Categorical Foundation for Layer Consistency in AHO-Net Models Supporting Workflow Management in Mobile Ad-Hoc Networks

Enrico Biermann, Tony Modica and Kathrin Hoffmann
Institut für Softwaretechnik und Theoretische Informatik
Technische Universität Berlin, Germany
{enrico,modica,hoffmann}@cs.tu-berlin.de

Forschungsberichte des Fachbereichs Informatik
Bericht-Nr. 2007/01, ISSN 1436-9915

Categorical Foundation for Layer Consistency in AHO-Net Models Supporting Workflow Management in MANETs*

E. Biermann, T. Modica, and K. Hoffmann

Institute for Software Technology and Theoretical Computer Science
Technical University Berlin, Germany

Abstract. In this paper we present a layered architecture for modeling workflows in **M**obile **A**d-Hoc **NET**works (MANETs) using algebraic higher order nets (AHO nets). MANETs are networks of mobile devices that communicate with each other via wireless links without relying on an underlying infrastructure, e.g. in emergency scenarios, where an effective coordination is crucial among team members, each of them equipped with hand-held devices.
Workflows in MANETs can be adequately modeled using a layered architecture, where the overall workflow, the team members' activities and the mobility issues are separated into three different layers, namely the workflow layer, the mobility layer and the team layer. Dividing the AHO net model into layers immediately rises the question of consistency. We suggest a formal notion of layer consistency requiring that the team layer is given by the mapping of the individual member's activities to the gluing of the workflow and the mobility layer. The main results concern the maintenance of the layer consistency when changing the workflow layer, the mobility layer and the team layer independently.

1 Introduction

Mobile Ad-Hoc Networks (MANETs) consist of mobile nodes, communicating independently of a stable infrastructure. The network topology is changed continuously depending on the actual position and availability of the nodes. A typical example is a group of team members communicating using hand-held devices and laptops as e.g. in the disaster recovery scenario in Section 2. Formal modeling of workflows in MANETs using algebraic higher order nets (AHO nets) has been first introduced in [4]. AHO nets are Petri nets with complex tokens, namely place/transition (P/T) nets as well as rules and net transformations for changing these nets. On this basis we present a layered architecture of the model that allows the separation of support activities concerning the network from activities concerning the intended workflow. This yields better and conciser models, since supporting the network connectivity has a much finer granularity than the more or less fixed workflow execution. The layered architecture of AHO net models of

* This work has been partly funded by the research project forMA$_l$NET

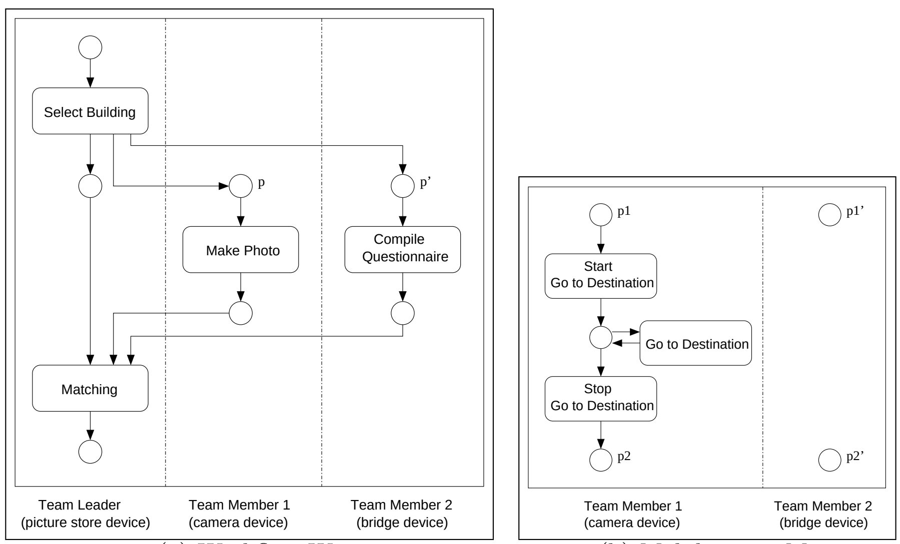
workflows in MANETs distinguishes three layers, the workflow layer, the mobility layer and the team layer. The workflow layer describes the overall workflow that is to be achieved by the whole team. The mobility layer describes the workflows in order to maintain the MANETs connectivity. The team layer describes the individual activities of the team members. Moreover, we provide a set of rules in each layer for the transformation of corresponding P/T-nets expressing different system states. As we distinguish different layers in which transformations are applied independently, the question comes up how these layers fit together. Layer consistency means that these layers together form a valid AHO net model of workflows in MANETs. In a mobile setting it is not realistic to expect consistency at all moments, so there are different degrees of inconsistency that are feasible during maintenance of consistency. Consider the subsequent possibilities for maintaining consistency in a layered AHO net model of workflows in MANETs: *Checking consistency* means that in all states of the AHO net modeling the workflows in MANETs consistency can be checked. *Guaranteed consistency* is given if each state of the AHO net is a consistent one, that is the rules are only applied when the conditions that guarantee consistency are satisfied. *Backtracking consistency* is the possibility to reach an inconsistent state, and to have then the possibility to backtrack the transformations until a consistent state is reached. *Restoring consistency* is the possibility of inconsistent states in the AHO net, but with a "recipe" to fix them. (So, backtracking could be considered as a special case.) This recipe provides conditions for the application of the next transformations. The notion of consistency we present in this paper can be used for all four possibilities. Consistency maintenance depends on the precise AHO net model. More precisely, the way consistency is maintained is given by the way rules are applied during the firing of the transitions of the AHO net model. Orthogonally, there are other notions of consistency that are relevant for an AHO net model of workflows in MANETs, e.g. the intended workflow of the whole team is covered by the individual activities of the team members. Another important consistency notion concerns the distributed behavior that means in which way the behavior of each member is interrelated with the behavior of the other team members. In the conclusion we hint at the possible formalization of such a team work consistency or behavior consistency in our approach.

The formal approach presented in this paper was developed in strong collaboration with some research projects[1] where an adaptive workflow management system for MANETs, specifically targeted to emergency scenarios, is partly realized resp. going to be implemented. Section 2 introduces an exemplary scenario of disaster management to illustrate our notions and results, while in Section 3 we discuss our approach to model workflows in MANETs using a layered architecture. The formalization to maintain consistency in layered architectures can be found in Section 4. Finally, we give a conclusion and discuss future work in 5.

[1] MOBIDIS - http://www.dis.uniroma1.it/pub/mecella/projects/MobiDIS, MAIS - http://www.mais-project.it, IST FP6 WORKPAD - http://www.workpad-project.eu/

2 Scenario: Emergency Management

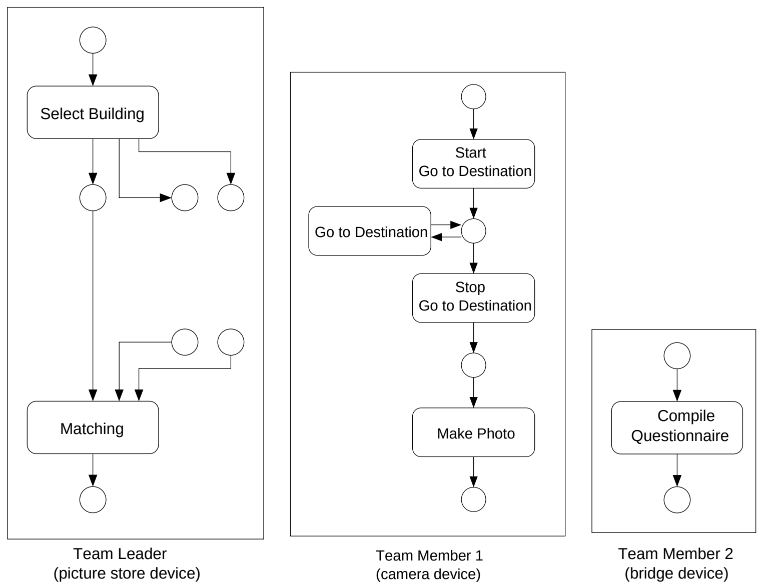
As a running example we use a scenario in archaeological disaster/recovery: after an earthquake, a team (led by a team leader) is equipped with mobile devices (laptops and PDAs) and sent to the affected area to evaluate the state of archaeological sites and the state of precarious buildings. The goal is to draw a situation map in order to schedule restructuring jobs. The team is considered as an overall MANET in which the team leader's device coordinates the other team members' devices by providing suitable information (e.g. maps, sensible objects, etc.) and assigning activities. A typical cooperative process to be enacted by the team is shown in Fig. 1(a), where the team leader has to select a building based on previously stored details of the area while team member 1 could take some pictures of the precarious buildings and team member 2 (after a visual analysis of a building) could fill in some specific questionnaires. Finally, these results have to be analyzed by the team leader in order to schedule next activities.



(a) Workflow W_0 (b) Mobility net M_0

Fig. 1. P/T-nets in the workflow and the mobility layer

In the following we exemplarily present P/T-nets called token nets for our scenario. As described above, Fig. 1(a) presents the workflow W_0 that has to be cooperatively executed by the team. The dashed lines are an additional information illustrating the relation among tasks and team members and are not a part of the P/T-net itself. There is a corresponding workflow $\underline{W}_0$ where the place **p** is represented by two places **p1** and **p2** (and similar place **p'**) to integrate movement activities. In Fig. 1(b) the token net M_0 presents the mobility aspect of team member 1 stating that he/she has to go to the selected destination while team member 2 stays put. Finally, in Fig. 2 there are three separate nets for the team layer showing the local view of each team member onto the workflow and the mobility net.



(a) P/T-net $\mathbf{t_0^0}$ (b) P/T-net $\mathbf{t_0^1}$ (c) P/T-net $\mathbf{t_0^2}$

Fig. 2. Team member nets in the team layer

To maintain consistency in a layered architecture first of all the teamwork net T_0 (see Fig. 3) has to be produced by gluing the workflow $\underline{W}_0$ and the mobility net M_0 (see Fig. 1). In more detail, the place **p** in the workflow W_0 is refined by the movement activities of team member 1. Moreover, the local view of each team member (see Fig. 2) is achieved by an inclusion into the teamwork net T_0, called activity arrow, that realizes the relation of team members to their activities. Thus, we start with a consistent layer environment (see Section 4).

In a particular scenario the movement of the device equipped with the camera could result in a disconnection from the others. To maintain the network connectivity and ensure a path among devices a layered architecture should be able to alert the mobility layer to select a possible “bridge” device (e.g., the one owned by team member 2) to follow the “going-out-of-range” camera device. In general this may result in a change of the MANET topology. Specifically, the current mobility net and the P/T-net of team member 2 have to be transformed in order to adapt it to the evolving network topology.

Thus, according to the requirements of our scenario, the structure of the token nets in Figs. 1 and 2 has to be changed to react to incoming events, e.g. to avoid a ”going-out-of-range”-situation. In general we consider the change of the net

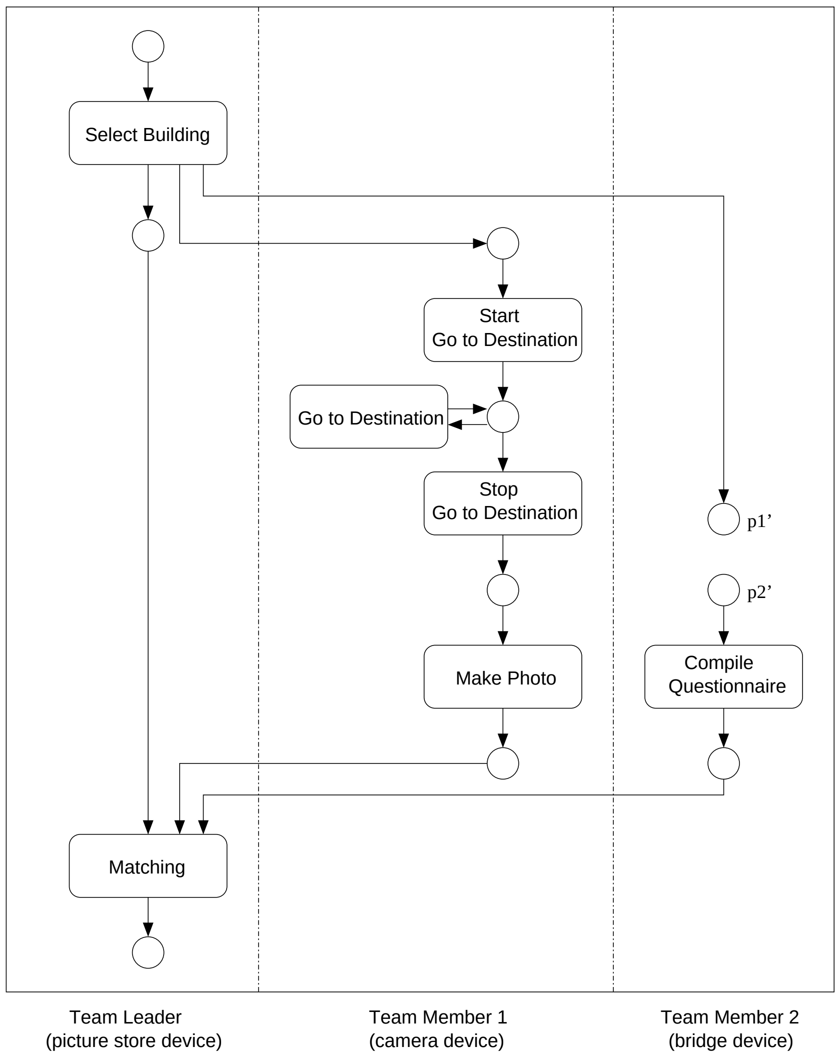


Fig. 3. Teamwork net T_0

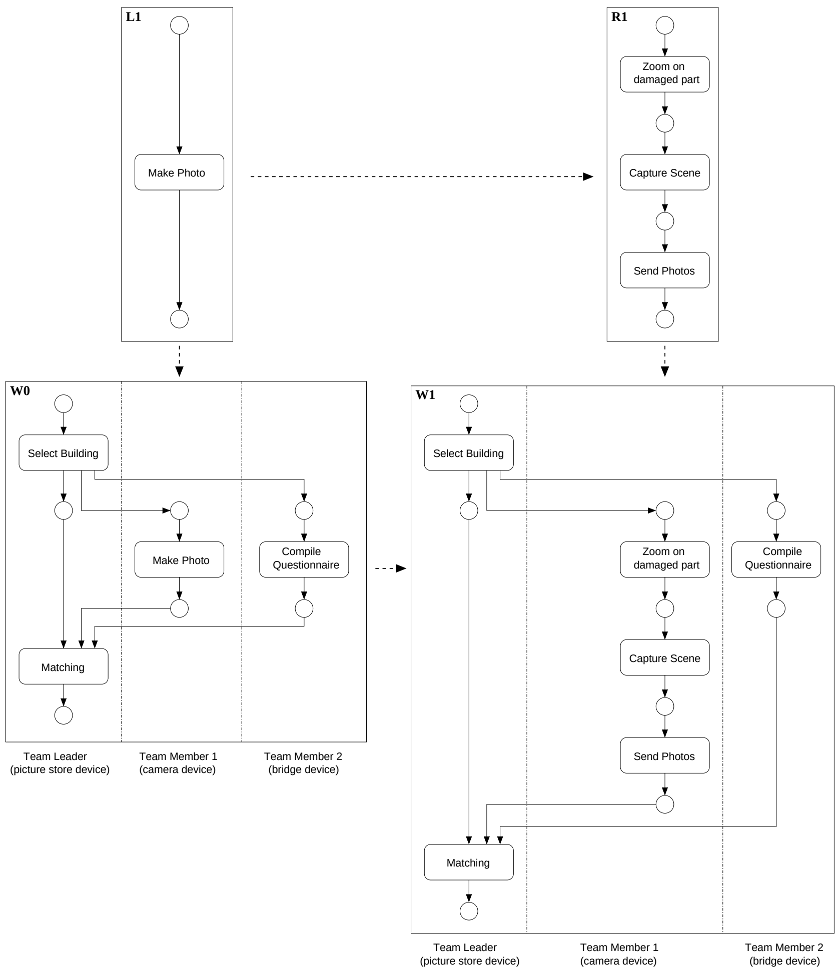


Fig. 4. Rule r_{photo} in the workflow layer and its application

structure as a rule-based transformation of P/T-nets. This theory is inspired by graph transformation systems [15] that were generalized to net transformation systems [7]. The existence of several consistency and compatibility results for net transformation systems is highly profitable for maintaining consistency of workflows in MANETs.

The basic idea behind net transformation systems is the stepwise development of P/T-nets by appropriate rules. Think of these rules as replacement systems, where the left-hand side of a rule is replaced by the right-hand side. A transformation from a P/T-net N_0 to a P/T-net N_1 by a rule r is denoted by $N_0 \stackrel{r}{\Longrightarrow} N_1$.

In our example team member 1 has to refine his/her activity of making photos. For this reason the structure of the workflow W_0 in Fig. 1(a) is changed using the rule r_{photo} depicted in the upper row in Fig. 4 resulting in the new workflow W_1 in Fig. 4. Assume a probable disconnection while team member 1 is going to the previously selected destination. Here the rule r_{follow} in Fig. 5 maintains the network connectivity by adding movement activities for team member 2 to follow team member 1, i.e. $M_0 \stackrel{r_{follow}}{\Longrightarrow} M_1$. Analogously, the net structure of the local view of team member 2 has to be adapted to include these movement activities. So, we provide the rule r_{m2} in Fig. 6 for the team layer to change the structure of the token net $\mathbf{t_0^2}$, i.e. $\mathbf{t_0^2} \stackrel{r_{m2}}{\Longrightarrow} \mathbf{t_1^2}$. Note that these rules are applied independently so that consistent transformations cannot be guaranteed in general. But we present in Section 4 layer consistency conditions to maintain consistency of a layered architecture in MANETs, i.e. after the application of specific rules we have again a consistent layer environment.

First of all, the rule r_{photo} is compatible with place refinement because it preserves all involved places (Cond. 1 in Theorem 1). For the same reason, the rules r_{photo} and r_{follow} are independent of the interface given by the overlapping of the workflow W_0 and the mobility net M_0 (Cond. 2 in Theorem 1). Moreover, we obtain the parallel rule r (see Fig. 7) consisting of both r_{photo} and r_{follow}. In a next step we focus on the rule r_{m2} in Fig. 6, which is compatible on the one hand with the parallel rule r, i.e. the reduction to those activities of rule r being relevant for team member 2 is equivalent to rule r_{m2} (Cond. 3 in Theorem 1); on the other hand the transformation $\mathbf{t_0^2} \stackrel{r_{m2}}{\Longrightarrow} \mathbf{t_1^2}$ is compatible with the transformation $T_0 \stackrel{r}{\Longrightarrow} T_1$, because there is a corresponding inclusion of the resulting token net $\mathbf{t_1^2}$ into the teamwork net T_1 in Fig. 9 (Cond. 4 in Theorem 1). So, we achieve again a consistent layer environment, i.e. the teamwork net T_1 is given by the gluing of the workflow W_1 and the mobility net M_1, and there are inclusions from the restructured local views of team member nets seen in Fig. 8 to the teamwork net T_1.

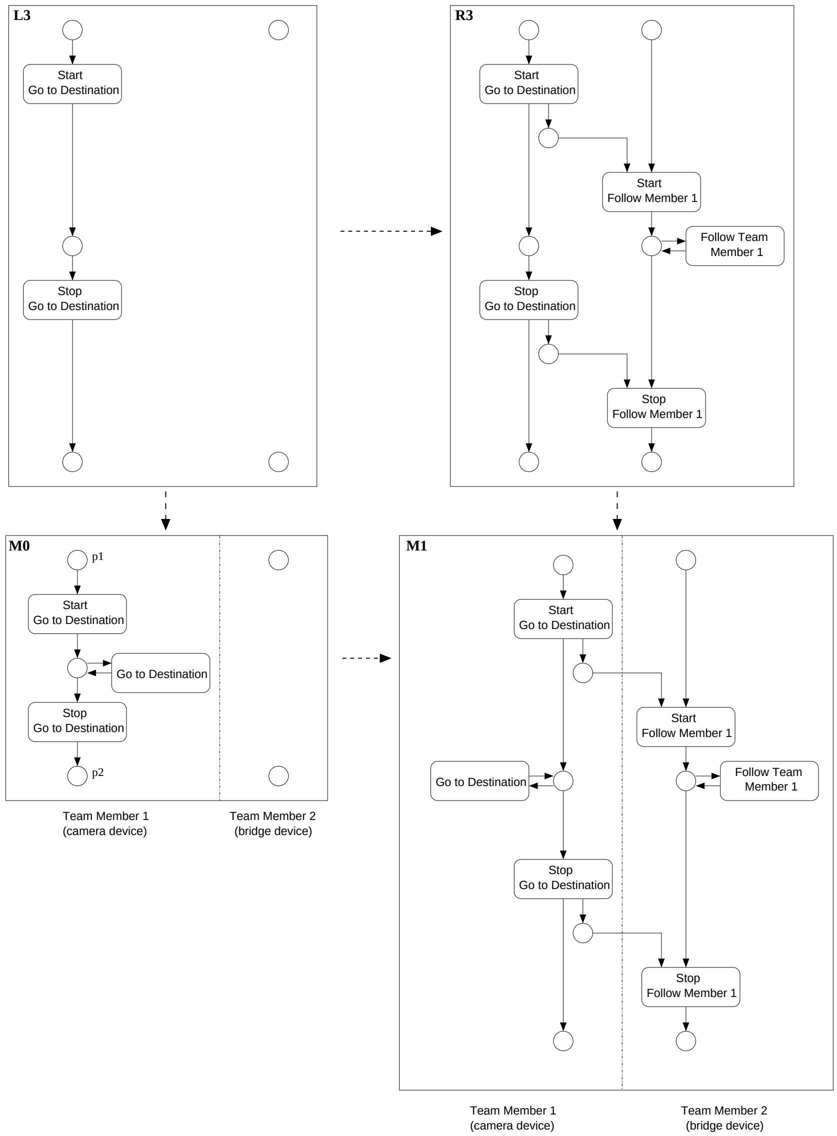


Fig. 5. Rule r_{follow} in the mobility layer and its application

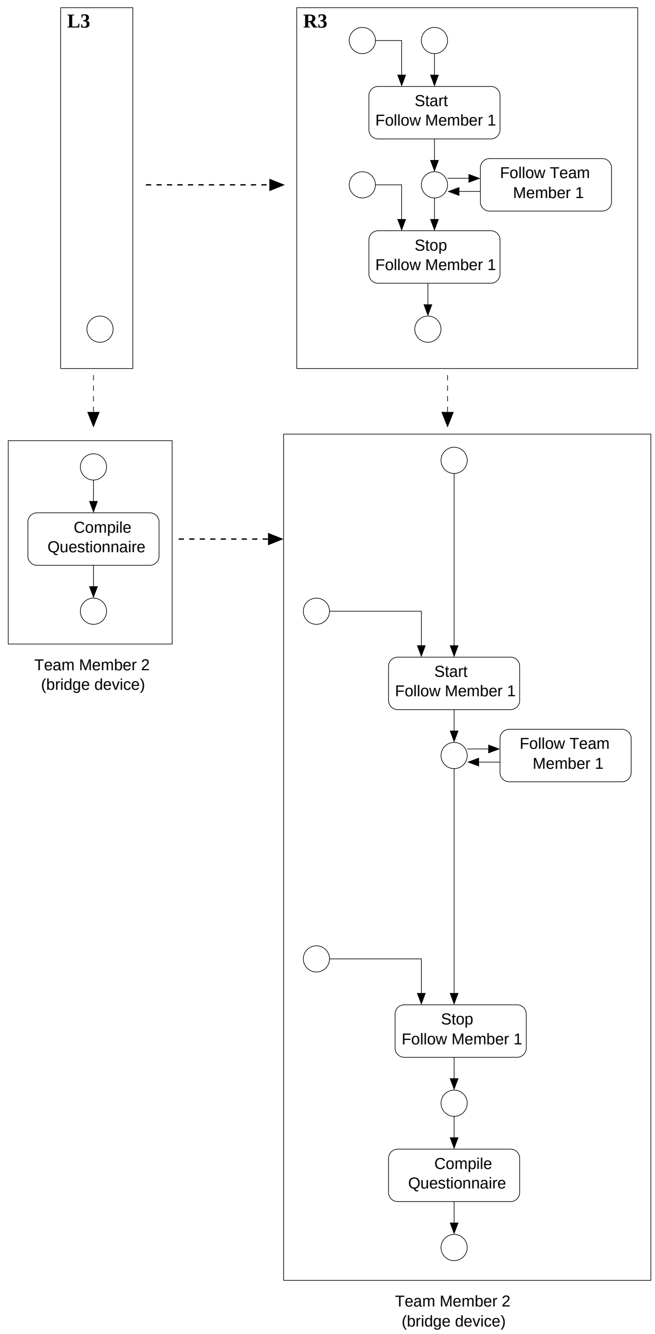


Fig. 6. Rule r_{m2} and its application to $\mathbf{t_0^2}$

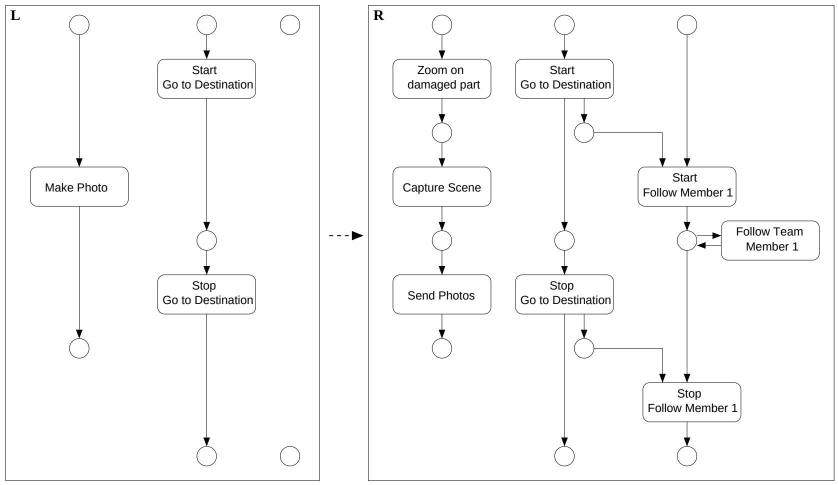


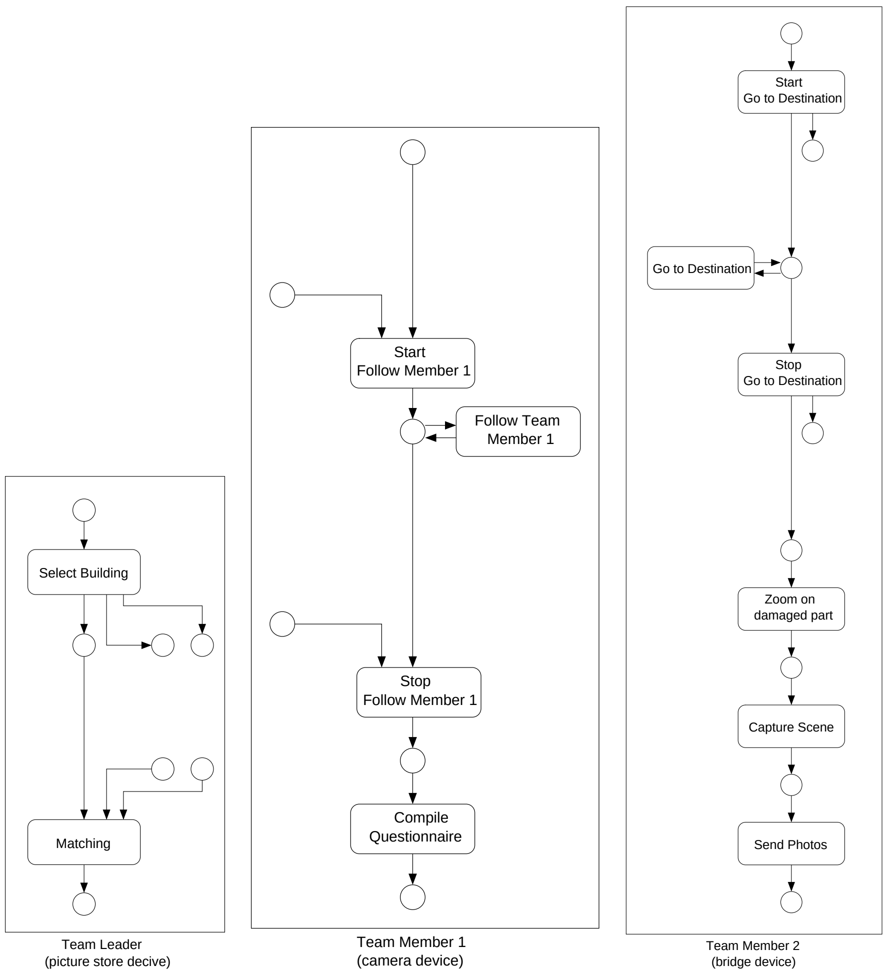
Fig. 7. Parallel rule $r = r_{photo} + r_{follow}$

3 Layered Architectures of Mobile Ad-hoc Networks

In [4] a model for MANETs is described by a global workflow and its transformation by a global set of rules. Following the observation that a workflow in MANETs consists of different aspects we provide a layered architecture as depicted in Fig. 10 to get a more adequate model. We separate movement activities from general activities and allow a local view of team members that is most important in such an unstable environment. From a practical point of view the MANET topology often has to be restructured to maintain the network connectivity resulting in a change of movement activities while general activities are more or less fixed during the workflow execution. Thus, the global workflow, based on a predictive layer, is separated into three different layers. Each of these layers is equipped with its own P/T-nets and transformation rules. The advantage is that we exploit some form of control on rule application by assigning a set of rules to a specific layer. Under these restrictions transformations can be realized in a specific layer of our model.

The predictive layer signals probable disconnections to the upper mobility layer. The predictive layer implements a probabilistic technique [5] that is able to predict whether in the next instant all devices will still be connected. The mobility layer summarizes movement activities of the involved team members and is in charge of managing those situations when a peer is going to disconnect. The team layer realizes the local view of team members onto the workflow and the mobility net. Here, a P/T-net describes those activities being relevant for one team member. Finally, the workflow layer represents in terms of a P/T-net[2]

[2] Note that we have a P/T-net that describes the workflow, but this needs not be a workflow net in the sense of [16].



(a) P/T-net $\mathbf{t_1^0}$ (b) P/T-net $\mathbf{t_1^1}$ (c) P/T-net $\mathbf{t_1^2}$

Fig. 8. Team member nets in team layer after rule application

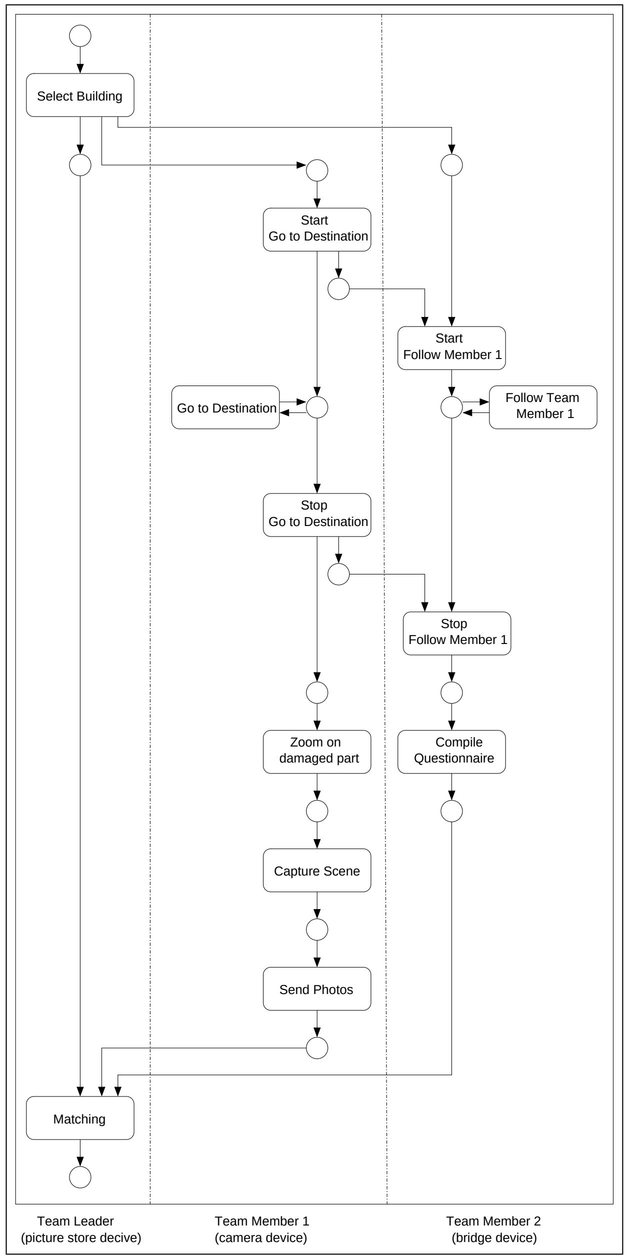


Fig. 9. Teamwork net T_1 after application of rules r_{photo} and r_{follow}

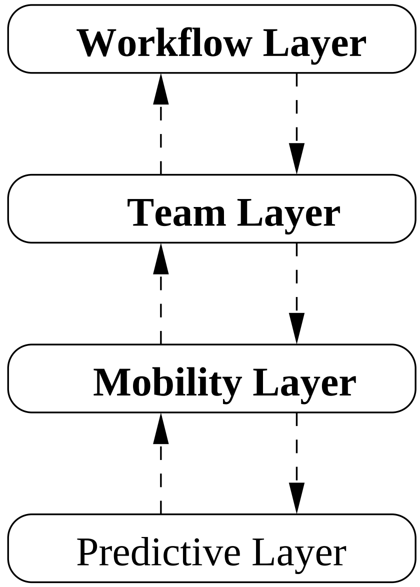


Fig. 10. MANET architecture

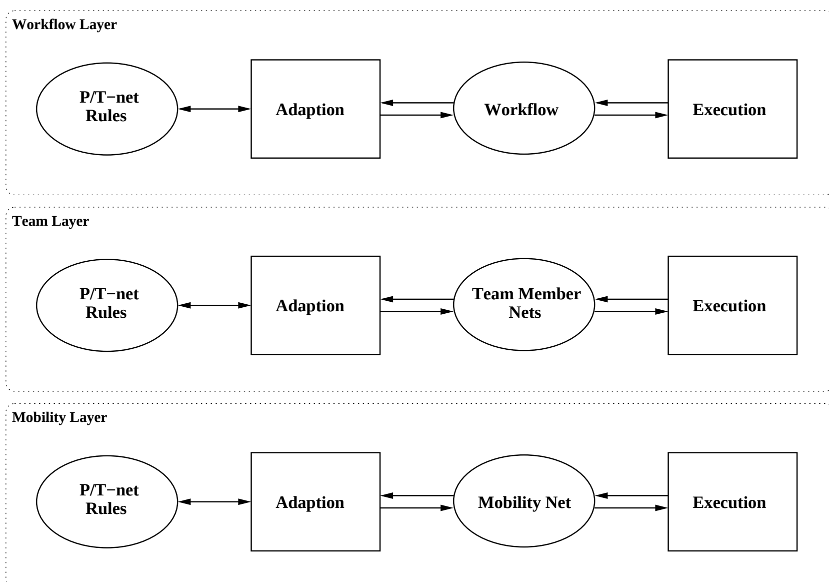


Fig. 11. Layered architecture for supporting cooperative work on MANET

the cooperative work of the team but excludes movement activities. The layered architecture is formalized by a layered AHO net (see Fig. 11 for a schematic view), so that rules in a certain layer are provided for transformations of corresponding P/T-nets, e.g. to react to some incoming events. In general, AHO nets [9] combine an algebraic data type part and Petri nets by the inscription of net elements with terms over the given data structure. Technically, the data type part of the AHO net in Fig. 11 consists of P/T-nets, the well-known token game, rules and rule-based transformation in the sense of the double pushout (DPO) approach [7], where all of them are specified by appropriate sorts and operations. In this way, P/T-nets and rules can be used as tokens in our model, and the token game and rule-based transformations can be implemented in the net inscriptions. Moreover, places in the layered AHO net are either system or rule places, i.e. the state of our model is given by an appropriate marking consisting of token nets and token rules. Token rules are static, i.e. rules represented as tokens do not move and remain unchanged on the corresponding rule places (indicated by the double arrow). In short, firing a transition **Adaption** changes the structure of a corresponding token net according to an appropriate token rule (for details we refer to [9]). Specifically, the mobility layer is in charge of catching disconnection events incoming from the predictive layer and modifying the mobility net (e.g. adding a "Follow Member X" activity) by applying transformation rules.

The P/T-nets presented in Section 2 are possible markings of our AHO net model in Fig. 11. Fig. 1(a) depicts the token net W_0 for the workflow layer, i.e. it represents the current marking of the place **Workflow** in Fig. 11. For the mobility layer the token net M_0 is depicted in Fig. 1(b) (token net on the place **Mobility Net** in Fig. 11). Finally, the team member nets in Fig. 2 are a marking on the place **Team Member Nets** in our model. Note that in general we consider the marking of the token nets. This requires switching from P/T-nets to P/T-systems so that firing a transition **Execution** in our model (see Fig. 11) computes the successor marking of a token net. But in this paper we prefer the notion of P/T-nets because our main results focus on the structure of token nets. Analogously, for each layer a specific transformation rule is depicted in Figs. 4 and 5.

4 Concepts and Results for Layer Consistency

In this section we discuss the basic concepts for maintaining consistency in our approach. Consistency is defined for the layered architecture of workflows in MANETs, that is the *workflow layer*, the *mobility layer* and the *team layer*. We present a notion of consistency, that relates the layers to the team members' activities. Moreover, as discussed in section 2 we have rules and transformations for changes at the level of the workflow layer, of the mobility layer and for changing the individual activities of the team members. These rules and transformations allow the refinement of the workflow according to the imperatives of the network maintenance. To support the local views they have to be applied independently but must allow precise consistency maintenance. So, we give a precise definition of layer consistency and provide precise conditions that allow maintaining consistency. The main theorem states the conditions under which consistency can be maintained stepwise. This result can be extended, so that certain degrees of inconsistency are allowed, while restoring consistency is still possible. In subsection 4.4 we pick up the discussion on maintaining consistency in view of the notions we present subsequently.

4.1 Consistent Layer Environment

Based on the layered architecture for MANETs we have for the *workflow layer* a P/T-net W, for the *mobility layer* a P/T-net M and for the *team work layer* for each team member a P/T-net $\mathbf{t^m}$. For each team member $m = 1, ..., n$ we provide a net $\mathbf{t^m}$ representing their individual activities as well as the relation to the activities of the whole team and rules changing these activities. Here, we assume merely that $\mathbf{t^m}$ are P/T-nets. Alternatively we could require workflow or process nets (see discussion in Section 5). The activities of the team members consist of parts concerning their workflow as well as parts concerning their mobility. Team members can change their team member nets according to specific rules. The main goal of our approach is to model the changes that occur for reasons of the tasks to be achieved as well as the changes that are required because of the mobility issues. To that end we need the workflow W and rules r^W for transforming W, the mobility net M and rules r^M for transforming M as well as each team member's net $\mathbf{t^m}$ and rules $\mathbf{r^m}$ to transform these. These rules are given as net rules and transformations in the DPO approach [7] (see for example the P/T-net rules in Figs. 4 and 5 in Section 2). The nets W, M, and $\mathbf{t^m}$, as well as the rules r^W, r^M and $\mathbf{r^m}$ are the tokens in AHO net depicted in Fig. 11(b). Firing in this AHO net causes the transformation of nets in all three layers at the level of the tokens, i.e. the layer nets and their rules. Consistency of such a layered AHO net means in a broad sense that the workflow W, the mobility net M and the individual team member net $\mathbf{t^m}$ of each team member have to be related as depicted in Fig. 12. The interface net is assumed to be fixed throughout this paper, but it is easy to adapt our constructions to changing the interface as well.

$$\begin{tikzcd}
& I \arrow[dl] \arrow[dr] & & \\
M \arrow[dr] \arrow[rr, phantom, "(\mathbf{PO})"] & & \underline{W} \arrow[dl] \arrow[r, "pg"] & W \\
\mathbf{t^m} \arrow[r, dashed, "\alpha^m"] & T & &
\end{tikzcd}$$

Fig. 12. Consistent layer environment

Definition 1. *A* consistent layer environment *according to the layers in Fig. 11.b is given for the team members' nets* $\mathbf{t^1}, ..., \mathbf{t^n}$*, the workflow* W *and the mobility net* M *if the following conditions are satisfied:*

1. *In order to have refinement of places in* W *with subnets of* M *we allow replacing* W *by* $\underline{W} \xrightarrow{pg} W$*, where* pg *is a place gluing morphism (bijective on transitions and surjective on places).*
2. *There is the fixed interface net* I *included in* M *and* W*, so that a teamwork net* T *is obtained by the gluing of* M *and* W *along* I*, written* $T = M +_I W$*.*

3. *There are activity arrows for each team member* $\mathbf{t^1} \xrightarrow{\alpha^1} T, ..., \mathbf{t^n} \xrightarrow{\alpha^n} T$ *that are injective net morphism relating a team member's activities – given by the net* $\mathbf{t^m}$ *– to the teamwork net* T*.*

The nets $W, M, (\mathbf{t^1}, ..., \mathbf{t^n})$ and T correspond in our example to the nets W_0, M_0 and T_0 given in Figs. 1(a), 1(b) and 3, respectively. $\underline{W}_0$ is obtained from W_0 by splitting **p** in Fig. 1(a) into two places **p1** and **p2** that are unconnected (and similar **p'** into **p1'** and **p2'**). I consists of the places **p1**, **p2**, **p1'** and **p2'** included in $\underline{W}_0$ and also in M_0 in Fig. 1(b) where these places are the entry and exit places.

4.2 Transformations at different layers

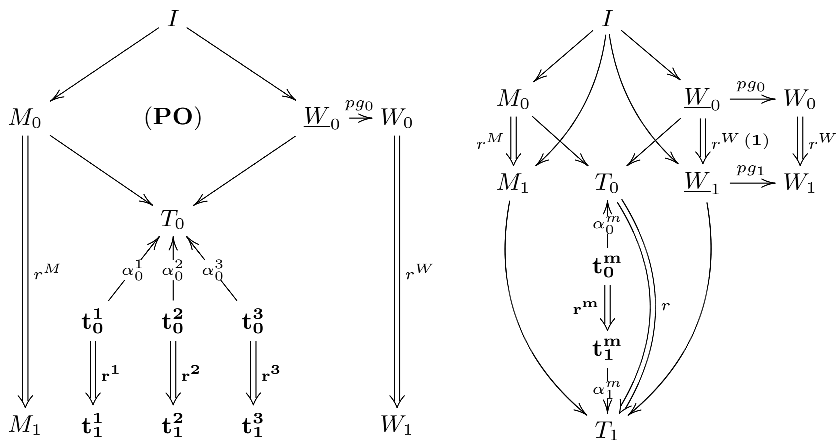
As mentioned before we want to model changes using rules and transformations at the different layers we have. The transformation of the mobility net M, the workflow W and the team members' activities $\mathbf{t^m}$ is achieved using net transformations as illustrated in Section 2. For more details on net transformations see [7].

Example 1. Starting at a consistent layer environment firing of the AHO net transitions **Workflow Adaption** in Fig. 11.b yields various transformations in the different layers. So, at the level of the tokens (i.e. nets and rules) we have then e.g. the situation depicted in Fig 13(a): There are rules in the mobility layer, in the workflow layer and three rules in the team layer that have been applied, yielding the following transformations $M_0 \xRightarrow{r^M} M_1$, $W_0 \xRightarrow{r^W} W_1$ as well as rules for each team member $\mathbf{t^1_0} \xRightarrow{\mathbf{r^1}} \mathbf{t^1_1}$, $\mathbf{t^2_0} \xRightarrow{\mathbf{r^2}} \mathbf{t^2_1}$ and $\mathbf{t^3_0} \xRightarrow{\mathbf{r^3}} \mathbf{t^3_1}$. This is the situation as discussed in Section 2 with the team members' nets $\mathbf{t^0_0}$, $\mathbf{t^1_0}$ and $\mathbf{t^2_0}$.

According to the discussion in Section 1 we now need conditions that allow maintaining consistency. We have to obtain the teamwork net that integrates the changes induced by the transformations above. The results for net transformations (see [7]) yield a variety of independence conditions for the sequential, parallel application of rules and for the compatibility with pushouts. Subsequently we develop the conditions for maintaining layer consistency based on transformations at the mobility and the workflow layer. Later in Cor. 2 we assume not only transformations, but transformation sequences.

Let there be the transformations $W_0 \xRightarrow{r^W} W_1$ and $M_0 \xRightarrow{r^M} M_1$. We first need to ensure *compatibility with place refinement.* This means that the rule r^W is also applicable to $\underline{W}_0$ and there exists a place-gluing morphism $pg_1 : \underline{W}_1 \to W_1$, such that the diagram (1) in Fig. 13(b) commutes.

Provided the *preservation of the interface* I, that is, the applications of the rules r^W and r^M are independent of I, there is the parallel rule $r = r^W + r^M$, so that the application of r to the teamwork net T_0 yields the transformation $T_0 \xRightarrow{r} T_1$, with $T_1 = M_1 +_I \underline{W}_1$. So, the first step to the next consistent layer environment is achieved. Now we restrict the transformation $T_0 \xRightarrow{r} T_1$



(a) State after some transformations

(b) A new consistent layer environment for $m = 1, 2, 3$

Fig. 13. Layer environment after transformations

to the transformations $\mathbf{t_0^m} \overset{\mathbf{r^m}}{\Longrightarrow} \mathbf{t_1^m}$ for each team member $m = 1, ..., n$. Since the team members' activities are represented by activity arrows, the rules have to be compatible with arrows. The *existence of activity rules* ensures that for each team member the rule $r = (L \leftarrow K \rightarrow R)$ is restricted to an activity rule $\mathbf{r^m} = (\mathbf{L^m} \leftarrow \mathbf{K^m} \rightarrow \mathbf{R^m})$, where $\mathbf{K^m}$ has to be the pullback (roughly an intersection) of $\mathbf{L^m}$ and K as well as the pullback of $\mathbf{R^m}$ and K.

Moreover, each activity rule $\mathbf{r^m}$ has to be the reduction of the corresponding rule r to that part being relevant for the team member m. The *conformance of activity rules and team member nets* means that $\mathbf{L^m}$ is additionally the pullback of $\mathbf{t_0^m}$ and L, and the application of an activity rule $\mathbf{r^m}$ to a team member net $\mathbf{t_0^m}$ yields the transformation $\mathbf{t_0^m} \overset{\mathbf{r^m}}{\Longrightarrow} \mathbf{t_1^m}$.

Then we can state our first main result, that provides the conditions for stepwise consistency maintenance.

Definition 2. *Given a rule $r^W = L^W \leftarrow K^W \rightarrow R^W$, a place-gluing morphism $pg_0 : \underline{W}_0 \rightarrow W_0$ and a transformation $W_0 \overset{r^W}{\Longrightarrow} W_1$. The rule r^W is called compatible with place refinement, if it also applicable to $\underline{W}_0$ and there exist a place-gluing morphisms $pg_1 : \underline{W}_1 \rightarrow W_1$ such that the diagram in Fig. 14 comutes.*

Definition 3. *Given a rule $r = L \leftarrow K \rightarrow R$, a rule $\mathbf{r^m} = \mathbf{L^m} \leftarrow \mathbf{K^m} \rightarrow \mathbf{R^m}$ is an activity rule for r, if there exist morphisms $\mathbf{L^m} \rightarrow L$, $\mathbf{K^m} \rightarrow K$ and $\mathbf{R^m} \rightarrow R$ such that $(PB1)$ and $(PB2)$ are pullbacks.*

$$
\begin{tikzcd}
\underline{W}_0 \arrow[r, "pg_0"] \arrow[d, Rightarrow, "r^W"] & W_0 \arrow[d, Rightarrow, "r^W"] \\
\underline{W}_1 \arrow[r, "pg_1"] & W_1
\end{tikzcd}
$$

Fig. 14. Compatibility with place refinement

$$
\begin{tikzcd}
 & L & & K \arrow[ll] \arrow[rr] & & R \\
\mathbf{L^m} \arrow[ur] & & \mathbf{K^m} \arrow[ll] \arrow[rr] \arrow[ur] \arrow[ul, phantom, "(\mathbf{PB1})"] & & \mathbf{R^m} \arrow[ur] \arrow[ul, phantom, "(\mathbf{PB2})"] &
\end{tikzcd}
$$

Fig. 15. *Activity rule*

Definition 4. *Given an activity rule* $\mathbf{r^m} = \mathbf{L^m} \leftarrow \mathbf{K^m} \rightarrow \mathbf{R^m}$ *for* $r = L \leftarrow K \rightarrow R$ *and an activity arrow* $\mathbf{t_0^m} \stackrel{\alpha^{m_0}}{\rightarrow} T_0$*. The activity rule* r^m *is called conformant to activity arrow* α^m *if there exist morphisms* $L \rightarrow T_0$ *and* $L^m \rightarrow \mathbf{t_0^m}$ *such that* $(PB3)$ *in Fig. 16 becomes a pullback.*

Lemma 1. *Given an activity rule* $\mathbf{r^m} = \mathbf{L^m} \leftarrow \mathbf{K^m} \rightarrow \mathbf{R^m}$ *for a rule* $r = L \leftarrow K \rightarrow R$ *and a conformant rule application* $\mathbf{L^m} \rightarrow \mathbf{t_0^m}$*, then there exists an activity arrow* $\mathbf{t_1^m} \stackrel{\alpha_1^m}{\rightarrow} T_1$*.*

Proof. From the preconditions we have the diagram shown in Fig. 16 with the top squares being pullbacks due to Def. 3 and the left square being a pullback due to Def. 4. Next we construct $\mathbf{t'^m_0}$ as the pullback object of $\mathbf{t_0^m} \rightarrow T_0 \leftarrow T_0'$ such that the bottom square becomes a pullback as well. Since $\mathbf{K^m} \rightarrow \mathbf{L^m} \rightarrow \mathbf{t_0^m} \rightarrow T_0 = \mathbf{K^m} \rightarrow K \rightarrow T_0' \rightarrow T_0$ we can construct the morphism $\mathbf{K^m} \rightarrow \mathbf{t'^m_0}$ as the induced pullback morphism over $\mathbf{t'^m_0}$ using the universal property of the bottom pullback such that the left cube commutes. Now we can construct $\mathbf{t_1^m}$ as the pushout object over $\mathbf{R^m} \leftarrow \mathbf{K^m} \rightarrow \mathbf{t'^m_0}$. Finally we obtain the activity arrow $\mathbf{t_1^m} \stackrel{\alpha_1^m}{\rightarrow} T_1$ as the induced pushout morphism over $\mathbf{t_1^m}$ since $\mathbf{K^m} \rightarrow \mathbf{t'^m_0} \rightarrow \mathbf{t_1^m} \rightarrow T_1 = \mathbf{K^m} \rightarrow \mathbf{R^m} \rightarrow R \rightarrow T_1$ such that the right cube commutes. The constructed activity arrow $\mathbf{t_1^m} \stackrel{\alpha_1^m}{\rightarrow} T_1$ can be seen in Fig. 17.

$$
\begin{tikzcd}
 & L \arrow[dd] & & K \arrow[ll] \arrow[rr] \arrow[dd] & & R \arrow[dd] \\
\mathbf{L^m} \arrow[ur] \arrow[dd] & & \mathbf{K^m} \arrow[ll] \arrow[rr] \arrow[ur] \arrow[ul, phantom, "(\mathbf{PB1})"] & & \mathbf{R^m} \arrow[ur] \arrow[ul, phantom, "(\mathbf{PB2})"] & \\
 & T_0 \arrow[ul, phantom, "(\mathbf{PB3})"] & & T_0' \arrow[ll] \arrow[rr] & & T_1 \\
\mathbf{t_0^m} \arrow[ur, "\alpha_0^m"] & & & & &
\end{tikzcd}
$$

Fig. 16. Given activity arrow

$$
\begin{tikzcd}
 & L & & K \arrow[ll] \arrow[rr] \arrow[dd] & & R \arrow[dd] \\
\mathbf{L^m} \arrow[ur] \arrow[dd] & & \mathbf{K^m} \arrow[ll] \arrow[rr] \arrow[ur] \arrow[dd] & & \mathbf{R^m} \arrow[ur] \arrow[dd] & \\
 & T_0 \arrow[from=uu] & & T'_0 \arrow[ll] \arrow[rr] & & T_1 \\
\mathbf{t_0^m} \arrow[ur, "\alpha_0^m"] & & \mathbf{t'^m_0} \arrow[ll] \arrow[rr] \arrow[ur, "\alpha'^m_0"] & & \mathbf{t_1^m} \arrow[ur, "\alpha_1^m"] &
\end{tikzcd}
$$

Fig. 17. Constructed activity arrow

Corollary 1. *Given the cubes in Fig. 17, the front squares are pushout, i.e. given the rule* $\mathbf{r^m} = \mathbf{L^m} \leftarrow \mathbf{K^m} \rightarrow \mathbf{R^m}$ *we get* $\mathbf{t_0^m} \overset{\mathbf{r^m}}{\Longrightarrow} \mathbf{t_1^m}$.

Proof.
The top, left and bottom squares in the left cube of Fig. 17 are pullbacks either by preconditions or by construction. By composing the top and left pullback we also have pullback $PB1$ in Fig. 18. Since the bottom square is also a pullback, the right square in the left cube becomes a pullback due to pullback decomposition since the outer and right diagram are pullbacks as seen in Fig. 19. Petri nets are a weak adhesive category as shown in [8] and [6], and since all rule morphisms and the activity arrows are injective, the left back square is a weak van Kampen square. Using the VK-property (see App. A, Definition 5) in the left cube of Fig. 17 with the back square being a pushout with $K \rightarrow L$ injective and the top, left, right and bottom squares being pullbacks we obtain that the front square is a pushout.
The front square in the right diagram is a pushout by construction. Due to the VK-property using the front and back squares being pushouts and the top and left side square being pullbacks we get that the right bottom square and the right side square are pullbacks.

$$
\begin{tikzcd}
\mathbf{K^m} \arrow[r] \arrow[d] \arrow[dr, phantom, "(\mathbf{PB1})"] & \mathbf{t_0^m} \arrow[d] \\
K \arrow[r] & T_0
\end{tikzcd}
$$

Fig. 18. PB composition

$$
\begin{tikzcd}
\mathbf{K^m} \arrow[r] \arrow[d] & \mathbf{t'^m_0} \arrow[r] \arrow[d] \arrow[dr, phantom, "(\mathbf{PB3})"] & \mathbf{t_0^m} \arrow[d] \\
K \arrow[r] & T'_0 \arrow[r] & T_0
\end{tikzcd}
$$

Fig. 19. PB decomposition

Theorem 1 (Stepwise Consistency Maintenance). *Given a consistent layer environment* $T_0 = M_0 +_I \underline{W}_0$ *with the place gluing* $\underline{W}_0 \overset{pg_0}{\rightarrow} W_0$ *and the activity arrows* $\mathbf{t_0^m} \overset{\alpha_0^m}{\rightarrow} T_0$ *for each member* $m = 1, ..., n$*, then the transformations* $W_0 \overset{r^W}{\Longrightarrow} W_1$, $M_0 \overset{r^M}{\Longrightarrow} M_1$ *and the transformations* $\mathbf{t_0^m} \overset{\mathbf{r^m}}{\Longrightarrow} \mathbf{t_1^m}$ *yield again a consistent layer environment* $T_1 = M_1 +_I \underline{W}_1$ *with the place gluing* $\underline{W}_1 \overset{pg_1}{\rightarrow} W_1$ *and* *the activity arrows* $\mathbf{t_1^m} \overset{\alpha_1^t}{\to} T_1$ *for each* m*, provided the following* layer consistency conditions *hold:*

1. *compatibility with the place refinement, i. e. the rule* r^W *is compatible with the morphism* pg_0*,*
2. *preservation of the interface* I*, i.e. the application of the rules* r^W *and* r^M *are independent of* I*,*
3. *existence of activity rules, i.e. for each* m *there are activity rules* $\mathbf{r^m}$ *over the parallel rule* $r = r^W + r^M$ *and*
4. *conformance of activity rules and team member nets, i.e.* $\mathbf{t_0^m} \overset{\mathbf{r^m}}{\Longrightarrow} \mathbf{t_1^m}$ *is compatible with* $T_0 \overset{r}{\Longrightarrow} T_1$*.*

Proof. We have to proof that given the conditions stated in the theorem the three properties of a consistent layer environment are fulfilled:

1. There exists a place gluing morphism $pg_1 : \underline{W}_1 \to W_1$.
2. The result of the application of the parallel rule $r = (r^M + r^W)$ to T_0 is equal to the gluing of W_1 and M_1 along I.
3. There exist activity arrows $\alpha_1^m : \mathbf{t_1^m} \to T_1$ for each team member net $\mathbf{t_1^m}$.

1. First we will show the existence of the place gluing morphism $pg_1 : \underline{W}_1 \to W_1$ as depicted in Fig. 13(b).
Because Cond. 1 states that rule r^W is compatible with place refinement, we can obtain pg_1 via Def. 2.
2. Next we show that the result of applying the parallel rule $r = (r^M + r^W)$ to T_0 is equal to the gluing of W_1 and M_1 along I. The interface I is preserved by the application of r^M and r^W due to Cond. 2 of the theorem. Using the Union Theorem (see App. A, Theorem 3) we know that the gluing is compatible with the transformation, such that $T_1 = M_1 +_I W_1$ is equal to $T_0 \overset{r}{\Longrightarrow} T_1$.
3. Finally we have to show that there are activity arrows $\alpha_1^m : \mathbf{t_1^m} \to T_1$ for each team member net $\mathbf{t_1^m}$. With Cond. 3 and Cond. 4 and Lemma 1 an activity arrow α_1^m can be constructed.

Example 2. Considering the example in Section 2, outlined in Fig. 13(a) we have the following situation: The rule r_{photo} is compatible with place refinement because it preserves all involved places. For the same reason, the rules r_{photo} and r_{follow} are independent of the interface given by the overlapping of the workflow $\underline{W}_0$ and the mobility net M_0 and we obtain the parallel rule r consisting of both r_{photo} and r_{follow}.

In a next step we focus on the rule r_{m2} in Fig. 6 that is compatible on the one hand with the parallel rule r, i.e. the reduction to those activities of rule r being relevant for team member 2 is equivalent to rule r_{m2}; on the other hand the transformation $\mathbf{t_0^2} \overset{r_{m2}}{\Longrightarrow} \mathbf{t_1^2}$ is compatible with the transformation $T_0 \overset{r}{\Longrightarrow} T_1$, because there is a corresponding inclusion of the resulting token net $\mathbf{t_1^2}$ into the teamwork net T_1. Thus, we have the pushout $T_1 = M_1 +_I \underline{W}_1$ and the construction of the activity rule for each team member yields the activity arrows $\mathbf{t_1^m} \overset{\alpha_1^m}{\to} T_1$. So, we obtain the consistent layer environment depicted in Fig 13(b), where $r^W = r_{photo}$ and $r^M = r_{follow}$.

Corollary 2 (Restoring Consistency). *Given a consistent layer environment, shortly* ($\mathbf{t_0^m} \overset{\alpha_0^m}{\to} T_0 = M_0 +_I \underline{W}_0 \overset{pg_0}{\to} W_0$) *and transformation sequences* $M_0 \overset{*}{\Longrightarrow} M_{n_M}$ *via* r_i^M *and* $W_0 \overset{*}{\Longrightarrow} W_{n_W}$ *via* r_j^W *and the transformation steps* $\mathbf{t_0^m} \overset{\mathbf{r^m}}{\Longrightarrow} \mathbf{t_1^m}$ *leading to a possibly inconsistent state depicted (see Fig. 20). Then it is possible to get an intermediate layer consistent environment* ($\mathbf{t_1^m} \overset{\alpha_1^m}{\to} T_1 = M +_I \underline{W} \overset{pg}{\to} W$) *and a next consistent layer environment* $\mathbf{t_2^m} \overset{\alpha_2^m}{\to} T_2 = M_{n_M} +_I \underline{W}_{n_W} \overset{pg_{n_W}}{\to} W_{n_W}$ *if the following conditions hold:*

1. *The transformation sequence* $M_0 \overset{*}{\Longrightarrow} M_{n_M}$ *can be decomposed, such that* $(M_0 \overset{*}{\Longrightarrow} M_{n_M}) = (M_0 \overset{r^M}{\Longrightarrow} M \overset{\bar{r}^M}{\Longrightarrow} M_{n_M})$ *for suitable rules* r^M *and* $\bar{r}^M$*.*
2. *The transformation sequence* $W_0 \overset{*}{\Longrightarrow} W_{n_W}$ *can be decomposed, such that* $(W_0 \overset{*}{\Longrightarrow} W_{n_W}) = (W_0 \overset{r^W}{\Longrightarrow} W \overset{\bar{r}^W}{\Longrightarrow} W_{n_W})$ *for suitable rules* r^W *and* $\bar{r}^W$*.*
3. *The layer consistency conditions in Theorem 1 hold for* (r^W, r^M) *with* $\mathbf{r^m}$*.*
4. *There exist transformation steps* $\mathbf{t_1^m} \overset{\mathbf{\bar{r}^m}}{\Longrightarrow} \mathbf{t_2^m}$ *such that the layer consistency conditions in Theorem 1 hold for* $(\bar{r}^M, \bar{r}^W)$ *with* $\bar{r}^m$*.*

Proof. First we proof the consistency of the intermediate state reached via the transformation sequences $M_0 \overset{r^M}{\Longrightarrow} M$ and $W_0 \overset{r^W}{\Longrightarrow} W$. Since the rules r^M and r^W fulfill the conditions stated in Theorem 1, we obtain the activity arrows $\mathbf{t_1^m} \overset{\alpha_1^m}{\to} T_1$, the place gluing morphism $\underline{W} \overset{pg}{\to} W$ and T_1 as the gluing of $M +_I \underline{W}$. Thus, we have the consistent layer environment ($\mathbf{t_1^m} \overset{\alpha_1^m}{\to} T_1 = M +_I \underline{W} \overset{pg}{\to} W$). The next layer environment $\mathbf{t_2^m} \overset{\alpha_2^m}{\to} T_2 = M_{n_M} +_I \underline{W}_{n_W} \overset{pg_{n_W}}{\to} W_{n_W}$ is also consistent since $\bar{r}^M$ and $\bar{r}^W$ also fulfill the conditions in Theorem 1 (see Fig. 21).

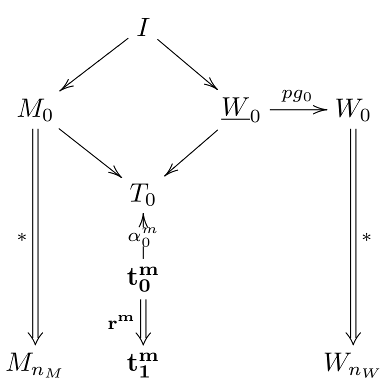


Fig. 20. Possibly inconsistent

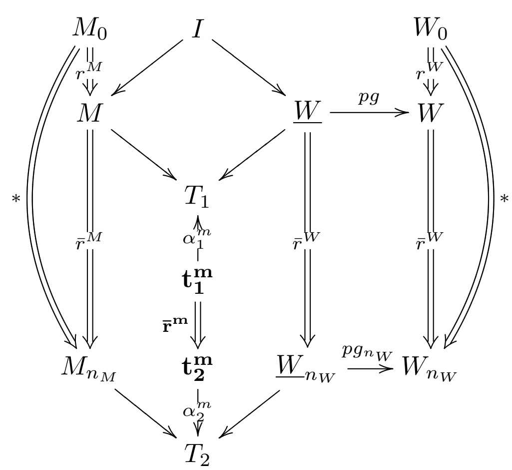


Fig. 21. Restoring the next consistent layer environment

4.3 Transformation on the interface

In the previous subsections the interface has remained fixed which has the disadvantage that both the mobility net and the workflow net might contain a number of seperated places from the beginning. Thus, we also take transformations of the interface I into account, i.e. given a transformation $I \stackrel{p}{\Longrightarrow} J$ and transformations of the mobility and workflow nets we yield a next consistent layer environment. For that purpose we need the notion of specialization *spec*(p) for the given transformation. A spezialization of a transformation can be considered as a new rule, which can again be applied to another net (see Def. 4 in App. A).

Theorem 2. *Given a consistent layer environment* $T_0 = M_0 +_I \underline{W}_0$ *with the place gluing* $\underline{W}_0 \stackrel{pg_0}{\to} W_0$*, the activity arrows* $\mathbf{t_0^m} \stackrel{\alpha_0^m}{\to} T_0$*, a transformation* $I \stackrel{p}{\Longrightarrow} J$ *and transformations* $M_0 \stackrel{spec(p)}{\Longrightarrow} M_1$*,* $W_0 \stackrel{spec(p)}{\Longrightarrow} W_1$*, then the transformations yield again a consistent layer environment* $T_1 = M_1 +_J \underline{W}_1$ *with the place gluing* $\underline{W}_1 \stackrel{pg_1}{\to} W_1$ *and the activity arrows* $\mathbf{t_1^m} \stackrel{\alpha_1^t}{\to} T_1$ *for each* m*, provided the following conditions hold:*

1. *compatibility with the place refinement, i. e. the specialization rule spec*(p) *is compatible with the morphism* pg_0*,*
2. *existence of activity rules, i.e. for each* m *there are activity rules* $\mathbf{r^m}$ *over* p *and*
3. *conformance of activity rules and team member nets, i.e.* $\mathbf{t_0^m} \stackrel{\mathbf{r^m}}{\Longrightarrow} \mathbf{t_1^m}$ *is compatible with* $T_0 \stackrel{p}{\Longrightarrow} T_1$*.*

Proof. The activity arrows $\mathbf{t_1^m} \stackrel{\alpha_1^t}{\to} T_1$ and the place gluing morphism $\underline{W}_1 \stackrel{pg_1}{\to} W_1$ are given by Lemma 1. With the given specialization *spec*(p) it is possible to obtain T_1 as the union of $M_1 +_J \underline{W}_1$ by using the Union Theorem for Interfaces (see App. A). The result is a new consistent layer environment $T_1 = M_1 +_J \underline{W}_1$ with the place gluing $\underline{W}_1 \stackrel{pg_1}{\to} W_1$ and the activity arrows $\mathbf{t_1^m} \stackrel{\alpha_1^m}{\to} T_1$.

4.4 Maintaining Consistency

The notions and results we have introduced above concern the fundamental understanding of consistency in MANETs. As mentioned in the introduction other notions of consistency are possible and desirable. The AHO net model given in Fig. 11.b merely presents the rough structure but abstracts especially from the details of the firing conditions. The exact formulation of the firing conditions models the way the rules are applied in the different layers. Hence the formulation of the firing conditions of the AHO net constitutes the way consistency is dealt with. The discussion below abstracts from realization issues, as e.g. the complexity of the task to find morphisms between nets. Considering the possibilities discussed in the introduction we have:

– *Checking consistency:* The AHO net in Fig. 11.b allows the application of arbitrary rules and it can be checked for a consistent layer environment. Since we have a formal definition of consistency, it can be checked whether a certain state of an AHO net model for MANETs is a consistent layer environment. There need to be the fixed interface I, the token nets M and W on the places **Mobility Net** and **Workflow**, respectively and the token nets $\mathbf{t^m}$ for each team member m on the place **Team Member Nets**, so that they present a consistent layer environment. This means there are nets T and $\underline{W}$, so that there is a place gluing morphism $\underline{W} \to W$, T is the gluing of M and $\underline{W}$ along I and there are m activity arrows $\mathbf{t^m} \stackrel{\alpha^m}{\to} T$.
– *Guaranteed consistency:* Theorem 1 ensures transformations so that each state is consistent. Then the AHO net in Fig. 11.b may allow only the application of rules that satisfy these conditions. Moreover, the parallel firing of the transitions in the different layers has to be ensured to have consistency in each state.
– *Backtracking:* Since all rules are symmetric (as one of the characteristics of the DPO approach) the inverse rules can be applied in the inverse order. Then the AHO net in Fig. 11.b may allow the application of arbitrary rules, but requires a storage of the transformations. Then an explicit backtracking can be achieved by firing the transitions in the AHO net but using only the inverse rules.
– *Restoring consistency:* Corollary 2 gives conditions for restoring consistency. Then the AHO net in Fig. 11.b may allow only the application of rules that satisfy these conditions. An explicit restoration is possible using the transformations constructed in the corollary. Note that here we merely treat transformation sequences for the mobility and the workflow layer. Restoring consistency after transformation sequences at the team layer is very closely related to the question of team work consistency (see Section 5 for a short discussion)

5 Conclusion

The use of a layered architecture for modeling workflows in MANETs has the advantage of separating different views with different granularity, but rises the question of consistency immediately. In this paper, we have presented the notion of layer consistent environment stating that the views in the workflow layer, the mobility layer and the team layer fit together. Since the main modeling advantage of AHO nets is the possibility to model net transformations we have introduced maintenance means for the AHO net for workflows in MANETs that take changes modeled by net transformation into account.

Related work on distribution of workflows in a possibly mobile setting can be found e.g. in [3, 11, 18] where a unique workflow is divided on the one hand in different autonomous workflows and on the other hand the resulting workflows are adapted by using inheritance resp. graph rules. In contrast we present a layered architecture, where a global workflow and its transformation are separated into three different parts, each of them relevant for a specific aspect of workflows in MANETs.

Outlook In this paper we have presented the first results of a larger research activity[3] concerning formal modeling and analysis of MANETs. So, there is a large amount of most interesting and relevant open questions. The subsequent issues concern questions directly related to the work presented here:

Behaviour of token nets The behaviour of the token nets has been treated in previous papers [4] and has been deliberately excluded here. The nets in the different layers have their own behaviour that is executed by firing the corresponding transitions in the AHO net (see Fig. 11.b). This directly leads to a most challenging consistency issue, namely how the individual processes related to each other. A very elegant solution would be to use the theory for open nets [2].

Team work consistency Other relevant notions of consistency concern e.g. the consistency between each team member's activities and the complete teamwork. It should be ensured that the team members' activities together cover the complete team work. This can be realized in our categorical approach using a given topology graph to glue the team members' nets together, then team consistency is given if this gluing corresponds to the teamwork net T. Another possiblity is to use the activity arrows itself and demand them to be jointly surjective, such that the whole teamwork net T is covered by at least one team member. Then again, team consistency has to be maintained during transformations in the different layers.

Restriction of activities In this paper we have used arbitrary P/T-nets without further restrictions for modeling the layers as well as the team members' activities. Nevertheless, syntactic restrictions, e.g restricting the team members' activities to (non)-deterministic processes as well as semantic restrictions, e.g. using the approach of workflow nets in the sense of [16] for all involved nets, may be useful. The restriction of the P/T-nets in the different layers requires some additional treatment. To restrict team members' activities to (non)-deterministic processes the approach to the categorical formulation of processes of (open) nets in [2] can be adopted successfully. The team members' activities are then given by a process of the teamwork net. The technical constructions we presented in this paper are compatible with the process notions, mainly since the projection of processes along injections are given by pullbacks as well.

Property preserving rules Especially in the area of workflow modeling properties like safety and liveness are of importance. In [13,17] inheritance preserving rules and property preserving rules, respectively, are formalized, so that restructuring of workflows preserves properties. Thus, another interesting aspect of future work is to study an integration of preserving rules into the AHO net in Fig. 11.b. To do that, on the one hand the set of token rules would have to be restricted to these kinds of rules and on the other hand the firing conditions would have to be adequately specified.

---

[3] The research project *Formal modeling and analysis of flexible processes in mobile ad-hoc networks* (forMA$_{\text{l}}$NET) of the German Research Council.

Tool support We plan to develop a tool for our approach. For the application of net transformation rules, this tool will provide an export to AGG [1], a graph transformation engine as well as a tool for the analysis of graph transformation properties like termination and rule independence. Furthermore, the token net properties could be analyzed using the Petri Net Kernel [10], a tool infrastructure for Petri nets different net classes.

References

1. AGG Homepage. http://tfs.cs.tu-berlin.de/agg.
2. P. Baldan, A. Corradini, H. Ehrig, and R. Heckel. Compositional Semantics of Open Petri Nets based on Deterministic processes. *MSCS*, 15(1):1–35, 2005.
3. Luciano Baresi, Andrea Maurino, and Stefano Modafferi. Workflow partitioning in mobile information systems. In *IFIP TC 8 Working Conference on Mobile Information Systems*, pages 93–106, 2004.
4. P. Bottoni, F. De Rosa, K. Hoffmann, and M. Mecella. Applying Algebraic Approaches for Modeling Workflows and their Transformations in Mobile Networks. *Mobile Information Systems*, 2(1):51–76, 2006.
5. F. De Rosa, A. Malizia, and M. Mecella. Disconnection Prediction in Mobile Ad hoc Networks for Supporting Cooperative Work. *IEEE Persavive Computing*, 4(3), 2005.
6. H. Ehrig, K. Ehrig, U. Prange, and G. Taentzer. *Fundamentals of Algebraic Graph Transformation*. EATCS Monographs in Theoretical Computer Science. Springer, 2006.
7. H. Ehrig and J. Padberg. Graph grammars and Petri net transformations. In *Lectures on Concurrency and Petri Nets Special Issue Advanced Course PNT*, volume 3098 of *Lecture Notes in Computer Science*, pages 496–536. Springer, 2004.
8. H. Ehrig and U. Prange. Weak Adhesive High-Level Replacement Categories and Systems: A Unifying Framework for Graph and Petri Net Transformations. In K. Futatsugi, J.-P. Jouannaud, and J. Meseguer, editors, *Algebra, Meaning and Computation. Essays Dedicated to J.A. Goguen*, volume 4060, pages 235–251, 2006.
9. K. Hoffmann, T. Mossakowski, and H. Ehrig. High-Level Nets with Nets and Rules as Tokens. In *Proc. of 26th Intern. Conf. on Application and Theory of Petri Nets and other Models of Concurrency*, volume 3536 of *Lecture Notes in Computer Science*, pages 268–288. Springer, 2005.
10. E. Kindler and M. Weber. The Petri Net Kernel - An Infrastructure for Building Petri Net Tools. *Software Tools for Technology Transfer*, 3(4):486–497, 2001.
11. Andrea Maurino and Stefano Modafferi. Partitioning rules for orchestrating mobile information systems. *Personal and Ubiquitous Computing*, 9(5):291–300, 2005.
12. J. Padberg, H. Ehrig, and L. Ribeiro. Algebraic high-level net transformation systems. *Mathematical Structures in Computer Science*, 5:217–256, 1995.
13. J. Padberg and M. Urbasek. Rule-Based Refinement of Petri Nets: A Survey. In *Advances in Petri nets: Petri Net Technologies for Modeling Communication Based Systems*, volume 2472 of *Lecture Notes in Computer Science*, pages 161–196. Springer, 2003.
14. F. Parisi-Presicce. On modifying high level replacement systems. *Electr. Notes Theor. Comput. Sci.*, 44(4), 2001.
15. G. Rozenberg. *Handbook of Graph Grammars and Computing by Graph Transformations, Volume 1: Foundations*. World Scientific, 1997.

16. W. M. P. van der Aalst and K. van Hee. *Workflow Management: Models, Methods, and Systems.* MIT Press, 2001.
17. Wil M. P. van der Aalst and Twan Basten. Inheritance of workflows: an approach to tackling problems related to change. *Theor. Comput. Sci.*, 270(1-2):125–203, 2002.
18. Wil M. P. van der Aalst and Mathias Weske. The P2P Approach to Interorganizational Workflows. In *Proc. of 13th Intern. Conf. on Advanced Information Systems Engineering*, volume 2068 of *Lecture Notes in Computer Science*, pages 140–156. Springer, 2001.

A Formal Foundation

Definition 5 (Van kampen property [6]).
A pushout (1) *is a van Kampen (VK) square if for any commutative cube* (2) *with* (1) *in the bottom and back faces being pullbacks holds: the top is pushout* $\Leftrightarrow$ *the front faces are pullbacks.*

$$\begin{tikzcd}
& A \arrow[dl] \arrow[dr] & \\
B \arrow[dr] \arrow[rr, phantom, "(1)"] & & C \arrow[dl] \\
& D &
\end{tikzcd}
\qquad
\begin{tikzcd}
& & C' \arrow[dll] \arrow[dr] \arrow[dd] & \\
A' \arrow[dd] \arrow[dr] & & & B' \arrow[dll] \arrow[dd] \\
& D' \arrow[dd] & C \arrow[dll] \arrow[dr] & \\
A \arrow[dr] & & & B \arrow[dll] \\
& D & &
\end{tikzcd}$$

Theorem 3 (Union theorem [12]).
Given two Petri nets N_1*,* N_2*, an interface net* I *and two transformations* $N_1 \overset{r_1}{\Longrightarrow} M_1$ *and* $N_2 \overset{r_2}{\Longrightarrow} M_2$*, such that the union* $N_1 +_I N_2 = N_3$ *is parallel independent from* r_1 *and* r_2*, then the following diagram commutes:*

$$\begin{tikzcd}
N_1, N_2 \arrow[r, tail, "I"] \arrow[d, Rightarrow, "{r_1,r_2}"'] \arrow[dr, phantom, "(=)"] & N_3 \arrow[d, Rightarrow, "r_1+r_2"] \\
M_1, M_2 \arrow[r, tail, "I"] & M_3
\end{tikzcd}$$

Theorem 4 (Specialization [14]).
Let $r = (L \leftarrow K \rightarrow R)$ *be a rule and* $G \overset{(r,m)}{\Longrightarrow} H$ *be a transformation with match morphism* $L \xrightarrow{m} G$*. Then* $spec(r) = (G \leftarrow D \rightarrow H)$ *is a new rule obtained from* r *by specializing the context of application via the morphism* $K \rightarrow D$*.*

$$\begin{tikzcd}
L \arrow[d, "m"'] & K \arrow[l] \arrow[r] \arrow[d] & R \arrow[d] \\
G \arrow[d, "m'"'] & D \arrow[l] \arrow[r] \arrow[d] & H \arrow[d] \\
G' & D' \arrow[l] \arrow[r] & H'
\end{tikzcd}$$

If $(spec(r), m') : G' \Longrightarrow H'$ *then* $(r, m \circ m') : G' \Longrightarrow H'$*.*

Theorem 5 (Union theorem for interfaces).
Given a union $G_1 +_I G_2 = G$, the transformation $I \overset{r}{\Longrightarrow} J$ and the transformations $G_1 \overset{spec(r)}{\Longrightarrow} H_1$ and $G_2 \overset{spec(r)}{\Longrightarrow} H_2$ via the spezialization of r (see Theorem 4), then we have $G \overset{spec(r)}{\Longrightarrow} H$ and $H_1 +_J H_2 = H$ so that the diagram below commutes.

$$
\begin{tikzcd}
G_1, G_2 \arrow[r, tail, "I"] \arrow[d, Rightarrow, "{spec(r),spec(r)}"'] \arrow[dr, phantom, "(=)"] & G \arrow[d, Rightarrow, "spec(r)"] \\
H_1, H_2 \arrow[r, tail, "J"] & H
\end{tikzcd}
$$

Proof. We have the pushouts (1), (2) and (3) as well as the pushouts (9) and (10) (given by $G_1 \overset{spec(r)}{\Longrightarrow} H_1$) and the pushouts (11) and (12) (given by $G_2 \overset{spec(r)}{\Longrightarrow} H_2$). Then we obtain the diagram depicted below:

$$
\begin{tikzcd}
& L \arrow[d] \arrow[drrr, phantom, "\mathbf{(1)}"] & & & K \arrow[lll] \arrow[rrr] \arrow[d] \arrow[drrr, phantom, "\mathbf{(2)}"] & & & R \arrow[d] & \\
& I \arrow[dl] \arrow[ddr] & & & D \arrow[lll] \arrow[rrr] \arrow[dl] \arrow[ddr] & & & J \arrow[dl] \arrow[ddr] & \\
G_1 \arrow[ddr] \arrow[dr, phantom, "\mathbf{(3)}"] & & & C_1 \arrow[lll] \arrow[rrr] & & & H_1 & & \\
& & G_2 \arrow[dl] & & & C_2 \arrow[lll] \arrow[rrr] & & & H_2 \\
& G & & & & & & &
\end{tikzcd}
$$

Next we construct pushout C as square (4) over $C_1 \leftarrow D \rightarrow C_2$ and pushout H as square (5) over $H_1 \leftarrow J \rightarrow H_2$.

$$
\begin{tikzcd}
& I \arrow[dl] \arrow[ddr] & & & D \arrow[lll] \arrow[rrr] \arrow[dl] \arrow[ddr] & & & J \arrow[dl] \arrow[ddr] & \\
G_1 \arrow[ddr] \arrow[dr, phantom, "\mathbf{(3)}"] & & & C_1 \arrow[lll] \arrow[rrr] \arrow[ddr] \arrow[dr, phantom, "\mathbf{(4)}"] & & & H_1 \arrow[ddr] \arrow[dr, phantom, "\mathbf{(5)}"] & & \\
& & G_2 \arrow[dl] & & & C_2 \arrow[lll] \arrow[rrr] \arrow[dl] & & & H_2 \arrow[dl] \\
& G & & & C & & & H &
\end{tikzcd}
$$

So now we have $G_1 \overset{spec(r)}{\Longrightarrow} H_1$ and $G_2 \overset{spec(r)}{\Longrightarrow} H_2$ with $H_1 +_J H_2 = H$.

It remains to show $G \overset{spec(r)}{\Longrightarrow} H$. We are going to construct the pushouts (6) and (7), then the composition of pushouts directly leads to $G \overset{spec(r)}{\Longrightarrow} H$.

$$\begin{tikzcd}
I \arrow[d] \arrow[dr, phantom, "\mathbf{(9)}"] & D \arrow[l] \arrow[r] \arrow[d] \arrow[dr, phantom, "\mathbf{(10)}"] & J \arrow[d] \\
G_1 \arrow[d] \arrow[dr, phantom, "\mathbf{(6)}"] & C_1 \arrow[l] \arrow[r] \arrow[d] \arrow[dr, phantom, "\mathbf{(7)}"] & H_1 \arrow[d] \\
G & C \arrow[l] \arrow[r] & H
\end{tikzcd}$$

First we construct the morphisms $C \to G$ and $C \to H$ using pushout (4).

$$\begin{aligned}
D \to C_1 \to G_1 \to G &= D \to I \to G_1 \to G \quad \text{(due to pushout (9))}\\
&= D \to I \to G_2 \to G \quad \text{(due to pushout (3))}\\
&= D \to C_2 \to G_2 \to G \text{ (due to pushout (11))}
\end{aligned}$$

So we obtain $C \to G$. Moreover, (6) is a commutative square.

$$\begin{tikzcd}
 & & D \arrow[dl] \arrow[dr] & & \\
 & C_1 \arrow[dl] \arrow[dr] \arrow[rr, phantom, "\mathbf{(4)}"] & & C_2 \arrow[dl] \arrow[dr] & \\
G_1 \arrow[drr] \arrow[rr, phantom, "(=)"] & & C \arrow[d] \arrow[rr, phantom, "(=)"] & & G_2 \arrow[dll] \\
 & & G & &
\end{tikzcd}$$

Analogously we obtain $C \to H$ and the commutative square (7).

$$\begin{tikzcd}
 & & D \arrow[dl] \arrow[dr] & & \\
 & C_1 \arrow[dl] \arrow[dr] \arrow[rr, phantom, "\mathbf{(4)}"] & & C_2 \arrow[dl] \arrow[dr] & \\
H_1 \arrow[drr] \arrow[rr, phantom, "(=)"] & & C \arrow[d] \arrow[rr, phantom, "(=)"] & & H_2 \arrow[dll] \\
 & & H & &
\end{tikzcd}$$

Next we have to show the universal pushout property of the commutative square (6). Given a $\hat{G}$ so that

$$I \to G_1 \to \hat{G} = I \to G_2 \to \hat{G}$$

Then we can induce $G \to \hat{G}$ using the pushout (3), such that

$$\begin{aligned} G_1 \to \hat{G} &= G_1 \to G \to \hat{G} \text{ and} \\ G_2 \to \hat{G} &= G_2 \to G \to \hat{G}. \end{aligned}$$

$$\begin{tikzcd}
 & I \arrow[dl] \arrow[dr] & \\
G_1 \arrow[dr] \arrow[ddr, bend right] \arrow[rr, phantom, "(\mathbf{3})"] & & G_2 \arrow[dl] \arrow[ddl, bend left] \\
 & G \arrow[d] \arrow[l, phantom, "(=)"] \arrow[r, phantom, "(=)"] & \\
 & \hat{G} &
\end{tikzcd}$$

Moreover, we have

$$\begin{aligned} C_1 \to G_1 \to \hat{G} &= C_1 \to G_1 \to G \to \hat{G} \text{ (due to pushout (3))} \\ &= C_1 \to C \to G \to \hat{G} \text{ (due to pushout (9))} \end{aligned}$$

So we obtain the morphism $G \to \hat{G}$, so that

$$\begin{aligned} G_1 \to \hat{G} &= G_1 \to G \to \hat{G} \text{ and} \\ C \to \hat{G} &= C \to G \to \hat{G}. \end{aligned}$$

Finally, the uniqueness of $G \to \hat{G}$ follows immediately from the pushout properties of (3).

Analogously we can show the universal property of the commutative square (7).

Then we conclude the proof using the composition of pushouts leading to the desired transformation $G \overset{spec(r)}{\Longrightarrow} H$.

$$\begin{tikzcd}
I \arrow[d] & D \arrow[l] \arrow[r] \arrow[d] \arrow[dl, phantom, "(\mathbf{9}+\mathbf{6})"] \arrow[dr, phantom, "(\mathbf{10}+\mathbf{7})"] & J \arrow[d] \\
G & C \arrow[l] \arrow[r] & H
\end{tikzcd}$$