1}\rangle \tag{11}$$

such that $j_1 = m$ or $T_{j_0,j_1}(i) \notin \operatorname{dom} T_{j_1}$. Since each $\overrightarrow{a_j}$ is a covering sequence, we have that for all $j \in [j_0, j_1]$, there is some $i_j < T_{j_0,j}(i)$ with $a_{i_j,j} \leq d$ or $\leq e$. So we can define $\Gamma^{d,e}(D \oplus E; x)$ with use $\gamma^{d,e}(y) = x_{i_j,j}$, and the clauses (1)–(3) of subsection 2.3 will hold. (We reiterate here the remark that the above technique only works for finite join-semidistributive lattices. For finite lattices in general, without the assumption of principal decomposability, the notion of covering sequence has to be generalized, requiring that only certain, but not all, initial segments of the sequence generate co-principal filters. E.g., even in the example of arbitrary finite upper semilattices at the end of subsection 2.4, it may not be possible to use covering sequences to generate co-principal filters of the form $L-[0, b]$ as outlined there, as the example of the lattice M_3, viewed as an upper semilattice, illustrates.)

3. Embedding Finite Quasilattices

We are now ready to add the meet requirements, in the context of covering sequences and covering arrays as defined above.

3.1. Meet Requirements $\mathcal{M}^{p,q,r}_{\Psi}$.

The basic strategy for a meet requirement is quite simple even though its interaction with the other requirements is very complicated: As the length of agreement between $\Psi(P)$ and $\Psi(Q)$ increases, the strategy defines more and more of $\Delta(R)$ by initially setting $\Delta(R; y)$ to the common value of $\Psi(P; y)$ and $\Psi(Q; y)$ and setting the use $\delta(y) \geq$ the uses $\psi(P; y)$ and $\psi(Q; y)$. Whenever $\Delta(R; y)$ is defined for some argument y, the strategy tries to have at least one of $\Psi(P; y)$ or $\Psi(Q; y)$ defined and agreeing with $\Delta(R; y)$. If that fails, then the strategy must destroy the computation $\Delta(R; y)$ by enumerating a number $\leq \delta(y)$ into R.

The main difficulty in the above strategy is the following typical scenario: After $\Delta(R;y)$ has become defined, both computations $\Psi(P;y)$ and $\Psi(Q;y)$ may become undefined, although never at the same time. Each may now return with the same value but a larger use, so the strategy outlined in the previous paragraph sees no reason to act yet. However, this creates the so-called "dangerous interval"

$$I_y = \Big(\delta(y), \min\{\psi(P;y), \psi(Q;y)\}\Big], \tag{12}$$

dangerous for $\Delta(R)$ since the enumeration of any number $z \in I_y$ into R (and thus into both P and Q by comparability requirements) will destroy both $\Psi(P;y)$ and $\Psi(Q;y)$ but not $\Delta(R;y)$. This now makes it necessary to enumerate another number $z' \leq \delta(y)$ into R to correct $\Delta(R;y)$. However, this z' may be in a dangerous interval $I_{y'}$ for some $y' < y$, possibly setting off a cascade of smaller and smaller numbers having to enter R until no dangerous interval is hit. (It is exactly this type of behavior which was at the heart of the proof of the nonembeddability of the lattice S_8 by Lachlan and Soare [8]. However, S_8 is not a join-semidistributive lattice, and a nonembeddability construction for join-semidistributive lattices, such as for the nonembeddable lattice L_{20} discovered by Lempp and Lerman [9], has to use dangerous intervals in a more subtle way.)

Since the incomparability strategies (which are, as we have now seen, the only ones *initiating* the enumeration of numbers into sets) are all finitary, we agree that we will never allow enumeration of any numbers y into dangerous intervals, but rather always directly enumerate the largest number $y' \leq y$ which does not hit a dangerous interval, thus never triggering the kind of cascade of enumerations described in the previous paragraph. This restriction will be implemented by the prohibition functions defined below in section 3.5 where we will also take into account the interaction between dangerous intervals of different meet strategies. However, even though only the incomparability strategies will *initiate* the enumeration of numbers into sets, the meet strategies may respond to other enumerations by enumerating numbers on their own, namely so-called *correction markers* to correct their functionals Δ. We will show that this can only happen for finite join-semidistributive quasilattices with at least three meets specified. However, up to this point, we do not know if correction markers are necessary at all, or whether correction will always be automatic by numbers entering purely for coverage reasons. (In the latter case, we would have a decision procedure for a fixed number of "gates", as we will outline in section 3.8.)

First of all, however, we will set up the machinery of pinball machine constructions and blocks which will allow us to make precise the implementation of the meet strategies.

3.2. **Pinball machine constructions.** Lattice embedding constructions are traditionally done using the so-called pinball machine construction introduced by Lerman [10] (see also Soare [15, Ch. VIII.5]). The rough idea is the following: Diagonalization witnesses are represented by "balls" originating from "holes" (corresponding to incomparability, i.e., $\mathcal{I}_{\Phi}^{a,b}$-requirements) which then have to pass by "gates" (corresponding to higher-priority meet requirements). Balls may either get permanently stuck at (i.e., be permanently restrained by) one of the gates below the hole, or they may pass by all of the finitely many gates below the hole and enter the "enumeration basket" (i.e., be enumerated into their target set). In addition to the balls corresponding to diagonalization witnesses, we need other balls (i.e., numbers) to "cover" the diagonalization witnesses (i.e., to generate a co-principal filter containing a). These other balls either originate at the same hole as the diagonalization witness (and then correspond to the elements of $\overrightarrow{a_0}$); or they originate at gates below a hole to "cover" balls currently at or above that gate now that some of the previously covering balls may have been enumerated. These latter balls correspond to "new" elements $a_{i,j}$ (for $j > 0$) of covering sequences $\overrightarrow{a_j}$ (i.e., for which $T_{j-1}^{-1}(i)$ is undefined).

Before explaining how the pinball machine construction helps satisfy the meet requirements, we first explain some of the simple mechanics of how the gates and the covering arrays interact.

3.3. **Blocks.** The main tool to combine the target array with the pinball machine construction is the notion of blocks. (We deviate here somewhat from the way Lerman [11, 12] defines blocks by slightly changing the definition of the functions h_k and $h_{j,k}$; however, our definition here is equivalent.)

Definition 10. Fix $n > 0$. (The intuition here will be that G_0 through G_{n-1} are gates (with gate G_0 the lowest, corresponding to the highest-priority requirement) below a hole H_n. For now, however, these gates can be viewed simply as giving us indices for the block functions.)

(1) A covering sequence $\langle a_0, \dots, a_l \rangle$, together with a function $f : [0, l] \to [0, n]$ and partial functions $h_k : [0, l] \to [0, l]$ (for $k < n$), forms a *blocked target sequence* if for all $k < n$,

$$\operatorname{ran} f = \{n\} \text{ or } \forall i, i' \leq l \left(i < i' \implies f(i) \leq f(i') < n \right); \tag{13}$$

$$\operatorname{dom} h_k = \{i \leq l \mid f(i) \geq k\}; \tag{14}$$

$$\forall i \in \operatorname{dom} h_k \Big(i \leq h_k(i)\Big); \tag{15}$$

$$\forall i, i' \in \operatorname{dom} h_k \Big(i < i' \implies h_k(i) \leq h_k(i')\Big); \tag{16}$$

$$\forall i, i' \in \operatorname{dom} h_k \Big(h_k(i) < h_k(i') \implies h_k(i) < i'\Big); \tag{17}$$

$$k < n-1 \implies \forall i, i' \in \operatorname{dom} h_k \Big(h_k(i) = h_k(i') \implies h_{k+1}(i) = h_{k+1}(i')\Big); \text{ and} \tag{18}$$

$$\forall i, i' \in \operatorname{dom} h_k \Big(h_k(i) = h_k(i') \implies f(i) = f(i')\Big). \tag{19}$$

(Intuitively, f indicates that the "ball" (number) x_i associated with a_i currently is at gate $G_{f(i)}$ (if $f(i) < n$), or at the hole H_n of the diagonalization requirement (if $f(i) = n$). Clause (13) now states that the balls either all reside at the hole H_n, or all reside at gates such that later balls in the sequence do not reside at lower gates. Each function h_k induces a partition of the balls at or above gate G_k into intervals called k-*blocks*, where $a_{h_k(i)}$ indicates the last element of the k-block; this is ensured by clauses (14)–(17). Clause (18) indicates that the k-blocks refine the $(k+1)$-blocks, while clause (19) states that any k-block resides at a single gate or hole. The intuition here is that each k-block consists of balls which pass gate G_k simultaneously (which explains the choice of the domain of h_k).) We denote the eth k-block of $\langle a_0, \dots, a_l\rangle$ by B^e_k, starting with $e = 0$.

(2) A covering array $\overrightarrow{A} = \langle \overrightarrow{a_0}, \dots, \overrightarrow{a_m}\rangle$ with transition functions $T_0, \dots, T_{m-1}$ (where each of the covering sequences $\overrightarrow{a_j} = \langle a_{0,j}, \dots, a_{l_j,j}\rangle$ is a blocked target sequence with functions f_j and $h_{j,k}$ for all $k < n$)) forms a *blocked target array* if for all $k < n$,

$$\forall j \leq m \,\forall i \leq l_j \Big(f_j(i) = n \iff j = 0\Big); \tag{20}$$

$$l_m = -1; \tag{21}$$

$$\forall j < m \,\forall i \in [0, l_j] - B^0_{j, f_j(0)} \Big(i \in \operatorname{dom} T_j \text{ and } f_j(i) = f_{j+1}(T_j(i))\Big); \tag{22}$$

$$\forall j < m \,\forall i \in B^0_{j, f_j(0)} \Big(i \notin B^0_{j,0} \iff i \in \operatorname{dom} T_j\Big); \tag{23}$$

$$\forall j < m \,\forall i \in B^0_{j, f_j(0)} - B^0_{j,0} \Big(f_{j+1}(T_j(i)) = \min\{k-1 \mid i \in B^0_{j,k}\}\Big); \text{ and} \tag{24}$$

$$\forall j < m \, \forall i, i' \in \operatorname{dom} T_j \Big(T_j(i), T_j(i') \in \operatorname{dom} h_{j+1,k} \implies (h_{j,k}(i) = h_{j,k}(i') \iff h_{j+1,k}(T_j(i)) = h_{j+1,k}(T_j(i')) \Big). \tag{25}$$

(Here clause (20) states that only the balls corresponding to $\overrightarrow{a_0}$ are at the hole; all balls corresponding to later $\overrightarrow{a_j}$ are at gates. Clause (21) states that only the last covering sequence $\overrightarrow{a_m}$ is empty. Clauses (22)–(24) exactly prescribe the motion of the balls in the pinball machine from step j to step $j+1$: Set $k_j = f_j(0)$, which is the (index of the) lowest gate containing a ball at step j. Now any ball not in the first k_j-block B^0_{j,k_j} at G_{k_j} remains at the same gate at which it was by clause (22); the balls in the first 0-block $B^0_{j,0}$ at G_{k_j} are enumerated by clause (23); and the balls in $B^0_{j,k_j} - B^0_{j,0}$ move down to gate G_{k-1} iff they are in $B^0_{j,k} - B^0_{k-1}$ by clause (24). (This is more restrictive than the definition of Lerman [11, 12], but by Lerman's proof, it still gives an embeddability criterion for finite join-semidistributive lattices.) Finally, clause (25) states that k-blocks are preserved from step j to step $j+1$ unless a ball is no longer in a k-block at step $j+1$, i.e., is already below gate G_k.)

(3) A blocked target array is a *blocked target array for an $\mathcal{I}^{a,b}_\Phi$-requirement* if furthermore $a = a_{l_0,0}$ and $b \notin (a_{0,0}, \dots, a_{l_0,0})$.

3.4. **A Single Meet $p \wedge q = r$.** We are now ready to consider in detail our first argument involving meet strategies. Following the philosophy of the introduction of this paper, we start by describing how to deal with a single meet in a quasilattice. Since the incomparability strategies (which are the only ones initiating the enumeration of numbers into sets) are finitary, it suffices to consider the interaction of a single $\mathcal{I}^{a,b}_\Phi$-requirement with a finite number of higher-priority $\mathcal{M}^{p,q,r}$-strategies (in the context of all comparability and join requirements). We first restrict ourselves to the case of a single gate G_0 since several gates for the same meet present no additional difficulties. We need to distinguish three cases, depending on the position of a and b relative to p, q, and r.

Case 1: $r \not\leq b$: Then there is no conflict since the enumeration of the diagonalization witness x into A can take place immediately upon finding a computation $\Phi(B; x) = 0$ and simultaneously with the enumeration of some number $x_{i,0}$ into a set $A_{i,0}$ where $a_{i,0} \leq r$ since $x_{i,0}$ can correct the meet functional $\Delta^{p,q}(R)$. The covering array can thus be chosen as $\overrightarrow{A} = \langle \overrightarrow{a_0}, \overrightarrow{a_1} \rangle$ with $\overrightarrow{a_0}$ consisting of a single 0-block and $\overrightarrow{a_1}$ empty where $r \in (a_{0,0}, \dots a_{l_0,0})$, i.e., all numbers $x_{i,0}$ associated with $a_{i,0}$ are enumerated immediately.

Case 2: $p \leq b$ (or symmetrically $q \leq b$): Then again there is no conflict since the enumeration of the diagonalization witness x into A can take place immediately upon finding a computation $\Phi(B; x) = 0$ and the functional $\Psi(P)$ will not be injured. The covering array can thus be chosen as $\overrightarrow{A} = \langle \overrightarrow{a_0}, \overrightarrow{a_1} \rangle$ with $\overrightarrow{a_0}$ consisting of a single 0-block and $\overrightarrow{a_1}$ empty where $p \notin (a_{0,0}, \dots a_{l_0,0})$, i.e., all numbers $x_{i,0}$ associated with $a_{i,0}$ are enumerated immediately.

Case 3: $r \leq b$ but $p, q \not\leq b$: This is the nontrivial case since the enumeration of the diagonalization witness x into A will typically require the enumeration of numbers into both P and Q while the preservation of the computation $\Phi(B; x) = 0$ does not allow the immediate enumeration into R. We resolve this problem by splitting the first covering sequence $\overrightarrow{a_0}$ with $a = a_{l_0,0}$ into 0-blocks $B^e_{0,0}$ such that for each e,

$$p \notin (B^e_{0,0}) \text{ or } q \notin (B^e_{0,0}). \tag{26}$$

(This can certainly be achieved by making each 0-block consist of a single element, but it probably makes more sense to use maximal 0-blocks satisfying clause (26).)

The second covering sequence $\overrightarrow{a_1}$ is now obtained from $\overrightarrow{a_0}$ by (i) deleting the first 0-block $B^0_{0,0}$ of $\overrightarrow{a_0}$; (ii) copying each subsequent 0-block $B^e_{0,0}$ of $\overrightarrow{a_0}$ (for $e > 0$) into the 0-block $B^{e-1}_{1,0}$ of $\overrightarrow{a_1}$; and (iii) adding a covering sequence generating $F(p)$ or $F(q)$ at the beginning of each 0-block $B^e_{1,0}$ depending on whether the remainder of $B^e_{1,0}$, coming from $B^{e+1}_{0,0}$, is contained in $F(p)$ or $F(q)$, respectively. The remaining covering sequences $\overrightarrow{a_j}$ (for $j > 1$) are now obtained from the previous covering sequence $\overrightarrow{a_{j-1}}$ by simply deleting the first 0-block of $\overrightarrow{a_{j-1}}$ until we end up with the empty sequence. Recall that the corresponding enumeration enumerates a number $x_{i,j}$ into $A_{i,j}$ when $T_j(i)$ is undefined, i.e., when $a_{i,j}$ is deleted from the sequence between step j and step $j + 1$.

This ensures that at each step of the enumeration, either $\Psi(P)$ or $\Psi(Q)$ will not be injured.

We illustrate the above with the example of the quasilattice in Figure 4.

Example 11. The covering array for a hole H_1 (for an incomparability requirement $\mathcal{I}^{a,b}_{\Phi}$) above a gate G_0 (for a meet requirement $\mathcal{M}^{p,q,r}_{\Psi}$)

consists of the following covering sequences

$$\begin{array}{r} \langle a_0, \quad\ \ a_1, \quad\ \ a_2, \quad\ \ a\rangle \\ \langle p', a_1, q', a_2, p', a\rangle \\ \langle q', a_2, p', a\rangle \\ \langle p', a\rangle \\ \langle\ \rangle \end{array}$$

where the transition maps are defined as indicated by vertical alignment. This uses that $a_1, a \in F(p) = [p', 1]$ whereas $a_0, a_2 \in F(q) = [q', 1]$.

Since the above can be generalized to n many gates G_0 through G_{n-1} for the same meet $p \wedge q = r$ by simply making all 0-blocks simultaneously also k-blocks for all $k < n$, this simple construction shows that any finite quasilattice with only one meet specified can be embedded into the c.e. degrees. This establishes clause (2) of our Main Statement.

Before we can address the embedding of quasilattices with more than one meet specified, we need to introduce the notion of a prohibition function, which will help us identify and avoid enumeration into dangerous intervals.

3.5. Prohibition functions and Lerman's Embeddability Criterion.

For each blocked target array $\overrightarrow{A} = \langle \overrightarrow{a_0}, \ldots, \overrightarrow{a_m}\rangle$, we define prohibition functions g_j (for each $j \leq m$, corresponding to each blocked

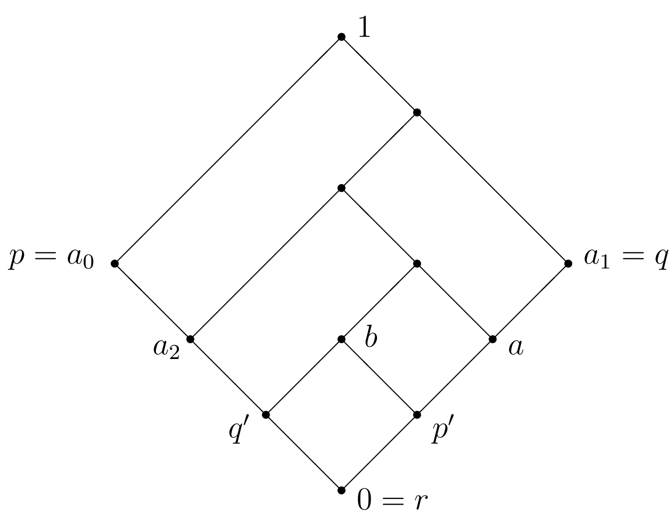

FIGURE 4. A quasilattice with one meet specified

target sequence $\overrightarrow{a_j}$), which associates each element $a_{i,j}$ of $\overrightarrow{a_j}$ with a subset $g_j(i)$ of $\{G_0, \dots, G_{n-1}\}$. (Here, each g_j will depend only on $\overrightarrow{A}$, and more specifically only on $\langle \overrightarrow{a_0}, \dots, \overrightarrow{a_j} \rangle$. Intuitively, $G_k \in g_j(i)$ will tell us that we cannot currently target new balls $\leq r_k$ at or before the k-block of $a_{i,j}$. Once the prohibition functions have been defined, we will define the notion of a *good* blocked target array, which implements this prohibition. This approach has the advantage that the various features of meet strategies in the literature ("minimal pair strategy", "Lachlan meet strategy", etc.) are unified into one single definition.)

Definition 12. Fix a blocked target array $\overrightarrow{A} = \langle \overrightarrow{a_0}, \dots, \overrightarrow{a_m} \rangle$. For each element $a_{i,j}$, we will have $g_j(i) \subseteq \{G_0, \dots, G_{n-1}\}$, so we fix a gate G_k (corresponding to a meet $p_k \wedge q_k = r_k$, for some $k < n$) and define whether $G_k \in g_j(i)$ by induction on $j \leq m$.

Let $k_j = f_j(0)$, which is the (index of the) lowest gate containing a ball at step j. If $j > 0$ and $i \in \operatorname{dom} h_{j,k}$ then fix $i_k \leq l_{j-1}$ maximal such that $T_{j-1}(i_k)$ is undefined or $\leq h_{j,k}(i)$. (I.e., $a_{i_k,j-1}$ is last element of the "preimage k-block" in $\overrightarrow{a_{j-1}}$ of the k-block of $a_{i,j}$ in $\overrightarrow{a_j}$. We leave i_k undefined if $i \notin \operatorname{dom} h_{j,k}$.)

Adding a gate: We add G_k to $g_j(i)$ iff $k < k_j$ and both $p_k, q_k \in (a_{0,j}, \dots, a_{h_{j,k}(i),j})$ but $r_k \notin (a_{0,j}, \dots, a_{h_{j,k}(i),j})$.

Deleting a gate: Suppose $j > 0$. Then we delete $G_k \in g_{j-1}(i_k)$ from $g_j(i)$ iff $k \leq k_{j-1}$ and at least one of p_k and q_k is not in $(a_{0,j}, \dots, a_{h_{j,k}(i),j})$.

Otherwise: If G_k is not added into, or deleted from, $g_j(i)$ by one of the above clauses, then $G_k \notin g_j(i)$ (if $j = 0$), or $G_k \in g_j(i)$ iff $G_k \in g_{j-1}(i_k)$ (if $j > 0$, respectively).

Note that whether $G_k \in g_j(i)$ depends only the k-block of $a_{i,j}$ since the definition of $g_j(i)$ only uses $h_{j,k}(i)$ but never i itself. Note also that whether $G_k \in g_j(i)$ can only change (from whether $G_k \in g_{j-1}(i_k)$) when there are no balls at G_0 through G_{k-1}, i.e., at an "expansionary stage" for G_k.

Intuitively, we add a gate G_k into $g_j(i)$ when the gates G_0 through G_k contain no balls and when the k-blocks up the k-block of $a_{i,j}$ target sets below both P_k and Q_k but not R_k, i.e., these k-blocks destroy both $\Psi_k(P_k)$ and $\Psi_k(Q_k)$ while not allowing the correction of $\Delta_k(R_k)$ at an expansionary stage for G_k when G_k should be extending the definition of $\Delta_k(R_k)$ since there are no balls at G_0 through G_k. We delete $G_k \in g_{j-1}(i_k)$ from $g_j(i)$ when the gates G_0 through G_{k-1} contain no balls and when the k-blocks up to the k-block of $a_{i,j}$ target no sets below P_k or no sets below Q_k, i.e., these k-blocks do not destroy one of $\Psi_k(P_k)$ and $\Psi_k(Q_k)$ at an expansionary stage for G_k. Otherwise, we leave $G_k \in g_j(i)$ iff $G_k \in g_{j-1}(i_k)$. This corresponds to the meet strategy outlined in section 3.1.

Definition 13. A blocked target array $\overrightarrow{A}$ is a *good blocked target array* if for any gate G_k (corresponding to a meet $p_k \wedge q_k = r_k$, for some $k < n$) with $k \leq k_j$ (where $k_j = f_j(0)$ is the (index of the) lowest gate containing a ball at step j), $G_k \notin g_j(0)$.

Note that by Definition 12, if $G_k \in g_j(0)$ while $k \leq k_j$ then both $p_k, q_k \in (B^0_{j,k})$, with, by Definition 12, (hereditarily) smaller numbers targeted for both P_k and Q_k than any number possibly targeted for R_k. Thus we cannot allow $G_k \in g_j(0)$ while $k \leq k_j$, as stated in Definition 13.

We can now state in full detail

Lerman's Embeddability Criterion. *(Lerman [11, 12]) A finite join-semidistributive lattice (or quasilattice) is embeddable into the c.e. degrees iff for any sequence of gates (corresponding to meet requirements, allowing repetition) and any hole (corresponding to an incomparability requirement), there is a good blocked target array.*

Lerman's Embeddability Criterion thus provides a Π^0_2-condition for the embeddability of finite join-semidistributive lattices, with the universal quantifier ranging over sequences of gates and the existential quantifier ranging over good blocked target arrays. Bounding these two quantifiers would yield an effective condition and thus a decision procedure.

3.6. **Two Gates for Two Meets $p_0 \wedge q_0 = r_0$ and $p_1 \wedge q_1 = r_1$.** Here we will encounter a sketch of the first nonembeddability proof, for a lattice we call L_{14}. We will also give a decidable (although rather complicated) criterion for the embeddability of finite quasilattices with two meets specified.

We first consider an incomparability strategy (hole H_2) having to deal with two higher-priority meet strategies (gates G_0 and G_1, one for each meet). We again distinguish cases, first depending on the position of a and b relative to p_1, q_1, and r_1:

Case 1: $r_1 \not\leq b$ or $p_1 \leq b$ or $q_1 \leq b$: Then the entire initial covering sequence $\overrightarrow{a_0}$ will be a single 1-block, i.e., immediately pass by gate G_1 and go on to gate G_0, where we will proceed as in the case of one meet (i.e., as in section 3.4).

Case 2: $r_1 \leq b$ and $p_1, q_1 \not\leq b$: We first handle a special subcase:

Case 2.1: $r_0 \leq b$: We begin by stating some simplifying assumptions we can make:

(1) It is useless to *add* an element $c \leq r_k$ to a block at gate G_k (for $k = 0, 1$) since this is only allowed, by Definition 13, when there is currently no element $\leq p_k$ or no element $\leq q_k$ in the blocked target sequence up to the k-block of c. Note that, by Definition 12, if such c is not added to the first k-block at G_k but to a later one, then the first k-block through the k-block of c can be combined into one k-block; but adding such c to the first k-block at G_k gives no advantage to providing coverage for the rest of the covering sequence. (However, it is possible to *add* some $c \leq r_1$ at G_0 without any restrictions since corresponding balls will be above any G_1-restraint; and to *add* some $c \leq r_0$ at G_1 as long as the goodness of the blocked target array is not violated.)

(2) Whenever there are balls at G_0, we may assume that, as in section 3.4, they are arranged in 0-blocks each beginning with a covering sequence for all of $F(p_0)$ or $F(q_0)$. This is since, by Definition 12, this cannot violate the goodness of the covering array and cannot otherwise restrict the covering sequence at G_0 or G_1. We may also assume the 0-blocks B at G_0 to alternatingly satisfy $(B) = F(p_0)$ or $(B) = F(q_0)$.

(3) Whenever there are at least two 0-blocks, B^0 and B^1, say, at G_0 then there is no need to add new elements at G_1 since $(B_0 \cup B_1) = F(p_0) \cup F(q_0) = F(r_0)$, which will cover anything allowed at G_1.

(4) Whenever an element $c \leq r_0$ is added at G_1, then we may add all of $F(r_1)$ (which by (1) is the maximal filter we can use) together with c. This is because once we target below r_0 at G_1 (and since we may not target below r_1 at G_1), targeting below all of $F(r_1)$ cannot violate the goodness of the blocked target array.

The above now allow us to define an effective decision procedure to decide whether, given two gates below one hole, there is a good blocked target array: There are only finitely many choices as to what to add at gate G_1 since duplicating elements at G_1 only helps if one targets below r_0; but then we can use all of $F(r_1)$ by remark (4) above. And by remark (1) above, the number of choices of what to add at gate G_0 is also effectively bounded, so there is an overall bound on the size of a potential good blocked target array, giving an effective decision procedure for its existence.

We illustrate the above with two examples, which analyze two quasilattices with two meets specified that are the same except for the position of one top p_1 of one meet. Surprisingly, this small difference makes one embeddable while making the other nonembeddable.

Example 14. Figure 5 shows an embeddable quasilattice with the two meets $p_0 \wedge q_0 = r_0$ and $p_1 \wedge q_1 = r_1$ specified. The good blocked target array for a hole H_2 (for an incomparability requirement $\mathcal{I}^{a,b}_{\Phi}$) above the gates G_0 and G_1 (for the corresponding meet requirements) consists of the following blocked target sequences

$$\begin{array}{r}
\langle m_{11},\quad m_{01} \qquad\quad ;\quad p_1; \qquad\quad a\rangle \\
\langle s\, m_{01}, m_{11}\, p_1;\quad p_1; \qquad\quad a\rangle \\
\langle m_{11}\, p_1;\quad p_1; \qquad s\, a\rangle \\
\langle r_0\, p_1; m_{11}, s\, a\rangle \\
\langle m_{11}, s\, a\rangle \\
\langle s\, a\rangle \\
\langle\ \rangle
\end{array}$$

where the transition maps are defined as indicated by vertical alignment; the 0-blocks and 1-blocks are separated by commas and semicolons, respectively; and the 0-blocks $\langle s\, m_{01}, m_{11}\, p_1\rangle$, $\langle m_{11}\, p_1\rangle$, and

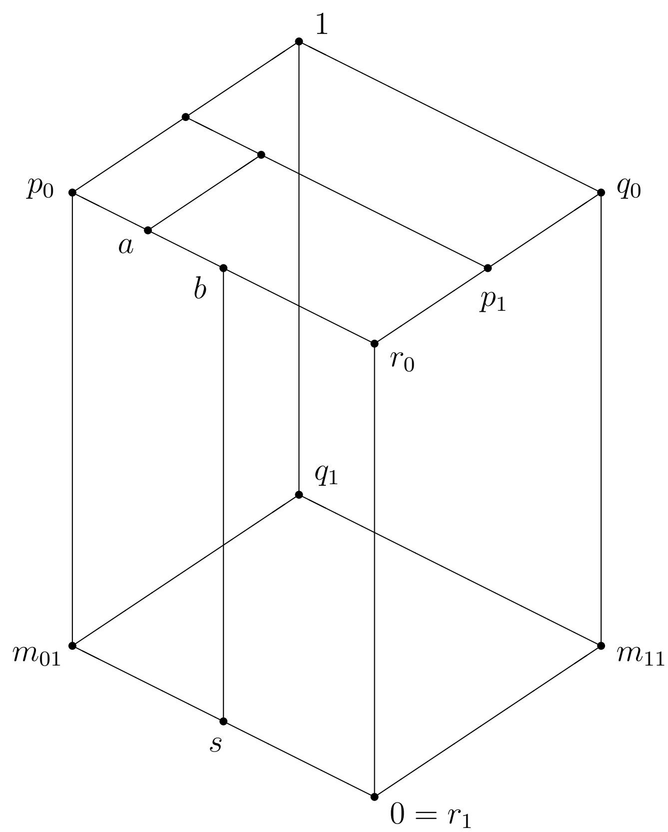

FIGURE 5. An embeddable quasilattice with two meets specified

$\langle m_{11} \rangle$ of the second, third, and fifth blocked target sequence, respectively, are at gate G_0, while all the other blocks are at gate G_1. This uses that

$$\begin{aligned}(s, m_{01}) &= F(q_0)\\ (m_{11}, p_1) &= F(p_0)\\ (r_0, p_1) &= F(q_1)\\ (m_{11}, s, a) &= F(p_1), \text{ and}\\ (s, a) &= F(q_0).\end{aligned}$$

Finally note that in the fourth blocked target sequence, we are not prohibited from inserting r_0 since $p_0 \notin (m_{11}, p_1)$.

Example 15. Figure 6 shows the nonembeddable quasilattice L_{14} with the two meets $p_0 \wedge q_0 = r_0$ and $p_1 \wedge q_1 = r_1$ specified. (Compared to the lattice L_{20} in Figure 3 of Lempp and Lerman [9], the nonembeddability

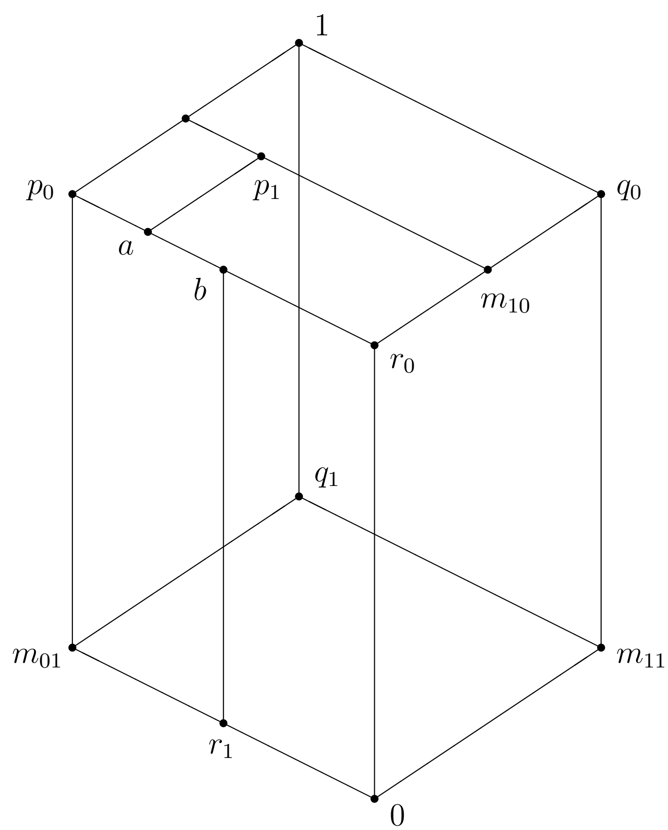

FIGURE 6. Lattice L_{14}

proof for L_{14} is much simpler since it requires only two meets instead of four.)

We will illustrate that there is no good blocked target array, and thus that L_{14} cannot be embedded into the c.e. degrees as follows.

The only possible blocked target sequences we can start with are the sequences $\langle m_{11}, m_{01}; m_{10}, a\rangle$ and $\langle m_{11}; m_{10}; m_{01}; a\rangle$ where the 0-blocks and 1-blocks are separated by commas and semicolons, respectively. (The first sequence could be split into more 1-blocks, at no advantage.) The proof that these two starting blocked target sequences will not yield a good blocked target array and thus will not lead to an embedding strategy are similar, so we only indicate the proof for the first sequence.

Starting from the blocked target sequence $\overrightarrow{a_0} = \langle m_{11}, m_{01}; m_{10}, a\rangle$, the second blocked target sequence $\overrightarrow{a_1}$ must contain $\langle m_{01}; m_{10}, a\rangle$ as a subsequence, with m_{01} at gate G_0 and the other elements at gate G_1. In order to have m_{01} pass by gate G_0, we must make it part of a prime filter $F \subseteq F(q_0)$, and the only choice here is $F = F(q_0)$; so the second blocked target sequence $\overrightarrow{a_1}$ must contain $\langle r_1\, m_{01}; m_{10}, a\rangle$ as a subsequence. This is still not a covering sequence since (r_1, m_{01}, m_{10}) is not a prime filter, so we need add either r_0 or m_{11} between m_{01} and m_{10}. By prohibition and since $p_0, q_0 \in (m_{01}, m_{11})$, we cannot add r_0, so the second blocked target sequence must be $\overrightarrow{a_1} = \langle r_1\, m_{01}, m_{11}; m_{10}, a\rangle$ (or some supersequence, or possibly with more 1-blocks, at no advantage) where $\langle r_1\, m_{01}, m_{11}\rangle$ is at gate G_0 and $\langle m_{10}, a\rangle$ is at G_1. Now the third blocked target sequence $\overrightarrow{a_2}$ must contain $\langle m_{11}; m_{10}, a\rangle$ as a subsequence. This is not yet a covering sequence since (m_{11}, m_{10}, a) is not a prime filter but needs r_0, r_1, or m_{01} before a. The former two are prohibited, so we need to use m_{01}, yielding $\overrightarrow{a_2} = \langle m_{11}; m_{10}; m_{01}; a\rangle$ or $\overrightarrow{a_2} = \langle m_{11}, m_{01}; m_{10}, a\rangle$ (or some supersequence thereof), i.e., we have returned to one of the two starting sequences. In this vein, one can show formally that L_{14} is not embeddable since any good covering array starting with one of the starting sequences keeps repeating a starting sequence (or some supersequence thereof) over and over.

Case 2.2: $r_0 \not\leq b$: This case is similar to Case 2.1 except that whenever a 0-block contains an element $c \leq r_0$ from the starting sequence, that 0-block (which can extend to the end of the current 1-block) can pass by gate G_0 without problems. A simple modification of the strategy in Case 2.1 thus also shows this case to be decidable.

3.7. **Two Meets** $p_0 \wedge q_0 = r_0$ **and** $p_1 \wedge q_1 = r_1$.

The full argument for two meets is very similar to the case of just two gates outlined in the previous section. We sketch the argument here.

First of all, note that we can think of consecutive gates for the same meet as just one single gate for the purpose of this construction since one gate, or several gates for the same meet, will impose the same restrictions on the allowable blocked target sequences. So assume from now on that any sequence of gates alternates between the two meets specified, say, even-indexed gates work for the meet $p_0 \wedge q_0 = r_0$ and odd-indexed gates work for the meet $p_1 \wedge q_1 = r_1$.

Next, note that due to clause (1) of Case 2.1 in section 3.6, we cannot add an element $c \leq r_l$ at a gate for the meet $p_l \wedge q_l = r_l$. Also, when we add an element $c \leq r_l$ at a gate for the meet $p_{1-l} \wedge q_{1-l} = r_{1-l}$, we can only add it where it is not prohibited.

Now observe that adding duplicate elements c neither $\leq r_0$ nor $\leq r_1$ is unnecessary since there is no additional coverage, and such elements cannot serve as correction markers. Thus we can bound the number of such elements added at any gate. Furthermore, we can argue, as in section 3.6, that any element $c \leq r_l$ added at a gate can be replaced by the full filter $F(r_{1-l})$. This yields as effective bound on the length of the blocked target sequences, thus giving us an effective bound on the length of blocked target arrays before blocked target sequences are repeated. Therefore, the embeddings problem for finite semi-distributive quasilattices with two meets specified is decidable as claimed in clause (3) of our Main Statement.

3.8. **Three Meets $p_i \wedge q_i = r_i$ for $i \leq 2$.** Remarks (1) and (4) in section 3.6 and their extension to the full two-gate case tell us that for a finite quasilattice with only two meets specified, there is no need for so-called "correction markers" to correct meet functionals Δ. The reason for this is that correction markers are balls targeted for elements $c \leq r_k$ at a gate $G_{k'}$ (for some $k' > k$) specifically to correct meet the functional $\Delta_k(R_k)$ such that these elements are not needed for coverage; however, we saw that for two meets, in the only possible case, namely, targeting below r_0 at gate G_1, we will only target so when needed for coverage, and then only with the full filter $F(r_1)$.

In the three-meet case, this may no longer be so. The partial evidence we have for this is only indirect in that we cannot point to a specific finite join-semidistributive lattice which is embeddable but only using correction markers. However, we have worked out a method of "trace trees" (coding the elements needed to cover other elements), which suggests that correction markers may be necessary for some finite join-semidistributive quasilattices with three meets specified. On the other hand, if correction markers are not needed (i.e., if all numbers entering sets are for coverage and not only to correct meet functionals $\Delta^{p,q}$), then the decision procedure for two meets outlined in the previous section will bound the number of good blocked target arrays to be considered in Lerman's Embeddability Criterion. This may then lead to the decidability of the lattice embeddings problem, provided one can also bound the number of gates to be considered.

References

[1] Ambos-Spies, K.; Jockusch, C. G. Jr.; Shore, Richard A.; and Soare, Robert I. *An algebraic decomposition of the recursively enumerable degrees and the coincidence of several degree classes with the promptly simple degrees*, Trans. Amer. Math. Soc. **281** (1984), 109–128.

[2] Ambos-Spies, Klaus and Lerman, Manuel, *Lattice embeddings into the recursively enumerable degrees*, J. Symbolic Logic **51** (1986), 257–272.

[3] Ambos-Spies, Klaus and Lerman, Manuel, *Lattice embeddings into the recursively enumerable degrees II*, J. Symbolic Logic **54** (1989), 735–760.

[4] Downey, Rodney G., *Lattice nonembeddings and initial segments of the recursively enumerable degrees*, Ann. Pure Appl. Logic **49** (1990), 97–119.

[5] Gorbunov, Viktor A., *Algebraic theory of quasivarieties*, Siberian School of Algebra and Logic, Consultants Bureau, New York, 1998.

[6] Lachlan, Alistair H., *Lower bounds for pairs of recursively enumerable degrees*, Proc. London Math. Soc. (3) **16** (1966), 537–569.

[7] Lachlan, Alistair H., *Embedding nondistributive lattices in the recursively enumerable degrees*, Conference in Mathematical Logic—London '70 (Proc. Conf., Bedford Coll., London, 1970), pp. 149–177, Lecture Notes in Math., Vol. 255, Springer-Verlag, Berlin, 1972.

[8] Lachlan, Alistair H. and Soare, Robert I., *Not every finite lattice is embeddable in the recursively enumerable degrees*, Adv. in Math. **37** (1980), 74–82.

[9] Lempp, Steffen and Lerman, Manuel, *A finite lattice without critical triple that cannot be embedded into the enumerable Turing degrees*, Ann. Pure Appl. Logic **87** (1997), 167–185.

[10] Lerman, Manuel, *Admissible ordinals and priority arguments*, in: "Cambridge Summer School in Mathematical Logic (Cambridge, 1971)", Springer-Verlag, Berlin, 1973, 311–344.

[11] Lerman, Manuel, *A necessary and sufficient condition for embedding ranked finite partial lattices into the computably enumerable degrees*, Ann. Pure Appl. Logic **94** (1998), 143–180.

[12] Lerman, Manuel, *A necessary and sufficient condition for embedding principally decomposable finite lattices into the computably enumerable degrees*, Ann. Pure Appl. Logic **101** (2000), 275–297.

[13] Lerman, Manuel, *Embeddings into the computably enumerable degrees*, Computability theory and its applications (Boulder, CO, 1999), pp. 191–205, Amer. Math. Soc., Providence, RI, 2000.

[14] Slaman, Theodore A. and Soare, Robert I., *Extension of embeddings in the computably enumerable degrees* Ann. of Math. (2) **154** (2001), 1–43.

[15] Soare, Robert I., *Recursively enumerable sets and degrees*, Perspectives in Mathematical Logic, Springer-Verlag, Berlin, 1987.

[16] Thomason, Steven K., *Sublattices of the recursively enumerable degrees*, Z. math. Logik Grundlagen Math. **17** (1971), 273–280.
[17] Yates, C. E. M., *A minimal pair of recursively enumerable degrees*, J. Symbolic Logic **31** (1966), 159–168.

Department of Mathematics, University of Wisconsin, Madison, WI 53706-1388, USA
E-mail address: lempp@math.wisc.edu

(Lerman, Solomon) Department of Mathematics, University of Connecticut, Storrs, CT 06269-3009, USA
E-mail address: mlerman@math.uconn.edu

E-mail address: david.solomon@uconn.edu