

Laboratory 6 MS

Problem 1:

1. Să se modeleze o linie de asamblare a automobilelor ce constau din următoarele subansamblu: șasiu, motor și transmisie. Fiecare subansamblu este o entitate generată de un bloc Entity Generator la intervale de timp t_{gen} . După generare, subansamblele sunt testate în stații un interval de timp t_{test} . Asamblarea subansamblelor durează $t_{ans} = 0.3$ h. Unitatea de timp a simulării este ora.

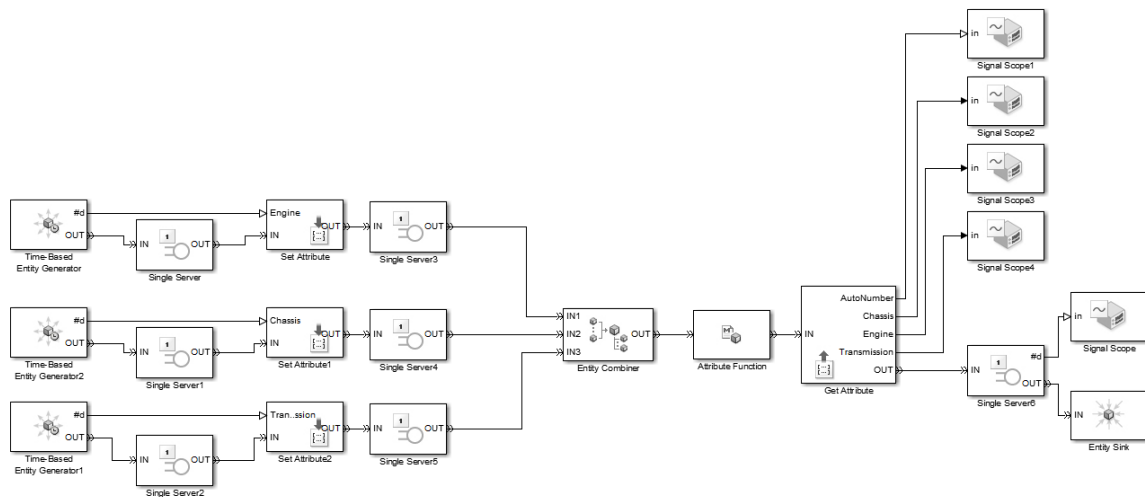
Subansamblu	t_{gen} [h]	t_{test} [h]
Motor	0.8	0.6
Transmisie	0.6	0.4
Sasiu	0.4	0.4

Tabelul 2. Date pentru linia de asamblare

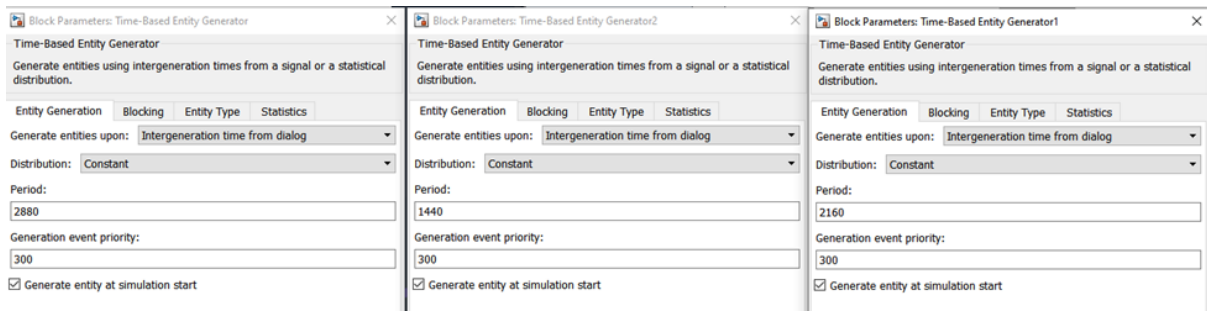
Se vor înregistra numărul de subansamblu generate, șasiu, motor, transmisie și numărul de automobile asamblate. Să se simuleze modelul pe durata de 200 ore. Numărul asociat șasiului se va calcula cu formula:

$$out_Auto_Number = Motor + 10 * Transmission + 100 * Chassis;$$

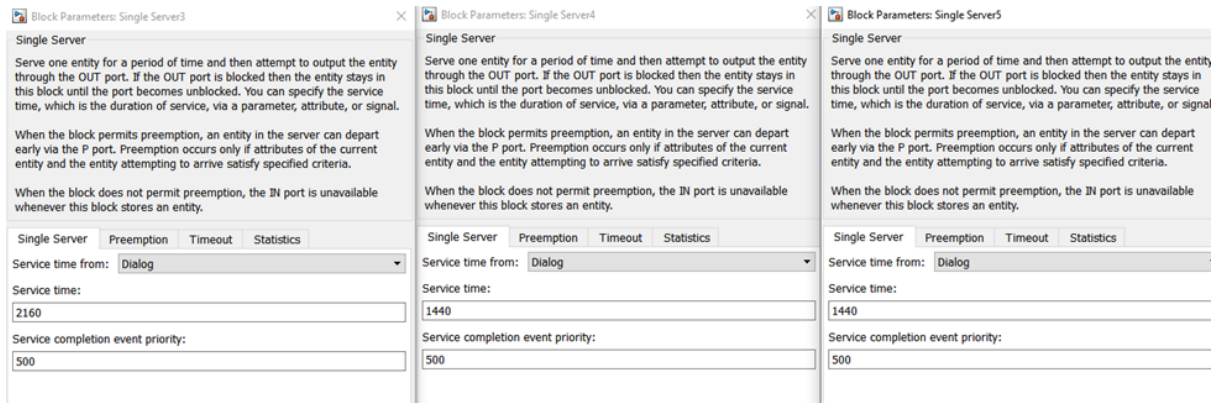
We designed the following schema:



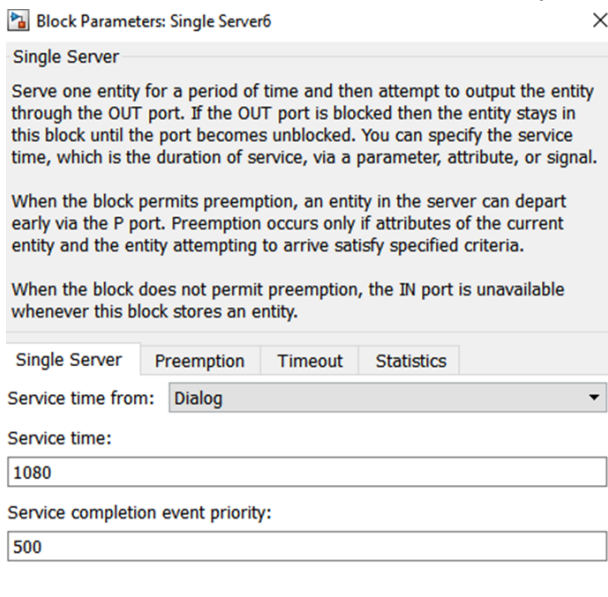
We used the following generators



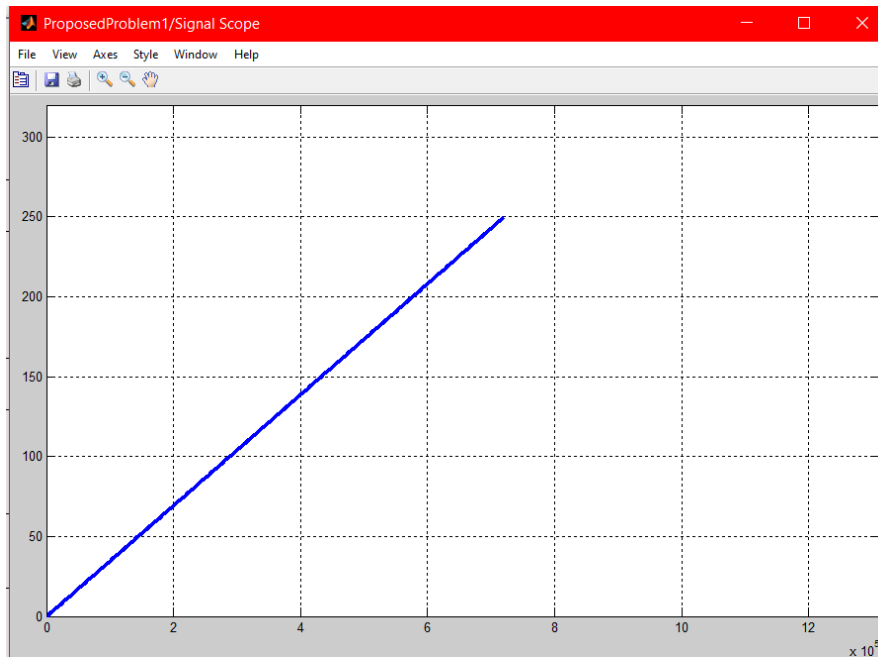
Every element is tested according to the specified time:



The last server takes care of the assembly part:



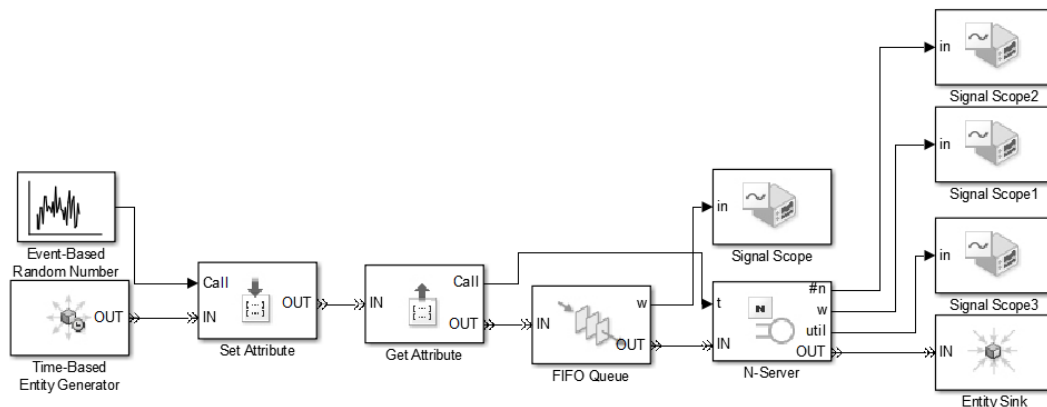
We get the following result:



Problem 2:

- La o companie telefonică există un serviciu de informații cu cinci operatori. Orice operator poate prelua orice cerere. Cererile apar la intervale cu distribuție exponențială cu valoarea medie 0.2 minute. Durata unei convorbiri este 1 ± 0.2 minute cu distribuție uniformă. Se va utiliza un multiserver cu capacitatea cinci. Să se simuleze sistemul pe o durată de 8 ore. Se vor afișa timpul mediu de așteptare în coadă, timpul mediu de deservire, gradul mediu de utilizare a serverului și numărul de entități din server.

We designed the following schema:



We used the following call generator:

Block Parameters: Time-Based Entity Generator

Time-Based Entity Generator

Generate entities using intergeneration times from a signal or a statistical distribution.

Entity Generation Blocking Entity Type Statistics

Generate entities upon: Intergeneration time from dialog

Distribution: Exponential

Initial seed:
12345

Mean:
12

Generation event priority:
300

☒ Generate entity at simulation start

We set the call duration with a random number generator:

Source Block Parameters: Event-Based Random Number

Event-Based Random Number

Generate random numbers from the specified distribution, parameters, and initial seed.

Parameters

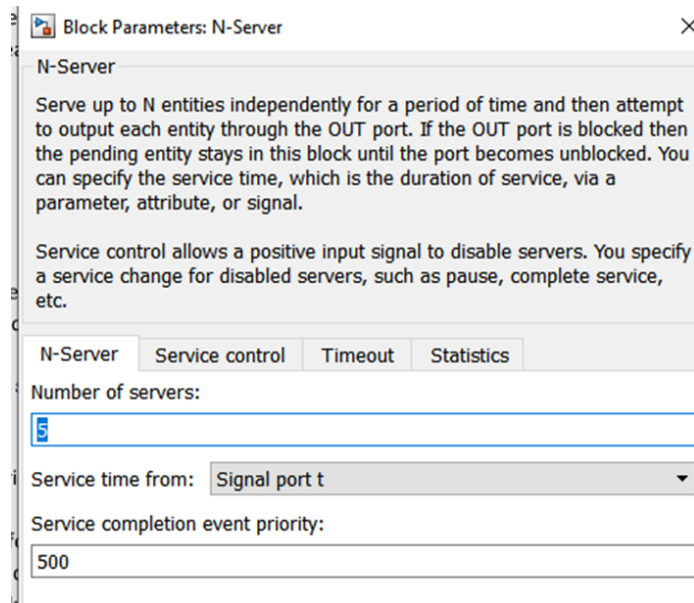
Distribution: Uniform

Minimum:
48

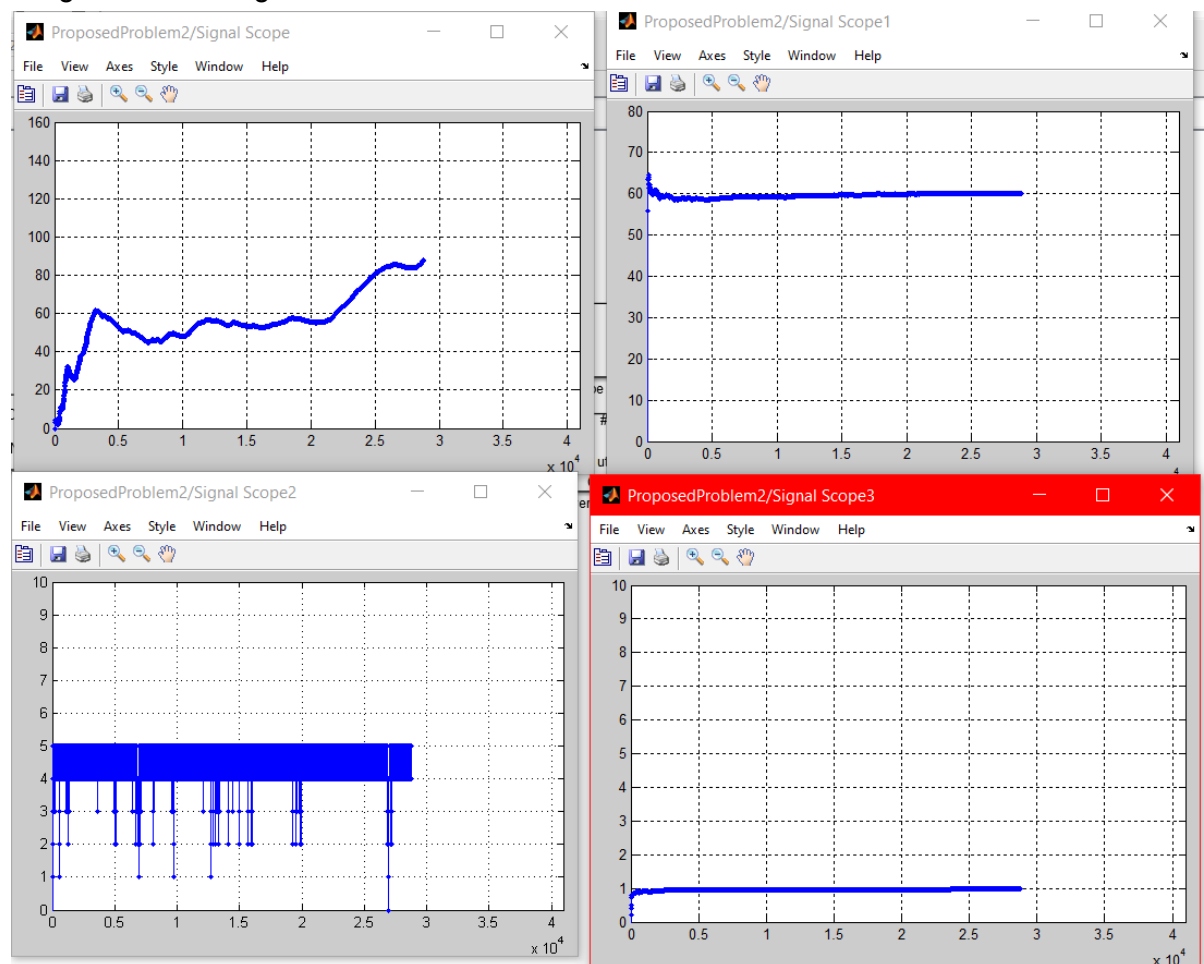
Maximum:
72

Initial seed:
12346

We also have the following multiserver:



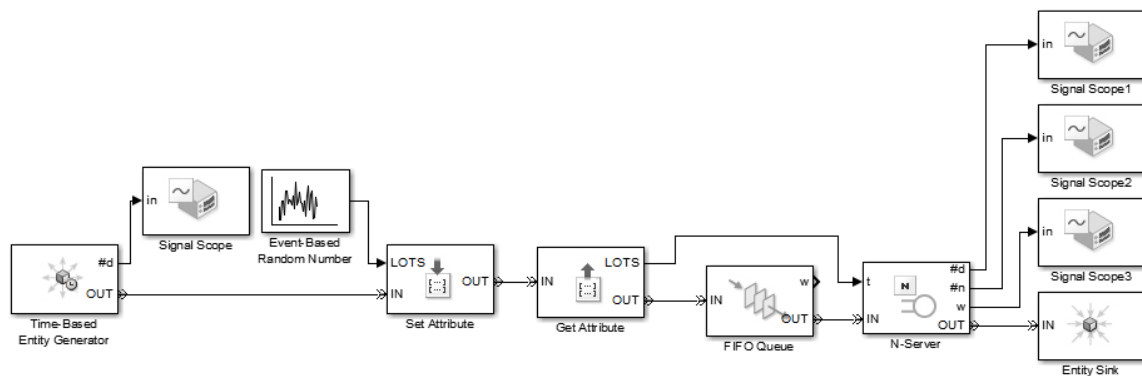
We get the following result:



Problem 3:

3. La o mașină unealtă sosesc loturi de piese pentru prelucrare. Loturile sosesc la intervale de $18 \pm 6s$ cu distribuție uniformă. Prelucrarea unui lot durează $32 \pm 4s$ cu distribuție uniformă. Loturile sunt prelucrate la o stație cu capacitatea 2. Să se simuleze modelul pe o durată de o oră. Se vor afișa numărul de loturi generate, timpul mediu de așteptare în coadă, timpul mediu de deservire, numărul de entități în server și numărul de loturi prelucrate. Pentru stația de prelucrare se va utiliza un multiserver cu capacitatea doi.

We designed the following schema:



We used the following entity generator:

Time-Based Entity Generator

Generate entities using intergeneration times from a signal or a statistical distribution.

Entity Generation Blocking Entity Type Statistics

Generate entities upon: Intergeneration time from dialog

Distribution: Uniform

Initial seed:
12367

Minimum:
12

Maximum:
24

Generation event priority:
300

The work time is determined with a random number generator:

Source Block Parameters: Event-Based Random Number

Event-Based Random Number

Generate random numbers from the specified distribution, parameters, and initial seed.

Parameters

Distribution: Uniform

Minimum:
28

Maximum:
36

Initial seed:
12332

The second workstation has a capacity of two:

Block Parameters: N-Server

N-Server

Serve up to N entities independently for a period of time and then attempt to output each entity through the OUT port. If the OUT port is blocked then the pending entity stays in this block until the port becomes unblocked. You can specify the service time, which is the duration of service, via a parameter, attribute, or signal.

Service control allows a positive input signal to disable servers. You specify a service change for disabled servers, such as pause, complete service, etc.

N-Server

Service control

Timeout

Statistics

Number of servers:

2

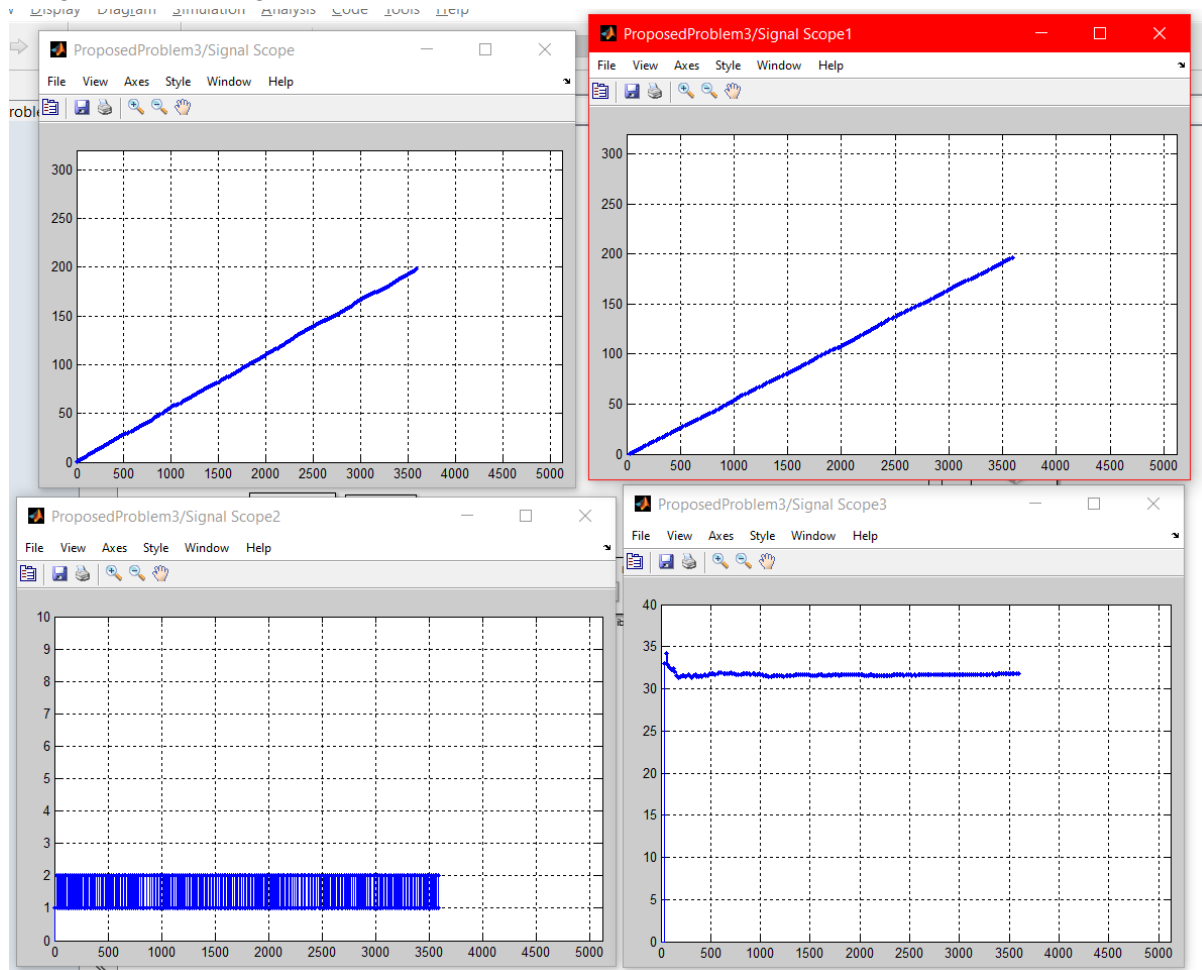
Service time from:

Signal port t

Service completion event priority:

500

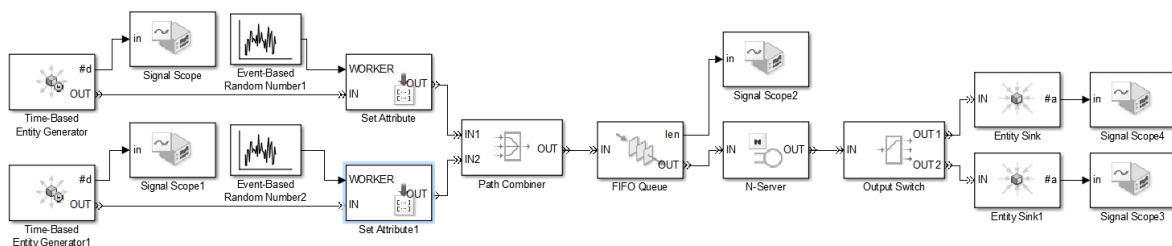
We got the following result:



Problem 4:

4. La o magazie pentru aprovizionarea cu materiale vin două tipuri de muncitori. Cei de primul tip vin la intervale de $42 \pm 3s$ cu distribuție uniformă, iar timpul de deservire este $30 \pm 9s$ cu distribuție uniformă. Muncitorii de tipul doi la intervale de $50 \pm 10s$ cu distribuție uniformă, iar timpul de deservire este $12 \pm 3s$ cu distribuție uniformă. Muncitorii de ambele tipuri așteaptă în aceeași coadă să fie deserviți. Să se simuleze modelul pe o durată de o oră. Se vor afișa numărul de muncitori de fiecare tip care vin pentru aprovizionare, lungimea cozii și numărul de muncitori deserviți de fiecare tip.

We designed the following schema:



First type of worker:

Block Parameters: Time-Based Entity Generator

Time-Based Entity Generator

Generate entities using intergeneration times from a signal or a statistical distribution.

Entity Generation Blocking Entity Type Statistics

Generate entities upon: Intergeneration time from dialog

Distribution: Uniform

Initial seed:

12343

Minimum:

39

Maximum:

45

Generation event priority:

300

☒ Generate entity at simulation start

Source Block Parameters: Event-Based Random Number1

Event-Based Random Number

Generate random numbers from the specified distribution, parameters, and initial seed.

Parameters

Distribution: Uniform

Minimum: 21

Maximum: 39

Initial seed: 12321

Second type of worker:

Block Parameters: Time-Based Entity Generator1

Time-Based Entity Generator

Generate entities using intergeneration times from a signal or a statistical distribution.

Entity Generation Blocking Entity Type Statistics

Generate entities upon: Intergeneration time from dialog

Distribution: Uniform

Initial seed: 12341

Minimum: 40

Maximum: 60

Generation event priority: 300

☒ Generate entity at simulation start

Source Block Parameters: Event-Based Random Number2

Event-Based Random Number

Generate random numbers from the specified distribution, parameters, and initial seed.

Parameters

Distribution: Uniform

Minimum: 9

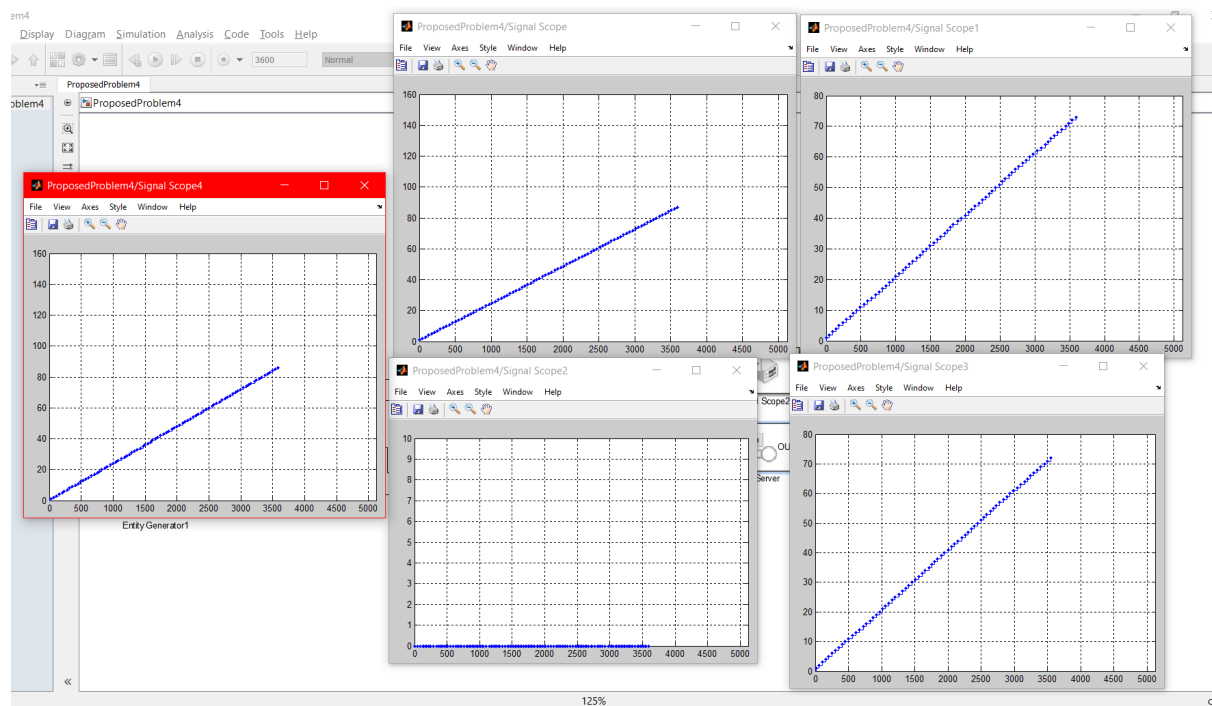
Maximum: 15

Initial seed: 12312

We have configured an n-server:

N-Server	Service control	Timeout	Statistics
Number of servers:			
<input type="text" value="2"/>			
Service time from: Attribute			
Attribute name:			
<input type="text" value="WORKER"/>			
Service completion event priority:			
<input type="text" value="500"/>			

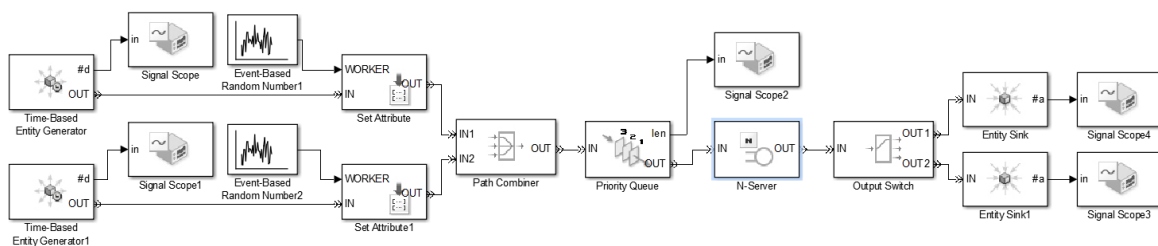
We get the following result:



Problem 5:

- Se va rezolva problema de mai sus când muncitorii de tipul doi sunt deserviți cu prioritate față de cei de tipul unu.

We only changed the normal queue with a priority queue



Block Parameters: Priority Queue

Priority Queue

Store entities in sorted sequence for an undetermined length of time. The Capacity parameter is the number of entities the queue can hold. The queue sorts entities according to the values of the specified attribute, in either ascending or descending order.

Priority Queue Timeout Statistics

Capacity:

Sorting attribute name:

Sorting direction:

The result is the following:

