On the notion of deadlocks in open nets

Richard Müller

Institut für Informatik, Humboldt-Universität zu Berlin, 10099 Berlin, Germany
richard.mueller@informatik.hu-berlin.de

Abstract. Before a system is built in Service Oriented Computing from interacting services, it is modeled and verified with regard to different behavioral correctness criteria, among others, deadlock freedom. For this purpose, open nets as a special class of Petri nets are frequently used. In this paper, we present and compare three formalizations of when two services interact deadlock freely. We specifically highlight subtle details of existing formalizations and propose a new formalization that matches the intuition in every case.

1 Introduction

In the context of *Service Oriented Computing*, a system is built from interacting *services* ([4]). A service is a well defined, self-contained component that provides standard *business functionality* which is accessible via a standardized *interface*. It interacts with other services through *communication*, while being independent of their state or context ([1]). Best engineering practice suggests a system to be modeled before it is physically built. *Open nets* ([3]) are a special class of Petri nets, which have been proven notably helpful in modeling the *behaviour* of open systems, e.g., services.

There exist well-developed methods to analyze the behavior of a service modeled as open net with regard to different correctness criteria ([6]), among others, *deadlock freedom*. Intuitively, a *deadlock* describes an unwanted situation in which further progress in relevant (but not necessarily all) parts of a system is impossible. A system reaching a deadlock is typically considered malfunctioning. This makes the absence of a reachable deadlock, i.e., deadlock freedom, desirable behavior. We call two services *deadlock free partners* if they interact deadlock freely. There exist various formalizations of deadlocks and deadlock free partners in the model of open nets ([2], [6]). In this paper, we present and analyze two existing notions and highlight deficits of these notions arising in subtle situations. Based on these insights, we propose a new formalization of deadlock free partners which overcomes these deficits. We finally compare all three notions.

The rest of this paper is organized as follows. In Sect. 2, we briefly introduce basic concepts of Petri nets and our formal model for services, i.e., open nets. Then, in Sect. 3, we present three different formalizations of deadlocks and deadlock free partners. Section 4 is devoted to the comparison of the different notions. Finally, Sect. 5 concludes the paper and gives directions for future work.

2 Preliminaries

In this section, we recall *open nets* as a Petri net model for services. We start from marked net structures, i.e. place/transition nets, with the usual meaning of the constituents and the standard firing rule. We assume the standard notions of *markings*, *enabledness*, *firing*, and *reachability* in Petri nets. Otherwise, see [5] for an introduction.

Definition 1 (open net structure, partner structures, composition). *Let $N = (P,T,F)$ be a net structure and let I, O be two disjoint sets of places of N, such that $^{\bullet}I = O^{\bullet} = \emptyset$, and the environment $^{\bullet}t \cup t^{\bullet}$ of each transition t of N contains at most* one *node of $I \cup O$. Then N together with (I,O) is an* open net structure*. (I,O) is the* interface *of N, $IP(N) =_{def} I \cup O$ is the set of* interface places *of N, and nodes in $IP(N) \cup I^{\bullet} \cup {}^{\bullet}O$ are* interface *nodes of N, opposed to* inner *nodes of N.*

For $i = 1,2$ let $N_i = (P_i,T_i,F_i)$ together with the interface (I_i,O_i) be open net structures. Without loss of generality, we assume each node x shared by N_1 and N_2 be an interface place: Otherwise, replace x by different instances in both open net structures. N_1 and N_2 are partner structures *iff $I_1 = O_2$ and $I_2 = O_1$. The* composition *of N_1 with N_2 is the open net structure $N_1 \oplus N_2 =_{def} (P_1 \cup P_2, T_1 \cup T_2, F_1 \cup F_2)$ together with the interface $(I \setminus O, O \setminus I)$, where $I = I_1 \cup I_2$ and $O = O_1 \cup O_2$.*

Inner nodes model a service's business functionality and interface nodes model a service's standardized interface. The concept of partners expresses two services' capability of proper communication. On the level of open nets, communication of two partners N_1 and N_2 is the firing of an interface transition of N_1 or N_2 in their composition $N_1 \oplus N_2$. Additionally, we distinguish situations which N_1 and N_2 shall reach together, i.e., *target markings*.

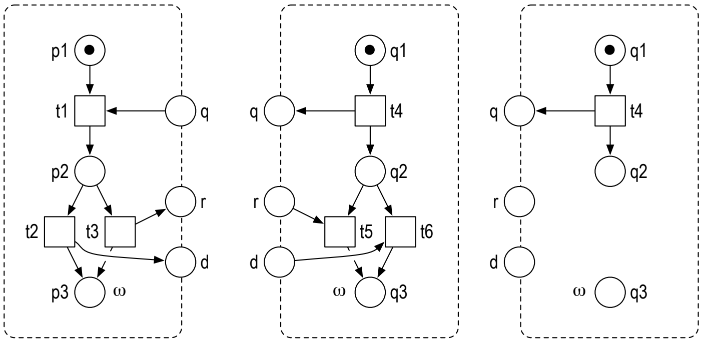
Definition 2 (open net, partners). *Let N be an open net structure, let m be a marking of N with no interface place marked, and let Ω be a set of markings of N with no interface place marked. Then N together with m and Ω is an* open net*, with m the* initial marking *of N, and Ω the set of* target markings *of N.*

Let N_1 and N_2 be two open nets with initial markings m_1, m_2 and target markings Ω_1, Ω_2, respectively. N_1 and N_2 are partners *iff N_1 and N_2 are partner structures. The initial marking m of $N_1 \oplus N_2$ is defined by $m =_{def} m_1 + m_2$, the target markings Ω of $N_1 \oplus N_2$ are $\Omega =_{def} \{m_1 + m_2 \mid m_1 \in \Omega_1, m_2 \in \Omega_2\}$.*

We depict the interface of an open net N by drawing its interface places on a dashed rectangle which contains all inner nodes and interface transitions of N. If N has exactly one target marking m, we indicate a place p of N with ω if and only if p is marked in m. See Fig. 1 for some examples. For an open net N, its *inner structure* $inner(N)$ is defined by removing the interface places of N, and restrict its initial marking and target markings accordingly.

3 Deadlocks

In the rest of this paper we assume freely chosen partners N and P. We shall compare several deadlock notions for open nets by means of N and P being *deadlock free partners*. Intuitively, deadlock free interaction is the avoidance of unwanted situations in which further progress in relevant parts of a system is impossible. In the model of open nets, we formulate this as follows: Whenever one relevant part, i.e., N or P, of the system $N \oplus P$ stops, than there is a wanted situation, i.e., a target marking, reachable in $N \oplus P$, or N and P will eventually communicate again. Massuthe [2] uses the notion of a *dead marking* to define a deadlock of two interacting services. This definition discriminates between target and non-target dead markings, the latter ones must not be reachable.



(a) N **(b)** P **(c)** Q

Fig. 1. The open nets N, P and Q. N and P are DF_{PM}-partners, N and Q are not.

Definition 3 (dead marking, deadlock$_{PM}$, DF_{PM}-partners). *A marking m of N is* dead *in N iff no transition of N is enabled in m. A dead non-target marking of N is a* deadlock$_{PM}$ *of N. P is* DF_{PM}-interacting *with N (synonymous to N and P are* DF_{PM}-partners*) iff no reachable marking of* $N \oplus P$ *is a deadlock*$_{PM}$ *of* $N \oplus P$*. We write* $DF_{PM}(N)$ *for the set of all open nets* DF_{PM}*-interacting with N, and* DF_{PM} *for the set of all* DF_{PM}*-partners.*

For example, the open net N (Fig. 1(a)) models a service which is able to receive a query (input place q), and then either replies to the query (output place r) or denies access (output place d). Afterwards, the service can terminate in the target marking $[p3]$. Typically, the decision between reply or denial is made depending on data inside the received query. As we abstract from data in open nets, we model the decision's outcome using non-determinism (interface transitions $t2$ and $t3$). The open net P (Fig. 1(b)) is DF_{PM}-interacting with N: Whether N sends a reply or denial, P is able to receive it. Thus, the only marking of $N \oplus P$ with no transition enabled is the target marking $[p3, q3]$ of $N \oplus P$, i.e., $[p3, q3]$ is no deadlock$_{PM}$. In contrast, the open net Q (Fig. 1(c)) is not DF_{PM}-interacting with N: There are two deadlocks$_{PM}$, $[p3, r, q2]$ and $[p3, d, q2]$, reachable in $N \oplus Q$.

Massuthe's definition provides a good first impression on how to formalize deadlocks in open nets. However, it is not precise enough. As an example, consider the open net R (Fig. 2(a)), which originates from Q by adding an *inner loop* (transition $t5$). R is as malfunctioning as Q: Two markings, $[p3, r, q2]$ and $[p3, d, q2]$, are reachable in $N \oplus R$, such that neither N nor R is in a target marking, and they will never communicate again. Intuitively, N and R should not be deadlock free partners. Nevertheless,

R is DF_{PM}-interacting with N, i.e., N and R are DF_{PM}-partners. This example shows a general drawback of Definition 3: *Every* open net N has at least one open net M which is DF_{PM}-interacting with N. M consists of an interface that is compatible with N's interface, and an inner loop without communication. Wolf [6] refines Massuthe's notion and introduces *responsiveness* of open nets to overcome endless loops without communication [1].

Definition 4 (responsiveness, DF_{KW}-partners). *An open net N is* responsive *iff, from each reachable marking in $inner(N)$, a marking is reachable in $inner(N)$ which is a target marking or which enables an interface transition of N. P is DF_{KW}*-interacting *with N iff P is responsive and no reachable marking of $N \oplus P$ is a deadlock$_{PM}$ of $N \oplus P$. N and P are DF_{KW}*-partners *iff N is DF_{KW}-interacting with P and P is DF_{KW}-interacting with N. We write $DF_{KW}(N)$ for the set of all open nets DF_{KW}-interacting with N, and DF_{KW} for the set of all DF_{KW}-partners.*

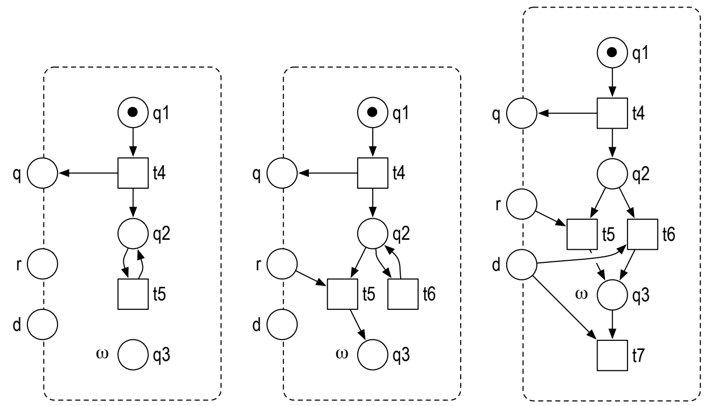


Fig. 2. Another three open nets R, S and T.

Responsiveness rules out some intuitively unwanted interacting open nets with endless loops without communication (like R in Fig. 2(a), which is not DF_{KW}-interacting with N), but unfortunately not all. Figure 2 depicts an example. The open net S (Fig. 2(b)) is responsive and DF_{KW}-interacting with N. However, we would intuitively classify the reachable marking $[p3, d, q2]$ of $N \oplus S$ as a deadlock: Neither N nor S is in a target marking, and they will never communicate again. Additionally, responsiveness is a rather strong restriction on partners. Intuitively, N and the open net T (Fig. 2(c)) are deadlock free partners: $N \oplus T$ has no reachable marking which enables transition $t7$. Nevertheless, T is not responsive and not DF_{KW}-interacting with N according to Def. 4.

[1] Additionally, Wolf restricts Def. 4 to bounded open nets and bounded communicating partners for decidability reasons [3]. As we just compare the notion of deadlocks and deadlock free partners (whether decidable or not), we do not employ this restriction.

In order to overcome those drawbacks, we propose a new, third definition of deadlocks and deadlock free partners. Our definition is based on the observation that a deadlock of two interacting services N and P is intuitively described from the view of one of the involved services, e.g., N, by three facts: (1) N is in an unwanted situation, (2) further progress in $N \oplus P$ by N alone is impossible, i.e., there is a need for interaction of P with N, and (3) P will never interact with N anymore. Therefore, we propose to describe deadlocks in open nets *locally*, i.e., from the view of one service, rather than from a global point of view on both partners. We do so by using the target markings of only one of the involved open nets, and by considering *silent* markings, which capture the absence of further communication between the partners.

Definition 5 (silent marking, deadlock$_{RM}$, DF_{RM}-partners). *A marking m of $N \oplus P$ is* silent *in $N \oplus P$ iff for all markings m' of $N \oplus P$ reachable from m holds: At most inner transitions of P are enabled in m'. A silent marking m of $N \oplus P$ such that $m_{|N}$ is no target marking of N, is a* deadlock$_{RM}$ *of $N \oplus P$. P is* DF_{RM}-interacting *with N iff no reachable marking of $N \oplus P$ is a deadlock$_{RM}$ of $N \oplus P$. N and P are* DF_{RM}-partners *iff N is DF_{RM}-interacting with P and P is DF_{RM}-interacting with N. We write $DF_{RM}(N)$ for the set of all open nets DF_{RM}-interacting with N, and DF_{RM} for the set of all DF_{RM}-partners.*

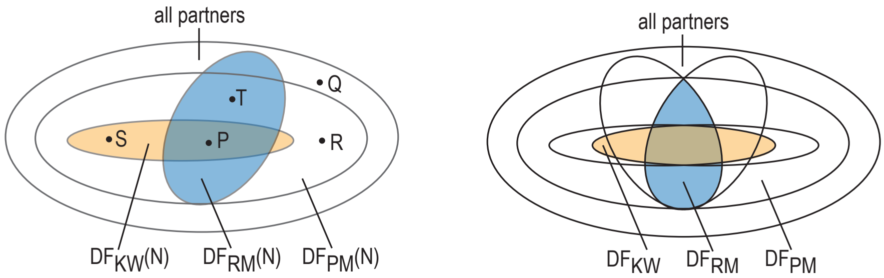
Notice the discrimination between wanted and unwanted situations by means of N's target markings instead of $N \oplus P$'s target markings like in Def. 3 and 4. Definition 5 matches our intuition in every previously mentioned example: N and P as well as N and T are DF_{RM}-partners, whereas all other partners are not.

4 Comparison

In this section, we shall compare Definitions 3, 4, and 5 in terms of the characterized deadlock free partners, and the complexity of the notions. Given an open net N, the set of open nets deadlock freely interacting with N generally differ, depending on which definition of deadlock free interaction is employed. Every open net DF_{KW}-interacting with N is DF_{PM}-interacting with N as well, the opposite is not true (see R and S in Fig. 2 for an example). An open net DF_{RM}-interacting with N can be DF_{PM}-interacting with N, DF_{KW}-interacting with N, or neither. Consider Fig. 3(a) for an illustration of above coherences as Venn diagram.

An open net DF_{RM}-interacting with N, which is not DF_{PM}-interacting with N, may seem counter-intuitive for a notion of deadlock free partners. However, Def. 5 says DF_{RM}-partners have to be mutually DF_{RM}-interacting with each other, therefore every two DF_{RM}-partners are DF_{PM}-partners as well. Again, this matches our intuition. Every two DF_{KW}-partners are DF_{PM}-partners. The sets of all DF_{KW}-partners and all DF_{RM}-partners lie diagonally: There exist DF_{KW}-partners, which are no DF_{RM}-partners (e.g. the open nets N and S in Fig. 1(a) and 2(b)), and vice versa (e.g. the open nets N and T in Fig. 1(a) and 2(c)). Figure 3(b) depicts above coherences.

All three presented formalizations differ in how complex it is to check whether two given open nets N and P are deadlock free partners or not. Checking according to Def. 4 is more complex than according to Def. 3, as responsiveness of N and P has to be checked additionally. Checking according to Def. 5 is more complex than according



(a) deadlock free interaction **(b)** deadlock free partners

Fig. 3. The sets of open nets deadlock freely interacting with N (Fig. 1(a)), and the sets of deadlock free partners.

to Def. 4, because it is more complex to check if a marking m of $N \oplus P$ is silent than to check if m is dead. To sum up, there is a trade-off between matching our intuitive deadlock notion and the complexity of a formalization.

5 Conclusion

We have presented two existing formalizations of when two services interact deadlock freely, and gave an idea of their deficits in subtle situations. Our approach for overcoming these deficits is a third formalization, describing deadlocks from a local point of view rather than globally. To the best of our knowledge, this is new. Finally, we compared all three formalizations, and highlighted their differences in complexity as well as matching our intuition. Though more complex, the new formalization matches our intuition in every case.

There exists strong theory for characterizing all deadlock free partners, based on the two existing formalizations ([2], [6]). It is part of further work to determine if those theory can be applied with the new formalization as well.

References

1. Alonso, G., Casati, F., Kuno, H.A., Machiraju, V.: Web Services - Concepts, Architectures and Applications. Data-Centric Systems and Applications, Springer (2004)
2. Massuthe, P.: Operating Guidelines for Services. Dissertation, Humboldt-Universität zu Berlin, Mathematisch-Naturwissenschaftliche Fakultät II; Eindhoven University of Technology (Apr 2009), iSBN 978-90-386-1702-2
3. Massuthe, P., Serebrenik, A., Sidorova, N., Wolf, K.: Can I find a partner? Undecidablity of partner existence for open nets. Inf. Process. Lett. 108(6), 374–378 (Nov 2008)
4. Papazoglou, M.P.: Service-oriented computing: Concepts, characteristics and directions. In: WISE. pp. 3–12. IEEE Computer Society (2003)
5. Reisig, W.: Petri Nets: An Introduction, Monographs in Theoretical Computer Science. An EATCS Series, vol. 4. Springer (1985)
6. Wolf, K.: Does my service have partners? T. Petri Nets and Other Models of Concurrency 2, 152–171 (2009)