

Aalto Yliopisto

SCI-C0200 - Fysiikan ja matematiikan menetelmien studio

## Tietokoneharjoitus 12: Pysäköintipaikan simulointi (Matlab)

Timi Turpeinen, 100740833, timi.turpeinen@aalto.fi  
Jonni Järvinen, 1027817, jonni.jarvinen@aalto.fi

Palautettu: 13. joulukuuta 2023

# Sisällys

<b>1</b>	<b>Johdanto:</b>	<b>3</b>
<b>2</b>	<b>A: Pysäköintipaikan simulointi</b>	<b>5</b>
<b>3</b>	<b>B: Lappuliisa pysäköintipaikalla</b>	<b>9</b>
<b>4</b>	<b>Lähdekoodit</b>	<b>15</b>
4.1	painokerroinfunktio . . . . .	15
4.2	A simulointi . . . . .	15
4.3	B simulointi . . . . .	17
4.3.1	Liisalappu funtkio . . . . .	17
4.3.2	Parkkipaikka funktio . . . . .	19
4.3.3	Liisalappu käynnin simulointi . . . . .	20
4.3.4	Kuukausi funktio . . . . .	22
4.3.5	Parkkikk funktio . . . . .	23
4.3.6	Liisalapun käynti simulointi kuukausittain, vuoden ajan	23

# 1 Johdanto:

Tarkastellaan Teknillisen korkeakoulun pysäköintipaikalle saapuvia autoja, kun tiedämme, että saapuvien autojen lukumäärä tunnin välein on satunnaista. Täten ottamalla havaintoja pitkältä aikaväliltä saapuvien autojen lukumääristä pysäköintipaikalle havaitsemme, että saapuvien autojen lukumäärä on normaalijakautunut tietyillä odotusarvoilla ja keskihajonnalla riippuen kellonajasta. Kello 7 aikaan aamulla esimerkiksi saapuvia autoja tulee odotetusti enemmän kuin myöhään illalla (esimerkiksi kello 21). Näiden havaintojen pohjalta saamme taulukon, jossa näkyy saapuvien autojen lukumäärien odotusarvot ja keskihajonnat riippuen vuorokaudenajasta seuraavasti:

<i>kellonaika</i>	$\lambda_i$	$\sigma_{\lambda_i}$
01	3	3
02	3	3
03	3	3
04	3	2
05	3	2
06	3	2
07	10	3
08	20	5
09	60	7
10	40	10
11	30	10
12	20	10
13	40	7
14	20	5
15	30	3
16	14	2
17	13	2
18	15	3
19	15	3
20	10	4
21	8	34
22	3	2
23	3	2
24	3	2

, jossa  $\lambda_i$  on tiettyä kellonaikana  $i$  saapuvien autojen lukumäärän normaalijakauman odotusarvo ja  $\sigma_{\lambda_i}$  vastaavasti tämän ajan keskihajonta. Saapuvien autojen lukumäärät ovat siis havaintojen perusteella normaalijakautuneet. Taulukosta havaitsemme, että Teknillisen korkeakoulun pysäköintipaikalle aamupäivällä on eniten tulijoita, kun teekkarit pyrkivät pääsemään luennoille.

Tämä Teknillisen korkeakoulun pysäköintipaikan simulointi voi meitä kiinnostaa, jotta voimme parantaa liikkuvutta ja esteettömyyttä korkeakouluun matkaamiselle ja siten optimoida sekä tehdä johtopäätöksiä, jotka parantavat pysäköintipaikan käytettävyyttä.

Havaintojen perusteella tiedämme myös, että päivän aikana parkkipaikalta autoja poistuu tietyn aikavälein. Autot poistuvat pysäköintipaikalta säännöllisen ajan kuluttua, eli autojen poistuminen riippuu aikaisempina hetkinä saapuneiden autojen lukumäärästä. Lisäksi autojen poistuminen pysäköintipaikalta on jokseenkin satunnaista, sillä pysäköintipaikalle tullut auto voi lähteä odotettua aikaisemmin tai myöhemmin.

Täten poistuvien autojen lukumäärää tietyllä ajanhetkellä  $t$ , joka olkoon nyt jokin vuorokauden kellonaika, kuvaa yhtälö, joka on muotoa:

$$\mu_t = (\sum_{i=1}^k p(i) \cdot l(t + i - k)) + e(t) \quad (1)$$

, jossa  $\mu_t$  on siis poistuvien autojen lukumäärä ajanhetkellä  $t$  ja  $p(i)$  on painokerroin,  $l(t + i - k)$  on saapuneiden autojen lukumäärä ajanhetkellä  $t + i - k$  ja  $e(t)$  on parametrein  $(0, \sigma_\mu^2)$ -jakautunut normaalijakauma, eli odostusarvo on 0 ja keskihajonta  $\sigma_\mu$ . Kyseessä siis ei ole (vaikka merkintä voisi näin indikoida) eksopenttijakauma, vaan normaalijaukama. Poistuvien autojen lukumäärää kuvaavassa kaavassa (1) on siis otettu huomioon poistuvien autojen satunnaisuus, että pysäköintipaikalla oleva säännölliset ajat, jonka takia lähtevien autojen lukumäärä riippuu aikasemmin tulneiden autojen lukumäärästä.

## 2 A: Pysäköintipaikan simulointi

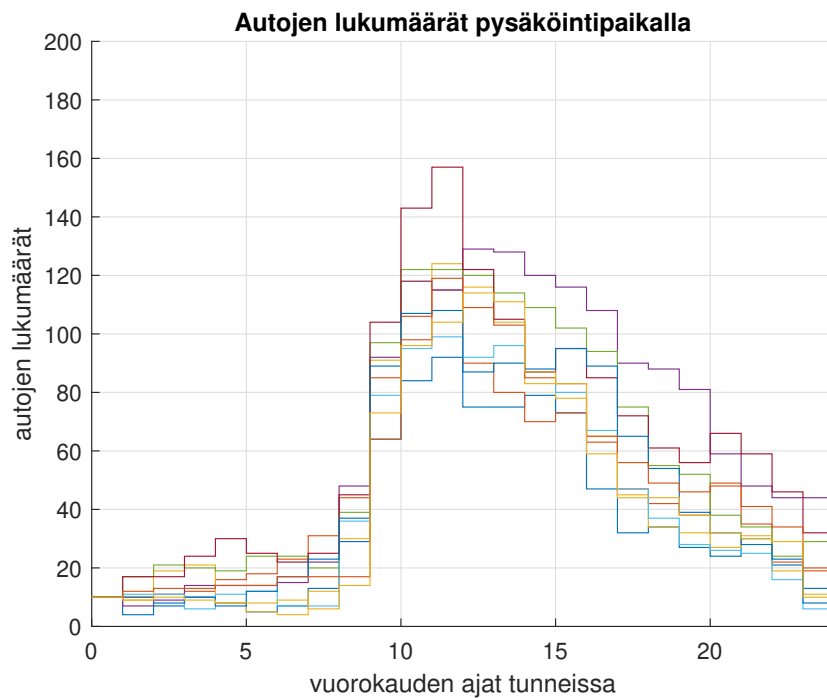
Simuloidaan nyt pysäköintipaikalla olevien, tulevien ja sieltä poistuvien autojen lukumääriä, johdannosta saaduilla tiedoilla ja havainnoilla jotka olemme saaneet pysäköintipaikan toiminnasta. Tehdään simulointia varten matemaattinen malli, jossa olemme huomioineet jokaisen ilmiön satunnaisuuden ja säännönmukaisuuden.

Oletetaan, että poistuvien autojen lukumäärissä  $\mu_t$  oleva viivetermi  $k$  on suuruudeltaan  $k = 5$ , joka on tehtävänannon perusteella aikayksikkö. Oletamme siis, että vasta vuorokauden aikana kello viiden jälkeen alkaa pysäköintipaikalta autoja poistumaan säännöllisesti riippuen, milloin ne ovat saapuneet ja että parkkipaikalle saapuneet autot lähtevät suunnilleen viiden tunnin kuluttua parkkipaikalta, mutta osa voi havaintojen perusteella lähteä aikaisemminkin kaavan (1) perusteella. Olkoon myös tässä poistuvien autojen  $\mu_t$  lausekkeessa painokerroin  $p(i)$  muotoa  $p(i) = \frac{0.7^i}{norm}$ , jossa  $norm = \sum_{j=1}^k p(j)$  eli parametri  $norm$  kuvaa, että haluamme normittaa painokertoimen, niin että vuorokauden aikana olevien painokertoimien summa olisi yksi. Oletetaan, että myös lähtevien autojen virhetermin varianssi olisi  $\sigma_\mu^2 = 5^2$  ja siten keskihajonta olisi  $\sigma_\mu = 5$  eli vakio kaikilla ajanhetkillä. Pysäköintipaikalle mahtuu rajallinen määrä autoja, joten oletetaan, että autopaikkoja on 200. Merkitään tätä simuloinnissamme parametrilla  $autot_{max} = 200$ . Lisäksi oletetaan vielä että vuorokauden alussa (eli keskiyöstä lähtien) parkkipaikalla on jo 10 autoa (joka ei ole todennäköistä, mutta mahdollista).

Simuloinnissa nyt siis meidän täytyy generoida havaituille satunnaismuuttujille, eli saapuville autoille ja poistuville autoille olevat satunnaisuudet erikseen joka kerta eri vuorokauden ajanhetkillä tietyillä säännöillä. Tiedämme, että saapuvien autojen lukumäärä ei voi ylittää pysäköintipaikan kapasiteettia eli arvoa  $autot_{max}$ , joten tämä on eräs rajoite, joka simuloinnissa täytyy ottaa huomioon. Myös poistuvien autojen lukumäärä ei saa olla suurempi kuin pysäköintipaikalla olevien autojen lukumäärä, sillä pysäköintipaikalta ei voi poistua sitä enempää autoja. Tämä on taas toinen rajoite pysäköintipaikan simuloinnissa eli niin sanottu "alarajoite". Lisäksi simuloinnissa otetaan huomioon normaali jakaumien satunnaislukujen generoinnissa, että generoitut luvut voivat vain olla ei-negatiivisia kokonaislukuja  $(0, 1, 2, 3, \dots)$ , sillä generoimme kokonaisia autojen lukumääriä. Lisäksi generoinnissa olemme valinneet, että autot ovat ei negatiivisia lukuja, eli tällöin  $\mu_t \geq 0$  ja  $arrive \geq 0$  lähdekoodin 4.2 mukaisesti. Tämä johtuu siitä, ettei poistuvia autoja voi olla esimerkiksi negatiivinen määrä. Sama pätee saapuville autoille.

Simuloinnissa lasketaan jokaisena vuorokauden kellonaikana generoidun saapuvien autojen lukumäärän, josta vähennetään generoitu poistuvien autojen

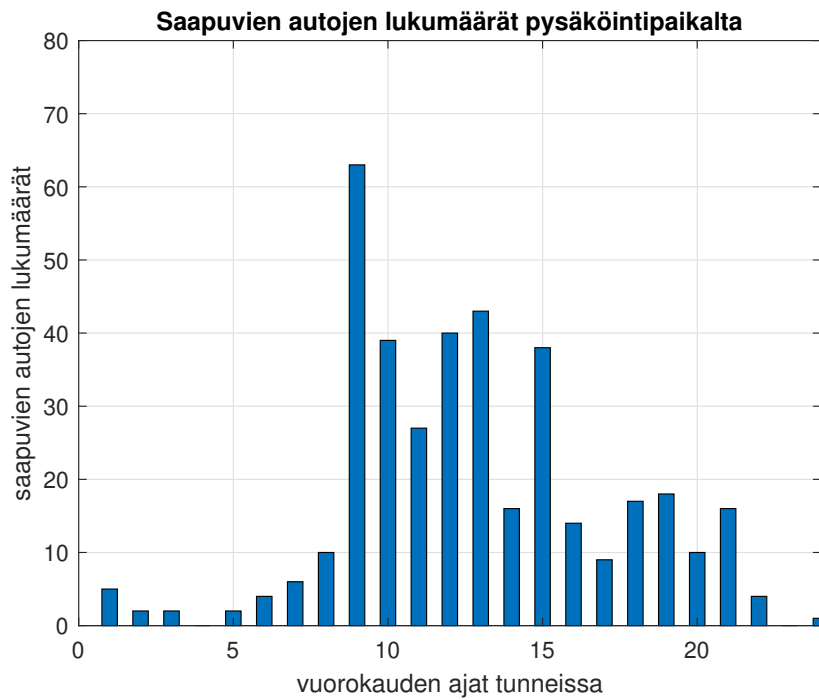
lukumäärä. Tällöin saamme lukumäärän  $a$ , joka kuvaa saapuvien ja lähtevien autojen erotusta tiettyinä kellonaikoina. Tällöin vuorokaudessa seuraavana kellonaikana pysäköintipaikalla olevien autojen lukumäärä on rekursiivisesti aikaisempina kellonaikoina olevien autojen lukumäärän ja lukumäärän  $a$  summa. Täten saamme simuloitua lähdekoodin 4.2 mukaisesti 10 eri variaatiota vuorokaudessa pysäköintipaikalla olevien autojen lukumääristä eri kellonaikoina. Muodostetaan simuloinnista saaduista tuloksista kuvaaja, jossa näkyy autojen lukumäärä pysäköintipaikalla eri vuorokauden kellonaikoina.



Kuva 1: Autojen lukumäärät pysäköintipaikalla vuorokauden aikana

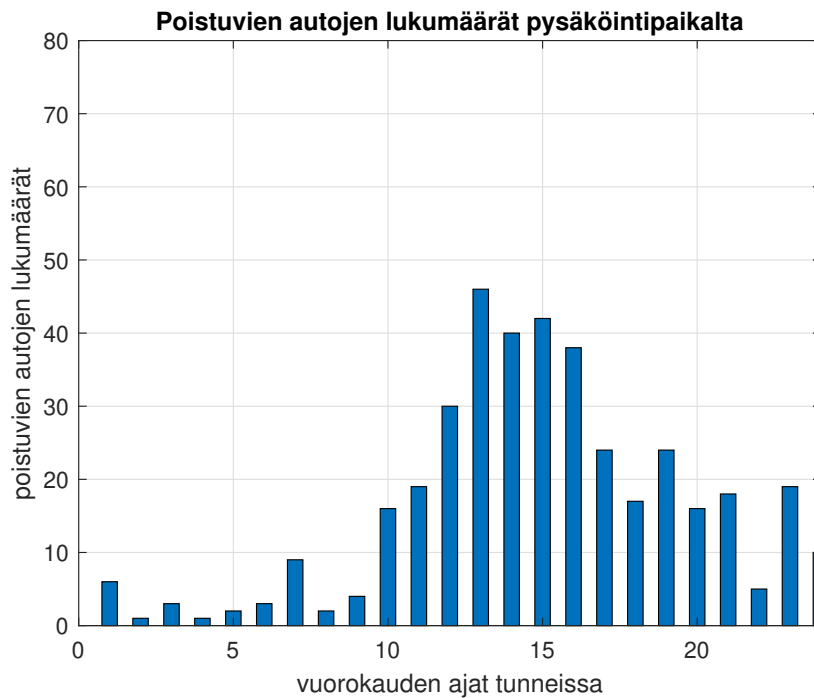
Kuvassa 1 on 10 eri variaatiota siitä, miten vuorokauden aikana Tekniikan korkeakoulun pysäköintipaikalla autojen lukumäärät kehittyvät riippuen kellonajasta. Kuvasta havaitsemme, että aamupäivällä kello 8-10 välissä autojen lukumäärä kasvaa eniten ja kello 11-12 välissä autoja pysäköintipaikalla on eniten. Tämän jälkeen kello 13-24 välillä autojen määrä vähenee tasaisesti, jolloin vuorokauden lopussa autojen lukumäärä on välillä  $[0, 50]$ .

Tehdään vielä kuvaajat, joissa näkyy yhden simulointikerran aikana saapuvien ja poistuvien autojen lukumäärät, jotta voimme vahvistaa simuloinnista saadut tulokset.



Kuva 2: Saapuvien autojen lukumäärät vuorokauden aikana

Kuvassa 2 näemme yhden mahdollisen lopputuloksen saapuvien autojen lukumäärästä vuorokauden aikana. Huomaamme kuvaajasta, että suurin osa autoista tulee klo 8-10 välillä, kuten monen variaation simuloinnista havaitsimme. Vastaavasti autoja tulee vähemmän vuorokauden lopussa ja alussa pysäköintipaikalle, joka käy järkeen pysäköintipaikkojen simuloinnissa.



Kuva 3: Poistuvien autojen lukumäärät vuorokauden aikana

Kuvasta 3 näemme yhden mahdollisen lopputuloksen poistuvien autojen lukumäärästä vuorokauden aikana. Kuvaajasta havaitsemme, että vuorokauden alussa autoja ei poistunut melkein ollenkaan pysäköintipaikalta, kuten olemme havainneet. Lopulta autoja alkaa poistua pysäköintipaikalta klo 10 jälkeen runsaan määrin, mikä havaittiin myös kuvassa 1 pysäköintipaikalla vähenevien autojen lukumäärissä.

Simulaation tulokset vaikuttavat järkeviltä. Yleisesti arkipäivänä Teknillisen korkeakoulun pysäköintipaikalla olevien autojen lukumäärä yöllä eli noin kello 0-5 ei yleisesti kasva, sillä ihmiset vielä nukkuvat tähän aikaan. Aamulla alkaa taas autojen määrä pysäköintipaikalla kasvamaan, minkä havaitsemme etenkin klo 8-10 välillä suurena saapuvien autojen lukumääränä. Päiväsaikaan pysäköintipaikalla on eniten autoja, kun taas iltapäivällä autojen määrä pysäköintipaikalla alkaa vähentyä työ-/koulupäivän loputtua.

Havaitsemme kuitenkin, ettei simulaatio täysin realistisesti kuvaa myöskään todellisen pysäköintipaikan autojen lukumäärä vuorokauden aikana. Esimerkiksi yöllä pysäköintipaikalle saapuu jonkin verran autoja, mikä on hyvin epärealistista, mutta seuraa havainnoista saatuja tuloksia, jotka mainittiin johdannossa. Lisäksi päivän päätteeksi autoja jää joissain variaatiossa pysäköintipaikalle hyvin paljon, mitä ei yleensä tapahdu. Huomaamme myös,



että simulointi vastaa luultavasti enemmän arkipäivänä autojen lukumääriä pysäköintipaikalla kuin viikonloppuna.

### 3 B: Lappuliisa pysäköintipaikalla

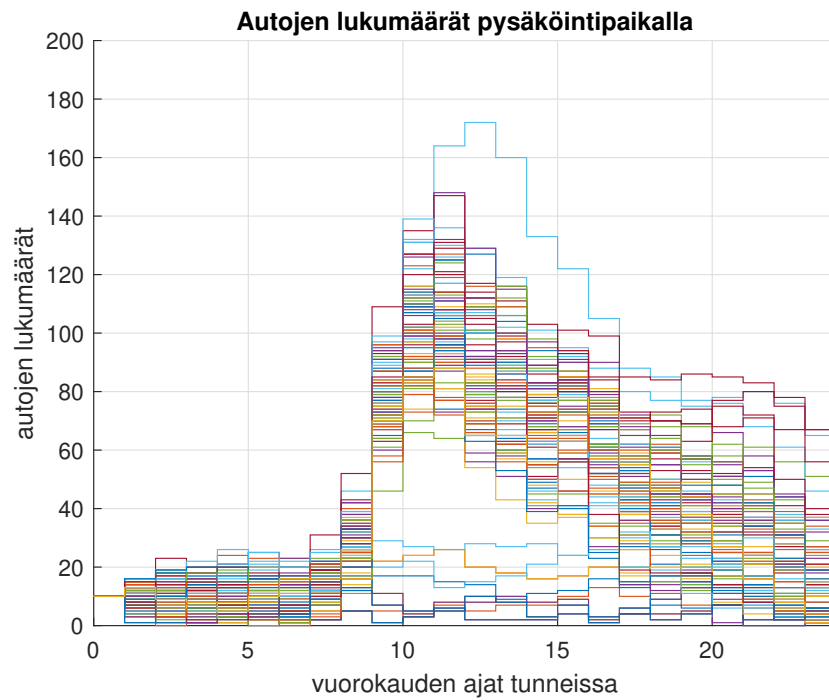
Tiedämme, että Teknillisen korkeakoulun pysäköintipaikalle voi tulla pysäköinninvalvoja silloin tällöin sakottamaan autoja, joilla ei ole virallista pysäköintioikeutta.

Pitkän aikavälin havaintojen jälkeen on havaittu, että pysäköinninvalvojan käyntiä seuraavana kahtena päivänä teekkarit karttavat pysäköintipaikkaa, jolloin saapuvien autojen lukumäärät ovat normaalijaukautuneet odotusarvolla  $\lambda_i = 3$  ja keskihajonnalla  $\sigma_{\lambda_i} = 2$  koko vuorokauden ajan. Tällöin siis lappuliisan käynti vaikuttaa suoranaisesti käyvien autojen lukumäärään pysäköintipaikalla.

Simuloidaan siis seuraavaksi, miltä näyttäisi jos lappuliisa käynti on satunnaista ja hän käy keskimäärin kerran kuukaudessa sakottamassa pysäköintipaikoilla olevia autoja. Tällöin siis käynnin satunnaisuus olisi tasajakautunut satunnaismuuttuja kuukauden päiviin verrattuna. Olkoon simuloinnissamme kuukaudessa päiviä 30. Lisäksi täytyy huomioda, että koska lappuliisa käy vain keskimäärin kerran kuukaudessa pysäköintipaikalla, voi hän käydä siis yhden kerran, useammin tai ei ollenkaan kuukaudessa. Täten siis jokaisena päivänä käynnin todennäköisyys on  $\frac{1}{30}$ . Generoidaan siis lappuliisan käyntiä joka päivä uudestaan, jolloin tämäkin satunnaisuus toteutuu.

Oletetaan myös, että tälläkin kertaa jokaisen päivän alussa pysäköintipaikalla olevien autojen lukumäärä on 10 autoa, ja oletetaan myös, että pysäköintipaikalla saapuvien ja poistuvien autojen lukumäärät ovat yhtä suuret viikonloppuna ja arkena. Tällöin saamme simulaation, joka toimii lähdekoodien 4.3.1 ja 4.3.2 sekä 4.3.3 mukaisesti.

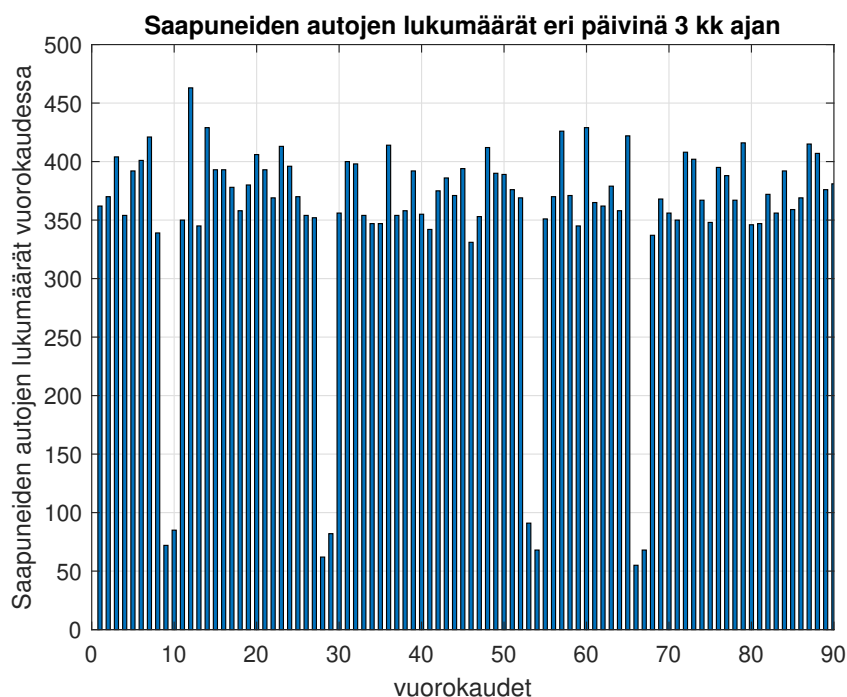
Havainnollistetaan nyt simulaatiomme tuloksia siitä, miten pysäköinninvalvojan käynti näkyy autojen määrässä 3 kuukauden aikana eri päivinä, ja miten autojen määrä kehittyy kyseisten päivien aikana kellonajan mukaisesti.



Kuva 4: Autojen käynti vuorokauden eri kellon aikoina, kolmen kuukauden päivien ajan

Kuvassa 4 on kuvaaja pysäköintipaikalla olevien autojen lukumäärän kehityksestä vuorokausien eri kellonaikoina kolmen kuukauden ajalta. Havaitsemme kuvaajasta, että lappuliisa on käynyt kolmen kuukauden aikana neljä kertaa, sillä kahdeksan päivän ajan pysäköintipaikalla käyvien autojen lukumäärä on selvästi ollut muita päiviä alhaisempi.

Tehdään vielä havainnollistavampi kuva kyseisestä simulaatiosta, jossa kuvaamme eri päivinä käyvien autojen lukumäärää kolmen kuukauden ajalta.



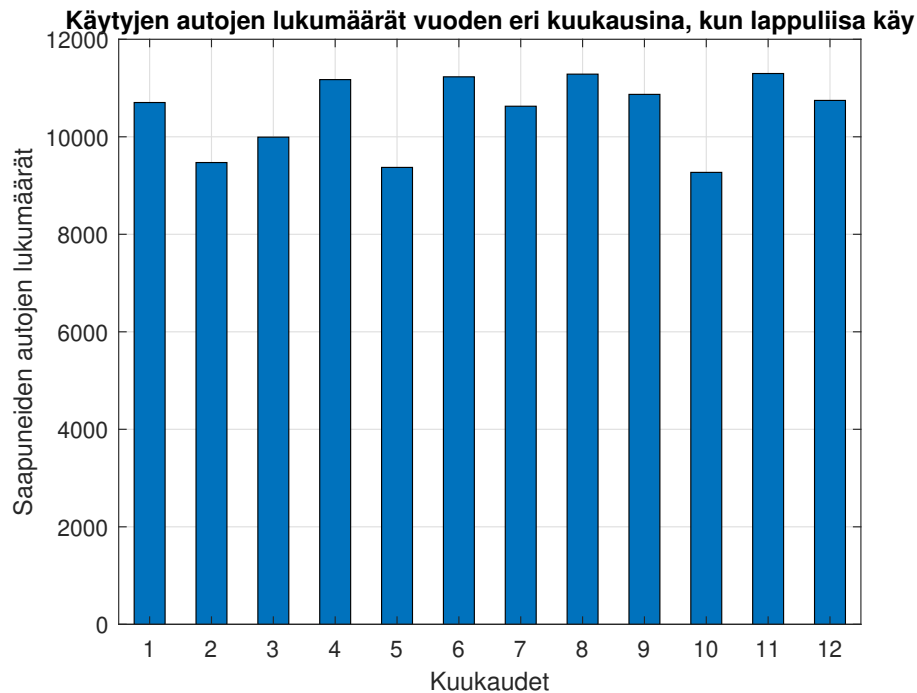
Kuva 5: Käytyjen autojen lukumäärät pysäköintipaikalla kolmen kuukauden ajalta

Kuvasta 5 havaitsemme jälleen, että lappuliisa on kyseisellä simulointikeralla käynyt taas neljästi kolmen kuukauden aikana sakottamassa tekkarien autoja pysäköintipaikalla. Keskimäärin siis lappuliisa on käynyt noin kerran kuukaudessa. Tämän havaitsemme siitä, että ensimmäisen kuukauden päivinä 9, 10, 28 ja 29, toisen kuukauden päivinä 23 ja 24 sekä kolmannen kuukauden 6 ja 7 päivinä pysäköintipaikalla käyvien autojen lukumäärä on paljon alhaisempi kuin muina kuukauden päivinä. Tämän perusteella voimme päätellä, että lappuliisa kävi pysäköintipaikalla ensimmäisen kuukauden päivinä 8. ja 27., toisen kuukauden 22. päivänä sekä kolmannen kuukauden 5. päivänä sakottamassa.

Kuvasta 5 havaitsemme myös, että kun lappuliisa on käynyt sakottamassa autoja, on seuraavien kahden päivän ajan pysäköintipaikalla käyneiden autojen lukumäärä keskimäärin noin 60 autoa. Tämä tarkoittaa, että keskimäärin pysäköintipaikalla käy päivittäin noin 300 autoa normaalipäivää vähemmän, joka on huomattava pudotus kävijämäärissä. Tämän perusteella pysäköintipaikkaa eniten käyttävä kohderyhmä on ilman pysäköintilupaa olevat, eli tekkarit.

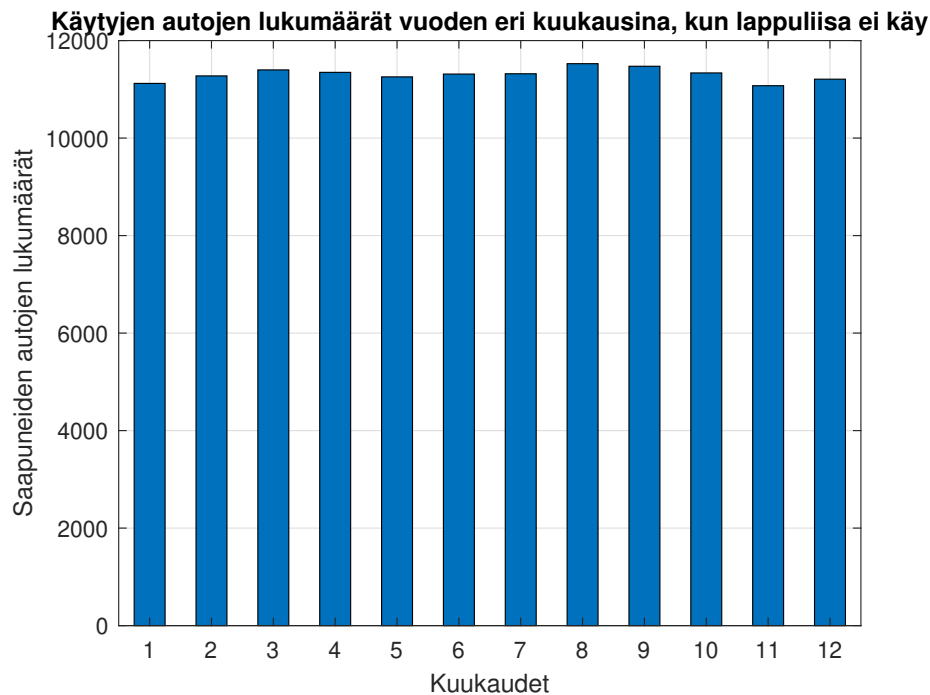
Simuloidaan vielä pidemmällä aikavälillä, eli noin vuoden ajan pysäköinti-

paikalla käyvien autojen lukumääriä kuukausien mukaisesti ja verrataan sitä autojen lukumääriin kuukausien ajan, jos lappuliisa ei kävisi pysäköintipaikalla. Saamme tällöin aikaisempien ehtojen mukaisesti simuloitua pysäköintipaikalla käyvien autojen lukumääriä lähdekoodien 4.3.4 ja 4.3.5 sekä 4.3.6 mukaisesti. Tehdään havainnollistavat kuvat simulointimme tuloksesta.



Kuva 6: Pysäköintipaikalla käyvien autojen lukumäärät kuukausien aikana, kun lappuliisa on käynyt

Kuvassa 6 näemme pysäköintipaikalla käyneiden autojen lukumääriä vuoden eri kuukausien aikana, kun lappuliisa on käynyt satunnaisesti eri aikoina sakottamassa pysäköintiluvottomia autoja. Keskimäärin kuitenkin lappuliisa on käynyt kerran kuukaudessa, mikä havaitaan tiettyjen kuukausien aikana pysäköintipaikalla käyvien autojen määrän laskusta.



Kuva 7: Pysäköintipaikalla käyneiden autojen lukumäärät kuukausien aikana, kun lappuliisa ei ole käynyt

Kuvasta 7 näemme pysäköintipaikalla käyneiden autojen lukumääriä vuoden eri kuukausien aikana, kun sakottajaa ei ole huomioitu simuloinnissa. Havaitsemme, että joka kuukausi pysäköintipaikalla käyneiden autojen lukumäärät ovat miltei samat.

Havaitsemme kuvien 6 ja 7 vertailusta, että jos lappuliisa käy, voi autojen lukumäärien erotus olla jopa 1 900 autoa kuukaudessa. Keskimäärin havaitsemme kuitenkin tämän erotuksen olevan noin 900 autoa. Tämä on myös hyvin paljon, sillä se tarkoittaisi, että pysäköintipaikalla kävisi keskimäärin päivässä noin 30 autoa vähemmän.

Lopputuloksena voimme siis todeta, että jos lappuliisa käy keskimäärin kerran kuukaudessa pysäköintipaikalla, on käyvien autojen lukumäärän lasku huomattava. Tämä on realistinen tulos simuloinnista.

Havaitsemme kuitenkin simuloinnin seurauksena, että suurin osa pysäköintipaikan käyttäjistä ovat ne, joilla ei ole pysäköintipaikalle lupaa pysäköidä. Tämä on epärealistista, sillä parkkipaikan oletetaan olevan mitoitettu juuri autoille, joilla löytyy pysäköintioikeus. Raportin havaintojen perusteella tämä hypoteesi ei kuitenkaan pidä paikkaansa ja huomaamme suurimman

pysäköintipaikkaa käyttävän ryhmän olevan teekkarit.

Johtopäätöksenä Teknillisen korkeakoulun pysäköintipaikan parkkipaikan käytön maksimoimiseksi voisi pysäköintiluvan saamisen kriteereitä laajentaa isommalle ryhmälle, jolloin pysäköintipaikan käyttö olisi järkevää rahallisesti ja se edistäisi korkeakoulun liikkuvuutta.

## 4 Lähdekoodit

### 4.1 painokerroinfunktio

```
function painokerroin = painokerroin(i, k)
p1 = 0.7^(i); % Osoittaja, joka riippuu indeksistä i
p0 = zeros(1,k); % Painokertoimen summa, eli vakio c, joka normittaa p(i):n
for j = 1:k          % c:n laskeminen
    p0(j) = 0.7^(j);
end

painokerroin = p1/sum(p0); %Painokerroin
end
```

### 4.2 A simulointi

```
% VK12 työharjoitus Studio
% a) kohta:
clear all
close all

figure
hold on

for c = 1:10          % Looppi, kuinka monta kertaa teemme erilaisissa variatiota.

    % Lambda: odotusarvo tietyllä kellonaikana saapuvien autojen lkm:ssä
    lambda = [3; 3; 3; 3; 3; 3; 10; 20; 60; 40; 30; 20;
              40; 20; 30; 14; 13; 15; 15; 10; 8; 3; 3; 3];
    % Sigma: keskihajonta tietyllä kellonaikana saapuvien autojen lkm:ssä
    sigma=[3; 3; 3; 2; 2; 2; 3; 5; 7; 10; 10;
            10; 7; 5; 3; 2; 2; 3; 3; 4; 4; 2; 2; 2];

    % Autojen lukumäärät eri kellonaikana, alustus (vektori)
    autot = [];
    % Autot hetkellä klo 0:00 on 10:
    autot(1) = 10;

    % Plot: aika
    t = 0:24;

    % Max autopaikkojen määrä parkkipaikalla
    autot_max = 200;
    % Viivetermi (vakio)
    k = 5;
```

```

% Virhetermin keskihajonta, poistuvien autojen lkm:ssä
sigma_mu = 5;

% painokertoimien alustus summan laskua varten.
pi = zeros(1, k);

% Poistuvat ja tulevien autojen lkm alustus
poistuvat = [];
tulevat = []; % Tulevat käytetään myös mu_t kohdan laskussa!

% päivän kellonaikojen looppi, lasketaan jokaiselle kellon ajalle
% omat autojen lkm, kun generoimme joka kerralla autojen lkm, olevat
% satunnaisuudet

for i = 1:24
    if autot < autot_max % Jos paikkoja jäljellä
        arrive = max( round(normrnd(lambda(i), sigma(i))), 0);
        % Autoja saapuu; generointi, luonnollinenluku!
        tulevat(end + 1) = arrive;
    end

    % Autojen poistumisen laskeminen
    if i > k % ehto liittyen aikaan i+j-k
        for j = 1:k % Painokertoimien summien lasku
            pi(j) = painokerroin(j,k)*tulevat(i+j-k); % Summa p(i)*l(t+i-k)
        end
        % Autojen poistumisen generointi, luonnollinenluku!
        mu1 = max( round( sum(pi) + normrnd(0,sigma_mu) ), 0);
        mu = min(mu1, autot(end));
        % Poistuvien autojen lkm voi olla max parkkipaikalla
        % olevien autojen verran
    else
        mu1 = max(round(normrnd(0,sigma_mu)),0);
        % Autojen lukumäärä, luonnollinenluku!
        mu = min(mu1, autot(end));
    end

    poistuvat(end+1) = mu; % Poistuvat vektorin täyttö

    a = autot(end) + arrive - mu; % Autojen netto lukumäärä
    autot_new = max(a,0); % Uusi määrä autoja parkkipaikalla
    % max:illa ei periaatteessa
    % mitään merkitystä
    autot(end + 1) = autot_new; % Autojen lkm vektorin täyttö
end

```



```

end

stairs(t, autot) % Autojen lkm vuorokauden aikana
                % kuva
end

grid on
title('Autojen lukumäärät pysäköintipaikalla')
xlabel('vuorokauden ajat tunneissa')
ylabel('autojen lukumäärät')
xlim([0,24])
ylim([0,200])

figure(2);
bar(poistuvat, 0.5) % Poistuvien autojen lkm yhden variaation aikana
grid on
title('Poistuvien autojen lukumäärät pysäköintipaikalta')
xlabel('vuorokauden ajat tunneissa')
ylabel('poistuvien autojen lukumäärät')
ylim([0,80])
xlim([0,24])

figure(3);
bar(tulevat, 0.5) % Tulevien autoje lkm yhden variaation aikana
grid on
title('Saapuvien autojen lukumäärät pysäköintipaikalta')
xlabel('vuorokauden ajat tunneissa')
ylabel('saapuvien autojen lukumäärät')
ylim([0,80])
xlim([0,24])

```

## 4.3 B simulointi

### 4.3.1 Liisalappu funtkio

```

function [t, autot, tulevat, poistuvat] = liisalappu

% VK12 työharjoitus Studio

% Odotusarvo ja keskihajonta saapuville autoille, kun lappuliisa on
%käynyt
lambda = 3;
sigma = 2;

```

```

% Autojen lukumäärät eri kellonaikana, alustus (vektori)
autot = [];
% Autoja hetkellä klo 0:00 10:
autot(1) = 10;

% Plot: aika
t = 0:24;

% Max autopaikkojen määrä parkkipaikalla
autot_max = 200;
% Viivetermi (vakio)
k = 5;
% Virhetermin keskihajonta, poistuvien autojen lkm:ssä
sigma_mu = 5;

% painokertoimien alustus summalle.
pi = zeros(1, k);

poistuvat = [];
tulevat = []; %1

for i = 1:24
    if autot < autot_max % Jos paikkoja jäljellä

        arrive = max( round(normrnd(lambda, sigma)), 0);
        % Autoja saapuu; generointi, luonnollinenluku!

        tulevat(end + 1) = arrive;
    end

    if i > k
        for j = 1:k % Autojen poistuminen
            pi(j) = painokerroin(j,k)*tulevat(i+j-k); % Summa p(i)*l(t+i-k)
        end

        mu1 = max( round( sum(pi) + normrnd(0,sigma_mu) ), 0);
        % Autojen poistumisen generointi, luonnollinenluku!
        mu = min(mu1, autot(end));
        % Poistuvien autojen lkm, max parkkipaikalla olevien autojen verran

    else
        mu1 = max(round(normrnd(0,sigma_mu)),0); %
        % Autojen lukumäärä, luonnollinenluku!
    end
end

```

```

        mu = min(mu1, autot(end));
    end
    poistuvat(end+1) = mu;

    a = autot(end) + arrive - mu;    % Autojen netto lukumäärä
    autot_new = max(a,0);            % Uusi määrä autoja parkkipaikalla
    autot(end + 1) = autot_new;
end
end

```

#### 4.3.2 Parkkipaikka funktio

```

function [t, autot, tulevat, poistuvat] = parkkipaikka

% VK12 työharjoitus Studio

% Lambda: odotusarvo tietyllä kellonaikana saapuvien autojen lkm:ssä
lambda = [3; 3; 3; 3; 3; 3; 10; 20; 60;
40; 30; 20; 40; 20; 30; 14; 13; 15; 15; 10; 8; 3; 3; 3];
% Sigma: keskihajonta tietyllä kellonaikana saapuvien autojen lkm:ssä
sigma = [3; 3; 3; 2; 2; 2; 3; 5; 7; 10;
10; 10; 7; 5; 3; 2; 2; 3; 3; 4; 4; 2; 2; 2];

% Autojen lukumäärät eri kellonaikana, alustus (vektori)
autot = [];
% Autoja hetkellä klo 0:00 10:
autot(1) = 10;

% Plot: aika
t = 0:24;

% Max autopaikkojen määrä parkkipaikalla
autot_max = 200;
% Viivetermi (vakio)
k = 5;
% Virhetermin keskihajonta, poistuvien autojen lkm:ssä
sigma_mu = 5;

% painokertoimien alustus summalle.
pi = zeros(1, k);

poistuvat = [];
tulevat = []; %1

```

```

for i = 1:24
    if autot < autot_max % Jos paikkoja jäljellä

        arrive = abs( round(normrnd(lambda(i), sigma(i))) );
        % Autoja saapuu; generointi, luonnollinenluku!

        tulevat(end + 1) = arrive;
    end

    if i > k
        for j = 1:k          % Autojen poistuminen
            pi(j) = painokerroin(j,k)*tulevat(i+j-k);%Summa p(i)*l(t+i-k)
        end

        mu1 = abs( round( sum(pi) + normrnd(0,sigma_mu) ));
        % Autojen poistumisen generointi, luonnollinenluku!
        mu = min(mu1, autot(end));
        % Poistuvien autojen lkm,
        % max parkkipaikalla olevien autojen verran

    else
        mu1 = abs(round(normrnd(0,sigma_mu)));
        % Autojen lukumäärä, luonnollinenluku!
        mu = min(mu1, autot(end));
    end
    poistuvat(end+1) = mu;

    a = autot(end) + arrive - mu; % Autojen lukumäärä
    autot_new = max(a,0);         % Uusi määrä autoja parkkipaikalla
    autot(end + 1) = autot_new;

end
end

```

#### 4.3.3 Liisalappu käynnin simulointi

```

% VK12 työharjoitus Studio
% b) kohta:
% 3 Kuukaudessa olevat päivän ajat, kun lappuliisa käy satunnaisesti
% keskimäärin 1 kerran kuukaudessa

clear all

```

```

close all

figure
hold on

A = [];          % Päivien tulevien autojen lkm
PL = [];         % Parkkiliisan jokaisen mahdollisen iteroidun päivän aikana
                % oleva alustus
                % tarkastelun vuoksi
saapuneet = []; % Kaikkien tulevien autojen joka tunti olevat määrät
                % samassa vektorissa, alustus.

for i = 1:3      % Kuukausien lasku
    c = 1;       % Ensimmäisen päivän alustus
    while c <= 30 % Kuukauden päivien autojen lasku

        parkkiliisa = round(30*rand(1,1));
        % Liisalapun mahdollinen käynti päivä

        if parkkiliisa + 1 == c % Jos Liisalappu käy kyseisenä päivänä
            PL(end+1) = parkkiliisa; % Lappuliisa

            while c < parkkiliisa + 3 % Lappuliisa seurauksena 2 seuraavan
                                    % päivän saapumiset
                [t, autot, tulevat, poistuvat] = liisalappu;
                stairs(t,autot)

                A(end + 1) = sum(tulevat); % päivien autojen lkm lisääminen
                c = c + 1;                 % Uuden päivän vaihto/lasku
                for h = 1:24
                    saapuneet(end + 1) = tulevat(h); % Saapuneet vektorin
                                                    % täyttö
                end
            end
        else % Jos Lappuliisa ei käy
            [t, autot, tulevat, poistuvat] = parkkipaikka;
            A(end + 1) = sum(tulevat); % Päivän saapuvat autot
            c = c + 1;                 % Uuden päivän vaihto/lasku
            for h1 = 1:24
                saapuneet(end + 1) = tulevat(h1);
            end
        end
    end
end

```

```

        stairs(t, autot)
        % Autojen lkm vuorokauden aikana    % kuva
    end
end

grid on
title('Autojen lukumäärät pysäköintipaikalla')
xlabel('vuorokauden ajat tunneissa')
ylabel('autojen lukumäärät')
xlim([0,24])
ylim([0,200])

% 3 Kuukauden aikana olevien saapuneiden autojen lkm päivittäin; kuva
figure(2);
bar(A, 0.5)
grid on
title('Saapuneiden autojen lukumäärät kuukauden eri päivinä')
xlabel('vuorokaudet')
ylabel('Saapuneiden autojen lukumäärät vuorokaudessa')
xlim([0,90])

% 3 kuukauden jokaisen tunnin aikana saapuvat autot visualisoinnin avuksi
figure(3);
stairs(1:1:2160,saapuneet)
grid on
title('Saapuneiden autojen lukumäärät tunneissa')
xlabel('Tunnit')
ylabel('Autojen lukumäärät')
xlim([0,2160])

```

#### 4.3.4 Kuukausi funktio

```

function [lkm] = kuukausi
% Lähes sama kuin liisalappu käynnin simulointi kerran kuukaudessa

A = [];          % Päivien tulevien autojen lkm
PL = [];         % Parkkiliisan jokaisen mahdollisen iteroidun päivän aikana
                % oleva alustus
                % tarkastelun vuoksi

c = 1;          %Ensimmäisen päivän alustus

while c <= 30    % Kuukauden päivien autojen lasku

```

```

parkkiliisa = round(30*rand(1,1));
% Liisalapun mahdollinen käynti päivä

if parkkiliisa + 1 == c % Jos Liisalappu käy kyseisenä päivänä
    PL(end+1) = parkkiliisa; % Lappuliisa

    while c < parkkiliisa + 3 % Lappuliisa seurauksena 2 seuraavan
        % päivän saapumiset
        [t, autot, tulevat, poistuvat] = liisalappu;
        A(end + 1) = sum(tulevat); % päivien autojen lkm lisääminen
        c = c + 1; % Uuden päivän vaihto/lasku
    end
else % Jos Lappuliisa ei käy
    [t, autot, tulevat, poistuvat] = parkkipaikka;
    A(end + 1) = sum(tulevat); % Päivän saapuvat autot
    c = c + 1; % Uuden päivän vaihto/lasku
end

stairs(t, autot)
% Autojen lkm vuorokauden aikana % kuva
end

lkm = sum(A); % Kuukauden aikana saapuvien autojen summa

```

#### 4.3.5 Parkkikk funktio

```

function [parkkilmkk] = parkkikk
    A = [];
    for c = 1:30
        % Looppi, kuinka monta kertaa teemme erilaisissa variatiota.
        [t, autot, tulevat, poistuvat] = parkkipaikka;

        A(end + 1) = sum(tulevat);

    end

    parkkilmkk = sum(A); % Kuukauden aikan saapuvien/käyvien autojen lkm

```

#### 4.3.6 Liisalapun käynti simulointi kuukausittain, vuoden ajan

```

% Viikko 12 B, 3
% Vuoden aikana eri kuukausien lkm generointi
clear all

```

```

close all

kuukaudetlappuliisa = [];
kuukaudet = [];

% Kun Lappuliisa käy
for m = 1:12
    kuukaudetlappuliisa(end + 1) = kuukausi;
end

% Jos Lappuliisa ei käy
for k = 1:12
    kuukaudet(end + 1) = parkkikk;
end

% Kun Lappuliisa käy, kuva
figure(1);
bar(kuukaudetlappuliisa, 0.5)
% Poistuvien autojen lkm yhden variaation aikana
grid on
title('Käytyjen autojen lukumäärät vuoden eri kuukausina, kun lappuliisa käy')
xlabel('Kuukaudet')
ylabel('Saapuneiden autojen lukumäärät')
xlim([0.5,12.5])

% Kun Lappuliisa ei käy, kuva
figure(2);
bar(kuukaudet, 0.5)      % Poistuvien autojen lkm yhden variaation aikana
grid on
title('Käytyjen autojen lukumäärät vuoden eri kuukausina, kun lappuliisa ei käy')
xlabel('Kuukaudet')
ylabel('Saapuneiden autojen lukumäärät')
xlim([0.5,12.5])

```