Examples and Proofs: Partial Order Semantics of Types of Nets

Robert Lorenz[1], Gabriel Juhás[2], and Sebastian Mauser[3]

[1] Department of Computer Science, University of Augsburg, Germany
robert.lorenz@informatik.uni-augsburg.de
[2] Faculty of Electrical Engineering and Information Technology
Slovak University of Technology, Bratislava, Slovakia
gabriel.juhas@stuba.sk
[3] Department of Applied Computer Science
Catholic University of Eichstätt-Ingolstadt, Germany
sebastian.mauser@ku-eichstaett.de

Examples

Example 1

• The *type of p/t-nets* is given by $\tau_{pt} = (\mathbb{N}, \mathbb{N} \times \mathbb{N}, \tau_{pt})$, where $n \xrightarrow{(i,j)} n'$ if and only if $n \geq i$ and $n' = n - i + j$. The abelian monoid of local events has the operation of componentwise addition and the identity element $(0, 0)$. The weight function W attaches a tuple of natural numbers to the arcs $(p, t) \in P \times T$ of a net of type τ_{pt}. The first component of the tuple is interpreted as the number of tokens consumed from t in p and the second component as the number of tokens produced from t in p. Then Definition 3 coincides with the standard occurrence rule of p/t-nets.
• The *type of elementary nets* is given by $\tau_{en} = (\{0, 1\}, \{nop, in, out, failure\}, \tau_{en})$, where $\tau_{en} = \{(0, nop, 0), (1, nop, 1), (0, out, 1), (1, in, 0)\}$. The abelian monoid of local events has the identity element nop and the operation $+$ given by $x+y = failure$ for $x, y \neq nop$. The weight function W attaches a label from $\{nop, in, out, failure\}$ to the arcs $(p, t) \in P \times T$ of a net of type τ_{en}. The local event nop is interpreted as no arc, in as an arc ingoing to t (from p), out as an arc outgoing from t (to p) and $failure$ as not allowed. Then Definition 3 coincides with the occurrence rule of elementary nets.
• The *type of pti-nets* is given by $\tau_{pti} = (\mathbb{N}, \mathbb{N} \times \mathbb{N} \times \mathbb{N}_\omega, \tau_{pti})$ for the a-posteriori semantics and $\tau_{\overleftarrow{pti}} = (\mathbb{N}, \mathbb{N} \times \mathbb{N} \times \mathbb{N}_\omega, \tau_{\overleftarrow{pti}})$ for the a-priori semantics. The abelian monoid of local events has the identity element $(0, 0, \omega)$ and the operation $+$ given by componentwise addition on the first two components and the minimum function on the third component, i.e. $(x, y, z) + (x', y', z') = (x + x', y + y', min(z, z'))$. For the a-posteriori semantics $(n, (i, j, k), n') \in \tau_{pti}$ if and only if $n \geq i$, $n + j \leq k$ and $n' = n - i + j$. For the a-priori semantics $(n, (i, j, k), n') \in \tau_{\overleftarrow{pti}}$ if and only if $n \geq i$, $n \leq k$ and $n' = n - i + j$. The weight function W attaches two natural numbers and an element in $\mathbb{N}_\omega$ to the arcs $(p, t) \in P \times T$ of a net of type τ_{pti} resp. $\tau_{\overleftarrow{pti}}$. The first two natural numbers are interpreted as for p/t-nets and the element in $\mathbb{N}_\omega$ as an inhibitor weight attached to an inhibitor arc ingoing to t (from p). Then Definition 3

coincides with the occurrence rule of pti-nets equipped with the a-posteriori resp. the a-priori semantics.

Example 2

Consider the type of nets $(\{s,s'\},\{0,1,2\},\{s \xrightarrow{1} s, s \xrightarrow{2} s', s' \xrightarrow{1} s, s \xrightarrow{0} s, s' \xrightarrow{0} s'\})$, where 0 is the identity element, $1+1=2$, $1+2=0$ and $2+2=1$, and the marked net $N=(P,T,W,m_0)$ of this type given by $P=\{p\}$, $T=\{t_1,t_2\}$, $m_0(p)=s$, $W(p,t_1)=1$, $W(p,t_2)=2$. Then the LPO $\mathrm{lpo}=(\{v_1,v_2\},\emptyset,l)$, $l(v_i)=t_i$ for $i=1,2$, is enabled in N. The step sequence t_1t_2 of lpo yields the final marking $m(p)=s'$, while the step sequence t_2t_1 of lpo yields the final marking $m(p)=s\neq s'$.

Example 3

Consider the case that an LPO $\mathrm{lpo}=(V,<,l)$ enabled w.r.t. a net of some type has two different final markings, where one enables a transition t and one does not enable t. Then the second statement of Lemma 3 would imply that an LPO $\mathrm{lpo}'=(V\cup\{v\},<\cup(V\times\{v\}),l')$, $l'|_V=l, l'(v)=t$, is enabled, although it is not enabled. An example for this is the enabled LPO from Example 2, where the final marking given by $m(p)=s$ enables t_2, but the final marking given by $m(p)=s'$ does not enable t_2.

Example 4

The type of nets $(\{s,s',s''\},\{0,a,b,c\},\{s \xrightarrow{a} s', s \xrightarrow{b} s'', s' \xrightarrow{b} s'', s'' \xrightarrow{a} s', s \xrightarrow{0} s, s' \xrightarrow{0} s', s'' \xrightarrow{0} s''\})$, where 0 is the identity element and $a+a=a+b=a+c=b+b=b+c=c+c=c$, satisfies the WISP. Given the marked net $N=(P,T,W,m_0)$ of this type given by $P=\{p\}$, $T=\{t_a,t_b\}$, $m_0(p)=s$, $W(p,t_a)=a$, $W(p,t_b)=b$, the LPOs $\mathrm{lpo}=(\{v_a,v_b\},\{(v_a,v_b)\},l)$, $l(v_a)=t_a$, $l(v_b)=t_b$ and $\mathrm{lpo}'=(\{v_a',v_b'\},\{(v_b',v_a')\},l')$, $l'(v_a')=t_a$, $l'(v_b')=t_b$ are enabled in N. Both LPOs comprise of one transition occurrence of t_a and one of t_b, but the final marking of lpo is given by $m(p)=s''$, while the final marking of lpo' is given by $m(p)=s'$.

Example 5

Consider a net $N=(P,T,W,m_0)$ fulfilling the WISP and not fulfilling the ISP, i.e. there is x_1,x_2 and a reachable marking m such that $m \xrightarrow{x_1+x_2} m'$ but not $m \xrightarrow{x_1x_2}$. Let $lpo'=(V',<',l')$ be an LPO in step form such that $m_0 \xrightarrow{\sigma_{\mathrm{lpo}'}} m$. Then the second statement of Lemma 8 would imply that an LPO $\mathrm{lpo}=(V'\cup V,<'\cup(V'\times V),l)$, $l|_{V'}=l'$, $|V|_l=x_1+x_2$, is enabled, although it is not enabled. An example for this situation is the type of nets $(\{s\},\{0,1,2\},\{s \xrightarrow{0} s\})$, where 0 is the identity element, $1+1=2$, $1+2=0$ and $2+2=1$ (monoid as in Example 2), and the the marked net $N=(P,T,W,m_0)$ of this type given by $P=\{p\}$, $T=\{t_1,t_2\}$, $m_0(p)=s$,

$W(p,t_1) = 1$, $W(p,t_2) = 2$. The LPO $\mathrm{lpo} = (\{v_1, v_2\}, \emptyset, l)$, $l(v_i) = t_i$ for $i = 1,2$ is not enabled in N, because the step sequences $t_1 t_2$ and $t_2 t_1$ are not enabled in m_0, although $t_1 + t_2$ is enabled in m_0 (here $m = m_0$ in the above notation, i.e. $V' = \emptyset$).

Example 6

• The type of nets in Example 2 does not satisfy the WISP, since $s \xrightarrow{(1)(2)} s'$ and $s \xrightarrow{1+2} s$.
• The type of nets in Example 5 satisfies the WISP but not the ISP, since $s \xrightarrow{1+2} s$ but not $s \xrightarrow{(1)(2)}$.
• The type of nets in Example 4 satisfies the WISP but not the PIP, since $s \xrightarrow{(a)(b)} s''$ and $s \xrightarrow{(b)(a)} s'$.
• Combining the previous two types to the type $(\{s, s', s''\}, \{0, 1, 2, a, b, c\}, \{s \xrightarrow{a} s', s \xrightarrow{b} s'', s' \xrightarrow{b} s'', s'' \xrightarrow{a} s', s \xrightarrow{0} s, s' \xrightarrow{0} s', s'' \xrightarrow{0} s''\})$, where 0 is the identity element, $1+1 = 2$, $1+2 = 0$, $2+2 = 1$ and $x + y = c$ for $x \in \{1, 2, a, b, c\}, y \in \{a, b, c\}$, yields an example of a type of nets satisfying the WISP, but neither the ISP nor the PIP.
• The type of nets not satisfying the PIP from Example 4, satisfies the ISP.
• The type of nets not satisfying the ISP from Example 5, satisfies the PIP. Also the type $\tau_{\overleftarrow{pti}}$ in Example 1 satisfies the PIP but not the ISP.
• The types τ_{pt}, τ_{en} and τ_{pti} in Example 1 satisfy the PIP and the ISP.

Example 7

The types of nets τ_{pt}, τ_{pti} and $\tau_{\overleftarrow{pti}}$ from Example 1 can be equipped with appropriate flow maps yielding flow types of nets (note that in each case the set of local states is the free abelian monoid $(\mathbb{N}, +, 0)$):
• $f_{pt}(i,j) = (i,j)$,
• $f_{pti}(i,j,k) = (i,j)$,
• $f_{\overleftarrow{pti}}(i,j,k) = (i,j)$.

Also τ_{en} from Example 1 can be interpreted as a flow type of nets by setting $LS = \mathbb{N} \supset \{0,1\}$ and $LE = \mathbb{N} \times \mathbb{N}$, where $(0,0)$, $(1,0)$ resp. $(0,1)$ correspond to nop, in resp. out and all other local events correspond to $failure$. The respective flow map is given by $f_{en}(i,j) = (i,j)$. Similar net classes such as nets with read arcs, capacities, etc. are also covered by flow types of nets.

Example 8

The characteristic of a free abelian monoid is that each element can uniquely up to the ordering be represented by the elements of a subset of generator elements. The abelian monoid $M = \mathbb{N} \times \mathbb{N} \setminus \{(0,1),(1,0)\}$ with the operation of componentwise addition does not have such property. The elements $(1,1)$, $(2,0)$ and $(0,2)$ cannot be represented as a sum of other elements of M, and the element $(2,2)$ can be represented as $2 \cdot (1,1)$ and $(2,0) + (0,2)$. Define $\tau = (M, M \times M, \tau)$ with $(n, (i,j), n') \in \tau$ if and only if $n \geq i$ and $n' = n - i + j$, and define $f(i,j) = (i,j)$. Consider the marked net of flow type τ defined by $T = \{t_1, t_2, t_3, t_4\}$, $P = \{p\}$, $W(p, t_1) = ((0, 0), (1, 1))$, $W(p, t_2) = ((0, 0), (1, 1))$, $W(p, t_3) = ((2, 0), (0, 0))$, $W(p, t_4) = ((0, 2), (0, 0))$, $m_0(p) = (0, 0)$. Then the LPO $\mathrm{lpo} = (\{v_1, v_2, v_3, v_4\}, \{(v_1, v_3), (v_1, v_4), (v_2, v_3), (v_2, v_4)\}, l)$ with $l(v_i) = t_i$ is enabled in this net. But it is not possible to assign appropriate token flows to the arcs of the LPO such that the token flow property is fulfilled. This is because v_1 and v_2 each produce the local state $(1, 1)$ yielding together $(2, 2)$, which is enough to enable v_3 and v_4 in one step consuming $(2, 0)$ resp. $(0, 2)$ ($(2, 0) + (0, 2) = (2, 2)$). But v_3 then consumes $(1, 0)$ from v_1 as well as from v_2. Since $(1, 0) \notin M$, there is no valid distribution of local states to the arcs of lpo such that the token flow property is satisfied. This example not only shows the need for generator local states, but also illustrates the essential concept of token flows that, in order to check the token flow property for an LPO, an appropriate distribution of the local states produced resp. consumed by an event to the outgoing resp. ingoing arcs of the event has to be found.

Example 9

Canonical blocking functions for the two types of pti-nets from Example 1 are given by $b_{pti}(n, (i, j, k)) = 1 \iff n + j \leq k$ resp. $b_{\overleftarrow{pti}}(n, (i, j, k)) = 1 \iff n \leq k$ in the case of τ_{pti} resp. $\tau_{\overleftarrow{pti}}$.

Proofs

Proof of Lemma 1

Let lpo be enabled, then lpo$'$ is enabled, since it is a prefix. Let $\mathrm{lpo}'' = (V', <'', l')$ be a step sequentialization of lpo$'$, then there is a step sequentialization $\mathrm{lpo}_s = (V, <_s, l)$ of lpo fulfilling $v' <_s c <_s v$ and $c\,\mathrm{co}_{<_s} c'$ for all $v' \in V', c, c' \in C, v \in V \setminus (V' \cup C)$ as well as $<'' = <_s |_{V' \times V'}$. The step sequence σ_{lpo_s} is enabled showing that $|C|_l$ is enabled in the final marking of lpo$'$ defined by lpo$''$.

If lpo is not enabled, either every proper prefix of lpo is enabled or not. In the second case there is a non-empty co-set of lpo having a prefix which is not enabled. In the first case consider a step sequentialization lpo_s of lpo, such that $\sigma_{lpo_s} = x_1 \dots x_n$ is not enabled (where x_n not empty). Since every proper prefix of lpo is enabled, $x_1 \dots x_{n-1}$ is enabled and x_n is not enabled in the follower marking of $x_1 \dots x_{n-1}$. Define C as the set of maximal events of lpo_s (corresponding to the step x_n) and consider the prefix lpo$'$ of lpo given by the set of events $V \setminus C$. Then lpo$'$ is enabled (since it is a proper prefix of lpo) and $|C|_l$ is not enabled in the final marking of lpo$'$ given by the step sequence $x_1 \dots x_{n-1}$ of lpo$'$.

Proof of Lemma 2

Given two step sequentializations lpo$'$ and lpo$''$ of lpo with $m_0 \xrightarrow{\sigma_{\mathrm{lpo}'}} m'$ and $m_0 \xrightarrow{\sigma_{\mathrm{lpo}''}} m''$, we have to show that $m' = m''$.

We consider a fixed order of the events $V = \{v_1, \dots, v_n\}$ such that $v_i < v_j$ implies $i < j$. We show that both m' and m'' coincide with the marking m''' given by $m_0 \xrightarrow{l(v_1)\dots l(v_n)} m'''$. For this we iteratively transform $\sigma_{\mathrm{lpo}'}$ to $l(v_1) \dots l(v_n)$. Each iteration yields a step sequence σ_i of lpo which is enabled (since lpo is enabled) and fulfills $m_0 \xrightarrow{\sigma_i} m'$ (since WISP holds).

First, we transform $\sigma_{\mathrm{lpo}'}$ to a linear sequence of lpo. Let $\sigma_{\mathrm{lpo}'} = x_1 \dots x_k$. If x_i (for some i) is not a single event, then x_i can be decomposed into two non-empty steps x', x'', i.e. $x_i = x' + x''$. Let $m_0 \xrightarrow{x_1 \dots x_{i-1}} m_{i-1}$ and $m_{i-1} \xrightarrow{x_i} m_i$. According to the WISP, also $m_{i-1} \xrightarrow{x'x''} m_i$ (the enabledness is ensured by the enabledness of lpo). Steps are decomposed in this way until the resulting step sequence is a linear sequence $t_1 \dots t_n$ of lpo with $m_0 \xrightarrow{t_1 \dots t_n} m'$.

Let $\mathrm{lpo}_{lin} = (V, <_{lin}, l)$ be a linearization of lpo such that $\sigma_{\mathrm{lpo}_{lin}} = t_1 \dots t_n$. In lpo_{lin} the nodes $\{v_1, \dots, v_n\}$ are totally ordered respecting the partial ordering given by $<$, but possibly in another order than given by the indices $1, \dots, n$. That means it may be that $v_i <_{lin} v_j$ and $i > j$, but only in the case $v_i \not< v_j$ and $v_j \not< v_i$, i.e. $v_i\,\mathrm{co}_< v_j$. In this situation we switch the positions of the events v_i and v_j in lpo_{lin} to get in several steps the linearization $v_1 \to v_2 \to \dots \to v_n$ of lpo. The different positions of nodes in the two linearizations $v_1 \to v_2 \to \dots \to v_n$ and lpo_{lin} can be related by a permutation π such that $\pi(i)$ is the position of the i-th node of lpo_{lin} in $v_1 \to v_2 \to \dots \to v_n$, i.e. the index of the i-th node of lpo_{lin} ($\pi^{-1}(i)$ is the position of v_i in lpo_{lin}).

Let π be the permutation on $\{1, \dots, n\}$ such that $v_{\pi(i)} <_{lin} v_{\pi(j)} \iff i < j$. If $\pi(i) = i$ for all $i \in \{1, \dots, n\}$, we are finished. Otherwise, consider that i is the first index satisfying $\pi(i) \neq i$ (obviously $\pi^{-1}(i) > i$). The idea is to "bubble-sort" the events $v_{\pi^{-1}(i)}$ from the position $\pi^{-1}(i)$ backwards to the position i, and to repeat this procedure until there is no such i. Since i was the first index with the property $\pi(i) \neq i$, we have $j = \pi(\pi^{-1}(i) - 1) > i = \pi(\pi^{-1}(i))$ implying $v_j \not< v_i$. Since $v_j <_{lin} v_i$ and lpo_{lin} is a linearization of lpo, we have $v_i \not< v_j$. It follows $v_i \operatorname{co}_< v_j$. Thus, removing $v_j <_{lin} v_i$ from $<_{lin}$ gives a step sequentialization of lpo. The associated step sequence $t_1 \dots t_{\pi^{-1}(i)-2}(t_{\pi^{-1}(i)-1} + t_{\pi^{-1}(i)}) \dots t_n$ is enabled. Moreover, by the WISP, $m_0 \overset{t_1 \dots t_{\pi^{-1}(i)-2}}{\longrightarrow} \tilde{m}$, $\tilde{m} \overset{t_{\pi^{-1}(i)-1} t_{\pi^{-1}(i)}}{\longrightarrow} \tilde{m}'$ and $\tilde{m} \overset{t_{\pi^{-1}(i)-1} + t_{\pi^{-1}(i)}}{\longrightarrow} \tilde{m}'$. Thus the final marking m' is preserved by the associated step sequence $t_1 \dots t_{\pi^{-1}(i)-2}(t_{\pi^{-1}(i)-1} + t_{\pi^{-1}(i)}) \dots t_n$. We can further introduce $v_i <_{lin} v_j$ to $<_{lin}$ yielding a linearization of lpo having the associated linear sequence $t_1 \dots t_{\pi^{-1}(i)-2} t_{\pi^{-1}(i)} t_{\pi^{-1}(i)-1} \dots t_n$, where by the WISP $\tilde{m} \overset{t_{\pi^{-1}(i)} t_{\pi^{-1}(i)-1}}{\longrightarrow} \tilde{m}'$. Thus, we have "bubble-sorted" v_i to one position backward preserving the final marking m' of the associated linear sequence of lpo. Repeating this procedure sorts each v_i to position i. This shows $m_0 \overset{l(v_1) \dots l(v_n)}{\longrightarrow} m'$.

Since the same procedure can be applied to lpo'', we also get $m_0 \overset{l(v_1) \dots l(v_n)}{\longrightarrow} m''$ proving $m' = m''$.

Proof of Lemma 3

The first part follows directly from Lemma 1. For the second part, let lpo be not enabled in N. Consider a prefix $\mathrm{lpo}_p = (V_p, <_p, l_p)$ of lpo (lpo_p might equal lpo) which is also not enabled and minimal with this property, i.e. every proper prefix of lpo_p is enabled. By Lemma 1 there is a non-empty co-set C of lpo_p and a prefix $\mathrm{lpo}' = (V', <', l')$ of C (within lpo_p) such that either lpo' is not enabled or $|C|_l$ is not enabled in some final marking of lpo'. Since lpo' is a proper prefix of lpo_p, the second case holds. By Lemma 2 the final marking of lpo' is unique, i.e. each step sequence σ of lpo' is enabled, but $|C|_l$ is not enabled in the follower marking of σ, showing the statement.

Proof of Lemma 4

Let (LS, LE, τ) fulfill the WISP and let N be of this type. Let m, m', m'' be reachable markings and x_1, x_2 be steps of transitions such that $m \overset{x_1+x_2}{\longrightarrow} m'$ and $m \overset{x_1 x_2}{\longrightarrow} m''$. By the occurrence rule, for each place p there holds $m(p) \overset{\sum_{t \in T}(x_1+x_2)(t)W(p,t)}{\longrightarrow} m'(p)$ and $m(p) \overset{\sum_{t \in T} x_1(t)W(p,t)}{\longrightarrow} m_{mid}(p) \overset{\sum_{t \in T} x_2(t)W(p,t)}{\longrightarrow} m''(p)$. Denoting $e_i = \sum_{t \in T} x_i(t)W(p,t)$ for $i = 1, 2$, $s = m(p)$, $s' = m'(p)$ and $s'' = m''(p)$ we get $m'(p) = s' = s'' = m''(p)$.

Let (LS, LE, τ) not satisfy the WISP. That means, there are local states s, s', s'' and local events e_1, e_2 such that $s \overset{e_1+e_2}{\longrightarrow} s'$, $s \overset{e_1 e_2}{\longrightarrow} s''$ and $s' \neq s''$. We construct a net $N = (P, T, W, m_0)$ of type τ not satisfying the WISP by $P = \{p\}$, $T = \{t_1, t_2\}$, $m_0(p) = s$, $W(p, t_1) = e_1$, $W(p, t_2) = e_2$.

Proof of Lemma 5

Let (LS, LE, τ) fulfill the PIP and let N be of this type. Let m, m', m'' be reachable markings and $x_1, \ldots, x_n$ and $x'_1, \ldots, x'_m$ be steps of transitions such that $m \xrightarrow{x_1 \ldots x_n} m'$, $m \xrightarrow{x'_1 \ldots x'_m} m''$ and $x_1 + \ldots + x_n = x'_1 + \ldots + x'_m$. Denote $e_i = \sum_{t \in T} x_i(t) W(p,t)$ and $e'_i = \sum_{t \in T} x'_i(t) W(p,t)$, $s = m(p)$, $s' = m'(p)$ and $s'' = m''(p)$. By the occurrence rule, for each place p there holds $s \xrightarrow{e_1 \ldots e_n} s'$ and $s \xrightarrow{e'_1 \ldots e'_m} s''$. This gives $m'(p) = s' = s'' = m''(p)$, since τ satisfies PIP.

Let (LS, LE, τ) not satisfy the PIP. That means there are local states s, s', s'', a multi-set of local events u and two partitions $u = u_1 + \ldots + u_n = u'_1 + \ldots + u'_m$ of u with $s \xrightarrow{e_1 \ldots e_n} s'$, $s \xrightarrow{e'_1 \ldots e'_m} s''$ and $s' \neq s''$, where $e_i = \sum_{e \in u_i} u_i(e) e$ and $e'_i = \sum_{e \in u'_i} u'_i(e) e$. We construct a marked net $N = (P, T, W, m_0)$ of type τ not satisfying the PIP by $P = \{p\}$, $T = LE$, $m_0(p) = s$, $W(p, e) = e$.

Proof of Lemma 6

Since lpo is enabled, there is a an enabled step sequence $x_1 \ldots x_n$ of lpo. Since lpo' is enabled, there is a an enabled step sequence $x'_1 \ldots x'_m$ of lpo'. Since $|V|_l = |V'|_{l'}$, we have $x_1 + \ldots + x_n = x'_1 + \ldots + x'_m$. From PIP we get, that both step sequences define the same final marking.

Proof of Lemma 7

Let (LS, LE, τ) fulfill the ISP and let N be of this type. Let m, m' be reachable markings and x_1, x_2 be steps of transitions such that $m \xrightarrow{x_1 + x_2} m'$. Then, by the occurrence rule, for each place p there holds $m(p) \xrightarrow{\sum_{t \in T} (x_1 + x_2)(t) W(p,t)} m'(p)$. Denoting $e_i = \sum_{t \in T} x_i(t) W(p,t)$ for $i = 1, 2$, $s = m(p)$ and $s' = m'(p)$ we get that $s \xrightarrow{e_1 e_2} s'$ by the ISP. This gives $m \xrightarrow{x_1 x_2} m'$.

Let (LS, LE, τ) not satisfy the ISP. That means there are local states s, s' and local events e_1, e_2 such that $s \xrightarrow{e_1 + e_2} s'$ but not $s \xrightarrow{e_1 e_2} s'$. We construct a net $N = (P, T, W, m_0)$ of type τ not satisfying the ISP by $P = \{p\}$, $T = \{t_1, t_2\}$, $m_0(p) = s$, $W(p, t_1) = e_1$, $W(p, t_2) = e_2$.

Proof of Lemma 8

Since every cut is a co-set, the first part is shown by Lemma 3. For the second part, let lpo be not enabled in N. Consider a prefix $\mathrm{lpo}_p = (V_p, <_p, l_p)$ of lpo which is also not enabled and minimal with this property, i.e. every proper prefix of lpo_p is enabled. By Lemma 1 there is a non-empty co-set C' of lpo_p and a prefix $\mathrm{lpo}' = (V', <', l')$ of C' (within lpo_p) such that either lpo' is not enabled or $|C'|_l$ is not enabled in the unique (Lemma 2) final marking of lpo'. Since lpo' is a proper prefix of lpo_p, the second case holds. We now extend C' to a cut C containing additionally maximal elements of lpo' and minimal elements of lpo w.r.t. $V \setminus V'$: Denote $D = \{v \in V \setminus V' \mid v' < v \Longrightarrow$

$v' \in V'\}$ and $E = \{v \in V' \mid v \operatorname{co}_< D \wedge (v < v' \Longrightarrow v' \notin V')\}$. The set $C = D \cup E$ is a cut of lpo fulfilling $C' \subseteq C$ (since $C' \subseteq D$). The prefix $\mathrm{lpo}'' = (V'', <'', l'')$ of C fulfills $V'' \subseteq V' = V'' \cup E$, and therefore is enabled by the minimality property of lpo_p. Assume that $|C|_l$ is enabled in the unique final marking of lpo''. By the ISP also the step sequence $|E|_l|D|_l$ is enabled in the final marking of lpo''. The marking reached through firing $|E|_l$ in the final marking of lpo'' is the final marking of lpo'. Therefore, since $C' \subseteq D$, again by the ISP $|C'|_l$ is enabled in the final marking of lpo', a contradiction. Thus, $|C|_l$ is not enabled in the final marking of the enabled prefix lpo'' of C, i.e. each step sequence σ of lpo'' is enabled, but $|C|_l$ is not enabled in the follower marking of σ (given by the final marking of lpo''), showing the statement.

Proof of Lemma 9

Let s, s', s'' be local states, u be a multi-set of local events and $u = u_1 + \ldots + u_n = u'_1 + \ldots + u'_m$ be two partitions of u. Denote $e_i = \sum_{e \in u_i} u_i(e)e$ and $e'_i = \sum_{e \in u'_i} u'_i(e)e$ and let $s \xrightarrow{e_1 \ldots e_n} s'$ and $s \xrightarrow{e'_1 \ldots e'_m} s''$. Denote $s = s_0 \xrightarrow{e_1} s_1 \xrightarrow{e_2} \ldots \xrightarrow{e_n} s_n = s'$. Then, since f_1 and f_2 are monoid morphisms, $s_i = s_{i-1} + f_2(e_i) - f_1(e_i) = s_{i-1} + f_2(\sum_{e \in u_i} u_i(e)e) - f_1(\sum_{e \in u_i} u_i(e)e) = s_{i-1} + \sum_{e \in u_i} u_i(e) f_2(e) - \sum_{e \in u_i} u_i(e) f_1(e)$ for each i. Putting all these equations together, we get $s' = s + \sum_{i=1}^{n}(\sum_{e \in u_i} u_i(e) f_2(e)) - \sum_{i=1}^{n}(\sum_{e \in u_i} u_i(e) f_1(e)) = s + \sum_{e \in u} u(e) f_2(e) - \sum_{e \in u} u(e) f_1(e)$. Analogously there holds $s'' = s + \sum_{e \in u} u(e) f_2(e) - \sum_{e \in u} u(e) f_1(e)$.

Proof of Lemma 10

Each step sequence $\sigma = y_1 \ldots y_n$ corresponding to a step sequentialization of lpo is enabled in m_0, i.e. $m_0 \xrightarrow{y_1} m_1 \xrightarrow{y_2} \ldots \xrightarrow{y_n} m_n$, where $(m_{i-1}(p), \sum_{t \in T} y_i(t) \cdot W(p,t), m_i(p)) \in \tau$. By the flow type of nets definition, we have $m_i(p) = m_{i-1}(p) - f_1(\sum_{t \in T} y_i(t) \cdot W(p,t)) + f_2(\sum_{t \in T} y_i(t) \cdot W(p,t)) = m_{i-1}(p) - \sum_{t \in T} y_i(t) \cdot f_1(W(p,t)) + \sum_{t \in T} y_i(t) \cdot f_2(W(p,t))$. For the final marking m_n of lpo we compute $m_n(p) = m_0(p) + \sum_{i=1}^{n}(\sum_{t \in T} y_i(t) \cdot f_2(W(p,t))) - \sum_{i=1}^{n}(\sum_{t \in T} y_i(t) \cdot f_1(W(p,t)))$ $= m_0(p) + \sum_{t \in T} |V|_l(t) \cdot f_2(W(p,t)) - \sum_{t \in T} |V|_l(t) \cdot f_1(W(p,t)) = m_0(p) + \sum_{v \in V} f_2(W(p, l(v))) - \sum_{v \in V} f_1(W(p, l(v)))$.

Proof of Theorem 1

Let LS be the free abelian monoid $\mathbb{N}^A$ over the set A. Define $\mathrm{lpo} = (V, <, l)$ through $V = V' \cup \{v_{min}, v_{max}\}$, $\forall v \in V' : v_{min} < v < v_{max}$, $l(v_{min}) \neq l(v_{max})$ and $l(v_{min}), l(v_{max}) \notin l(V')$. We will show the theorem by contradiction, i.e. we assume that there is a place p for which there does not exist a token flow function $x_p :< \to LS$ such that

(i) $\forall v \neq v_{max} : In_{x_p}(v) = f_1(W(p, l(v)))$.
(ii) $Out_{x_p}(v_{min}) = m_0(p)$ and $\forall v \neq v_{max} : Out_{x_p}(v) = f_2(W(p, l(v)))$.

Denote $V = \{v_0, \ldots, v_{|V|}\}$ such that $v_i < v_j$ implies $i < j$, in particular $v_0 = v_{min}$. Consider the set $\mathcal{X}$ of token flow functions which satisfy (i) and $Out_{x_p}(v_{min}) \geq m_0(p)$ and $\forall v \neq v_{max} : \ Out_{x_p}(v) \geq f_2(W(p, l(v)))$. Observe that this set is non-empty, e.g. the function x_p, defined by $x_p(v_{min}, v') = f_1(W(p, l(v')))$ for every $v' \in V'$, $x_p(v', v_{max}) = f_2(W(p, l(v')))$ for every $v' \in V'$, $x_p(v_{min}, v_{max}) = m_0(p)$ and $x_p(v, v') = 0$ for every $v < v'$, $v, v' \in V'$, is in $\mathcal{X}$. By assumption, none of the functions in $\mathcal{X}$ fulfils (ii).

We say that a function $x \in \mathcal{X}$ *does not fulfil* (ii) *for an index* i, if $i = 0$ and $Out_x(v_i) > m_0(p)$ or if $i > 0$ and $Out_x(v_i) > f_2(W(p, l(v_i)))$. Denote k_x the smallest index for which a flow function $x \in \mathcal{X}$ does not fulfil (ii). Let $\mathcal{X}_{sup} \subseteq \mathcal{X}$ be the non-empty set of all token flow functions $x \in \mathcal{X}$ which maximize k_x, i.e. such that there holds $\forall x', x'' \in \mathcal{X}_{sup} : \ k_{x'} = k_{x''}$ and $\forall x \in \mathcal{X}$, $\forall x' \in \mathcal{X}_{sup} : \ k_x \leqslant k_{x'}$. Denote $sup = k_x$ for $x \in \mathcal{X}_{sup}$. By assumption $sup < |V|$ (note that $sup = 0$ is possible).

Finally, choose a token flow function $x_0 \in \mathcal{X}_{sup}$ which minimizes $Out_{x_0}(v_{sup})$, i.e. such that there holds $\forall x \in \mathcal{X}_{sup} : \ Out_x(v_{sup}) \not< Out_{x_0}(v_{sup})$.

In the following, we construct from x_0 a co-set C' of lpo' such that $|C'|_l$ is not enabled in the final marking of the prefix $\mathrm{lpo}'' = (D', <|_{D' \times D'}, l|_{D'})$, $D' = \{v \in V' \mid v < C'\}$, of C'. By assumption such prefix is enabled w.r.t. N, because lpo' is enabled. By Lemma 10, there holds for the final marking m of lpo'':

$$m(p) = m_0(p) + \sum_{v \in D'} f_2(W(p, l(v))) - \sum_{v \in D'} f_1(W(p, l(v))).$$

To show that $|C'|_l$ is not enabled in m we show that there exists no m' such that $(m(p), \sum_{t \in T} |C'|_l(t) \cdot W(p, t), m'(p)) \in \tau$. By the flow type of nets definition it suffices to verify $m(p) \not\geq f_1(\sum_{t \in T} |C'|_l(t) \cdot W(p, t)) = \sum_{t \in T} |C'|_l(t) \cdot f_1(W(p, t)) = \sum_{v \in C'} f_1(W(p, l(v)))$, i.e.

$$(*) \quad m_0(p) + \sum_{v \in D'} f_2(W(p, l(v))) - \sum_{v \in D'} f_1(W(p, l(v))) \not\geq \sum_{v \in C'} f_1(W(p, l(v))).$$

This contradicts the enabledness of lpo'. To this end we next define the sets of nodes C' and D such that $D = D' \cup \{v_{min}\}$, D turns out to define the prefix of C' in lpo given by the nodes smaller than C' and C' is a co-set.

Consider $a \in A$ such that $Out_{x_0}(v_{sup})(a) > f_2(W(p, l(v_{sup})))(a)$ (resp. $Out_{x_0}(v_{sup})(a) > m_0(p)(a)$ if $sup = 0$). Let D be the set of all nodes $v \in V$ such that there exists a sequence of nodes $\sigma(v) = v^0 w^1 v^1 \ldots w^k v^k$ with $v^0 = v_{sup}$ and $v^k = v$ satisfying

(C1) $\forall j \neq m : w^j \neq w^m \wedge v^j \neq v^m$, and
(C2) $\forall j : \ x_0(v^j, w^{j+1})(a) > 0 \wedge v^j < w^j$.

Since $Out_{x_0}(v_{sup})(a) > 0$, the initial node $v_0 = v_{min}$ is in D. Moreover, $v_{sup} \in D$ (case $k = 0$). The node $v_{max} = v_{|V|}$ is not in D, since $v_{max} \neq v_{sup}$ and there is no node w with $v_{max} < w$.

Define

$$C' = \{w \in V \setminus D \mid \exists v \in D : \ x_0(v, w)(a) > 0\}$$

The set C' represents the step of transitions "consuming too much tokens" (of $l(v_{sup})$). We prove in several steps that C' is a co-set of lpo' having the prefix lpo'' as described before and satisfying $(*)$. The idea is that if this is not the case, along the paths $\sigma(v)$ token flow can be redistributed in such a way that the outtoken flow of v_{sup} w.r.t. a is reduced, while the outtoken flow of v_{sup} w.r.t. $a' \neq a$, the intoken flows of all nodes and the outtoken flows of nodes with index $i < sup$ are not changed. However, this is not possible by the choice of x_0.

Claim 1: $v_j \in D \Longrightarrow j \leqslant sup$

Assume $j > sup$. Then it is possible to construct a token flow function $x \in \mathcal{X}_{sup}$ with $Out_x(v_{sup}) < Out_{x_0}(v_{sup})$, which contradicts the choice of x_0, as follows: Let $\sigma(v) = v^0 w^1 v^1 \dots w^k v^k$ and set

$$\begin{aligned}\forall j : x(v^j, w^j)(a) &= x_0(v^j, w^j)(a) + 1\\ \forall j : x(v^j, w^{j+1})(a) &= x_0(v^j, w^{j+1})(a) - 1\\ else : x(v, v')(a') &= x_0(v, v')(a').\end{aligned}$$

Claim 2: $(v_j \in D) \Longrightarrow x_0(v_j, v_{max})(a) = 0$

From Claim 1 we deduce $j \leq sup$. Assume $x_0(v_j, v_{max})(a) > 0$. Then it is possible to construct a token flow function $x \in \mathcal{X}_{sup}$ with $Out_x(v_{sup}) < Out_{x_0}(v_{sup})$, which contradicts the choice of x_0, as follows: In the case $j < sup$, let $\sigma(v) = v^0 w^1 v^1 \dots w^k v^k$ and set

$$\begin{aligned}x(v^j, v_{max})(a) &= x_0(v^j, v_{max})(a) - 1,\\ \forall j : x(v^j, w^j)(a) &= x_0(v^j, w^j)(a) + 1,\\ \forall j : x(v^j, w^{j+1})(a) &= x_0(v^j, w^{j+1})(a) - 1,\\ else : x(v, v')(a') &= x_0(v, v')(a').\end{aligned}$$

In the case $j = sup$ set $x(v^j, v_{max})(a) = x_0(v^j, v_{max})(a) - 1$ and $x(v, v')(a') = x_0(v, v')(a')$ else.

Claim 2 shows that $v_{max} \notin C'$, i.e. $C' \subseteq V'$.

Claim 3: $\forall v \in V :\ (\exists w \in C' :\ v < w) \Longleftrightarrow v \in D$

$\Longrightarrow$: Let $w \in C'$ with $v < w$. We construct a sequence $\sigma(v) = v_{sup} \dots v$ fulfilling $(C1)$ and $(C2)$. By the definition of C' there is a node $v' \in D$ with $x_0(v', w)(a) > 0$. Let $\sigma(v') = v_{sup} w^1 v^1 \dots w^k v^k$. In the case $v = v^j$ for $j \in \{0, \dots, k\}$ it follows $v \in D$. We distinguish the following remaining cases:

- $(\exists j \in \{0, \dots, k\} :\ w^j = w)$: Denote m the smallest index with $w^m = w$. Then $v_{sup} w^1 v^1 \dots w^m v$ satisfies $(C1)$ and $(C2)$.
- $(\forall j \in \{0, \dots, k\} :\ w^j \neq w)$: $v_{sup} w^1 v^1 \dots w^k v' w v$ satisfies $(C1)$ and $(C2)$.

$\Longleftarrow$: Let $v \in D$, we will find $w \in C'$ with $v < w$. If $v = v_{sup}$, then there is $w = v_j$, $j > sup$, $x_0(v_{sup}, v_j)(a) > 0$. By Claim 1, $v_j \notin D$ and consequently $w = v_j \in C'$. If $v \neq v_{sup}$, let $\sigma(v) = v_{sup} w^1 v^1 \ldots w^k v^k$. By the definition of C', the node w_k can either be in C' or in D, because $x_0(v^{k-1}, w^k)(a) > 0$ and $v^{k-1} \in D$. We distinguish these cases:

- $w^k \in C'$: $v = v^k < w^k \in C'$.
- $w^k \in D$: Denote $\overline{v}$ a maximal node in the set $\{v' \in D \mid v < v'\}$ w.r.t. $<$ (the set is not empty since w^k is one of its elements). If $\overline{v} = v_{sup}$, then $v < \overline{v} < w \in C'$ (the existence of such w has already been shown). Otherwise let $\sigma(\overline{v}) = v_{sup} \overline{w}^1 \overline{v}^1 \ldots \overline{w}^l \overline{v}^l$ satisfy $(C1)$ and $(C2)$. Then $\overline{w}^l \notin D$ (otherwise $\overline{v}$ would not be maximal) and thus $\overline{w}^l \in C'$, because $x_0(\overline{v}^{l-1}, \overline{w}^l)(a) > 0$ and $\overline{v}^{l-1} \in D$. Consequently $v < \overline{v} < \overline{w}^l \in C'$.

Claim 3 in particular shows that C' is a co-set, because $v < w \in C' \Longrightarrow v \in D \Longrightarrow v \notin C'$. Moreover, it shows $D = \{v \in V \mid v < C'\}$, i.e. $D' = D \setminus \{v_{min}\} = \{v \in V' \mid v < C'\}$.

Claim 4: C' satisfies $(*)$

If $sup = 0$, i.e. $D = \{v_0\}$, then $(*)$ means $m_0(p) \not\geq Out_{x_0}(v_0)$ (because in this case for $v \in C'$: $f_1(W(p, l(v))) = In_{x_0}(v) = x_0(v_0, v)$) – this holds by assumption. Let $sup > 0$. According to Claim 1 there holds $Out_{x_0}(v_0) = m_0(p)$ and $\forall v \in D \setminus \{v_0, v_{sup}\} : Out_{x_0}(v) = f_2(W(p, l(v)))$. More precisely, according to Claim 2, we have $\sum_{v_0 \prec v',\, v' \in V'} x_0(v_0, v') = m_0(p)$ and $\forall v \in D \setminus \{v_0, v_{sup}\} : \sum_{v < v',\, v' \in V'} x_0(v, v') = f_2(W(p, l(v)))$. Finally, by assumption we have $\forall v \in D \cup C' : In_{x_0}(v) = f_1(W(p, l(v)))$ and $Out_{x_0}(v_{sup})(a) > f_2(W(p, l(v_{sup})))(a)$. Altogether we compute $m_0(p)(a) + \sum_{v \in D'} f_2(W(p, l(v)))(a) - \sum_{v \in D'} f_1(W(p, l(v)))(a) - \sum_{v \in C'} f_1(W(p, l(v)))(a) < \sum_{v \in D} (\sum_{v \prec v'} x_0(v, v')(a)) - \sum_{v \in D} (\sum_{v' \prec v} x_0(v', v)(a)) - \sum_{v \in C'} (\sum_{v' \prec v} x_0(v', v)(a)) = 0$.

The last equation holds since each summand $x_0(v, v')(a)$ either (i) equals 0, or (ii) is counted exactly once positively and once negatively: There are only summands $x_0(v, v')(a)$ with $v \in D$. For $(v, v') \in D \times (D \cup C')$ case (ii) holds according to Claim 3 and for $(v, v') \in D \times (V \setminus D)$ with $x_0(v, v') > 0$ we have $v' \in C'$ by definition – that means (ii) holds in each case (i) does not hold. Thus, we have $m_0(p)(a) + \sum_{v \in D'} f_2(W(p, l(v)))(a) - \sum_{v \in D'} f_1(W(p, l(v)))(a) < \sum_{v \in C'} f_1(W(p, l(v)))(a)$ showing (*).

Altogether $|C'|_l$ is not enabled in the final marking of the prefix lpo'' of C' within lpo'. By Lemma 1, lpo' is not enabled, a contradiction.

Proof of Lemma 11

Define $\mathrm{lpo} = (V, <, l)$ through $V = V' \cup \{v_{min}, v_{max}\}$, $\forall v \in V' : v_{min} < v < v_{max}$, $l(v_{min}) \neq l(v_{max})$ and $l(v_{min}), l(v_{max}) \notin l(V')$ and $\mathbf{x}$ such that $(\mathrm{lpo}, \mathbf{x})$ fulfills the token flow property. Assume lpo' is not enabled. Consider a prefix $\mathrm{lpo}_p = (V_p, <_p, l_p)$

of lpo' which is also not enabled and minimal with this property, i.e. every proper prefix of lpo_p is enabled. By Lemma 1 there is a non-empty co-set C' of lpo_p and a prefix (within lpo_p) $\mathrm{lpo}'' = (V'', <'', l'')$ of C' such that the step of transitions $|C'|_l$ is not enabled in the final marking m of lpo'' (lpo'' is enabled by the minimality property of lpo_p). By Lemma 10 we have $m(p) = m_0(p) + \sum_{v \in V''} f_2(W(p, l(v))) - \sum_{v \in V''} f_1(W(p, l(v)))$ for the final marking m of lpo'' for all $p \in P$. Denote $V''_m = V'' \cup \{v_{min}\}$. By the token flow property $m(p) = Out_{x_p}(v_{min}) + \sum_{v \in V''} Out_{x_p}(v) - \sum_{v \in V''} In_{x_p}(v) = \sum_{v \in V''_m}(\sum_{v < v'} x_p(v, v')) - \sum_{v \in V''}(\sum_{v' < v} x_p(v', v)) = \sum_{(v,v') \in < \cap (V''_m \times (V \setminus V''_m))} x_p(v, v') \geq \sum_{(v,v') \in < \cap (V''_m \times C')} x_p(v, v') = \sum_{v \in C'} In_{x_p}(v) = \sum_{v \in C'} f_1(W(p, l(v))) = f_1(\sum_{t \in T} |C'|_l(t) \cdot W(p, t))$. By the required property of (τ, f), we conclude that for $m'(p) = m(p) - f_1(\sum_{t \in T} |C'|_l(t) \cdot W(p, t)) + f_2(\sum_{t \in T} |C'|_l(t) \cdot W(p, t))$ we have $(m(p), \sum_{t \in T} |C'|_l(t) \cdot W(p, t), m'(p)) \in \tau$. Thus $|C'|_l$ is enabled in m, a contradiction.

Proof of Corollary 1

The "only if" statement follows from Theorem 1 and Lemma 1. The "if" part can be proven analogously to Lemma 11 additionally regarding the non-blocking property.