



UNIVERSIDADE DE BRASÍLIA
SIMULAÇÃO DE SISTEMAS

RELATÓRIO DA SIMULAÇÃO EM MATLAB DE UM CRUZAMENTO ENTRE RUAS COM SINAL DE TRÂNSITO

GUSTAVO PESSOA CAIXETA PINTO DA LUZ
15/0128720

Brasília - DF
2018

Sumário

1. Introdução	3
2. Análise do Problema.....	3
3. Descrição da Simulação	4
3.1 Código Comentado	4
3.2 Flowchart da Simulação	8
4. Resultados de Atrasos	12
5. Gráfico.....	13
6. Custo para a Sociedade	14
7. Conclusão	15

1. Introdução

A Simulação de Sistemas consiste na utilização de programas de computador que fazem do uso de técnicas matemáticas para representar o funcionamento de sistemas reais ou não. Com o uso de programas, fica mais clara a visualização de situações complexas, e com isso a tomada de decisão tem um maior embasamento. Para a Engenharia de Produção, o conhecimento de técnicas de Simulação de Sistemas é essencial para que problemas sejam resolvidos com uma modelagem adequada e mais próxima da realidade. Além disso, a Simulação é uma grande aliada à Pesquisa Operacional, um grande campo da Engenharia de Produção. Uma das formas de se simular sistemas, é utilizando softwares específicos para isso.

O MATLAB é um software que possui a sua própria linguagem utilizado principalmente por engenheiros que fornece um ambiente interativo amigável com diversos recursos. Um diferencial do software é o foco em cálculo numérico, a criação de gráficos e um menor tempo para solucionar problemas do que outras linguagens, como C e Basic.

O ambiente de um cruzamento entre duas vias pode ser simulado utilizando o MATLAB, o que permite um maior entendimento do contexto e gera uma visualização complexa da realidade a partir de gráficos e valores calculados matematicamente, aplicando a essência da Simulação de Sistemas.

2. Análise do Problema

O problema apresentado consiste em uma interseção de duas ruas que possuem sinais de trânsito: Elm e Main. Os carros chegam seguindo uma distribuição uniforme aleatória sendo que a rua Elm segue a distribuição (4,7) e a rua Main segue a distribuição (3,5), tudo em segundos. Isso significa que a rua Elm é mais movimentada do que a Main.

Os sinais ficam abertos por 30 segundos e 1 veículo passa a cada 2s, ou seja, 30 veículos passam por minuto. Para fins de análise, se considera que se não tiver fila, um veículo passa em 0 segundos (quando o sinal abre). Sobre o tempo de

espera, se um carro chega assim que o sinal ficou vermelho esperará 30 segundos, se ele chega 10 segundos antes de ficar verde e for o 2º veículo esperará 12 segundos (10 segundos até ficar verde + 2 segundos da passada do primeiro carro).

O desafio desse projeto é criar uma simulação de como ocorre o tráfego de carros no cruzamento durante 15 minutos. Para que isso fique claro o programa deve gravar o número de carros que chegou em cada uma das ruas e o número de carros que saiu de cada uma das ruas de acordo com o tempo de simulação. Para que a ilustração fique clara, o projeto mostra os dados em um gráfico que mostra os veículos pelo tempo de simulação.

Além disso, um problema é descobrir o atraso total de cada uma das ruas gerado pelo sinal, além da média dos atrasos após várias simulações terem acontecido. Avançando mais, o programa deve calcular o custo diário para a sociedade do atraso gerado pelos atrasos do cruzamento, o que pode servir inclusive de subsidio para gestores públicos tomarem decisões administrativas.

3. Descrição da Simulação

Para a descrição da Simulação, utilizou-se o código e um Flowchart.

O código ("intersection") contém a descrição detalhada de cada parte do código, de acordo com a parte analisada. Em linhas gerais, são geradas as chegadas e saídas nas duas ruas analisadas, o atraso das duas ruas, o atraso médio de 20 simulações e um gráfico de Carros x Tempo de ambas as ruas.

3.1 Código Comentado

```
function intersection

%Este código contém:
%1. O atraso em horas dos veículos das Ruas Elm e Main
%2. Uma média do atraso total das duas ruas e médio em 20 simulações
%3. O código que gera o gráfico de Carros x Tempo nas duas ruas

for K= 1:20 % A primeira coisa a ser feita é definir o número de runs da
simulação.
    %Utilizou-se 20 já que foi solicitado que fizessemos no mínimo 10
    %Primeiramente, definiremos a chegada das duas ruas (elm e main)

%   CHEGADA ELM
```

```

%As duas chegadas seguem um padrão:
%(1)primeiramente se define uma variável contadora das chegadas,
%(2)depois é gerado um tempo randômico de acordo com a distribuição
uniforme da rua
%(3)e o programa mostra que chega 1 carro na rua inicialmente

i=1; %(1)

chegada_elm(1)= unifrnd(4,7); %(2)

ncarros_chegam_elm(1)=1; %(3)

%gerar todas as chegadas na rua elm
while chegada_elm(i) <= 900; %A simulação durará 15 minutos, o que
representa 900 segundos
    T_chegada_elm= unifrnd (4,7); %é gerado um tempo randômico de chegada
    chegada_elm(i+1)= chegada_elm(i) + T_chegada_elm; %é gerado o próximo
tempo de chegada na rua
    ncarros_chegam_elm(i+1)= i+1; %cálculo do número de carros que estão
na rua
    i=i+1; %variável contadora que vai sendo atualizada
end
chegada_elm(i)=900; %limite da simulação, para que não sejam gerados
números maiores que 900.


%   CHEGADA MAIN
%as chegadas na rua main seguirão a mesma lógica da rua elm, apresentando
os 3 passos explicados.
%A única diferença entre as duas ruas é a distribuição uniforme randômica
i=1;
chegada_main(1)= unifrnd(3,5);
ncarros_chegam_main(i)=1;
while chegada_main(i)<= 900;
    T_chegada_main= unifrnd(3,5);
    chegada_main(i+1)= chegada_main(i) + T_chegada_main;
    ncarros_chegam_main(i+1)= i+1;
    i=i+1;
end
chegada_main(i)=900;


%   SAIDAS

%O segundo passo é gerar as saídas das duas ruas:
%Além da variável contadora (i) , temos uma variável auxiliar de controle
(j), que indica a alternância do sinal


%   SAIDA ELM
i=1;
j=1 ;
ncarros_saem_elm(i)=1;
saida_elm(1)=31; %Como o sinal da rua Elm se inicia fechado, o primeiro
carro sai de elm em 31.

```

```

while saida_elm(i)<=900;
    while(saida_elm(i)>=(j*30)+1) && (saida_elm(i)<((j+1)*30));
        if chegada_elm(i+1)< saida_elm(i); %Possibilidade: chegada do 2°
antes que a saída do 1°
            saida_elm(i+1)=saida_elm(i)+2; %saída após a saída do
anterior com o acréscimo de 2 segundos
            ncarros_saem_elm(i+1)=i+1; %conta quantos carros saíram

        else %2° chega e o 1° já saiu
            saida_elm(i+1)=chegada_elm(i+1); %chega no sinal e já sai

            ncarros_saem_elm(i+1)= i+1; %conta quantos carros saíram
        end
        i=i+1;
    end
    j=j+2; %próxima abertura do sinal
    saida_elm(i)=j*30+1;
end
saida_elm(i)=900;

%   SAIDA MAIN

%As saídas na rua main seguirão a mesma lógica da rua elm, apresentando
os mesmos passos explicados.
%(*)A adequação realizada foi o respeito a abertura do sinal em tempos
diferentes, que ocorre inicialmente de 1 a 30s.

i=1;
j=1;
ncarros_saem_main(i)=1;
saida_main(i)=chegada_main(i);

while saida_main(i)< 900
    while(saida_main(i)>=((j-1)*30)+1) && (saida_main(i)<=(j*30))%para se
adequar a condição apresentada (*)
        if chegada_main(i+1)<=(saida_main(i))
            saida_main(i+1)=saida_main(i)+2;
            ncarros_saem_main(i+1)=i+1;

        else
            saida_main(i+1)=chegada_main(i+1);

            ncarros_saem_main(i+1)= i+1;
        end
        i=i+1;
    end
    j=j+2;
    saida_main(i)=(j-1)*30+1;
end
saida_main(i)=900;

%   ATRASOS
%Para o cálculo dos atrasos, primeiramente deve-se criar uma nova
variável para normalizar o vetor de saídas das ruas.

%Depois de normalizado, é calculado o atraso para cada rua a partir da

```

```
%subtração entre saídas e chegadas e se calcula o atraso total da rua, de acordo com a variável m criada
```

```
%ATRASSO MAIN
```

```
normalizado_main= ncarros_saem_main(length(ncarros_saem_main));
```

```
for M=1:normalizado_main; %M vai de 1 até a ultima saída da rua
    atraso_main(M)= saida_main(M)-chegada_main(M);
end
```

```
atraso_total_main=sum(atraso_main);
```

```
%ATRASSO ELM
```

```
%Para se calcular o atraso na rua Elm, utilizou-se a mesma lógica do atraso em main.
```

```
%A única diferença foi a variável utilizada(E).
```

```
normalizado_elm= ncarros_saem_elm(length(ncarros_saem_elm));
```

```
for E=1:normalizado_elm;
    atraso_elm(E)= saida_elm(E)-chegada_elm(E);
end
atraso_total_elm=sum(atraso_elm);
```

```
%ATRASSO TOTAL MÉDIO
```

```
%Primeiramente, fez-se a soma entre os atrasos totais de elm e main.
```

```
%Depois, foi feita uma média desses atrasos, o que aparece em segundos.
```

```
atraso_elm_e_main(K)=atraso_total_elm + atraso_total_main; %O K é o mesmo do início do programa (linha 2)
```

```
end
```

```
%Os atrasos totais aparecem aqui para que seja gerado somente um valor
```

```
atraso_total_elm = mean(atraso_total_elm)
```

```
atraso_total_main = mean(atraso_total_main)
```

```
atraso_medio= mean(atraso_elm_e_main) %em segundos
```

```
%ATRASSO TOTAL EM HORAS
```

```
atraso_em_horas= (atraso_medio)/3600 %transformando o atraso em horas
```

```
% GRÁFICO
```

```
%Código para Plotar o Gráfico de Carros x Tempo nas duas ruas
```

```
plot(chegada_elm,ncarros_chegam_elm,'k') , hold on; %chegadas elm
```

```
plot(saida_elm,ncarros_saem_elm), hold on; %saidas elm
```

```
plot(chegada_main,ncarros_chegam_main), hold on; %chegadas main
```

```
plot(saida_main,ncarros_saem_main), hold off; %saidas main
```

```
title('Gráfico de Carros por Tempo nas duas ruas');
```

```
xlabel('Tempo');
```

```
ylabel('Carros');
```

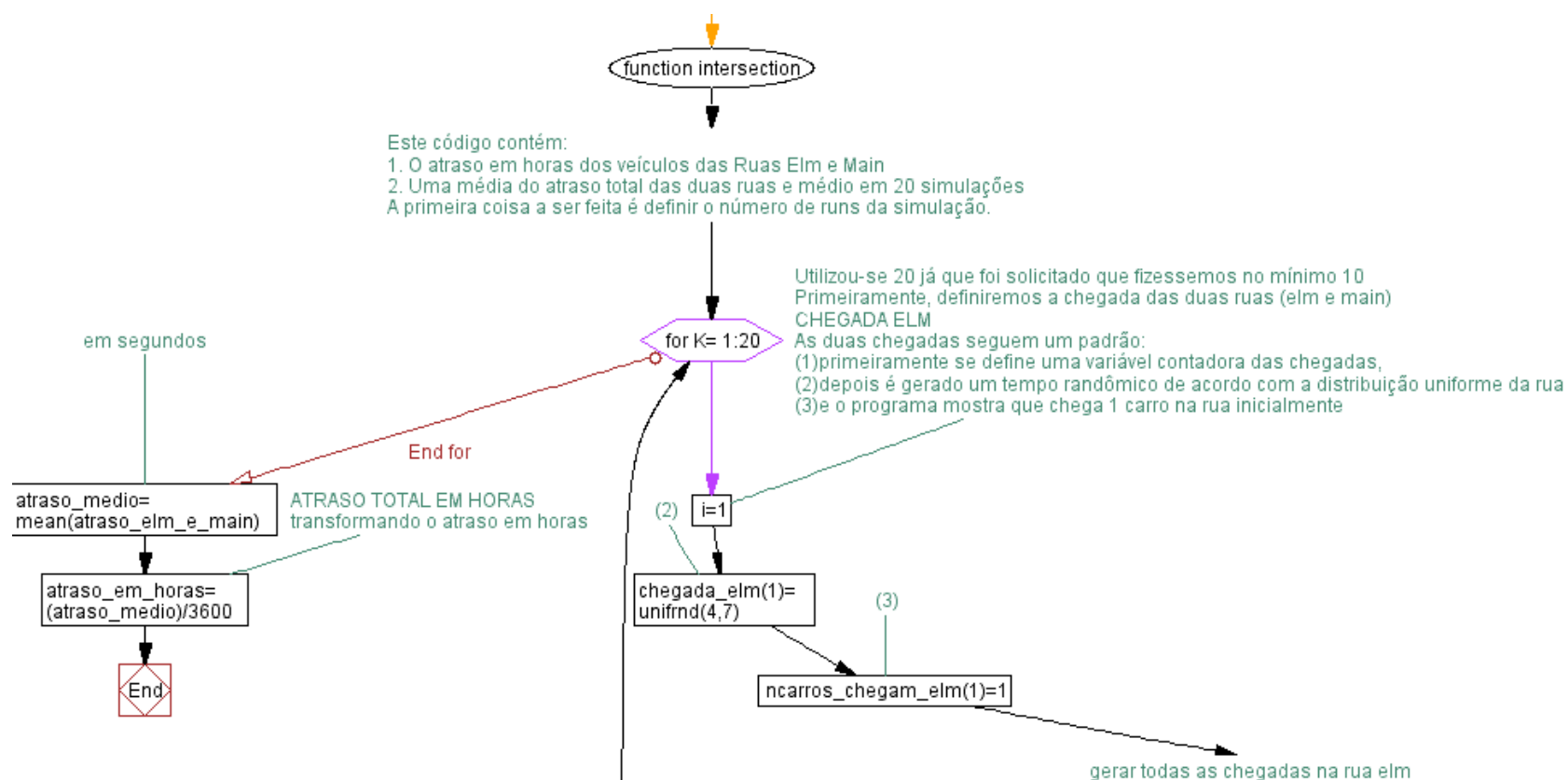
```
axis ([0 950 0 300]);
```

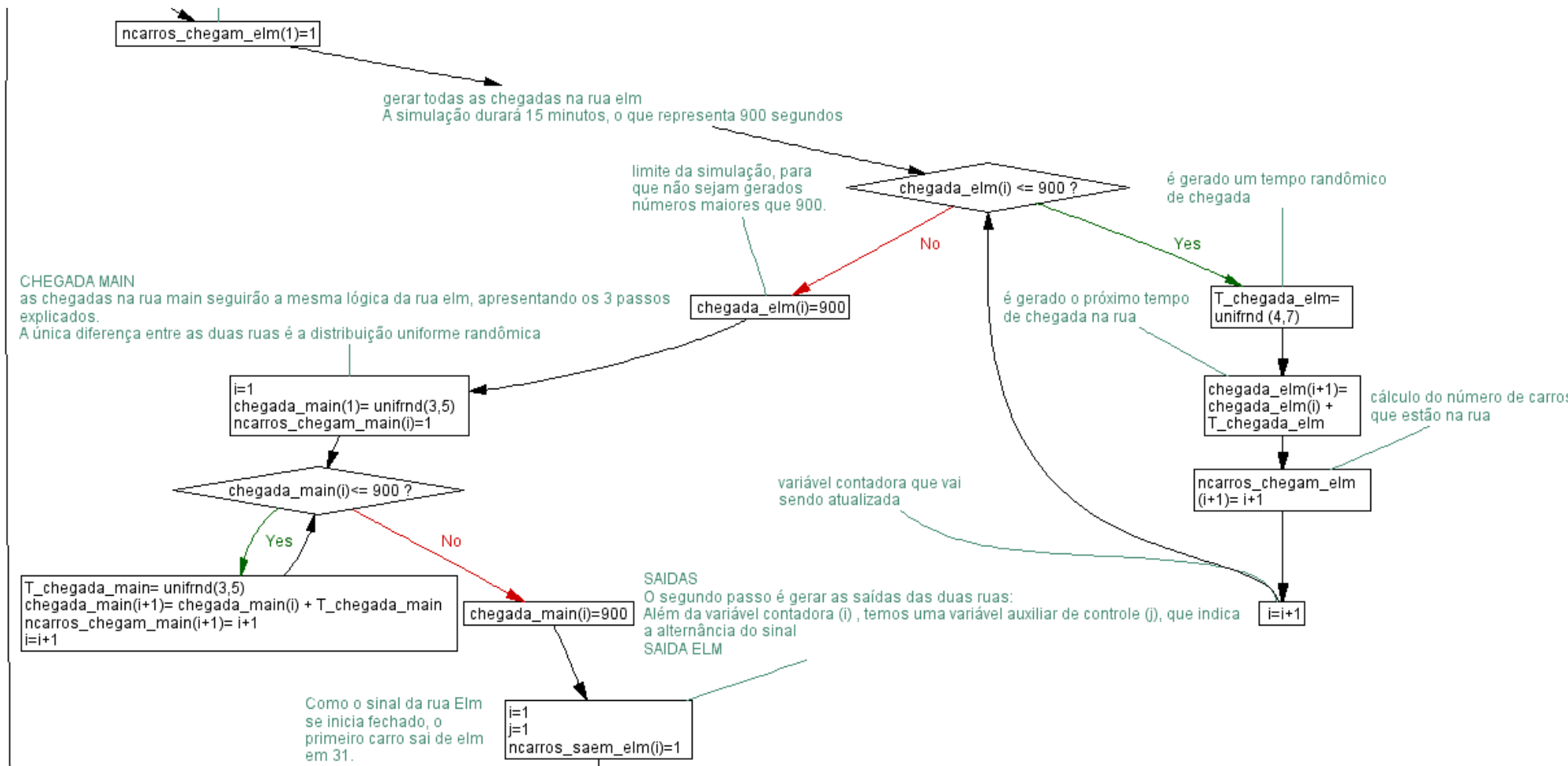
```
legend('\color{orange}Rua Elm', '\color{blue}Rua Main');
```

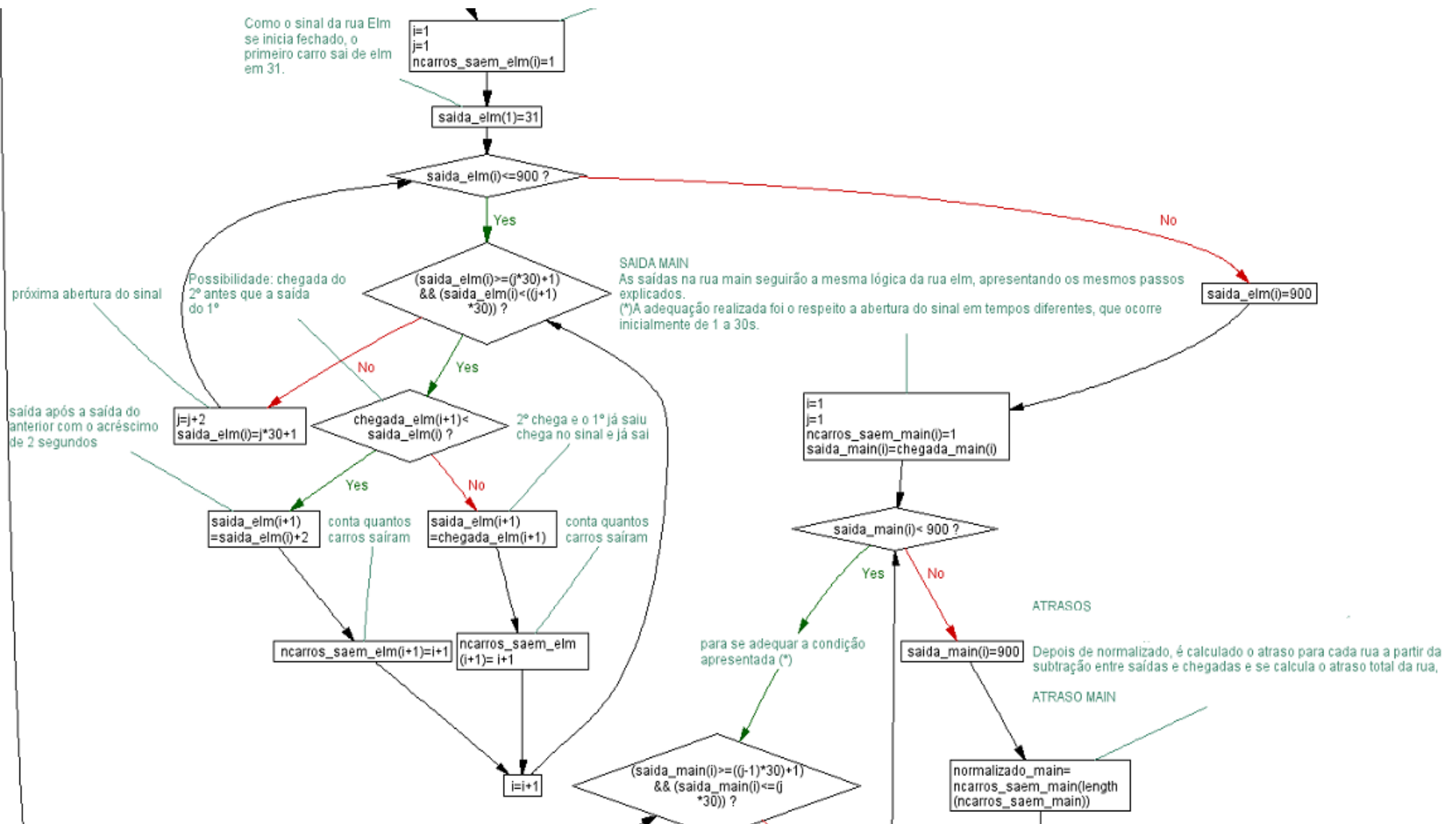
```
end
```

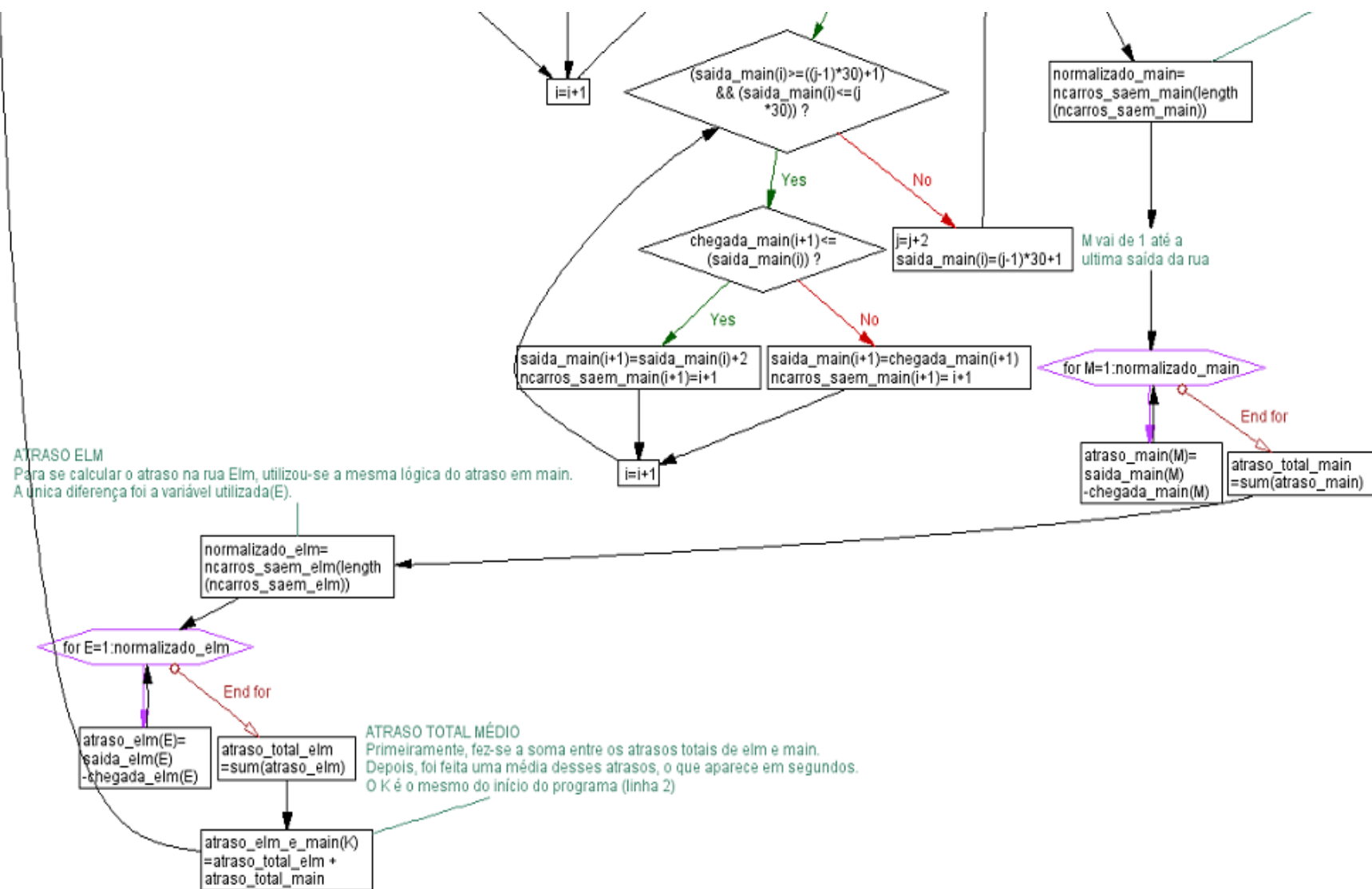
3.2 Flowchart da Simulação

O flowchart foi dividido para que ocorra a melhor compreensão de cada parte do código:





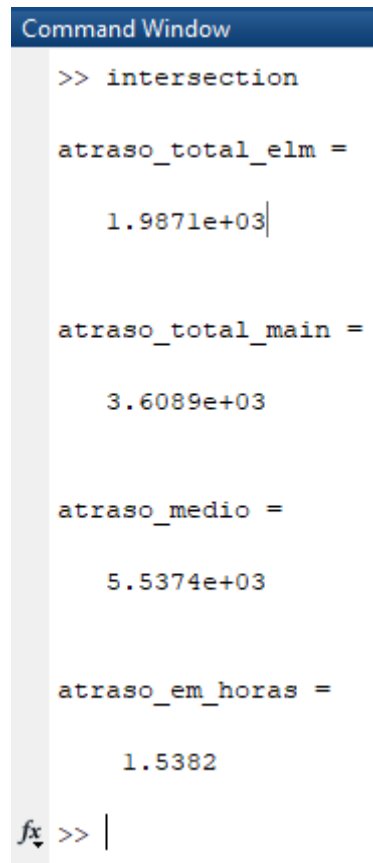




4. Resultados de Atrasos

Para o cálculo de atraso, foi feita a subtração entre chegadas e saídas nas ruas para cada uma delas. Depois, foi feita uma média da soma entre os atrasos das duas ruas para determinar o atraso total médio, que foi convertido em horas.

O MATLAB encontrou os seguintes atrasos:



```
Command Window
>> intersection

atraso_total_elm =

    1.9871e+03|

atraso_total_main =

    3.6089e+03

atraso_medio =

    5.5374e+03

atraso_em_horas =

    1.5382

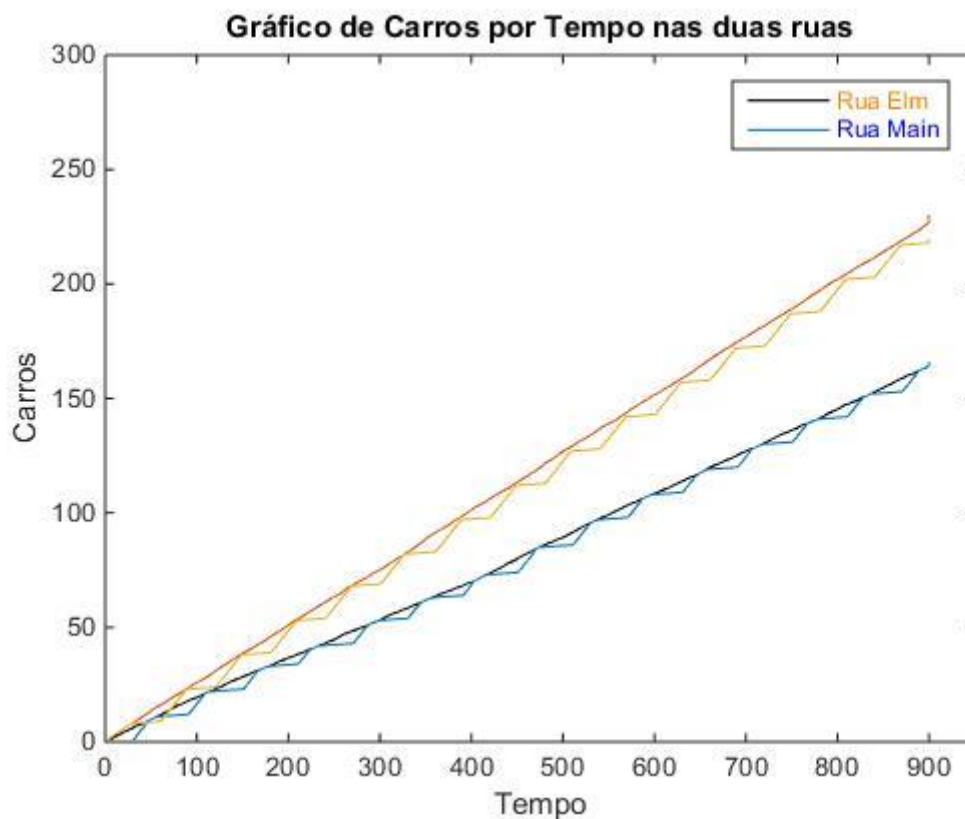
fx >> |
```

- 1- O 1º valor indica o atraso total da rua Elm
- 2- O 2º valor indica o atraso total da rua Main:
- 3- O 3º valor indica o atraso médio total das 20 Simulações (em segundos) e o 4º valor simplesmente transforma esse atraso em horas.

Obs: O código utilizado está no tópico 3.1 deste documento ([Código com Explicações](#)).

5. Gráfico

Abaixo pode-se visualizar o gráfico de uma das simulações, ilustrando as chegadas e saídas das duas ruas:



A linha alaranjada mostra as chegadas e saídas da Rua Elm, sendo que as chegadas são representadas pela linha mais uniforme e as saídas são representadas pelos “dentes”. A mesma lógica é seguida na Rua Main, representada pela linha azulada.

Obs: O código utilizado está no tópico 3.1 deste documento ([Código com Explicações](#)).

6. Custo para a Sociedade

Para o cálculo de custo para a sociedade em 24 horas de simulação, utilizou-se de outro código (“umdia”), a partir de alterações pontuais. Assim, onde aparecia 900 segundos (15 minutos) passou a aparecer 86400 segundos (24 horas).

Além disso, foram adicionadas as últimas 2 linhas abaixo ao código:

```
%Os atrasos totais aparecem aqui para que seja gerado somente um valor

atraso_total_elm = mean(atraso_total_elm)

atraso_total_main = mean(atraso_total_main)

atraso_medio = mean(atraso_elm_e_main) %em segundos

%ATRASSO TOTAL EM HORAS
atraso_em_horas = (atraso_medio)/3600 %transformando o atraso em horas

%supondo que os motoristas ganham em média R$ 4000 por mes.

custo_para_sociedade = (atraso_em_horas*4000)/160
```

Para esse cálculo, assim como instruído, foi feita a suposição que os motoristas ganham em média R\$ 4000 por mês e que trabalham 8 horas por dia, 5 dias na semana, 4 semanas por mês (160 horas).

Após cálculos, o MATLAB encontrou um custo de R\$ 9.564 por dia:

```
custo_para_sociedade =

    9.5645e+03
```

7. Conclusão

Como dito na Análise do Problema já se sabia que a rua Main é mais movimentada do que a Elm, de acordo com as distribuições uniformes randômicas. A partir da análise do gráfico e análise separada dos valores de chegada, saída e atrasos por meio de breakpoints, pode-se perceber que há um grande número de chegadas na rua Main e pouca saída, o que indica que essa rua deveria ter um maior tempo com o sinal aberto. A rua Elm já possui um menor número de chegadas e menor tendência a engarrafamentos do que a Main, o que fica nítido quando se projeta o gráfico a longo prazo.

Para continuidade do projeto, como um trabalho futuro, seria ideal estimar o tempo ótimo de abertura do sinal de cada uma das ruas de modo a otimizar o atraso e custo de acordo com as chegadas e saídas.