A ADDITIONAL MATERIAL FOR SECTION 4

A.1 MSO-definable properties

In this sections we give MSO formulas for some MSO-definable properties that are used throughout the paper.

Transitive Closure. Given a binary relation R, we can express its reflexive transitive closure R^* in MSO as

$$x\,R^*\,y = \forall X.(x \in X \;\wedge\; forward\_closed(X)) \implies y \in X$$
$$forward\_closed(X) = \forall z.\forall t.(z \in X \;\wedge\; z\,R\,t) \implies t \in X$$

The transitive (but not necessarily reflexive) closure of R can also be expressed as

$$x\,R^+\,y = \forall X.\;\big(\forall z,t\;(z \in X \cup \{x\} \wedge z\,R\,t) \implies t \in X\big) \implies y \in X$$

Acyclicity. Given a binary relation R, we can use MSO to express the acyclicity of R, or equivalently, the fact that its transitive closure R^+ is irreflexive.

$$\Phi_{acyclic} = \neg\exists x.(x\,R^+\,x).$$

A.2 Omitted proofs of Section 4

Mailbox. We show here that the two alternative definitions of mb-MSC that we gave are equivalent.

Proposition A.1. *Definition 2.5 and Definition 4.2 of* mb*-MSC are equivalent.*

Proof. ($\Rightarrow$) We show that if M is a mb-MSC, according to Definition 4.2, then it is also a mb-MSC, according to Definition 2.5. By definition of $\preceq_{\mathtt{mb}}$, we must have (i) $s \preceq_{\mathtt{mb}} s'$ for any two matched send events s and s' addressed to the same process, such that $r \rightarrow^+ r$, where $s \lhd r$ and $s' \lhd r'$, and (ii) $s \preceq_{\mathtt{mb}} s'$, if s and s' are a matched and an unmatched send event, respectively. If $\preceq_{\mathtt{mb}}$ is a partial order, we can find at least one linearization $\rightsquigarrow$ such that $\preceq_{\mathtt{mb}} \subseteq \rightsquigarrow$; such a linearization satisfies the conditions of Definition 2.5.
($\Leftarrow$) We show that if M is not a mb-MSC, according to Definition 4.2, then it is also not a mb-MSC, according to Definition 2.5. Since $\preceq_{\mathtt{mb}} = (\rightarrow \cup \lhd \cup \sqsubset_{\mathtt{mb}})^*$ is not a partial order, $\preceq_{\mathtt{mb}}$ must be cyclic[7]. If $\preceq_{\mathtt{mb}}$ is cyclic, it means that we cannot find a linearization $\rightsquigarrow$ such that $\preceq_{\mathtt{mb}} \subseteq \rightsquigarrow$. In other words, we cannot find a linearization where (i) $s \rightsquigarrow s'$ for any two matched send events s and s' addressed to the same process, such that $r \rightarrow^+ r$, where $s \lhd r$ and $s' \lhd r'$, and (ii) $s \rightsquigarrow s'$, if s and s' are a matched and an unmatched send event, respectively. It follows that M is not a mb-MSC also according to Definition 2.5. □

FIFO 1−n. We show that the two alternative definitions of onen-MSC that we gave are equivalent.

Proposition A.2. *Definition 2.6 and Definition 4.3 of* onen*-MSC are equivalent.*

Proof. ($\Rightarrow$) We show that if M is a onen-MSC, according to Definition 4.3, then it is also a onen-MSC, according to Definition 2.6. By definition of $\preceq_{\mathtt{1n}}$, we must have (i) $r \preceq_{\mathtt{1n}} r'$ for any two receive events r and r' whose matched send events s and s' are such that $s \rightarrow^+ s'$, and (ii) $s \preceq_{\mathtt{1n}} s'$, if s and s' are a matched and an unmatched send event executed by the same process, respectively. If $\preceq_{\mathtt{1n}}$ is a partial order, we can find at least one linearization $\rightsquigarrow$ such that $\preceq_{\mathtt{1n}} \subseteq \rightsquigarrow$; such a linearization satisfies the conditions of Definition 2.6.
($\Leftarrow$) We show that if M is not a onen-MSC, according to Definition 4.3, then it is also not a onen-MSC, according to Definition 2.6. Since $\preceq_{\mathtt{1n}} = (\rightarrow \cup \lhd \cup \sqsubset_{\mathtt{1n}})^*$ is not a partial order, $\preceq_{\mathtt{1n}}$ must be

[7] $\preceq_{\mathtt{mb}}$ is reflexive and transitive by definition, if it were also acyclic it would be a partial order

cyclic. If $\preceq_{1n}$ is cyclic, it means that we cannot find a linearization $\rightsquigarrow$ such that $\preceq_{1n} \subseteq \rightsquigarrow$. In other words, we cannot find a linearization where (i) $r \rightsquigarrow r'$ for any two receive events r and r' whose matched send events s and s' are such that $s \rightarrow^{+} s'$, and (ii) $s \rightsquigarrow s'$, if s and s' are a matched and an unmatched send event executed by the same process, respectively. It follows that M is not a onen-MSC also according to Definition 2.6. □

FIFO n−n. We show here the missing proofs for the equivalence of the two definitions of nn-MSC that we gave.

Proposition A.3. *Let M be an MSC. Given two matched send events s_1 and s_2, and their respective receive events r_1 and r_2, $r_1 \sqsubset_{nn} r_2 \implies s_1 \sqsubset_{nn} s_2$.*

Proof. Follows from the definition of $\sqsubset_{nn}$. We have $r_1 \sqsubset_{nn} r_2$ if either:

- $r_1 \preceq_{1n/mb} r_2$. Two cases: either (i) $s_1 \preceq_{1n/mb} s_2$, or (ii) $s_1 \not\preceq_{1n/n1} s_2$. The first case clearly implies $s_1 \sqsubset_{nn} s_2$, for rule 1 in the definition of $\sqsubset_{nn}$. The second too, because of rule 3.
- $r_1 \not\preceq_{1n/n1} r_2$, but $r_1 \sqsubset_{nn} r_2$. This is only possible if rule 2 in the definition of $\sqsubset_{nn}$ was used, which implies $s_1 \preceq_{1n/mb} s_2$ and, for rule 1, $s_1 \sqsubset_{nn} s_2$.

□

Proposition 4.5. *Let M be an MSC. If $\sqsubset_{nn}$ is cyclic, then M is not a nn-MSC.*

Proof. According to Definition 2.7, an MSC is FIFO n−n if it has at least one nn-linearization. Note that, because of how it is defined, any nn-linearization is always both a mb and a onen-linearization. It follows that the cyclicity of $\preceq_{1n/mb}$ (not $\sqsubset_{nn}$) implies that M is not FIFO n−n, because it means that we are not even able to find a linearization that is both mb and FIFO 1−n. Moreover, since in a nn-linearization the order in which messages are sent matches the order in which they are received, and unmatched send events can be executed only after matched send events, a nn-MSC always has to satisfy the constraints imposed by the $\sqsubset_{nn}$ relation. If $\sqsubset_{nn}$ is cyclic, then for sure there is no nn-linearization for M. □

Proposition 4.8. *Given an MSC M, Algorithm 1 terminates correctly if $\sqsubset_{nn}$ is acyclic.*

Proof. We want to prove that, if $\sqsubset_{nn}$ is acyclic, step 4 of the algorithm is never executed, i.e. it terminates correctly. Note that the acyclicity of $\sqsubset_{nn}$ implies that the EDG of M is a DAG. Moreover, at every step of the algorithm we remove nodes and edges from the EDG, so it still remains a DAG. The proof proceeds by induction on the number of events added to the linearization.
Base case: no event has been added to the linearization yet. Since the EDG is a DAG, there must be an event with in-degree 0. In particular, this has to be a send event (a receive event depends on its respective send event, so it cannot have in-degree 0). If it is a matched send event, step 1 is applied. If there are no matched send events, step 2 is applied on an unmatched send. We show that it is impossible to have an unmatched send event of in-degree 0 if there are still matched send events in the EDG, so either step 1 or 2 are applied in the base case. Let s be one of those matched send events and let u be an unmatched send. Because of rule 4 in the definition of $\sqsubset_{nn}$, we have that $s \sqsubset_{nn} u$, which implies that u cannot have in-degree 0 if s is still in the EDG.
Inductive step: we want to show that we are never going to execute step 4. In particular, Step 4 is executed when none of the first three steps can be applied. This happens when there are no matched send events with in-degree 0 and one of the following holds:

- *There are still matched send events in the EDG with in-degree > 0, there are no unmatched messages with in-degree 0, and there is no receive event r with in-degree 0 in the EDG, such that r is the receive event of the first message whose sent event was already added to the linearization.* Since the EDG is a DAG, there must be at least one receive event with in-degree 0. We want to show that, between these receive events with in-degree 0, there is also the receive event r of the first message whose send event was added to the linearization, so that we can apply step 3 and step 4 is not executed. Suppose, by contradiction, that r has in-degree > 0, so it depends on other events. For any maximal chain in the EDG that contains one of these events, consider the first event e, which clearly has in-degree 0. In particular, e cannot be a send event, because we would have applied step 1 or step 2. Hence, e can only be a receive event for a send event that was not the first added to the linearization (and whose respective receive still has not been added). However, this is also impossible, since $r_e \sqsubset_{\mathsf{nn}} r$ implies $s_e \sqsubset_{\mathsf{nn}} s$, according to Proposition A.3, and we could not have added s to the linearization before s_e. Because we got to a contradiction, the hypothesis that r has in-degree > 0 must be false, and we can indeed apply step 3.

- *There are still matched send events in the EDG with in-degree > 0, there is at least one unmatched message with in-degree 0, and there is no receive event r with in-degree 0 in the EDG, such that r is the receive event of the first message whose sent event was already added to the linearization.* We show that it is impossible to have an unmatched send event of in-degree 0 if there are still matched send events in the EDG. Let s be one of those matched send events and let u be an unmatched send. Because of rule 4 in the definition of $\sqsubset_{\mathsf{nn}}$, we have that $s \sqsubset_{\mathsf{nn}} u$, which implies that u cannot have in-degree 0 if s is still in the EDG.
- *There are no more matched send events in the EDG, there are no unmatched messages with in-degree 0, and there is no receive event r with in-degree 0 in the EDG, such that r is the receive event of the first message whose sent event was already added to the linearization.* Very similar to the first case. Since the EDG is a DAG, there must be at least one receive event with in-degree 0. We want to show that, between these receive events with in-degree 0, there is also the receive event r of the first message whose send event was added to the linearization, so that we can apply step 3 and step 4 is not executed. Suppose, by contradiction, that r has in-degree > 0, so it depends on other events. For any maximal chain in the EDG that contains one of these events, consider the first event e, which clearly has in-degree 0. In particular, e cannot be a send event, because by hypothesis there are no more send events with in-degree 0 in the EDG. Hence, e can only be a receive event for a send event that was not the first added to the linearization (and whose respective receive still has not been added). However, this is also impossible, since $r_e \sqsubset_{\mathsf{nn}} r$ implies $s_e \sqsubset_{\mathsf{nn}} s$ (see Proposition A.3), and we could not have added s to the linearization before s_e. Because we got to a contradiction, the hypothesis that r has in-degree > 0 must be false, and we can indeed apply step 3.

We showed that, if $\sqsubset_{\mathsf{nn}}$ is acyclic, the algorithm always terminates correctly and computes a valid nn-linearization. □

B ADDITIONAL MATERIAL FOR SECTION 5

Proposition B.1. *Every* co-*MSC is a* p2p-*MSC.*

Proof. According to Definition 2.4, and MSC is co if, for any two send events s and s', such that $\lambda(s) = Send(\_, q, \_)$, $\lambda(s') = Send(\_, q, \_)$, and $s \leq_{hb} s'$, we have either (i) $s, s' \in Matched(M)$ and $r \rightarrow^* r'$, where r and r' are two receive events such that $s \lhd r$ and $s' \lhd r'$, or (ii) $s' \in Unm(M)$. The conditions imposed by the Definition 2.3 of p2p are clearly satified by any co-MSC; in particular, note that $s \rightarrow^+ s'$ implies $s \leq_{hb} s'$. □

Proposition 5.3. *Every* mb-*MSC without unmatched messages is a* onen-*MSC.*

Proof. We show that the contrapositive is true, i.e. if an MSC is not FIFO $1\text{-}n$ (and it does not have unmatched messages), it is also not mailbox. Suppose M is an asynchronous MSC, but not FIFO $1\text{-}n$. There must be a cycle ξ such that $e \preceq_{1n} e$, for some event e. Recall that $\preceq_{1n} = (\rightarrow \cup \sqsubset_{1n} \cup \sqsubset_{mb})^*$ and $\leq_{hb} = (\rightarrow \cup \sqsubset_{1n})^*$. We can always explicitely write a cycle $e \preceq_{1n} e$ only using $\sqsubset_{1n}$ and $\leq_{hb}$. For instance, there might be a cycle $e \preceq_{1n} e$ because we have that $e \sqsubset_{1n} f \leq_{hb} g \sqsubset_{1n} h \sqsubset_{1n} i \leq_{hb} e$. Consider any two adiacent events r_1 and r_2 in the cycle ξ, where ξ has been written using only $\sqsubset_{1n}$ and $\leq_{hb}$, and we never have two consecutive $\leq_{hb}$. We have two cases:

(1) $r_1 \sqsubset_{1n} r_2$. By definition of $\sqsubset_{1n}$, r_1 and r_2 must be two receive events, since we are not considering unmatched send events, and $s_1 \rightarrow^+ s_2$, where s_1 and s_2 are the send events that match with r_1 and r_2, respectively.

(2) $r_1 \leq_{hb} r_2$. Since M is asynchronous by hyphotesis, ξ has to contain at least one $\sqsubset_{1n}$; recall that we also wrote ξ in such a way that we do not have two consecutive $\leq_{hb}$. It is not difficult to see that r_1 and r_2 have to be receive events, since they belong to ξ. Let s_1 and s_2 be the two send events such that $s_1 \lhd r_1$ and $s_2 \lhd r_2$. We have two cases:
   (a) s_2 is in the causal path between r_1 and r_2, i.e. $s_1 \lhd r_1 \leq_{hb} s_2 \lhd r_2$. In particular, note that $s_1 \leq_{hb} s_2$.
   (b) s_2 is not in the causal path between r_1 and r_2, hence there must be a message m_k received by the same process that executes r_2, such that $r_1 \leq_{hb} s_k \lhd r_k \rightarrow^+ r_2$, where r_k is the send event of m_k. Since messages m_k and m_2 are received by the same process and $r_k \rightarrow^+ r_2$, we should have $s_k \sqsubset_{mb} s_2$, according to the mailbox semantics. In particular, note the we have $s_1 \leq_{hb} s_k \sqsubset_{mb} s_2$.

   In both case (a) and (b), we conclude that $s_1 \preceq_{mb} s_2$.

Notice that, for either cases, a relation between two receive events r_1 and r_2 implies a relation between the respective send events s_1 and s_2, according to the mailbox semantics. It follows that ξ, which is a cycle for the $\preceq_{1n}$ relation, always implies a cycle for the $\preceq_{mb}$ relation. □

Proposition B.2. *Every* nn*-MSC is a* onen*-MSC.*

Proof. Consider Definition 2.7 and Definition 2.6. They are identical, except for the fact that in the FIFO $n\text{-}n$ case we consider any two send events, and not just those that are sent by a same process. This is enough to show that each nn-linearization is also a onen-linearization and, therefore, each nn-MSC is a onen-MSC. □

Proposition B.3. *Every* rsc*-MSC is a* nn*-MSC.*

Proof. Consider Definition 2.8 and Definition 2.7. Let us pick an rsc-linearization $\rightsquigarrow$. If every send event is immediately followed by its matching receive event, and we do not have unmatched messages, then $\rightsquigarrow$ is also a nn-linearization; note that, for any two send events s and s' such that $s \rightsquigarrow s'$, we also have $r \rightsquigarrow r'$, where $s \lhd r$ and $s' \lhd r'$. It follows that each rsc-MSC is a nn-MSC. □

C ADDITIONAL MATERIAL FOR SECTION 6

C.1 Communicating finite state machines

We now recall the definition of communicating systems (aka communicating finite-state machines or message-passing automata), which consist of finite-state machines A_p (one for every process $p \in \mathbb{P}$) that can communicate through channels from $\mathbb{C}$.

Definition C.1. A *system of communicating finite state machines* over the set $\mathbb{P}$ of rocesses and the set $\mathbb{M}$ of messages is a tuple $\mathcal{S} = (A_p)_{p\in\mathbb{P}}$. For each $p \in \mathbb{P}$, $A_p = (Loc_p, \delta_p, \ell_p^0)$ is a finite transition system where Loc_p is a finite set of local (control) states, $\delta_p \subseteq Loc_p \times \Sigma_p \times Loc_p$ is the transition relation, and $\ell_p^0 \in Loc_p$ is the initial state.

Given $p \in \mathbb{P}$ and a transition $t = (\ell, a, \ell') \in \delta_p$, we let $source(t) = \ell$, $target(t) = \ell'$, $action(t) = a$, and $msg(t) = m$ if $a \in Send(\_, \_, m) \cup Rec(\_, \_, m)$.

Let $M = (\mathcal{E}, \rightarrow, \lhd, \lambda)$ be an MSC. A *run* of $\mathcal{S}$ on M is a mapping $\rho : \mathcal{E} \rightarrow \bigcup_{p \in \mathbb{P}} \delta_p$ that assigns to every event e the transition $\rho(e)$ that is executed at e. Thus, we require that (i) for all $e \in \mathcal{E}$, we have $action(\rho(e)) = \lambda(e)$, (ii) for all $(e, f) \in \rightarrow$, $target(\rho(e)) = source(\rho(f))$, (iii) for all $(e, f) \in \lhd$, $msg(\rho(e)) = msg(\rho(f))$, and (iv) for all $p \in \mathbb{P}$ and $e \in \mathcal{E}_p$ such that there is no $f \in \mathcal{E}$ with $f \rightarrow e$, we have $source(\rho(e)) = \ell_p^0$.

We write $L_{\mathsf{asy}}(\mathcal{S})$ to denote the set of MSCs M that admit a run of $\mathcal{S}$. Intuitively, $L_{\mathsf{asy}}(\mathcal{S})$ is the set of all asynchronous behaviors of $\mathcal{S}$.

C.2 Proof of Theorem 6.2

Theorem 6.2. *Let* $\mathsf{com} \in \{\mathsf{asy}, \mathsf{co}, \mathsf{p2p}, \mathsf{mb}, \mathsf{onen}, \mathsf{nn}, \mathsf{rsc}\}$ *and* $k \geq 1$ *be fixed. Then the following problem is decidable: given a system* $\mathcal{S}$ *and a MSO specification* φ*, decide whether* $L_{\mathsf{com}}(\mathcal{S}) \cap \mathsf{MSC}^{k\text{-stw}} \subseteq L(\varphi)$.

Proof. Let com, C, $\mathcal{S}$, and φ be fixed. We showed in Section 4 that there is a MSO formula φ_{com} that defines $\mathsf{MSC}_{\mathsf{com}}$. There is also a MSO formula $\varphi_{\mathcal{S}}$ such that $L_{\mathsf{asy}}(\mathcal{S}) = L(\varphi_{\mathcal{S}})$.[8] Putting everything together, we have

$$\begin{aligned} & L_{\mathsf{com}}(\mathcal{S}) \cap \mathsf{MSC}^{k\text{-stw}} \subseteq L(\varphi) \\ \iff\ & L_{\mathsf{asy}}(\mathcal{S}) \cap \mathsf{MSC}_{\mathsf{com}} \cap \mathsf{MSC}^{k\text{-stw}} \subseteq L(\varphi) \\ \iff\ & L(\varphi_{\mathcal{S}}) \cap L(\varphi_{\mathsf{com}}) \cap \mathsf{MSC}^{k\text{-stw}} \subseteq L(\varphi) \\ \iff\ & \mathsf{MSC}^{k\text{-stw}} \subseteq L(\varphi \vee \neg\varphi_{\mathsf{com}} \vee \neg\varphi_{\mathcal{S}}) . \end{aligned}$$

The latter is decidable by Courcelle's theorem [Courcelle 2010]. □

C.3 Proof of Theorem 6.4

In order to prove Theorem 6.4, we first need to introduce some concepts and give preliminary proofs.

Definition C.2 (Prefix). Let $M = (\mathcal{E}, \rightarrow, \lhd, \lambda) \in \mathsf{MSC}$ and consider $E \subseteq \mathcal{E}$ such that E is $\leq_{hb}$-*downward-closed*, i.e, for all $(e, f) \in \leq_{hb}$ such that $f \in E$, we also have $e \in E$. Then, the MSC $M' = (E, \rightarrow \cap (E \times E), \lhd \cap (E \times E), \lambda')$, where λ' is the restriction of $\mathcal{E}$ to E, is called a *prefix* of M.

If we consider a set E that is $\leq_{1\mathsf{n}}$-*downward-closed*, we call M' a onen-*prefix*. If the set E is $\sqsubset_{\mathsf{nn}}$-*downward-closed*, we call M' a nn-*prefix*. Note that every onen or nn-prefix is also a prefix, since $\leq_{hb} \subseteq \leq_{1\mathsf{n}}$ and $\leq_{hb} \subseteq \sqsubset_{\mathsf{nn}}$.

Note that the empty MSC is a prefix of M. We denote the set of prefixes of M by $Pref(M)$, whereas $Pref_{\mathsf{onen}}(M)$ and $Pref_{\mathsf{nn}}(M)$ are used for the FIFO $1{-}\mathsf{n}$ and the FIFO $\mathsf{n}{-}\mathsf{n}$ variants, respectively. This is extended to sets $L \subseteq \mathsf{MSC}$ as expected, letting $Pref(L) = \bigcup_{M \in L} Pref(M)$.

Proposition C.3. *For* $\mathsf{com} \in \{\mathsf{asy}, \mathsf{p2p}, \mathsf{co}, \mathsf{mb}\}$*, every prefix of a* com*-MSC is a* com*-MSC.*

Proof. For $\mathsf{com} = \mathsf{asy}$ it is true by definition. For $\mathsf{com} = \{\mathsf{p2p}, \mathsf{mb}\}$ it was already shown to be true in [Bollig et al. 2021a], so we just consider $\mathsf{com} = \mathsf{co}$. Let $M = (\mathcal{E}, \rightarrow, \lhd, \lambda) \in \mathsf{MSC}_{\mathsf{co}}$ and let

[8] The formula simply encodes the existence of a run of $\mathcal{S}$ on the MSC using a MSO variable X_l for each control state l, with the meaning that X_l is the set of events before which the local communicating automaton was in state l. See [Bollig and Gastin 2019, Theorem 3.4] for a detailed proof.

$M_0 = (\mathcal{E}_0, \to_0, \lhd_0, \lambda_0)$ be a prefix of M. By contradiction, suppose that M_0 is not a co-MSC. There must be two distinct $s, s' \in \mathcal{E}_0$ such that $\lambda(s) = Send(\_, q, \_)$, $\lambda(s') = Send(\_, q, \_)$, $s \leq_{hb}^{(M_0)} s'$ and either (i) $r' \to^+ r$, where r and r' are two receive events such that $s \lhd r$ and $s' \lhd r'$, or (ii) $s \in Unm(M_0)$ and $s' \in Matched(M_0)$. In both cases, M would also not be a co-a MSC, since $\mathcal{E}_0 \subseteq \mathcal{E}$, $\to_0 \subseteq \to$, and $\lhd_0 \subseteq \lhd$. This is a contradiction, thus M_0 has to be causally ordered. □

Note that this proposition is not true for the FIFO 1−n and the FIFO n−n communication models. Fig. 9 shows an example of nn-MSC with a prefix that is neither a nn-MSC nor a onen-MSC.

Proposition C.4. *Every* onen*-prefix of a* onen*-MSC is a* onen*-MSC.*

Proof. Let $M = (\mathcal{E}, \to, \lhd, \lambda) \in \mathsf{MSC}_{\mathsf{onen}}$ and let $M_0 = (\mathcal{E}_0, \to_0, \lhd_0, \lambda_0)$ be a onen-prefix of M, where $\mathcal{E}_0 \subseteq \mathcal{E}$. Firstly, the $\preceq_{\mathsf{1n}}$-downward-closeness of $\mathcal{E}_0$ guarantees that M_0 is still an MSC. We need to prove that it is a onen-MSC. By contradiction, suppose that M_0 is not a onen-MSC. Then, there are distinct $e, f \in \mathcal{E}_0$ such that $e \preceq_{\mathsf{1n}}^{(M_0)} f \preceq_{\mathsf{1n}}^{(M_0)} e$, where $\preceq_{\mathsf{1n}}^{(M_0)} = (\to_0 \cup \lhd_0 \cup \sqsubset_{\mathsf{1n}}^{(M_0)})^*$. As $\mathcal{E}_0 \subseteq \mathcal{E}$, we have that $\to_0 \subseteq \to$, $\lhd_0 \subseteq \lhd$, $\sqsubset_{\mathsf{1n}}^{(M_0)} \subseteq \sqsubset_{\mathsf{1n}}$. Clearly, $\preceq_{\mathsf{1n}}^{(M_0)} \subseteq \preceq_{\mathsf{1n}}$, so $e \preceq_{\mathsf{1n}} f \preceq_{\mathsf{1n}} e$. This implies that M is not a onen-MSC, because $\preceq_{\mathsf{1n}}$ is cyclic, which is a contradiction. Hence M_0 is a onen-MSC. □

Proposition C.5. *Every* nn*-prefix of a* nn*-MSC is a* nn*-MSC.*

Proof. Let $M = (\mathcal{E}, \to, \lhd, \lambda) \in \mathsf{MSC}_{\mathsf{nn}}$ and let $M_0 = (\mathcal{E}_0, \to_0, \lhd_0, \lambda_0)$ be a nn-prefix of M, where $\mathcal{E}_0 \subseteq \mathcal{E}$. Firstly, the $\sqsubset_{\mathsf{nn}}^{(M)}$-downward-closeness of $\mathcal{E}_0$ guarantees that M_0 is still an MSC. We need to prove that it is a nn-MSC. By contradiction, suppose that M_0 is not a nn-MSC. Then, there are distinct $e, f \in \mathcal{E}_0$ such that $e \sqsubset_{\mathsf{nn}}^{(M_0)} f \sqsubset_{\mathsf{nn}}^{(M_0)} e$. As $\mathcal{E}_0 \subseteq \mathcal{E}$, we have that $\to_0 \subseteq \to$, $\lhd_0 \subseteq \lhd$, $\preceq_{\mathsf{1n/mb}} \subseteq \preceq_{\mathsf{1n/mb}}$. Clearly, $\sqsubset_{\mathsf{nn}}^{(M_0)} \subseteq \sqsubset_{\mathsf{nn}}^{(M)}$, so $e \sqsubset_{\mathsf{nn}}^{(M)} f \sqsubset_{\mathsf{nn}}^{(M)} e$. This implies that M is not a nn-MSC, because $\sqsubset_{\mathsf{nn}}^{(M)}$ is cyclic, which is a contradiction. Hence M_0 is a nn-MSC. □

The next lemma is about the prefix closure of a communicating system and it follows from Proposition C.3.

Proposition C.6. *For all* com $\in \{\mathsf{asy}, \mathsf{p2p}, \mathsf{mb}, \mathsf{co}\}$*,* $L_{\mathsf{com}}(\mathcal{S})$ *is prefix-closed:* $Pref(L_{\mathsf{com}}(\mathcal{S})) \subseteq L_{\mathsf{com}}(\mathcal{S})$*.*

Similar results also hold for the FIFO 1−n and FIFO n−n communication models.

Proposition C.7. $L_{\mathsf{onen}}(\mathcal{S})$ *is* onen*-prefix-closed:* $Pref_{\mathsf{onen}}(L_{\mathsf{onen}}(\mathcal{S})) \subseteq L_{\mathsf{onen}}(\mathcal{S})$*.*

Proof. Given a system $\mathcal{S}$, we have that $L_{\mathsf{onen}}(\mathcal{S}) = L_{\mathsf{p2p}}(\mathcal{S}) \cap \mathsf{MSC}_{\mathsf{onen}}$. Note that, because of how we defined a onen-prefix, we have that $Pref_{\mathsf{onen}}(L_{\mathsf{onen}}(\mathcal{S})) = Pref(L_{\mathsf{onen}}(\mathcal{S})) \cap \mathsf{MSC}_{\mathsf{onen}}$. Moreover, $Pref(L_{\mathsf{onen}}(\mathcal{S})) \subseteq Pref(L_{\mathsf{p2p}}(\mathcal{S}))$, and $Pref(L_{\mathsf{onen}}(\mathcal{S})) \subseteq L_{\mathsf{p2p}}(\mathcal{S})$ for Proposition C.6. Putting everything together, $Pref_{\mathsf{onen}}(L_{\mathsf{onen}}(\mathcal{S})) \subseteq L_{\mathsf{p2p}}(\mathcal{S}) \cap \mathsf{MSC}_{\mathsf{onen}} = L_{\mathsf{onen}}(\mathcal{S})$. □

Proposition C.8. $L_{\mathsf{nn}}(\mathcal{S})$ *is* nn*-prefix-closed:*

$$Pref_{\mathsf{nn}}(L_{\mathsf{nn}}(\mathcal{S})) \subseteq L_{\mathsf{nn}}(\mathcal{S}).$$

Proof. Given a system $\mathcal{S}$, we have that $L_{\mathsf{nn}}(\mathcal{S}) = L_{\mathsf{p2p}}(\mathcal{S}) \cap \mathsf{MSC}_{\mathsf{nn}}$. Note that, because of how we defined a nn-prefix, we have that $Pref_{\mathsf{nn}}(L_{\mathsf{nn}}(\mathcal{S})) = Pref(L_{\mathsf{nn}}(\mathcal{S})) \cap \mathsf{MSC}_{\mathsf{nn}}$. Moreover, $Pref(L_{\mathsf{nn}}(\mathcal{S})) \subseteq Pref(L_{\mathsf{p2p}}(\mathcal{S}))$, and $Pref(L_{\mathsf{nn}}(\mathcal{S})) \subseteq L_{\mathsf{p2p}}(\mathcal{S})$ for Proposition C.6. Putting everything together, $Pref_{\mathsf{nn}}(L_{\mathsf{nn}}(\mathcal{S})) \subseteq L_{\mathsf{p2p}}(\mathcal{S}) \cap \mathsf{MSC}_{\mathsf{nn}} = L_{\mathsf{nn}}(\mathcal{S})$. □

In this last part we prove a series of statements to conclude that, when we have a STW-bounded class C, the synchronizability problem can be reduced to bounded model-checking, which we showed to be decidable in Theorem 6.2.

Proposition C.9. *Let $k \in \mathbb{N}$ and $C \subseteq \mathsf{MSC}^{k\text{-stw}}$. For all $M \in \mathsf{MSC} \setminus C$, we have $(Pref(M) \cap \mathsf{MSC}^{(k+2)\text{-stw}}) \setminus C \neq \emptyset$.*

Proof. Already proved in [Bollig et al. 2021a], but we adapt the proof to our setting. Let k and C be fixed, and let $M \in \mathsf{MSC} \setminus C$ be fixed. If the empty MSC is not in C, then we are done, since it is a valid prefix of M and it is in $\mathsf{MSC}^{(k+2)\text{-stw}} \setminus C$. Otherwise, let $M' \in Pref(M) \setminus C$ such that, for all $\leq_{hb}$-maximal events e of M', removing e (along with its adjacent edges) gives an MSC in C. In other words, M' is the "shortest" prefix of M that is not in C. We obtain such an MSC by successively removing $\leq_{hb}$-maximal events. Let e be a $\leq_{hb}$-maximal event of M', and let $M'' = M' \setminus \{e\}$. Since M' was taken minimal in terms of number of events, $M'' \in C$. So Eve has a winning strategy with $k+1$ colors for M''. Let us design a winning strategy with $k+3$ colors for Eve for M', which will show the claim.

Observe that the event e occurs at the end of the timeline of a process (say p), and it is part of at most two edges:

- one with the previous p-event (if any)
- one with the corresponding send event (if e is a receive event)

Let e_1, e_2 be the two neighbours of e. The strategy of Eve is the following: in the first round, mark e, e_1, e_2, then erase the edges (e_1, e) and (e_2, e), then split the remaining graph in two parts: M'' on the one side, and the single node graph $\{e\}$ on the other side. Then Eve applies its winning strategy for M'', except that initially the two events e_1, e_2 are marked (so she may need up to $k+3$ colors). □

We have similar results also for the FIFO $1-\mathsf{n}$ and FIFO $\mathsf{n}-\mathsf{n}$ communication models.

Proposition C.10. *Let $k \in \mathbb{N}$ and $C \subseteq \mathsf{MSC}^{k\text{-stw}}$. For all $M \in \mathsf{MSC}_{\mathsf{onen}} \setminus C$, we have*

$$(Pref_{\mathsf{onen}}(M) \cap \mathsf{MSC}^{(k+2)\text{-stw}}) \setminus C \neq \emptyset.$$

Proof. Let k and C be fixed, and let $M \in \mathsf{MSC}_{\mathsf{onen}} \setminus C$ be fixed. If the empty MSC is not in C, then we are done, since it is a valid onen-prefix of M and it is in $\mathsf{MSC}^{(k+2)\text{-stw}} \setminus C$. Otherwise, let $M' \in Pref_{\mathsf{onen}}(M) \setminus C$ such that, for all $\leq_{1\mathsf{n}}$-maximal events e of M', removing e (along with its adjacent edges) gives an MSC in C. In other words, M' is the "shortest" prefix of M that is not in C. We obtain such an MSC by successively removing $\leq_{1\mathsf{n}}$-maximal events. Let e be $\leq_{1\mathsf{n}}^{(M')}$-maximal and let $M'' = M' \setminus \{e\}$. Since M' was taken minimal in terms of number of events, $M'' \in C$. The proof proceeds exactly as the proof of Proposition C.9. □

Proposition C.11. *Let $k \in \mathbb{N}$ and $C \subseteq \mathsf{MSC}^{k\text{-stw}}$. For all $M \in \mathsf{MSC}_{\mathsf{nn}} \setminus C$, we have*

$$(Pref_{\mathsf{nn}}(M) \cap \mathsf{MSC}^{(k+2)\text{-stw}}) \setminus C \neq \emptyset.$$

Proof. Let k and C be fixed, and let $M \in \mathsf{MSC}_{\mathsf{nn}} \setminus C$. If the empty MSC is not in C, then we are done, since it is a valid nn-prefix of M and it is in $\mathsf{MSC}^{(k+2)\text{-stw}} \setminus C$. Otherwise, let $M' \in Pref_{\mathsf{nn}}(M) \setminus C$ such that, for all $\sqsubset_{\mathsf{nn}}^{(M)}$-maximal events e of M', removing e (along with its adjacent edges) gives an MSC in C. In other words, M' is the "shortest" prefix of M that is not in C. We obtain such an MSC by successively removing $\sqsubset_{\mathsf{nn}}^{(M)}$-maximal events. Let e be $\sqsubset_{\mathsf{nn}}^{(M')}$-maximal and let $M'' = M' \setminus \{e\}$. Since M' was taken minimal in terms of number of events, $M'' \in C$. The proof proceeds exactly as the proof of Proposition C.9. □

The following proposition is the last ingredient that we need to prove Theorem 6.4.

Proposition C.12. *Let $\mathcal{S}$ be a communicating system,* $\mathsf{com} \in \{\mathsf{asy}, \mathsf{p2p}, \mathsf{co}, \mathsf{mb}, \mathsf{onen}, \mathsf{nn}, \mathsf{rsc}\}$, *$k \in \mathbb{N}$, and $C \subseteq \mathsf{MSC}^{k\text{-stw}}$. Then, $L_{\mathsf{com}}(\mathcal{S}) \subseteq C$ iff $L_{\mathsf{com}}(\mathcal{S}) \cap \mathsf{MSC}^{(k+2)\text{-stw}} \subseteq C$.*

Proof. For com ∈ {asy, p2p, co, mb}, the proposition follows from Proposition C.9. For com ∈ {onen, nn}, it follows from Proposition C.10 and Proposition C.11, respectively. □

Theorem 6.4. *For any* com ∈ {asy, p2p, co, mb, onen, nn} *and for all class of MSCs* $\mathcal{C}$*, if* $\mathcal{C}$ *is STW-bounded and MSO-definable, then the* $(\mathsf{com}, \mathcal{C})$*-synchronizability problem is decidable.*

Proof. According to Proposition C.12, we have $L_{\mathsf{com}}(\mathcal{S}) \subseteq \mathcal{C}$ iff $L_{\mathsf{com}}(\mathcal{S}) \cap \mathsf{MSC}^{(k+2)\text{-stw}} \subseteq \mathcal{C}$. The latter is decidable according to Theorem 6.2. □

C.4 Proof of Proposition 6.8

Proposition 6.8. *The following problem is undecidable: given a communicating system* $\mathcal{S}$*, is every MSC in* $L_{\mathsf{co}}(\mathcal{S})$ *weakly synchronous?*

The proof is very similar to the one of [Bollig et al. 2021b, Theorem 20] for the p2p case. We do the same reduction from the Post correspondence problem. The original proof considered a p2p system $\mathcal{S}$ with four machines (P1, P2, V1, V2), where we have unidirectional communication channels from provers (P1 and P2) to verifiers (V1 and V2). In particular notice that all the possible behaviors of $\mathcal{S}$ are causally ordered, i.e. $L_{\mathsf{p2p}}(\mathcal{S}) \subseteq \mathsf{MSC}_{\mathsf{co}}$; according to how we built our system $\mathcal{S}$, it is impossible to have a pair of causally-related send events of P1 and P2[9], which implies that causal ordering is already ensured by any possible p2p behavior of $\mathcal{S}$. The rest of the proof is identical to the p2p case.

[9] There is no channel between P1 and P2, and we only have unidirectional communication channels from provers to verifiers; it is impossible to have a causal path between two send events of P1 and P2.