

Reduction of Order Structures

Łukasz Mikulski¹ and Andrey Mokhov² and Marcin Piątkowski¹

¹ Nicolaus Copernicus University, Poland

² Newcastle University, United Kingdom

Abstract—Relational order structures are used to describe and investigate properties of concurrent systems. To reduce the complexity of order structures, one typically considers only their essential components, which, in the case of partial orders, leads to the notion of Hasse diagrams. We lift this notion to the level of generalised mutex order structures, which are used to model not only causal dependencies but also weak causality and mutual exclusion. We provide a new and more concise axiomatic definition of these structures, investigate their important properties, and present efficient algorithms for computing their *reduction* and *closure*. The algorithms are implemented in a publicly available software tool with graphical user interface.

I. INTRODUCTION

The interleaving semantics of concurrent systems is often represented by a set of equivalent runs in the system. Analysis of these sets often requires processing of a large number of sequences that have a lot in common. By adopting the non-interleaving semantics, it is possible to extract concurrency-related similarities from sets of equivalent runs, thereby simplifying the representation and reducing the analysis time. In the ideal scenario, this results in a single order structure, common for all equivalent runs. Such a structure, focusing on causal dependencies, has a form of an acyclic binary relation. This relation can be transformed into a *partially ordered set* using the *transitive closure* algorithm. The real benefit (from the point of view of the size of the representation) is achieved by the *transitive reduction*, instead of closure, of the relation on events to the smallest possible form that keeps all essential dependencies (not necessary in a direct form). Such a reduced partially ordered set is called the Hasse diagram.

In some distributed systems, the equivalence of possible sequences of single events is often inadequate and one needs to operate on *sequences of sets of events that occur simultaneously*. One example is clocked hardware circuits, where sets of events may occur simultaneously within a single subsystem, yet still be partially ordered between different subsystems whose clock signals are independent. A model of observable runs for describing such systems was studied in [4], where the paradigms of concurrency theory based on step sequences as observations were proposed. In the same paper, one of less expressive paradigms was equipped with a model of combined traces together with their structures called combined partially ordered sets (or stratified order structures using the nomenclature of [2]). The most general paradigm was given by generalised traces, represented by *generalised mutex-order structures* [2]. Since generalised mutex-order structures are in this case counterparts of partially ordered sets, it is natural to ask the question about their closure and reduction operations.

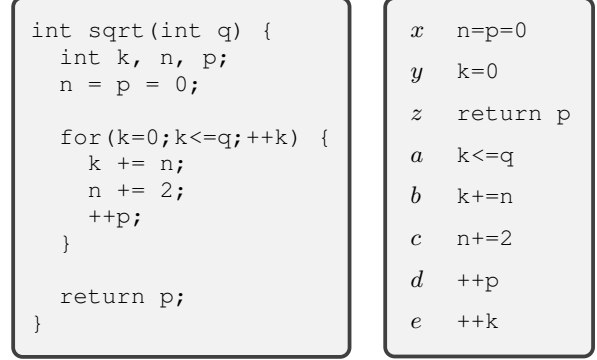


Figure 1. A C++ function computing $\lceil \sqrt{q} \rceil$ and a labelling of its statements.

As an another example, consider the C++ function `sqrt` shown in Figure 1 along with a labelling of each primitive statement. In this case we are interested in causal dependencies between statements, which are based on variable modifications and associated data dependencies. We assume that each such statement consists of two atomic parts: computing the new value (together with all required read operations) and storing the new value (a single write operation). Moreover, we allow a set of statements to be executed in parallel. An example run of this function for the input $q = 4$ can be abstracted by the following sequence of statements (using the statement labelling in Figure 1):

$x y a b c d e a b c d e a z.$

There are multiple valid sequences in which the statements can be executed without violation of causal dependencies; moreover, some statements can be executed in parallel. For example, any two consecutive occurrences of d and e can be executed either in parallel or in any order. On the other hand, while any two consecutive occurrences of a , b and c can be executed in parallel (i.e. do all reads, and then do all writes), we cannot change their order in any way. Note that data dependencies are not the only reason for restricting the order between statements. As discussed in [5], the order of statements with arbitrary computational effects can be observed if the underlying effect structure is non-commutative.

In this paper we present a framework, which allows us to model the concurrent behaviour of the aforementioned (and similar) case in a clear and precise way. At the end of the paper this motivating example is investigated in more details.

As already pointed out in [6], in the reduction of stratified order structures we need to face some problems, which were not present in the case of partially ordered sets. In particular, it is not trivial to achieve the uniqueness of the reduction. One possible approach is to use the concept of indivisible sets of events and folded structures, as proposed in [6]. In this paper we introduce algorithms for computing the reduction and closure of a consistent order structure in the time complexity quadratically dependent on the size of the structure. The algorithms have been implemented in a software tool that is publicly available at [7].

The main contributions of this paper are as follows.

- We formally define order structures and present a new concise axiomatic definition of generalised mutex-order structures in Section II.
- A decomposition of the mutual exclusion relation into *aligned* and *crossing* mutexes is discussed in Section III. The decomposition is an essential component of the formal definitions of the closure, folding, integration and reduction operations on order structures, that are introduced in Sections IV-VII.
- We present algorithms for the reduction and closure of order structures in Section VIII.

II. ORDER STRUCTURES

A *reflexive order structure* (ros) $S = (X, \Rightarrow, \sqsubset)$ is a relational structure satisfying the following conditions:

- \Rightarrow is irreflexive and symmetric.
- \sqsubset is reflexive
- \Rightarrow and \sqsubset are *separable*, that is:

$$(\sqsubset^* \cap (\sqsubset^{-1})^* \cap \Rightarrow) = \emptyset.$$

Relations \Rightarrow and \sqsubset stand for *mutual exclusion* (mutex) and *weak causality* between pairs of events from X . Intuitively, \Rightarrow and \sqsubset have meanings similar to that of \neq and \leq in arithmetic, however, the set of events X is not necessarily totally ordered. The separability condition prohibits the situation when two events are both simultaneous (because they belong to a cycle in \sqsubset) and mutually exclusive (because of \Rightarrow) at the same time.

We also define the *containment* relationship of relational structures over the same signature. Let $S_1 = (X, \Rightarrow_1, \sqsubset_1)$ and $S_2 = (X, \Rightarrow_2, \sqsubset_2)$ be two relational structures. We say that S_1 is *contained* in S_2 (or S_2 *extends* S_1) if

- $\Rightarrow_1 \subseteq \Rightarrow_2$,
- $\sqsubset_1 \subseteq \sqsubset_2$,

and denote it by $S_1 \triangleleft S_2$.

Example 1. Let $S = (X, \Rightarrow, \sqsubset)$ be as in Figure 2 (solid edges denote relation \Rightarrow , dashed arcs denote relation \sqsubset , solid arcs stand for $\Rightarrow \cap \sqsubset$). For readability reasons, in all figures depicting relational structures we do not draw loops associated with the reflexivity of relations \sqsubset .

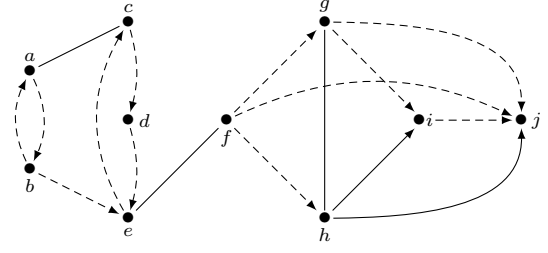


Figure 2. An example of reflexive order structure S .

A. Generalised mutex order structures

Let $G = (X, \Rightarrow, \sqsubset)$ be a relational structure satisfying the following axioms for all $a, b, c, d \in X$:

- G1: $a \Rightarrow b \implies b \Rightarrow a$.
- G2: $a \sqsubset a \wedge a \neq a$.
- G3: $a \sqsubset b \sqsubset c \implies a \sqsubset c$.
- G4: $a \sqsubset b \sqsubset a \wedge a \Rightarrow c \implies b \Rightarrow c$.
- G5: $a \sqsubset b \sqsubset d \wedge a \sqsubset c \sqsubset d \wedge b \Rightarrow c \implies a \Rightarrow d$.

Then G is called (*reflexive*) *generalised mutex order structure* (*gmos*)¹, while the set of all generalised mutex order structures is denoted by *GMOS*. Note that every *gmos* is a reflexive order structure, in particular the separability condition follows from axioms **G2** and **G4**.

Note that above definitions of *ros* and *gmos* differ from previous definitions in that the weak causality relation is reflexive (as opposed to being irreflexive as, e.g., in [2]). In our experience, this change significantly simplifies the theory, in particular it makes some previously required axioms redundant as will be demonstrated below. With the help of the ALG tool [1] we have checked that reflexive and irreflexive theories have the same number of models, which confirms that there is a one-to-one correspondence between reflexive and irreflexive order structures. We have also checked the minimality (with respect to the number of axioms) of the previous axiomatic definition of *gmos* as well as of the above definition, thereby proving that the reduction in the number of axioms is a consequence of the reflexivity choice.

Consider the following theorem:

$$\text{T1: } a \sqsubset b \sqsubset c \wedge (a \Rightarrow b \vee b \Rightarrow c) \implies a \Rightarrow c.$$

This theorem was an axiom in [2], but we can now prove it from **G5**, reflexivity and transitivity. Indeed, by instantiating **G5** with $a = b$ we can derive the following:

$$\begin{aligned} a \sqsubset a \sqsubset d \wedge a \sqsubset c \sqsubset d \wedge a \Rightarrow c &= \\ a \sqsubset d \wedge a \sqsubset c \sqsubset d \wedge a \Rightarrow c &= \\ a \sqsubset c \sqsubset d \wedge a \Rightarrow c &\implies \\ a \Rightarrow d. & \end{aligned}$$

¹A structure satisfying axioms G1, G3-G5 and G2': $a \not\sqsubset a \wedge a \neq a$ in place of G2 is called in [2] generalised mutex order structure. In [3], referring to [4], another name for such structures was proposed – invariant order structures. Since we do not discuss in this paper invariants of concurrent systems, we use the former name.

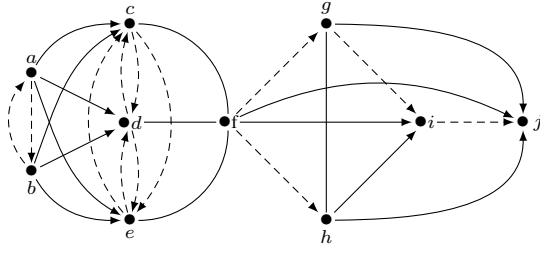


Figure 3. An example of generalised mutex order structure.

This proves one half of **T1** corresponding to the first disjunction term. To prove the other half, **G5** needs to be instantiated with $b = d$.

According to the remark below, one can also eliminate **G1**.

Remark 2. By changing axiom **G4** into

$$a \sqsubset b \sqsubset a \wedge a \rightleftharpoons c \implies c \rightleftharpoons b$$

(mind the order of events in the last term), one can make axiom **G1** redundant. Indeed, $b \rightleftharpoons a$ follows directly from $a \rightleftharpoons b$, $a \sqsubset a$ (axiom **G2**) and modified **G4**.

Example 3. Recall the reflexive order structure S from Example 1 and let $G = (X, \rightleftharpoons, \sqsubset)$ be as in Figure 3. Then G satisfies axioms **G1** to **G5** (hence it is a generalised mutex order structure) and $S \triangleleft G$ (i.e., S is contained in G).

III. DECOMPOSING THE MUTEX RELATION

We can decompose the mutex relation \rightleftharpoons into relations \prec and \parallel as follows:

- $\prec \stackrel{\text{df}}{=} \sqsubset \cap \rightleftharpoons$.
- $\parallel \stackrel{\text{df}}{=} \rightleftharpoons \setminus (\prec \cup \prec^{-1})$.

Proposition 4. For every gmos, (X, \prec) is a strict partial order.

Proof. A strict partial order relation must be transitive (that is, $a \prec b \prec c \implies a \prec c$) and irreflexive ($a \not\prec a$). The irreflexivity directly follows from axiom **G2**. The transitivity can be proved by using axioms **G2**, **G3** and **G5**. If $a \prec b \prec c$ then, by the above definition, both $a \sqsubset b \sqsubset c$ and $a \rightleftharpoons b \rightleftharpoons c$ hold. **G2** states that $a \sqsubset a$, and from **G3** we immediately have $a \sqsubset c$. Now, combining $a \sqsubset a \sqsubset c$, $a \sqsubset b \sqsubset c$, and $a \rightleftharpoons b$ yields $a \rightleftharpoons c$ via **G5**. Hence, $a \prec c$ holds as required. \square

We further call \prec and \parallel *aligned* and *crossing mutexes*, respectively. Intuitively, mutexes in \prec are aligned with the weak causality relation \sqsubset , while mutexes in \parallel cross it. This classification of mutexes is useful for derivation of the reduction of generalised mutex order structure. In particular, according to axiom **G5**, crossing mutexes can induce aligned ones, but not vice versa. Therefore, all crossing mutexes must belong to the reduction of a structure, while the aligned mutexes induced by them can be dropped without ambiguity.

Remark 5. Another important distinction between aligned and crossing mutexes of generalised mutex order structures is that the latter lead to arbitration: for $a \parallel b$ a non-monotonic

decision needs to be made on whether a occurs before b (that is, $a \prec b$) or vice versa ($b \prec a$).

For any crossing mutex $m = (x, y)$ we can define the set $\alpha(m)$ of all aligned mutexes **G5**-induced by m as follows:

$$\alpha(m) = (\bullet x \cap \bullet y) \times (x \bullet \cap y \bullet),$$

where $\bullet x = \{z \mid z \sqsubset x\}$ and $x \bullet = \{z \mid x \sqsubset z\}$ are the *preset* and the *postset* of an event x with respect to \sqsubset . For a given set of crossing mutexes $C \subseteq (X \times X) \setminus Id$, the corresponding set of **G5**-induced aligned mutexes will be denoted as $\alpha(C) \subseteq X \times X$ with $\alpha(C) = \bigcup_{m \in C} \alpha(m)$.

For a single crossing mutex m we can compute the set $\alpha(m)$ according to definition in time $O(|\sqsubset|)$. Hence, $\alpha(\parallel)$ can be computed in time $O(|\sqsubset| \cdot |\parallel|)$.

On the other hand, for every aligned mutex $m = (a, b)$ we also consider the corresponding set of **G5**-induced aligned mutexes and denote it by $\beta(m)$. The set $\beta(m)$ can be represented as follows:

$$\beta(m) = \bullet a \times b \bullet \setminus \{(a, b)\},$$

We can extend it to any set of aligned mutexes $A \subseteq X \times X$ by $\beta(A) = \bigcup_{m \in A} \beta(m)$. For a single aligned mutex m we can compute the set $\beta(m)$ according to definition in time $O(|\prec|)$. Hence, $\beta(\prec)$ can be computed in time $O(|\prec|^2)$.

Example 6. Recall the reflexive order structure S from Example 1 and the generalised mutex order structure G from Example 3.

In the case of the former structure we can observe three crossing mutexes (i.e. $C = \{(a, c), (e, f), (g, h)\}$) and two aligned mutexes (i.e. $A = \{(h, i), (h, j)\}$).

In the case of the structure G we have four crossing mutexes and eleven aligned mutexes. Moreover, $\alpha(C) = \{f\} \times \{i, j\}$ with two implied aligned mutexes, while $\beta(A) = \{f, h\} \times \{j\} \cup (f, i)$ imply three aligned mutexes (and include $\alpha(C)$). Note that the aligned mutex (f, j) is implied by the crossing mutex (g, h) as well as by any of four other aligned mutex (namely (f, i) , (h, i) , (g, j) or (h, j)).

We now formulate a proposition describing relationships between the sets of mutexes induced by other mutexes.

Proposition 7. Let $G = (X, \rightleftharpoons, \sqsubset)$ be a generalised mutex order structure and let $m_1 = (a, b), m_2 = (c, d) \in X \times X$.

- If $m_1 \in \parallel$ and $m_2 \in \alpha(m_1)$ then $m_2 \in \prec$ and $\beta(m_2) \subseteq \alpha(m_1)$.
- If $m_1 \in \prec$ and $m_2 \in \beta(m_1)$ then $m_2 \in \prec$ and $\beta(m_2) \subseteq (\beta(m_1) \cup \{m_1\})$.

Proof. Let us assume $(a, b) \in \parallel$. By the definition of α , we have $c \in \bullet a \cap \bullet b$ and $d \in a \bullet \cap b \bullet$. Hence, $c \sqsubset a \sqsubset d$ and $c \sqsubset b \sqsubset d$. By axiom **G5**, $c \rightleftharpoons d$, while by axiom **G3**, $c \sqsubset d$, therefore $(c, d) \in \prec$, as required.

Let $(e, f) \in \beta((c, d))$. Then, by definition of β , $e \in \bullet c$ and $f \in d \bullet$. Hence, $e \sqsubset c \sqsubset a \sqsubset d \sqsubset f$ and $e \sqsubset c \sqsubset b \sqsubset d \sqsubset f$. Therefore, by axiom **G3**, $e \sqsubset a \sqsubset f$ and $e \sqsubset b \sqsubset f$, and so $(e, f) \in \alpha((a, b))$.

Assume now that $(a, b) \in \prec$. By the definition of β , $c \in \bullet a$ and $d \in \bullet b$. Hence, $c \sqsubset a \sqsubset b \sqsubset d$, and so $c \sqsubset a \sqsubset d$ and $c \sqsubset b \sqsubset d$. By axiom **G5**, $c \equiv d$, while by axiom **G3**, $c \sqsubset d$, therefore $(c, d) \in \prec$, as required.

Let $(e, f) \in \beta((c, d))$. Then, by definition of β , $e \in \bullet c$ and $f \in \bullet d$. Hence, $e \sqsubset c \sqsubset a$ and $b \sqsubset d \sqsubset f$. Therefore, by axiom **G3**, $e \sqsubset a$ and $b \sqsubset f$, and so $(e, f) \in (\beta((a, b)) \cup \{(a, b)\})$ (it does not matter whether (a, b) is equal to (e, f) or not). \square

Remark 8. One can notice that in principle it is possible to completely eliminate aligned mutexes as follows. Let $a \prec b$ be an aligned mutex which is not induced by any crossing mutex. Then one can create two auxiliary elements $x \equiv y$ and embed them in the order structure so that $a \sqsubset x \sqsubset b$ and $a \sqsubset y \sqsubset b$. Now, $a \prec b$ is **G5**-induced from the crossing mutex $x \equiv y$. This procedure can be used to eliminate all aligned mutexes from a given order structure, which shows that crossing mutexes are more fundamental than aligned ones. We believe that this is an important observation and that it deserves further study, however such a study is outside of the scope of this paper.

IV. CLOSURE

A reflexive order structure $S = (X, \equiv, \sqsubset)$ can be *closed* to produce a uniquely defined generalised mutex order structure $S^c = (X, \equiv^c, \sqsubset^c)$ so that the following conditions are met:

- $S \triangleleft S^c$.
- $S_1 \triangleleft S_2 \in \text{GMOS} \implies S_1^c \triangleleft S_2^c$ (S^c is the smallest closed order structure extending S).

The intent of the closure operation is similar to that of the *transitive closure* of a relation R into a transitive relation R^+ : we add a minimal number of new elements into S so that the result S^c conforms to the axioms of generalised mutex order structures presented in Section II. Note that we drop the word ‘transitive’, because the concept of transitivity is not defined for the mutex relation \equiv .

The uniqueness of the above (in the case of irreflexive relation \sqsubset) is proven in [3], while the precise definition of the closure (also in the irreflexive case) is given in [2]. Adapted to the notation used in this paper, the definition can be expressed as:

- $a \equiv^c b \iff a \sqsubset^* b \iff (\equiv \cup \nabla^{sym}) \circ \sqsubset^* b$.
- $a \sqsubset^c b \iff a \sqsubset^* b$,

where $a \nabla b \iff \exists c, d, a \sqsubset^* c \sqsubset^* b \wedge a \sqsubset^* d \sqsubset^* b \wedge c \equiv d$, and $a \sqsubset^* b \iff a \sqsubset^* b \sqsubset^* a$.

We define an equivalence relation \equiv_{gmos} on order structures, relating those that have the same closure. Formally, we say that $S \equiv_{\text{gmos}} T \iff S^c = T^c$ and denote the equivalence class of a structure S by $[S]_{\text{gmos}}$.

Example 9. Recall S from Figure 2 and G from Figure 3. Note that despite $S \triangleleft G$, the structure G is not a closure of the structure S . The reason is the ineligible mutex (g, j) . The proper closure S^c is depicted in Figure 4.

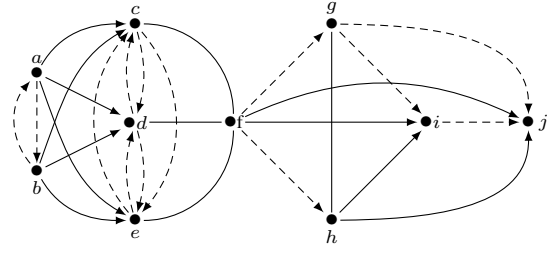


Figure 4. The closure of the structure S from Example 1.

V. FOLDING

A reflexive order structure $S = (X, \equiv, \sqsubset)$ can be *folded* into a more compact one by contracting the maximal equivalence classes induced by strongly connected components of \sqsubset (we denote this maximal equivalence relation by \equiv_f). The folded structure will be denoted as $S^f = (X^f, \equiv^f, \sqsubset^f)$ where X^f denotes the set of equivalence classes, while relations \equiv^f and \sqsubset^f are lifted versions of \equiv and \sqsubset , respectively. More formally, for all $x, y \in X^f$ the following holds:

- $x \equiv^f y \iff \exists a \in x, b \in y, a \equiv b$.
- $x \sqsubset^f y \iff \exists a \in x, b \in y, a \sqsubset b$.

The above can be reformulated in the case of generalised mutex order structures as follows:

- $a \equiv_f b \iff a \sqsubset b \sqsubset a$.
- $X^f = X / \equiv_f$.
- $[a] \equiv^f [b] \iff a \equiv b$.
- $[a] \sqsubset^f [b] \iff a \sqsubset b$.

Example 10. Recall the structure S from Example 1. In this case we have two non-trivial (of size greater than one) equivalence classes of the relation \equiv_f - namely $[a]_f = \{a, b\}$ and $[c]_f = \{c, d, e\}$. The folding S^f of the structure S is depicted in Figure 5.

Proposition 11. Let $G = (X, \equiv, \sqsubset)$ be a generalised mutex order structure. Then (X^f, \equiv^f) and (X^f, \prec^f) are a weak and a strict partial orders, respectively.

Proof. Recall that a weak partial order is a reflexive, transitive and antisymmetric relation, whereas a strict partial order is a irreflexive and transitive relation.

By the definition of folding, axioms **G2** and **G3**, relation \sqsubset^f is both reflexive and transitive. To prove its antisymmetry (for all $x, y: x \sqsubset y \wedge y \sqsubset x \implies x = y$) let us suppose that $[x]_f \sqsubset^f [y]_f$ and $[y]_f \sqsubset^f [x]_f$. Hence, $x \sqsubset y \sqsubset x$ and so $x \equiv_f y$, which gives $[x]_f = [y]_f$.

Let $x \prec y \prec z$. By **G3**, $x \sqsubset z$, while by **G5**, with $a = x$, $b = y$, $c = d = z$, $x \equiv z$. Hence by the definition of folding \prec^f is transitive.

Suppose that $[x]_f \prec^f [x]_f$. Then there exist $y, z \in [x]_f$ such that $y \equiv z$ and $x \sqsubset y \sqsubset x$ and $x \sqsubset z \sqsubset x$. Using **G4** twice we get $x \equiv x$, which is in contradiction with **G2**. Hence \prec^f is irreflexive. \square

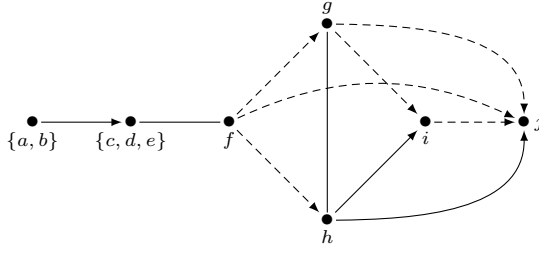


Figure 5. The folding of the structure S from Example 1.

An important property of folding is that it commutes with closure. We precede the corresponding proposition by three technical lemmas.

Lemma 12. *Let $S = (X, \Rightarrow, \sqsubset)$ be a reflexive order structure. Then $\sqsubset^* = \sqsubset^c = (\sqsubset^c)^*$ and the following are equivalent:*

- $a \equiv_f b$
- $a \sqsubset^* b \sqsubset^* a$
- $a \sqsubset^c b \sqsubset^c a$
- $a \sqsubset^{\circledast} b \sqsubset^{\circledast} a$

Proof. The equalities $\sqsubset^* = \sqsubset^c = (\sqsubset^c)^*$ follows directly from the definition of closure and idempotence of the star operation (i.e. $(\sqsubset^*)^* = \sqsubset^*$).

If $a \equiv_f b$ then, by the definition of folding, a and b are in the same strongly connected component of \sqsubset , hence $a \sqsubset^* b$ and $b \sqsubset^* a$.

Moreover $a \sqsubset^* b \sqsubset^* a$, is by the definition of closure, equivalent to $a \sqsubset^c b \sqsubset^c a$ and by the definition of \circledast with $a \sqsubset^{\circledast} b$ (and $b \sqsubset^{\circledast} a$).

As a result we argued the equivalence of $a \sqsubset^* b \sqsubset^* a$ with all other formulations. \square

Lemma 13. *Let $S = (X, \Rightarrow, \sqsubset)$ be a reflexive order structure. Then $a \sqsubset^* b$ is equivalent to $[a]_f(\sqsubset^f)^*[b]_f$.*

Proof. Let $a \sqsubset^* b$. Then there exist a sequence $\{c_i\}_{i=1}^n$ of elements from X such that $a \sqsubset c_1 \sqsubset c_2 \sqsubset \dots \sqsubset c_n \sqsubset b$. Hence, $[a]_f \sqsubset^f [c_1]_f \sqsubset^f [c_2]_f \sqsubset^f \dots \sqsubset^f [c_n]_f \sqsubset^f [b]_f$, and so $[a]_f \sqsubset^{f*} [b]_f$.

On the other hand, let $[a]_f \sqsubset^{f*} [b]_f$. Then there exist a sequence $\{[c_i]_f\}_{i=1}^n$ of elements from X^f such that $[a]_f \sqsubset^f [c_1]_f \sqsubset^f [c_2]_f \sqsubset^f \dots \sqsubset^f [c_n]_f \sqsubset^f [b]_f$. Hence, by Lemma 12, there exist two sequences $\{c_i^{in}\}_{i=1}^n, \{c_i^{out}\}_{i=1}^n$ and a^{out} and b^{in} such that

- $c_i^{in}, c_i^{out} \in [c_i]_f, a^{out} \in [a]_f, b^{in} \in [b]_f$
- $c_i^{out} \sqsubset c_{i+1}^{in}, a^{out} \sqsubset c_1^{in}, c_n^{out} \sqsubset b^{in}$

Hence, $a \sqsubset^* a^{out} \sqsubset c_1^{in} \sqsubset c_1^{out} \sqsubset c_2^{in} \sqsubset c_2^{out} \sqsubset \dots \sqsubset c_n^{in} \sqsubset c_n^{out} \sqsubset b^{in} \sqsubset b$, which concludes the proof. \square

Lemma 14. *Let $S = (X, \Rightarrow, \sqsubset)$ be a reflexive order structure and $a, b \in X$. Then*

$$\begin{aligned} \exists_{c,d} a \sqsubset^* c \sqsubset^* b \wedge a \sqsubset^* d \sqsubset^* b \wedge [c]_f \Rightarrow^f [d]_f \\ \iff \\ \exists_{x,y} a \sqsubset^* x \sqsubset^* b \wedge a \sqsubset^* y \sqsubset^* b \wedge x \Rightarrow y. \end{aligned}$$

Proof. Let c, d be such that $a \sqsubset^* c \sqsubset^* b \wedge a \sqsubset^* d \sqsubset^* b \wedge [c]_f \Rightarrow^f [d]_f$. Then, by the definition of folding, there exists $c^{out} \in [c]_f$ and $d^{in} \in [d]_f$, such that $c \Rightarrow d$. By Lemma 12, we get $c \sqsubset^* c^{out} \sqsubset^* c$ and $d \sqsubset^* d^{in} \sqsubset^* d$, so $a \sqsubset^* c \sqsubset^* c^{out} \sqsubset^* c \sqsubset^* b \wedge a \sqsubset^* d \sqsubset^* d^{in} \sqsubset^* d \sqsubset^* b$ and $a \sqsubset^* c^{out} \sqsubset^* b \wedge a \sqsubset^* d^{in} \sqsubset^* b$. Hence we can take $x = c^{out}$ and $y = d^{in}$.

On the other hand, let x, y be such that $a \sqsubset^* x \sqsubset^* b \wedge a \sqsubset^* y \sqsubset^* b \wedge x \Rightarrow y$. By the definition of folding, $[x]_f \Rightarrow^f [y]_f$, hence we can take $a = x$ and $b = y$. \square

Proposition 15. *The folding and closure operations commute, that is $(S^c)^f = (S^f)^c$.*

Proof. Let $S = (X, \Rightarrow, \sqsubset)$ be an order structure. Let $S^{cf} = (X^{cf}, \Rightarrow^{cf}, \sqsubset^{cf})$ be the folded closure of S , while $S^{fc} = (X^{fc}, \Rightarrow^{fc}, \sqsubset^{fc})$ be the closure of its folding. Note that the elements of S^{cf} and S^{fc} are equivalence classes of elements from the set X and, by Lemma 12, the equivalence relations that induce those classes are identical (since $\sqsubset^* = (\sqsubset^c)^*$ it does not matter if we initially close a structure).

In order to show the equality of S^{fc} and S^{cf} we shall prove that:

- $[a]_f \sqsubset^{cf} [b]_f \iff [a]_f \sqsubset^{fc} [b]_f$
- $[a]_f \Rightarrow^{cf} [b]_f \iff [a]_f \Rightarrow^{fc} [b]_f$

We start from the first equivalence. By Lemma 12, $[a]_f \sqsubset^{fc} [b]_f$ is equivalent to $[a]_f(\sqsubset^f)^*[b]_f$, which, by Lemma 13, is equivalent to $a \sqsubset^* b$, which, by Lemma 12, is equivalent to $a \sqsubset^c b$. By the definition of folding for GMOS this is equivalent to $[a]_f(\sqsubset^c)^f[b]_f$, which concludes this part of proof.

A bit more complex is the situation with mutexes. Let $[a]_f \Rightarrow^{fc} [b]_f$. By the definition of closure it is equivalent to $[a]_f(\sqsubset^f)^{\circledast} \circ (\Rightarrow^f \cup \nabla_f^{sym}) \circ (\sqsubset^f)^{\circledast} [b]_f$ where

$$\begin{aligned} [[a]_f \nabla_f [b]_f] &\iff \\ \exists_{c,d} [a]_f(\sqsubset^f)^*[c]_f(\sqsubset^f)^*[b]_f \wedge \\ [a]_f(\sqsubset^f)^*[d]_f(\sqsubset^f)^*[b]_f \wedge [c]_f \Rightarrow^f [d]_f, \end{aligned}$$

which, by Lemma 13 and Lemma 12 can be reduced to an equivalent formulation: $[a]_f \Rightarrow^f [b]_f$ or $\exists_{c,d,a} a \sqsubset^* c \sqsubset^* b \wedge a \sqsubset^* d \sqsubset^* b \wedge [c]_f \Rightarrow^f [d]_f$ or $\exists_{c,d,b} c \sqsubset^* a \wedge b \sqsubset^* d \sqsubset^* a \wedge [c]_f \Rightarrow^f [d]_f$. By Lemma 14, this is equivalent to $[a]_f \Rightarrow^f [b]_f$ or $\exists_{c,d,a} a \sqsubset^* c \sqsubset^* b \wedge a \sqsubset^* d \sqsubset^* b \wedge c \Rightarrow d$ or $\exists_{c,d,b} c \sqsubset^* a \wedge b \sqsubset^* d \sqsubset^* a \wedge c \Rightarrow d$. By the definition of folding, this is equivalent to $\exists_{a' \in [a]_f, b' \in [b]_f} a' \Rightarrow b'$ or $\exists_{c,d,a} c \sqsubset^* b \wedge a \sqsubset^* d \sqsubset^* b \wedge c \Rightarrow d$ or $\exists_{c,d,b} c \sqsubset^* a \wedge b \sqsubset^* d \sqsubset^* a \wedge c \Rightarrow d$, which can be reformulated in an equivalent way to $a \sqsubset^{\circledast} \circ \Rightarrow \circ \sqsubset^{\circledast} b$ or $a \sqsubset^{\circledast} \circ \nabla_f^{sym} \circ \sqsubset^{\circledast} b$. By the definition of closure, this is equivalent to $a \Rightarrow^c b$, which by the definition

of folding for GMOS is equivalent to $[a]_f \equiv^{cf} [b]_f$, and this concludes the proof. \square

Folding can be computed using classical algorithms for finding strongly connected components, nominating representatives and erasing one by one connections between vertices that are not representatives. The required computation time is $O(|\sqsubset| + |X|)$ (for strongly connected components, see [8]), plus $O(|\sqsubset| + |\equiv|)$ (in order to transfer both relations to the folding). In the case of generalised mutex order structures, this can be simplified to browsing all vertices and for every visited x dropping each y for which $y \sqsubset x \sqsubset y$.

Note that the existence of a relationship between two equivalence classes of events requires corresponding relationship between two events from those classes. Hence, $|\equiv^f| \leq |\equiv|$ and $|\sqsubset^f| \leq |\sqsubset|$.

Proposition 16. *Let $G = (X, \equiv, \sqsubset)$ be a generalised mutex order structure, $G^f = (X^f, \equiv^f, \sqsubset^f)$ be its folding, and $m_1 = (a, b), m_2 = (c, d) \in X^f \times X^f$.*

- *If $m_1 \in |\sqsubset^f|$ and $m_2 \in \alpha(m_1)$ then $m_2 \in \prec^f$ and $\beta(m_2) \subseteq \alpha(m_1)$.*
- *If $m_1 \in \prec^f$ and $m_2 \in \beta(m_1)$ then $m_2 \in \prec^f$ and $\beta(m_2) \subseteq \beta(m_1)$.*

Proof. The first part follows directly by Proposition 7 and the definition of folding for GMOS. For similar reasons it is enough to prove that $m_1 \notin \beta(m_2)$. Indeed, suppose that $m_1 \in \beta(m_2)$. Then there exist $x, y, p, q \in X$ such that $[x]_f = a$, $[y]_f = b$, $[p]_f = c$, and $[q]_f = d$. Moreover, $x \prec y$ and $p \sqsubset x$ and $y \sqsubset q$. On the other hand, we know that $m_2 \in \beta(m_1)$, hence $x \sqsubset p$ and $q \sqsubset y$. Summing up, $p \sqsubset x \sqsubset p$ and $q \sqsubset y \sqsubset q$, which means that $[p]_f = [x]_f$ and $[q]_f = [y]_f$. By definition of β , $[p]_f \neq [x]_f$ or $[q]_f \neq [y]_f$, which is a contradiction. Hence, our supposition does not hold and this completes the proof. \square

Lemma 17. *Let $S = (X, \equiv, \sqsubset)$ be a reflexive order structure, $S^f = (X^f, \equiv^f, \sqsubset^f)$ be its folding, $S^c = (X, \equiv^c, \sqsubset^c)$ its closure and $a, b \in X$ such that $a \equiv^c b$ but $[a]_f \not\equiv^f [b]_f$. Then there exists $c, d \in X$ such that $(c, d) \in |\sqsubset^c| \wedge ([a]_f, [b]_f) \in \alpha(([\![c]_f, [d]_f])$ or $(c, d) \in \prec^c \wedge ([a]_f, [b]_f) \in \beta(([\![c]_f, [d]_f])$.*

Proof. Let $a \equiv^c b$ but $[a]_f \not\equiv^f [b]_f$. Then, naturally, $[a]_f \not\equiv^{fc} [b]_f$ and so $[a]_f(\sqsubset^f)^{\otimes} \circ (\equiv^f \cup \nabla^f) \circ (\sqsubset^f)^{\otimes} [b]_f$. By Lemma 12 we can simplify this by reducing compositions with $(\sqsubset^f)^{\otimes}$ obtaining $[a]_f(\equiv^f \cup \nabla^f)[b]_f$ and, since $[a]_f \not\equiv^f [b]_f$ we can reduce it further to $[a]_f \nabla^f [b]_f$. Hence, by the definition of ∇ , we get that $\exists [c]_f, [d]_f [a]_f(\sqsubset^f)^* [c]_f(\sqsubset^f)^* [b]_f \wedge [a]_f(\sqsubset^f)^* [d]_f(\sqsubset^f)^* [b]_f \wedge [c]_f \equiv^f [d]_f$. By Lemma 14 we can choose c, d in such a way that $c \equiv d$. Moreover, by Lemma 12, $a \sqsubset^c c$ and $a \sqsubset^c d$ and $c \sqsubset^c b$ and $d \sqsubset^c b$. As a result $a \in \bullet c$, $a \in \bullet d$, $b \in c \bullet$ and $b \in d \bullet$ in G^c . And so $[a]_f \in \bullet [c]_f$, $[a]_f \in \bullet [d]_f$, $[b]_f \in [c]_f \bullet$ and $[b]_f \in [d]_f \bullet$ in G^{fc} .

Since $c \equiv d$ and $[a]_f \not\equiv^f [b]_f$, we have $([c]_f, [d]_f) \in |\sqsubset^{fc}|$ or $([c]_f, [d]_f) \in \prec^{fc}$ and $([c]_f, [d]_f) \neq ([a]_f, [b]_f)$. In the first case $([a]_f, [b]_f) \in \alpha(([\![c]_f, [d]_f])$, while in the second case $([a]_f, [b]_f) \in \beta(([\![c]_f, [d]_f])$, which concludes the proof. \square

Lemma 18. *Let $S = (X, \equiv, \sqsubset)$ be a reflexive order structure, $S^f = (X^f, \equiv^f, \sqsubset^f)$ be its folding, $S^c = (X, \equiv^c, \sqsubset^c)$ its closure and $a, b, c, d \in X$. Then*

- *$(a, b) \in |\sqsubset^c|$ implies $[a]_f \equiv^f [b]_f$;*
- *$(a, b), (c, d) \in \prec^c \wedge ([a]_f, [b]_f) \in \beta(([\![c]_f, [d]_f])$ implies $([c]_f, [d]_f) \notin \beta(([\![a]_f, [b]_f])$.*

Proof. Let $a \equiv^c b$ and $a \not\sqsubset^c b$. Then naturally $[a]_f \equiv^{cf} [b]_f$ and $[a]_f \not\sqsubset^{cf} [b]_f$, hence by Proposition 23, $[a]_f \equiv^{fc} [b]_f$ and $[a]_f \not\sqsubset^{fc} [b]_f$ and, by Lemmas 12 and 13, $a \not\sqsubset^* b$. Suppose that $[a]_f \not\equiv^f [b]_f$. Then $[a]_f(\sqsubset^f)^{\otimes} \circ (\equiv^f \cup \nabla^f) \circ (\sqsubset^f)^{\otimes} [b]_f$. By Lemma 12 we can simplify this by reducing compositions with $(\sqsubset^f)^{\otimes}$ obtaining $[a]_f(\equiv^f \cup \nabla^f)[b]_f$ and, since $[a]_f \not\equiv^f [b]_f$ we can reduce it further to $[a]_f \nabla^f [b]_f$. Hence, by the definition of ∇ , we get that $\exists [c]_f, [d]_f [a]_f(\sqsubset^f)^* [c]_f(\sqsubset^f)^* [b]_f \wedge [a]_f(\sqsubset^f)^* [d]_f(\sqsubset^f)^* [b]_f \wedge [c]_f \equiv^f [d]_f$. But, by the definition of $*$, this means that $[a]_f(\sqsubset^f)^* [b]_f$ which gives an obvious contradiction with the initial assumption, hence the supposition that $[a]_f \not\equiv^f [b]_f$ cannot hold and $[a]_f \equiv^f [b]_f$ indeed.

Suppose now that $(a, b), (c, d) \in \prec^c \wedge ([a]_f, [b]_f) \in \beta(([\![c]_f, [d]_f]) \wedge ([c]_f, [d]_f) \in \beta(([\![a]_f, [b]_f])$. Note that we are considering \prec^c , hence we can assume that order structure under consideration is G^c . By Proposition 16, $\beta(([\![a]_f, [b]_f]) = \beta(([\![c]_f, [d]_f])$, hence $([a]_f, [b]_f) \in \beta(([\![a]_f, [b]_f])$, which is in contradiction with the definition of β . Therefore, $(a, b), (c, d) \in \prec^c \wedge ([a]_f, [b]_f) \in \beta(([\![c]_f, [d]_f])$ implies $([c]_f, [d]_f) \notin \beta(([\![a]_f, [b]_f])$, which concludes the proof. \square

Lemma 19. *Let $S = (X, \equiv, \sqsubset)$ be a reflexive order structure and $a, b \in X$. If $a \sqsubset^* b$ then there exists a sequence $\{[c_j]_f\}_{j=1}^n$ of elements from X_f such that $[a]_f = [c_1]_f$, $[c_n]_f = [b]_f$ and $[c_j]_f \sqsubset^f [c_{j+1}]_f$ and $[c_{j+1}]_f \not\sqsubset^f [c_j]_f$.*

Proof. By the definition of transitive closure there exists a sequence $\{d_k\}_{k=1}^m$ which satisfies $a = d_1$, $d_m = b$ and $d_k \sqsubset d_{k+1}$, hence naturally $[a]_f = [d_1]_f$, $[d_m]_f = [b]_f$ and $[d_k]_f \sqsubset^f [d_{k+1}]_f$.

Note that if $[d_{k+1}]_f \sqsubset^f [d_k]_f$ then, by Lemmas 12 and 13, $[d_k]_f = [d_{k+1}]_f$. Hence we can drop redundant elements from $\{d_k\}_{k=1}^m$ obtaining required $\{[c_j]_f\}_{j=1}^n$ with additional property $[c_{j+1}]_f \not\sqsubset^f [c_j]_f$. \square

Example 20. *Recall the structure S from Example 1, its closure S^c from Example 9 and folding S^f from Example 10. The closure of S^f equal to the folding of S^c is depicted in Figure 6.*

Remark 21. Folding is an essential step for the reduction operation defined in Section VII, as it ensures the uniqueness of the result. An unfolded structure can have multiple candidates for a minimal structure with the same closure as demonstrated by the following example:

- $X = \{a, b, c\}$.
- $a \equiv c \equiv b$.
- $a \sqsubset b \sqsubset a$.

Here, there are two candidates for a minimal structure with the same closure: one containing mutex $a \equiv c$, and the other containing mutex $b \equiv c$ (since existence of one of them

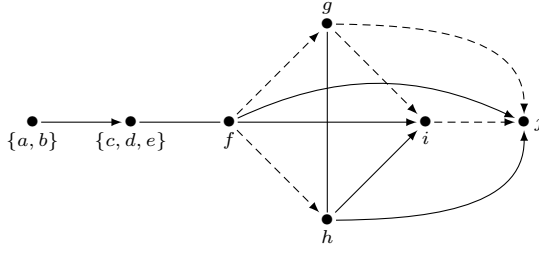


Figure 6. The folding of the structure S^c from Example 9 equal to the closure of the structure S^f from Example 10.

implies the other). If we fold this structure events a and b will fall into one equivalence class $\{a, b\}$ and the reduction will contain mutex $\{a, b\} \rightleftharpoons \{c\}$. Note that folding is not required for finding the *closure* as the latter is always unique.

VI. INTEGRATION

Let $S = (X, \rightleftharpoons, \sqsubset)$ be a reflexive order structure and $S^f = (X^f, \rightleftharpoons^f, \sqsubset^f)$ be its folding. The integration will be denoted by $S^i = (X, \rightleftharpoons^i, \sqsubset^i)$, where:

- $a \rightleftharpoons^i b$ if $[a]_f \rightleftharpoons^f [b]_f$.
- $a \sqsubset^i b$ if $[a]_f \sqsubset^f [b]_f$.

Note that for a generalised mutex order structure G we have $G^i = G$, and that any integration satisfies axioms **G1**, **G2** (as any order structure) and **G4**.

We formulate the following two propositions to characterise the integration operation. The first one formalises the relationship between closure and integration (which can be considered a partial closure): the closures of S , S^i and S^c are equal (the last one is due to the idempotence of the closure).

Proposition 22. $S^c \in [S]_{gmos}$ and $S^i \in [S]_{gmos}$.

Proof. The first part bases on results already proved in [3] in the case of irreflexive traces. Hence it is sufficient to prove that $S \triangleleft S^i \triangleleft S^c$.

Assume first that $a \sqsubset b$ ($a \rightleftharpoons b$). Then, by the definition of folding $[a]_f \sqsubset^f [b]_f$ ($[a]_f \rightleftharpoons^f [b]_f$ respectively), and so, by the definition of integration, $a \sqsubset^i b$ ($a \rightleftharpoons^i b$ respectively).

Let $a \sqsubset^i b$. Then, by the definition of integration, $[a]_f \sqsubset^f [b]_f$, and so there exist $x \in [a]_f$ and $y \in [b]_f$ such that $x \sqsubset y$. Hence, by Lemma 12, $a \sqsubset^* x \sqsubset y \sqsubset^* b$ and so (also by Lemma 12) $a \sqsubset^c b$.

Finally, assume that $a \rightleftharpoons^i b$. Then, by the definition of integration, $[a]_f \rightleftharpoons^f [b]_f$, and so there exist $x \in [a]_f$ and $y \in [b]_f$ such that $x \rightleftharpoons y$. By Lemma 12, $a \sqsubset^* x$ and $y \sqsubset^* b$, hence $a \sqsubset^* \circ \rightleftharpoons \circ \sqsubset^* b$ and so $a \rightleftharpoons^c b$, which concludes the proof. \square

We also prove that one can intersect integrations of two structures with the same closure without losing any essential information. This is a fundamental property for the discussion conducted in this paper.

Proposition 23. If $T \in [S]_{gmos}$ then $(S^i \cap T^i) \in [S]_{gmos}$.

Proof. Let $S^i = (X, \rightleftharpoons^i, \sqsubset^i)$, $T^i = (X, \rightleftharpoons^i, \sqsubset^i)$, $(S^i \cap T^i) = (X, \rightleftharpoons_{\cap}, \sqsubset_{\cap})$, and $S^c = T^c = (X, \rightleftharpoons^c, \sqsubset^c)$. Note that it is sufficient to prove that S^i is included in the closure of $(S^i \cap T^i)$ (since the proof for similar inclusion with T^i instead of S^i may be conducted analogously).

Let us focus on foldings S^f and T^f , which by the definition of integration are compact versions of S^i and T^i respectively. We argue that every element of S^f is included in the closure of $S^f \cap T^f$. Assume that $a \sqsubset_S^i b$ (equivalently $[a]_f \sqsubset_S^f [b]_f$) but $a \not\sqsubset_{\cap} b$. Since $a \sqsubset^c b$ we get that $a \sqsubset_T^*$ and so $[a]_f \sqsubset_T^f [b]_f$. Hence, by Lemma 19 there exist a sequence $\{[c_j]_f\}_{j=1}^n$ of elements from X_f such that $[a]_f = [c_1]_f$, $[c_n]_f = [b]_f$ and $[c_j]_f \sqsubset_T^f [c_{j+1}]_f$ and $[c_{j+1}]_f \not\sqsubset_T^f [c_j]_f$. If for all $i = 1 \dots n-1$ we have $[c_j]_f \sqsubset_S^f [c_{j+1}]_f$, we are done. Otherwise we proceed in the same way with any $[c_j]_f \not\sqsubset_S^f [c_{j+1}]_f$ (searching for another sequences and exchanging roles of S and T). Note that all new elements in those sequences need to be distinct, since otherwise all elements between them would be equal and we obtain a contradiction with Lemma 19. Finally, we cannot proceed infinitely, since X is finite and in each step we engage new elements from X , so at the end we construct a sequence $\{[d_k]_f\}_{k=1}^m$ such that $[a]_f = [d_1]_f$, $[d_m]_f = [b]_f$ and $(d_k)_{\cap} \sqsubset (d_{k+1})_{\cap}$.

Let $a \rightleftharpoons_S^i b$ but $a \not\rightleftharpoons_T^i b$ (hence also $[a]_f \not\rightleftharpoons_T^f [b]_f$). Then, by Lemma 17 there exists $[c]_f \rightleftharpoons_S^f [d]_f$ such that $([a]_f, [b]_f) \in \alpha((\leftarrow [c]_f, \leftarrow [d]_f))$ or $([a]_f, [b]_f) \in \beta((\leftarrow [c]_f, \leftarrow [d]_f))$. In the first case, $c \rightleftharpoons_{\cap} d$ and by the definition of α , $a \rightleftharpoons_{\cap} b$ and we are done. In the second case, if $[c]_f \rightleftharpoons_T^f [d]_f$ we got $c \rightleftharpoons_{\cap} d$ and using the definition of β obtain $a \rightleftharpoons_{\cap} b$. However, it might happen that $c \not\rightleftharpoons d$. Since $c \rightleftharpoons^c d$, we can repeat the procedure. Since X is finite and by Lemma 18, we cannot continue infinitely getting at the end $x \rightleftharpoons_{\cap} y$ such that (by multiple use of Proposition 16) $([a]_f, [b]_f) \in \beta((\leftarrow [x]_f, \leftarrow [y]_f))$. This proves that $a \rightleftharpoons_{\cap} b$ and concludes the proof. \square

As an immediate corollary, we can think about integration as a partial closure and simplify the closure operator for structures $S = S^i$ in the following way:

- $a \sqsubset^c b \iff a \sqsubset^* b$,
- $a \rightleftharpoons^c b \iff a(\rightleftharpoons \cup \nabla^{sym})b$.

Having a relational structure $S = (X, \rightleftharpoons, \sqsubset)$ and its folding $S^f = (X^f, \rightleftharpoons^f, \sqsubset^f)$ one can compute the integration S^i in time $O(|X|^2)$ (browsing all possible pairs of events $u, v \in X$ and setting relations between them according to relations between $[u]_f$ and $[v]_f$).

Example 24. Recall the reflexive order structure S from Figure 1. The integration S^i of S is depicted in Figure 7.

Remark 25. Note that Proposition 23 does not hold for arbitrary structures. Consider the following example already introduced in Remark 21:

- $S_1 = (X, \rightleftharpoons_1, \sqsubset)$, $S_2 = (X, \rightleftharpoons_2, \sqsubset)$.
- $X = \{a, b, c\}$.
- $a \rightleftharpoons_1 c$, $c \rightleftharpoons_2 b$.
- $a \sqsubset b \sqsubset a$.

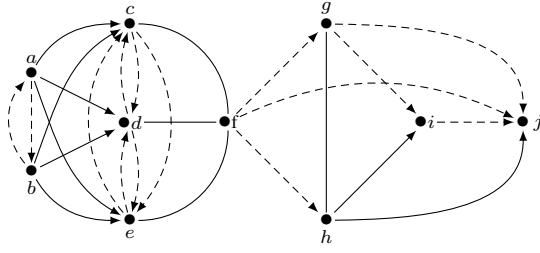


Figure 7. The integration of the structure S from Example 1.

Here, the closures of both S_1 and S_2 are equal (containing $a \rightleftharpoons^c c \rightleftharpoons^c b$), while the relation \rightleftharpoons of the intersection of S_1 and S_2 is empty.

VII. REDUCTION

The most important condition that has to be met by the reduction can be formulated as follows:

The reduction is a unique effect of a simplifying procedure that is an invariant for all structures with equal closure.

In other words, we want the equality of the reduction of two structures to be equivalent with the equality of their closures. Moreover, the reduction of S^f should be contained in S . Intuitively, the purpose of the reduction operation is similar to that of the *Hasse diagram* of a partial order: it is a canonical and concise representation of a structure, and can often be easier for visualisation and reasoning.

The *reduction* of a reflexive order structure S is denoted as $S^r = (X^r, \rightleftharpoons^r, \sqsubset^r)$ and is formally defined as the folding of

$$\bigcap T^i \text{ where } T \in [S]_{gmos}.$$

Alternatively, the *reduction* of a generalised mutex order structure G (denoted by $G^r = (X^r, \rightleftharpoons^r, \sqsubset^r)$) may be formally defined as:

- $X^r \stackrel{\text{df}}{=} X^f$.
- $\rightleftharpoons^r \stackrel{\text{df}}{=} \upharpoonright^f \cup \prec^d \cup (\prec^d)^{-1}$.
- $\sqsubset^r \stackrel{\text{df}}{=} \sqsubset^f \setminus ((\sqsubset^f \setminus Id_{X^f}) \circ (\sqsubset^f \setminus Id_{X^f}))$.

The folded structure $G^f = (X^f, \rightleftharpoons^f, \sqsubset^f)$ is as in Section V. Pair $(\prec^f, \upharpoonright^f)$ is obtained by decomposing \rightleftharpoons^f as explained in Section III. Reduced relation \prec^d is derived from \prec^f by dropping all **G5**-induced mutexes $\alpha(\upharpoonright^f)$ and $\beta(\prec^f)$:

$$\prec^d \stackrel{\text{df}}{=} \prec^f \setminus (\beta(\prec^f) \cup \alpha(\upharpoonright^f)).$$

The time and space complexity of the reduction is dominated by computation of $\alpha(\upharpoonright^f)$, and $\beta(\prec^f)$ which takes $O(|\upharpoonright^f| \cdot |\prec^f|)$ and $O(|\prec^f|^2)$, respectively.

Example 26. Recall the reflexive order structure S from Example 1. The reduction S^r of S is depicted in Figure 8.

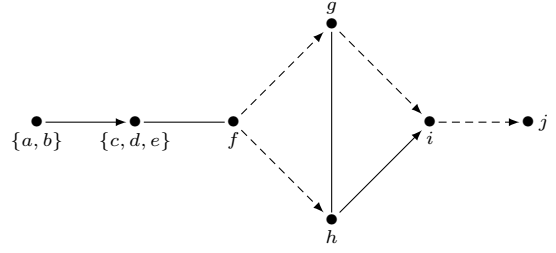


Figure 8. The reduction of the structure S from Example 1.

Algorithm 1: Reduction algorithm

$O(|S| + |X^f| \cdot |\sqsubset^f| + |\rightleftharpoons^f| \cdot |\rightleftharpoons^f \cap \sqsubset^{fc}|)$

Input: An order structure $S = (X, \rightleftharpoons, \sqsubset)$

Output: The reduction $S^r = (X^f, \rightleftharpoons^f, \sqsubset^f)$

```

// Compute the folding of  $S$  ( $S^f = (X^f, \rightleftharpoons^f, \sqsubset^f)$ )
1 Compute  $X^f$  using Tarjan's algorithm          /*  $O(|X| + |\sqsubset|)$  */
2 foreach  $(x, y) \in \rightleftharpoons$  do                        /*  $O(|\rightleftharpoons|)$  */
3   | Add  $([x]_f, [y]_f)$  to  $\rightleftharpoons^f$ 
4 foreach  $(x, y) \in \sqsubset$  do                          /*  $O(|\sqsubset|)$  */
5   | Add  $([x]_f, [y]_f)$  to  $\sqsubset^f$ 

// Compute the folded transitive closure  $\sqsubset^{fc}$ 
// and folded transitive reduction  $\rightleftharpoons^{fr}$ 
6 foreach  $x \in X^f$  do                                /*  $O(|X^f| \cdot |\sqsubset^f|)$  */
7   | foreach  $y \in X^f$  visited during BFS of the graph  $(X^f, \sqsubset^f)$  do
8     | Add  $(x, y)$  to  $\sqsubset^{fc}$ 
9 foreach  $(x, y) \in \sqsubset^f$  do                            /*  $O(|X^f| \cdot |\sqsubset^f|)$  */
10  | if  $\forall z \in X^f \setminus \{x, y\} (x, z) \notin \sqsubset^f \text{ or } (z, y) \notin \sqsubset^f$  then
11    | Add  $(x, y)$  to  $\rightleftharpoons^{fr}$ 

// Precompute  $(\rightleftharpoons^f \cap \sqsubset^{fc})$ 
12 foreach  $(x, y) \in \rightleftharpoons^f$  do                            /*  $O(|\rightleftharpoons^f|)$  */
13  | if  $(x, y) \in \sqsubset^{fc}$  then
14    | Add  $(x, y)$  to  $\rightleftharpoons^f \cap \sqsubset^{fc}$ 

// Compute the mutex reduction
15 Initialize  $\rightleftharpoons^{fr}$  as  $\rightleftharpoons^f$                                 /*  $O(|\rightleftharpoons^f|)$  */
16 foreach mutex  $(u, v) \in \rightleftharpoons^f$  do                        /*  $O(|\rightleftharpoons^f| \cdot |\rightleftharpoons^f \cap \sqsubset^{fc}|)$  */
17  | foreach aligned mutex  $(x, y) \in \rightleftharpoons^f \cap \sqsubset^{fc}$  do
18    | if  $(u, v) \in \sqsubset^{fc}$  //  $(u, v)$  is aligned
19      | then
20        | if  $x \sqsubset^{fc} u \wedge v \sqsubset^{fc} y \wedge (u, v) \neq (x, y)$  then
21          | remove  $(x, y)$  from  $\rightleftharpoons^{fr}$ 
22      | else
23        | if  $x \sqsubset^{fc} u \wedge x \sqsubset^{fc} v \wedge u \sqsubset^{fc} y \wedge v \sqsubset^{fc} y$  then
24          | remove  $(x, y)$  from  $\rightleftharpoons^{fr}$ 

```

VIII. ALGORITHM

In this section we present the details of the algorithm for reduction of order structures. After that we describe necessary changes to obtain the algorithm for closure.

The algorithm for the reduction procedure directly follows

the theory presented in previous sections. We start from computing equivalence classes of folding relation \equiv_f . Using Tarjan's algorithm [8], this can be done in time complexity $O(|X| + |\sqsubset|)$. Next we recompute the mutual exclusion and weak causality of the folded structure (in fact constructing the condensation graph). The complexity of this step is $O(|\equiv| + |\sqsubset|)$.

Having obtained $S^f = (X^f, \equiv^f, \sqsubset^f)$ we need to compute the transitive closure and transitive reduction of \sqsubset^f , namely, \sqsubset^{fc} and \sqsubset^{fr} . This can be done in $O(|X^f| \cdot |\sqsubset^f|)$ time.

In order to distinguish crossing and aligned mutexes we compute $\equiv^f \cap \sqsubset^{fc}$ in time complexity $O(|\equiv^f|)$.

The last and the most important part is the computation of the reduced set of folded mutexes. We do it by processing each aligned mutex from $\equiv^f \cap \sqsubset^{fc}$ and checking whether there exists a different mutex inducing it. Making a test we distinguish between crossing and aligned mutexes. The complexity of this final stage is $O(|\equiv^f| \cdot |\equiv^f \cap \sqsubset^{fc}|)$, giving the total time complexity of the reduction algorithm $O(|X| + |\equiv| + |\sqsubset| + |X^f| \cdot |\sqsubset^f| + |\equiv^f| \cdot |\equiv^f \cap \sqsubset^{fc}|)$.

Note that a part of the reduction algorithm is the computation of X^f and its closure X^{fc} . This is the large part of the folded version of the closure. In order to compute the remaining part (\equiv^{fc}) it is sufficient to change a bit the presented reduction algorithm. The range of the loop from the line 17 shall be changed from $\equiv^f \cap \sqsubset^{fc}$ to \sqsubset^{fc} and adding to \equiv^{fc} positively verified relationships instead of removing them from \equiv^{fr} . Thus, the time complexity of the whole procedure increases to $O(|S| + |X^f| \cdot |\sqsubset^f| + |\equiv^f| \cdot |\sqsubset^{fc}|)$.

IX. ANALYSIS OF THE MOTIVATING EXAMPLE

Recall the motivating example presented on Figure 1. Taking into account the way how program statements modify values of the variables we model the relations between them as follows. A more precise framework for dealing with labelled generalized mutex order structures can be found in [3]. Below we describe some typical situations and resulting structures.

The first problem with the simultaneity is the modification of the same variable at the same time by different occurrences of statements. Therefore, all equilabelled events are mutex-connected, which supports also the irreflexibility of simultaneity presumed in generalized mutex order structure. The other mutex-connected pairs (basing on modification of k) are (y, b) , (y, e) and (b, e) . Separately we need to consider weak causality between those pairs. In the case of equilabelled events we add weak causality to satisfy label linearity postulated in [3]. Since the order is important, such an operation is also necessary in the case of pairs (y, b) and (y, e) . On the other hand, we do not need to introduce any weak causality arcs between b and e .

The other problem is related to the different result of the parallel and sequential execution of two consecutive events. However, in this case we need to process each pair of statements one by one setting the weak causality between them at the same time. The example of such situation are the statements b and c . Their parallel execution is equivalent to

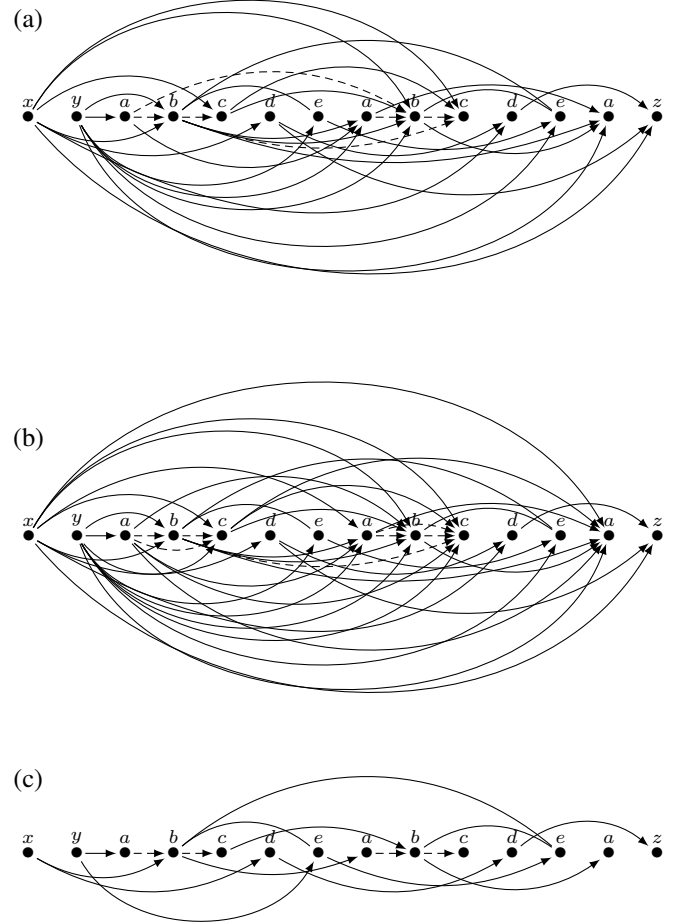


Figure 9. The generalized mutex order structure (a) for the function presented in Figure 1, its closure (b) and its reduction (c).

sequential execution bc , while cb produces the different result. This is why we put weak causality arc between all occurrences of b and c , adding a mutex in the case of (the event labelled by) c appearing before (the event labelled by) b . On the other hand, processing a consecutive execution of the pair of events labelled by d and e we do not add any relationships to GMOS, since those statements can be executed in parallel as well as in any order without affecting the result.

The resulting generalized mutex order structure is presented on Figure 9 (a), its closure on Figure 9 (b), while its reduction on Figure 9 (c). Note that both structures presented on Figures 9 (a) and (b) are rather complex, while the one depicted on Figure 9 (c) is more suitable for further investigation, especially it is easy to conclude that except from dependencies related to variable modifications one may consider also causal dependencies implied by program flow logic (the natural barriers/synchronization points generated by loop conditions).

X. SUMMARY AND FUTURE WORK

In the paper we discussed an analogue of transitive reduction for order structures. At the first glance a surprising phe-

nomenon is the possible enlargement of the structure during the reduction procedure. The main cause of this effect is, in contrast to partial orders, the lack of determinism in the choice of redundant relationships between events. We address this issue by transferring the problem to the realm of folded structures, which can be seen as compact versions of the original order structures. The causal part of the folded order structure becomes a partial order, restoring the uniqueness. In addition, it is impossible to enlarge an object during the reduction of a folded structure.

We discuss a new (compared to partial orders) type of relationships between events – the mutual exclusion. We distinguish the pure mutual exclusion (named crossing mutex) and prove that it is irreducible at the level of folded structures. After that, we discuss conditions under which the other type of mutual exclusion (named aligned mutex) may be reduced. We also propose an algorithm which utilises the distinction between crossing and aligned mutexes to restrict the number of objects to be compared by the reduction procedure. In this way we obtain an algorithm with the time complexity quadratic with respect to the size of the original structure.

The paper is supported by a software tool with a graphical user interface providing the implementation of the presented algorithms together with the visualization of the closure and reduction of order structures [7].

Our future work includes optimisation of the proposed reduction algorithm, as well as a further investigation of properties of generalised mutex order structures. In particular, it is interesting to study the distinction between aligned and crossing mutexes, which can lead to new axiomatic characterisations of generalised mutex order structures.

ACKNOWLEDGEMENTS

We are grateful to Maciej Koutny and Victor Khomenko for many fruitful discussions. This research was partially supported by EPSRC grant EP/N031768/1 (project POETS) and the Polish National Science Center through grant No.2013/09/D/ST6/03928.

REFERENCES

- [1] Ales Bizjak and Andrej Bauer. *ALG User Manual*, Faculty of Mathematics and Physics, University of Ljubljana, 2011.
- [2] Ryszard Janicki, Jetty Keijn, Maciej Koutny, and Łukasz Mikulski. Characterising concurrent histories. Technical Report 1, 2015.
- [3] Ryszard Janicki, Jetty Keijn, Maciej Koutny, and Łukasz Mikulski. Step traces. *Acta Informaticae*, 53(1):35–65, 2016.
- [4] Ryszard Janicki and Maciej Koutny. Structure of concurrency. *Theoretical Computer Science*, 112(1):5–52, 1993.
- [5] Simon Marlow, Simon Peyton Jones, Edward Kmett, and Andrey Mokhov. Desugaring haskell’s do-notation into applicative operations. In *Proceedings of the 9th International Symposium on Haskell*, pages 92–104. ACM, 2016.
- [6] Łukasz Mikulski and Maciej Koutny. Folded hasse diagrams of combined traces. *Information Processing Letters*, 114(4):208–216, 2014.
- [7] Marcin Piątkowski. Relational structure viewer . <http://folco.mat.umk.pl/rs/>, 2017.
- [8] Robert Endre Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2):146–160, 1972.