

Grundlagen: Betriebssysteme und Systemsoftware

Tutorübung 07

Lukas Döllner

lukas.doellner@tum.de

<https://lukas-doellner.de/gbs>

WiSe 2023/24

04.12.2023 – 10.12.2023



Eine Fabrik stellt verschiedene Arten von Möbeln her. Diese Möbel werden in einem Lager mit begrenzter Kapazität gelagert. Die gelagerten Möbel werden von Mitarbeitern zur Laderampe geschleppt, von wo sie an Händler ausgefahren werden. Die Rampe hat ebenfalls eine begrenzte Kapazität. Das Ausfahren der Ware zum jeweiligen Geschäft übernimmt je ein Lastwagen. In den Geschäften werden die Möbel letztendlich an Kunden verkauft.

a)* Modelliere für das gegebene Szenario ein Petrinetz mit folgenden Randbedingungen:

- Die Fabrik produziert **Betten** und **Schränke**.
- Die Kapazität des Lagers beträgt **500 Möbelstücke**.
- **Ein Mitarbeiter** übernimmt den Transport von Möbelstücken aus dem Lager zur Rampe.
- Auf der Rampe können **10 Möbelstücke** lagern.
- **Zwei Lastwagen** fahren Möbelstücke aus. Die Kapazität eines LKWs beträgt **1 Möbelstück**.
- **Zwei Geschäfte** nehmen die Ware ab.
 - ▶ Das erste Geschäft verkauft nur **Betten**. Es können maximal **30 Betten** dort gelagert werden.
 - ▶ Das zweite Geschäft verkauft sowohl Betten als auch Schränke. Es kann je **20 Betten** und **20 Schränke** aufnehmen.

Anregungen zur Konstruktion des Petrinetzes:

- Überlege Dir eine geeignete Realisierungsmöglichkeit für das zugrunde liegende Producer-Consumer Problem (Fabrik produziert, Kunden konsumieren) in einem Petrinetz.
- Verwende die in der Vorlesung eingeführten Kontrollstrukturen, um sicherzustellen, dass z.B. nicht mehr Betten bzw. Schränke als tatsächlich vorhanden ausgeliefert werden.
- Achte auf die Transportmodellierung. Erlaube nicht das Aufsammeln von neuen Waren, wenn die alten noch nicht ausgeliefert wurden (ein Transporter kann nicht gleichzeitig am Start und am Ziel sein).
- Überprüfe Deine Modellierung. Das zu konstruierende Petrinetz ist komplex, wenn es alle Anforderungen abdeckt (Zum Vergleich: Die Mustermodellierung hat 15 Stellen). Mache Dich mit den Modellierungsmöglichkeiten vertraut, die Dir ein Petrinetz bietet.

Aufgabe 2

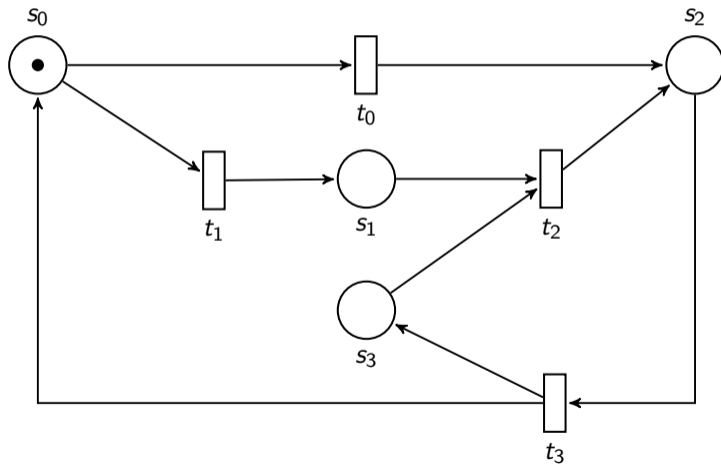
Betten und Schränke

b) Diskussion: Überlege Dir, ob das modellierte Petrinetz Transitionen besitzt, die verhungern könnten.

Aufgabe 3

Petrinetze

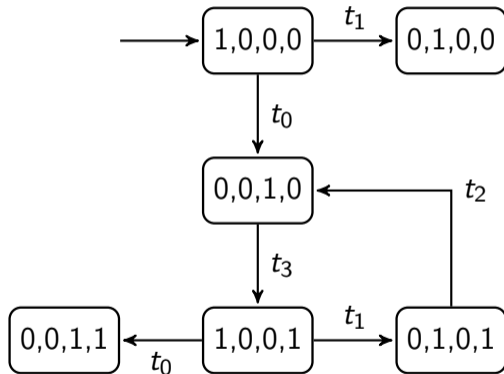
Gegeben sei folgendes **boolesches** Petrinetz:



Aufgabe 3

Petrinetze

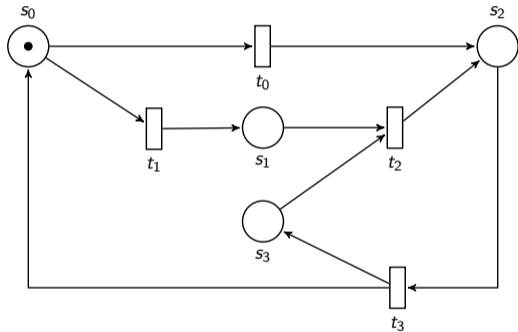
a)* Ist eine Verklemmung erreichbar?



Aufgabe 3

Petrinetze

- b)* Ändert sich der Erreichbarkeitsgraph, wenn natürlichzahlige Belegungen zugelassen werden?
Hinweis: Du musst nicht den gesamten Graphen zeichnen.

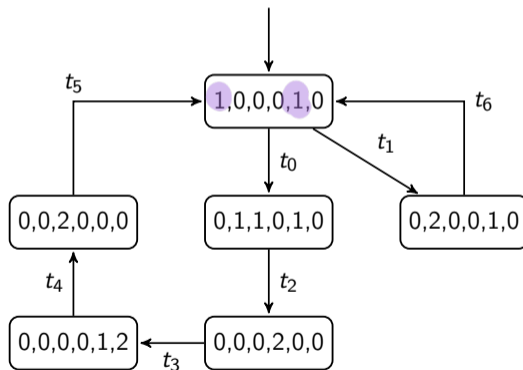


c)* Ist bei dem booleschen Netz ein Verhungern einer Transition möglich? Ändert sich das bei natürlichzahligen Belegungen?

Aufgabe 4

Erreichbarkeitsgraph2Petriernet

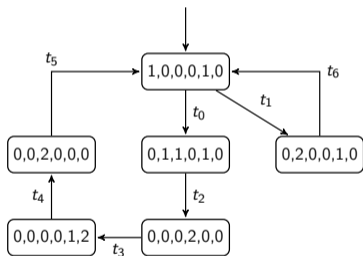
Gegeben sei der folgende Erreichbarkeitsgraph, wobei jede Node eine Belegung $(s_0, s_1, s_2, s_3, s_4, s_5)$ repräsentiert:



Aufgabe 4

Erreichbarkeitsgraph2Petriernet

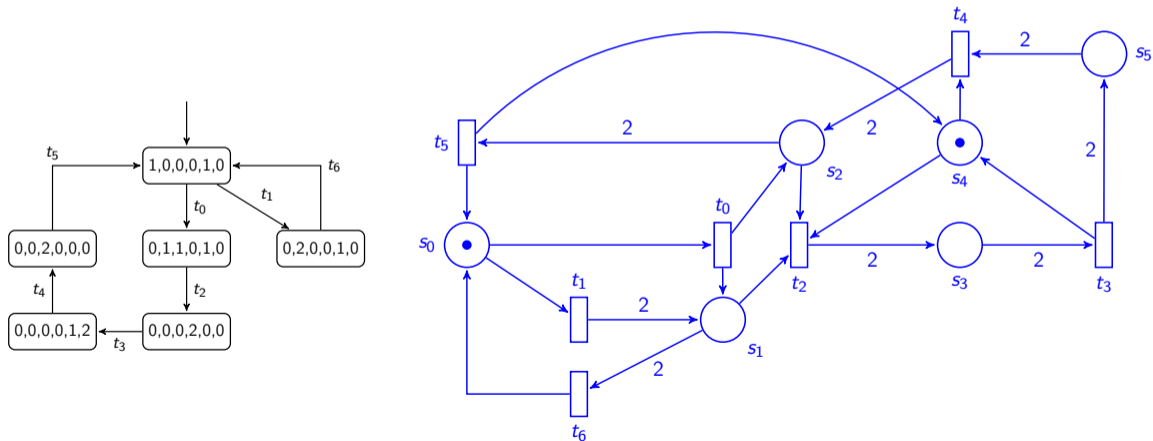
a)* Modelliere das zu dem Erreichbarkeitsgraphen zugehörige Petrinetz mit der Startmarkierung $(1,0,0,0,1,0)$.



Aufgabe 4

Erreichbarkeitsgraph2Petriernet

a)* Modelliere das zu dem Erreichbarkeitsgraphen zugehörige Petrinetz mit der Startmarkierung $(1,0,0,0,1,0)$.



- **Boolesch:** \forall erreichbare Zustände z und \forall Stellen s gilt: s hat in z stets ≤ 1 Tokens.
(Diese Eigenschaft kann in einem Netz vorgegeben sein; Dann hat jede Stelle eine implizite Kapazitätsbegrenzung von 1.)
- Eine Transition t ist im Zustand z **schaltbereit** \iff Vor- und Nachbedingungen erfüllt.
 - ▶ **Vorbedingung:** Alle Stellen, aus denen Token genommen werden sollen, haben genügend.
 - ▶ **Nachbedingung:** Alle Stellen, in die Token gelegt werden sollen, haben genügend Platz.
- **Verklemmung** (eines Zustandes z): \nexists Transition t , die in z schaltbereit ist.
- **Verklemmungsfreiheit:** \nexists erreichbarer Zustand z , sodass z verklemmt ist.
- **Lebendigkeit:** \forall erreichbare Zustände z , \forall Transitionen t gilt: \exists einen neuen, von z erreichbaren Zustand z' , in dem t schaltbereit ist.
- **Verhungern** (einer Transition t): \exists unendliche Schaltsequenz, sodass t unendlich oft schaltbereit ist, aber nur endlich oft schaltet.
- **Fairness:** \forall Transitionen t gilt: t verhungert nicht.

Aufgabe 4

Erreichbarkeitsgraph2Petri net

b)* Ist das zu einem Erreichbarkeitsgraphen zugehörige Petrinetz eindeutig?

c)* Argumentiere anhand des Erreichbarkeitsgraphen, welche der Dir bekannten Eigenschaften das dazugehörige Petrinetz besitzt.