Teoria współbieżności

Laboratorium 12

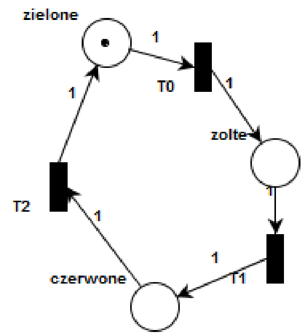
Sieci Petriego

Albert Gierlach

15.01.2021

1. Ćwiczenie

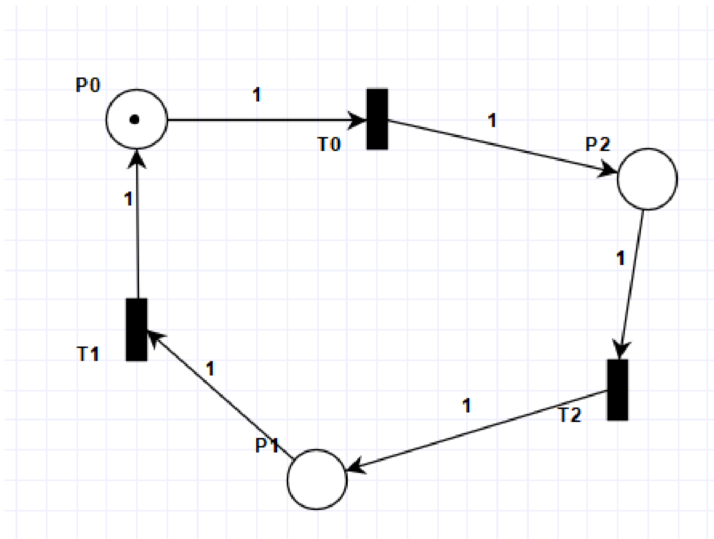
Maszyna stanów. Prosty model maszyny stanów swiateł ulicznych przedstawia sieć na rysunku poniżej:



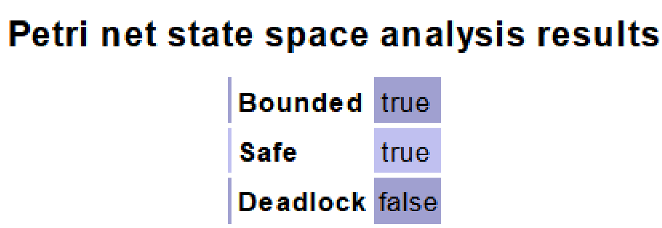
Stanami sa miejsca sieci, zaś znacznik pokazuje w jakim stanie aktualnie sie znajdujemy.

- Narysować przyklad w symulatorze.
- Sprawdzić własciwości sieci (ograniczoność, bezpieczenstwo i możliwy deadlock) w symulatorze Pipe w menu "State Space Analysis".
- Wygenerować graf osiągalności "Reachability/Coverability Graph". Zaobserwować:
  - Jakie znakowania są osiagalne ?
  - Ile wynosi maksymalna liczba znaczników w każdym ze znakowań ? Jakie mozemy wyciągnac z tego wnioski n.t. ograniczoności i bezpieczenstwa?
  - Czy kazde przejście jest przedstawione jako krawedz w grafie ? Jaki z tego wniosek n.t. zywotnosci przejsc ?
  - Czy wychodzac od dowolnego wezla grafu (znakowania) mozna wykonac dowolne przejscie ? Jaki z tego wniosek n.t. zywotnosci sieci? Czy sa mozliwe zakleszczenia ?
- Wykonać analizę niezmiennikow (wybrac w menu "Invariant Analysis").
  - wynik analizy niezmiennikow przejsc (T-invariants) pokazuje nam, ile razy trzeba odpalic dane przejscie (T), aby przekształcic znakowanie poczatkowege z powrotem do niego samego (wynik nie mowi nic o kolejnosci odpalen). Z wyniku mozemy m.in. wnioskowac o odwracalnosci sieci.
  - wynik analizy niezmiennikow miejsc (P-invariants) pokazuje nam zbiory miejsc, w ktorych laczna suma znacznikow sie nie zmienia. Pozwala to wnioskowac n.t. zachowawczosci sieci (czyli wlasnosci, gdzie suma znacznikow pozostaje stala) oraz o ograniczonosci miejsc.

2. Rozwiązanie



Rysunek 1: Zbudowany graf w programie PIPE2

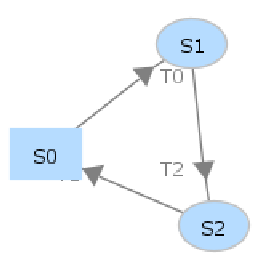


Petri net state space analysis results

Bounded	true
Safe	true
Deadlock	false

Rysunek 2: Wyniki "State Space Analysis"

Jak widzimy sieć jest ograniczona (ponieważ liczba tokenów wewnątrz sieci zawsze jest równa jeden), bezpieczna (jest 1-ograniczona) i nie wystąpi w niej deadlock (nie ma sytuacji, w której nie moglibyśmy przejść dalej).



Rysunek 3: Wyniki "Reachability/Coverability Graph"

Widzimy, że każde znakowanie jest osiągalne, a maksymalna liczba znaczników w kazdym z nich wynosi jeden. Stąd wniosek, że sieć jest bezpieczna i ograniczona. Każde z przejść jest pokazane jako krawędź w grafie, czyli zawsze można wystartować z dowolnego stanu - oznacza to, że każde przejście jest żywe. Wychodzac od dowolnego znakowania można wykonać dowolne przejście - wynika z tego, że sieć jest żywa oraz nie są możliwe zakleszczenia.

Petri net invariant analysis results

T-Invariants

T0	T1	T2
1	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

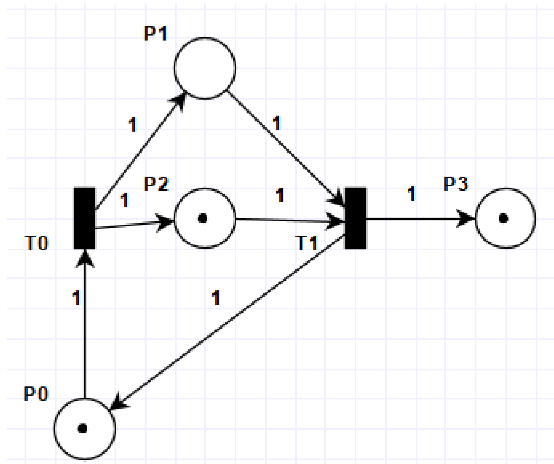
Rysunek 4: Wyniki "Invariant Analysis"

Wynik analizy niezmienników przejść mówi, że sieć jest odwracalna, ponieważ, aby wrócić do stanu startowego, należy przejść przez każdą tranzycję. Z kolei wynik analizy niezmienników miejsc określa miejsca, w których suma znaczników pozostaje stała - sieć jest ograniczona i zachowawcza.

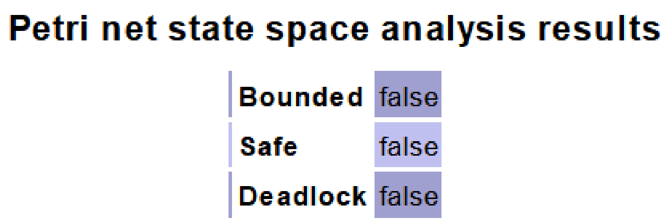
3. Zadanie 1

Wymyslic wlasna maszyne stanow, zasymulowac przyklad i dokonac analizy grafu osiagalnosci oraz niezmiennikow j.w.

4. Rozwiązanie



Rysunek 5: Stworzona sieć



Petri net state space analysis results

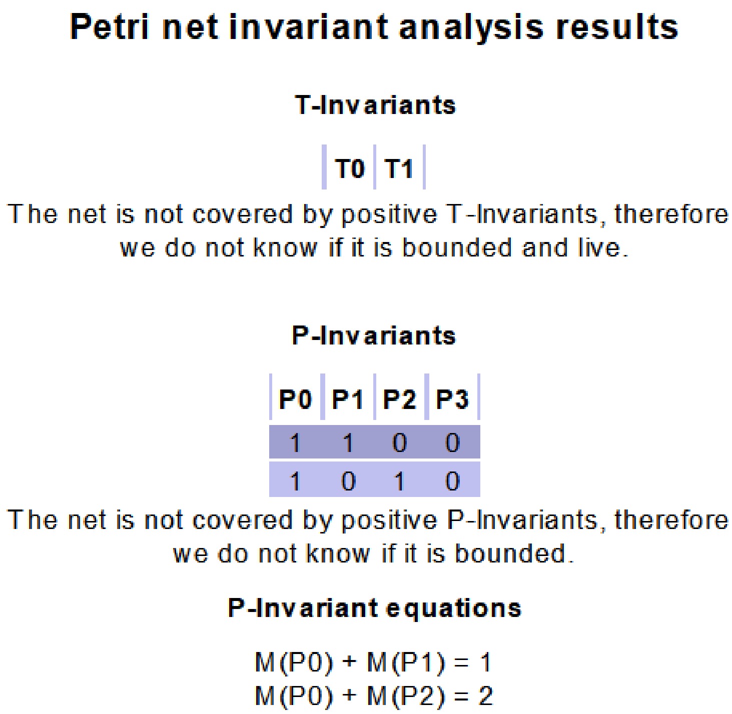
Bounded	false
Safe	false
Deadlock	false

Rysunek 6: Wyniki "State Space Analysis"

Sieć nie jest ograniczona, ponieważ tranzycja T1 będzie powodować produkowanie dodatkowych tokenów, czyli w miejscu P3, będzie ciągle przybywać tokenów (w nieskończoność). Sieć nie jest bezpieczna ponieważ nie jest 1-ograniczona (w P3 pojawia się więcej niż jeden token). Nie ma możliwości deadlocka, ponieważ zawsze mamy możliwość przejść do innego stanu (poza przechodzeniem z P3).

Rysunek 7: Wyniki "Reachability/Coverability Graph"

Każde ze znakowań jest osiągalne, każde przejście jest żywe, a więc sieć też 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

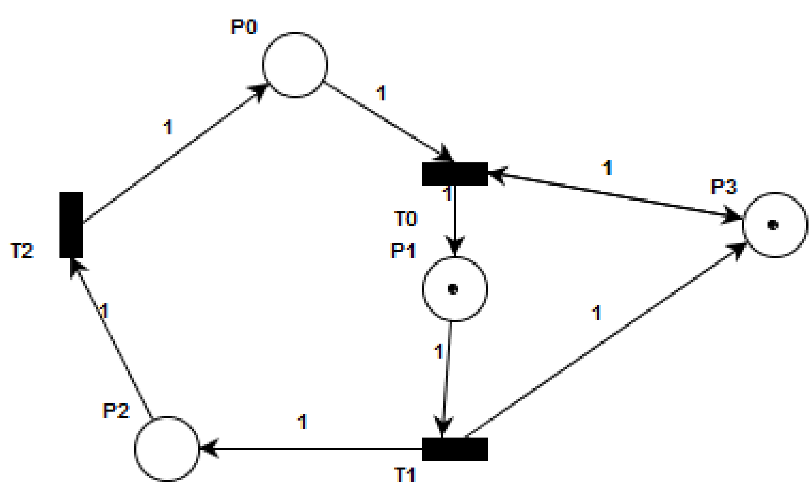
M(P0) + M(P1) = 1
M(P0) + M(P2) = 2

Rysunek 8: Wyniki "Invariant Analysis"

Widzimy, że T-invariants jest puste - stąd wynika, że sieć jest nieodwracalna. Z kolei z równań P-invariants możemy wyciągnąć wniosek, że sieć nie jest bezpieczna, zachowawcza, ograniczona.

5. Zadanie 2

Zasymulowac sieć jak ponizej:



Dokonac analizy niezmiennikow przejsc. Jaki wniosek mozna wyciagnac o odwracalnosci sieci? Wygenerowac graf osiagalnosci. Prosze wywnioskowac z grafu, czy siec jest zywa. Prosze wywnioskowac czy jest ograniczona. Objasnic wniosek.

6. Rozwiązanie

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

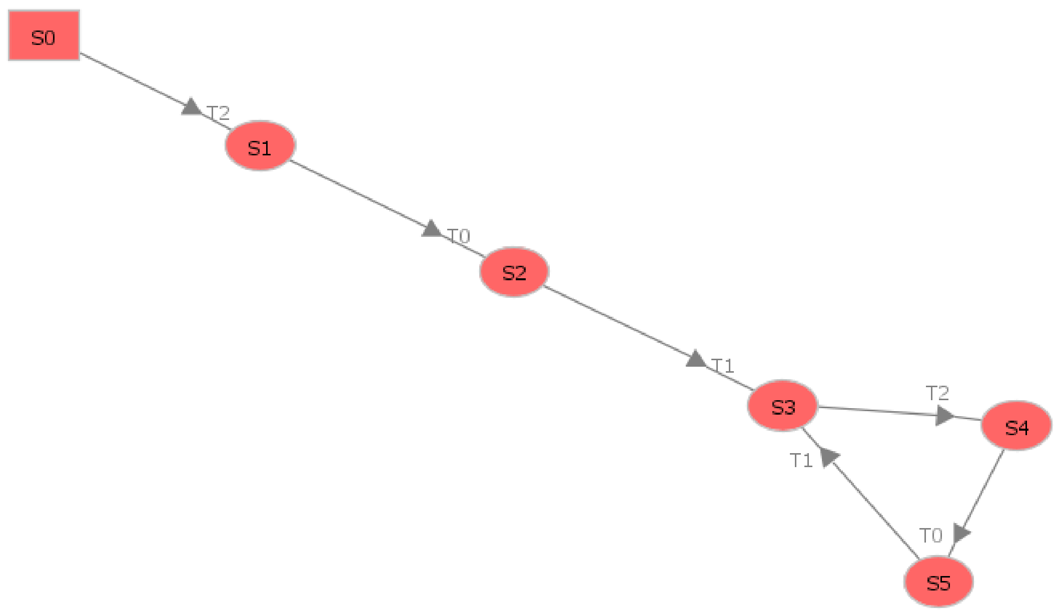
P0	P1	P2	P3
1	1	1	0

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 9: Wyniki "Invariant Analysis"



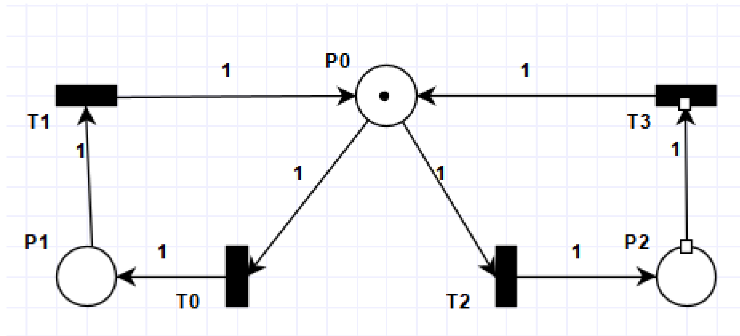
Rysunek 10: Wyniki "Reachability/Coverability Graph"

- Odwracalność - nie, można to stwierdzić na podstawie wektora (a właściwie jego braku) T-invariants
- Żywotność - tak, każde z przejść musi się wykonać, a później wykonują się w pętli
- Ograniczoność - nie, ilość tokenów w P3 rośnie w nieskończoność.

7. Zadanie 3

Zasymulowac wzajemne wykluczanie dwoch procesow na wspolnym zasobie. Dokonac analizy niezmiennikow. Wyjasnij znaczenie rownan (P-invariant equations). Ktore rownanie pokazuje dzialanie ochrony sekcji krytycznej?

8. Rozwiązanie



Rysunek 11: Przygotowana sieć przedstawiająca procesy, które wzajemnie się wykluczają

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

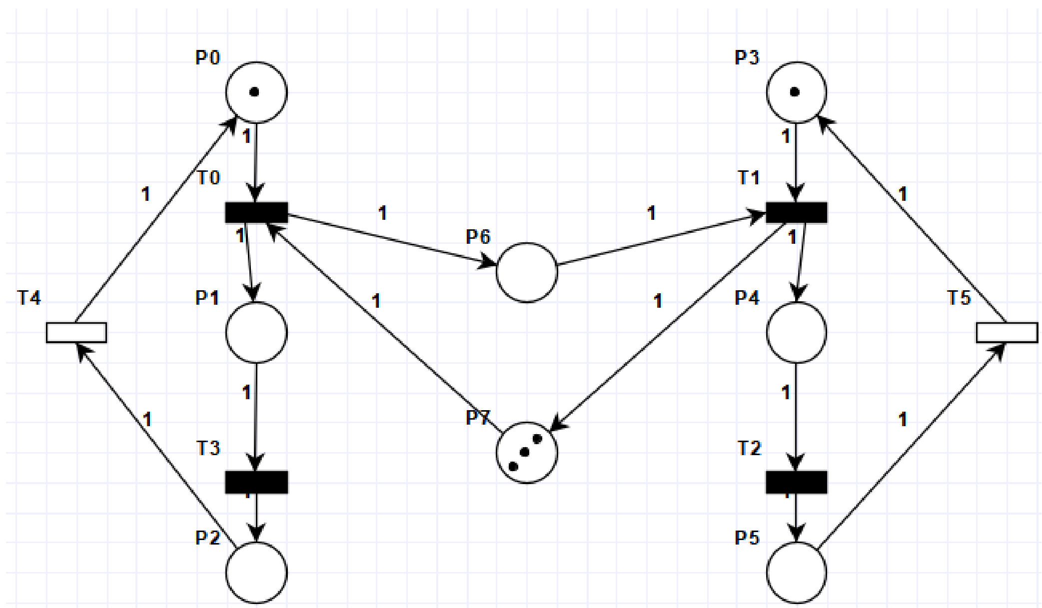
Rysunek 12: Wyniki "Invariant Analysis"

Gdy przyjrzymy się równaniom P-invariant, możemy zauważyć, że występują w nim wszystkie trzy stany. Stan P0 odpowiada za to, że zasób jest wolny, a stany P1 i P2 oznaczają, że zasób jest zajęty przez jeden z procesów. Po prawej stronie równania znajduje się liczba jeden, co oznacza, że suma tokenów we wszystkich stanach zawsze wynosi jeden, czyli token zawsze znajduje się w jednym ze stanów, a na tym polega ochrona sekcji krytycznej.

9. Zadanie 4

Uruchomic problem producenta i konsumenta z ograniczonem buforem (mozna posluzyc sie przykladem, menu:file, examples). Dokonac analizy niezmiennikow. Czy siec jest zachowawcza? Ktore rownanie mowi nam o rozmiarze bufora?

10. Rozwiązanie



Rysunek 13: Problem producenta i konsumenta z ograniczonym 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	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

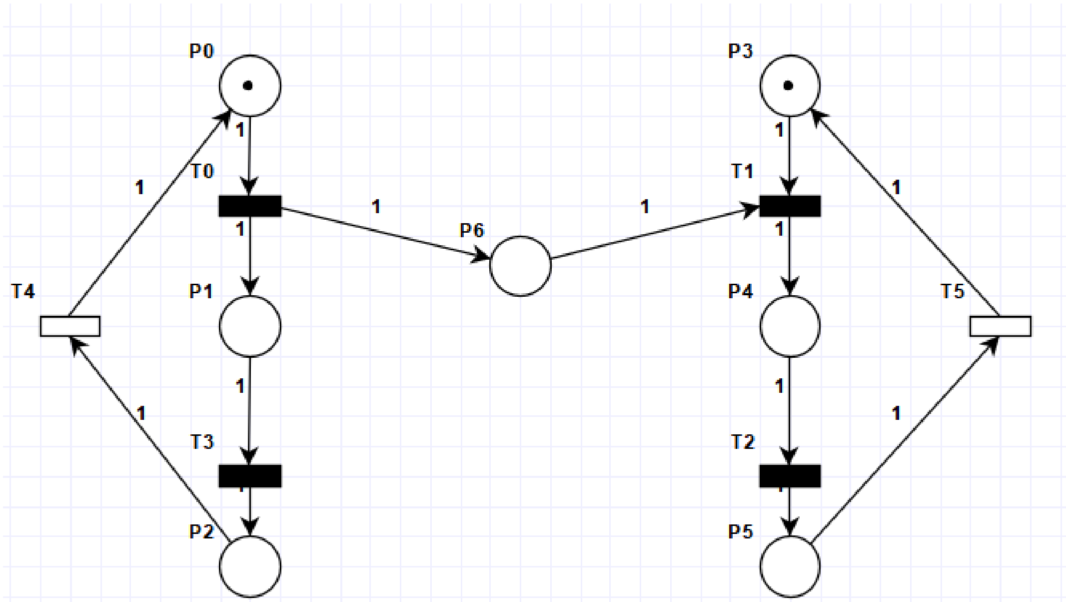
Rysunek 14: Wyniki "Invariant Analysis"

Sieć jest zachowawcza, ponieważ każda tranzycja produkuje tyle samo tokenów ile pobiera. Na podstawie wektora T-invariants widzimy, że sieć jest odwracalna. Sieć jest żywa gdyż wszystkie przejścia mogą być wykonane. O wielkości bufora mówi nam równanie nr. 3. Stan P6 odpowiada za miejsca zajęte, a P7 za miejsca wolne.

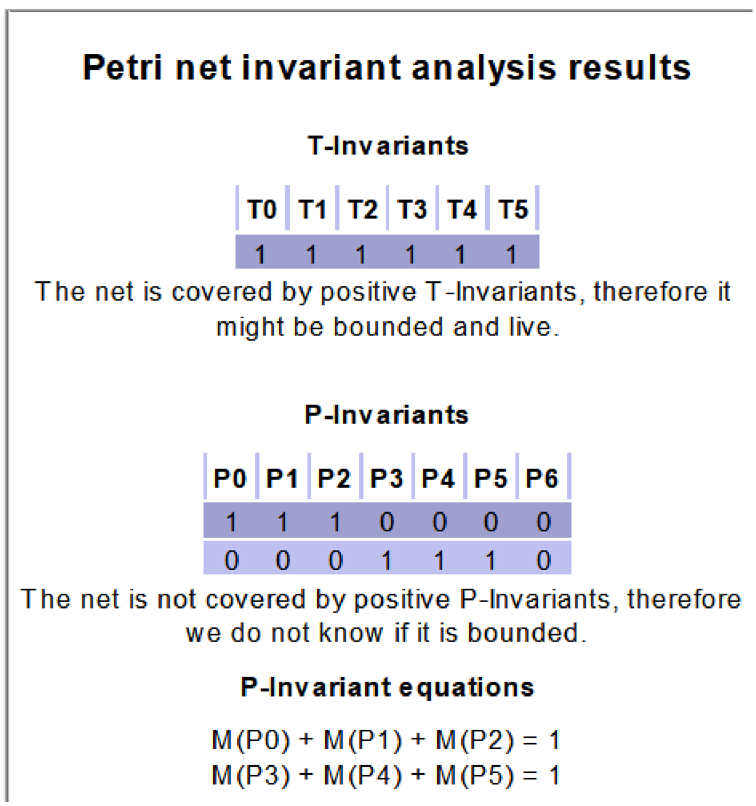
11. Zadanie 5

Stworzyc symulacje problemu producenta i konsumenta z nieograniczonym buforem. Dokonac analizy niezmiennikow. Zaobserwowac brak pelnego pokrycia miejsc.

12. Rozwiązanie



Rysunek 15: Problem 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

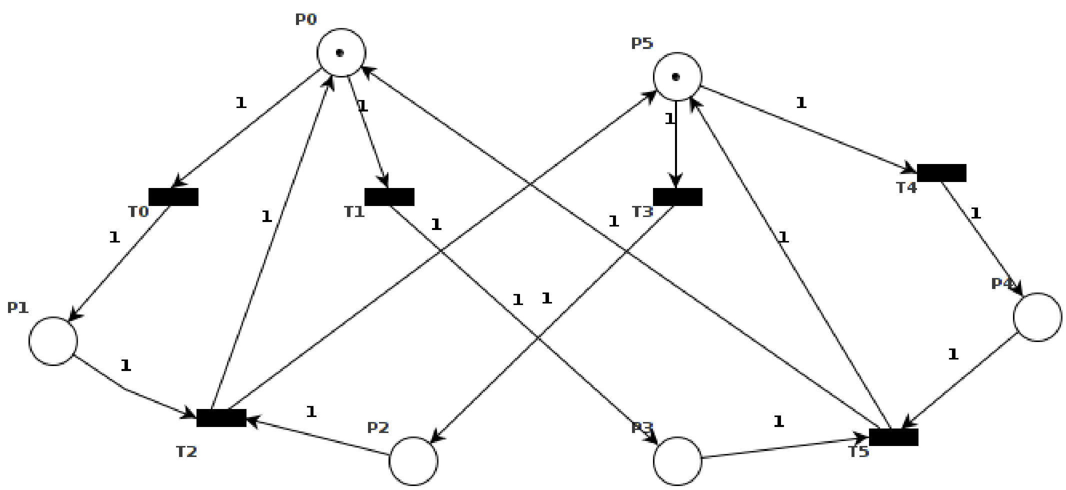
Rysunek 16: Wyniki "Invariant Analysis"

Z wektora T-invartiants - sieć odwracalna. Jest też żywa, ponieważ każdy stan jest osiągalny. W sekcji P-invariant equations obserwujemy, brak wystąpienia stanu P6, czyli bufora. Jest tak dlatego, iż jest on nieograniczony i nie da się go określić równaniem. Pozostałe równania opisują procesy wytwarzania i konsumowania tokenów.

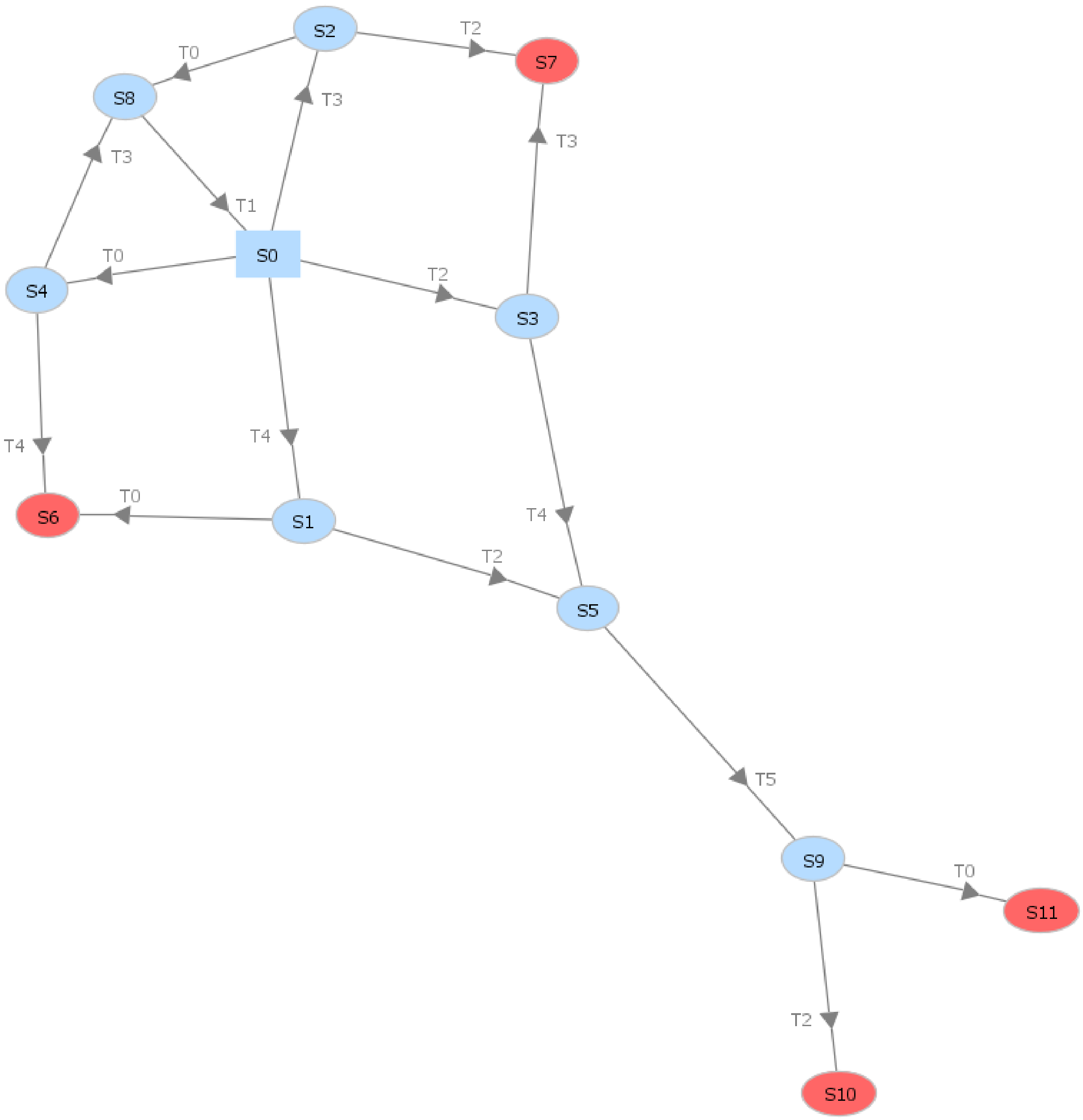
13. Zadanie 6

Zasymulowac prosty przyklad ilustrujacy zakleszczenie. Wygenerowac graf osiagalnosci i zaobserwowac znakowania, z ktoroch nie mozna wykonac przejsc. Zaobserwowac wlasciwosci sieci w "State Space Analysis". Ponizej przyklad sieci z mozliwoscia zakleszczenia (mozna wymyslic inny):

14. Rozwiązanie

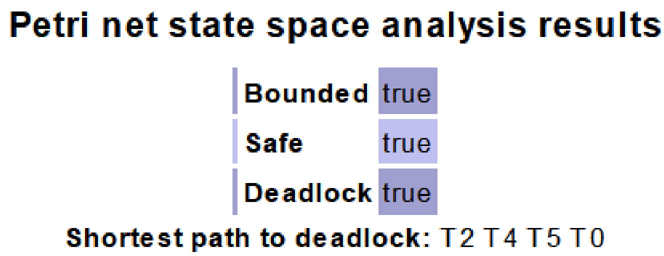


Rysunek 17: Zamodelowana sieć



Rysunek 18: Wyniki "Reachability/Coverability Graph"

Graf pozwala jednoznacznie stwierdzić, że stanami, w których nastąpi deadlock są stany: S6, S7, S10, S11 - wchodząc do nich nie można już przejść dalej. Wynik analizy "State Space" pozwala poznać najkrótszą ścieżkę prowadzącą do zakleszczenia.



Rysunek 19: Wyniki "State Space Analysis"

15. Wnioski

Powyższe przykłady ukazują, że sieci Petriego pozwalają modelować różne systemy. Procesy zazwyczaj przebiegają według jakiegoś określonego modelu, dzieki temu możemy zbadać właściwości tego procesu, a dzięki temu poznać zależności między poszczególnymi podsystemami. Sieci Petriego pozwalają także unaocznić przebieg i zakończenie danego systemu, co może być użyteczne w wyszukiwaniu problemów (np. zakleszczeń).

16. Bibliografia

- http://jedrzej.ulasiewicz.staff.iiar.pwr.wroc.pl/ProgramowanieWspolbiezne/wyklad/Sieci-Petriego15.pdf?fbclid=IwAR3euBljzKlFJS0nbfJpHYjRgv8tzs_rAG7fSj84x2too3zE7nf5Ja0b2yA
- http://sirius.cs.put.poznan.pl/~inf89721/MiAPB/Nowe/5%20-%20Analiza%20sieci%20Petriego.pdf
- https://pl.wikipedia.org/wiki/Sie%C4%87_Petriego#:~:text=Sie%C4%87%20Petriego%20%E2%80%93%20j%C4%99zyk%20modelowania%20dyskretnych,zosta%C5%82y%20zdefiniowane%20w%20latach%2060.
- http://robert.wojcik.staff.iiar.pwr.wroc.pl/dydaktyka/dzienne/ina/miasi_l1.pdf