

Modelare și Simulare

Traffic Light Phases Optimization

Florea Alexandru-Daniel

1.Descrierea sistemului modelat

În acest proiect, se simulează trei intersecții diferite cu semafoare. De la stânga spre dreapta, prima intersecție este cea în cruce, conectând 4 drumuri, iar următoarele două intersecții sunt în formă de T, conectând 3 drumuri fiecare. Mașinile vin de pe toate cele 6 drumuri exterioare și pot ieși pe diverse ieșiri în funcție de drumul pe care intră, evitându-se cazurile inutile, proiectul concentrându-se posibilitățile de mișcare ale mașinilor care pot duce la congestionarea traficului.

Modelul urmărește fluidizarea traficului prin creșterea sau reducerea timpului culorii verzi a semafoarelor. Având 3 intersecții diferite, se poate observa, în momentul în care vin mai multe mașini care dintre aceste intersecții devin aglomerate și îngreunează traficul celorlalte intersecții.

Pentru fiecare semafor avem câte un slider, pentru a crește sau a reduce timpul culorii verzi a semaforului. Schimbând aceste valori, se poate crește sau scade timpul petrecut de fiecare mașină în intersecții, precum și numărul de mașini care trec prin aceste intersecții.

Între fiecare schimbare a unui semafor, se aprinde culoarea galbenă a semaforului pentru 3 secunde.

În model mai există și un grafic de tip histogramă care conține numărul de mașini care au ieșit din intersecții. Tot odată, sub grafic se regăsește timpul mediu petrecut de o mașină în intersecții. Odată cu

Mașinile din model sunt modelate ca și agenți, iar drumurile și partea logică sunt modelate cu ajutorul Road Traffic Library.

Valorile numerice care caracterizează agenții de tip Car sunt:

Numele valorii	Descrierea	Valoarea
Length	Lungimea mașinii	5 meter
Initial speed	Viteza cu care intră mașina în model	60 km/h
Preferred speed	Viteza cu care mașina vrea să conducă	60 km/h
Max acceleration	Accelerația maximă pe care o poate avea mașina	1,8m/s ²
Max deceleration	Valoarea maximă a frânei pe care o poate face mașina	4,2 m/s ²

Agenții de tip Car, au o variabilă `startTime`, de tip `double` care ia valoarea timpului la care mașina a intrat în model. Această se folosește pentru a calcula timpul mediu petrecut de o mașină în model.

Alte valori care se găsesc în model sunt valorile de timp pentru fiecare semafor, acestea au fiecare niște valori implicite cu care încep, cât și anumite intervale din care pot lua valori. Toate aceste valori au tipul `Time`.

Numele	Descrierea	Valoarea minimă	Valoarea implicită	Valoarea maximă
pX_WE	Semafoarele de vest și est ale primei intersecții	1	30	60
pX_NS	Semafoarele de nord și sud ale primei intersecții	1	20	60
pT1_WE	Semafoarele de vest și est ale celei de-a doua intersecții	1	35	60
pT1_N	Semaforul din nord a celei de-a doua intersecții	1	15	60
pT2_WE	Semafoarele de vest și est ale ultimei intersecții	1	20	60
pT2_S	Semaforul din sud al ultimei intersecții	1	40	60

2. Tehnica de modelare folosită

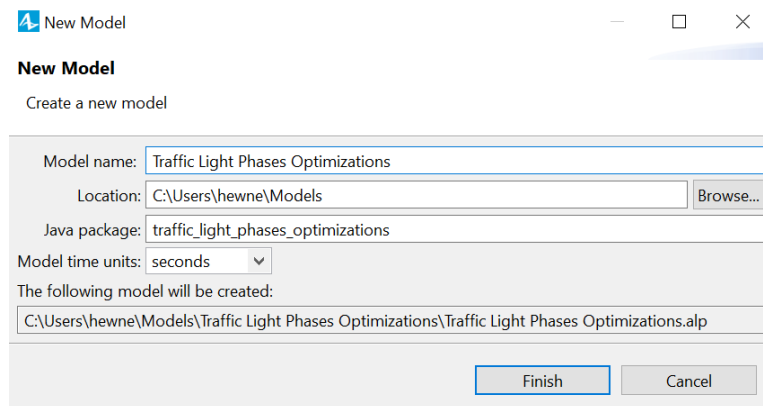
Tehnica de modelare folosită este DE – Discrete Event. Aceasta este un tip de simulare care consideră un sistem ca și o colecție discretă de evenimente, în care fiecare eveniment are un efect definit în sistem. Aceasta este folosită în aplicații de logistică sau spre exemplu aplicații care simulează modul de funcționare al unei fabrici. Aceste evenimente se pot aplea prin mai multe metode: pot fi apelate atunci când s-a terminat un alt eveniment, pot fi programate să fie apelate la un anumit moment, sau chiar la întâmplare.

O simulare de tip Discrete Event, de obicei evidența statisticilor sistemului. Spre exemplu, în această simulare, statisticile care se folosesc sunt timpul mediu al unei mașini în intersecții și numărul de mașini care ies.

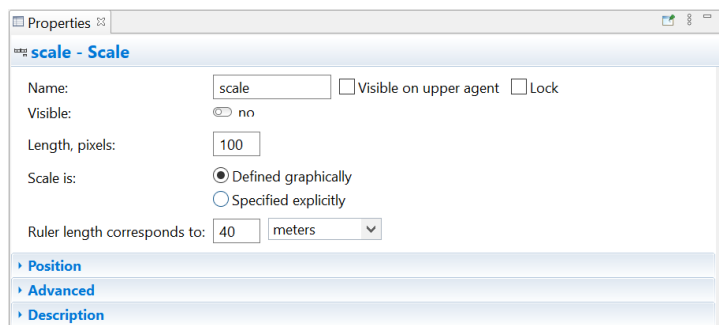
3. Pașii de construire a modelului

1. Crearea modelului

În partea de sus a programului, apăsați pe **File->New->Model**, introducem numele modelului, în acest caz Traffic Light Phases Optimization, alegem locația unde să se salveze proiectul și alegem unitatea de timp a modelului să fie în **secunde**.

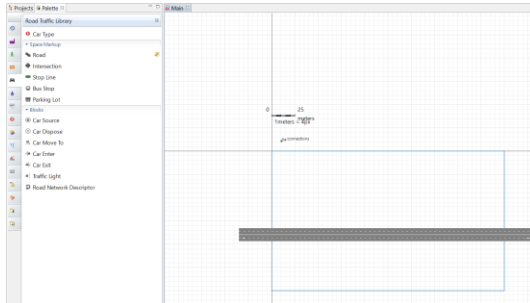


Primul lucru pe care trebuie să îl faceți pentru acest model, este să setați scale-ul la 40 de metri ca și în poza de mai jos.

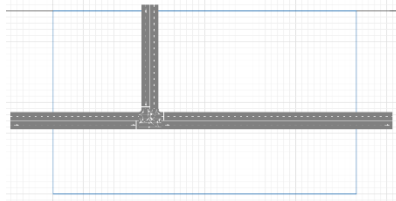


2. Crearea unui drum și intersecții

Din secțiunea Palette, alegeți din meniul din stânga **Road Traffic Library**. Faceți dublu click pe **Road**, care va porni modul de desenare, după care faceți un click în model la începutul drumului, iar apoi dublu click la finalul drumului, ca și în poza de mai jos.



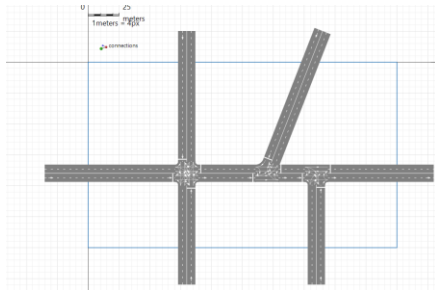
Pentru crearea unei intersecții, trebuie să se mai creeze un drum. Faceți din nou dublu click pe elementul Road din paleta de Road Traffic Library și creați un nou drum perpendicular pe primul.



Acest lucru v-a duce la crearea unei intersecții în formă de T.

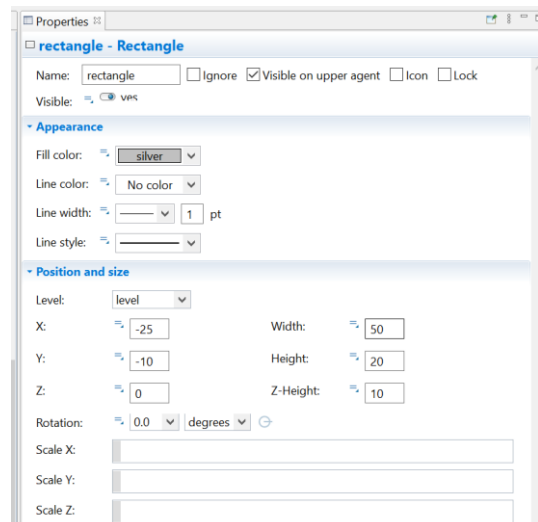
Pentru crearea unei intersecții de tip cruce, trebuie să mai creați încă un drum, venind din partea de jos.

Procedați la fel ca și mai sus pentru a obține intersecțiile ca și în modelul final.

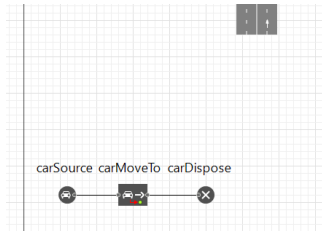


3. Crearea mașinilor

Se crează un Agent type cu numele Car, fără nici o animație și fără parametrii. Din paleta de Presentation, tragem un rectangle. În proprietățile acestuia, la categoria de Appearance selectăm silver la fill color și no color la line color. Iar la categoria de Position and size, folosiți atributele din poza din dreapta. Din paleta de Agent, creați o variabilă cu numele startTime, de tip double și cu valoarea inițială "time()". În proprietățile agentului, la categoria de Agent actions, On startup se va introduce codul următor: `rectangle.setFill(spectrumColor(uniform_discr(10), 10));`



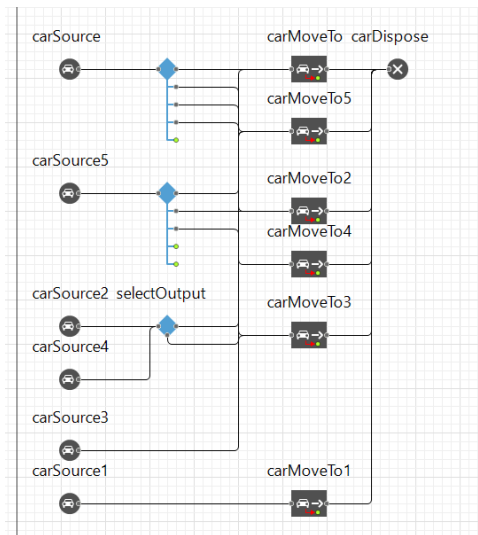
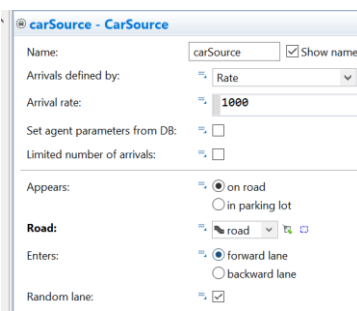
Pentru a crea mașini, ne vom folosi de block-urile din paleta de Road Traffic Library. Pentru început, trageți blocul **Car Source** sub model, după care trageți și un block **Car Move To** imediat lângă el, în așa fel încât să fie unite. După acest block, trageți și un block **Car Dispose**. La fiecare dintre aceste block-uri, vom selecta la categoria de Car la New car, agentul Car pe care l-am creat.



După aceste operații, modelul ar trebui să arate ca și cel din poza din stânga.

În proprietățile block-ului car source, la **Road**, selectați drumul pe care trebuie să vină mașina, iar la Enters, selectați banda pe care vine mașina. Numele drumurilor se pot afla dacă faceți un click pe ele. Tot în proprietățile acestui block se selectează Arrival rate-ul conform proiectului original și se selectează per second ca unitate.

Creați câte un Car Source block și un Car Move To pentru fiecare drum, și selectați drumul la atributul Road din Properties. La final conectați fiecare block de tip Car Move To la block-ul car Dispose. Acest lucru se poate face dacă dați dublu click pe bulina din dreapta a block-ului, și faceți un path până la car Dispose.

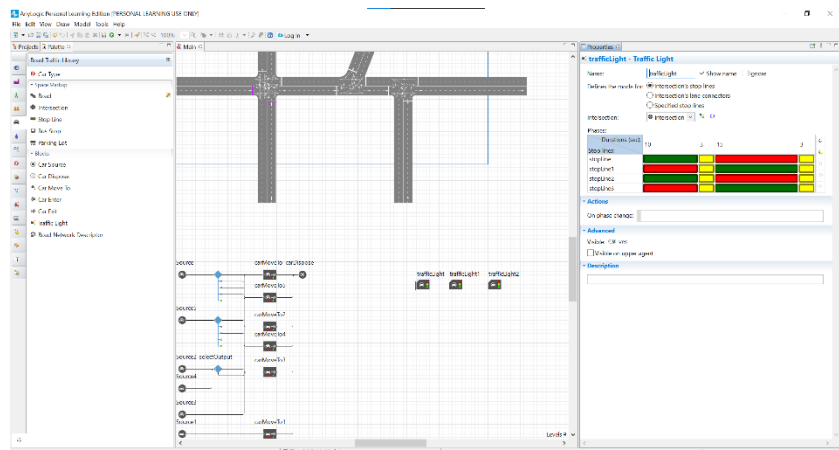


4. Alegirile drumurilor mașinilor

Din paleta de Process Modeling Library, se folosește câte un block de tip select Output pentru fiecare carSource necesar. Probabilitățile fiecărui output se modifică precum este necesar, spre exemplu în primul fiecare probabilitate în afară de 5 este 0,2, iar 5-ul este 0, deoarece avem doar 4 output-uri posibile. După acestea ar trebui să se obțină diagrama din dreapta.

5. Adăugarea de semafoare

Din paleta de Road Traffic Library, se adaugă 3 block-uri de tip Traffic Light. La fiecare se alege câte o intersecție. Fiecare intersecție se configurează conform proiectului original. Spre exemplu, primul semafor, pentru intersecția din stânga, are 4 faze, una în care semafoarele din nord și sud sunt verzi, după care 3 secunde de galben, după care semafoarele din vest și est sunt verzi, după care încă 3 secunde de galben. Modelul ar trebui să arate ca poza din dreapta.

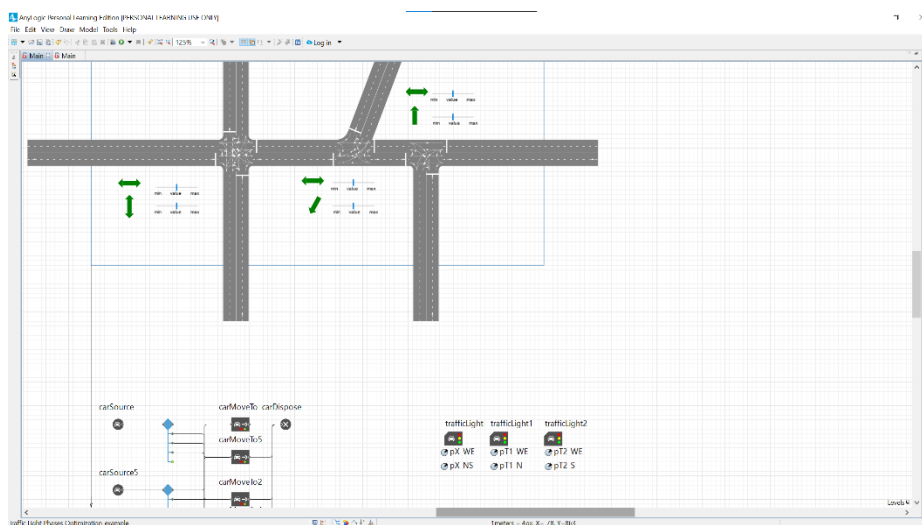


6. Adăugarea de parametrii și slidere

Din paleta de Agents, vom adăuga câte 2 parametrii pentru fiecare semafor. Denumirile acestora vor fi formate din litera “p” de la parametru, urmat de tipul intersecției (T sau X) urmat de “_” și orientarea acestuia. Parametrii vor fi de tip Time, cu unitatea secunde și valorile implicite din tabelul de la capitolul 1. În categoria de Value editor, la Label vom pune Tipul intersecției urmat de orientarea acestuia, neprescurtat. Fiecare dintre acești parametrii trebuie introdus în blocurile de tip semafor aferente, la durata de tip a culorii verzi a semaforului pe care o modifică fiecare.

Din paleta de Controls, tragem câte un slider pentru fiecare parametru. Selectăm pentru fiecare în proprietățile acestora la Link to, parametrul pe care îl modifică. Modificăm valorile minime și maxime conform tabelului din capitolul 1. La categoria de Action, vom pune “timeInSystem.reset();”.

Pentru crearea săgeții, selectați din Paleta Presentation Polyline și faceți forma săgeții. Aceasta nu va avea line color și va avea culoarea green. După crearea acesteia, o puteți duplica pentru fiecare slider. Pentru fiecare slider, adăugați din paleta de presentation 3 texte, min, value și max. În textul value, adăugați „slider.getValue()”, unde slider reprezintă numele slider-ului sub care se află textul. După toate aceste schimbări, modelul ar trebui să arate ca și în poza din dreapta.

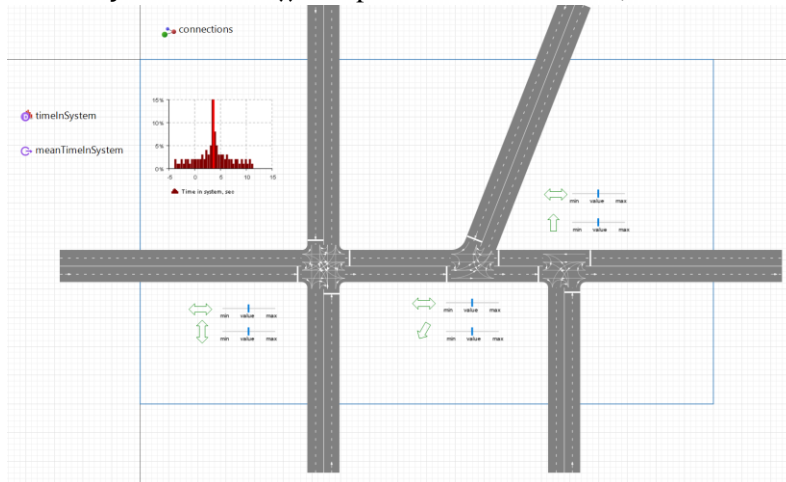


7. Adăgarea Histogramei

Din paleta de Analysis, adăugăm un element de tip Histogram Data și îl numim timeInSystem. La number of intervals schimbăm în 1, iar la Data update, selectăm Do not update data automatically. Tot din aceeași paleta adăugăm și un element Output, pe care îl numim meanTimeInSystem, și la care îi dăm valoarea “timeInSystem.mean()”.

Din aceeași paletă, adăugăm și un chart de tip Histogram, îl numim timeInSystemSeconds și bifăm Show mean. La Data la Histogram punem timeInSystem, și selectăm de la Appearance, la Bars relative width 100%.

În block-ul `carDispose`, adăugăm codul `"timeInSystem.add(time()-car.startTime);"`. Peste diagramă, în partea dreaptă sus, se adaugă textul „Cars exited: „ și codul `"Cars exited: " + timeInSystem.count();"`. După aceste schimbări, modelul ar trebui să arate ca și în poză.



8. Adăugarea de view-uri

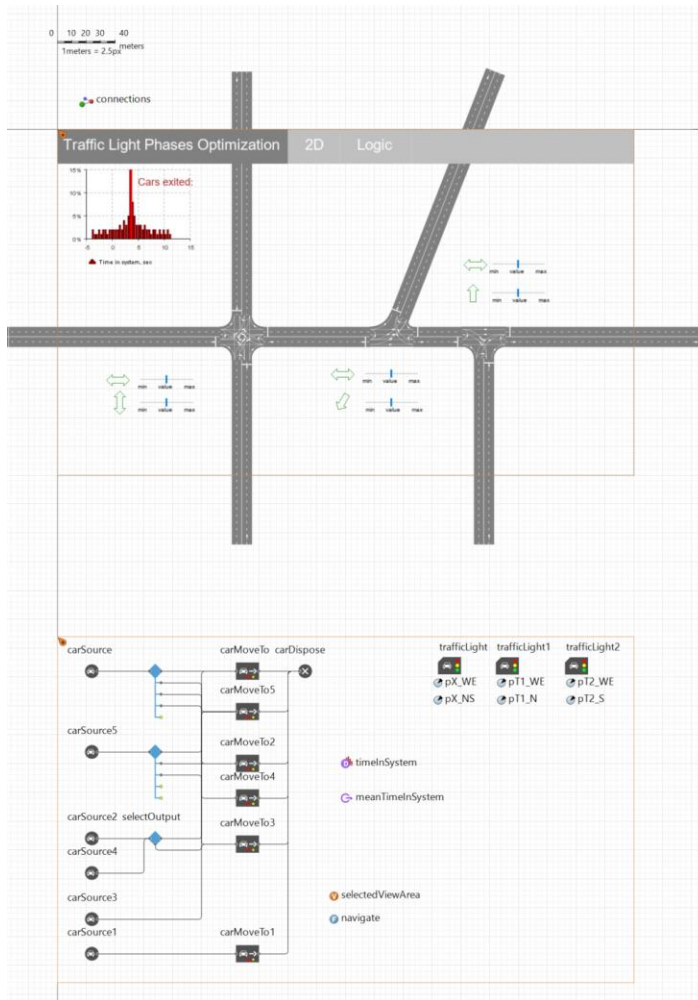
Din paleta de Presentation, dăm tragem un obiect de tip view Area. Repoziționăm elementele astfel încât să încapă în view. La nume punem `viewLogic`, iar la Title punem `Logic`. Adăugăm încă un element de tip View Area peste view-ul principal. Acesta va avea numele `view2D` și titlul `2D`.

Adăugăm o variabilă numită `selectedViewArea`, de tipul `ViewArea`, și cu valoarea inițială de `view2D`. Tot din paleta de Agents, adăugăm o funcție numită `navigate`, cu argumentul `viewArea` și body-ul următor:

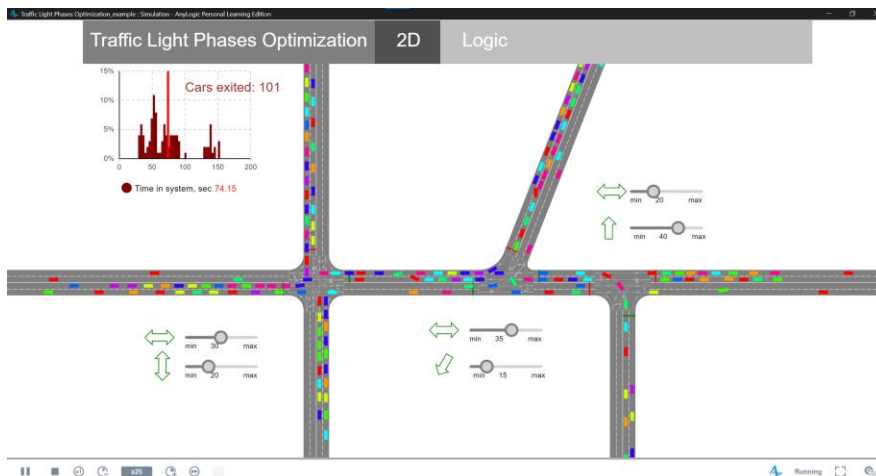
```
selectedViewArea = viewArea;
viewArea.navigateTo();
groupMainMenu.setPos( viewArea.getX(), viewArea.getY() );
```

Pentru realizarea unui meniu, adăugăm 4 obiecte de tip rectangle din paleta Presentation. Primul de culoare gri închis. Al doilea rectangle va avea la atributul de fill color `"selectedViewArea == view2D ? new Color(80, 80, 80) : silver"`, iar la Advanced, On click `"navigate(view2D);"`. Al treilea dreptunghi, va avea la atributul de fill color `"selectedViewArea == viewLogic ? new Color(80, 80, 80) : silver"`, iar la Advance, On click `"navigate(viewLogic);"`. Ultimul dreptunghi va avea culoarea silver. Din aceste dreptunghiuri și text se va face un grup numit `groupMainMenu`. Peste primul dreptunghi se va pune text-ul `Traffic Light Phases Optimization`, peste al doilea textul `2D`, iar peste al treilea textul `Logic`.

În acest punct modelul este gata, și ar trebui să arate ca și în poza de mai jos.



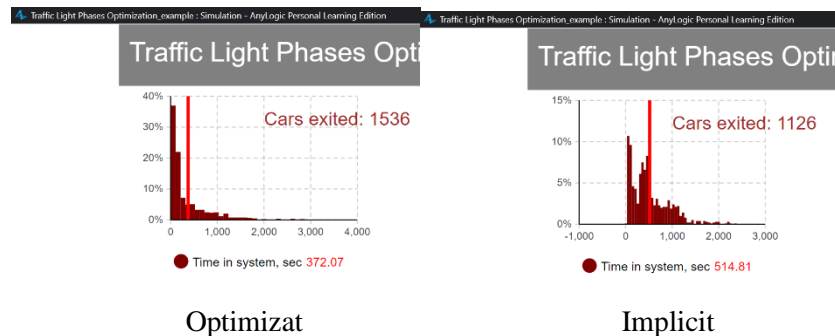
Pasul final este de a rula modelul, acest lucru se face din meniul de sus de pe butonul verde de run. Odată rulat, putem schimba valorile timpilor de la semafoare și view-ul în care ne aflăm.



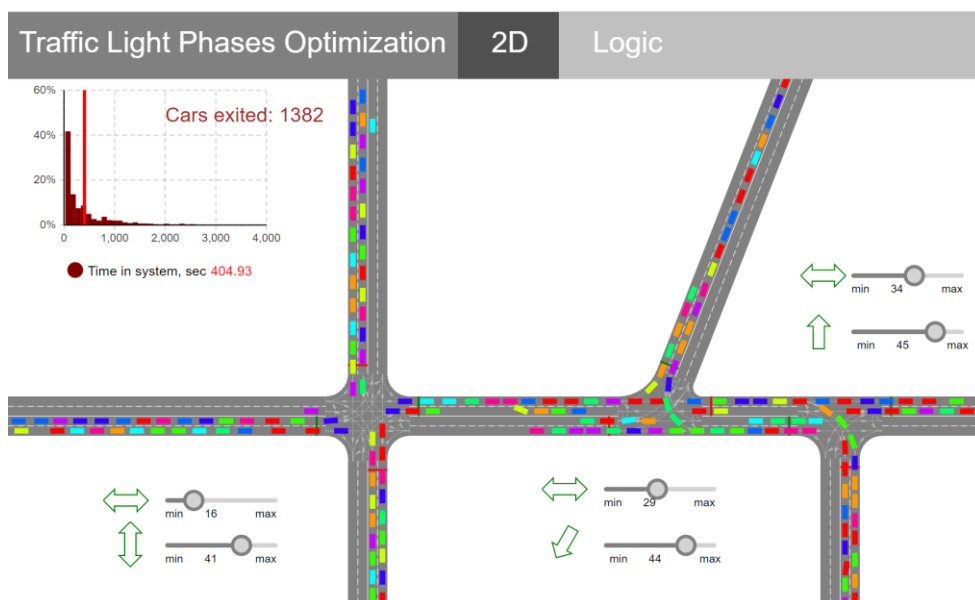
4. Valorile indicilor de performanță și experimente

Indicii pe performanță ai modelului sunt cele 6 variabile care definesc timpul culorii verzi a semaforului. În funcție de aceștia, poate crește sau scade numărul de mașini care trec prin intersecții. Scopul este de a trece cât mai multe mașini și de a sta cât mai puțin în intersecții.

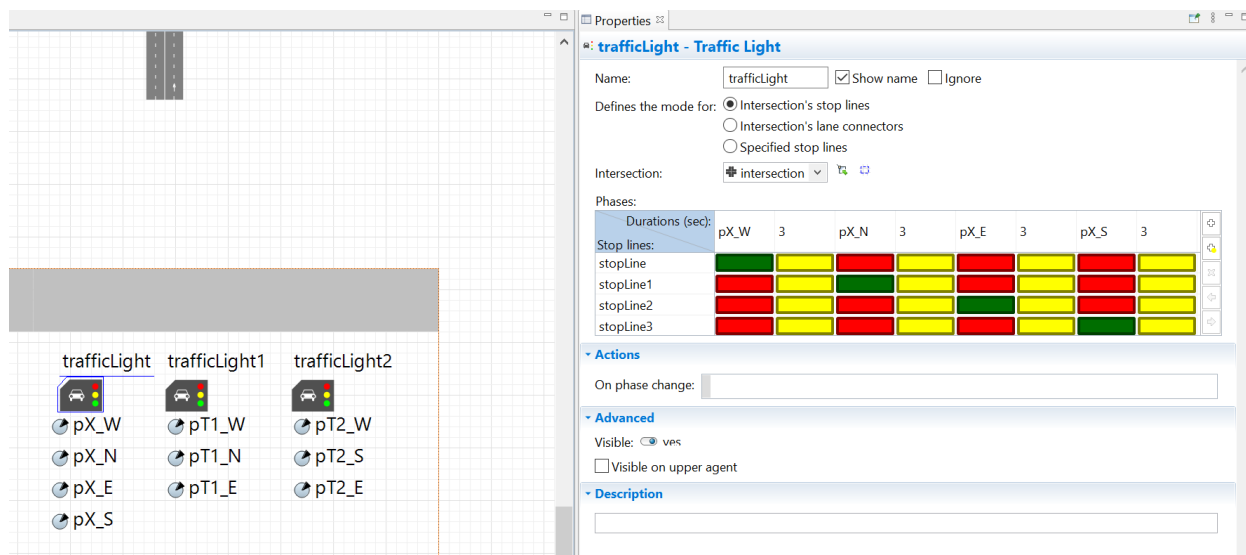
Dacă rulăm un nou experiment, de optimizare, vom obține cele mai bune valori pentru aceste intersecții, folosindu-ne de timpul mediu petrecut de o mașină în simulare ca și optimisation objective. Modificăm timpul de rulare pentru a face ca simularea să dureze 30D de minute (simulation time). Începând cu valori minime de 15 și valori maxime de 30, am rulat multiple experimente, după fiecare experiment am luat cele mai bune valori și le-am pus ca valori de început pentru următorul experiment de optimizare. După ce am rulat asta de mai multe ori, am crescut încet valorile maxime până la 50 în final. După ce am obținut cele mai bune rezultate, le-am înlocuit cu valorile implicite ale parametrilor, iar rezultatele se pot observa în pozele de mai jos.



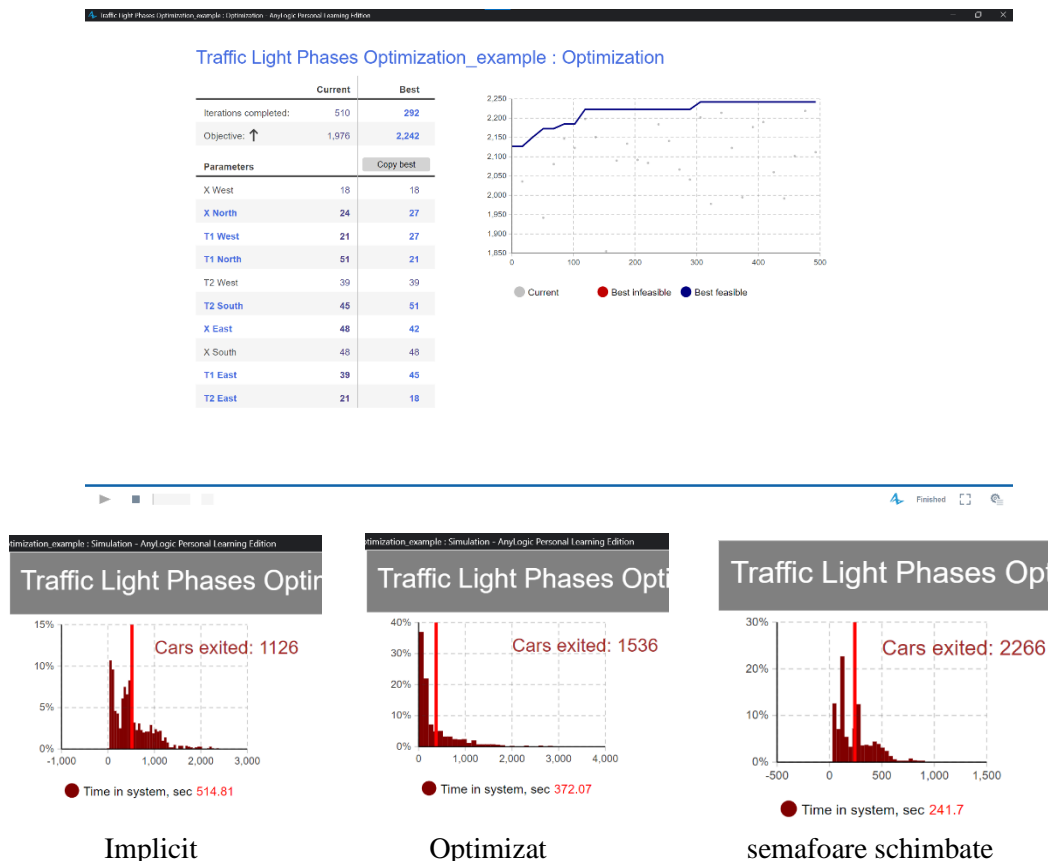
Se poate observa faptul că folosind valorile obținute în urma proceselor de optimizare, a scăzut timpul mediu pe care îl petrece o mașină în intersecții, iar numărul de mașini care trec a crescut. Însă, chiar dacă numerele sunt mai bune, în timpul simulării încă se pot observa îngreunarea sau chiar blocarea unor intersecții în unele momente, fapt ce arată că încă este loc de îmbunătățire.



Al doilea experiment pe care l-am rulat implică schimbarea semafoarelor astfel încât o un singur semafor din cele 4 (sau 3) ale unei intersecții să fie verde la un moment dat. Acest lucru înseamnă ca pentru primul semafor vom avea 4 parametrii, iar pentru celelalte 3 vom avea câte 3 parametrii pentru fiecare. Parametrii și fazele primului semafor se găsesc în poza de mai jos.



După mai multe succesiune de rulare a experimentului de optimizare asupra noului model, se poate observa o îmbunătățire majoră în timpul mediu petrecut de o mașină, dar și în numărul de mașini care trec prin intersecții. Rezultatele finale ale optimizărilor sunt următoarele:



5. Concluzii

După efectuarea experimentelor, se poate observa că eficiența traficului este în directă legătură cu timpul culorii verzi a semafoarelor. În funcție de acești timpi, intersecțiile pot fi mai eficiente sau mai puțin eficiente. Pentru a ajunge la cele mai bune numere pentru timpul culorii verzi a semafoarelor, experimentul de optimizare trebuie rulat de mai multe ori aducând modificări în valorile de step sau valorile minime/maxime.

Cu valorile implicite, acest model nu rulează la capacitate maximă, iar în timpul simulării se pot observa mari aglomerații, care încep într-o intersecție, iar apoi cauzează aglomerație și în celelalte. Timpul mediu petrecut de o mașină în intersecții este de 510 de secunde, acest număr fiind destul de mare. Numărul de mașini care trec prin intersecții este de aproximativ 1100 de mașini.

Cu valorile obținute după multiple procese de optimizare, încă se mai pot observa aglomerații în intersecții, însă nu atât de multe și dese. Timpul mediu petrecut de o mașină în intersecții este de aproximativ 370 de secunde, o îmbunătățire majoră față de valorile implicite. Numărul de mașini care trec prin intersecții a crescut și acesta, fiind de aproximativ 1500 de mașini.

Cu valorile obținute în urma schimbării fazelor semafoarelor și totodată a noilor timpi conform cu experimentele de optimizare realizate asupra acestor timpi, încă se pot observa aglomerații, însă intersecțiile nu se blochează aproape deloc, iar blocajele care se întâmplă țin foarte puțin. Timpul mediu petrecut de o mașină în intersecții este de aproximativ 240 de secunde, asta fiind mai puțin de jumătate din timpul pe care îl petrece o mașină într-o intersecție cu valorile implicite. Numărul de mașini care trec prin intersecții este și el dublat față de intersecțiile cu valori implicite.

În concluzie, cea mai bună metodă de a scădea aglomerația intersecțiilor și de a reduce semnificativ numărul blocajelor în intersecții este de a folosi un număr diferit de secunde pentru culoarea verde a fiecărui semafor în parte, în funcție de intersecțiile în care se formează aglomerații cel mai des.

Bibliografie

AnyLogic 8 in Three Days by Ilya Grigoryev

https://en.wikipedia.org/wiki/Discrete-event_simulation

<https://www.bmt.org/industries/defence-and-security/discrete-event-simulation/>

<https://www.sciencedirect.com/topics/engineering/discrete-event-simulation>

<https://www.anylogic.com/use-of-simulation/discrete-event-simulation/>