Englander ~~2071~~
2186

REPORT NO. DOT-TSC-OST-71-6

TRAFFIC CIRCLE MODEL

IRWIN ENGLANDER

TRANSPORTATION SYSTEMS CENTER
55 BROADWAY
CAMBRIDGE, MA. 02142

DEPARTMENT OF TRANSPORTATION
UNITED STATES OF AMERICA

MAY 1971
PRELIMINARY MEMORANDA

Prepared for

OFFICE OF THE SECRETARY OF TRANSPORTATION
WASHINGTON ,D.C. 20590

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Traffic Circle Model		5. Report Date May 1971 6. Performing Organization Code TCD
7. Author(s) Irwin Englander		8. Performing Organization Report No. DOT-TSC-OST-71-6
9. Performing Organization Name and Address Transportation Systems Center 55 Broadway Cambridge, Ma. 02142		10. Work Unit No. R-1070 11. Contract or Grant No. DOT/TSC-16
12. Sponsoring Agency Name and Address Department of Transportation Office of the Secretary Washington, D.C. 20590		13. Type of Report and Period Covered Preliminary Memoranda 14. Sponsoring Agency Code OS-12

15. Supplementary Notes

16. Abstract

This report describes a dynamic model of a traffic circle which has been implemented on a CRT display terminal.

The model includes sufficient parameters to allow changes in the structure of the traffic circle, the frequency of traffic introduced to the circle by each entering road, the model of entering the circle (right of way), the frequency of traffic leaving the circle by each road, and vehicular spacing toward the circle.

17. Key Words	18. Distribution Statement
Dynamic Model Traffic Circle CRT Display Terminal	Unclassified-Unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	14	

MODEL OF A SIMPLE TRAFFIC CIRCLE

1.0 Introduction

This report describes a dynamic model of a traffic circle which has been implemented on a CRT display terminal.

The model includes sufficient parameters to allow changes in the structure of the traffic circle, the frequency of traffic introduced to the circle by each entering road, the model of entering the circle (right of way), the frequency of traffic leaving the circle by each road, and vehicular spacing around the circle.

The model shows the stream of traffic in motion around the circle. Accidents and/or traffic jams have a definite probability of occurence and are also displayed as they occur. The traffic flow and the occurence of accidents and/or traffic jams will be changeable through the display terminal. The assumptions associated with this model are listed below.

1) Each feeder road will supply traffic for the circle according to its own Poisson distribution.
2) Cars available on the feeder roads will enter the circle according to a probability function which will depend on the velocity and distance of the nearest approaching car in the circle.
3) Exit to the feeder roads will be determined according to a prespecified probability function. It is assumed that all exiting cars can be handled by the feeder roads as needed with continuous flow (i.e. exit roads will cause

no road jams).

4) Cars in the circle will travel with velocities dependent on the distance from the car in front. Acceleration and velocity bounds for all traffic will be included.

2.0 Display Assumptions

A CRT display terminal was utilized for the implementation of this model.

The basic assumptions for the CRT implementation are listed below.

(1) The traffic circle is displayed as a fixed circle.

(2) The feeder roads are displayed as fixed straight lines which stop where they touch the traffic circle.

(3) The entry/exit-points are where the feeder roads touch the traffic circle.

(4) Cars are displayed as small circles which will move from feeder roads around the circle and vanish upon exiting the circle.

(5) Feeder road entry queues are indicated by one car and a numerical display of the number of cars in the queue.

(6) Traffic jams will be indicated by flashing the entire display.

(7) Accidents will be indicated by flashing the display of the cars involved.

3.0 The Model

The model has five parts. These include (a) adjusting the queue size at each intersection, (b) determining if the first car in the queue enters the circle, (c) determining if a car in the circle leaves at the intersection, (d) adjusting velocities of all cars, and (e) updating all car positions.

The first routine adjusts the queue size at intersection j. Assume there are x_j cars in queue j. Then the question is whether a car will be added to the queue. First, $p_j\ (x_j+1)$, the probability of having $x_j + 1$ cars in queue j is computed. Next a random number p_r is computed. If $p_r \geq p_j\ (x_j + 1)$, then x is incremented.

The second routine determines if a car from queue j enters the circle. Since the cars in the circle have the right of way it can only enter when there is a gap between cars in the circle.

However given an adequate gap not every car will enter. If the intersection j is at angle Θ_j and the nearest car i is at Θ_i with velocity v_i a gap constant q_j is defined.

$$q_j = \frac{(\Theta_j - \Theta_i)\, r_c}{v_i\ \varepsilon_q} \qquad (1)$$

where r_c is the radius of the circle and ε_q is an arbitrary constant.

A random number p_r is generated and if $q_j > p_r$ the car enters. This has two features in that it says a car enters most frequently

when the car spacing is wider or cars in the circle move slower.

The second routine also determines ℓ, the intersection at which the car leaves the circle. Given $p_{\ell,j}$, the probability that a car entering at intersection j leaves at intersection ℓ, then if p_r is a random number ℓ is chosen to be the smallest value meeting criterion (2).

$$p_r < \sum_{K=1}^{\ell} p_{k,j} \qquad (2)$$

The third routine determines whether a car i leaves at intersection j during interval Δt. The first requirement is that the value of ℓ from routine two equals j. The second requirement is that the car must be close to the entrance as defined by criterion (3).

$$(\Theta_j - \Theta_i)\, r_c < v_i\, \Delta t \qquad (3)$$

Having computed all entrances and exits, routine four updates velocities and positions by criteria (4) for interval n.

$$v_{i,n} = \frac{r_c\,(\Theta_{i+1} - \Theta_i)}{\alpha r} \qquad (4a)$$

$$v_{i,n} \leq v_{max} \qquad (4b)$$

$$\beta_d \leq v_{i,n} - v_{i,n-1} \leq \beta_a \qquad (4c)$$

$$v_{i,n} \leq \frac{r_c(\Theta_\ell - \Theta_i)}{\gamma} + v_{exit} \quad (4d)$$

Criterion (4a) states that the velocity is chosen based on spacing between a car and the car in front. Criterion (4b) states that the velocity will never exceed a maximum velocity. Criterion (4c) states a car has a maximum acceleration and a maximum deceleration. Criterion (4d) states a car will slow down as it approaches its exit intersection.

Finally after updating velocities, new positions for interval n are computed.

$$\Theta_{i,n} = \Theta_{i,n-1} + \frac{v_i}{r_c} \Delta t \quad (5)$$

This sequence of routines is repeated for each interval.

4.0 Results

The traffic circle model was used in a demonstration showing the effect of circle radius and speed limit. The circle radius was found to have no effect. This occured since increasing the circle radius led to increased velocities. Our rule on car separation which requires one car length separation for every ten miles per hour meant, independent of radius, each car occupied the same number of radians and limited car entry from the side road.

The effect of speed limit can be explained by noting capacity of the circle depends on cars per second going around the circle.

This term is the product of cars per radian times radians per second. Reducing the speed limit increases the former term and decreases the latter term. In our circle the product has a maximum at fifteen miles per hour.

5.0 Discussion

The structure of the model may be changed in several ways.

1) The status of the feeder roads (i.e. main road, side road) can be changed by altering the feeder and exit probability distribution parameters.

2) A by-pass of the circle by a feeder road can be created by altering the entry and exit probability distribution parameters. (The by-pass will not be physically displayed.)

3) The radius of the circle and the number of feeder roads may be changed.

Other options may be achieved by changing the vehicle spacing parameters and the velocity change parameters. Since the program will be highly molular, even the probability distribution functions will be changeable.

Possible future effort could be in expanding the number of lanes in the circle, including traffic signal effects at entry/exit points and including effects due to traffic signals on the feeder roads. These and other studies can be implementable in a modular fashion allowing a step by step increase in the sophistication of the model.

APPENDIX A: Computer Program

```
C MAIN PROGRAM FOR MODEL
      DIMENSION TI(51),TK(51),VI(51),VK(51),LI(51),LK(51)
      DIMENSION TJ(6),XX(6),XL(6)
       INTEGER QLINE,PNTR,X,Y
      COMMON/A/TI,TK,VI,VK,LI,LK
      COMMON/XX/X(50)
      COMMON/YY/Y(50)
      COMMON/PP/PNTR(2)
      COMMON/NN/NCARS,NROADS
      COMMON/QQ/QLINE(8)
      COMMON/B/TJ,XX,XL
      COMMON/C/NCAR,RC,EQ,VEN,DT
      COMMON/D/ VMAX,BA,BD,ALR
       COMMON/E/JX
       COMMON/F/M
       JX=9
       PII=3.14159/180.
       TJ(6)=-100.
       TK(51)=-100.
       TI(51)=-100.

      CALL SETRAN(JX)
C INPUT RC,EQ,VEN
       WRITE(1,12)
   11 FORMAT(20H RC,VEN,EQ,           )
       WRITE(1,11)
       WRITE(1,12)
      READ(1,1) RC,VEN,EQ,DT
      ROTRAD=RC
       RC=RC/5280.
   13 FORMAT(20H VMAX,BA,BD,ALR       )
       WRITE(1,13)
       WRITE(1,12)
      READ(1,1) VMAX,BA,BD,ALR
      I=0
   14 FORMAT(21H THETA,VELOCITY,LEAVE)
       WRITE(1,14)
    2 I=I+1
       NCAR=I-1
       WRITE(1,12)
       IF(I.GT.50) GO TO 20
      READ(1,9) TI(I),VI(I),LI(I)
      TI(I)=TI(I)*PII
    9 FORMAT(2E10.5,I10)

    1 FORMAT(4F10.5)
   12 FORMAT(9X,1H.,9X,1H.,9X,1H.,9X,1H.)
      IF(TI(I).GE.0.) GO TO 2
   20 CONTINUE
      J=0
   15 FORMAT(20H ANGLE,QUEUE,LAMBDA   )
       WRITE(1,15)
    4 J=J+1
       M=J-1

       IF(J.GE.6) GO TO 1000
      WRITE(1,12)
      READ(1,1) TJ(J),XX(J),XL(J)
      TJ(J) =TJ(J)*PII
   45 FORMAT(I10)
      IF(TJ(J).GE.0.) GO TO 4
 1000 CALL ROTARY(ROTRAD,M,TJ,PNTR)
```

```
      NROADS=M
   10 J=0
    6 J=J+1
      IF(TJ(J).LT.0.) GO TO 7
      CALL QUEUE(XX(J),XL(J))
       QLINE(J)=IFIX(XX(J))
      GO TO 6
    7 CALL ENTER
      CALL VELOC
      CALL POSIT
      CALL ORDER
  100 FORMAT(6H NCARS,I5)
       NCARS=NCAR
       CALL CONVRT(ROTRAD,10.,TI)
      CALL CHANGE
      GO TO 10
      END
$1
      SUBROUTINE QUEUE  (X,XL)
       COMMON/E/JX
       CALL RAN(JX,PR)
      N=X+1.
       XF=1.0
      DO 25 I=1,N
       AII=FLOAT(I)
      XF=XF*AII
   25 CONTINUE
      PX=((XL**N)*EXP(-XL))/XF
      IF(PX.GE.PR) X=X+1.
   15 FORMAT(5H QUEU,2F10.5)
       RETURN
       END
$1
      SUBROUTINE ENTER
      DIMENSION TI(51),TK(51),VI(51),VK(51),LI(51),LK(51)
      DIMENSION TJ(6),XX(6),XL(6)
      COMMON/A/TI,TK,VI,VK,LI,LK
      COMMON/B/TJ,XX,XL
      COMMON/C/NCAR,RC,EQ,VEN,DT
      COMMON/D/ VMAX,BA,BD,ALR
       COMMON/E/JX
       COMMON/F/M
      J=1
      K=1
      I=1
   47 QJ=1.
       IF(TI(I).LT.0.) GO TO 17
    3 IF(TI(I).GT.TJ(J)) GO TO 2
   15 FORMAT(3I5,3F10.5)
       IF(VI(I).EQ.0.) GO TO 17
      QJ=((TJ(J)-TI(I))*RC)/(VI(I)*EQ)
   17  CALL  RAN(JX,PR)
  100 FORMAT(3H QJ,F10.5,F10.5)
      IF(QJ.LT.PR.OR.XX(J).LE.0.) GO TO 4
       IF(VI(I-1).LE.0..OR.I.LE.1) GO TO 37
         QJ=RC*(TI(I-1)-TJ(J))
       IF(QJ.LT.0.006) GO TO 4
   37 TK(K)=TJ(J)
       VK(K)=VEN
       XX(J)=XX(J)-1.
      CALL EXIT(J,L)
      LK(K)=L
      K=K+1
    4  IF(TI(I).LT.0.) GO TO 78
      CALL LEAVE(QJ,I,EQ,DT,LI(I),J)
   78 J=J+1
```

```
      IF(J.LE.M ) GO TO 47
      IF(K.GT.50) RETURN
        GO TO 70
    2 TK(K)=TI(I)
      VK(K)=VI(I)
      LK(K)=LI(I)
      K=K+1
      I=I+1
       IF(K.GE.51) RETURN
       GO TO 47
   70  IF(TI(I).GT.0.) GO TO 95
      NCAR =K-1
      TK(K)=-100.
      RETURN
   95 TK(K)=TI(I)
      LK(K)=LI(I)
       VK(K)=VI(I)
       K=K+1
       I=I+1
       GO TO 70
       END
$1
      SUBROUTINE EXIT (J,L)
      DIMENSION PL(5,5)
       COMMON/E/JX
       COMMON/F/M
       AV=0.20
      AB=0.20
       DO 17 K=1,M
       I=K+J
       IF(I.GT.M)I=I-M
       PL(J,K)=AV+AB
       AB=AV+AB
   17 CONTINUE

       CALL RAN(JX,PR)
      DO 1 I=1,M
      L= I

      IF(L.GT.M) L=L-M
      IF(PR.LE.PL(J,I) ) GO TO 2
    1 CONTINUE
    2 RETURN
      END
$1
      SUBROUTINE LEAVE(QJ,I,EQ,DT,LII,J)
    1 FORMAT(4H LII,2I5)
      IF(LII.NE.J) RETURN
      QL=(QJ*EQ)/DT
    2 FORMAT(3H QL,F8.4)
      IF(QL.GT.1.) RETURN
      I=I+1
      RETURN
      END
$1
      SUBROUTINE VELOC
      COMMON/C/NCAR,RC,EQ,VEN,DT
      COMMON/D/ VMAX,BA,BD,ALR
      COMMON/B/TJ,XX,XL
      COMMON/A/TI,TK,VI,VK,LI,LK
      DIMENSION TJ(6),XX(6),XL(6)
      DIMENSION TI(51),TK(51),VI(51),VK(51),LI(51),LK(51)
      TPI=6.28318
       IF (NCAR.LE.1) GO TO 7
       I=1
```

```
      TP=TK(NCAR)+TPI
    2 IF(TK(I).LT.0.) RETURN
      LI(I)=LK(I)
      VI(I)=(RC*(TP-TK(I)))/ALR
      IF(VI(I).GT.VMAX) VI(I)=VMAX
      DV=VI(I)-VK(I)
      IF(DV.LT.BA) VI(I)=VK(I)+BA
      IF(DV.GT.BD) VI(I)=VK(I)+BD
      IF(VI(I).LE.0.) VI(I)=0.
      TP=TK(I)
      I=I+1
      GO TO 2
    7 VI(1)=VK(1)
      LI(1)=LK(1)
      RETURN
      END
$1
      SUBROUTINE POSIT
      COMMON/C/NCAR,RC,EQ,VEN,DT
      COMMON/A/TI,TK,VI,VK,LI,LK
      DIMENSION TI(51),TK(51),VI(51),VK(51),LI(51),LK(51)
      IF(NCAR.LE.0)RETURN
      PI=6.28318
      NCAR=0.
      I=0
    2 I=I+1
      IF(I.GT.50) RETURN
      TI(I)=-100.
      IF(TK(I).LT.0.) RETURN
      TI(I)=TK(I)+(VI(I)*(DT/RC))
      IF(I.LE.1) GO TO 22
      IF(TI(I).GT.TI(I-1)) TI(I)=TI(I-1)
   22 NCAR=I
    1 FORMAT(4H TII,F8.4)
      GO TO 2
      END
$1
      SUBROUTINE SETRAN (JX)
      CALL RAN(JX,X)
      RETURN
      END
$1
      SUBROUTINE RAN(JX,YFL)
      IY=JX*899
      IF(IY)5,6,6
    5 IY=IY+32767 +1
    6 YFL=IY
      YFL=YFL/32767.
      JX=IY
      RETURN
      END
$1
      SUBROUTINE ORDER
      COMMON/A/TI,TK,VI,VK,LI,LK
      COMMON/C/NCAR,RC,EQ,VEN,DT
      DIMENSION TI(51),TK(51),VI(51),VK(51),LI(51),LK(51)
      IF(NCAR.LE.1)RETURN
      PI=6.28318
   22 IF(TI(1).GE.PI) TI(1)=TI(1)-PI
      IF(TI(1).LT.0.) TI(1)=0.
      IF(TI(1).GE.TI(2)) RETURN
      TK(NCAR) =TI(1)
      VK(NCAR)=VI(1)
      LK(NCAR)=LI(1)
      DO 7 J=2,NCAR
      TK(J-1) =TI(J)
```

```
      VK(J-1)=VI(J)
      LK(J-1)=LI(J)
    7 CONTINUE
      DO 76 J=1,NCAR
      TI(J)=TK(J)
      VI(J)=VK(J)
    2 FORMAT(4H TIJ,F8.4)
      LI(J)=LK(J)
   76 CONTINUE
      GO TO 22
      END
$1
      SUBROUTINE ROTARY (ROTRAD,NROADS,ANGLE,PNTR)
      DIMENSION ANGLE(6)
      INTEGER X(6),Y(6),PNTR(2)
C     DISPLAY COMMON
      COMMON /DSPCOM/DSPFLG,DSPMAX/DSPUVR/DSPSZE,DSPORG,DSPINK,DSPUSR,
     X   DSPFLT,DSPRET,DSPTIM,DSPTOT
      INTEGER DSPFLG,DSPMAX,DSPSZE,DSPORG,DSPINK,DSPUSR,DSPFLT,DSPRET,
     X   DSPTIM,DSPTOT
C
      COMMON/DSPBFR/DSPBUF(2000)
      INTEGER DSPBUF
C
C     LIGHT PEN COMMON
      COMMON /LPNCOM/LPNBLE,LPNBUF(50)
      INTEGER LPNBLE,LPNBUF,LPNWRT
      EQUIVALENCE (LPNBUF(1),LPNWRT)
C
C     CHANNEL COMMON
      COMMON /CHNCOM/ICHADR
C
C
      CALL CLRBUF(1)
      XI=ROTRAD*3.
      IX=IFIX(XI)
      CALL CIRCLE(512,512,IX)
      DO 1 I=1,NROADS
      RANG=ANGLE(I)
      CRANG=COS(RANG)
      SRANG=SIN(RANG)
      IXDS=XI*CRANG+512.
       X(I)=IXDS
      IYDS=XI*SRANG+512.
       Y(I)=IYDS
      CALL SETPT(X(I),Y(I))
      IXDS=(XI+200.)*CRANG+512.
       X(I)=IXDS
      IYDS=(XI+200.)*SRANG+512.
       Y(I)=IYDS
      CALL VECTOR(X(I),Y(I))
    1 CONTINUE
      PNTR(1)=6+NROADS*5
      DO 2 I=1,NROADS
      CALL SETPT(X(I),Y(I))
      CALL DSPCHR(3,2H00,2)
    2 CONTINUE
       PNTR(2)=PNTR(1)+NROADS*4
      RETURN
      END
$1
      SUBROUTINE CONVRT(ROTRAD,CARRAD,THETA)
      COMMON/XX/X(50)
      COMMON/YY/Y(50)
      COMMON/PP/PNTR(2)
      COMMON/NN/NCARS,NROADS
```

```
IEF285I   SYSOUT                                        SYSOUT
IEF285I   VOL SER NOS=          .
IEF285I   SYS71092.T000000.RP008.READ.R0000002          DELETED
IEF285I   VOL SER NOS=          .
```