Vicinity Respecting Homomorphisms for Abstracting System Requirements

Jörg Desel[1] and Agathe Merceron[2]

[1] Katholische Universität Eichstätt-Ingolstadt
joerg.desel@ku-eichstaett.de
[2] Beuth Hochschule für Technik Berlin
merceron@beuth-hochschule.de

Abstract. This paper is concerned with structuring system requirements on an abstract conceptual level. Channel/Agency Petri nets are taken as a formal model. They allow to represent functional aspects as well as data aspects of the requirements in a graphical way. Vicinity respecting homomorphisms are presented as a means to refine and abstract these nets. They preserve paths, i.e., dependencies between computational elements and they preserve important structural properties of nets, such as S- and T-components, siphons and traps and the free choice property. These properties have important interpretations for marked Petri nets and can therefore be used for the analysis of system models at more concrete levels.

Key words: Channel/Agency Nets, Homomorphisms, Abstraction

Preface

This paper is a short version of [6], a report that was written 13 years ago. More motivation, more examples and illustrations as well as proofs can be found in the report. It gathers, generalizes and deepens results obtained in [4, 16].

Since that time, research concentrated on abstraction techniques for Petri nets on a behavioral level, i.e. morphisms have been defined that preserve occurrence sequences or other behavioral notions. Structural relations between the respective nets appeared as a consequence of behavioral relations. For example, [14] concentrates on abstraction techniques for high-level Petri nets and explicitly distinguishes his behavior-oriented approach from our structure-based work. Our work is different because we concentrate on local structural properties of the relation between nets, i.e., on properties of the homomorphism, and derive global structural properties which have consequences for behavior.

Another line of research considers abstraction and modularity techniques for Petri nets based on graph grammars, see e.g. [8]. Considering relations between Petri nets representing conceptual models, which do not necessarily have a formal behavior, was continued in [18, 19]. It is important to notice that this work was not published in the Petri net community but in our intended application domain

– hence it points out that there is a demand for structural abstraction techniques of process models.

Recently, there is a renewed interest in construction techniques for (unmarked) Petri nets, applied for requirement analysis [7] and in the context of modular Petri nets [25] that are composed via identification of common nodes.

Whereas our work was recognized in various papers, we are not aware of any publication that continued this line of research, studying global consequences of local restrictions for Petri net homomorphisms, i.e. preserving global properties and derived behavioral properties.

Today, instead of Petri nets, diagram techniques of the UML play a more important role in practice and in theory. However, all these diagram languages are essentially graphs. Only "Activity Diagrams" have an explicit relationship to Petri nets. So future work will concentrate on possible generalizations of our results to this kind of diagrams.

1 Introduction

A nontrivial task in the design of large and complex systems is to organize the requirements into a coherent structure. Usually, this organization is a gradual process which involves refinement and abstraction between different conceptual levels of the system. In this paper we take Channel/Agency Petri nets [22, 24] to model systems and propose vicinity respecting homomorphisms as a means to refine and abstract these nets.

Channel/Agency Petri nets are a Petri net model where all elements of a net are labelled by informal descriptions. They have been proposed for the conceptual modelling of the architecture of information systems e.g. in [1, 2, 13]. As shown in [23, 24] they can be used for different levels of abstraction, in particular in the early phases of system and software engineering. On a low level of abstraction containing all details nets can be equipped with markings and a notion of behavior which simulates the behavior of the modelled system. In this way Petri nets can be used as a means for prototyping.

We introduce vicinity respecting homomorphisms of Petri nets to formalize the refinement and abstraction relations between nets. This encompasses modular techniques because each composition of subsystems may be viewed as an identification of the respective interface elements and thus as a particular abstraction. Vicinity respecting homomorphisms rely on the graph structure of a net. They are special graph homomorphisms that are able to formalize abstractions including contractions of graphs not only in their breadth but also in their length.

The definition of vicinity respecting homomorphisms is based on the local vicinities of elements. This concept suffices to preserve important global structural properties like connectedness. If two elements of a net are connected by a path then the respective system components are in a causal dependency relation. Because they preserve paths, vicinity respecting homomorphisms not only respect dependency but also its complementary relation independency.

Petri nets not only allow to combine data- and function-oriented views of a system. They also allow to concentrate on either aspect. The data aspect including nondeterministic choice is reflected by S-components. T-components represent an activity-oriented view, where only transitions are branched. Petri nets that are covered by S- and T-components allow for a compositional interpretation of these two aspects. We show that vicinity respecting net homomorphisms preserve coverings by S- and T-components. As a consequence, they respect the notions of choice (a forward branching place) and of synchronization (a backward branching transition).

In Section 2 we investigate homomorphisms of arbitrary graphs. Section 3 introduces Petri nets and transfers the notion of vicinity respecting homomorphisms to them. In Section 4 we show that vicinity respecting homomorphisms respect coverings by S- and T-components of Petri nets and draw consequences for Petri nets composition. Siphons and traps are concepts known from Petri net theory that allow for an analysis of the data contained in sets of places [3]. Section 5 proves that vicinity respecting homomorphisms preserve siphons, traps and the free choice property. Finally, Section 6 concludes this paper.

2 Graph Homomorphisms

Petri nets are special graphs. Vicinity respecting homomorphisms will be defined for arbitrary graphs in this section.

Figure 1 shows a model of a sender/receiver system on the left hand side and a coarser view of the same system in the middle. The left model can be viewed as a refinement of the right model. The interrelation between the graphs is given by a mapping which is a particular graph homomorphism. As we shall see, in this example dependencies between vertices of the source graph are strongly related to dependencies between vertices of the target graph.

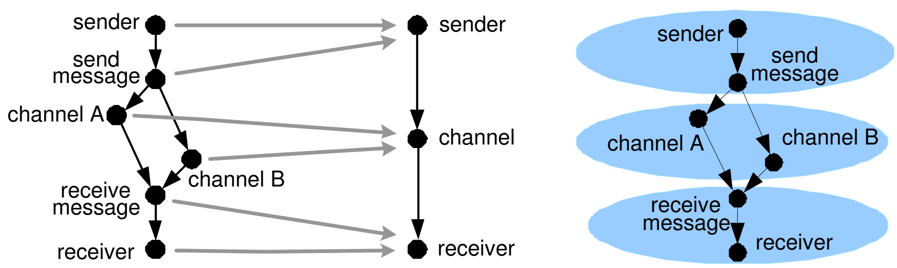

Fig. 1. A graph homomorphism as a mapping and as a quotient

We start with a formal introduction of graphs and related concepts. We consider only finite directed graphs without multiple edges and without loops.

Definition 1. *A* graph *is a pair* (X, F) *where* X *is a finite set (*vertices*) and* $F \subseteq X \times X$ *(*edges*). A* loop *is an edge* (x, x)*. A graph is said to be* loop-free *if no edge is a loop.*

The classical notion of graph homomorphism [20] respects edges in the sense that the images of connected vertices are again connected. Since we also consider contractions of loop-free graphs, where two connected vertices are mapped to one vertex without a loop, a slightly more liberal definition will be employed; we allow the images of connected vertices to be either connected or identical.

Definition 2. *Let* (X, F) *and* (X', F') *be graphs. A mapping* $\varphi\colon X \to X'$ *is a* graph homomorphism *(denoted by* $\varphi\colon (X, F) \to (X', F')$*) if, for every edge* $(x, y) \in F$*, either there is an edge* $(\varphi(x), \varphi(y)) \in F'$ *or* $\varphi(x) = \varphi(y)$*.*

To describe the environment of an element we shall use the notions of pre- and post-sets and related notions of pre- and post-vicinities.

Definition 3. *Given a graph* (X, F) *and* $x \in X$*, we denote by* ${}^\bullet x = \{y \in X \mid (y, x) \in F\}$ *the pre-set of* x *and by* $x^\bullet = \{y \in X \mid (x, y) \in F\}$ *the post-set of* x*. The pre-vicinity of* x *is* ${}^\odot x = \{x\} \cup {}^\bullet x$*, the post-vicinity of* x *is* $x^\odot = \{x\} \cup x^\bullet$*.*

$\varphi({}^\bullet x) \subseteq {}^\bullet(\varphi(x))$ does not hold for arbitrary graph homomorphisms because in case of contractions elements of ${}^\bullet x$ can be mapped to $\varphi(x)$, and similarly for post-sets. However, we get:

Proposition 1. *Let* (X, F) *and* (X', F') *be graphs. A mapping* $\varphi\colon X \to X'$ *is a graph homomorphism iff, for all* $x \in X$*,* $\varphi({}^\odot x) \subseteq {}^\odot(\varphi(x))$ *and* $\varphi(x^\odot) \subseteq (\varphi(x))^\odot$*.*

Definition 4. *A sequence* $x_1, x_2 \dots x_n$ $(n \geq 1)$ *of vertices of a graph is a* path *if there exist edges* $(x_1, x_2), \dots, (x_{n-1}, x_n)$ *of the graph. A graph is* strongly connected *if for any two vertices* x *and* y *there exists a path* $x \dots y$*.*

We allow a single element to be a path. Since consecutive vertices of a graph can be mapped onto a single element without a loop, the sequence of images of some path elements is not necessarily a path of the target graph. So we define, for loop-free graphs, the image of a path to ignore stuttering of vertices.

Definition 5. *Let* (X, F)*,* (X', F') *be loop-free graphs and* $\varphi\colon (X, F) \to (X', F')$ *a graph homomorphism. The image of a path* $x_1 \dots x_n$ *of* (X, F) *is defined by*

$$\varphi(x_1 \dots x_m) = \begin{cases} \varphi(x_1) & \text{if } m = 1 \\ \varphi(x_1 \dots x_{m-1}) & \text{if } m > 1 \text{ and } \varphi(x_{m-1}) = \varphi(x_m) \\ \varphi(x_1 \dots x_{m-1})\varphi(x_m) & \text{if } m > 1 \text{ and } \varphi(x_{m-1}) \neq \varphi(x_m) \end{cases}$$

Graph homomorphisms do not preserve edges but they preserve paths:

Lemma 1. *Let* (X, F)*,* (X', F') *be loop-free graphs and* $\varphi\colon (X, F) \to (X', F')$ *a graph homomorphism. If* $x_1 \dots x_n$ *is a path of* (X, F) *then* $\varphi(x_1 \dots x_n)$ *is a path of* (X', F') *leading from* $\varphi(x_1)$ *to* $\varphi(x_n)$*.*

Surjectivity is a first condition when graph homomorphisms are used for abstractions. Surjective graph homomorphisms preserve strong connectivity:

Corollary 1. *Let (X, F), (X', F') be loop-free graphs and $\varphi\colon (X, F) \to (X', F')$ a surjective graph homomorphism. If (X, F) is strongly connected then (X', F') is also strongly connected.*

Surjectivity concerns vertices only. An additional requirement is that every edge of a target graph reflects a connection between respective vertices of the source graph. We call such a graph homomorphism a quotient.

Definition 6. *Let (X, F), (X', F') be loop-free graphs. A surjective graph homomorphism $\varphi\colon (X, F) \to (X', F')$ is called* quotient *if, for every edge $(x', y') \in F'$, there exists an edge $(x, y) \in F$ such that $\varphi(x) = x'$ and $\varphi(y) = y'$.*

The name "quotient" is justified because for quotients, target graphs are determined up to renaming by the equivalence classes of vertices that are mapped onto the same vertex (see [5]). Therefore, we can represent quotients graphically by solely depicting equivalence classes as shown in Figure 1, right hand side.

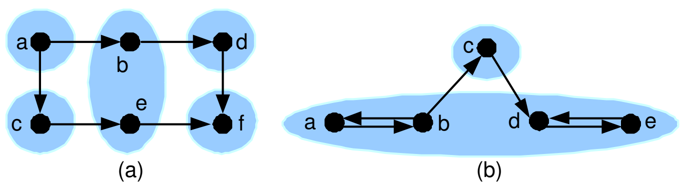

Fig. 2. Examples of graph quotients

When thinking of (X', F') as of an abstraction of (X, F), dependencies between nodes of X' that are represented through paths have to mirror dependencies already present in X. Therefore we look for a converse of Lemma 1. For quotients, this lemma has a weak converse: every path of the target graph with at most two vertices is the image of a path of the source graph. The same does not necessarily hold for longer paths, as shown in Figure 2(a). The target graph has a path $\varphi(a)\varphi(b)\varphi(f)$ which is not the image of a path of the source graph. What is wrong with this homomorphism? The post-vicinity of b is $\{b, d\}$. The post-vicinity of the image of b contains three vertices, namely $\varphi(b), \varphi(d)$ and $\varphi(f)$. So the image of the post-vicinity of b is properly included in the post-vicinity of the image. We say that the post-vicinity is not respected and define homomorphisms that respect vicinities of vertices:

Definition 7. *Let (X, F), (X', F') be loop-free graphs. A graph homomorphism $\varphi\colon (X, F) \to (X', F')$ is called pre-vicinity respecting if, for every $x \in X$, either $\varphi({}^{\odot}x) = {}^{\odot}(\varphi(x))$ or $\varphi({}^{\odot}x) = \{\varphi(x)\}$. φ is called post-vicinity respecting if, for every $x \in X$, either $\varphi(x^{\odot}) = (\varphi(x))^{\odot}$ or $\varphi(x^{\odot}) = \{\varphi(x)\}$. φ is called vicinity respecting if it is pre-vicinity respecting and post-vicinity respecting.*

The following theorem states that for surjective post-vicinity respecting homomorphisms of strongly connected graphs there is a converse of Lemma 1. By symmetry, the same holds for pre-vicinity respecting homomorphisms.

Theorem 1. *Let (X, F), (X', F') be loop-free graphs such that (X, F) is strongly connected, and $\varphi\colon (X, F) \to (X', F')$ a surjective post-vicinity respecting graph homomorphism. Let $x'_1 \dots x'_m$ be a path of (X', F') such that, for $1 \leq i < m$, $x'_i \neq x'_{i+1}$ (no stuttering). Then there is a path $x_1 \dots x_n$ of (X, F) satisfying $\varphi(x_1 \dots x_n) = x'_1 \dots x'_m$.*

The example in Figure 2(b) shows that in the previous theorem it is necessary that the source graph (X, F) is strongly connected. This graph homomorphism φ is a vicinity respecting quotient. The target graph has a path $\varphi(c)\varphi(d)\varphi(c)$ which is not the image of a path of the source graph.

Corollary 2. *Let (X, F), (X', F') be loop-free graphs such that (X, F) is strongly connected and let $\varphi\colon (X, F) \to (X', F')$ be a surjective post-vicinity respecting graph homomorphism. Then φ is a quotient.*

The following result is weaker than Corollary 2 but holds for arbitrary surjective mappings.

Lemma 2. *Let (X, F), (X', F') be loop-free graphs such that (X, F) is strongly connected and $|X'| > 1$. If $\varphi\colon (X, F) \to (X', F')$ is a surjective mapping then, for every $x' \in X'$, there are arcs $(y, x_1), (x_2, z) \in F$ with $\varphi(x_1) = x' = \varphi(x_2)$ and $\varphi(y) \neq x' \neq \varphi(z)$.*

Concentrating on different elements which are mapped onto the same image instead of comparing source graph and target graph leads to another aspect of vicinity respecting homomorphisms in the case of quotients.

Proposition 2. *Let (X, F), (X', F') be loop-free graphs and $\varphi\colon (X, F) \to (X', F')$ a quotient. φ is vicinity respecting iff for all $x, y \in X$ satisfying $\varphi(x) = \varphi(y)$:*
1. *$\varphi({}^{\odot}x) = \{\varphi(x)\}$ or $\varphi({}^{\odot}y) = \{\varphi(y)\}$ or $\varphi({}^{\odot}x) = \varphi({}^{\odot}y)$;*
2. *$\varphi(x^{\odot}) = \{\varphi(x)\}$ or $\varphi(y^{\odot}) = \{\varphi(y)\}$ or $\varphi(x^{\odot}) = \varphi(y^{\odot})$.*

3 Net Homomorphisms

A net can be seen as a loop-free graph (X, F) where the set X of vertices is partitioned into a set S of *places* and a set T of *transitions* such that F may not relate two places or two transitions. Formally:

Definition 8. *A triple $N = (S, T, F)$ is called* net *if: S and T are disjoint sets and $F \subseteq (S \times T) \cup (T \times S)$. The set $X = S \cup T$ is the set of* elements *of the net.*

This definition allows to consider nets with isolated elements, i.e. elements with empty pre- and post-sets. We do not consider markings and behavioral notions but concentrate on the structure of net models.

We use the following convention: indices and primes used to denote a net N are carried over to all parts of N. For example, speaking of a net N_i', we implicitly understand $N_i' = (S_i', T_i'; F_i')$ and $X_i' = S_i' \cup T_i'$.

The $\bullet$-notation for pre- and post-sets and the $\odot$-notation for pre- and post-vicinities of single elements carries over to nets. Moreover, we will employ the $\bullet$-notation for sets of elements as usual: The pre-set of a set of elements is the union of pre-sets of elements of the set, and similar for post-sets.

The transitions of a net model the active subsystems, i.e. functions, operators, transformers etc. They are only connected to places which model passive subsystems, i.e. data, messages, conditions etc. On a conceptual level, it is not always obvious to classify a subsystem active or passive. The decision to model it by a place or by a transition is based on the interaction of the subsystem with its vicinity. As an example, consider a channel that is connected to functional units that send and receive data through the channel. Then the channel has to be modelled by a place. In contrast, if the channel is connected to data to be sent on one side and to already received data on the other side then the channel is modelled by a transition. As we shall see, a transition may represent a subsystem that is modelled by a net containing places and transitions on a finer level of abstraction. The same holds respectively for places.

Homomorphisms of Petri nets are particular graph homomorphisms that additionally respect the type of relation between the elements given by arrows [21, 11]. Since we again allow contractions, places can be mapped to transitions and transitions can be mapped to places. However, if two connected elements are not mapped to the same element of the target net, then the place of the two has to be mapped to a place and the transition has to be mapped to a transition. So Definition 2 becomes for Petri nets:

Definition 9. *Let N, N' be nets. A mapping $\varphi : X \to X'$ is called* net homomorphism*, denoted by $\varphi : N \to N'$, if for every edge $(x, y) \in F$ holds:*
- if $(x, y) \in F \cap (S \times T)$ then either $(\varphi(x), \varphi(y)) \in F' \cap (S' \times T')$ or $\varphi(x) = \varphi(y)$,
- if $(x, y) \in F \cap (T \times S)$ then either $(\varphi(x), \varphi(y)) \in F' \cap (T' \times S')$ or $\varphi(x) = \varphi(y)$.

This definition is equivalent to the one given in [10].

Lemma 3. *Let $\varphi : N \to N'$ be a net homomorphism. Then:*
- if a transition $t \in T$ is mapped to a place s' then $\varphi({}^{\odot}t \cup t^{\odot}) = \{s'\}$,
- if a place $s \in S$ is mapped to a transition t' then $\varphi({}^{\odot}s \cup s^{\odot}) = \{t'\}$.

Corollary 3. *Let $\varphi : N \to N'$ be a net homomorphism and let $(x, y) \in F$ such that $\varphi(x) \neq \varphi(y)$. Then $\varphi(x) \in S'$ iff $x \in S$ and $\varphi(y) \in S'$ iff $y \in S$.*

For Petri nets the vicinity respecting homomorphism definition can be split into two notions: homomorphisms that respect the vicinity of places and homomorphisms that respect the vicinity of transitions.

Definition 10. *Let $\varphi: N \to N'$ be a net homomorphism.*

1. *φ is* S-vicinity respecting *if, for every $x \in S$:*
   (a) $\varphi({}^{\odot}x) = {}^{\odot}(\varphi(x))$ or $\varphi({}^{\odot}x) = \{\varphi(x)\}$ and
   (b) $\varphi(x^{\odot}) = (\varphi(x))^{\odot}$ or $\varphi(x^{\odot}) = \{\varphi(x)\}$.
2. *φ is* T-vicinity respecting *if, for every $x \in T$:*
   (a) $\varphi({}^{\odot}x) = {}^{\odot}(\varphi(x))$ or $\varphi({}^{\odot}x) = \{\varphi(x)\}$ and
   (b) $\varphi(x^{\odot}) = (\varphi(x))^{\odot}$ or $\varphi(x^{\odot}) = \{\varphi(x)\}$.
3. *φ is* vicinity respecting *if it is both S- and T-vicinity respecting*

A subnet of a net is generated by its elements and preserves the flow relation between its elements. We will be interested in subnets that are connected to the remaining part only via places or only via transitions.

Definition 11. *Let N be a net. The $\bullet$-notation refers to N in the sequel.*

1. *$X_1 \subseteq X$ generates the* subnet $N_1 = (S \cap X_1, T \cap X_1; F \cap (X_1 \times X_1))$.
2. *N_1 is called* transition-bordered *if ${}^{\bullet}S_1 \cup S_1^{\bullet} \subseteq T_1$.*
3. *N_1 is called* place-bordered *if ${}^{\bullet}T_1 \cup T_1^{\bullet} \subseteq S_1$.*

A single transition of a net constitutes a transition-bordered subnet and a place constitutes a place-bordered subnet. Net homomorphism allow to map places to transitions and vice versa. Nevertheless, the role of active and passive components of a net are preserved in the following sense. The refinement of a transition-bordered subnet is a transition-bordered subnet, i.e., the reverse image of the elements of a transition-bordered subnet generates a transition-bordered subnet of the source net. Similarly, the set of elements of the source net that are mapped to some place-bordered subnet of the target net constitute a place-bordered subnet of the source net. The following results have been proved in [9] in a topological framework.

Proposition 3. *Let $\varphi: N \to N'$ be a net homomorphism.*

1. *If N_1' is a transition-bordered subnet of N' then $\{x \in X \mid \varphi(x) \in X_1'\}$ generates a transition-bordered subnet of N.*
2. *If N_1' is a place-bordered subnet of N' then $\{x \in X \mid \varphi(x) \in X_1'\}$ generates a place-bordered subnet of N.*

4 Transformation of S- and T-components

Recall that an S-component of a net yields a data-oriented view of a part of the system. An S-component can contain nondeterministic choices that are modelled by branching places, i.e. by places with more than one output transitions. It does however not contain aspects of concurrency, whence its transitions are not branched [3]. Similarly, T-components concentrate on functional aspects. They do not contain branching places. Formally S-components and T-components are particular subnets.

Definition 12. *Let N be a net. The $\bullet$-notation refers to N in the sequel.*

1. *A strongly connected transition-bordered subnet N_1 of N is called* S-component *of N if, for every $t \in T_1$, $|{}^\bullet t \cap S_1| \leq 1 \wedge |t^\bullet \cap S_1| \leq 1$. N is* covered by S-components *if there exists a family of S-components (N_i), $i \in I$, such that for every $x \in X$ there exists an $i \in I$ such that $x \in X_i$.*
2. *A strongly connected place-bordered subnet N_1 of N is called* T-component *of N if, for every $s \in S_1$, $|{}^\bullet s \cap T_1| \leq 1 \wedge |s^\bullet \cap T_1| \leq 1$. N is* covered by T-components *if there exists a family of T-components (N_i), $i \in I$, such that for every $x \in X$ there exists an $i \in I$ such that $x \in X_i$.*

Definition 13. *Let $\varphi: N \to N'$ be a net homomorphism and N_1 a subnet of N. The net $(\varphi(X_1) \cap S', \varphi(X_1) \cap T'; \{(\varphi(x), \varphi(y)) \mid (x, y) \in F_1 \wedge \varphi(x) \neq \varphi(y)\})$ is called the net image of N_1 by φ. It is denoted by $\varphi(N_1)$. By $\varphi_{N_1}: X_1 \to \varphi(X_1)$ we denote the restriction of φ to X_1, with the range of φ restricted to $\varphi(X_1)$.*

φ_{N_1} is surjective by definition. Note that $\varphi(N_1)$, the net image of N_1, is not necessarily a subnet of the target net N_1. Figure 3(a) gives an example.

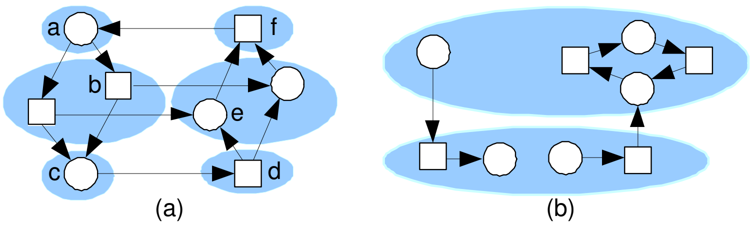

Fig. 3. Examples of net quotients

Proposition 4. *If $\varphi: N \to N'$ is a net homomorphism and N_1 is a subnet of N then $\varphi_{N_1}: N_1 \to \varphi(N_1)$ is a quotient.*

Corollary 4. *A net homomorphism $\varphi: N \to N'$ is a quotient if and only if $N' = \varphi(N)$ and in this case $\varphi = \varphi_N$.*

S-vicinity respecting net homomorphisms map a strongly connected transition-bordered subnet either onto a single element or onto a strongly connected transition-bordered subnet:

Proposition 5. *Let $\varphi: N \to N'$ be an S-vicinity respecting net homomorphism and N_1 a strongly connected transition-bordered subnet of N. Define $N_1' = \varphi(N_1)$.*

1. *N_1' is a subnet of N'.*
2. *If $|X_1'| > 1$ then N_1' is a transition-bordered subnet of N'.*
3. *$\varphi_{N_1}: N_1 \to N_1'$ is S-vicinity respecting.*

The example in Figure 3(b) shows that being strongly connected is a necessary prerequisite for Proposition 5. In Figure 3(a) we gave an example of a strongly connected subnet which is not a transition-bordered subnet. Its image by the S-vicinity respecting quotient is not a subnet of the target net. An S-component is in particular a strongly connected transition-bordered subnet. For respecting coverings by S-components, stronger hypotheses have to be assumed. Let us continue considering the S-vicinity respecting quotient shown in Figure 4(a). This net is covered by S-components. The net homomorphism φ is an S-vicinity respecting quotient. However, the target net is not covered by S-components. Observe that the restriction of φ to any S-component is not T-vicinity respecting. Consider the S-component N_1 containing b. The image of N_1 is the entire target net. We have $\varphi_{N_1}(\{a,b\}) \neq \{\varphi_{N_1}(a)\} = \{u\}$ but $\varphi_{N_1}(\{a,b\}) = \{u,w\} \neq (\varphi_{N_1}(a))^{\odot} = \{u,v,w\}$. The net image of N_1 is not an S-component of the target net.

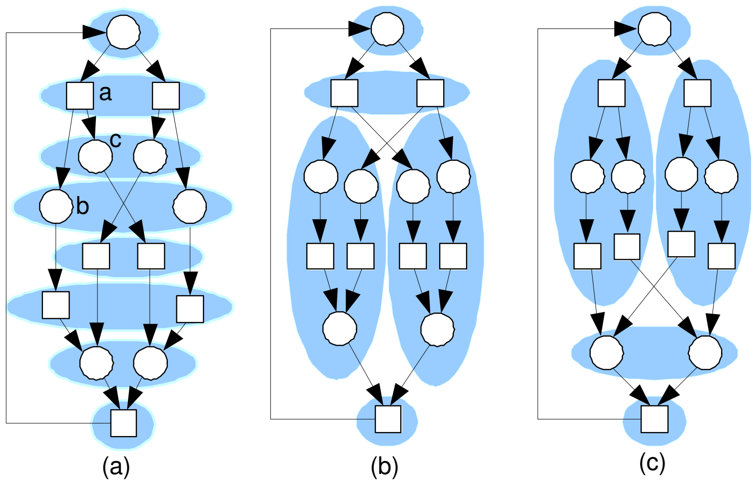

Fig. 4.

In Figure 4(b), the quotients restricted to any S-component are T-vicinity respecting. Remember that quotients can by simply drawn by depicting the equivalence classes of elements that are identified by the quotient, as shown for arbitrary graphs in Section 1.

Proposition 6. *Let $\varphi\colon N \to N'$ be an S-vicinity respecting net homomorphism and N_1 an S-component of N. Define $N_1' = \varphi(N_1)$ and suppose $\varphi_{N_1}\colon N_1 \to N_1'$ is T-vicinity respecting. If $|X_1'| > 1$ then N_1' is an S-component of N'.*

From Proposition 6 we deduce:

Theorem 2. *Let N be a net, covered by a family (N_i), $i \in I$ of S-components. Let $\varphi: N \to N'$ be an S-vicinity respecting quotient such that, for all $i \in I$, $\varphi_{N_i}: N_i \to \varphi(N_i)$ is T-vicinity respecting. Then N' is covered by S-components.*

By Proposition 5(3), 'φ is S-vicinity respecting' implies for all $i \in I$: 'φ_{N_i} is S-vicinity respecting'. So all the φ_{N_i} have to be both S- and T-vicinity respecting. However, this alone does not imply that φ is S-vicinity respecting and is not sufficient for N' to be covered by S-components as is shown in Figure 5(a). For the S-component N_1 of this net, shown in Figure 5(b), φ_{N_1} is S-and T-vicinity respecting. However, φ is not S-vicinity respecting and N' is not covered by S-components.

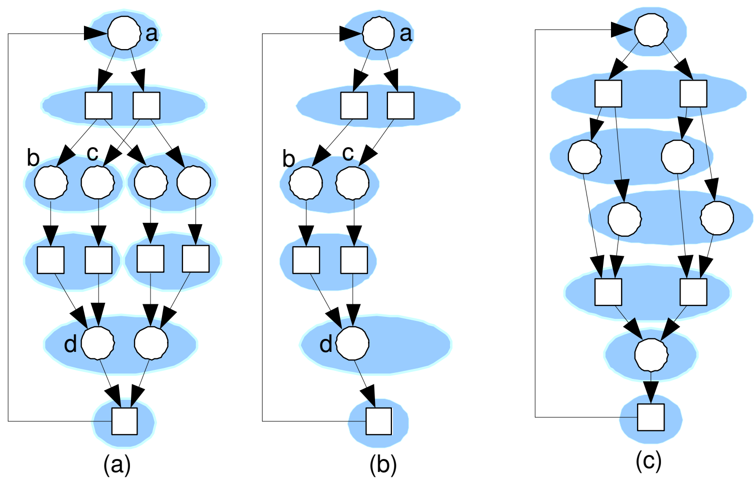

Fig. 5.

Theorem 2 implies that, given a family of S-components which cover the source net, a respective covering of the target net is obtained by the images of the S-components which are not mapped to single non-isolated places.

The choice of a covering family of S-components is decisive. In the example of Figure 5(c), the quotient is vicinity respecting. Its restriction to either the S-component N_1 which contains the respective left places or to the S-component N_2 which contains the respective right places is T-vicinity respecting. Taking the other two possible S-components as a cover of N, the restriction of φ to any of these S-components is not T-vicinity respecting. So the choice of an abstraction and the choice of an S-component covering are not independent.

By duality we get:

Corollary 5. *Let $\varphi: N \to N'$ be a T-vicinity respecting net homomorphism and N_1 a strongly connected place-bordered subnet of N. Define $N_1' = \varphi(N_1)$. Then:*

1. *N_1' is a subnet of N';*

2. If $|X'_1| > 1$ then N'_1 is a place-bordered subnet of N';
3. $\varphi_{N_1}: N_1 \to N'_1$ is T-vicinity respecting.

The dual version of Theorem 3.8 reads as follows:

Theorem 3. *Let N be a net, covered by a family $(N_i), i \in I$ of T-components. Let $\varphi: N \to N'$ be a T-vicinity respecting quotient such that, for all $i \in I$, $\varphi_{N_i}: N_i \to \varphi(N_i)$ is S-vicinity respecting. Then N' is covered by T-components.*

The net homomorphisms depicted in Figure 4(b) and 4(c) are vicinity respecting. Their restrictions to any S-component or T-component are also vicinity respecting. Hence their net images are covered by S- and T-components.

A particular case of Theorem 2 is the composition of S-components; the source net N is the disjoint union of a family of S-components and the mapping, restricted to each of these S-components, is injective (and hence a fortiori T-vicinity respecting). We can reformulate our result as a property of net homomorphisms as follows: For every place a of an S-component N_1 of a net N the entire vicinity belongs to the S-component as well by definition. Therefore the natural injection $\psi_1: N_1 \to N$ is S-vicinity respecting but not necessarily surjective. A covering by S-components $N_i (i \in I)$ can be expressed by a set of net homomorphisms $\psi_i (i \in I)$ such that each element of N is in $\psi_i(N_i)$ for at least one i. Using the disjoint union of the S-components ($\biguplus N_i$), the net homomorphisms ψ_i induce a quotient ψ from $\biguplus N_i$ to N. Now Theorem 2 reads as follows. Given

- a family (N_i), $i \in I$ of strongly connected nets with $|{}^\bullet t| \leq 1$, $|t^\bullet| \leq 1$ for all transitions t (S-components),
- S-vicinity respecting injective net homomorphisms $\psi_i: N_i \to N (i \in I)$ such that the induced mapping ψ is a quotient (i.e., N is covered by the N_i),
- an S-vicinity respecting quotient $\varphi: N \to N'$ such that φ_{N_i} is T-vicinity respecting for all $i \in I$,

we can find injective S-vicinity respecting mappings $\psi'_i: \varphi(N_i) \to N'$ such that the induced mapping $\psi': \bigcup \varphi(N_i) \to N'$ is surjective (N' covered by the $\varphi(N_i)$).

Again, by duality we can use the same formalism to capture the composition of T-components.

5 Siphons, Traps and Free Choice Property

Definition 14. *Let N be a net.*
A siphon of a net is a nonempty set of places A satisfying ${}^\bullet A \subseteq A^\bullet$.
A trap of a net is a nonempty set of places A satisfying $A^\bullet \subseteq {}^\bullet A$.
A siphon (trap) is minimal if it does not strictly include any other siphon (trap).

For marked Petri nets, siphons and traps are used to deduce behavioral properties of the system [3]. Also at the conceptual level of Channel/Agency nets, they can be used to analyze aspects of the data and information flow in the modelled system. Roughly speaking, if a set of places is a trap then information cannot get completely lost in the component modelled by these places. For the places of a siphon, it is not possible to add information without taking data from the siphon into account.

Minimal siphons and traps are particularly important for the analysis of marked Petri nets. We will show that vicinity respecting net homomorphisms map minimal siphons either onto singletons or onto siphons of the target net, and similarly for minimal traps. We begin with a preliminary result.

Proposition 7. *[3] Let A be a minimal siphon of a net N. Then the subnet generated by ${}^\bullet A \cup A$ is strongly connected.*

Theorem 4. *Let $\varphi\colon N \to N'$ be a surjective S-vicinity respecting net homomorphism. If A is a minimal siphon of N then either $\varphi(A)$ is a single node (place or transition) or $\varphi(A) \cap S'$ is a siphon of N'.*

By symmetrical arguments, an analogous result holds for traps:

Theorem 5. *Let $\varphi\colon N \to N'$ be a surjective S-vicinity respecting net homomorphism. If A is a minimal trap of N then either $\varphi(A)$ is a single node (place or transition) or $\varphi(A) \cap S'$ is a trap of N'.*

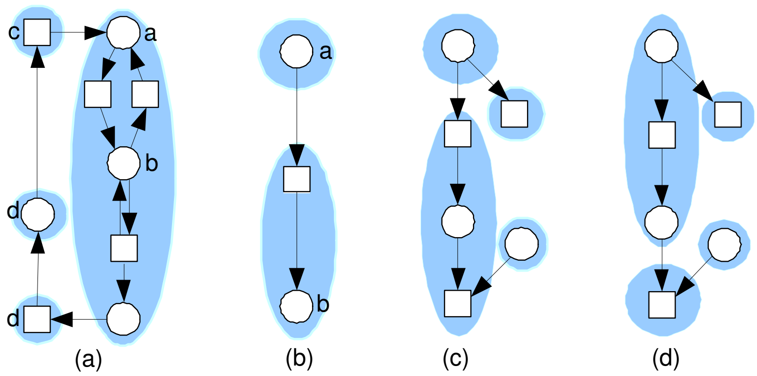

Fig. 6.

Figures 6(a) and 6(b) show that the strong connectedness of ${}^\bullet A \cup A$, implied by its minimality, and the fact that the image of A does have more than one element are necessary conditions. We close this section establishing that vicinity respecting quotients respect free choice Petri nets. Important behavioral properties are characterized in terms of traps, siphons for these nets and the class of free choice Petri nets which is covered by S- and T-components is well established [3]. In a free choice net, if two transitions share some input places, then they share all their input places.

Definition 15. *A net N is called free choice if for any two places s_1 and s_2 either $s_1^\bullet \cap s_2^\bullet = \emptyset$ or $s_1^\bullet = s_2^\bullet$.*

Theorem 6. *Let N be a free choice net and $\varphi: N \to N'$ be a vicinity respecting quotient. Then N' is free choice as well.*

Figures 6(c) and 6(d) show that for the previous theorem, S-vicinity respecting and T-vicinity respecting alone are not sufficient.

6 Conclusion

Structuring system requirements is a gradual process which involves refinement/abstraction between different conceptual levels. Abstractions should bear formal relations with refinements because otherwise the analysis of some abstraction will be of no help for the induced refinement. We argued that vicinity respecting homomorphisms give a possible solution to these requirements for graph-based models of distributed systems. They provide a method to perform graphical abstraction/refinement such that every element is either glued together with its vicinity or its vicinity is the vicinity of its image.

The vicinity respecting concept is a local notion because its definition only uses local vicinities. However, it has global consequences since it preserves paths and, consequently, connectedness properties. For Petri nets, vicinity respecting homomorphisms preserve moreover important structural properties such as S- and T-components, siphons and traps and the free-choice property.

Other concepts for refinement and abstraction of Petri nets and of morphisms [26, 17] have been proposed in the literature. However, all these approaches are concerned with marked Petri nets and aim results involving the behavior given by the token game. In contrast, we are concerned with the preliminary task of structuring software requirements down to a working system and aim at structure preservation. Generally, abstraction in our sense is more general than behavior preserving abstraction. However, structure influences behavior. The transition refinement considered in [12] turns out to induce a vicinity respecting homomorphism from the refined net to the coarser net.

References

1. Bruno, G.: Model-Based Software Engineering. Chapman and Hall (1995)
2. Deiters, W. and Gruhn, V.: The FUNSOFT net approach to software process management. International Journal on Software Engineering and Knowledge Engineering Vol. 4 No. 2 (1994)
3. Desel, J. and Esparza, J.: Free Choice Petri Nets. Cambridge Tracts in Theoretical Computer Science 40, Cambridge University Press (1995)
4. Desel, J. and Merceron A.: Vicinity respecting net morphisms. Advances in Petri Nets 1990, LNCS Vol. 483 pp. 165–185, Springer-Verlag (1991)
5. Desel, J.: On abstractions of nets. Advances in Petri Nets 1991, LNCS Vol. 524 pp. 78–92, Springer-Verlag (1991)

6. Desel, J. and Merceron, A.: Vicinity respecting homomorphisms for abstracting system requirements. Bericht No. 337 of Institut AIFB, Universität Karlsruhe (1996), available via `www.informatik.ku-eichstaett.de`
7. Desel, J. and Petrucci, L.: Aggregating views for Petri net model construction. Petri Nets and Distributed Systems (PNDS 2008), Workshop at the 29th International Conference on Application and Theory of Petri Nets and Other Models of Councurrency, Xi'an, China (2008)
8. Ehrig, H., Hoffmann, K. and Padberg, J.: Transformations of Petri Nets, Electr. Notes Theor. Comput. Sci. Vol. 148 No. 1, pp. 151–172 (2006)
9. Fernández C., C.: Net Topology I. Interner Bericht der GMD ISF-75-9 GMD St. Augustin, Germany (1975). Net Topology II. Interner Bericht der GMD ISF-76-2, GMD St. Augustin, Germany (1976)
10. Genrich, H.J., Lautenbach, K. and Thiagarajan, P.S.: Elements of general net theory. Net Theory and Applications, Lecture Notes in Computer Science Vol. 84 pp. 21–163, Springer-Verlag (1980)
11. Genrich, H.J. and Stankiewicz-Wiechno, E.: A dictionary of some basic notions of net theory. Net Theory and Applications, LNCS Vol. 84 pp. 519–535, Springer-Verlag (1980)
12. van Glabbeek, R. and Goltz, U.: Refinements of actions in causality based models. Stepwise Refinement of Distributed Systems - Models, Formalisms, Correctness; REX Workshop, LNCS Vol. 430 pp. 267–300, Springer-Verlag (1989)
13. van Hee, K.M.: Information Systems Engineering - a Formal Approach. Cambridge University Press (1994)
14. Lakos, C.: Composing abstractions of coloured Petri nets. Application and Theory of Petri Nets, LNCS Vol. 1825, pp. 323–345, Sppringer-Verlag (2000)
15. Luckham, D.C., Kenney, J.J. et al.: Specification and analysis of system architecture using Rapide. IEEE Transactions on Software Engineering, Special Issue on Software Architecture. Vol. 21, No. 4, April 1995
16. Merceron, A.: Morphisms to preserve structural properties of Petri nets. Computer Science – Research and Applications, Plenum Press, pp. 439–454 (1994)
17. Meseguer, J. and Montanari, U.: Petri nets are monoids. Information and Computation Vol. 88, pp. 105–155 (1990)
18. Mikolajczak, B. and Wang, Z.: Conceptual Modeling of Concurrent Systems through Stepwise Abstraction and Refinement Using Petri Net Morphisms. Conceptual Modeling - ER 2003, LNCS Vol. 2813, pp. 433-445, Springer-Verlag (2003)
19. Mikolajczak, B. and Wang, Z.: Conceptual Modeling of Concurrent Information Systems with General Morphisms of Petri Nets. Intelligent Information Systems, Advances in Soft Computing, pp. 535-539, Springer-Verlag (2003)
20. Ore, O.: Theory of Graphs. American Mathematical Society, Colloquium Publications; Vol. XXXVIII (1962)
21. Petri, C.A.: Concepts of net theory. Mathematical Foundations of Computer Science: Proceedings of Symposium and Summer School, High Tatras, Sep. 3-8, 1973
22. Petri, C.A.: Introduction to General Net Theory. Net Theory and Applications, LNCS Vol. 84, pp. 1–20, Springer-Verlag (1980)
23. Reisig, W.: Petri nets in software engineering. Advances in Petri Nets 1986 – Part 2. LNCS Vol. 255, pp. 63–96, Springer-Verlag (1987)
24. Reisig, W.: A Primer in Petri Net Design. Springer-Verlag (1992)
25. Reisig, W.: The universal net composition operator. To appear at Petri Nets and Other Models of Concurrency (ATPN) 2009, LNCS, Springer-Verlag (2009)
26. Winskel, G.: Petri nets, algebras, morphisms and compositionality. Information and Computation 72, pp. 197–238 (1987)