Loop Home Markings (Looms) in Free-Choice Nets with Home Clusters

Thomas M. Prinz[1,*,†], Wil M. P. van der Aalst[2,†], Christopher T. Schwanen[2,†] and Yongsun Choi[3,†]

[1] *Course Evaluation Service, Friedrich Schiller University Jena, 07743 Jena, Germany*
[2] *Chair of Process and Data Science (PADS), RWTH Aachen University, 52074 Aachen, Germany*
[3] *Department of Industrial and Management Engineering, Inje University, 50834 Gimhae, Republic of Korea*

Abstract

Loops play a crucial role in Petri nets since important properties such as liveness depend on them and, in proper nets, it requires loops for an unbounded number of tokens. This paper investigates loops as non-trivial strongly connected components and introduces the concept of *looms* as a kind of home markings of loops. A loom describes a marking of a loop that can re-occur independently of tokens possibly being outside of this loop. For cyclic nets with a home marking, the home marking is a loom. However, in non-cyclic nets, loops may have exits so that looms build a helpful concept of describing that the loop behaves "ordinary". Furthermore, this paper shows seven properties of loops in free-choice nets with a home cluster. They include that all exits and entries of loops are places and that each smallest enabling marking of a cluster containing a loop exit describes a loom. Finally, this paper sketches potential applications of looms through loop unrolling and identifying nested loop structures.

Keywords

Net, Home marking, Loop, Free-choice, Loom, Nested Loops, Unrolling

1. Introduction

A *home marking* of a Petri net describes a marking that is reachable from all reachable markings of this net [1, 2]. For example, *cyclic* Petri nets have an initial marking that is a home marking. This allows cyclic nets to always reach their initial marking as a kind of "recovery point" regardless of what decisions were or will be performed [3]. For automatons in practice (e. g., cash machines), the benefit of having such a home marking is obvious. However, home markings have not to be recovery points allowing for refreshing; they can also define "termination points". For example, in sound workflow nets (with all transitions and places being aligned between a single source and a single sink place), a marked sink place is a home marking representing its final marking [4]. A special kind of a home marking is a *home cluster*.

A *cluster* of a Petri net describes a subgraph containing a minimal set of transitions and places for which it holds that if all places have a token in a marking, then all the cluster's transitions are enabled [1, 2]. Note that, for example, in workflow nets, there is a cluster consisting only of the single sink place. A *home cluster* of a net is a cluster whose enabling marking (i. e., the marking in which only all cluster places have exactly one token) is a home marking of the net. For example, a cyclic Petri net with a home cluster is *perpetual* [5] (see Figure 1 with home cluster $C = \{p_7, p_8, t_7\}$ as an example). If a workflow net's sink place is a home cluster, then the net is sound [4] (cf. Figure 2 with the home cluster $\{o\}$).

In this paper, we investigate "home markings" of *loops* in Petri nets. A *loop* is a non-trivial strongly-connected component (SCC) [6], thus, each node within a loop has paths to all other nodes in the loop. A loop has a kind of home marking if it has a re-occurring behavior being independent of the rest of the net. We call such a kind of home marking a *loom* (short for loop home marking). In Petri nets, not every loop

PNSE'26, International Workshop on Petri Nets and Software Engineering, 2026
*Corresponding author.
†The order in which these authors are listed indicates the extent of their contributions.
Thomas.Prinz@uni-jena.de (T. M. Prinz); wvdaalst@pads.rwth-aachen.de (W. M. P. v. d. Aalst); schwanen@@pads.rwth-aachen.de (C. T. Schwanen); yschoi@inje.edu (Y. Choi)
0000-0001-9602-3482 (T. M. Prinz); 0000-0002-0955-6940 (W. M. P. v. d. Aalst); 0000-0002-3215-7251 (C. T. Schwanen); 0000-0002-8605-4055 (Y. Choi)

must have a loom. The primary research question addressed in this paper is *how the behavior of loops can be analyzed and represented without considering the rest of a net*. More specifically, we will investigate loops in free-choice nets (a well-studied class of nets [1, 7, 8, 9]) with a home cluster [4] and will show that each loop in such nets has a loom being representable as a cluster. Obviously, if the investigated net is perpetual (such as that in Figure 1), it is a loop by definition and has a home cluster as loom.

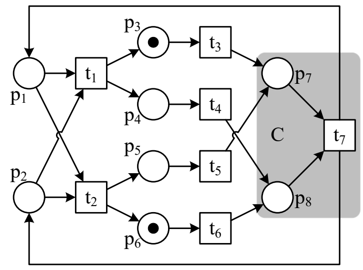


Figure 1: A (free-choice) perpetual net [1] with home cluster C.

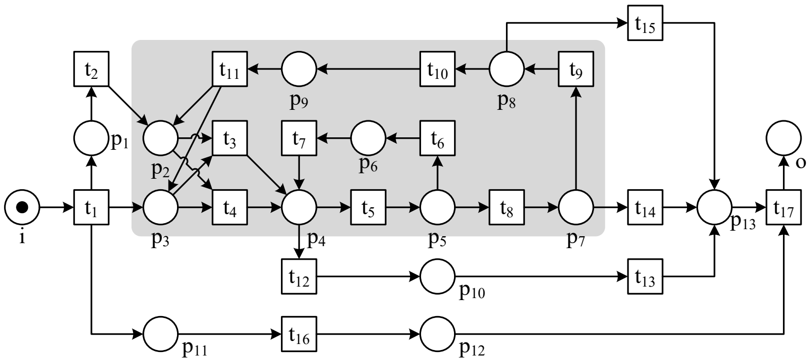


Figure 2: A marked free-choice workflow net with a loop (light gray rounded rectangle).

In the (workflow) net in Figure 2, there is a loop highlighted in a light gray rounded rectangle. The net's reachable marking $[p_1, p_3, p_{11}]$ has a token in the loop on p_3, one token on a path to the loop (p_1), and one "independent" token on p_{11}. Regarding the loop, the sub-marking $[p_3]$ cannot re-occur without other places of the loop (i. e., without p_2). Thus, it is *not* a loom. However, a loom of this loop is the sub-marking $[p_4]$ that can be described by the cluster $\{p_4, t_5, t_{12}\}$. Concentrating on all reachable markings from any marking covering $[p_4]$, the loop behaves "independently" from other subgraphs of the net (e. g., from tokens on places p_{11} or p_{12}). Consequently, knowing looms will simplify analysis as starting from this loom, the related loop can be analyzed separately from the rest of the net.

The scope of this paper is to introduce looms and to identify them in free-choice nets with a home cluster. Furthermore, this paper will sketch two potential applications of looms: (1) identifying loop structures within loops (so-called *nested* loops) enabling a recursive investigation of loops and (2) *unrolling* loops. Note that not every loop in any net must have a loom, just as not every net has a home marking.

The paper is structured as follows: Section 2 introduces basic notions for the paper followed by a description of loops and the formalization of the concept of looms in general in Section 3. Subsequently, Section 4 derives basic properties of loops for free-choice nets with a home cluster. These properties result in the finding that each loop in a free-choice net with a home cluster has a loom in Section 5. Eventually, the paper concludes with Section 6.

2. Preliminaries

This paper is based on standard notions of Petri nets, which are defined in the following. Readers who are firm with typical Petri net notions can skip this section until Section 2.3 about rooted disentangled paths.

2.1. Multisets, Petri nets, and Paths

$\mathscr{B}(A)$ is the set of all *multisets* over some set A. For a multiset $b \in \mathscr{B}(A)$, $b(a)$ denotes the number of times element $a \in A$ appears in b. For example, $b_1 = [\,]$, $b_2 = [x,x,y]$, $b_3 = [x,y,z]$, $b_4 = [x,x,y,x,y,z]$, and $b_5 = [x^3, y^2, z]$ are multisets over the set $A = \{x,y,z\}$. b_1 is the empty multiset, b_2 and b_3 consist of three elements, and $b_4 = b_5$, i. e., the ordering of elements is irrelevant and b_5 uses a more compact notation for repeating elements. The standard set operators can be extended to multisets, e. g., $x \in b_2$, $b_2 \uplus b_3 = b_4$, $b_5 \setminus b_2 = b_3$, etc. For intersections of multisets, we use a projective variant with $a \upharpoonright b := [x^y \mid a(x) = y \wedge x \in b]$, e. g., $b_2 \upharpoonright b_3 = [x^2, y]$.

Definition 1 (Petri nets)**.** A *Petri net* (or simply a *net*) N is a triple (P,T,F) where P and T are finite, disjoint sets of *places* and *transitions*, and $F \subseteq (P \times T) \cup (T \times P)$ is the *flow* relation. ⌟

$P \cup T$ can be interpreted as *nodes* and F as *flows* between those nodes. For $x \in P \cup T$, $\bullet x := \{ y \mid (y,x) \in F \}$ is the *preset* of x (all directly preceding nodes) and $x\bullet := \{ y \mid (x,y) \in F \}$ is the *postset* of x (all directly succeeding nodes). Each node in $\bullet x$ is an *input* of x and each node in $x\bullet$ is an *output* of x. The preset and postset of a set of nodes $X \subseteq P \cup T$ is defined as $\bullet X := \bigcup_{x \in X} \bullet x$ and $X\bullet := \bigcup_{x \in X} x\bullet$, respectively. N is *proper* iff $\forall t \in T\colon\ \bullet t \neq \varnothing\ \wedge\ t\bullet \neq \varnothing$. N is *free-choice* iff $\forall t_1, t_2 \in T\colon\ \bullet t_1 \cap \bullet t_2 \neq \varnothing \implies \bullet t_1 = \bullet t_2$.

A *path* $\rho = (n_1, \ldots, n_m)$ is a sequence of nodes $n_1, \ldots, n_m \in P \cup T$ with $m \geqslant 1$ and $\forall i \in \{1, \ldots, m-1\}\colon\ n_i \in \bullet n_{i+1}$. Note that places and transitions alternate on paths. $\langle (n_1, \ldots, n_m) \rangle$ denotes the set of all nodes on the path. The *concatenation* of two paths $\rho \cdot (a_1, \ldots, a_l)$, $l \geqslant 1$, is possible if $n_m = a_1$ and results in a new path $\rho \cdot (a_1, \ldots, a_l) := (n_1, \ldots, n_m, a_2, \ldots, a_l)$. If all nodes of a path are pairwise different, the path is *acyclic*; otherwise, it is *cyclic*. $Paths(x,y)$ denotes the set of all paths between nodes x and y, where $x, y \in P \cup T$. N is *acyclic* if all its paths are acyclic.

A net N is *connected* if, in its "undirected" version $N' = (P, T, \{ (n_1, n_2) \mid (n_1, n_2) \in F\ \vee\ (n_2, n_1) \in F \})$, each node has a path to all other nodes. A net $N_s = (P_s, T_s, F_s)$ is a *subnet* of $N = (P,T,F)$ iff $P_s \subseteq P$, $T_s \subseteq T$, and $F_s \subseteq F \cap \big((P_s \times T_s) \cup (T_s \times P_s)\big)$. *Note that this means that the subnet does not require to contain all related flows from the original net.* N_s is a *strongly connected component* (SCC) of N if (1) N_s is a subnet of N, (2) each node of N_s has paths to all other nodes in N_s, and (3) it is not possible to add further nodes and flows while retaining the other two properties (1) and (2) [10].

Definition 2 (Workflow Nets)**.** A *workflow net* $WN = (P,T,F,i,o)$ is a net (P,T,F) with $i, o \in P$, $\bullet i = o\bullet = \varnothing$. i is the *source* and o is the *sink* of WN. All nodes are on a path from i to o. ⌟

2.2. Markings, Reachability, Properties, and Home Markings

The behavior of nets is defined via *markings*, which describe the number of *tokens* on places.

Definition 3 (Marking)**.** A *marking* M of a net $N = (P,T,F)$ is a multiset of places, $M \in \mathscr{B}(P)$. (N,M) is a *marked net*. ⌟

Transitions whose input places all have tokens are *enabled* in a marking and can be fired, leading to the net's semantics:

Definition 4 (Enabledness, Firing, and Reachability)**.** Let (N,M) be a marked net $N = (P,T,F)$. A transition $t \in T$ is *enabled* in M, denoted as $(N,M)[t\rangle$, iff every place in $\bullet t$ contains at least one token in M, $\bullet t \subseteq M$. $en(N,M) := \{ t \in T \mid (N,M)[t\rangle \}$ is the set of enabled transitions in M.

If t is enabled in M, then t may *fire*, which removes one token from each of t's input places and adds one token to each of t's output places. $M' = (M \setminus \bullet t) \uplus t\bullet$ is the marking resulting from firing t in (N,M). $(N,M)[t\rangle(N,M')$ denotes that t is enabled in (N,M) and firing t would result in (N,M').

A sequence $\sigma = \langle t_1, t_2, \ldots, t_n \rangle \in T^*$ is an *occurrence sequence* if there are markings $M_1, M_2, \ldots, M_{n+1}$ such that $\forall 1 \leqslant i \leqslant n\colon\ (N,M_i)[t_i\rangle(N,M_{i+1})$ (this includes the empty sequence $\sigma = \langle\,\rangle$ with $M_1 = M_{n+1}$). Thus, σ leads from (N,M_1) to (N,M_{n+1}), denoted as $(N,M_1)[\sigma\rangle(N,M_{n+1})$. We say that $t_1, t_2, \ldots, t_n$ *occur* in σ. A place $p \in P$ occurs in σ, denoted by $p \in \sigma$, iff $\exists M_i \in \{ M_1, \ldots, M_{n+1} \}\colon\ p \in M_i$.

A marking M' is *reachable* from M if there is an occurrence sequence σ such that $(N,M)[\sigma\rangle(N,M')$. $R(N,M) := \{ M' \in \mathscr{B}(P) \mid \exists \sigma \in T^*\colon\ (N,M)[\sigma\rangle(N,M') \}$ denotes the set of all reachable markings of (N,M). ⌟

Certain behavioral properties of nets are known to be important for analysis:

Definition 5 (Live, Bounded, Safe, and Dead)**.** Let (N,M) be a marked net $N = (P,T,F)$. (N,M) is *live* iff for every reachable marking $M' \in R(N,M)$ and for every transition $t \in T$, there is a reachable marking $M'' \in R(N,M')$ with $t \in en(N,M'')$. (N,M) is *k-bounded* for a $k \in \mathbb{N}$ iff for every reachable marking $M' \in R(N,M)$ and for every place $p \in P\colon\ M'(p) \leqslant k$. (N,M) is *bounded* iff there is a $k \in \mathbb{N}$ such that (N,M) is k-bounded. (N,M) is *safe* iff (N,M) is 1-bounded. A place $p \in P$ is *dead* in (N,M) iff $\forall M' \in R(N,M)\colon\ p \notin M'$. A transition $t \in T$ is *dead* in (N,M) iff $\forall M' \in R(N,M)\colon\ t \notin en(N,M')$. ⌟

Workflow nets can be *sound*.

Definition 6 (Soundness [11])**.** A workflow net $WN = (P,T,F,i,o)$ with its initial marking $[\,i\,]$ and its final marking $[\,o\,]$ is *sound* iff

(1) from any reachable marking of $[\,i\,]$, $[\,o\,]$ is reachable: $\forall M \in R(WN,[\,i\,]):\ \ [\,o\,] \in R(WN,M)$ and

(2) there is no *dead* transition in WN: $\forall t \in T\ \exists M,M' \in R(WN,[\,i\,]):\ \ (WN,M)[t\rangle(WN,M')$. ⌟

The free-choice workflow net in Figure 2 is sound. This paper is about free-choice nets with a *home cluster*. *Home markings* are the general concept of home clusters:

Definition 7 (Home Marking)**.** Let (N,M) be a marked net. A marking M_H is a *home marking* iff $\forall M' \in R(N,M):\ \ M_H \in R(N,M')$. ⌟

For a sound workflow net (P,T,F,i,o) with its initial marking $[\,i\,]$, $[\,o\,]$ is a home marking.

2.3. Clusters and Rooted Disentangled Paths

Home clusters and *rooted disentangled paths* as well as their related properties were introduced in [4]. If the reader demands for a more detailed introduction, we kindly refer to this work. At first, we need the concept of *clusters*:

Definition 8 (Cluster)**.** Let $N = (P,T,F)$ be a net and $x \in P \cup T$ one of its nodes. The *cluster* of x, denoted $[\![x]\!]$, is the smallest set such that:

$$x \in [\![x]\!] \quad \wedge \quad p \in ([\![x]\!] \cap P) \implies p\bullet \subseteq [\![x]\!] \quad \wedge \quad t \in ([\![x]\!] \cap T) \implies \bullet t \subseteq [\![x]\!].$$

The set $[\![N]\!] := \{\,[\![x]\!] \mid x \in P \cup T\,\}$ contains all clusters of N. ⌟

$[\![t_1]\!] = \{\,p_1,t_1\,\}$, $[\![p_2]\!] = \{\,p_2,p_3,t_3,t_4\,\}$, and $[\![p_7]\!] = \{\,p7,t_9,t_{14}\,\}$ are examples of clusters in Figure 2. Note, in a workflow net, there is only one cluster consisting only of a single place—the sink cluster. Since a cluster contains all input places of its included transitions, it is possible to describe a minimal marking enabling all transitions of the cluster.

Definition 9 (Smallest Enabling Marking)**.** Let $N = (P,T,F)$ be a net and $C \in [\![N]\!]$ one of its clusters. $Mrk(C) := [\,p \in P \cap C\,]$ is the *smallest marking enabling* C. ⌟

In Figure 2, $[\,p_2,p_3\,]$ is the smallest enabling marking $Mrk([\![p_2]\!])$ of cluster $[\![p_2]\!] = \{\,p_2,p_3,t_3,t_4\,\}$. A cluster whose smallest enabling marking is a home marking is called a *home cluster*:

Definition 10 (Home Cluster)**.** Let (N,M) be a marked net. C is a *home cluster* of (N,M) iff $C \in [\![N]\!]$ and $Mrk(C)$ is a home marking of (N,M). ⌟

$[\![o]\!]$ is the home cluster of a workflow net with its sink o. Home clusters are fix points for analysis, for which it is known that they are reachable from all other markings. There cannot be any "deadlock". Actually, every free-choice net with a home cluster is safe [4]:

Lemma 1 (Safeness [4] (Cor. 5.12))**.** Let (N,M) be a marked proper free-choice net with a home cluster. (N,M) is safe. □

Later on, we will use paths in proofs to explain how tokens "travel" through the nets. It is of benefit if these paths are *rooted disentangled paths*, where each place belongs to a different cluster:

Definition 11 (Rooted Disentangled Paths)**.** Let $N = (P,T,F)$ be a net. A path $\rho = (a_1,a_2,\ldots,a_{n-1},a_n) \in Paths(a_1,a_n)$ is a *disentangled path* of N iff $[\![a_1]\!] \neq [\![a_n]\!]$[1] and $\forall 1 \leqslant i < j \leqslant n:\ \{\,a_i,a_j\,\} \subseteq P \implies [\![a_i]\!] \neq [\![a_j]\!]$; thus ρ starts and ends in different clusters and all places of ρ are in different clusters. A disentangled path is C-rooted iff $a_n \in C$ and $C \in [\![N]\!]$. $\mathscr{P}_N^R(a_1,a_n)$ denotes the set of all $[\![a_n]\!]$-rooted disentangled paths from a_1 to a_n in N. Similarly, $\mathscr{P}_N^X(a_1)$ denotes the set of all X-rooted disentangled paths from a_1 to any node of cluster X in N. ⌟

[1]This ensures that, even if a_1 and a_n are transitions, a_1 and a_n are in different clusters.

In Figure 2, (p_3,t_4,p_4,t_5,p_5) is a $[\![p_5]\!]$-rooted disentangled path, whereas $(p_3,t_4,p_4,t_5,p_5,t_8,p_7,t_9,p_8,t_{10},p_9,t_{11},p_2)$ is *not* such a path as $[\![p_3]\!] = [\![p_2]\!]$. Van der Aalst [4] has shown that, if there is a path between two places, then there is at least one disentangled path between them:

Lemma 2 (Existence of Rooted Disentangled Paths [4] (Lemma 5.9))**.** Let $N = (P,T,F)$ be a free-choice net, C a cluster of N, $p \in P$, and $q \in C \cap P$. It holds that: $\mathit{Paths}(p,q) \neq \emptyset \implies \mathscr{P}^R(p,q) \neq \emptyset$. ◻

Furthermore, each rooted disentangled path in a free-choice net with home cluster is safe (i. e., the sum of all tokens on such path is at most 1 in each reachable marking):

Lemma 3 (Rooted Disentangled Paths are Safe [4] (Lemma 5.11))**.** Let (N,M) be a marked proper free-choice net having a home cluster C. For any reachable marking $M' \in R(N,M)$ and C-rooted disentangled path ρ: $|M' \upharpoonright \langle\rho\rangle| \leqslant 1$. ◻

Following Van der Aalst [4], each marked proper free-choice net (N,M) with a home cluster can be reduced to a net without dead nodes (called *cleaned net* of (N,M)) while this cleaned net represents the same reachable markings as (N,M).

Definition 12 (Cleaned Nets)**.** Let (N,M) be a marked proper free-choice net $N = (P,T,F)$ with a home cluster. The *cleaned net* $\mathit{clean}(N,M) := (P_C,T_C,F_C)$ of N regarding M is a subnet of N defined as follows:

$$P_C := \{\, p \in P \mid \exists i \in M\colon\ \mathit{Paths}(i,p) \neq \emptyset \,\},$$

$$T_C := \{\, t \in T \mid \exists i \in M\colon\ \mathit{Paths}(i,t) \neq \emptyset \,\}, \qquad F_C := F \cap \big((P_C \times T_C) \cup (T_C \times P_C)\big).$$ ⌟

3. Loops and Looms

In the focus of this work are *loops* being a concept of graphs representing control flows:

Definition 13 (Loops, Nested Loops, Loop Entries, Loop Exits, and Loop Exit Areas [12])**.** A *loop* $L = (P_L,T_L,F_L)$ of a net N is a non-trivial SCC of N (i. e., an SCC not only consisting of one node). $\mathit{Loops}(N)$ denotes the set of all loops of N. A loop L_n is called *nested loop* of L regarding a set of flows $F_n \subset F_L$ if L_n is a loop of $(P_L,T_L,F_L \backslash F_n)$.

$en \in P_L \cup T_L$ is a *loop entry* of L iff $\bullet en \backslash (P_L \cup T_L) \neq \emptyset$ and $ex \in P_L \cup T_L$ is a *loop exit* of L iff $ex\bullet \backslash (P_L \cup T_L) \neq \emptyset$. $\mathit{Entries}_N(L)$ denotes the set of all loop entries and $\mathit{Exits}_N(L)$ denotes the set of all loop exits of L related to N.

$\mathit{ExitArea}_N(L,C) = \{\, ea \in P \cup T \mid \exists o \in \big(\mathit{Exits}_N(L)\bullet \backslash (P_L \cup T_L)\big)\ \exists \rho_{o\to C} \in \mathscr{P}_N^C(o)\colon\ ea \in \langle\rho_{o\to C}\rangle \,\}$ denotes the *exit area* of L related to net N and a cluster C. The exit area contains all nodes outside of L lying on a C-rooted disentangled path starting in a node of L. ⌟

The entire net in Figure 1 constitutes a loop. This loop has no entry and no exit. Figure 2 highlights a loop with a light-gray rounded rectangle. This loop has p_2 and p_3 as loop entries and p_4, p_7, and p_8 as loop exits. Regarding the cluster $[\![o]\!]$, the exit area of this loop consists of nodes $\{\, t_{12}, p_{10}, t_{13}, t_{14}, t_{15}, p_{13}, t_{17}, o \,\}$. If we remove the flow (t_8, p_7), the set $\{\, p_4, t_5, p_5, t_6, p_6, t_7 \,\}$ describes a “nested” loop of the outer loop regarding the removed flows.

Instead of loops, one can also just talk about SCCs. We intentionally decided to use the term “loop” as (1) we already used this term in previous work (e. g., [12]), (2) the term builds a bridge to compiler theory, software engineering, programming languages, and related fields, and (3) although they are related, we want to distinguish this term from cyclic Petri nets, which are nets constituting a single SCC.

Loops play an important role in the behavior of Petri nets. They are able to describe repeating actions as well as recovery points. For example, liveness is a useful property of Petri nets. It ensures that any transition can eventually fire from any reachable marking and requires that the entire Petri net is in a loop. For example, the net in Figure 1 is live. Home markings and clusters in cyclic or perpetual Petri nets enable a recovery. However, not all Petri nets are cyclic; some have loops with entries and exits. For those, it would be beneficial if the behavior of the loops can be investigated separately. An important step in this direction are *looms*, a short form of *loop home markings*. A loom is a marking of a loop that is “loop-sub-marking reachable” from any marking with a token on a place of the loop:

Definition 14 (Loom)**.** A *loom* of a loop $L = (P_L, T_L, F_L)$ of a marked net (N, M) is a marking $M_{loom} \subseteq \mathscr{B}(P_L)$ of L, for which it holds that:

$$\forall M' \in R(N,M) : \; M' \upharpoonright P_L \neq [\,] \implies \exists M'' \in R(N,M') : \; M'' \upharpoonright P_L = M_{loom}.$$

A loom of L can be described by a cluster C of L if $Mrk(C)$ is a loom of L. ⌟

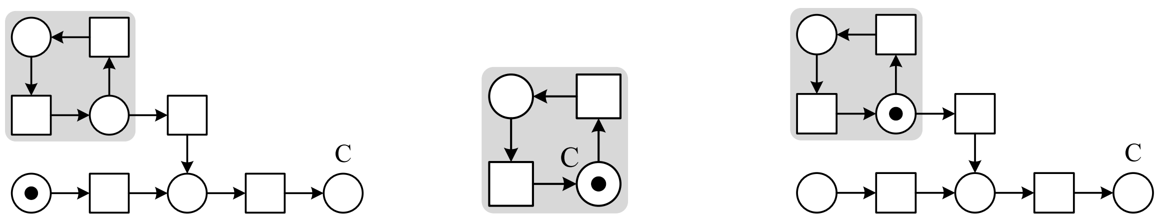
Note that each marking containing a loom requires to be reachable from itself. For example, the home cluster $C = \{ p_7, p_8, t_7 \}$ of the perpetual net in Figure 1 with $Mrk(C) = [\, p_7, p_8 \,]$ represents a loom of the loop covering the net. Figure 2 has a reachable marking $[\, p_1, p_3, p_{11} \,]$, where place p_3 as part of the loop L (light gray rounded rectangle) already contains a token. One loom of L is $[\, p_4 \,]$. As the reader can verify, $[\, p_4 \,]$ is "loop-sub-marking reachable" from $[\, p_1, p_3, p_{11} \,]$, from $[\, p_4, p_{11} \,]$, and from any other marking, in which some place of L contains a token. We can see that the concurrent token to this loop on p_{11} or p_{12} does not influence the behavior of the loop described by the loom $[\, p_4 \,]$. Note that $[\![p_4]\!]$ is a loom of L described by a cluster and that L has other looms such as $[\![p_7]\!] = \{ p_7, t_9, t_{14} \}$, $[\![p_8]\!] = \{ p_8, t_{10}, t_{15} \}$, and $[\![p_2]\!] = \{ p_2, p_3, t_3, t_4 \}$. Thus, similarly to a home marking, a loom does not need to be unique for each loop. But, unlike a home marking, once a token leaves a loop, the loom is not reachable anymore.

4. Loop Properties in Free-Choice Nets with a Home Cluster

The identification of looms is not trivial. Fortunately, loops within free-choice nets with a home cluster have some beneficial properties that will help us in Section 5 to identify looms in this class of nets.

4.1. Kinds of Loops

In the following, we will consider general kinds of loops. This helps to restrict the set of loops being of interest for the identification of looms. These general kinds of loops are presented in Figure 3. Assume a net (N, M) with a loop L. It may happen that there is no path from a place selected in M to any node of L (cf. Figure 3 (a)). Naturally, all nodes of this loop are dead in M as no token can reach them by any path.



(a) A net with a dead loop. (b) A net with a live loop. (c) A net with a loop that is not dead.

Figure 3: Three marked free-choice nets with (a) a dead loop, (b) a non-dead loop without loop exit (and entry), and (c) a non-dead loop with loop exit.

Definition 15 (Dead Loop)**.** A loop L of a marked proper free-choice net (N, M) with a home cluster is *dead* if at least one of its transitions is dead in (N, M). ⌟

Of course, the behavior of dead loops is not of interest. We will remove dead loops by applying Definition 12. This definition introduced the notion of cleaned nets, $clean(N, M)$. By cleaning a net (N, M), all nodes and flows are removed for which there is no path starting from any initially marked place in M. As a consequence, cleaning a net removes *dead* loops so that all the net's loops are *not dead*:

Lemma 4 (Loops are Not Dead in Cleaned Nets)**.** Let (N, M) be a cleaned marked proper free-choice net with a home cluster C. Each loop of (N, M) is not dead. ⌟

Proof. Following Definition 6.4 of Van der Aalst [4], a cleaned marked proper free-choice net (N, M) with $N = (P, T, F)$ and a home cluster C can be short-circuited at C by adding a fresh transition $t_c \notin T$ with $\bullet t_c = C \cap P$ and $t_c \bullet = \{ x \in M \}$. Thus, once the home cluster C is marked, the net is back in its initial marking. This short-circuited net is live and bounded and, thus, safe (see Theorem 6.6 in [4]). As a consequence, each node is live in such a net and, therefore, also each loop. Since inserting this transition does not change the set of reachable markings, all nodes and loops in the original net are not dead. ✓ □

Since dead loops are not of interest, we assume in the following that each considered net is “clean” or was cleaned before. Considering *cleaned* nets, there are two remaining, general kinds of loops:

(i) loops *without* loop exits (Figure 3 (b)) and

(ii) loops *with at least one* loop exit (Figure 3 (c)).

A loop *without* loop exit must contain the home cluster since once a token is within such a loop, the token is trapped (the loop is a *trap* [1, 2, 13]). On the other hand, if a loop *has at least one* loop exit, then this loop cannot contain a home cluster:

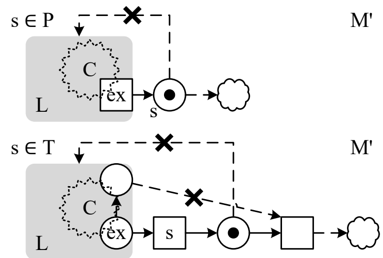
Lemma 5 (Loops with Loop Exits do not Contain a Home Cluster)**.** Let (N,M) be a cleaned marked proper free-choice net.

$$(N,M) \text{ has a home cluster } C \implies$$
$$\forall L = (P_L,T_L,F_L) \in Loops(N):\ \big(Exits(L) \neq \varnothing \implies (P_L \cup T_L) \cap C = \varnothing\big) \qquad \lrcorner$$

Proof. Let (N,M) be a cleaned marked proper free-choice net. The proof is done by contradiction:

$$(N,M) \text{ has a home cluster } C \ \wedge\ \exists L = (P_L,T_L,F_L) \in Loops(N):\ \big(Exits(L) \neq \varnothing \ \wedge\ (P_L \cup T_L) \cap C \neq \varnothing\big) \qquad (1)$$

Following this assumption (1), let (N,M) have a home cluster C, and let $L = (P_L,T_L,F_L) \in Loops(N)$ be a loop of (N,M) with $Exits(L) \neq \varnothing$ and $(P_L \cup T_L) \cap C \neq \varnothing$. By Definition 13 of loop exits, it holds: $\forall ex \in Exits(L)\ \exists s \in ex\bullet:\ s \notin P_L \cup T_L$. Let $ex \in Exits(L)$ be a loop exit with $s \in ex\bullet \setminus (P_L \cup T_L)$. Furthermore, by Definition 13 of loops as SCC, s does not have a path to any node of L. Thus, s does not have a path to $(P_L \cup T_L) \cap C$. Since L is not dead, it holds:



$$\exists M' \in R(N,M):\ s \in M'\ (s \in P)\ \ \vee\ s\bullet \subseteq M'\ (s \in T).$$

Let M' be such a reachable marking. There are exactly two cases:

$s \in P$: I. e., the loop exit ex is a transition ($ex \in T$). Since L is an SCC, the token on s in M' cannot turn back to L. Furthermore, by Definition 8 of clusters, s is not part of any cluster intersecting with L. Thus, $s \notin Mrk(C)$ and, consequently: $Mrk(C) \notin R(N,M')$. ↯

$s \in T$: I. e., $ex \in P$ and s was fired with $s\bullet \subseteq M'$. Since L is an SCC, $s\bullet \cap (P_L \cup T_L) = \varnothing$. Furthermore, if any transition of $(s\bullet)\bullet$ had an input place in L, then there would be a transition of $(s\bullet)\bullet$ being in L, too, because of N’s free-choiceness. However, this would imply that s would have a path back to L contradicting that L is an SCC or that ex is a loop exit with $s \notin P_L \cup T_L$. Therefore, no cluster containing any place of $s\bullet$ intersects L (i. e., even not the home cluster). Consequently: $Mrk(C) \notin R(N,M')$. ↯

By Definition 10, C is not a home cluster of (N,M). Thus, (1) is a contradiction. and Lemma 5 holds. ✓ □

If a loop *has a loop exit*, the loop has a non-empty *exit area* (regarding a cluster C by Definition 13). Remember, such an exit area contains all places and transitions being on a C-rooted disentangled path starting “behind” the loop. Now, assume a marking with tokens in the loop and in the loop’s exit area. Since there are paths, we can imagine that the tokens within the loop could get to the tokens in the exit area—the net is unsafe. Thus, the exit area should be free of tokens as long as there is a token in the loop. This holds for the net in Figure 2. For the net in Figure 4, the area after the loop is not empty if t_4 fires.

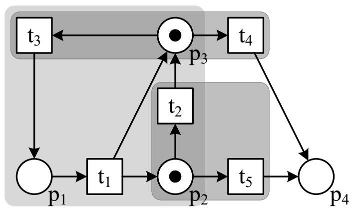


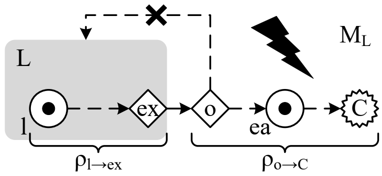
Figure 4: A loop with two exits p_2 and p_3. If t_4 fires, p_2 inside and p_4 outside the loop are marked. Furthermore, the loop exit clusters containing p_2 and p_3 are not mutually exclusive.

Theorem 1 (Token-Free Exit Area)**.** Let (N,M) be a cleaned marked proper free-choice net $N = (P,T,F)$ and C is a cluster.

$$(N,M) \text{ has } C \text{ as a home cluster } \implies$$
$$\forall L = (P_L,T_L,F_L) \in Loops(N)\ \forall M_L \in R(N,M):\ M_L \restriction P_L \neq [\,] \implies M_L \restriction ExitArea_N(L,C) = [\,] \qquad \lrcorner$$

Proof. Let (N,M) be a cleaned marked proper free-choice net $N = (P,T,F)$ and C is a cluster. This proof is done by contradiction:

$$(N,M) \text{ has } C \text{ as a home cluster } \wedge \tag{2}$$
$$\exists L = (P_L, T_L, F_L) \in Loops(N)\ \exists M_L \in R(N,M)\colon\ M_L \upharpoonright P_L \neq [\,] \ \wedge\ M_L \upharpoonright ExitArea_N(L,C) \neq [\,]$$



By this assumption, let $L = (P_L, T_L, F_L) \in Loops(N)$ with $M_L \in R(N,M)$ so that $M_L \upharpoonright P_L \supseteq [l] \neq [\,]$ and $M_L \upharpoonright ExitArea_N(L,C) \supseteq [ea] \neq [\,]$. It directly follows from $ExitArea_N(L,C) \supseteq \{ea\}$ by Definition 13 that there is an output o of a loop exit ex being in the exit area of L (regarding C):

$$\exists o \in \big(Exits_N(L)\bullet \backslash (P_L \cup T_L)\big)\ \exists \rho_{o\to C} \in \mathscr{P}^C_N(o)\colon\ ea \in \langle \rho_{o\to C}\rangle.$$

Let $ex \in Exits_N(L)$ and $o \in ex\bullet \backslash (P_L \cup T_L)$ be such nodes and $\rho_{o\to C} \in \mathscr{P}^C_N(o)$ be such a C-rooted disentangled path for which it holds that $ea \in \langle \rho_{o\to C}\rangle$. By Definition 13 of loops, $\rho_{o\to C}$ is outside of L: $\langle \rho_{o\to C}\rangle \cap (P_L \cup T_L) = \varnothing$. By Definition 13 of loops and Lemma 2, there is a $[\![ex]\!]$-rooted disentangled path $\rho_{l\to ex}$ from l to ex in L. Since $\langle \rho_{o\to C}\rangle \cap (P_L \cup T_L) = \varnothing$, it follows that: $\langle \rho_{l\to ex}\rangle \cap \langle \rho_{o\to C}\rangle = \varnothing$. Therefore, their concatenation is also a C-rooted disentangled path: $\rho = \rho_{l\to ex} \cdot \big((ex) \cdot \rho_{o\to C}\big) \ \wedge\ \rho \in \mathscr{P}^C_N(l)$. By assumption (2), it follows: $|M_L \upharpoonright \langle \rho \rangle| \geqslant |[l, ea]| \geqslant 2$. This contradicts Lemma 3 that each C-rooted disentangled path is safe in (N,M). Therefore, (N,M) cannot have C as a home cluster. $\lightning$ (2) is a contradiction. For this reason, Theorem 1 must hold. ✓ □

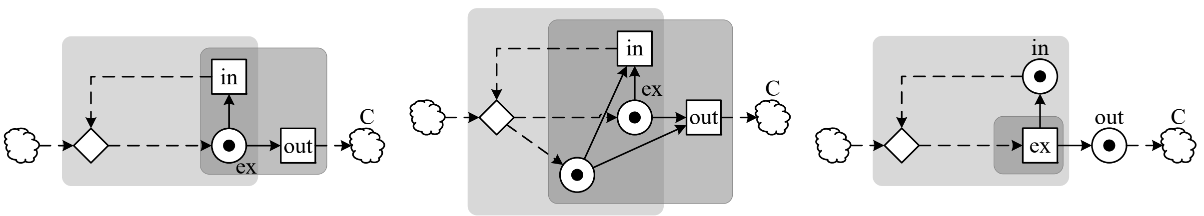
This leads us to the first finding in simplified terms:

Finding 1. *As long as there is a token in a loop in a cleaned marked proper free-choice net with a home cluster, then all places in the loop's exit area are unmarked.*

4.2. Properties of Loop Exits

Arguing about the markings related to an exit-less loop is more easily, as it intersects the home cluster by Lemma 5. Thus, we have the home cluster as pre-defined "recovery point" that is reachable from all other reachable markings. A loop *with* loop exits *cannot* contain a home cluster by Lemma 5. The behavior of such loops in terms of markings is more sophisticated. For this reason, the remainder of this section focuses on loops with loop exits.

If a loop has a loop exit, then this exit has at least two outputs by Definition 13 of loops: One within and one outside the loop. Furthermore, a loop exit can be within a cluster consisting of multiple places and transitions. Figure 5 compares three available kinds of loop exits (where loop entries are illustrated as diamonds as they are not yet considered): (a) a place being in a cluster containing only this place and its output transitions, (b) some places in a cluster with their output transitions, and (c) a transition. Case (c) may lead to an unbounded number of tokens after the loop whereas cases (a) and (b) should work as expected for "classic" loops. This seems intuitive since the concurrency after firing a loop exit transition cannot be joined within the loop, therefore, probably causing unsafeness outside the loop.



(a) A single place with its output transitions as loop exit.

(b) A cluster of multiple places and transitions as loop exit.

(c) A transition with its output places as loop exit.

Figure 5: Three abstract loops (a), (b), and (c) with different kinds of loop exits. The entries are visualized as diamonds as their type (place or transition) is not yet considered.

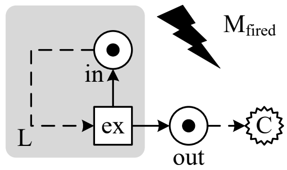
The following theorem formally shows that loop exits have to be places:

Theorem 2 (Loop Exits are Places)**.** Let (N,M) be a cleaned marked proper free-choice net $N = (P,T,F)$. If (N,M) has a home cluster, then all loop exits are places:

$$(N,M) \text{ has a home cluster } C \implies \forall L \in Loops(N)\ \forall ex \in Exits(L)\colon\ ex \in P$$ ⌟

Proof. Let (N,M) be a cleaned marked proper free-choice net $N = (P,T,F)$. We consider the contradiction:

$$(N,M) \text{ has a home cluster } C \;\wedge\; \exists L \in \mathit{Loops}(N)\; \exists ex \in \mathit{Exits}(L):\; ex \in T \qquad (3)$$



By this assumption, we start with (N,M) has a home cluster C. Furthermore, let L be one of N's loops with $L = (T_L, P_L, F_L) \in \mathit{Loops}(N)$ that has a loop exit $ex \in \mathit{Exits}(L)$ with $ex \in T$. By Definition 13 of loops, it follows that one output place *in* of *ex* is within L and one output place *out* of *ex* is outside of L:

$$\exists\{\,in, out\,\} \subseteq ex\bullet:\; in \in P_L \;\wedge\; out \notin P_L.$$

Since L has at least one loop exit, it follows by Lemma 5 that the home cluster C is not part of L, and, therefore, it holds by Definition 13 of exit areas of loops: $out \in \mathit{ExitArea}_N(L,C)$. As (N,M) has C as a home cluster and L is not dead by Lemma 4, there is a reachable marking M' from M, in which $[\![ex]\!]$ is enabled: $\exists M_{ex} \in R(N,M):\; \mathit{Mrk}([\![ex]\!]) \subseteq M_{ex}$. Let M_{ex} be such a marking. Since $[\![ex]\!]$ is enabled in M_{ex}, it can be fired such that *in* and *out* have tokens: $\exists M_{fired} \in R(N,M_{ex}):\; [\,in, out\,] \subseteq M_{fired}$. Thus, it follows directly: $\exists M_{fired} \in R(N,M_{ex}):\; M_{fired} \restriction P_L \supseteq [\,in\,] \;\wedge\; M_{fired} \restriction \mathit{ExitArea}_N(L,C) \supseteq [\,out\,]$. This contradicts Theorem 1. Thus, (N,M) cannot have C as a home cluster. $\lightning$ (3) is a contradiction. For this reason, Theorem 2 must hold. ✓ □

This theorem leads to the second finding in simplified terms:

> **Finding 2.** *Each loop exit of a loop in a cleaned marked proper free-choice net with a home cluster is a place.*

For simplicity, we call each cluster that contains a loop exit a *loop exit cluster* in the following. If a loop exit cluster fires while there remain tokens within the loop, then this immediately violates Theorem 1 that the loop's exit area must be empty. As a consequence, once all places of a loop exit cluster are marked, then no other place of the same loop can have a token in the same marking. Note that in loops *without* exits, the home cluster fulfills the same property: if the home cluster is marked, then the loops are free of tokens except the tokens in this cluster.

The net in Figure 4 contains a loop highlighted with a light gray rounded rectangle. The loop exits p_2 and p_3 with their respective clusters are highlighted with dark gray rounded rectangles. Both exit clusters are enabled at the same time. If t_4 fires, p_4 gets a token while the token on p_3 remains. Consequently, the net cannot have a home cluster. This is formulated and shown by the following theorem:

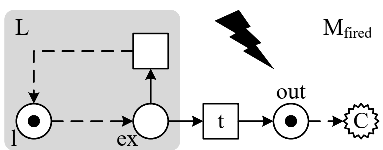
Corollary 3 (Token-Free Loops)**.** Let (N,M) be a cleaned marked proper free-choice net $N = (P,T,F)$. If (N,M) has a home cluster, then in each reachable marking, in which a loop exit cluster is marked, the corresponding loop is free of tokens except the tokens in this cluster:

$$\begin{gathered}(N,M) \text{ has a home cluster } C \implies \\ \forall L = (P_L, T_L, F_L) \in \mathit{Loops}(N)\; \forall ex \in \mathit{Exits}(L)\; \forall M_{ex} \in R(N,M): \\ \big(\mathit{Mrk}([\![ex]\!]) \subseteq M_{ex} \implies M_{ex} \restriction P_L = \mathit{Mrk}([\![ex]\!])\big)\end{gathered}$$

⌟

Proof. Let (N,M) be a cleaned marked proper free-choice net $N = (P,T,F)$. We consider the contradiction:

$$\begin{gathered}(N,M) \text{ has a home cluster } C \;\wedge \qquad (4)\\ \exists L = (P_L, T_L, F_L) \in \mathit{Loops}(N)\; \exists ex \in \mathit{Exits}(L)\; \exists M_{ex} \in R(N,M):\; \big(\mathit{Mrk}([\![ex]\!]) \subseteq M_{ex} \;\wedge\; M_{ex} \restriction P_L \neq \mathit{Mrk}([\![ex]\!])\big)\end{gathered}$$



By this assumption, let $L = (P_L, T_L, F_L) \in \mathit{Loops}(N)$ with $ex \in \mathit{Exits}(L)$ and $M_{ex} \in R(N,M)$ so that $\mathit{Mrk}([\![ex]\!]) \subseteq M_{ex}$ and thus $M_{ex} \restriction P_L \supseteq [\,l\,] \uplus \mathit{Mrk}([\![ex]\!])$. Since (N,M) is cleaned proper free-choice and has home cluster C, (N,M) is safe by Lemma 1 and, thus, $l \notin [\![ex]\!]$ and $ex \in P$ (by assumption (4) and Theorem 2). In M_{ex}, $[\![ex]\!]$ is enabled. Thus, a transition $t \in ([\![ex]\!] \cap T) \backslash T_L$ (outside of L) can be fired in M_{ex}: $\exists M_{fired} \in R(N,M):\; [\,l\,] \uplus t\bullet \subseteq M_{fired}$.

Let $M_{fired} \supseteq [\,l\,] \uplus t\bullet$ be such a marking with $out \in t\bullet \backslash P_L$. Since L has at least one loop exit, it follows by Lemma 5 that the home cluster C is not part of L, and, therefore, it holds by Definition 13 of exit areas of loops: $out \in \mathit{ExitArea}_N(L,C)$. Thus it follows directly: $\exists M' \in R(N,M):\; M' \restriction P_L \supseteq [\,l\,] \;\wedge\; M' \restriction \mathit{ExitArea}_N(L,C) \supseteq [\,out\,]$. This contradicts Theorem 1. Therefore, (N,M) cannot have C as a home cluster. $\lightning$ (4) is a contradiction. For this reason, Corollary 3 must hold. ✓ □

This leads us to the third finding in simplified terms:

> **Finding 3.** *If a loop exit cluster in a cleaned marked proper free-choice net with a home cluster is fully marked, the rest of the corresponding loop is unmarked.*

Following from this finding, if the tokens enabling a loop exit cluster of a loop are the only remaining tokens in this loop, then all loop exit clusters of the same loop must be mutually exclusive. A loop L as a black box eventually fires a single transition of $Exits(L)\bullet$: L behaves similarly to a single place.

The net in Figure 2 has three loop exit clusters $[\![p_4]\!]$, $[\![p_7]\!]$, and $[\![p_8]\!]$, which are mutually exclusive. As the reader can confirm, only one transition of those clusters finally fires. Thus, they are mutually exclusive as well. Contrarily, in Figure 4, the loop exit clusters $[\![p_2]\!]$ and $[\![p_3]\!]$ are not mutually exclusive. As a result, this net cannot have a home cluster. This is shown by the following theorem:

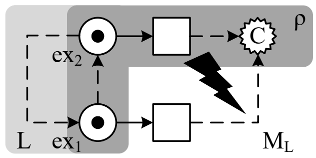
Theorem 4 (Loop Exit Clusters are Mutually Exclusive)**.** Let (N,M) be a cleaned marked proper free-choice net $N=(P,T,F)$. If (N,M) has a home cluster, then all loop exit clusters of a loop are mutually exclusive:

$$(N,M) \text{ has a home cluster } C \implies$$
$$\forall L \in Loops(N)\ \forall M_L \in R(N,M)\ \forall \{ex_1, ex_2\} \subseteq Exits(L), [\![ex_1]\!] \neq [\![ex_2]\!]:\ ex_1 \in M_L \implies ex_2 \notin M_L \quad \lrcorner$$

Proof. Let (N,M) be a cleaned marked proper free-choice net $N=(P,T,F)$. We prove by contradiction:

$$(N,M) \text{ has a home cluster } C \wedge$$
$$\exists L \in Loops(N)\ \exists M_L \in R(N,M)\ \exists \{ex_1, ex_2\} \subseteq Exits(L), [\![ex_1]\!] \neq [\![ex_2]\!]:\quad ex_1 \in M_L \wedge ex_2 \in M_L \tag{5}$$

Following this assumption, let $L \in Loops(N)$ and $M_L \in R(N,M)$ with $\{ex_1, ex_2\} \subseteq Exits(L)$, $[\![ex_1]\!] \neq [\![ex_2]\!]$, so that $[ex_1, ex_2] \subseteq M_L$. Since (N,M) has C as a home cluster by assumption (5), it follows by Theorem 2 and Definition 8 of clusters: $\{ex_1, ex_2\} \subseteq P \wedge ex_1\bullet \subseteq [\![ex_1]\!] \wedge ex_2\bullet \subseteq [\![ex_2]\!]$. By assumption (5), $[\![ex_1]\!] \neq [\![ex_2]\!]$, and by Definition 13 of loops and Lemma 2, there is a $[\![ex_2]\!]$-rooted disentangled path $\rho_{ex_1 \to ex_2}$ from ex_1 to ex_2: Furthermore, since L has loop exits ex_1 and ex_2, C cannot be part of L by Lemma 5. Thus, by Lemma 2, there is a C-rooted disentangled path from ex_2 outside of L (except ex_2):



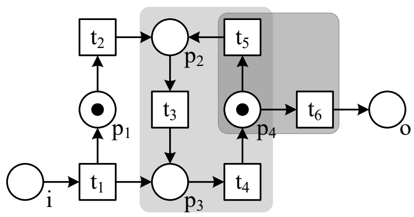
$$\mathscr{P}_N^C(ex_2) \neq \varnothing \wedge \rho_{ex_2 \to C} \in \mathscr{P}_N^C(ex_2) \wedge \langle \rho_{ex_2 \to C} \rangle \cap (P_L \cup T_L) = \{ex_2\}.$$

Since L is an SCC by Definition 13 of loops, L is maximal and, therefore, $\rho_{ex_1 \to ex_2}$ does not overlap with $\rho_{ex_2 \to C}$ (except ex_2): $\langle \rho_{ex_1 \to ex_2} \rangle \cap \langle \rho_{ex_2 \to C} \rangle = \{ex_2\}$. Their concatenation is, therefore, also a C-rooted disentangled path: $\rho = \rho_{ex_1 \to ex_2} \cdot \rho_{ex_2 \to C} \wedge \rho \in \mathscr{P}_N^C(ex_1)$. By assumption (5), it follows: $|M_L \upharpoonright \langle \rho \rangle| \geqslant |[ex_1, ex_2]| \geqslant 2$. This contradicts Lemma 3 that each C-rooted disentangled path of (N,M) is safe. Therefore, there is no reachable marking, in which both ex_1 and ex_2 have tokens. $\lightning$ (5) is a contradiction. Thus, Theorem 4 must hold. ✓ □

This leads us to the fourth finding in simplified terms:

> **Finding 4.** *All loop exit clusters of a loop in a cleaned marked proper free-choice net with a home cluster are mutually exclusive.*

The findings presented previously have implications for all markings regarding a loop when a loop exit cluster is fully marked. Furthermore, we know by Theorem 1 that the *exit area* of a loop is free of tokens as long as there is a token within the loop. Now, we investigate the area *in front* of loops. When a loop has an enabled loop exit cluster, then it should be impossible that additional tokens can enter the loop. Otherwise, this would violate Corollary 3 that the loop is unmarked except the exit cluster. Consequently, the area in front of the loops must be unmarked in such situations. In the net of Figure 2, transitions t_3 and t_4 (with their common cluster) collect all tokens coming from outside the loop and synchronize them before reaching any loop exit. Contrarily, the free-choice workflow net on the right has a reachable marking from $[i]$, in which the loop (highlighted in light gray) has a token on the loop exit cluster of p_4. However, the "entrance area" is not empty because of the token on p_1 that can enter the loop at p_2. This net cannot have a home cluster as the following corollary states:

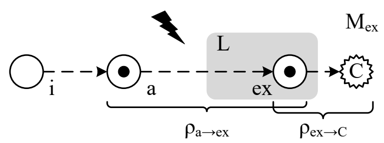


Corollary 5 (Token-Free Entrance Area)**.** Let (N,M) be a cleaned marked proper free-choice net $N = (P,T,F)$. If (N,M) has a home cluster, then, once a loop exit of a loop is marked, all paths from places with tokens in the initial marking to this loop exit are free of tokens:

$$\begin{gathered}(N,M) \text{ has a home cluster } C \implies \\ \forall L \in \mathit{Loops}(N)\ \forall ex \in \mathit{Exits}(L)\ \forall M_{ex} \in R(N,M),\, ex \in M_{ex} \\ \forall i \in M\ \forall \rho_{i\to ex} \in \mathscr{P}_N^R(i,ex)\colon\ M_{ex} \upharpoonright \langle \rho_{i\to ex} \rangle = [ex]\end{gathered}$$ ⌟

Proof. The proof is done by contradiction for a cleaned marked proper free-choice net (N,M) with $N = (P,T,F)$, i. e., there is a C-rooted disentangled path $\rho_{i\to ex}$ from an initially marked place i to a marked loop exit ex with more tokens than that on ex:

$$\begin{gathered}(N,M) \text{ has a home cluster } C \wedge \\ \exists L \in \mathit{Loops}(N)\ \exists ex \in \mathit{Exits}(L)\ \exists M_{ex} \in R(N,M),\, ex \in M_{ex}\ \exists i \in M\ \exists \rho_{i\to ex} \in \mathscr{P}_N^R(i,ex)\colon\ M_{ex} \upharpoonright \langle \rho_{i\to ex} \rangle \neq [ex].\end{gathered} \tag{6}$$



Following from assumption (6), let $L = (P_L, T_L, F_L) \in \mathit{Loops}(N)$ with $ex \in \mathit{Exits}(L)$, $M_{ex} \in R(N,M)$ with $ex \in M_{ex}$, $i \in M$, and $\rho_{i\to ex} \in \mathscr{P}_N^R(i,ex)$ with $M_{ex} \upharpoonright \langle \rho_{i\to ex} \rangle \neq [ex]$. Since $ex \in M_{ex}$ and $M_{ex} \upharpoonright \langle \rho_{i\to ex} \rangle \neq [ex]$, it must hold: $|M_{ex}| \geqslant 2 \wedge [ex] \subset (M_{ex} \upharpoonright \langle \rho_{i\to ex} \rangle)$. For this reason, there has to be a further place a on $\rho_{i\to ex}$ with a token in M_{ex}: $\exists a \in \rho_{i\to ex}\colon\ a \neq ex \wedge a \in M_{ex}$. Let a be such a place. Since $a \in \rho_{i\to ex}$, there is a subpath $\rho_{a\to ex}$ of $\rho_{i\to ex}$ from a to ex (which is also a $[\![ex]\!]$-rooted disentangled path). Since L has a loop exit, C cannot be part of L by Lemma 5. Thus, by Lemma 2, there is a C-rooted disentangled path $\rho_{ex\to C}$ from ex outside of L (except ex): $\mathscr{P}_N^C(ex) \neq \varnothing \wedge \rho_{ex\to C} \in \mathscr{P}_N^C(ex) \wedge \langle \rho_{ex\to C} \rangle \cap P_L = \{ex\}$. Since L is an SCC by Definition 13 of loops and, therefore, maximal, $\rho_{a\to ex}$ does not overlap with $\rho_{ex\to C}$ (except ex): $\langle \rho_{a\to ex} \rangle \cap \langle \rho_{ex\to C} \rangle = \{ex\}$. Their concatenation is therefore also a C-rooted disentangled path: $\rho = \rho_{a\to ex} \cdot \rho_{ex\to C} \wedge \rho \in \mathscr{P}_N^C(ex)$. By assumption (6), it follows: $|M_{ex} \upharpoonright \langle \rho \rangle| \geqslant |[a,ex]| \geqslant 2$. This contradicts Lemma 3 that the C-rooted disentangled path ρ is safe in (N,M) with home cluster C. Therefore, (N,M) cannot have C as a home cluster. ↯ (6) is a contradiction. For this reason, Corollary 5 must hold. ✓ □

This leads us to the fifth finding in simplified terms:

Finding 5. *When a loop exit in a cleaned marked proper free-choice net with a home cluster is marked, the area in front of this loop is free of tokens.*

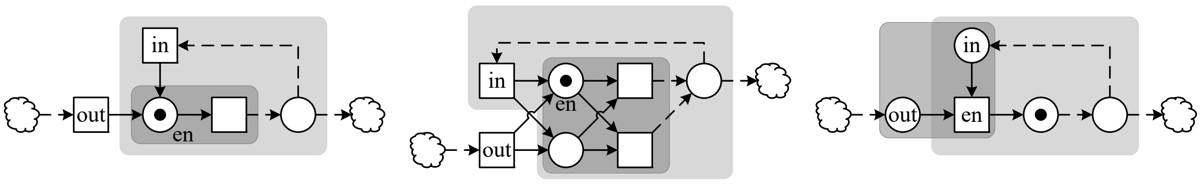
Summarized, all previous findings allow us to precisely describe markings in which a loop's exit cluster is enabled: The loop, the area in front of the loop, and the exit area of a loop are unmarked except the marked exit cluster. Thus, a loop as a black box behaves similar to a single place for the subnet being connected with the loop's exits.

4.3. Property of Loop Entries

Loop entries serve as ways into loops but once a token is within a loop, the loop entries are like any other place or transition within the loop. Corollary 5 states that once a loop exit cluster is enabled, no additional token may enter the loop through a loop entry. However, the difficulty is that as long as no loop exit cluster is enabled, additional tokens might still enter the loop. Furthermore, a non-dead loop must not even have any loop exit since, by Lemma 5, the home cluster is then at least partially within this loop. However, in such a situation, the home cluster replaces the missing loop exit clusters.

Similarly to loop exits, loop entries can either be places or transitions as illustrated in Figure 6. The figure visualizes abstractly three available kinds of loop entries: (a) a single place as loop entry seeming valid, (b) a free-choice cluster (sometimes called a "bridge") as loop entry with multiple places seeming also valid, and (c) a transition as loop entry, which may be dead. The latter case should not be possible in a well behaved model: if an entry is a transition, it has at least an input place outside of and at least an input place within the loop. Thus, it either could not get tokens from inside the loop when the loop is entered. Or, it could not get tokens from outside the loop when the loop iterates. Both cases lead to a dead entry transition.

Figure 7 (a) shows a free-choice workflow net with a loop that has two loop entries with one of them being a transition, t_2. If the loop is entered, t_2 can fire using the tokens created by t_1. However, once



(a) A single place as loop entry. (b) A cluster of multiple places as loop entry. (c) A transition as loop entry.

Figure 6: Three abstract loops (a), (b), and (c) with different kinds of loop entries.

(a) The transition t_2 as loop entry … (b) …and its attempt to repair it.

Figure 7: A loop (a) with a transition t_2 as loop entry and (b) its attempt to "repair" it with an additional flow (t_3, p_1). However, this repair moves p_1 into the loop such that p_1 and p_2 are loop entries.

transition t_3 fires, t_2 is dead. One can try to repair the situation by introducing a new flow from t_3 to p_1 to give both input places of t_3 tokens (cf. Figure 7 (b), a "bridge"). However, the introduction of this bridge changes the loop entries: since the loop was extended by the new flow (t_3, p_1), p_1 and p_2 are loop entries now—all loop entries are places and the repaired net has a home cluster $[\,o\,]$.

Theorem 6 (Loop Entries)**.** Let (N,M) be a cleaned marked proper free-choice net $N = (P,T,F)$. If (N,M) has a home cluster, then all loop entries are places:

$$(N,M) \text{ has a home cluster } C \implies \forall L \in \mathit{Loops}(N)\ \forall en \in \mathit{Entries}(L):\ en \in P$$

⌟

Proof. Let (N,M) be a cleaned marked proper free-choice net $N = (P,T,F)$. We prove by contradiction:

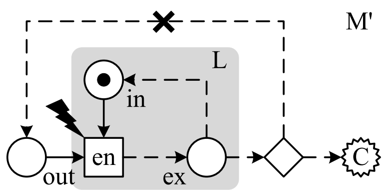
$$N \text{ has a home cluster } C\ \wedge\ \exists L \in \mathit{Loops}(N)\ \exists en \in \mathit{Entries}(L):\ en \in T. \tag{7}$$

By this assumption (7), let $L = (P_L, T_L, F_L) \in \mathit{Loops}(N)$ be a loop with $en \in \mathit{Entries}(L)$ and $en \in T$. It follows from Definition 13 of loops: $|\bullet en| \geqslant 2$. Let $out \in \bullet en \backslash P_L$ be an input place of en from outside of L, and $in \in \bullet en \cap P_L$ be an input place of en from inside of L (by Definition 13). Furthermore, let $ex \in \mathit{Exits}(L)$ be a loop exit of L. If $\mathit{Exits}(L) = \varnothing$, then let $ex \in (C \cap P)$. Following from (N,M) has a home cluster C by assumption (7) and by Theorem 2: $ex \in P$. Following also from (N,M) has a home cluster C by assumption (7) and that L is not dead by Lemma 4: $\exists M_{ex} \in R(N,M):\ \mathit{Mrk}([\![ex]\!]) \subseteq M_{ex}$. Let M_{ex} be such a marking. In M_{ex}, all places of L except that of $[\![ex]\!]$ do not have tokens by Corollary 3 or by assumption (7) that C is a home cluster: $M_{ex} \upharpoonright (P_L \backslash [\![ex]\!]) = [\,]$. Furthermore, following from Corollary 5 and C is a home cluster, the area before the loop is also free of tokens:

$$\forall i \in M\ \forall \rho_{i\to ex} \in \mathscr{P}_N^R(i, ex):\ M_{ex} \upharpoonright \langle \rho_{i\to ex} \rangle = [ex].$$

Therefore, *out* does not get a token in any reachable marking M' from M_{ex}:

$$\forall M' \in R(N, M_{ex}):\ out \notin M'. \tag{8}$$



Since L is an SCC by Definition 13 of loops, it is maximal. By Lemma 2, there is a disentangled path from ex to in (if $in \in [\![ex]\!]$, then $in = ex$). As (N,M) has C as home cluster by assumption (7), C is not in L since $\mathit{Exits}(L) \neq \varnothing$ by Lemma 5, and L is not dead by Lemma 4:

$$\exists M' \in R(N, M_{ex}):\ in \in M'. \tag{9}$$

It follows from (8) and (9): $\exists M' \in R(N, M_{ex})\ \forall M'' \in R(N, M'):\ in \in M'\ \wedge\ out \notin M''$. By Definition 5, en is dead in M' but not a home cluster. Following from Definition 10, C is not a home cluster of (N,M). ↯ (7) is a contradiction. Theorem 6 must hold. □

The last theorem leads to our sixth finding in simplified terms:

Finding 6. *Each loop entry of a loop in a cleaned marked proper free-choice net with a home cluster is a place.*

This section proved that loop entries and loop exits are places in marked free-choice nets with a home cluster. Furthermore, it unveiled crucial properties about markings in which loop exit clusters have tokens. Based on these properties, loops must have a kind of "loop-related home marking"—a *loom*.

5. Looms in Free-Choice Nets with a Home Cluster

In the previous section, we derived several properties of loops in cleaned marked proper free-choice nets with a home cluster. These properties eventually help to specify and, thus, identify looms in such nets. It turns out that each loop exit cluster of a loop describes a loom or, if such a loop does not have any exit, the home cluster describes a loom of the loop:

Theorem 7 (Loop Exit Clusters are Looms)**.** Let (N,M) be a cleaned marked proper free-choice net $N=(P,T,F)$. If (N,M) has a home cluster C, then each loop L has C or all its loop exit clusters as looms:

$$(N,M) \text{ has a home cluster } C \implies$$
$$\forall L \in \mathit{Loops}(N):\ C \text{ is a loom cluster of } L \quad \vee \quad \forall ex \in \mathit{Exits}(L):\ [\![ex]\!] \text{ is a loom (cluster) of } L \quad ⌟$$

Proof. Let (N,M) be a cleaned marked proper free-choice net $N=(P,T,F)$ with a home cluster C and a loop $L=(P_L,T_L,F_L) \in \mathit{Loops}(N)$. There are exactly two cases for L:

Case 1: $\mathit{Exits}(L)=\varnothing$. Since (N,M) has a home cluster C and is cleaned, L is not dead by Lemma 4. As $\mathit{Exits}(L)=\varnothing$ (L cannot be left) and L is not dead, there must exist a reachable marking M' such that in every marking M'' being reachable from M' a token is on a place of L:

$$\exists M' \in R(N,M)\ \forall M'' \in R(N,M'):\ M' \restriction P_L \neq [\,] \ \wedge\ M'' \restriction P_L \neq [\,].$$

Since $\mathit{Mrk}(C)$ as home cluster must be reachable from each such M', it must follow that:

$$\mathit{Exits}(L)=\varnothing \implies P_L \cap C \neq \varnothing.$$

We will show that C is entirely in L: We know by Theorem 6 that all loop entries of L are places. For this reason, it follows: $\forall t \in T_L:\ \bullet t \subseteq P_L$. From this and $\mathit{Exits}(L)=\varnothing$, it follows that: $\forall Cl \in [\![L]\!]:\ Cl \subseteq P_L \cup T_L$. Thus, all places of the home cluster C are within P_L: $P_L \cap C = \mathit{Mrk}(C)$. Furthermore, by Definition 10 of a home cluster, $\mathit{Mrk}(C)$ is reachable from any marking being reachable from M. Thus, it holds that:

$$\forall M' \in R(N,M):\quad M' \restriction P_L \neq [\,] \implies \mathit{Mrk}(C) \in R(N,M')\ \wedge\ \mathit{Mrk}(C) \restriction P_L = \mathit{Mrk}(C).$$

By Definition 14, C is a loom of L. ✓

Case 2: $\mathit{Exits}(L) \neq \varnothing$. Let $ex \in \mathit{Exits}(L)$ be *any* loop exit (being a place by Theorem 2). Since (N,M) has a home cluster and L has a loop exit, it follows by Lemma 5: $(P_L \cup T_L) \cap C = \varnothing$. For this reason, it follows: $\forall M_L \in R(N,M):\ M_L \restriction P_L \neq [\,] \implies \mathit{Mrk}(C) \in R(N,M_L)$. As a consequence, L "must be left by any loop exit" such as ex and it holds:

$$\forall M_L \in R(N,M):\ M_L \restriction P_L \neq [\,] \implies \big(\forall ex \in \mathit{Exits}(L)\, \exists M_{ex} \in R(N,M_L):\ ([\![ex]\!] \cap P_L) \subseteq M_{ex}\big).$$

Let M_{ex} be such a marking for a loop exit ex. By Corollary 3, it follows for any such M_{ex} in the free-choice net (N,M) with a home cluster C: $\forall ex \in \mathit{Exits}(L):\ M_{ex} \restriction P_L = [\![ex]\!] \cap P_L$; and, therefore:

$$\forall M_L \in R(N,M):\ M_L \restriction P_L \neq [\,] \implies \big(\forall ex \in \mathit{Exits}(L)\, \exists M_{ex} \in R(N,M_L):\ (M_{ex} \restriction P_L = [\![ex]\!]\big).$$

By Definition 14, for all $ex \in \mathit{Exits}(L)$, $[\![ex]\!]$ is a loom. ✓

Both cases for L hold. Thus, either the home cluster C or each loop exit cluster of L describes a loom. ✓ □

This leads us to our last finding in simplified terms:

Finding 7. *The home cluster or each loop exit cluster of a loop in a cleaned marked proper free-choice net with a home cluster is a loom.*

The above finding implies that detecting a loom of cleaned free-choice nets with a home cluster is possible by considering the loop exits of each loop. Thus, the algorithm of Tarjan for detecting SCCs [14] and investigating each place of a loop, whether it is a loop exit, enable a linear time detection of looms.

Knowing the looms of a net is beneficial for understanding the behavior of its loops. Consider a loop L with a loom λ: Once a marking is reached that contains λ, the behavior of L can be substituted in the state space, thus, leading to the state space's reduction. Such a loop L with a loom λ can also be separated and investigated independently as marked proper free-choice net (L,λ) (as illustrated in Figure 8 (a) for the loop in Figure 2). In doing this, this so-called *marked loop net* consists of the loop L with the loom λ as initial marking. The net is free-choice since L contains all places of each cluster following the proof of Theorem 7. Moreover, since the loom in the original net describes its re-occurring behavior (starting and ending in the loom), such marked loop net is live and safe with λ as a home cluster.

(a) The separated loop of Figure 2 as a live and safe marked loop net. It contains a nested loop regarding flow (t_8, p_7) following Definition 13 of nested loops.

(b) Separating the loom cluster $[\![p_7]\!]$ into a sink p_7' and a marked source p_7. This allows for investigating a loop like a workflow net.

Figure 8: Figure (a) presents the marked loop net of the loop in Figure 2 regarding the loom cluster $[\![p_7]\!]$ as initial marking. This allows for investigating the loop separated from the original net. By taking the *incoming* flow (t_8, p_7) of the selected place of the loom $[\,p_7\,]$ in (b) unveils the nested loop $\{\,p_4, t_5, p_5, t_6, p_6, t_7\,\}$ with entry p_4 and exit p_5.

A loom can unveil "nested loops" such as in the loop of Figure 2 being highlighted in Figure 8 (a) and (b) with a dark gray rounded rectangle. Detecting a nested loop is not trivial since a loop is an SCC (and therefore maximal) while the nested loop is hidden within an SCC. *Consequently, there cannot be a definition of nested loops without focusing on any kind of subgraph of the loop.* However, since we know that a loom defines the home marking of a marked loop net, we can imagine to split up the loop net at the loom. This can be achieved by separating the marked places of the loom into source and sink places (e. g., p_7 in Figure 8 (b) is separated into p_7' and p_7). Naturally, the loop net starts at the source places within an initial marking and ends in the sink places as the final marking where both markings represent the loom. This splitting up of the loop "destroys" the previous SCC and searching SCCs in this net reveals any new loops if they are present. Regarding Definition 13 of loops, this is equal to having a nested loop regarding the set of incoming flows of all places marked in the loom (cf. Figure 8 (a)).

An important property of nested loops is that they again have own looms since the surrounding loop net is a cleaned marked proper free-choice net with the loom cluster as a home cluster. *This means that we can investigate looms recursively for all nested loops.*

Note that nested loops may not be unique: If a loop has multiple looms, then taking different looms may result in slightly different or even no nested loops. For example, taking loom $[\,p_4\,]$ in Figure 2 would not unveil the nested loop. Since the algorithm for identifying SCCs by Tarjan [14] is linear in its computational complexity, it is feasible in quadratic time complexity to investigate all looms (as loop exit clusters) and their nested loops. Depending on the purpose of the analysis, we can either take a cluster with the lowest number of nested loops (to quickly achieve an acyclic loop net) or the cluster with the highest number of nested loops (to describe the behavior of the net in as much detail as possible). *In compiler theory, it is common to have different approaches to identify nested loops; this is not a drawback of any particular method, but rather a matter of choice depending on the specific application [15].*

Another benefit of looms is that they enable to *unroll* loops. Unrolling loops inserts duplicates of a loop to make its iterations explicit, but it requires that a loop has some kind of "recovery point". Without such a point, inserting duplicates may change the behavior of the net. Fortunately, looms are such recovery points. Consequently, we can safely unroll a loop at a loom while retaining the overall behavior. Assume we unroll a loop once and cut the loop to hinder further iterations. For the resulting net, the state space would consist of those markings from the original net being reachable from the initial marking. However, the reachability from any reachable marking (i. e., between the first and further iterations of the loop) would not be represented. Consequently, we unroll a loop twice. This enables the state space to also represent the reachability between markings of the first and the second iteration. Because of the properties of looms, these reachable markings represent all iterations. Thus, if we unroll all loops exactly twice until there is no remaining loop, the resulting net is eventually acyclic while it retains the same reachable markings as the original net, including the reachability relation between these markings. This helps to investigate and construct the state space for problems related to reachability more efficiently. In addition to checking reachability, potential applications include the computation of the *4C Spectrum* [16, 17] as a set of behavioral relations. Figure 9 illustrates the unrolled net of the example in Figure 2. It contains two duplicates of the loop inserted at its loom cluster $[[p_7]]$. Note that the nested loop consisting of $\{p4, t5, p5, t6, p6, t7\}$ is not yet unrolled as the example just sketches the idea of loop unrolling.

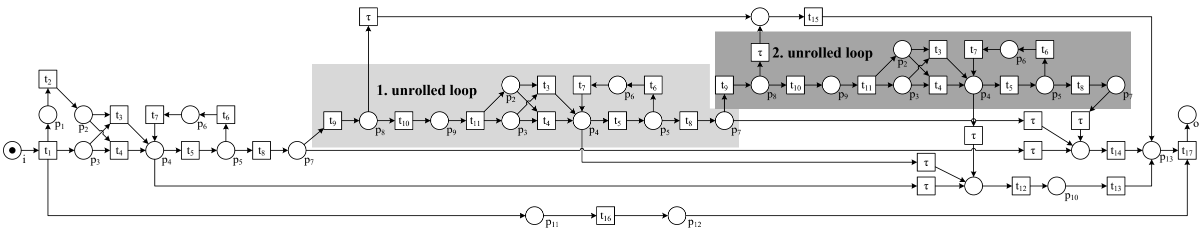


Figure 9: The unrolled version of the net in Figure 2. τ-transitions were inserted to merge flows before t_{12}, t_{14}, and t_{15}. Note that the nested loop consisting of $\{p4, t5, p5, t6, p6, t7\}$ was not unrolled for reasons of readability.

Summarizing the findings of this section, looms are a powerful concept for studying and understanding the behavior of loops and their nets. In free-choice nets with a home cluster, looms(1) can be identified as loop exit clusters, (2) help to detect nested loops with own looms, and (3) allow for unrolling loops.*Of course, future work must provide a formal proof of the free-choiceness, liveness, and safeness of nested loops, as well as the fact that reachability is preserved when loops are unrolled.*

6. Conclusion

The behavior of loops in Petri nets is sometimes difficult to capture, although important properties (such as liveness and boundedness) depend on it. This paper introduced *looms* as a kind of home markings of loops. Looms describe sub-markings that can re-occur independently of tokens outside of the corresponding loop. For cyclic nets with a home marking, the home marking itself is a loom. However, if a loop has exits, it cannot contain a home marking but might have a loom. This paper showed seven properties of loops in free-choice nets with a home cluster such as that all loop entries and exits are places and that each cluster containing an exit of a loop describes a loom.

The Petri net, business process management, and software engineering communities benefit from the concept of looms as it can help to understand complex nets with loops, or process models with re-occurring behavior. Thus, looms should simplify their analysis. Furthermore, the concept of looms is general so that it can be investigated for more general classes of nets. The seven properties about loops in free-choice nets with home clusters can further help to explain and analyze nets and process models.

In the future, the properties defined here, together with looms, can be used to formalize the constant unrolling of loops in free-choice nets with a home cluster such that the resulting free-choice net describes the same reachable markings as the original net. In addition, we plan to investigate looms in extended classes of nets to underline their usefulness.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] J. Desel, J. Esparza, Free Choice Petri Nets, Cambridge University Press, 1995.

[2] E. Best, J. Desel, J. Esparza, Traps Characterize Home States in Free Choice Systems, Theor. Comput. Sci. 101 (1992) 161–176. doi:10.1016/0304-3975(92)90048-K.

[3] T. Araki, T. Kasami, Decidable Problems on the Strong Connectivity of Petri Net Reachability Sets, Theor. Comput. Sci. 4 (1977) 99–119. doi:10.1016/0304-3975(77)90059-7.

[4] W. M. P. van der Aalst, Free-choice Nets with Home Clusters are Lucent, Fundam. Informaticae 181 (2021) 273–302. doi:10.3233/FI-2021-2059.

[5] W. M. P. van der Aalst, Markings in Perpetual Free-Choice Nets Are Fully Characterized by Their Enabled Transitions, in: V. Khomenko, O. H. Roux (Eds.), Application and Theory of Petri Nets and Concurrency - 39th International Conference, PETRI NETS 2018, Bratislava, Slovakia, June 24-29, 2018, Proceedings, volume 10877 of *Lecture Notes in Computer Science*, Springer, 2018, pp. 315–336. doi:10.1007/978-3-319-91268-4_16.

[6] T. M. Prinz, J. Klaus, N. R. T. P. van Beest, Pushing the Limits: Concurrency Detection in Acyclic Sound Free-Choice Workflow Nets in $O(P^2 + T^2)$, in: M. Köhler-Bussmeier, D. Moldt, H. Rölke (Eds.), Proceedings of the International Workshop on Petri Nets and Software Engineering 2024 co-located with the 45th International Conference on Application and Theory of Petri Nets and Concurrency (PETRI NETS 2024), June 24 - 25, 2024, Geneva, Switzerland, volume 3730 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2024, pp. 132–154. URL: https://ceur-ws.org/Vol-3730/paper08.pdf.

[7] P. S. Thiagarajan, K. Vos, A Fresh Look at Free Choice Nets, Inf. Control. 61 (1984) 85–113. doi:10.1016/S0019-9958(84)80052-2.

[8] E. Best, Structure Theory of Petri Nets: the Free Choice Hiatus, in: W. Brauer, W. Reisig, G. Rozenberg (Eds.), Petri Nets: Central Models and Their Properties, Advances in Petri Nets 1986, Part I, Proceedings of an Advanced Course, Bad Honnef, Germany, 8-19 September 1986, volume 254 of *Lecture Notes in Computer Science*, Springer, 1986, pp. 168–205. doi:10.1007/BFB0046840.

[9] J. Esparza, Reachability in Live and Safe Free-Choice Petri Nets is NP-Complete, Theor. Comput. Sci. 198 (1998) 211–224. doi:10.1016/S0304-3975(97)00235-1.

[10] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, Introduction to Algorithms, 3rd Edition, MIT Press, 2009.

[11] W. M. P. van der Aalst, Verification of Workflow Nets, in: P. Azéma, G. Balbo (Eds.), Application and Theory of Petri Nets 1997, 18th International Conference, ICATPN '97, Toulouse, France, June 23-27, 1997, Proceedings, volume 1248 of *Lecture Notes in Computer Science*, Springer, 1997, pp. 407–426. doi:10.1007/3-540-63139-9_48.

[12] T. M. Prinz, Y. Choi, N. L. Ha, Soundness unknotted: An efficient soundness checking algorithm for arbitrary cyclic process models by loosening loops, Inf. Syst. 128 (2025) 102476. doi:10.1016/J.IS.2024.102476.

[13] E. Best, R. Devillers, Petri Net Primer - A Compendium on the Core Model, Analysis, and Synthesis, Springer, 2024. doi:10.1007/978-3-031-48278-6.

[14] R. E. Tarjan, Depth-First Search and Linear Graph Algorithms, SIAM J. Comput. 1 (1972) 146–160. doi:10.1137/0201010.

[15] G. Ramalingam, On loops, dominators, and dominance frontiers, ACM Trans. Program. Lang. Syst. 24 (2002) 455–490. doi:10.1145/570886.570887.

[16] A. Polyvyanyy, M. Weidlich, R. Conforti, M. L. Rosa, A. H. M. ter Hofstede, The 4C Spectrum of Fundamental Behavioral Relations for Concurrent Systems, in: G. Ciardo, E. Kindler (Eds.),

Application and Theory of Petri Nets and Concurrency - 35th International Conference, PETRI NETS 2014, Tunis, Tunisia, June 23-27, 2014. Proceedings, volume 8489 of *Lecture Notes in Computer Science*, Springer, 2014, pp. 210–232. doi:10.1007/978-3-319-07734-5_12.

[17] T. M. Prinz, T. Welsch, N. L. Ha, Recognizing Relationships: Detecting the 4C Spectrum in $o(p^2+t^2)$ for Acyclic Sound Process Models, in: J. Borbinha, T. P. Sales, M. M. da Silva, H. A. Proper, M. Schnellmann (Eds.), Enterprise Design, Operations, and Computing - 28th International Conference, EDOC 2024, Vienna, Austria, September 10-13, 2024, Revised Selected Papers, Lecture Notes in Computer Science, Springer, 2024, pp. 281–299. doi:10.1007/978-3-031-78338-8_15.