Laboratorium 12

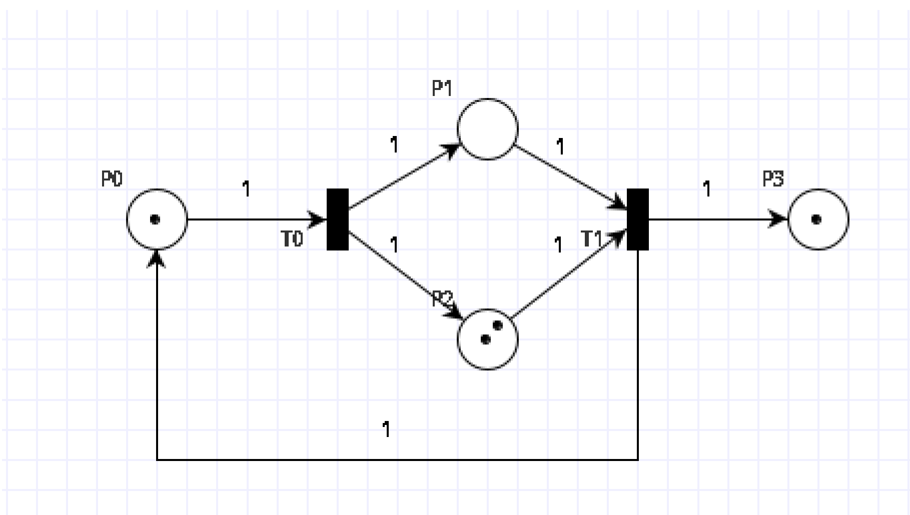
Łukasz Wala
AGH, Wydział Informatyki, Elektroniki i Telekomunikacji
Teoria Współbieżności 2022/23

Kraków, 10 stycznia 2023

1 Zadania

1.1 Zadanie 1

Wymyślić własną maszynę stanów, zasymulować przykład i dokonać analizy grafu osiągalności oraz niezmienników.



Rysunek 1: Maszyna stanów

Petri net state space analysis results

Bounded	false
Safe	false
Deadlock	false

Rysunek 2: Wynik dla *State Space Analysis*

Sieć nie jest ograniczona, ponieważ w P3 będzie przybywać tokenów w nieskończoność, z tej racji nie jest również bezpieczna (1-ograniczona). Nie ma deadlock-a, ponieważ zawsze jest możliwość wykonania każdego z przejść.

Rysunek 3: Wykres *Reachablility/Coverability Graph*

Każde ze znakowań jest osiągalne, każde z przejść jest żywe, czyli sieć również jest żywa.

Petri net invariant analysis results

T-Invariants

T0	T1

The net is not covered by positive T-Invariants, therefore we do not know if it is bounded and live.

P-Invariants

P0	P1	P2	P3
1	1	0	0
1	0	1	0

The net is not covered by positive P-Invariants, therefore we do not know if it is bounded.

P-Invariant equations

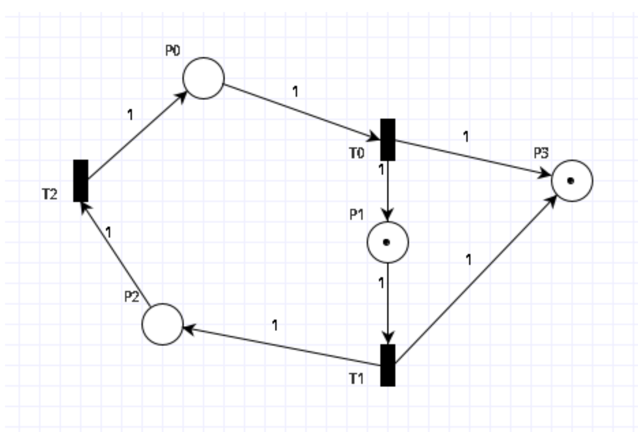
M(P0) + M(P1) = 1
M(P0) + M(P2) = 3

Rysunek 4: Wynik *Invariant analysis*

T-invariants jest puste, co sugeruje, że sieć jest nieodwracalna.

1.2 Zadanie 2

Zasymulować sieć jak poniżej. Dokonać analizy przejść. Jaki wniosek można wyciągnąć o odwracalności sieci? Wygenerować graf osiągalności. Proszę wywnioskować z grafu, czy sieć jest żywa. Proszę wywnioskować czy jest ograniczona. Objaśnić wniosek.



Petri net invariant analysis results

T-Invariants

T0	T1	T2

The net is not covered by positive T-Invariants, therefore we do not know if it is bounded and live.

P-Invariants

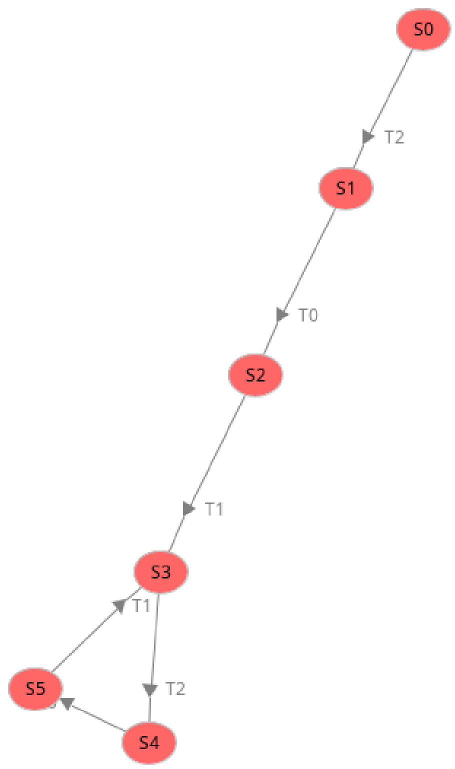
P0	P1	P2	P3
1	1	1	0

The net is not covered by positive P-Invariants, therefore we do not know if it is bounded.

P-Invariant equations

M(P0) + M(P1) + M(P2) = 1

Rysunek 5: Wynik *Invariant analysis*

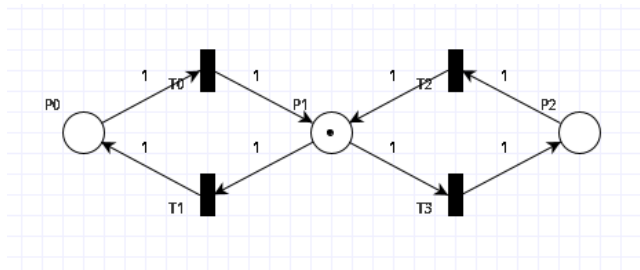


Rysunek 6: Wynik *Reachability/Coverability Graph*

Brak odwracalności, co można stwierdzić na podstawie braku wektora *T-invariants*. Sieć jest żywa - każde przejście musi się wykonać, następnie wykonują się w pętli. Nie jest ograniczona, liczba tokenów w P3 rośnie w nieskończoność.

1.3 Zadanie 3

Zasymulować wzajemne wykluczanie się dwóc procesów na wspólnym zasobie. Dokonać analizy niezmienników. Wyjaśnić znaczenie równań (*P-invariant equations*). Czy sieś jest zachowawcza? Które równanie mówi nam o rozmiarze bufora?



Rysunek 7: Sieć przedstawiająca wykluczające się procesy

Znakowanie P1 oznacza stan wolny zasobu, znakowania P0 oraz P1 ”konkurują” o zasób.

Petri net invariant analysis results

T-Invariants

T0	T1	T2	T3
1	1	0	0
0	0	1	1

The net is covered by positive T-Invariants, therefore it might be bounded and live.

P-Invariants

P0	P1	P2
1	1	1

The net is covered by positive P-Invariants, therefore it is bounded.

P-Invariant equations

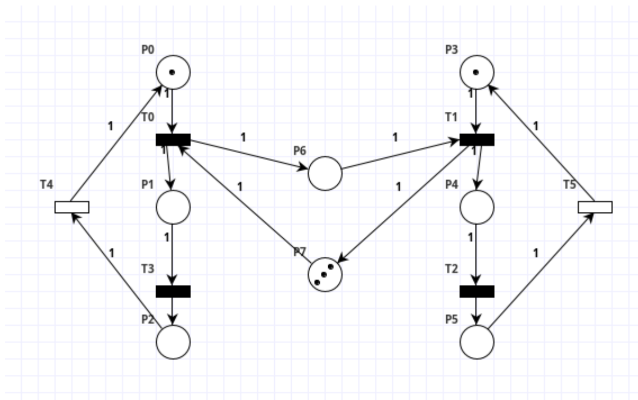
M(P0) + M(P1) + M(P2) = 1

Rysunek 8: Wynik *Invariant analysis*

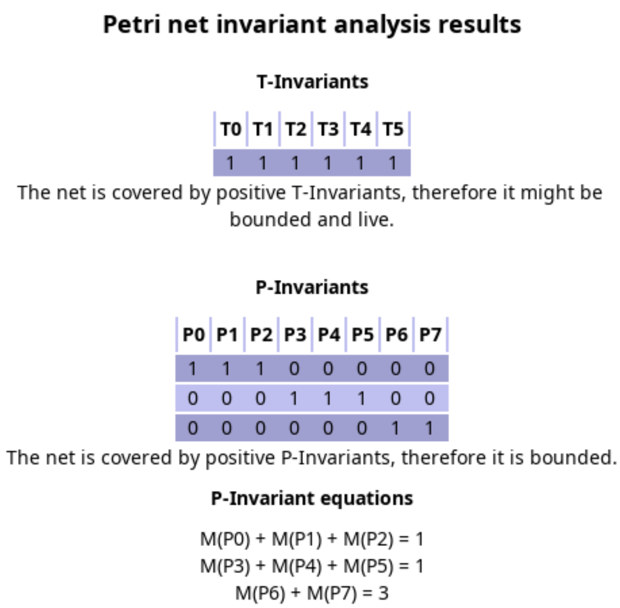
W równaniu występują wszystkie trzy znakowania. Znakowanie P1 oznacza że zasób jest wolny, P0 oraz P2 oznaczają, że zasób jest zajęty przez jeden z procesów. Liczba jeden po prawej stronie równania oznacza, że suma tokenów we wszystkich stanach zawsze wynosi jeden, czyli token zawszy musi być tylko w jednym ze znakowań, dzięki temu zasób jest chroniony przed jednoczesnym dostępem przez dwa procesy.

1.4 Zadanie 4

Uruchomić problem producenta i konsumenra z ograniczonym buforem. Dokonać analizy niezmienników. Czy sieć jest zachowawcza? Które równanie nam o rozmiarze bufora?



Rysunek 9: Sieć dla problemu producenta i konsumenta



Petri net invariant analysis results

T-Invariants

T0	T1	T2	T3	T4	T5
1	1	1	1	1	1

The net is covered by positive T-Invariants, therefore it might be bounded and live.

P-Invariants

P0	P1	P2	P3	P4	P5	P6	P7
1	1	1	0	0	0	0	0
0	0	0	1	1	1	0	0
0	0	0	0	0	0	1	1

The net is covered by positive P-Invariants, therefore it is bounded.

P-Invariant equations

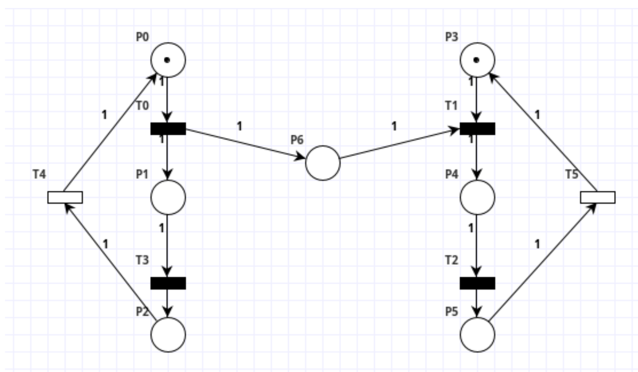
M(P0) + M(P1) + M(P2) = 1
M(P3) + M(P4) + M(P5) = 1
M(P6) + M(P7) = 3

Rysunek 10: Wynik *Invariant analysis*

Sieć jest zachowawcza, ponieważ każde przejście produkuje tyle samo tokenów ile pobiera. Na podstawie *T-Invariants* można wnioskować, że sieć jest odwracalna. Sieć jest żywa, ponieważ wszystkie przejścia mogą być wykonane. O wielkości bufora mówi równanie trzecie, znakowanie P6 odpowiada za miejsca zajęte, a P7 za miejsca wolne.

1.5 Zadanie 5

Stworzyć symulację problemu producenta i konsumenta z nieograniczonym buforem. Dokonać analizy niezmienników. Zaobserwować brak pełnego pokrycia miejsc.



Rysunek 11: Sieć dla problemu producenta i konsumenta z nieograniczonym buforem

Petri net invariant analysis results

T-Invariants

T0	T1	T2	T3	T4	T5
1	1	1	1	1	1

The net is covered by positive T-Invariants, therefore it might be bounded and live.

P-Invariants

P0	P1	P2	P3	P4	P5	P6
1	1	1	0	0	0	0
0	0	0	1	1	1	0

The net is not covered by positive P-Invariants, therefore we do not know if it is bounded.

P-Invariant equations

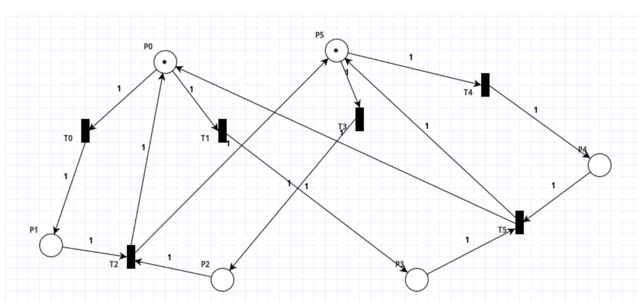
M(P0) + M(P1) + M(P2) = 1
M(P3) + M(P4) + M(P5) = 1

Rysunek 12: Wynik *Invariant analysis*

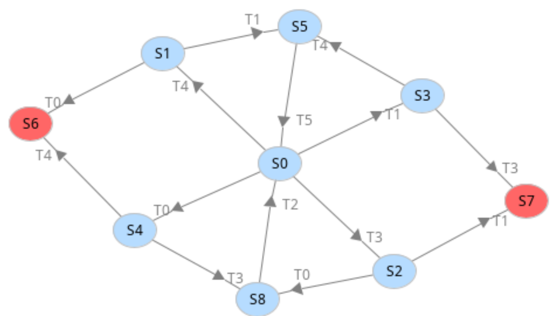
Na podstawie wektora *T-invariants* widać, że sieć jest odwracalna. Jest też żywa, ponieważ każde znakowanie jest osiągalne. Brak równania zawierającego P6 w sekcji *P=invariants* oznacza, że to znakowanie (symbolizujące bufor) jest nieograniczone.

1.6 Zadanie 6

Zasymulować prosty przykład ilustrujący zakleszczenie. Wygenerować graf osiągalności i zaobserwować znakowania, z których nie można wykonać przejść. Zaobserwować właściwości sieci w *State Space Analysis*. Poniżej przykład sieci z możliwością zakleszczenia.

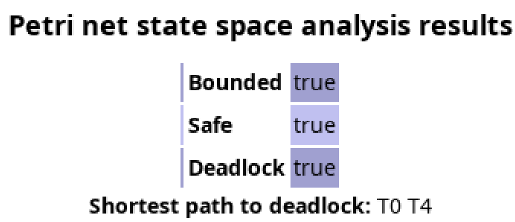


Rysunek 13: Przykład sieci z zakleszczeniem



Rysunek 14: Wynik *Reachability/Coverability Graph*

Graf łatwo pozwala stwierdzić, że stanami, w których nastąpi deadlock, są S6 oraz S7.



Rysunek 15: Wynik *State Space Analysis*

Powyżej wydać, że najkrótszą ścieżką do zakleszczenia są przejścia T0 oraz T4.

2 Wnioski

Powyższe przykłady pokazują możliwości sieci Petriego do formalnego modelowania problemów współbieżnych, np. problemu producentów i konsumentów. Dzięki takiemu formalnemu modelowi można zbadać pewne właściwości problemu, jego przebieg, co może być przydatne przy np. unikaniu zakleszczeń czy innych rodzajów niepowodzeń.

3 Bibliografia

1. N.J. Dingle, W.J. Knottenbelt and T. Suto. PIPE2: A Tool for the Performance Evaluation of Generalised Stochastic Petri Nets