# Laporan Tugas Besar Dasar Pemodelan dan Simulasi *Nagel-Shreckenberg Model*

Oleh:

Kelompok 2 (Muhammad Affan H.(1301174618), M Aqiel Hilman M.(1301170428),

Yazid Rahman A.(1301174194) )

## Bab I. Pendahuluan

Model *Nagel-ShreckenbergI* adalah suatu model yang men-simulasikan suatu kondisi kepadatan lalu lintas. Model ini merupakan suatu jenis *Cellular automation* untuk aliran kendaraan yang dapat menirukan suatu kemacetan, seperti memperlihatkan penurunan rata rata kecepatan kendaraan jika jalanan yang dilalui sedang padat. Model ini menunjukkan bahawa kemacetan merupakan suatu fenomena kolektif yang terjadi dikarenakan adanya interakasi antara kendaraan yang ada di lintasan atau jalanan, pada saat kepadatan lalu lintas sedang padat, maka jarak antara kendaraan juga semakin rendah

Pada simulasi ini hanya akan kami lakukan dengan 1 jalur dan akan ada beberapa parameter sebagai berikut:

M = 100 # Panjang lintasan

p = 0.3 # Probabilitas pengereman

v0 = 0 # kecepatan awal

N = 10 # banyak kendaraan

tmax = 100 # waktu maksimum

vmax = 5 # kecepatan maksimum

dt = 1 # time step

## Bab II. Ketepatan Algoritma

Berikut ini adalah algoritma yang kami gunakan untuk simulasi:

* Fungsi getJarak (figure 1) adalah fungsi yang digunakan untuk mengukur jarak antara suatu kendaraan dengan kendaraan di depannya, fungsi ini memiliki 4 kondisi, yaitu;
  + Kondisi untuk mobil terakhir untuk mengukur jarak dengan mobil yang ada di paling belakang karena lintasaanya merupakan siklis,
  + Kondisi untuk menghitung jarak apabila mobil terakhir berada di belakang mobil paling awal
  + Kondisi umum apabila suatu mobil ke-i berada di belakang mobil ke-i+1
  + Kondisi apabila suatu mobil ke-i berada di ujung lintasan dan mobil ke-i+1 ada di awal lintasan
* Fungsi getGas, fungsi getgas berfungsi untuk memberi perubahan posisi pada kendaraan, sama halnya apabila pemilik kendaraan mengakselerasi kendaraannya, didalam fungsi getGas ini, terdapat probabilitas random yang memungkinkan kendaraan melakukan pengurangan kecepatan secara tiba tiba karena sesuatu yang random (pada kenyataan mungkin saja ada seseorang menyebrang jalan tba tiba, dan sejenisnya), perubahan posisi ini disimpan pada suatu array yang kemudian akan di aplikasikan pada fungsi updatePosisi
* Fungsi updatePosisi, fungsi ini mengaplikasikan perubahan posisi yang telah dilakukan pada fungsi sebelumnya (fungsi getGas)
* Program utama, untuk menjalankan simulasi

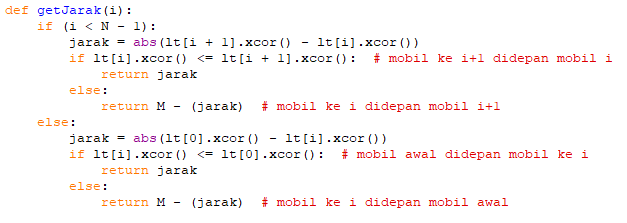


Figure fungsi getJarak

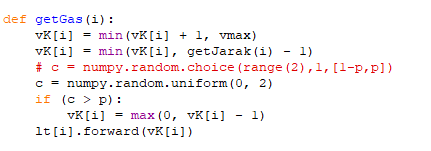


Figure fungsi getGas

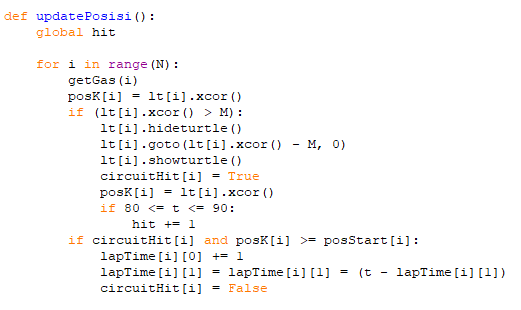


Figure fungsi UpdatePosisi

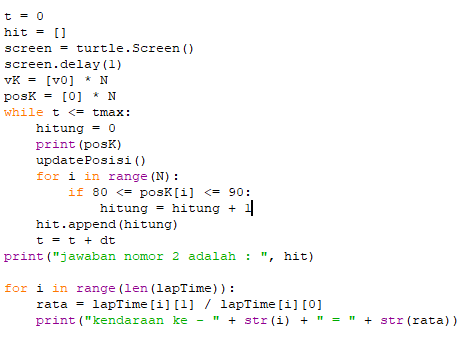


Figure program utama

## Bab III. Hasil Simulasi

Berikut ini adalah hasil simulasi yang dijalankan, dimana baris paling atas menunjukkan kepadatan pada rentang waktu yang telah ditentukan (80 – 90).

Dan sisanya merupaakn waktu rata rata suatu kendaraan untuk mebali ke tempatnya.

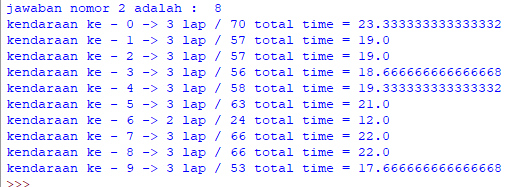


Figure 5 Hasil simulasi

## Bab IV. Analisis

Untuk kepadatan kendaraan pada interval *x*80-*x*90 terdapat 6 kendaraan yang dapat melintasi lintasan secara penuh.

Untuk perhitungan rata rata kecepatan setiap kendaraan untuk kembali ke posisi awalnya didapatkan dengan cara menghitung berapa kali kendaraan tersebut melewati posisi awalnya dan waktu yang dibutukan untuk sampai ke posisi awalnya lagi, kemudian parameter itu diakumuluasikan dan dibagi untuk mendapatkan rata rata waktu setiap kendaraan untuk mencapai posisi awalnya.