NASA Technical Memorandum 81186

(NASA-TM-81186) A NEW ALGORITHM FOR HORIZONTAL CAPTURE TRAJECTORIES (NASA) 21 p HC A02/MF A01 CSCL 17G

N80-22297

Unclas
G3/04 46848

A New Algorithm for Horizontal Capture Trajectories

John D. McLean

March 1980

National Aeronautics and
Space Administration

NASA Technical Memorandum 81186

A New Algorithm for Horizontal Capture Trajectories

John D. McLean, Ames Research Center, Moffett Field, California

NASA
National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

A NEW ALGORITHM FOR HORIZONTAL CAPTURE TRAJECTORIES

John D. McLean

Ames Research Center

SUMMARY

A new algorithm, which will transfer an aircraft from a given initial position and heading to some final position and heading, has been developed for the on-board synthesis of horizontal flightpaths. These so-called capture trajectories have the basic form of a turn along a circular arc followed by a segment of straight flight and a subsequent circular turn. The two circular arcs may have different radii. The algorithm finds all such solutions possible, of which at least two and at most four exist, and selects the one with the minimum path length. Degenerate conditions in which one or more of the basic segments is missing are handled without difficulty. The solution to this problem is derived, and a Fortran listing of the algorithm is provided.

INTRODUCTION

Ames Research Center has been conducting studies involving use of on-board synthesis of flightpaths for area-navigation-equipped aircraft operating in the terminal area. The first step in the synthesis of a complete trajectory is the generation of the ground track. A useful criterion for generating the paths is to minimize the ground-track distance. The minimum-distance ground track, if flown at constant ground speed, has been shown by Erzberger and Lee (ref. 1) to consist of a series of straight lines and circular arcs. Pecsvaradi (ref. 2) developed algorithms for the on-board synthesis of horizontal flightpaths in that form. The flightpaths (e.g., as shown in fig. 1) consist of a fixed portion established by predetermined waypoints and a capture trajectory that will transfer the aircraft from some arbitrary initial position and heading to the desired final position and heading defined by the capture waypoint. The basic form of the capture path is a turn followed by a segment of straight flight and subsequently a final turn.

Flight-test experience has shown that the horizontal capture paths derived from the minimum distance criterion provide efficient and operationally acceptable terminal area trajectories for a variety of applications (refs. 3 and 4). The capture feature has been the subject of many favorable comments by pilots; however, it has been found that when the aircraft and capture waypoints are separated by less than four turning radii, the algorithm of reference 2, used in earlier experiments, often failed to find a capture path even though it is known (ref. 1) that at least two such paths always exist.

The new capture algorithm developed here always yields the minimum-distance horizontal path of the basic turn-straight-turn form, including degenerate cases in which one or more segments of the basic capture path are missing. The set of possible solutions for the minimum-distance capture path (ref. 2) is not restricted to the basic turn-straight-turn pattern but may also include a sequence of three circular arcs. The latter option applies to a fairly restricted region in which the initial and final points are close to each other; it is eliminated from the algorithm in the interest of simplicity (as was done in ref. 2). The following section describes the algorithm and presents the derivations of the equations used in it. The additional equations needed for implementation of the three-turn case are derived in appendix A, and a Fortran listing of the algorithm is given in appendix B.

ANALYSIS

Figure 2 is used to explain the problem and to define the variables. The turns are arcs of the circles shown in the figure and the straight portion of the trajectory must be a line tangent to both circles. Since the initial and final turns may be either clockwise or counterclockwise there are four possible combinations of turning directions, two with the initial and final turns in the same direction and two with the turns in opposite directions. Figure 3 illustrates one solution of each type. If a given pair of circles is entirely separate—that is, if no part of one circle lies within the other—it is possible to draw four tangent lines between the pair, but for only one of the four will the vector $\overline{D}$ along the tangent line from the initial to the final circle coincide with the direction of rotation at both tangent points, as shown in the figure. Thus, there are at most four real solutions of the turn-straight-turn form. The conditions for a real solution are derived next with the aid of figure 2.

For convenience, the vector $\overline{Q}$ from the center of the initial turn to the center of the final turn is horizontal in the figure; however, the coordinate system is arbitrary. Furthermore, the location of the centers of the circles relative to an earth-fixed coordinate system—and hence the direction and magnitude of $\overline{Q}$—depends on the direction of the turns. The constants and variables have been chosen so that the following discourse applies to all possible combinations of turning direction.

Figure 2(a) is for the case where both turns are in the same direction and the tangent vector $\overline{D}$ does not cross $\overline{Q}$; in figure 2(b) the turns are in opposite directions and $\overline{D}$ crosses $\overline{Q}$. Initially the aircraft is at (X_1,Y_1) in some inertial Cartesian coordinate system with heading H_1, defined as positive clockwise from the X-axis, and $\overline{v}_1$ is a unit vector in the direction of the velocity. The vector distance from (X_1,Y_1) to the center of the turn is given by $\overline{u}_1 R_1$, where R_1 is the radius of turn and $\overline{u}_1$ is a unit vector normal to $\overline{v}_1$ and positive to the right of $\overline{v}_1$. Therefore, the vector from (X_1,Y_1) to the center (XC_1,YC_1) is $R_1\overline{u}_1$ for a right turn and $-R_1\overline{u}_1$ for a left turn. The directions of turn are accounted for by writing the radius vector as $R_1S_1\overline{u}_1$ where $S_1 = +1.0$ for right turns and $S_1 = -1.0$ for left turns. Similarly the direction of the final turn is denoted by S_2.

The aircraft moves along the circle from (X_1, Y_1) to the tangent point (X_2, Y_2), which has a radius vector $R_1 S_1 \overline{u}_2$. Vector $\overline{D}$ is the tangent vector from (X_2, Y_2) at the end of the initial turn to (X_3, Y_3) at the beginning of the final turn. The radius vector at (X_3, Y_3) is $R_2 S_2 \overline{u}_3$, but since $\overline{u}_2$ and $\overline{u}_3$ must be normal to D, $\overline{u}_2 = \overline{u}_3$. Likewise, the headings, H_2 and H_3, at the two tangent points are equal. The final turn ends at (X_4, Y_4) with heading H_4 and radius vector $R_2 S_2 \overline{u}_4$.

Using this notation we can write

$$\overline{D} + R_2 \overline{u}_2 S_2 = R_1 \overline{u}_2 S_1 + \overline{Q}$$

or

$$\overline{Q} = \overline{D} + \overline{u}_2 \, (R_2 S_2 - R_1 S_1) \tag{1}$$

and therefore, since $\overline{D}$ and $\overline{u}_2$ are perpendicular,

$$D = \left[Q^2 - (R_2 S_2 - R_1 S_1)^2\right]^{1/2} \tag{2}$$

where by definition

$$Q = \left[(XC_2 - XC_1)^2 + (YC_2 - YC_1)^2\right]^{1/2} \tag{3}$$

It can be seen from equation (2) that no real solution exists if $Q < |R_2 S_2 - R_1 S_1|$. When the turns are in opposite directions $S_1 = -S_2$, and there is no real solution for $Q < (R_1 + R_2)$, that is, if the circles intersect. On the other hand, for rotations in the same direction $S_1 = S_2$, and a real solution exists unless $Q < |R_2 - R_1|$, that is, unless one circle lies entirely within the other.

If a real solution exists, then from the definition of the radius vectors

$$R_1 \overline{u}_1 S_1 = \begin{pmatrix} -R_1 S_1 \sin H_1 \\ R_1 S_1 \cos H_1 \end{pmatrix} \tag{4}$$

and

$$R_1 \overline{u}_1 S_1 = \begin{pmatrix} XC_1 - X_1 \\ YC_1 - Y_1 \end{pmatrix} \tag{5}$$

Equating equations (4) and (5) gives

$$\left.\begin{aligned} XC_1 &= X_1 - R_1 S_1 \sin H_1 \\ YC_1 &= Y_1 + R_1 S_1 \cos H_1 \end{aligned}\right\} \tag{6}$$

Similarly

$$\left.\begin{aligned} XC_2 &= X_4 - R_2S_2 \sin H_4 \\ YC_2 &= Y_4 + R_2S_2 \cos H_4 \end{aligned}\right\} \tag{7}$$

The radius vectors at the tangent points can be used in the same manner to compute the components of $\overline{X}_2$ and $\overline{X}_3$

$$X_2 = XC_1 + R_1S_1 \sin H_2 \tag{8a}$$

$$Y_2 = YC_1 - R_1S_1 \cos H_2 \tag{8b}$$

$$X_3 = XC_2 + R_2S_2 \sin H_2 \tag{9a}$$

$$Y_3 = YC_2 - R_2S_2 \cos H_2 \tag{9b}$$

Subtracting (8a) from (9a) and (8b) from (9b) gives the components of $\overline{D}$:

$$\left.\begin{aligned} X_3 - X_2 &= XC_2 - XC_1 + (R_2S_2 - R_1S_1) \sin H_2 \\ Y_3 - Y_2 &= YC_2 - YC_1 - (R_2S_2 - R_1S_1) \cos H_2 \end{aligned}\right\} \tag{10}$$

Another expression for the components of $\overline{D}$ is:

$$\left.\begin{aligned} X_3 - X_2 &= D \cos H_2 \\ Y_3 - Y_2 &= D \sin H_2 \end{aligned}\right\} \tag{11}$$

Equating (10) and (11) gives

$$\left.\begin{aligned} D \cos H_2 &= (XC_2 - XC_1) + (R_2S_2 - R_1S_1) \sin H_2 \\ D \sin H_2 &= (YC_2 - YC_1) - (R_2S_2 - R_1S_1) \cos H_2 \end{aligned}\right\} \tag{12}$$

Equations (12) can be solved for

$$\sin H_2 = \frac{(YC_2 - YC_1)D - (R_2S_2 - R_1S_1)\ (XC_2 - XC_1)}{Q^2} \tag{13}$$

$$\cos H_2 = \frac{(XC_2 - XC_1)D + (R_2S_2 - R_1S_1)\ (YC_2 - YC_1)}{Q^2} \tag{14}$$

$$\tan H_2 = \frac{(YC_2 - YC_1)D - (R_2S_2 - R_1S_1)\ (XC_2 - XC_1)}{(XC_2 - XC_1)D + (R_2S_2 - R_1S_1)\ (YC_2 - YC_1)} \tag{15}$$

Note that the quantities $(XC_2 - YC_1)/Q$ and $(YC_2 - YC_1)/Q$ are the sine and cosine, respectively, of the heading angle of $\overline{Q}$.

Equations (6)-(9) and (13)-(15) completely specify a capture trajectory for one combination of S_1 and S_2; however, the length of the trajectory is needed in order to determine which of the feasible trajectories gives the minimum distance. The first turn angle is

$$\left.\begin{aligned} &TR_1 = (H_2 - H_1) + 2\pi C_1 S_1 \\ &\text{where} \\ &C_1 = \begin{cases} 0 \text{ if } S_1 \ (H_2 - H_1) \geq 0 \\ 1 \text{ if } S_2 \ (H_2 - H_1) < 0 \end{cases} \end{aligned}\right\} \tag{16}$$

The second turn is

$$\left.\begin{aligned} &TR_2 = (H_4 - H_2) + 2\pi C_2 S_2 \\ &\text{where} \\ &C_2 = \begin{cases} 0 \text{ if } S_2 \ (H_4 - H_2) \geq 0 \\ 1 \text{ if } S_2 \ (H_4 - H_2) < 0 \end{cases} \end{aligned}\right\} \tag{17}$$

The lengths TD_i of the two arcs are

$$\begin{aligned} TD_1 &= R_1 |TR_1| \\ TD_2 &= R_2 |TR_2| \end{aligned} \tag{18}$$

and the length D of the straight segment is given by equation (2), and the total length of the capture path is

$$DT = D + TD_1 + TD_2 \tag{19}$$

It is possible to construct a "switching diagram" from which the flightpath yielding the minimum distance solution for any set of initial and final conditions is determined without finding all possible solutions. However, the switching diagrams are very complex, and from the computation standpoint it is simpler to solve for the length of all possible trajectories, as is done in the Fortran subroutine (appendix B).

APPLICATIONS AND EXAMPLE TRAJECTORIES

The algorithm, which was incorporated into an on-board guidance system for the Augmentor Wing Jet STOL Research Aircraft (AWJSRA), has been tested in simulations and during flight tests (ref. 4). In this application the fixed portion of the flightpath referred to in the introduction may include a number of predetermined waypoints (e.g., as shown in fig. 3). The subroutine is called sequentially to synthesize the path between successive pairs of waypoints working backward from the final one. The path between waypoints 3 and 4 is synthesized by entering the following input variables

$$X_1 = X(WP_3) \qquad X_4 = X(WP_4)$$

$$Y_1 = Y(WP_3) \qquad Y_4 = Y(WP_4)$$

$$R_1 = 0 \qquad R_2 = 0$$

$$H_1 = 0 \qquad H_4 = 0$$

Note that if $R_1 = 0$ then points (X_1,Y_1), (XC_1,YC_1), and (X_2,Y_2) are identical and the value of H_1 is arbitrary; the same is true of (X_4,Y_4), (XC_2,YC_2), and (X_3,Y_3) and the heading H_4. The resulting trajectory moves from (X_1,Y_1) to (X_4,Y_4) with heading

$$H_2 = \tan^{-1} \frac{Y_4 - Y_1}{X_4 - X_1}$$

which (if waypoints 3 and 4 are specified properly) is the runway heading. The path from waypoint 2 to waypoint 3 is synthesized by calling the subroutine with the input variables

$$X_1 = X(WP_2) \qquad X_4 = X(WP_3)$$

$$Y_1 = Y(WP_2) \qquad Y_4 = Y(WP_3)$$

$$H_1 = 0 \qquad H_4 = H(WP_3)$$

$$R_1 = 0 \qquad R_2 = R_2 \text{ (an input constant)}$$

As in the previous case, (X_1,Y_1), (XC_1,YC_1), and (X_2,Y_2) are identical and H_4 is the heading at waypoint 3 computed in the previous step. The heading of the tangent path and the coordinates of the beginning of the turn are returned in H_2 and (X_3,Y_3), respectively. The synthesis of the fixed path is carried out in this fashion for successive pairs of waypoints until the initial waypoint is reached. Then the algorithm is used to synthesize the capture trajectory to one of the fixed waypoints.

The photographs of the simulator cockpit display in figure 4 illustrate some close-in capture trajectories which would not have been found with the previous algorithm. If necessary for compatibility with ATC procedures, the algorithm could easily be modified to allow the pilot to select any of the possible capture trajectory patterns instead of the one giving the minimum path length.

CONCLUDING REMARKS

The solution of the turn-straight-turn capture flightpath problem developed here involves generating all possible solutions, of which there are at least two and at most four, and then selecting the one with the shortest path length. Because all possible solutions are obtained, criteria other than minimum path length could be used. For example, a nonminimum path length solution may be chosen in order to avoid a restricted airspace region. The closed-form equations determining the solutions are easily solved on currently available area-navigation computers or on a microprocessor. Other important flightpath problems in the terminal area can be approached via the capture solution. These include path stretching, holding patterns, and flight through a sequence of waypoints, all of which can be formulated as a sequence of turn-straight-turn capture problems.

APPENDIX A

DERIVATION OF THE THREE CIRCULAR ARC PATTERNS

It can be shown by geometrical construction that there are at most four turn-turn-turn type trajectories for the horizontal capture problem, namely, two each of the right-left-right and left-right-left patterns. However, it was shown in reference 1 that the middle turn must exceed π radians for a minimum-path-length trajectory. This requirement eliminates one trajectory of each pattern leaving at most two eligible three-turn trajectories.

The problem is illustrated in figure 5 for the right-left-right pattern. The vectors in the figure are defined as before, recognizing that in this case $S_2 = S_1$. In order to satisfy the requirement that the middle turn exceed π radians its center must be on the opposite side of $\overline{Q}$ from the straight segment of the right-straight-right solution shown for comparison. Furthermore, no three-arc solution exists for $Q > R_1 + R_2 + 2R_3$.

Let H_Q be the heading of $\overline{Q}$, defined in the text and define a unit vector $\overline{v}_0$ with heading angle, H_0 as follows

$$H_0 = H_Q + S_1 \frac{\pi}{2}$$

Then $\overline{v}_0$ is perpendicular to $\overline{Q}$ and points in the direction of flight where $\overline{Q}$ intersects the circle of the initial turn. From the law of cosines

$$\cos A' = \frac{Q^2 + (R_1 + R_3)^2 - (R_2 + R_3)^2}{2Q\ (R_1 + R_3)} \tag{A1}$$

$$\cos B' = \frac{Q^2 + (R_2 + R_3)^2 - (R_1 + R_3)^2}{2Q\ (R_1 + R_3)} \tag{A2}$$

The direction of turn is accounted for by defining

$$A = S_1 A' \tag{A3}$$

and

$$B = S_1 B' \tag{A4}$$

Using these definitions it can be seen from the figure that

$$\left.\begin{aligned} H_5 &= H_0 + A \\ H_6 &= H_0 - B + \pi \end{aligned}\right\} \tag{A5}$$

From the definition of H_0 and the equations derived in the text

$$\sin H_0 = \frac{S_1(YC_2 - YC_1)}{Q} \tag{A6}$$

$$\cos H_0 = \frac{-S_1(XC_2 - XC_1)}{Q} \tag{A7}$$

Equations (A5) can be used with double-angle trigonometric identities to obtain expressions for $\sin H_5$, $\cos H_5$, $\sin H_6$ and $\cos H_6$, in terms of H_0, A, and B. Changing appropriate subscripts in equations (8) and (9) in the text gives

$$\left.\begin{aligned} X_5 &= XC_1 + R_1S_1 \sin H_5 \\ Y_5 &= YC_1 - R_1S_1 \cos H_5 \end{aligned}\right\} \tag{A8}$$

$$\left.\begin{aligned} X_6 &= XC_2 + R_2S_1 \sin H_6 \\ Y_6 &= YC_2 - R_2S_1 \cos H_6 \end{aligned}\right\} \tag{A9}$$

The turn angles are calculated as follows

$$TR_1 = (H_5 - H_1) + 2\pi C_5 S_1 \tag{A10}$$

where

$$C_5 = \begin{cases} 0 \text{ if } (H_5 - H_1)S_1 \geq 0 \\ 1 \text{ if } (H_5 - H_1)S_1 < 0 \end{cases}$$

$$TR_2 = (H_4 - H_6) + 2\pi C_6 S_1 \tag{A11}$$

where

$$C_6 = \begin{cases} 0 \text{ if } (H_4 - H_6)S \geq 0 \\ 1 \text{ if } (H_4 - H_6)S < 0 \end{cases}$$

$$TR_3 = (H_6 - H_5) - 2\pi C_7 S_1 \tag{A12}$$

where

$$C_7 = \begin{cases} 0 \text{ if } (H_5 - H_6)S_1 \geq 0 \\ 1 \text{ if } (H_5 - H_6)S_1 < 0 \end{cases}$$

The total length of the trajectory is therefore

$$TR = R_1|TR_1| + R_2|TR_2| + R_3|TR_3| \tag{A13}$$

APPENDIX B

FORTRAN SUBROUTINE FOR CAPTURE ALGORITHM

```
      SUBROUTINE NEWPSI(X1,Y1,H1,R1 ,X4,Y4,R2 ,H4,X2,Y2,H2,
     1X3,Y3,TR1,TR2,TC1,TC2,D)
C
C   INPUTS-- INITIAL POSITION, HEADING, AND TURN RADIUS  X1,Y1,H1,R1
C            FINAL   POSITION, HEADING, AND TURN RADIUS  X4,Y4,H4,R2
C    IF FINAL HEADING ISNT SPECIFIED SET R2 TO 0.
C    H1 AND H4 MUST BE IN RANGE OF -PI TO PI
C
      TWOPI=6.28318531
    1 CONTINUE
      COSH4=COS(H4)
      SINH4=SIN(H4)
      COSH1=COS(H1)
      SINH1=SIN(H1)
C
C    DTMIN IS MINIMUM PATH LENGTH
      DTMIN=1.E+10
      KFM=0
      KFS=1
C
C    SELECT PATTERN TYPE
   12 CONTINUE
C
C    RIGHT-STRAIGHT-RIGHT
      S1=1.0
      S2=1.0
      GO TO 20
   14 CONTINUE
C
C    LEFT-STRAIGHT-RIGHT
      S1=-1.
      S2=1.
      KFS=2
      GO TO 20
   16 CONTINUE
C
C    RIGHT-STRAIGHT-LEFT
```

```
      S1=1.
      S2=-1.
      KFS=3
      GO TO 20
   19 CONTINUE
C
C     LEFT-STRAIGHT-LEFT
      S1=-1.
      S2=-1.
      KFS=4
   20 CONTINUE
C
C
C     FIND CENTERS OF TURNS
      SIGNR1=S1*R1
      SIGNR2=S2*R2
      XC1=X1-SIGNR1*SINH1
      YC1=Y1+SIGNR1*COSH1
      XC2=X4-SIGNR2*SINH4
      YC2=Y4+SIGNR2*COSH4
C
C     QSQ IS SQUARE OF DISTANCE BETWEEN CENTERS
      XN1=XC2-XC1
      YN1=YC2-YC1
      QSQ=XN1**2+YN1**2
C
C     DSQ IS SQUARE OF TANGENT PATH
      DSG=SIGNR1-SIGNR2
      DSQ=QSQ-DSG**2
C
C     IF DSQ<0  NO SOLUTION
      IF(DSQ.LT.0.) GO TO 36
      D=SQRT(DSQ)
C
C     H2 IS HEADING OF TANGENT VECTOR
      COSH2=(D*XN1-DSG*YN1)/QSQ
      SINH2=(DSG*XN1+D*YN1)/QSQ
      H2=ATAN2(SINH2,COSH2)
C
C     COMPUTE LENGTHS OF TURNS
      TR1=H2-H1
      IF(R1.EQ.0.) TR1=0.
      IF((TR1*S1).GE.0.) GO TO 24
      TR1=TR1+TWOPI*S1
```

```
   24 CONTINUE
      TR2=H4-H2
      IF(R2.EQ.0.) TR2=0.
      IF((TR2*S2).GE.0.) GO TO 30
      TR2=TR2+TWOPI*S2
   30 CONTINUE
      TD1=R1*ABS(TR1)
      TD2=R2*ABS(TR2)
C
C    DTOT IS TOTAL LENGTH OF PATH
      DTOT=TD1+TD2+D
      IF(KFM.EQ.7) GO TO 500
C
C     FINDS (UP TO ) FOUR SOLUTIONS AND USES SHORTEST
C
C
C    COMPARE DTOT WITH PREVIOUS MINIMUM
      IF(DTMIN.LE.DTOT) GO TO 36
      DTMIN=DTOT
      KFM=KFS
   36 CONTINUE
      GO TO (14,16,18,40),KFS
   40 CONTINUE
      KFS=KFM
      KFM=7
      GO TO (12,14,16,500),KFS
  500 CONTINUE
C
C    COMPUTE POSITIONS OF TANGENT POINTS
      X2=XC1+SIGNR1*SINH2
      Y2=YC1-SIGNR1*COSH2
      X3=XC2+SIGNR2*SINH2
      Y3=YC2-SIGNR2*COSH2
      RETURN
      END
```

REFERENCES

1. Erzberger, H.; and Lee, H. Q.: Optimum Horizontal Guidance Techniques for Aircraft. J. Aircraft, vol. 8, no. 2, Feb. 1971, pp. 95-101.

2. Pecsvaradi, Thomas: Four-Dimensional Guidance Algorithms for Aircraft in an Air Traffic Control Environment. NASA TN D-7829, 1975.

3. Lee, Homer Q.; Neuman, Frank; and Hardy, Gordon H.: 4-D Area Navigation System Description and Flight Test Results. NASA TN D-7874, 1975.

4. Erzberger, Heinz; and McLean, John D.: Fuel-Conservative Guidance System for Powered-Lift Aircraft. AIAA Guidance and Control Conference, August 6-8, 1979, Boulder, Colorado, pp. 79-1709. (Also available as NASA TM-78595.)

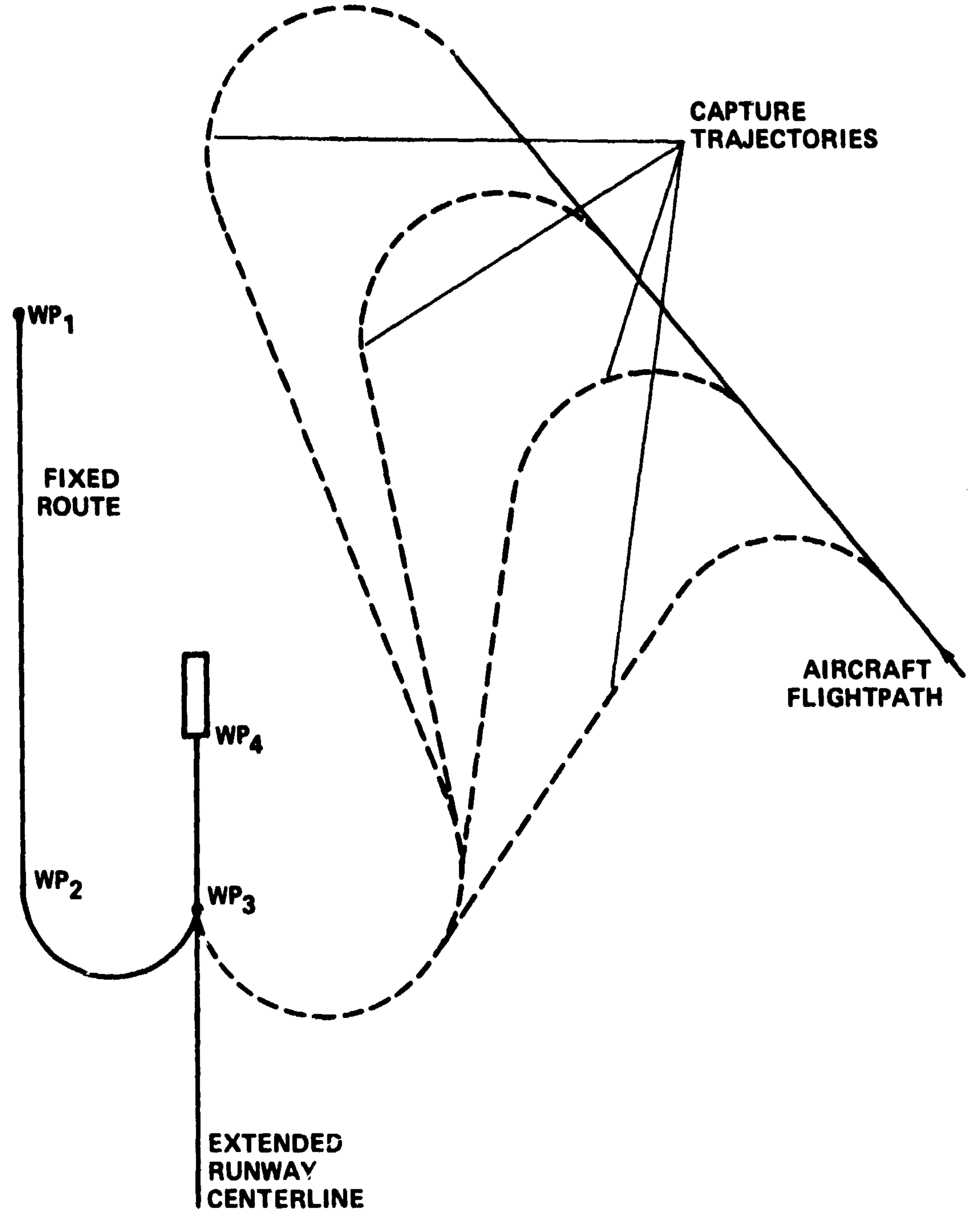
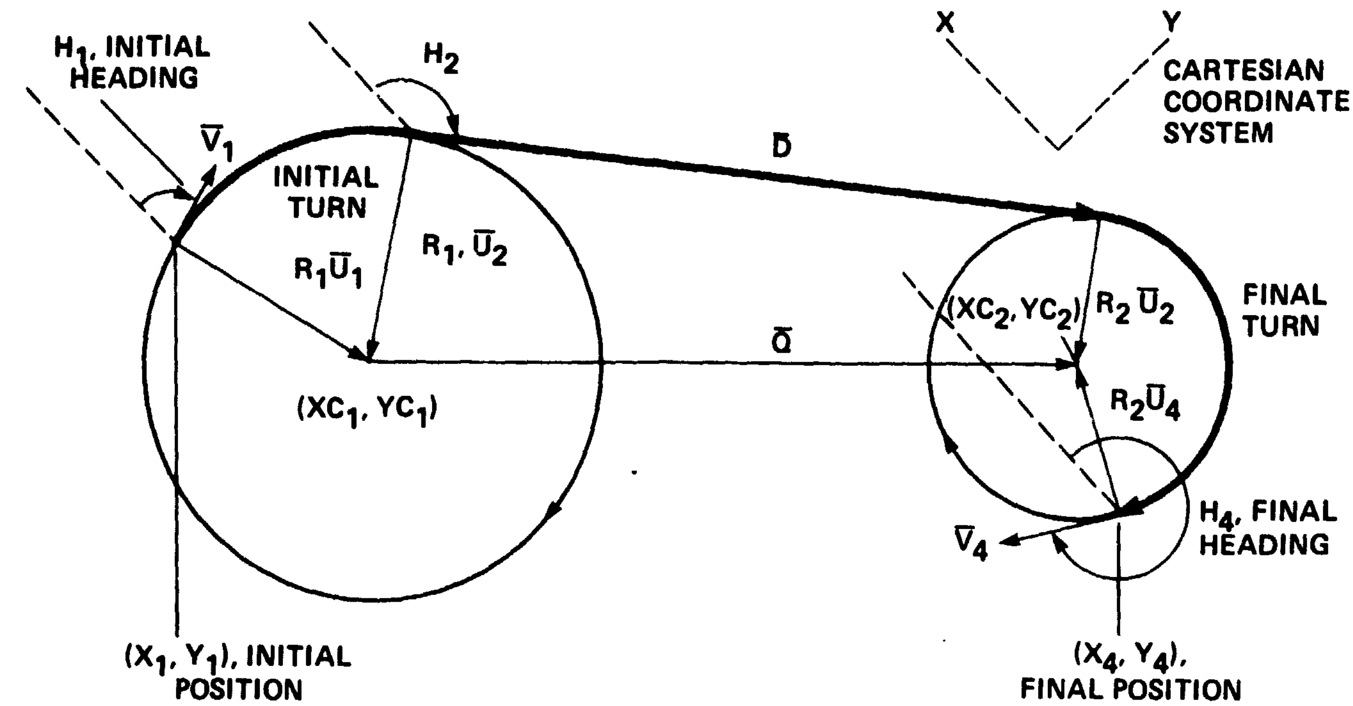
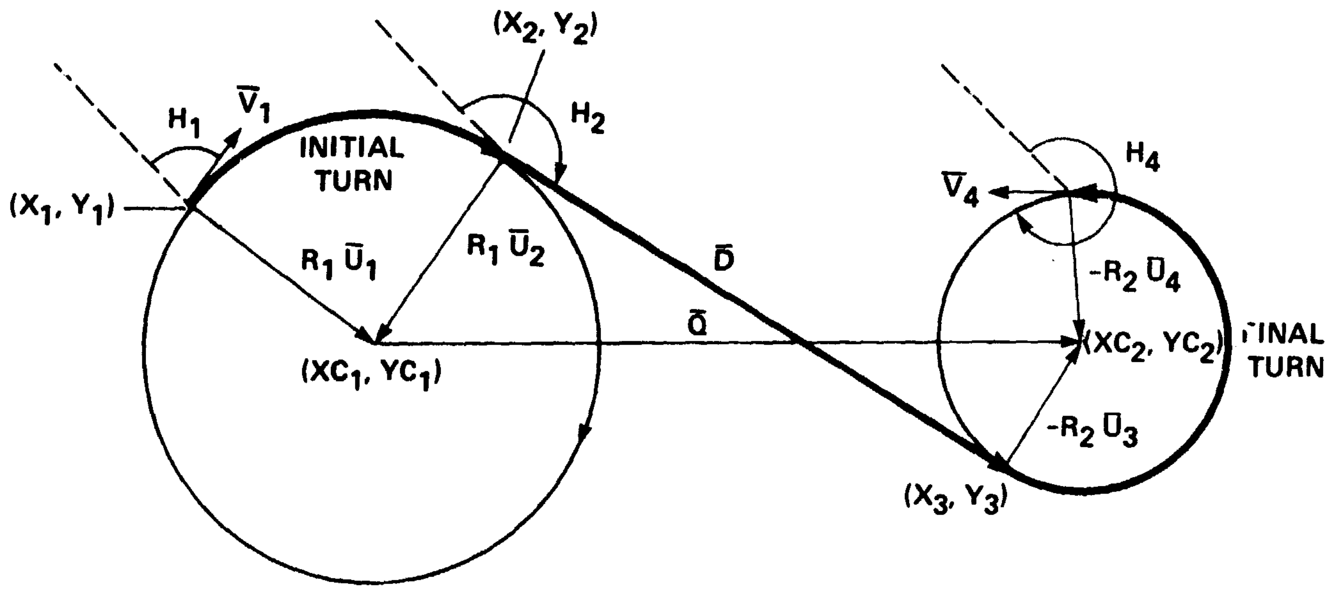


Figure 1.- Sequence of capture paths.



(a) BOTH TURNS CLOCKWISE



(b) SECOND TURN COUNTERCLOCKWISE

Figure 2.- Calculation of capture paths.

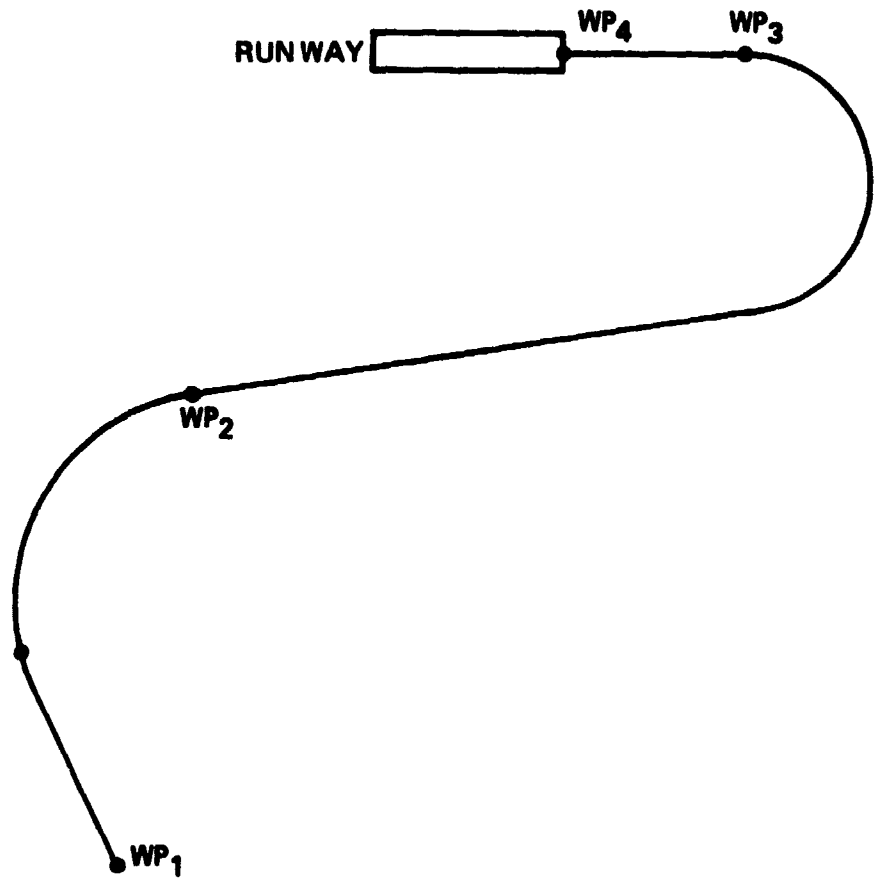
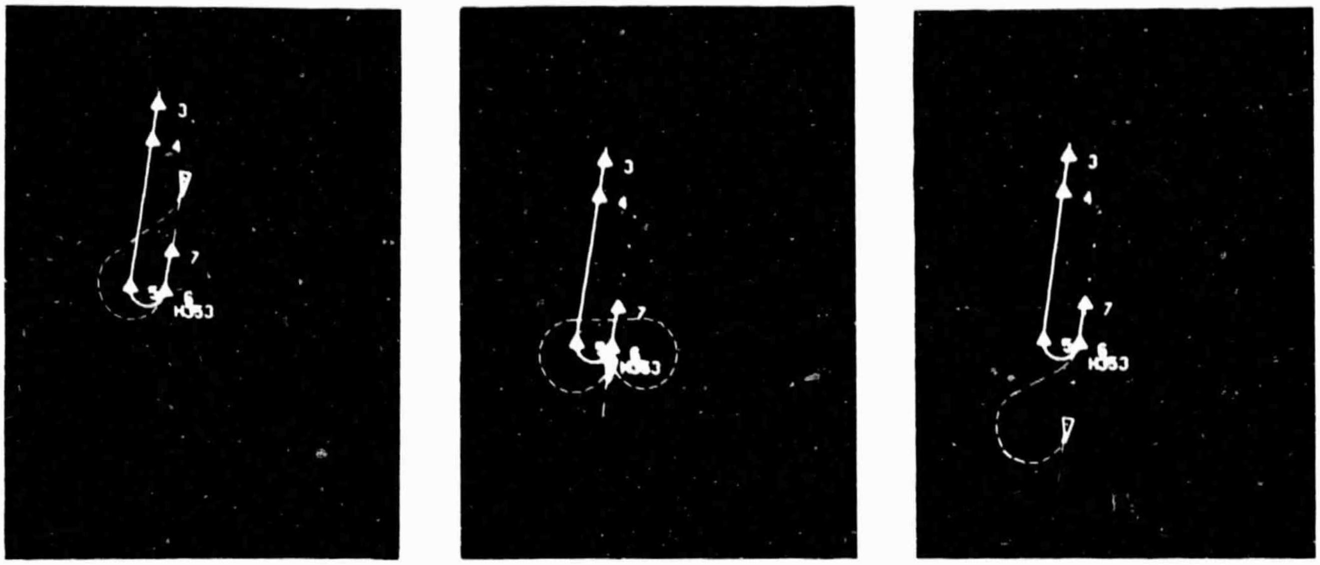
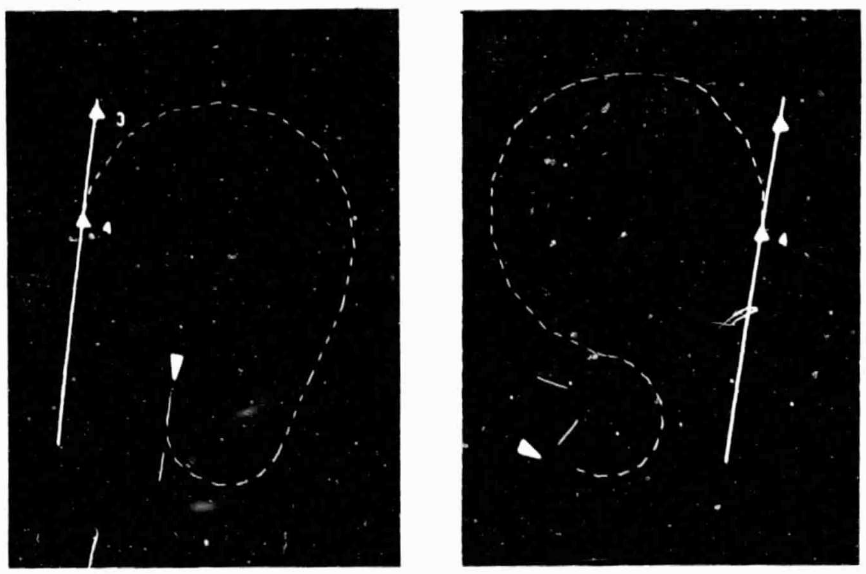


Figure 3.- Synthesis of fixed path through a sequence of waypoints.



(a) SEQUENCE OF CAPTURES OF WAYPOINT 6 FROM DOWNWIND APPROACH. TURN RADII 5000 ft, INITIAL SPEED = 140 knots, SPEED AT WAYPOINT 6 = 73 knots.



(b) TWO CAPTURES OF WAYPOINT 4. TURN RADII 6300 ft AND 3600 ft. INITIAL SPEED = 100 knots, SPEED AT WAYPOINT 4 = 140 knots. WIND = 20 knots.

Figure 4.- Photographs of capture trajectories on simulator cockpit display.

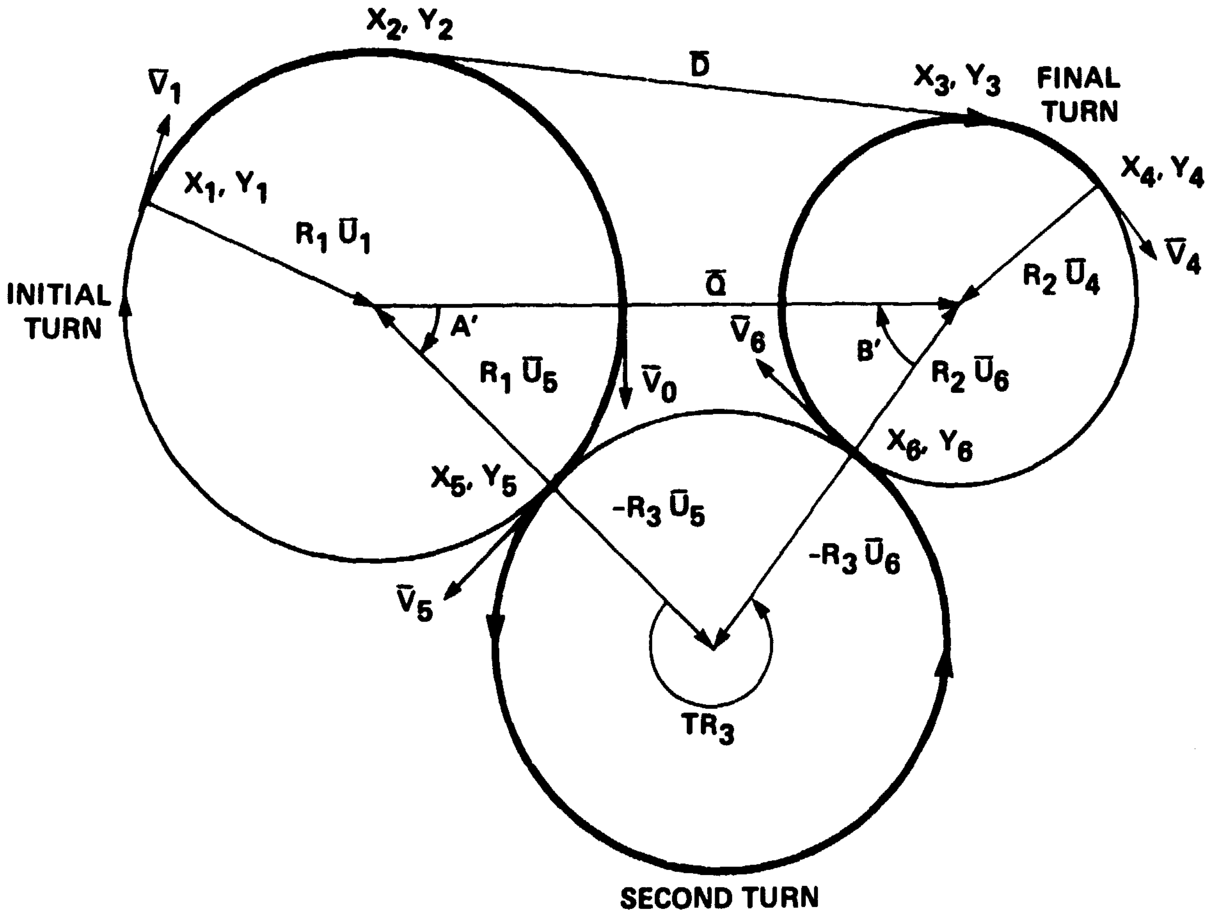


Figure 5.- Right-left-right pattern with corresponding right-straight-right case.