

## Aircraft's U-turn Maneuvers Applied in 3-Dimensional Scene Driven

Jiening Wang<sup>\*1</sup>, Jiang Qu<sup>2</sup>, Mei Dong<sup>3</sup>

<sup>1,2</sup>College of Air traffic Management, Civil Aviation University of China, Tianjin, China, 300300

<sup>3</sup>Northeast Regional Air Traffic Management Bureau, Civil Aviation Administration of China, Shenyang, China, 110043, e-mail:

\*Corresponding author, e-mail: wang\_jiening@aliyun.com<sup>\*1</sup>, tjqujiang@qq.com<sup>2</sup>, duikong2137@sina.com<sup>3</sup>

### Abstract

*U-turn maneuver is a special taxiing behavior on the ground. This paper studied the behavior and aimed to research the problem of position and attitude during the movement. From the perspective of airport tower control simulation, the calculation methods that the aircraft did U-turn maneuver under the nonholonomic constraints were proposed. According to the real geometrical constraints and kinematic equations, a method, suitable for simulation, deduction and calculation, was designed. At last, with the help of MATLAB, the validity of this method was confirmed. Compared with dynamics model, this method was easy to be computed and more suitable for the application to scene driven. This paper, moreover, designed the structural frame of 3D scene driven for the tower control simulation.*

**Keywords:** U-turn maneuver, nonholonomic constraints, geometrical constraints, kinematic equation, 3D scene driven

**Copyright © 2014 Institute of Advanced Engineering and Science. All rights reserved.**

### 1. Introduction

There are two modes when the aircraft is taxiing on the ground such as towing and self-propulsion. Pushback is a maneuver that aircraft closed its engines is pushed back from the gate to taxiway by the trailer. Reference [1] outlined a future picture, from the perspective of energy saving and emission reduction, which adopted the trailer technology from the process of pushback to the whole taxiing and pointed out the key issues related to the study. But it couldn't propose a scientific and effective computational method. In fact, the aircraft's pushback can be abstracted as a problem of N-Trailer movement [2, 3]. The key is to solve the problem with nonholonomic methods [4-6] and acquire aircraft attitude behavior under the path-followed constraints.

U-Turn is a maneuver that aircraft takes off in the opposite direction, which used to be concluded two methods. The first method is called edge-of-runway (EOR) maneuver; it is conducted by placing the aircraft parallel to the side of the runway and then initiating the turn at any point. The second method is called the center-of-runway (COR) maneuver; it is conducted by starting from the middle of the runway, traversing to the side of the runway at any angle, and then initiating the turn as soon as the nose gear reaches the edge of the runway. The COR method tends to allow for larger turn margins, due to a shift in the center of rotation toward the edge of the runway [7]. Reference [8] proposed a geometrical constrain when aircraft do the U-Turn maneuver. Reference [7] discussed the U-Turn maneuver, from the perspective of landing gear and aircraft dynamics, compute ground movement with application of Matlab Simulink and Kinetic model and provide the basis for the design of landing gear. The process of solving problem is complex and it requires a complete landing gear dynamics model, so it is difficult to be applied to 3D visual simulation driven.

Many different factors should be taken into account when comes to dynamics model of aircraft maneuver; such as aircraft type, load and handling characteristics and the like. Others, however, also have an effect on the dynamic characteristics of aircraft ground maneuver. Such as physical characteristics of the aerodrome, aircraft tire characteristics, weather conditions and the like. With application of dynamics model, it can be more accurately describe the behavior of aircraft ground maneuver and reflect the movement aircraft do in reality. However, solving the dynamic model of aircraft ground maneuver is a complex process, which employs the

establishment, including hydraulic power of landing gear, dynamic model of aircraft tires and aircraft aerodynamic model, which contains many nonlinear problems and most of all have singular solution. Currently, the main method is to establish corresponding to simulation model with the help of Matlab Simulink, and to support the solution of dynamic differential equation by using bifurcation theory [7]. This paper, mainly from the perspective of tower control, proposes a U-Turn Kinematic model, which, with respect to dynamic model, simplifies the solution process. The model is regarded as a rigid body and transforms Kinematic equations to Chained forms to solve this problem by constructing nonholonomic constrain equation. Above all, this paper presents a nonholonomic constraint equation and proposes trajectory calculation model with chained equations under the circumstance of U-Turn maneuver. All in all, it displays the realizable principle and steps of proposed methods.

## 2. Aircraft Maneuver Model

Consider the actual operation of airport tower control, the model introduces the following basic assumptions: 1) the aircraft can be regarded as a rigid body; 2) the center of the aircraft main landing gear tracks with taxiway center line; 3) aircraft pushed back with a constant turn rate; 4) slip does not occur in the aircraft movement.

### 2.1. The Basic Equations

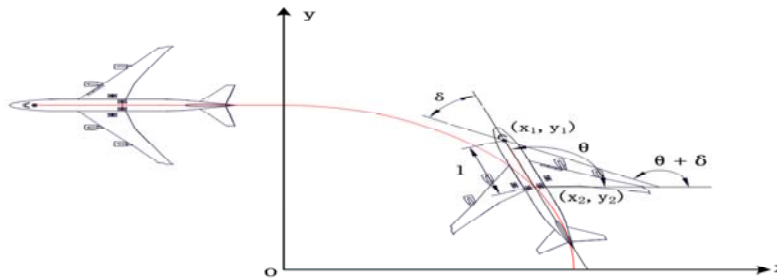


Figure 1. Aircraft Taxing Maneuver

In order to meet the assumption, as shown in Figure 1, we consider the maneuver constraints. The equations of aircraft maneuver constraint are:

$$\begin{aligned} \dot{x}_2 \sin \theta - \dot{y}_2 \cos \theta &= 0, & x_1 &= x_2 - l \cos \theta \\ \dot{x}_1 \sin(\theta + \delta) - \dot{y}_1 \cos(\theta + \delta) &= 0, & y_1 &= y_2 + l \sin \theta \end{aligned} \quad (1)$$

Where  $x_1, y_1$  and  $x_2, y_2$  are center position of aircraft's nose gear and main landing gear in the Cartesian coordinates respectively.  $\theta$  is the angle measured from fuselage to the X-axis.  $\delta$  is the steering angle.  $l$  is the wheel base, distance from nose gear to main gears. Obviously, the aircraft maneuver constraint is obtained as:

$$\begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \\ \dot{\theta} \\ \dot{\delta} \end{bmatrix} = \begin{bmatrix} \cos \theta \\ \sin \theta \\ \frac{1}{l} \tan \delta \\ 0 \end{bmatrix} v_1 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v_2. \quad (2)$$

There is a model singularity at  $\delta = \pm \pi/2$ , where the first vector field has a discontinuity. This corresponds to the aircraft becoming jammed when the front wheel is normal to the longitudinal axis of the body. However, the importance of this singularity is limited, due to the restricted range of the steering angle  $\delta$  in most practical cases.

From the Equation (2), we can regard it as a nonlinear system  $\dot{q} = G(q)v$ , which represents the kinematic model of the aircraft. Here,  $q$  is the  $n$ -vector of aircraft generalized coordinates,  $v$  is the  $m$ -vector of input velocities ( $m < n$ ), and the columns  $g_i (i = 1, \dots, m)$  of matrix  $G$  are smooth vector fields. As satisfying the constraint of assumption 4), it cannot be solved by integration. So the aircraft maneuver is typical problem solved by nonholonomic forms.

## 2.2. Aircraft Maneuver Soluble Model

For the nonholonomic constraint model of Equation (2), this paper employs chained forms [9] to solve the problem. The chained model of two inputs without sliding is listed as follows:

$$\begin{cases} \dot{\zeta}_1 = u_1 \\ \dot{\zeta}_2 = u_2 \\ \dot{\zeta}_3 = \zeta_2 u_1 \\ \dots \\ \dot{\zeta}_n = \zeta_{n-1} u_1 \end{cases} \quad (3)$$

In fact, nonholonomic movement can be, according to Lie algebra bracket operator, extended to  $n$ -dimensional space, resulting in employing the chained representation model [10, 11]. The equation is displayed as follows:

$$v_2 = -3u_1 \sin \theta \sin^2 \delta / (l \cos^2 \theta) + u_2 l \cos^3 \theta \cos^2 \delta. \quad (4)$$

$$v_1 = u_1 / \cos \theta. \quad (5)$$

$$\zeta_1 = x_2 \quad \zeta_2 = \frac{1}{l} \sec^3 \theta \tan \delta \quad \zeta_3 = \tan \theta \quad \zeta_4 = y_2 \quad \text{and} \quad u_1 = \dot{\zeta}_1, \quad u_2 = \dot{\zeta}_2. \quad (6)$$

Where  $v_1$  is the driving velocity of aircraft movement, and  $v_2$  is the steering velocity input.

Equation (4) satisfies the representation of chained model of Equation (3). According to the aircraft maneuver, given in Figure 1, and assumption (2), we get the coordinates  $x_2(t)$  and  $y_2(t)$  that the position of center of main landing gear of the aircraft at the time  $t$ , which can be expressed as the chained model:

$$\dot{\zeta}_1 = u_1 \quad \dot{\zeta}_2 = u_2 \quad \dot{\zeta}_3 = \zeta_2 u_1 \quad \dot{\zeta}_4 = \zeta_3 u_1. \quad (7)$$

Obviously:

$$\zeta_1(t) = x_2(t). \quad (8)$$

$$\zeta_2(t) = (\ddot{y}_2(t)\dot{x}_2(t) - \dot{y}_2(t)\ddot{x}_2(t)) / \dot{x}_2^3(t). \quad (9)$$

$$\zeta_3(t) = \dot{y}_2(t) / \dot{x}_2(t). \quad (10)$$

$$\zeta_4(t) = y_2(t). \quad (11)$$

$$u_1(t) = \dot{x}_2(t). \quad (12)$$

$$u_2(t) = (\ddot{y}_2(t)\dot{x}_2^2(t) - \ddot{x}_2(t)\dot{y}_2(t)\dot{x}_2(t) - 3\ddot{y}_2(t)\dot{x}_2(t)\ddot{x}_2(t) + 3\dot{y}_2(t)\dot{x}_2^2(t)) / \dot{x}_2^4(t). \quad (13)$$

$(x_2, y_2, v_1, v_2, \theta, \delta)$  is the center of main landing gear of movement parameters, which can be employed. As an open-loop control, aircraft maneuver trajectories is calculated dynamically as follows:

$$\begin{bmatrix} x_1(t + \Delta t) \\ y_1(t + \Delta t) \\ x_2(t + \Delta t) \\ y_2(t + \Delta t) \end{bmatrix} = \begin{bmatrix} x_1(t) + v_1(t)\Delta t \cos(\theta(t) + \delta(t)) / \cos(\delta(t)) \\ y_1(t) + v_1(t)\Delta t \sin(\theta(t) + \delta(t)) / \cos(\delta(t)) \\ x_2(t) + v_1(t)\Delta t \cos(\theta(t)) \\ y_2(t) + v_1(t)\Delta t \sin(\theta(t)) \end{bmatrix}. \quad (14)$$

Where  $x_1(t)$ ,  $y_1(t)$  and  $x_2(t)$ ,  $y_2(t)$  are the coordinate value of nose gear and center of main gears at the time  $t$  respectively. Then  $\theta(t)$ ,  $\delta(t)$ ,  $v_1(t)$  are also the corresponding values at the time  $t$ .

### 3. U-Turn Maneuver Model

This paper mainly discusses the center-of-runway (COR) maneuver. The aircraft, for the purpose of doing U-Turn maneuver, is manipulated by the following five steps [7]:

- (1) To align the aircraft with the edge of the runway.
- (2) To apply all the thrust in the process of aircraft's movement.
- (3) To increase aircraft's thrust on the outboard while decreasing the thrust on the inboard engines.
- (4) To apply the inboard brake pedal.
- (5) To adjust the nose gear steering angle to steady state.

As shown in Figure 2, from the perspective of simulation, U-Turn maneuver can be divided into three stages. The first phase (A-B-C) is that the aircraft aligns with the runway edge. The second phase (C-D) is that aircraft does the turn movement. The third phase (D-E-F) is that aircraft aligns with centerline after turning.

There are some geometric constraints and movement constraints when aircraft does U-Turn on the runway. The main geometric constraints are listed as follows:

Make sure that the outermost main gear is not beyond the runway edge, when aircraft taxis aligning with the runway edge. Namely,

$$(l_m + r_{out})(1 - \cos \eta) > L_{min}. \quad (15)$$

Where  $l_m$  is the track width, distance between outermost wheels of left and right outer gears, measured from outer wheel-plane.  $r_{out}$  is the distance measured from outermost main gears to turn-center.  $\eta$  is the angle measured from fuselage to centerline of the runway, and  $L_{min}$  is the minimum safety distance measured from main gears near the edge of runway to runway edge.

Make sure that the nose gear is not beyond the runway edge when aircraft does turn maneuver. Namely,

$$l_m + r_{out} + r_{thead} < W_r. \quad (16)$$

Where  $r_{thead}$  is the distance measured from nose gear to turn-center, and  $W_r$  is the width of runway.

As shown in Figure 2, it depicts the turning radius and turning center when aircraft does turn maneuver.

$$r_{out} = l \tan(\pi / 2 - \delta) + l_m / 2. \quad (17)$$

$$r_{mid} = l \tan(\pi / 2 - \delta). \quad (18)$$

$$r_{thead} = l / \cos(\pi / 2 - \delta) . \tag{19}$$

Where  $r_{imid}$  is the distance measured from turn-center to center of main gears. The maneuver of aircraft is depicted in Figure 3, the equation are:

$$\begin{cases} \dot{x}_1 = v_1 \cos \theta_1 \\ \dot{y}_1 = v_1 \sin \theta_1 \\ \dot{\delta} = v_2 \\ \dot{x}_2 = v_1 \cos \theta_2 \cos(\delta) \cdot \\ \dot{y}_2 = v_1 \sin \theta_2 \cos(\delta) \\ \dot{\theta}_2 = \frac{v_1}{l} \sin(\delta) \end{cases} \tag{20}$$

Where  $\theta_1$  and  $\theta_2$  are the angle measured from direction of velocity to the X-axis and the angle measured from fuselage to the X-axis respectively.

The parameters of taxiing attitude can be expressed by  $[\dot{x}_1, \dot{y}_1, \dot{x}_2, \dot{y}_2, \dot{\delta}, \dot{\theta}_2]$ . We can only be considered the velocity in the calculation because the direction of speed is determined by geometrical constraint of the path.

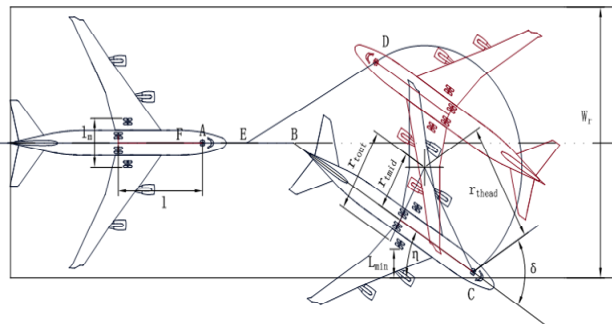


Figure 2. The Process Of U-Turn Maneuver

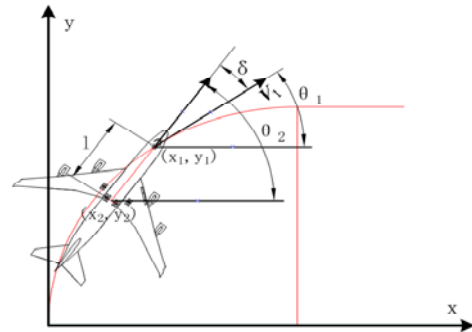


Figure 3. Constraints of U-turn Maneuver

According to the requirement of aircraft taxiing on the ground, its nose gear should taxi by following the centerline of turn-taxiing. This paper establishes a parameter model of aircraft taxiing attitude in accordance with open-loop control, which forms a state control that can drive the scene operating. Obviously, while the aircraft operations, it can be wrote:

$$\begin{bmatrix} x_1(t + \Delta t) \\ y_1(t + \Delta t) \\ x_2(t + \Delta t) \\ y_2(t + \Delta t) \end{bmatrix} = \begin{bmatrix} x_1(t) + v_1(t)\Delta t \cos(\theta_1(t)) \\ y_1(t) + v_1(t)\Delta t \sin(\theta_1(t)) \\ x_2(t) + v_1(t)\Delta t \cos(\theta_2(t)) \cos(\delta(t)) \\ y_2(t) + v_1(t)\Delta t \sin(\theta_2(t)) \cos(\delta(t)) \end{bmatrix} . \tag{21}$$

In the phase of straight to turn,  $\delta(t)$  is [12]:

$$\begin{aligned} \delta(t) &= \delta_f [1 - e^{-(0.336 - 1.053 R_n) \frac{v}{R} t}] \\ \delta_f &= \cos^{-1} \left( \frac{\sqrt{R^2 - l^2}}{R} \right) \quad R = \frac{r}{l} \end{aligned} \tag{22}$$

Where  $\delta_f$  is the final steady-state steering angle for a circle with a specific radius.  $R$  is radius of turn measured from the turn-center to the bottom of the nose gear strut, normalized to the wheel base.  $r$  is radius of turn measured from turn-center to bottom of the nose gear strut, and  $V$  is magnitude of velocity at nose gear, normalized to wheel base units.

To ensure the turn- straight smooth conversion of  $\delta(t)$ , this paper adopts hyperbolic functions to reconcile the change of  $\delta(t)$ , as follows:

$$\delta(t) = \frac{\delta_{im}}{2} \left[ 1 + \tanh\left(\frac{\delta_r}{\delta_{im}}(t_d - 2t)\right) \right]. \tag{23}$$

Where  $\delta_r$  is the fixed maximum steering rate.  $\delta_{im}$  is the value of  $\delta(t)$  when the aircraft taxis from turn to straight line, and  $t_d$  is the duration that  $\delta(t)$  changes from  $\delta_{im}$  to zero(  $t_d = k \delta_{im} / \delta_r$  ).

The algorithm is described as follows:

1) Firstly, according to geometry parameters, steering angle and digging angle of aircraft, we can get the appropriate parameter such as  $\delta, r_{out}, r_{mid}$  and  $r_{thead}$  based on Equation (15)-(19) so as to determine the constraint parameter when aircraft does U-Turn maneuver.

2) According to the iteration of Equation (21), (22) that taxiing from straight line to turn, we can get the movement parameters.

3) According to the iteration of Equation (21), (23) that taxiing from turn to straight line, we can get the movement parameters.

We can conclude that, during the U-turn maneuver, adjust the value of k to smooth the curve in the 3 phases (straight-turn, straight-turn-straight, turn-straight).

#### 4. Visual Simulation Model

##### 4.1. Scene Graph Generation

The visual simulation of airport control tower is a distributed multi-channel system, which updates the multi-channel visual synchronous drawing in real-time through receiving the instructions from the central control units. This paper builds scene graph with *OpenSceneGraph* (OSG) [13]. The aircraft's node structure is depicted in scene graph and shown in Figure 4. We can operate the *AircraftGeomNode* dynamically through operating the *TransFormNode* and the *RotateTransNode*. Where *RotateTransNode* is controlled by quaternions. That is to say, `osg::Matrixf::rotate(osg::Quat(angle,axis))` controls its node.

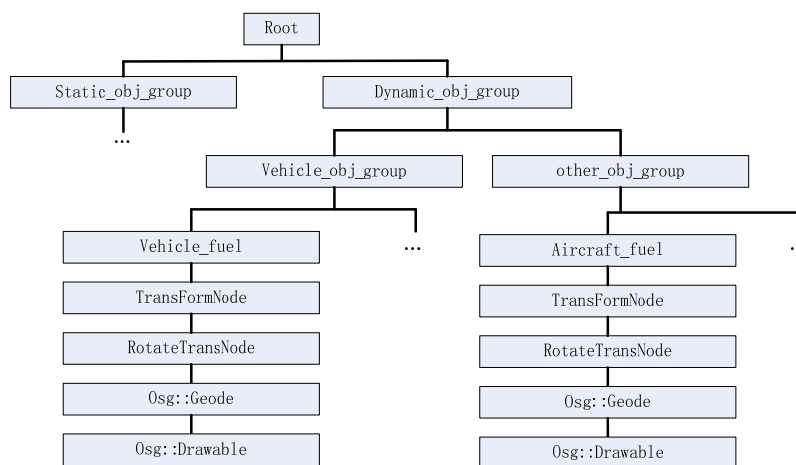


Figure 4. Scene Graph Structure of OSG

Above all, to express the taxiing manipulation parameters, we build the similar quaternions for each aircraft:  $behavior_i(a,s)$ . Where  $\alpha = I(t)I(t + \Delta t)$ ,  $I$  is vector that the center of aircraft's main gear points to the center of nose gear.  $S$  is position vector of aircraft's nose gear.

#### 4.2. Scene Driven design

In the phases of implementation, considering the aircrafts' movement on the ground with inter-independence, the multi-thread technology is adopted in the simulation. That is to say, according to simulate flight plan, we can establish an independent thread for each aircraft. Module structure is shown in Figure 5, which includes 3 parts: Aircraft movement behavior computing threads and management module, threads sharing data management module and scene driven data sending threads module.

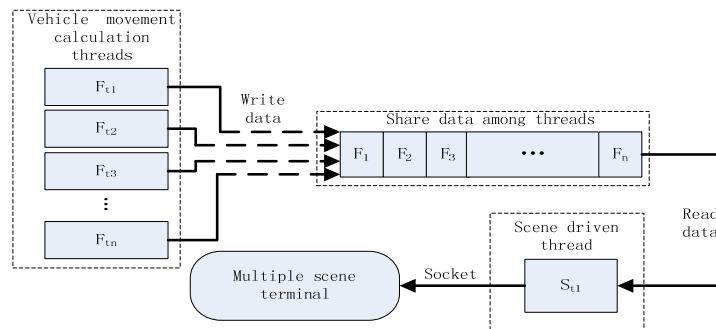


Figure 5. Implement Modular Architecture for Scene Driven

At first, initialization and establish computing threads of aircraft movement, we apply the proposed methods to compute the movement behavior and then write the results into threads sharing data. According to the requirements of scene update, scene driven data sending threads module read the threads sharing data as a one-time work and send to the scene terminal, through the Socket, to update the scene. In the three models, Where  $F_{t_1}, \dots, F_{t_n}$  mean  $n$  aircrafts' computing threads.  $F_1, \dots, F_n$  mean computing threads of  $n$  aircrafts' position and attitude.  $S_{t_i}$  mean scene driven data sending threads.

Aircraft movement behavior computing threads and management module is mainly responsible for creating and managing the calculation of aircrafts' movement behavior. Firstly, to form a uniform data registration model with *QVariant* features, we can define, through *Q\_DECLARE\_METATYPE (calEnvironment)*, calculation-required environmental data, which include the flight number, type, movement path, operation behavior, initial position/attitude and the like. Secondly, set up the following computing thread class:

```
class FlightBehaviorCal : public QThread
{
    Q_OBJECT
public:
    Explicit FlightBehaviorCal(const QVariant & env);
public slots:
    //response function of control instructions and
    status operation function of motion behavior //
signals:
    //information function of running status//
private:
    void run();
    //computing threads allowed and required environmental variable data//
}
```

The core is to implement the proposed calculation methods in the `run()`, and write the result of `behaviori` into the threads sharing data. Thirdly, the threads management module `QMap<QString, QPointer<flightBehaviorCal > > flightBehaviorCals` are built to realize the management to threads build, start and stop.

Threads sharing data management module is mainly responsible for storage and management of position and attitude data, which are generated in the flight simulation. The structure of storage objects are `QMap<QString, QPointer<behavior>>`, which build a fixed written structure for each flight. According to the `QReadLocker`, `QWriteLocker` and `QReadWriteLock`, it is mainly solve the multi-thread mutex problem.

Scene driven data sending threads module is mainly responsible for regularly locking the threads sharing data and reading all the sharing data. It can be sent to scene terminal by Socket and drive the scene to be updated. Its structure is similar to the computing threads of aircraft movement behavior, but it is prior to the aircrafts'.

## 5. Mathematical Simulation and Analysis

In order to prove the calculation of the U-Turn behavior proposed in this paper, we make a corresponding calculation of taxiing behavior using A320 aircraft. The U-Turn stimulation contains all the processes that aircraft starts to taxi from runway centerline to runway centerline after doing turn-maneuver, which mainly calculate the position coordinates in the process of turning of nose gear and center of main gears. In this paper, the wheel base of A320 is given to 12.64m, with 4m/s of turn rate.

The result of U-Turn simulation is shown in Figure 6. To smooth the curve, we should adjust the value of  $k$ . As a result, we get  $k_1=3.76$ ,  $k_2=1.69$ ,  $k_3=10.00$ . The red line is the trajectory of center of nose gear. Blue one is the main gears'. The variation of steering angle, such as A320, is shown in Figure 7 during U-Turn maneuver on the runway.

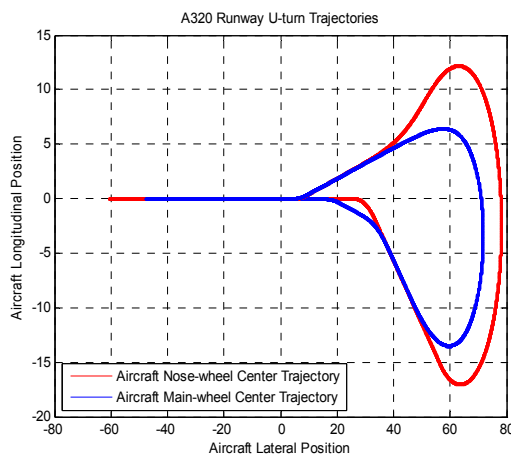


Figure 6. Numerical Simulation of U-turn Maneuver

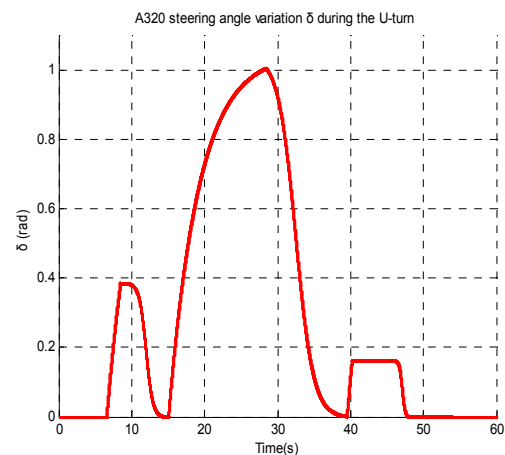


Figure 7. Variation of Steering Angle

## 6. Conclusion

a) For the performance of computational requirements of tower control visual simulation in real-time, we propose the kinematic model based on nonholonomic constraints and the computational method of U-Turn maneuver behavior. Compared to the dynamic model, the method is simple, easy to adjust, with small amount of calculation and more relevant to the performance requirements of visual simulation.

b) In this paper, with detailed U-Turn maneuver concerned, we propose the computational method of the whole movement. Through the simulation, it proves that the method is validity and correctness.



c) The method proposed in this paper, with aircraft-oriented computational independence, is suitable for parallel calculation in multi-threaded mode. We should better enhance the state-design of ground maneuver in the future research and further improve the computational efficiency based on the different maneuvers behavior and operational phases.

### Acknowledgements

This research was supported by Tianjin Municipal Science and Technology Project (No. 11ZCKFGX04200) and Fundamental Research Funds for the Central Universities (No. 3122013P008)

### References

- [1] Coetzee EB, et al. Analysis of medium-speed runway exit maneuvers. *AIAA-2010-1553*. 2010.
- [2] Michalek M. *Tracking control strategy for the Standard N-trailer mobile robot –geometrically motivated approach*. Bukowy Dworsk: 8th Robot Motion and Control International Workshop. 2011.
- [3] Lee JH, et al. A passive multiple trailer system with off-axle hitching. *International Journal of Control Automation and Systems*. 2004; 2(3): 289-297.
- [4] Wang Yue-Chao, Jing Xing-Jian. Steering and control of nonholonomic wheeled mobile robots using artificial fields. *Acta Automatica Sinica*. 2002; 28(5): 777-783.
- [5] Tanner HG, et al. Nonholonomic navigation and control of cooperating mobile manipulators. *IEEE Transactions on Robotics and Automation*. 2003; 19(1): 53-64.
- [6] Bushnell LG, et al. *Steering three-input chained form nonholonomic systems using sinusoids: the fire truck example*. Groningen: Proceedings of the European Control Conference. 1993.
- [7] Coetzee E, et al. Application of bifurcation methods to the prediction of low-speed aircraft ground performance. *Journal of Aircraft*. 2010; 47(4): 1248-1255.
- [8] Chai S, Mason W. Landing gear integration in aircraft conceptual design. *AIAA-1996-4038*. 1996.
- [9] Murray RM, Sastry SS. *Steering nonholonomic systems in chained form*. Brighton: Proceedings of the 30th Conference on Decision and Control. 1991:1121-1126
- [10] Duleba I, Khefifi W. *A lie algebraic method of motion planning for driftless nonholonomic systems*. Brussels: Fifth International Workshop on Robot Motion and Control. 2005.
- [11] YOU Liang-Gu. An exploration of orientation representation by lie algebra for robotic application. *IEEE Transaction on System, Man and Cybernetics*. 1990; 20(1): 243-248.
- [12] Etienne Coetzee, et al. Analysis of Medium-Speed Runway Exit Manoeuvres. *AIAA Modeling and Simulation Technologies*. 2010.
- [13] Wang Rui, Qian Xue Lei. *Openscenegraph 3D rendering engine design and practice*, BeiJing: TsingHua University Press. 2009.