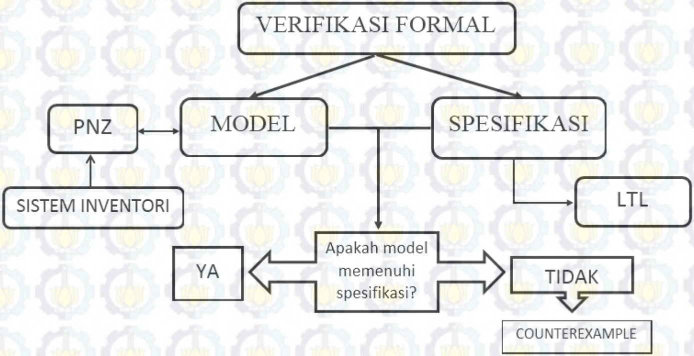
Verifikasi Formal Petri Net dengan *Counter* pada Sistem Inventori

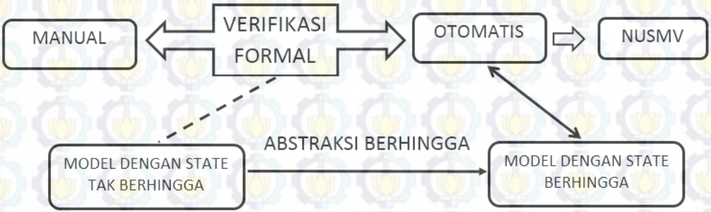
Oleh:
Ruvita Iffahtur Pertiwi 1214 201 202
Dosen Pembimbing :
Dr. Dieky Adzkiya, S.Si., M.Si.
Dr. Subiono, M.S.

20 Mei 2016

Latar Belakang



Latar Belakang



Rumusan Masalah

Berdasarkan latar belakang yang telah diberikan, maka permasalahan dalam penelitian ini adalah sebagai berikut.

1. Bagaimana abstraksi berhingga Petri net dengan *counter* pada sistem inventori?

Rumusan Masalah

Berdasarkan latar belakang yang telah diberikan, maka permasalahan dalam penelitian ini adalah sebagai berikut.

1. Bagaimana abstraksi berhingga Petri net dengan *counter* pada sistem inventori?
2. Apakah Petri net dengan *counter* pada sistem inventori memenuhi spesifikasi LTL?

Batasan Masalah

Pada penelitian ini, diberikan batasan masalah sebagai berikut.

1. Bilangan bulat yang digunakan adalah bilangan bulat tak negatif

Batasan Masalah

Pada penelitian ini, diberikan batasan masalah sebagai berikut.

1. Bilangan bulat yang digunakan adalah bilangan bulat tak negatif
2. Jumlah marking yang *reachable* adalah 1.

Tujuan Penelitian

Berdasarkan rumusan masalah yang diberikan, maka tujuan dari penelitian ini adalah sebagai berikut.

1. Mengetahui abstraksi berhingga sistem transisi Petri net dengan *counter* pada sistem inventori yang memiliki jumlah state tak berhingga menjadi berhingga.

Tujuan Penelitian

Berdasarkan rumusan masalah yang diberikan, maka tujuan dari penelitian ini adalah sebagai berikut.

1. Mengetahui abstraksi berhingga sistem transisi Petri net dengan *counter* pada sistem inventori yang memiliki jumlah state tak berhingga menjadi berhingga.
2. Memverifikasi Formal Petri net dengan *counter* pada sistem sistem inventori menggunakan spesifikasi LTL.

Manfaat Penelitian

1. Diketahui salah satu kelas Petri net baru yaitu Petri net dengan *counter* (PNZ).

Manfaat Penelitian

1. Diketahui salah satu kelas Petri net baru yaitu Petri net dengan *counter* (PNZ).
2. Diperoleh verifikasi Petri net dengan *counter* pada suatu sistem inventori menggunakan spesifikasi LTL.

Manfaat Penelitian

1. Diketahui salah satu kelas Petri net baru yaitu Petri net dengan *counter* (PNZ).
2. Diperoleh verifikasi Petri net dengan *counter* pada suatu sistem inventori menggunakan spesifikasi LTL.
3. Sebagai salah satu refrensi yang memiliki sistem dengan data yang bernilai bilangan bulat dapat dimodelkan dalam Petri net dengan *counter* untuk kemudian dianalisis atau diverifikasi sistemnya.

Manfaat Penelitian

1. Diketahui salah satu kelas Petri net baru yaitu Petri net dengan *counter* (PNZ).
2. Diperoleh verifikasi Petri net dengan *counter* pada suatu sistem inventori menggunakan spesifikasi LTL.
3. Sebagai salah satu refrensi yang memiliki sistem dengan data yang bernilai bilangan bulat dapat dimodelkan dalam Petri net dengan *counter* untuk kemudian dianalisis atau diverifikasi sistemnya.
4. Sebagai salah satu bahan refrensi untuk penelitian selanjutnya khususnya berkaitan pada verifikasi formal dan Petri net dengan *counter*.

Manfaat Penelitian

1. Diketahui salah satu kelas Petri net baru yaitu Petri net dengan *counter* (PNZ).
2. Diperoleh verifikasi Petri net dengan *counter* pada suatu sistem inventori menggunakan spesifikasi LTL.
3. Sebagai salah satu refrensi yang memiliki sistem dengan data yang bernilai bilangan bulat dapat dimodelkan dalam Petri net dengan *counter* untuk kemudian dianalisis atau diverifikasi sistemnya.
4. Sebagai salah satu bahan refrensi untuk penelitian selanjutnya khususnya berkaitan pada verifikasi formal dan Petri net dengan *counter*.
5. Sebagai salah satu kontribusi untuk pengembangan ilmu pengetahuan Matematika.

Penelitian Terdahulu

- Franck, Raymond, dan Hanna (2009) pada artikel "*Efficient Reachability Graph Representation of Petri Net With Unbounded Counters*"

Penelitian Terdahulu

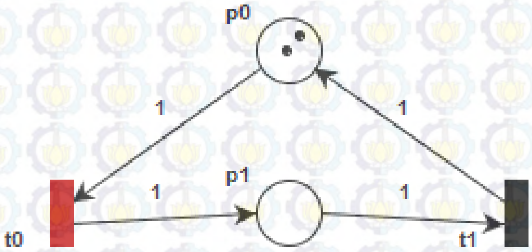
- Franck, Raymond, dan Hanna (2009) pada artikel "*Efficient Reachability Graph Representation of Petri Net With Unbounded Counters*"
- Dieky Adzkiya (2014) pada disertasinya yang berjudul "*Finite Abstraction of Max-Plus-Linear Systems*"

Penelitian Terdahulu

- Franck, Raymond, dan Hanna (2009) pada artikel "*Efficient Reachability Graph Representation of Petri Net With Unbounded Counters*"
- Dieky Adzkiya (2014) pada disertasinya yang berjudul "*Finite Abstraction of Max-Plus-Linear Systems*"
- Marcin, Agnieszka, dan Jerzy (2014) pada artikel "*Methods of Translation of Petri Nets to NuSMV Language*"

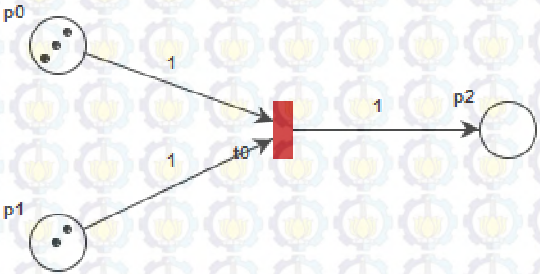
Contoh Petri Net

Petri net merupakan salah satu alat untuk memodelkan sistem *event* diskrit. Informasi mengenai *event* dan keadaan ini masing-masing dinyatakan dengan transisi dan *place*.



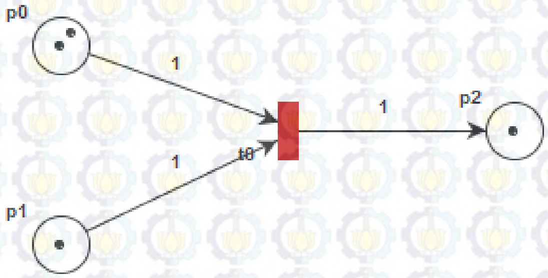
Gambar: Petri Net Sederhana

Contoh Petri Net Bertanda



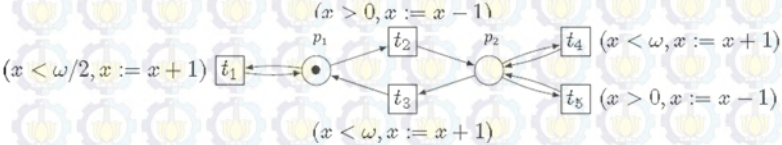
Gambar: Contoh Transisi *Enable*

Contoh Dinamika Petri Net



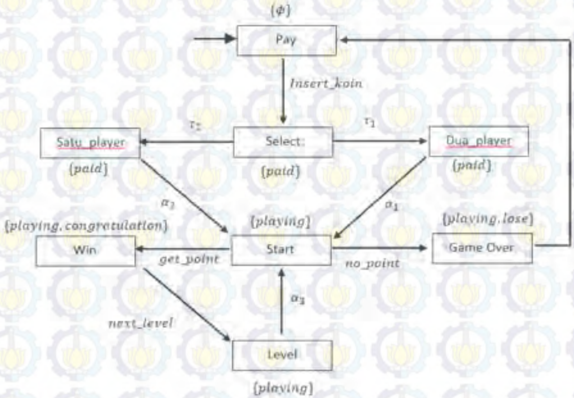
Gambar: Contoh Sesudah Transisi t_0 Di-*fire*

Contoh Petri Net dengan *Counter*



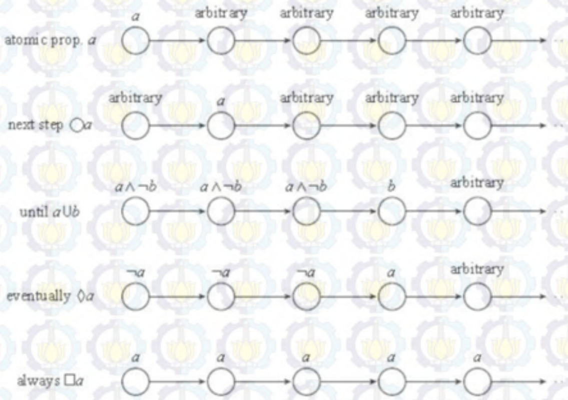
Gambar: Contoh Petri Net dengan *Counter* (PNZ)

Contoh Sistem Transisi



Gambar: Contoh Sistem Transisi pada Mesin Dingdong

Sintaksis dan Semantik LTL



Gambar: Semantik dari Operator Temporal

NuSMV

NuSMV adalah salah satu model *checker* untuk *temporal logic*. Jika diberikan sebuah model dengan state yang berhingga dan satu atau lebih formula, NuSMV dapat digunakan untuk memeriksa secara otomatis apakah model memenuhi spesifikasi tersebut atau tidak. Formula dapat diekpresikan dengan menggunakan LTL.

Tahapan Penelitian

1. Studi literatur.
2. Mengkontruksi sebuah Petri net dengan *counter* dari suatu sistem inventori.
3. Mengubah model menjadi sistem transisi.
4. Melakukan abstraksi berhingga untuk memperoleh state yang berhingga.
5. Mendesain spesifikasi LTL formula.
6. Memverifikasi sistem dengan implementasi pada NuSMV.
7. Menganalisis hasil verifikasi. Pada tahap ini dapat diketahui apakah model memenuhi spesifikasi atau tidak.
8. Publikasi.
9. Penulisan Tesis.

Kontruksi PNZ dari Sistem Inventori

Sistem inventori pada penelitian ini dilakukan oleh seorang Agen dengan transaksi berupa pembelian barang, penyimpanan barang pada gudang, dan penjualan kembali barang. Barang yang diperdagangkan hanya dua barang yaitu barang A dan barang B dengan masing-masing satu satuan. Tidak terdapat biaya pemesanan dan penyimpanan. Barang A dan barang B disimpan dalam satu gudang dengan kapasitas tertentu.

Kontruksi PNZ dari Sistem Inventori

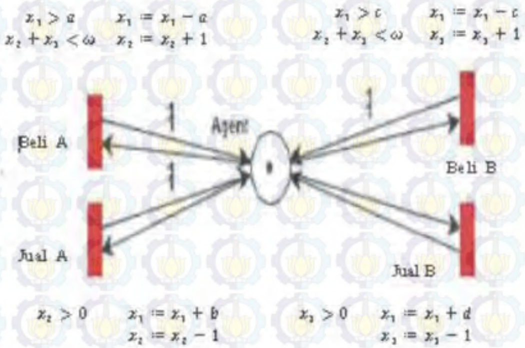
Variabel-variabel yang digunakan yaitu

- x_1 = banyaknya uang,
- x_2 = banyaknya barang A,
- x_3 = banyaknya barang B.

Konstanta-konstanta yang digunakan yaitu

- a = harga beli barang A,
- b = harga jual barang A,
- c = harga beli barang B,
- d = harga jual barang B,
- ω = kapasitas gudang.

PNZ Sistem Inventori



Gambar: PNZ Sistem Inventori

Diberikan nilai awal pada masing-masing variabel yaitu

- $x_1 = 250$
- $x_2 = 0$
- $x_3 = 0$

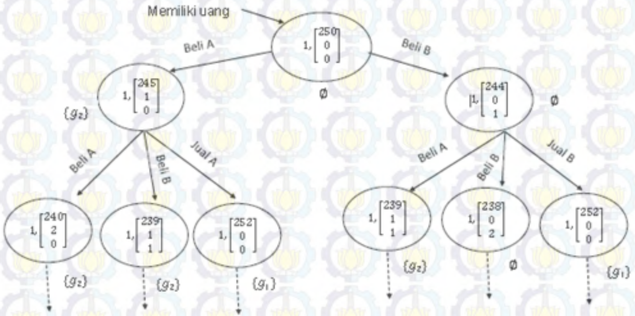
dan diberikan nilai-nilai pada konstanta sebagai berikut

- $a = 5$
- $b = 7$
- $c = 6$
- $d = 8$
- $\omega = 40$

state space:

$$0 \leq x_1$$
$$0 \leq x_2 \leq \omega$$
$$0 \leq x_3 \leq \omega$$

Sistem transisi dari PNZ



Gambar: Pelabelan Sistem Transisi PNZ

Abstraksi Berhingga dari PNZ

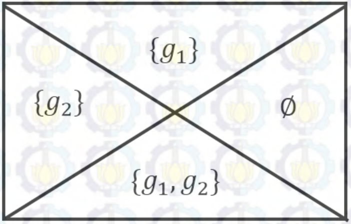
Diberikan *atomic preposition* yaitu $AP = \{g_1, g_2\}$ dengan

- g_1 adalah himpunan state yang memenuhi $\{x | x_1 > 250\}$, dimana g_1 berarti mendapat keuntungan.
- g_2 adalah himpunan state yang memenuhi $\{x | x_2 > 0\}$, dimana g_2 berarti tersedianya barang A dalam gudang.

Abstraksi Berhingga

Langkah-langkah abstraksi berihingga sebagai berikut.

- Partisi state *space* menjadi beberapa kelas ekuivalen, dimana setiap kelas memiliki label (AP) yang sama.



Partisi state space

Berdasarkan *AP* yang diketahui terdapat empat partisi yaitu $\bar{S}_1, \bar{S}_2, \bar{S}_3$, dan $\bar{S}_4$ dsebagai berikut.

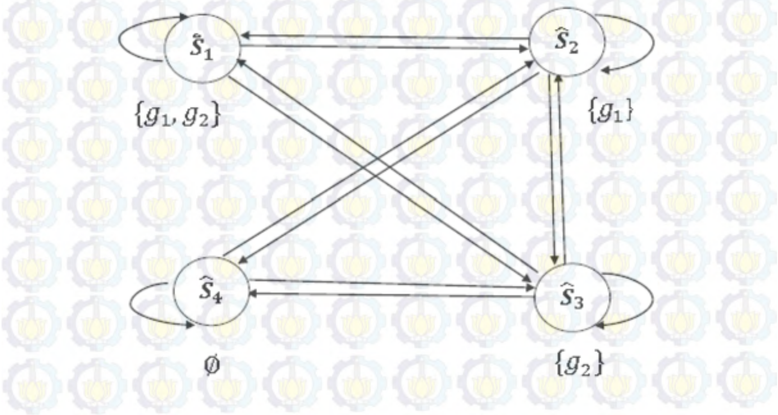
- $g_1 \wedge g_2 = \bar{S}_1 = \{(1, [x_1, x_2, x_3]) | x_1 > 250, x_2 > 0\}$
- $g_1 \wedge \neg g_2 = \bar{S}_2 = \{(1, [x_1, x_2, x_3]) | x_1 > 250, x_2 \leq 0\}$
- $\neg g_1 \wedge g_2 = \bar{S}_3 = \{(1, [x_1, x_2, x_3]) | x_1 \leq 250, x_2 > 0\}$
- $\neg g_1 \wedge \neg g_2 = \bar{S}_4 = \{(1, [x_1, x_2, x_3]) | x_1 \leq 250, x_2 \leq 0\}$

Sehingga diperoleh 4 state abstrak yaitu $\hat{S}_1, \hat{S}_2, \hat{S}_3$, dan $\hat{S}_4$.

Menentukan transisi

1. Transisi-transisi yang mungkin dari $\hat{S}_1$ adalah $\hat{S}_1 \rightarrow \hat{S}_1$, $\hat{S}_1 \rightarrow \hat{S}_2$, dan $\hat{S}_1 \rightarrow \hat{S}_3$.
2. Transisi-transisi yang mungkin dari $\hat{S}_2$ adalah $\hat{S}_2 \rightarrow \hat{S}_1$, $\hat{S}_2 \rightarrow \hat{S}_2$, $\hat{S}_2 \rightarrow \hat{S}_3$, dan $\hat{S}_2 \rightarrow \hat{S}_4$.
3. Transisi-transisi yang mungkin dari $\hat{S}_3$ adalah $\hat{S}_3 \rightarrow \hat{S}_1$, $\hat{S}_3 \rightarrow \hat{S}_2$, $\hat{S}_3 \rightarrow \hat{S}_3$, dan $\hat{S}_3 \rightarrow \hat{S}_4$.
4. Transisi-transisi yang mungkin dari $\hat{S}_4$ adalah $\hat{S}_4 \rightarrow \hat{S}_2$,$\hat{S}_4 \rightarrow \hat{S}_3$, dan $\hat{S}_4 \rightarrow \hat{S}_4$.

Berdasarkan kemungkinan transisi-transisi yang terjadi pada state abstrak, diperoleh sistem transisi abstrak pada gambar dibawah ini.



Gambar: Sistem Transisi Abstrak

Contoh Spesifikasi

Diketahui $AP = \{g_1, g_2\}$ dengan himpunan state yang memenuhi g_1 adalah $\{x|x_1 > 250\}$ sedangkan himpunan state yang memenuhi g_2 adalah $\{x|x_2 > 0\}$, dimana g_1 berarti mendapat keuntungan dan g_2 berarti tersedianya barang A dalam gudang, maka beberapa spesifikasi yang dapat digunakan adalah:

- Pada transaksi berikutnya Agen selalu mendapatkan keuntungan, formulanya nya adalah $\bigcirc(\Box\{x|x_1 > 250\})$ atau $\bigcirc(\Box g_1)$.

Contoh Spesifikasi

- Pada suatu waktu barang A selalu ada dalam gudang, formulanya adalah $\Diamond(\Box\{x|x_2 > 0\})$ atau $\Diamond(\Box g_2)$.

Contoh Spesifikasi

- Pada suatu waktu barang A selalu ada dalam gudang, formulanya adalah $\Diamond(\Box\{x|x_2 > 0\})$ atau $\Diamond(\Box g_2)$.
- Pada suatu waktu yang sama Agen memperoleh keuntungan dan tersedianya barang A dalam gudang , formulanya adalah $\Diamond(\{x|x_1 > 250\} \wedge \{x|x_2 > 0\})$ atau $\Diamond(g_1 \wedge g_2)$.

Contoh Spesifikasi

- Pada suatu waktu barang A selalu ada dalam gudang, formulanya adalah $\Diamond(\Box\{x|x_2 > 0\})$ atau $\Diamond(\Box g_2)$.
- Pada suatu waktu yang sama Agen memperoleh keuntungan dan tersedianya barang A dalam gudang , formulanya adalah $\Diamond(\{x|x_1 > 250\} \wedge \{x|x_2 > 0\})$ atau $\Diamond(g_1 \wedge g_2)$.
- Pada saat Agen tidak mendapat keuntungan tetapi memiliki barang A dalam gudang, maka suatu saat Agen dapat memperoleh keuntungan, formulanya adalah $\Diamond((\{x|x_1 \leq 250\} \wedge \{x|x_2 > 0\}) \rightarrow \{x|x_1 > 250\})$ atau $\Diamond((\leftharpoondown g_1 \wedge g_2) \rightarrow \bigcirc g_1)$.

Implementasi

```
MODULE main
VAR
        state : {s1, s2, s3, s4};
ASSIGN
        init(state) := s1;
        next(state) := case
                state = s4 : {s2, s3};
                state = s3 : {s1, s2, s4};
                state = s2 : {s1, s3, s4};
                state = s1 : {s2, s3};
                TRUE : {s1, s2, s3, s4};
        esac;
DEFINE
        g1 := state=s1 | state=s2;
        g2 := state=s1 | state=s3;
LTLSPEC X (G g1);
LTLSPEC F (G g2);
LTLSPEC F (g1 & g2);
LTLSPEC F (! g1 & g2 -> X g1);
```

Gambar: Sistem Transisi Abstrak pada NuSMV

Hasil Implementasi

```
-- specification  X ( G g1)  is false
-- as demonstrated by the following execution sequence
Trace Description: LTL Counterexample
Trace Type: Counterexample
  -> State: 1.1 <-
    state = s1
    g2 = TRUE
    g1 = TRUE
  -> State: 1.2 <-
    state = s3
    g1 = FALSE
  -- Loop starts here
  -> State: 1.3 <-
    state = s1
    g1 = TRUE
  -> State: 1.4 <-
    state = s2
    g2 = FALSE
  -> State: 1.5 <-
    state = s1
    g2 = TRUE
-- specification  F ( G g2)  is false
-- as demonstrated by the following execution sequence
Trace Description: LTL Counterexample
Trace Type: Counterexample
  -- Loop starts here
  -> State: 2.1 <-
    state = s1
    g2 = TRUE
    g1 = TRUE
  -> State: 2.2 <-
    state = s2
    g2 = FALSE
  -> State: 2.3 <-
    state = s1
    g2 = TRUE
-- specification  F (g1 & g2)  is true
-- specification  F ((!g1 & g2) -> X g1)  is true
```

Analisa Hasil Implementasi

- Spesifikasi $\bigcirc(\Box g_1)$ yang berarti bahwa pada transaksi berikutnya Agen selalu mendapatkan keuntungan,spesifikasi tidak dipenuhi dengan contoh *counterexample* yaitu pada state 1.2 state S_3 dituliskan $g_1 = false$ karena pada S_3 yang berarti $\hat{S}_3$ memliki label $\{g2\}$ berarti bahwa pada state tersebut $\{x_1 \leq 250, x_2 > 0\}$ sedangkan yang diinginkan adalah state berikutnya selalu $\{x|x_1 > 250\}$, maka spesifikasi tidak dipenuhi.

Analisa Hasil Implementasi

- Spesifikasi $\bigcirc(\Box g_1)$ yang berarti bahwa pada transaksi berikutnya Agen selalu mendapatkan keuntungan,spesifikasi tidak dipenuhi dengan contoh *counterexample* yaitu pada state 1.2 state S_3 dituliskan $g_1 = false$ karena pada S_3 yang berarti $\hat{S}_3$ memliki label $\{g2\}$ berarti bahwa pada state tersebut $\{x_1 \leq 250, x_2 > 0\}$ sedangkan yang diinginkan adalah state berikutnya selalu $\{x|x_1 > 250\}$, maka spesifikasi tidak dipenuhi.
- Spesifikasi $\Diamond(\Box g_2)$ yang berarti bahwa pada suatu waktu barang A selalu ada dalam gudang, spesifikasi tidak dipenuhi dengan contoh *counterexample* yaitu pada state 2.2 state S_2 dituliskan $g_2 = false$ karena pada S_2 yang berarti $\hat{S}_2$ memliki label $\{g1\}$ berarti bahwa pada state

Analisa Hasil Implementasi

- Spesifikasi $\Diamond(g_1 \wedge g_2)$ yang berarti bahwa pada suatu waktu yang sama akhirnya Agen memperoleh keuntungan dan tersedianya barang A dalam gudang , terpenuhi. Sebagai contoh state S_1 ke S_1.

Analisa Hasil Implementasi

- Spesifikasi $\Diamond(g_1 \wedge g_2)$ yang berarti bahwa pada suatu waktu yang sama akhirnya Agen memperoleh keuntungan dan tersedianya barang A dalam gudang , terpenuhi. Sebagai contoh state S_1 ke S_1.
- Spesifikasi $\Diamond((\leftharpoondown g_1 \wedge g_2) \rightarrow \bigcirc g_1)$ yangberarti pada saat Agen tidak mendapat keuntungan tetapi memiliki barang A dalam gudang, maka akhirnya suatu saat Agen dapat memperoleh keuntungan, terpenuhi. Sebagai contoh yang memenuhi $(\leftharpoondown g_1 \wedge g_2)$ adalah state S_3, terdapat transisi $S_3 \rightarrow S_1$ dan $S_3 \rightarrow S_2$ sedemikian sehingga memenuhi dari S_3 yang berarti $\{x_1 \leq 250, x_2 > 0\}$ berikutnya dapat menjadi $\{x_1 > 250, x_2 > 0\}$ atau menjadi $\{x_1 > 250\}$).

- Jadi dengan *initial* state $\hat{S}_1$, sistem transisi abstrak ada yang memenuhi spesifikasi dan ada yang tidak memenuhi spesifikasi.

- Jadi dengan *initial* state $\hat{S}_1$, sistem transisi abstrak ada yang memenuhi spesifikasi dan ada yang tidak memenuhi spesifikasi.
- Apabila sistem transisi abstrak memenuhi spesifikasi, maka sistem transisi asli pasti juga memenuhi spesifikasi tersebut.

Kesimpulan

1. Dengan menerapkan metode abstraksi berhingga, sistem transisi dari PNZ yang memiliki jumlah state tak berhingga dapat dibentuk menjadi sistem transisi abstrak dengan jumlah state berhingga.
2. Berdasarkan implementasi pada NuSMV, diperoleh hasil verifikasi terhadap sistem transisi abstrak dari PNZ sebagai berikut.
   - Spesifikasi $\bigcirc(\Box g_1)$ yang berarti bahwa pada transaksi berikutnya Agen selalu mendapatkan keuntungan, tidak dipenuhi.

- Spesifikasi $\Diamond(\Box g_2)$ yang berarti bahwa pada suatu waktu barang A selalu ada dalam gudang, tidak dipenuhi.
- Spesifikasi $\Diamond(g_1 \wedge g_2)$ yang berarti bahwa pada suatu waktu yang sama akhirnya Agen memperoleh keuntungan dan tersedianya barang A dalam gudang , dipenuhi.
- Spesifikasi $\Diamond((\leftharpoondown g_1 \wedge g_2) \rightarrow \bigcirc g_1)$ yang berarti pada saat Agen tidak mendapat keuntungan tetapi memiliki barang A dalam gudang, maka akhirnya suatu saat Agen dapat memperoleh keuntungan, dipenuhi.

Jika sistem transisi abstrak memenuhi spesifikasi, maka sistem transisi kongkrit-nya juga memenuhi spesifikasi. Tetapi, jika sistem transisi abstrak tidak memenuhi spesifikasi, tidak dapat dikatakan sistem transisi kongkrit-nya tidak memenuhi spesifikasi.

Saran

Pada tesis ini hanya dibahas mengenai abstraksi dan verifikasi formal PNZ pada sistem inventori, diharapkan penelitian selanjutnya dapat mengkonstruksi metode abstraksi berhingga secara umum pada PNZ. Juga dapat menggunakan sistem lain agar dapat diketahui bagaimana verifikasi formalnya.

Daftar Pustaka

- Adzkiya, D., (2014), *Finite Abstraction of Max-Plus-Linear Systems*, Disertasi, Technische Universiteit Delft, Deflt.
- Barier, C., dan Katoen, J. P., (2007), *Principle of Model Checking*, The Mit Press, Cambrigdge.
- Cassandras, C. G., dan Lafortune, S., (2008), *Introduction to Discrete Event Systems Second Edition*, New York: Springer.
- Murata, T., (1989), Petri Nets: Properties, Analysis and Applications, *Proceding of The IEEE*, hal. 541-580.
- Pommereau, F., Devillers, R., dan Klaudel, H., (2009), Efficient Reachability Graph Representation of Petri Nets With Unbounded Counters, *Electronic Notes in Theoretical Computer Science*, Vol. 239, hal. 119-129.

Daftar Pustaka

- Subiono, (2015), *Aljabar Min-Max Plus dan Terapannya*, Instittut Teknologi Sepuluh Nopember, Surabaya.
- Szpyrka, M., Biernacka, A., dan Biernacki, J., (2014), Methods of Translation of Petri Nets to NuSMV Language, Ed: Zeugman, L., *CEUR Workshop Proceedings*, Germany, Vol. 1269, hal. 245-256.