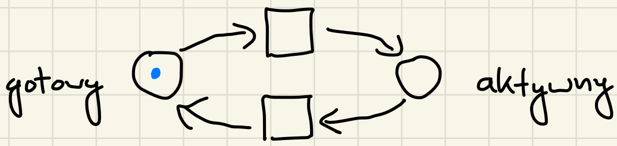
Ćwiczenia 1 10.10

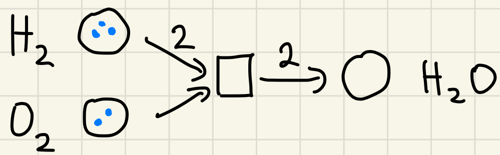
Sieci Petriego – krótkie wprowadzenie

* przykład: stan procesu



* przykład: reakcje chemiczne

$$2H_2 + O_2 \to 2H_2O$$



Elementarna sieć Petriego z konfiguracją:

* zbiór **miejsc** P i **tranzycji** T, $P \cap T = \emptyset$
* zbiór **łuków** $F \subseteq (P \times T) \cup (T \times P)$
* konfiguracja $M \subseteq P$

Ogólna sieć:

* F to łuki z przypisanymi wagami $(\in \mathbb{N})$
* $M : P \to \mathbb{N}$ jako uogólnienie konfiguracji

Kiedy można odpalić tranzycję ?

* w elementarnych sieciach, gdy mamy po żetonie na miejscach wejściowych (ozn. $^{\bullet}t$), a miejsca wyjściowe ($t^{\bullet}$) są puste
* w ogólnych sieciach, gdy na miejscach wejściowych liczby żetonów są co najmniej takie jak wagi łuków

od teraz mamy na myśli sieci ogólne

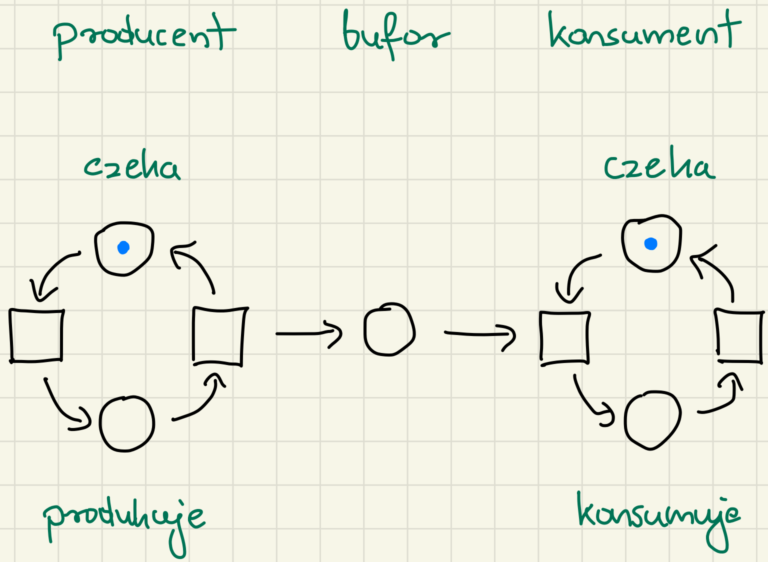
Ważne własności (mówimy o sieci z konfiguracją!)

* osiągalność – czy z konfiguracji początkowej można na skutek odpalenia pewnej sekwencji tranzycji przejść do innej ?
* żywotność – każda tranzycja jest żywa; tranzycja t jest żywa, gdy z każdej osiągalnej konfiguracji można odpalić pewną sekwencję tranzycji, po których możliwe jest odpalenie tranzycji t

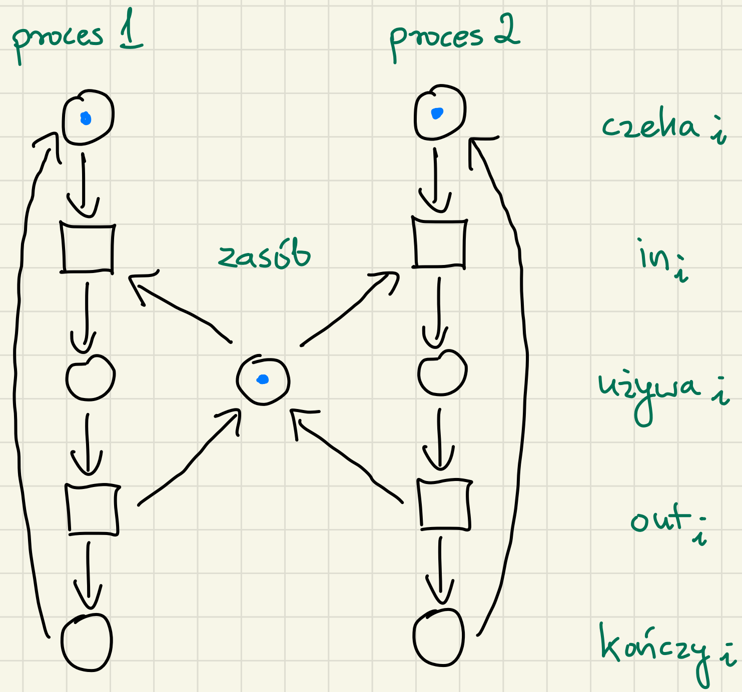
* **k-ograniczoność** – każde miejsce jest k-ograniczone; miejsce jest k-ograniczone, jeśli w każdej osiągalnej konfiguracji zawiera nie więcej niż k żetonów
* **ograniczoność** – sieć z konfiguracją jest ograniczona, jeśli istnieje $K \in \mathbb{N}$, takie że mamy K-ograniczoność
* **brak blokady** – nie można osiągnąć konfiguracji w której wszystkie tranzycje są martwe (nie można ich odpalić)

Możemy też mówić o strukturalnych własnościach sieci, które nie zależą od wybranej konfiguracji. Przykładowo, sieć jest **strukturalnie ograniczona**, jeśli dla każdej możliwej konfiguracji początkowej M sieć z konfiguracją M jest ograniczona.

1. Zamodeluj za pomocą sieci elementarnych klasyczne problemy współbieżności : producenta / konsumenta i wzajemne wykluczanie



* producent i konsument nie komunikują się bezpośrednio
* jedyna komunikacja zachodzi przez bufor



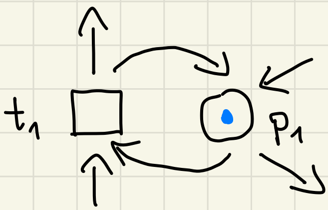
* procesy 1 i 2 nigdy nie komunikują się ze sobą bezpośrednio, tzn. nie wykonują wspólnych akcji (tranzycje)

* Łatwo uogólnić na więcej procesów

* nie pilnujemy tutaj, aby każdy proces, który chce skorzystać z zasobu, mógł tego w końcu dokonać (problem zagłodzenia)

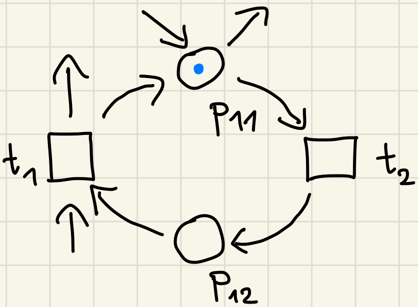
2. Eliminacja ciasnych pętli, eliminacja wag > 1

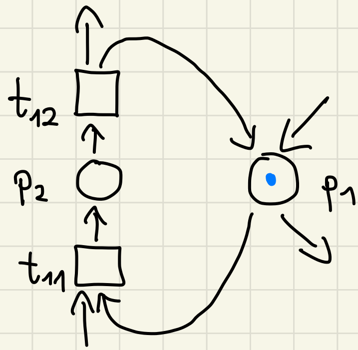
ciasna pętla = miejsce i tranzycja są połączone w obie strony



I sposób

II sposób





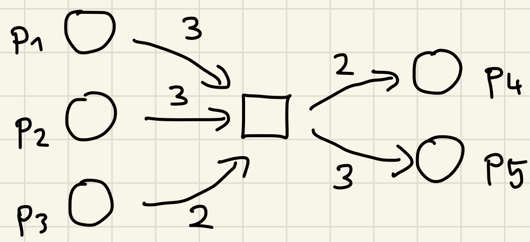
t_2 musi przewidzieć, kiedy żeton z p_{11} będzie potrzebny t_1... (potencjalna blokada)

przy tej modyfikacji nie ma problemu z potencjalną „blokadą" miejsca p_1

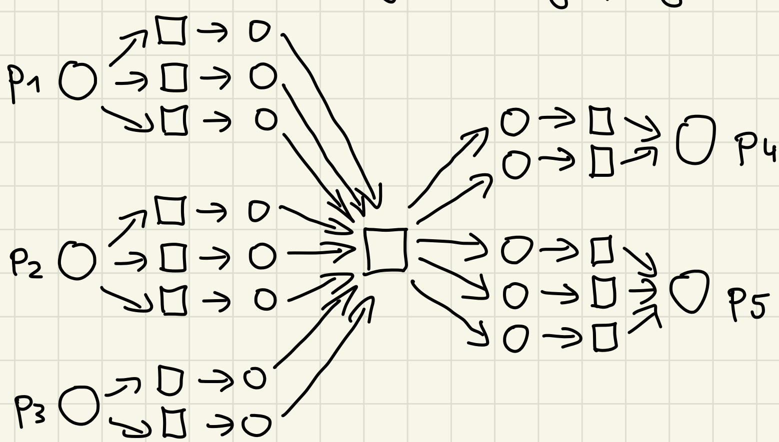
* żeby „poprawić" I sposób można dodać tranzycję przerzucającą żeton z p_{12} na p_{11}

* założenie ${}^\bullet t \cap t^\bullet = \phi$ upraszcza analizę własności sieci

Eliminacja wag > 1:

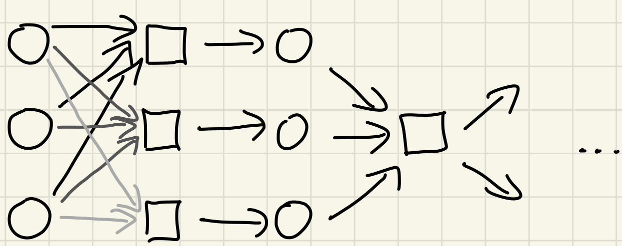


pierwszy pomysł

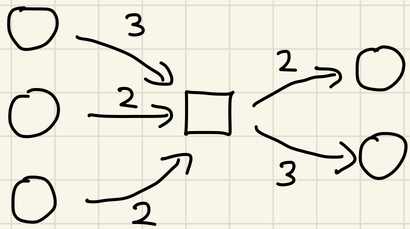


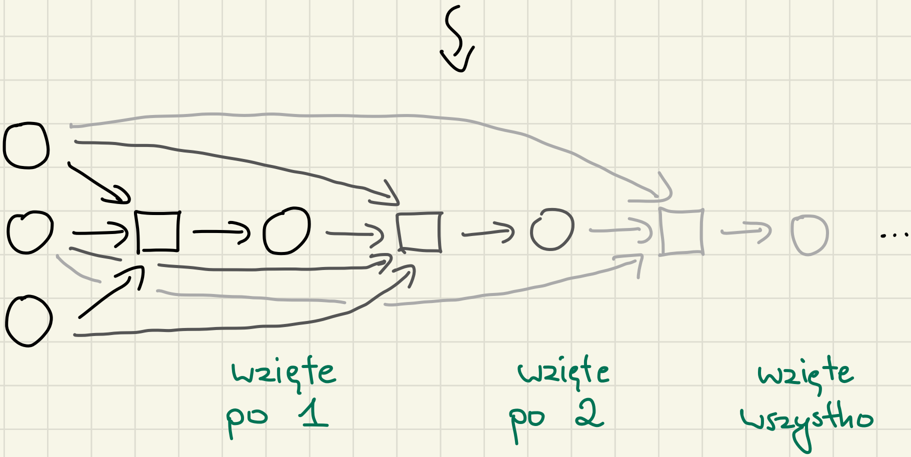
Trochę za dużo nowych tranzycji ...

Drugi pomysł:



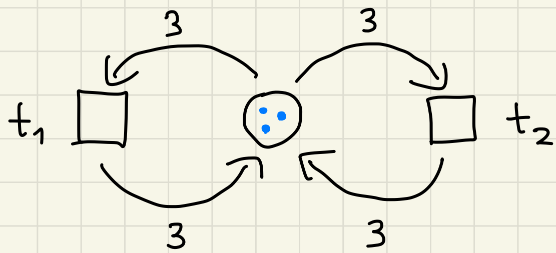
Eliminacja wag > 1 (jeszcze inny sposób)





* analogicznie dla wyjściowych łuków
* można też dodać tranzycje cofające poszczególne kroki pobierania po jednym żetonie
* WAZNE: zachowuje się relacja osiągalności (na podstawowych polach), ale niekoniecznie inne własności (np. brak blokady)

Spójrzmy na sieć:



Ma ona własność braku blokady, ale

1° po podstawowym przekształceniu możemy nową sieć zablokować (t_1 zabierze 1, t_2 - 2)

2° sieć z możliwością powrotu nie ma tego problemu, ALE gdy weźmiemy konfigurację początkową z dwoma żetonami, to dana sieć jest zablokowana, a nowa - nie

Wniosek: jeśli interesuje nas osiągalność, robienie zaproponowanych przekształceń niczego nie zmienia; z pozostałymi własnościami trzeba uważać

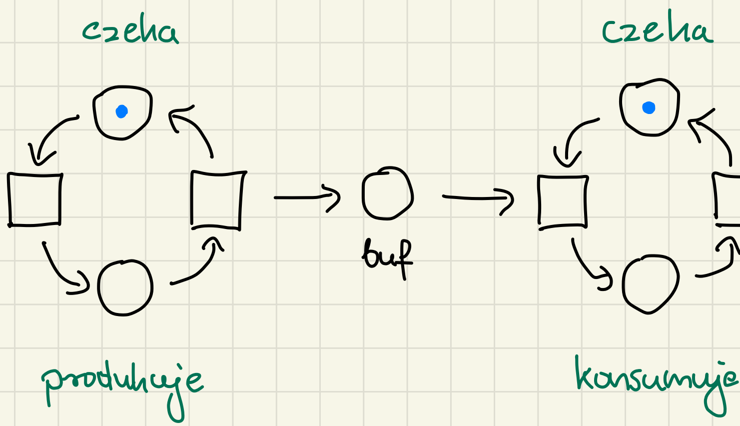
3. Równoważność sieci elementarnych oraz 1-ograniczonych sieci ogólnych

$\Leftarrow$

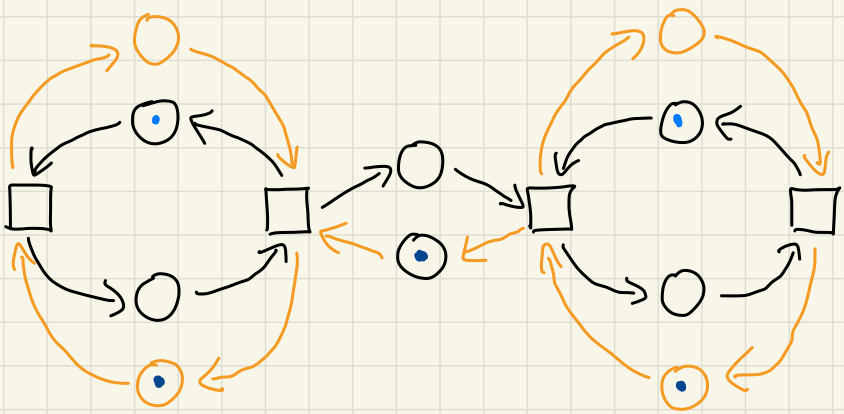
* możemy założyć, że nie ma wag > 1, wpp tranzycje z takimi łukami można usunąć
* otrzymana sieć jest siecią elementarną, bo początkowa konfiguracja kładzie na pola pojedyncze żetony, a w dowolnej osiągalnej konfiguracji M każda żywa tranzycja t spełnia $t^{\bullet} \cap M = \emptyset$

$\Rightarrow$

* aby zapewnić 1-ograniczoność, musimy rozbić każde miejsce p na dwa nowe p_e i p_f oznaczające odpowiednio: pole p jest puste (empty) i pole p jest pełne (full), tj. ma położony jeden żeton

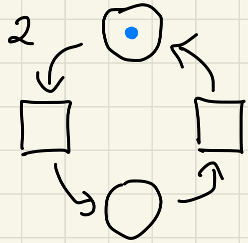


Bez zmiany sieć producent/konsument nie jest ograniczona (przez pole buf, które może zebrać dowolną liczbę tokenów).

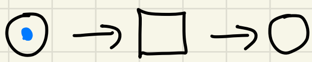


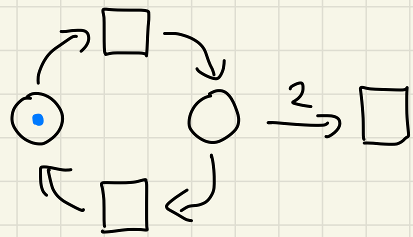
* niezmiennik: p_e i p_f mają łącznie 1 żeton
* każdemu biegowi w podstawowej sieci w oczywisty sposób odpowiada bieg nowej sieci

4. Sieć żywa, ale nie ograniczona

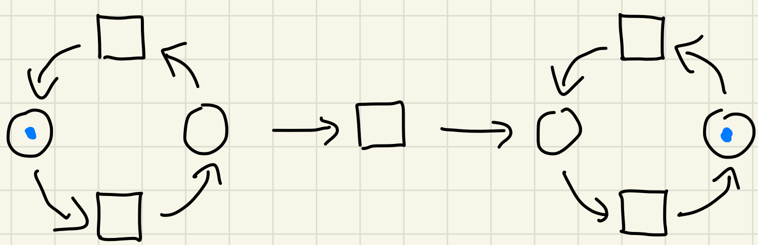


5. Sieć ograniczona, ale nie żywa





A jeśli chcemy jeszcze mieć brak blokady?



6. Czy żywotność jest monotoniczna w ogólnych sieciach Petriego?

zadanie do pomyślenia w domu (nieobowiązkowe)