International Journal of Multimedia and Ubiquitous Engineering
Vol. 7, No. 2, April, 2012

A Dynamics Model of Rotor Blades for Real-time Helicopter Simulation

Suwan Park
IS Lab, TSonNet Co., Ltd.
Daejeon, Korea
ipsuwan@gmail.com

Nakhoon Baek *†
Kyungpook Nat'l Univ.
Daegu, Korea
oceancru@gmail.com
(corresponding author)

Kwan-Woo Ryu
Kyungpook Nat'l Univ.
Daegu, Korea
kwryu@knu.ac.kr

Abstract

We present a dynamics model of rotor blades for real-time helicopter simulation. Collisions between the air flow and the moving blades make helicopters fly. In aerodynamics, or even in computer simulations, they precisely analyzed the collisions between the fluid(air) and the solid object(blades), and calculated the differential equations from the collisions. Thus, it was hard for them to generate real-time helicopter motions due to massive computations for calculating the equations. In this paper, we start from a geometric model of rotor blades, which reflects the characteristics of real world blades due to the various factors from helicopter aerodynamics, although some factors should be simplified to show real-time behaviors. Based on this geometric model, we present a dynamics model for calculating the forces due to the rotor blades colliding with air flows. Our dynamics model interprets the collisions between the fluid and the solid objects as the action-reaction forces, as originally Newton did. Finally, we present the force equations suitable for the existing rigid-body simulation systems, instead of fluid-dynamics equations. We implement a prototype system for helicopter motions, and it shows sufficient real-time processing behavior with ordinary PC's.

1 Introduction

In this paper, we present a dynamics model for the real-time helicopter simulation. Helicopters fly up to the sky through rotating the rotor blades(rigid bodies) in the air(fluid). Thus, to simulate helicopter motions, we need to handle interactions between rigid bodies and fluid. In helicopter aerodynamics or helicopter engineering, they traditionally use computationally precise equations, to design or to verify real-world helicopter models[3, 13, 14]. However, from the viewpoint of simulation or computer graphics area, these equations are too complex and over-precise to achieve real-time simulation.

In computer simulation and its related areas, we cannot find any literature directly related to the helicopter motions, to the best of our knowledge. Although there are several

---

[*] Corresponding author: oceancru@gmail.com

[†] This work was supported by the IT R&D program of MKE/KEIT. [10041145, Self-Organized Softwareplatform(SOS) for welfare devices]

cases of real-time helicopter motions in the field of computer games, we also failed to get detailed literatures. Recently, we have a great deal of progress in fluid dynamics simulation area, and a few real-time simulation methods are available at this time[8, 19, 15, 16]. In contrast, we have only few results for handling interactions between solids and fluids, and they are still hard to show real-time behaviors[2, 4, 17]. Thus, it would be difficult to achieve real-time simulation of helicopters, using these traditional simulation methods.

C. Yuksel and others[20] used a height field-based method for fluids and an existing rigid-body simulation method for rigid bodies, respectively, to finally simulate rigid bodies floating on the fluids. Simplifying the interactions between rigid bodies and fluids, they finally suggest a set of simple equations, to reduce overall processing time.

For real-time generation of visually plausible helicopter motions, we concentrate on the motion of rigid bodies such as helicopter bodies and rotor blades, while approximate fluid-related calculations as simple as possible. Additionally, since we have no need to visually express the fluid flows, real-time helicopter simulations are possible though approximately calculating only the forces applied on the rigid bodies rather than fluid actions. We represent the details of this more practical method in this paper.

Most researches in helicopter engineering area are usually focusing on the motion of main rotors[6, 9, 10, 18]. While rotating, the blades of the main rotor collide with air molecules, to generate forces to fly the helicopter up and to change its directions in the sky. The momentum theory and the blade element theory (also known as strip theory) in helicopter engineering are the classical mechanics interpretations for these phenomena[3, 13, 14]. These two theories describe the same helicopter motions from different points of view, and have their own pros and cons.

The momentum theory is suitable for calculating the lift forces of helicopters, but hard to explain the interactions between the rotor blade and surrounding air flows. In the case of blade element theory, they use infinitesimal airfoil elements to treat the rotor blades as miniaturized airplane wings. Although it can overcome the drawbacks of momentum theory, it usually requires extremely heavy computations.

As a typical example in helicopter engineering, NASA used the multi-body analysis of rotors, for the aero-elastic test of V-22 tilt-rotors[6]. Here, rotors are interpreted as articulated multi-bodies with constraint forces, to finally calculate the aero-elastic forces. Based on the blade element theory, the forces acting on the rotor blades are calculated with experimentally measured coefficients. Since the purpose of this research was verification of rotor blade designs, they used complex and accurate equations to calculate the forces on the blades. Although it shows good numerical results for design and verification purposes, it is unsuitable for real-time computer graphics and game applications. Recently, real-time analysis and verification of rotor blade motions are investigated[10]. However, these results are still unavailable to graphics applications.

In this paper, we start from the blade element theory and derive a dynamics model for rotor blades, and thus finally achieve real-time helicopter simulation. Our results are based on the physically-based rigid-body modeling. Our method calculates the forces on the blades as action-reaction forces between air molecules and rigid bodies, which can be easily derived from Newtonian physics[5, 7].

Our method actually consists of two stages. First, we need a geometric model of rotor blades, which corresponds to the real-world blade shape design, as shown in the next section. Derivations based on the traditional blade element theory are followed. Then, we calculate the forces on the rotor blades using physically-based modeling approaches. We present a

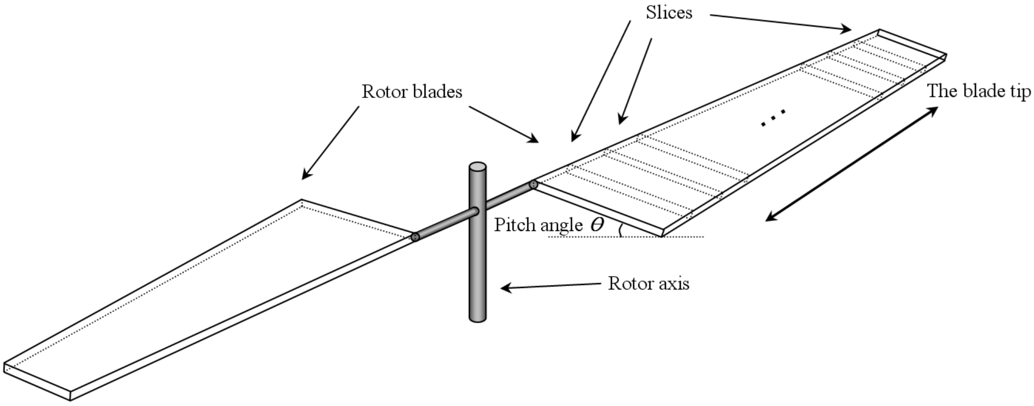


Figure 1. The geometric model of our rotor blades.

somewhat simplified computation model for real-time processing without significant loss of reality. Experimental results show that visually plausible helicopter motions are achieved in real time. Conclusion and future work are followed.

2 Background Works

In this section, we design geometric shapes of rotor blades, and present the forces derived from the blade element theory.

2.1 A Geometric Model for Rotor Blades

Real-world rotor blade designers usually search for compromises between good hovering behavior and forward flight efficiency, in addition to light weights, low noises, minimized oscillations, etc., as followings[13]. Rotor blades usually have high aspect ratios, to minimize drag forces due to blade tip vortices, where aspect ratios are the lengths of blade spans to those of chords. Airflow is fastest at the blade tips and easy to generate vortices acting as downward drag forces. These downward drag forces are degraded as the aspect ratios increased.

Rotor blades are twisted to have smaller pitch angels along to the blade tips, and thus generate equally distributed lift forces when hovering. Larger twist angles, however, make trembling, noises, and engine power loss, especially for forward flights. Instead of excessively changing the angle of attack, the blades need tapers, which make the width of blades narrower as approaching to the blade tips.

The above mentioned factors are essential ones for designing real-world helicopters. For virtual simulations, however, some of them can be excluded. Even in the real world, they choose some compromises due to conflicting factors. For example, larger twist angles give stable hovering characteristics, but power loss and trembling for forward flights. In the case of visual simulations, we have no need to consider all these conflicts. In this paper, we aim at the real-time visually plausible simulations, and thus, we selected some guide lines for our blade design as follows, to simultaneously pursue both of reality and processing speed.

Firstly, we assume that the rotor and its blades are rigid bodies. Although real world blades are somewhat flexible, its flexibility is ignorable. This rigid body assumption would

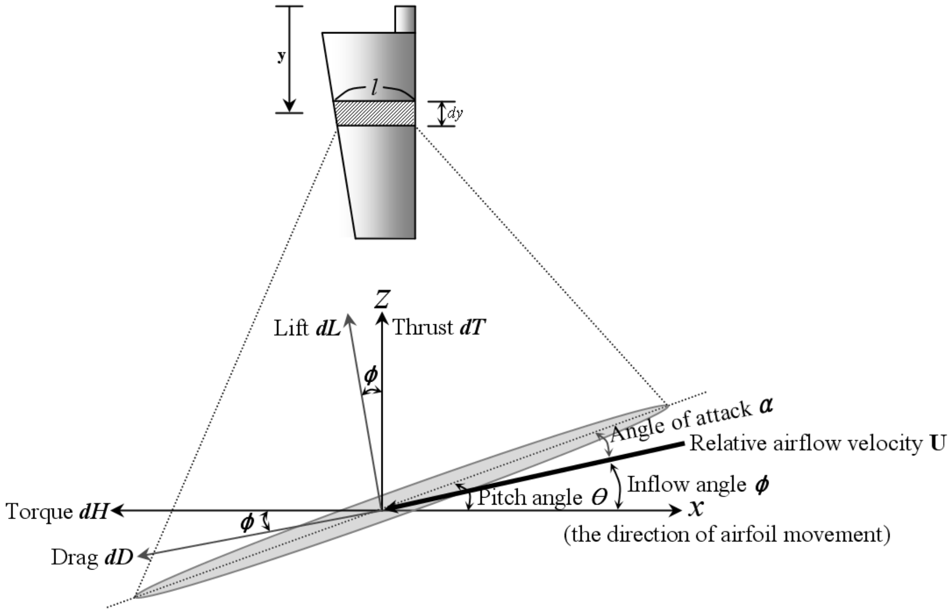


Figure 2. An airfoil element and forces acting on it.

naturally simplify the overall calculation. Secondly, we will use a two-bladed rotor model, as shown in Figure 1. Each blade has a pitch angle θ, with respect to the rotation plane. Thirdly, to approximate the tapering effect, the width of a blade narrows as approaching to the blade tips. Fourthly and finally, we assume that the blades are infinitesimally thin plates. In the real world, streamlined blades with varying thickness generate the best lift forces, due to the Coandă effect[5], and planar blades require much more engine powers. Since we are free from engine powers and energy problems, we choose the simple planar blades, to more simplify equations with respect to the pitch angle and the angle of attack, as presented in the following section.

As shown in Figure 1, each blade consists of n planar slices. To approximate the tapering effect, the widths narrowed, as approaching to the tips. Each slice has its own geometric information including the position vector, the normal vector and the surface area. To dynamically simulate the helicopter motions, the mass and the moment of inertia for blades should be calculated. After geometrically design the rotor shapes, we use Mirtich's method[11] to calculate the physical properties including mass and moment of inertia.

2.2 Blade element theory

According to the blade element theory, the blade can be interpreted as a set of infinitesimal airfoil elements, and their physical properties are integrated along the blade. Airfoil elements are actually a slice of the blades, as shown in Figure 2.

Without loss of generality, we assume that the rotor axis acts as the z axis and the airfoil (and its corresponding blade) moves to the positive x direction. The direction from the rotor axis to the blade tip becomes the y axis, to construct a right-handed coordinate system. With the angular velocity ω, the airfoil rotates with the linear velocity of ωy, where y is the length from the rotor axis to the airfoil. The blades collide with air molecules, and forces are generated. The life force acts to the direction perpendicular to the airflow velocity **U**, while the drag force along the airflow direction. The thrust force and torque acts with respect to the moving direction of the blade, as shown in Figure 2. Finally, the thrust force makes the helicopter fly up in the sky.

The magnitude of the lift force dL and that of the draft force dD on an infinitesimal airfoil can be calculated as follows[14]:

$$\begin{aligned} dL &= \frac{1}{2}\rho C_{\mathrm{L}}\mathbf{U}^2 l dy \qquad (1)\\ dD &= \frac{1}{2}\rho C_{\mathrm{D}}\mathbf{U}^2 l dy, \qquad (2)\end{aligned}$$

where rho is the airflow density, **U** is the relative airflow velocity, and l and dy are the length and width of the airfoil, respectively. C_{L} and C_{D} are lift and drag coefficients, which are determined by the blade shape and the angle of attack.

The thrust force dT and the torque dH with respect to the rotor axis can be derived from the lift force dL and the drag force dD, as follows:

$$\begin{aligned} dT &= dL\cos\phi - dD\sin\phi \qquad (3)\\ dH &= (dL\sin\phi + dD\cos\phi)\, y, \qquad (4)\end{aligned}$$

where ϕ is the inflow angle between the airflow velocity vector and the airfoil velocity vector and y is the length from the rotation axis to the airfoil.

For the real-world streamlined blades, the air flows are inclined by the inflow angle ϕ, mainly due to the circulating inflows. Actually, the effect of circulating inflows is insignificant for forward flights. Using blades with proper twists and tapers, it is also insignificant even for hovering actions. Thus, assuming a very small inflow angle ϕ, we can approximate the direction of the airfoil movement as that of the airflow. Furthermore, approximating $\sin\phi \approx \phi$ and $\cos\phi \approx 1$, Equations (3) and (4) can be approximated as follows:

$$\begin{aligned} dT &= dL\cos\phi - dD\sin\phi \;\approx\; dL - \phi dD \qquad (5)\\ dH &= (dL\sin\phi + dD\cos\phi)\, y \;\approx\; (\phi dL + dD)\, y, \qquad (6)\end{aligned}$$

Through integrating these infinitesimal forces, we finally get the total forces acting on the blades. In the next section, we will represent another way of deriving these aerodynamics equations as our new dynamics model.

3 A Dynamics Model for Rotor Blades

In this section, we represent an approximated calculation of forces acting in the rotor blades, with geometric shapes described in the previous section.

3.1 Forces acting on a slice

As described in the previous section, we can calculate physical forces acting on the blades through integrating their corresponding infinitesimal forces over the airfoils. Using Equations (1) to (6), we can calculate the forces analytically or numerically[3, 13, 14].

We take approximations to the analytical equations, to achieve real-time solutions. In our geometric model of the rotor blades, the slices can be regarded as discrete approximations

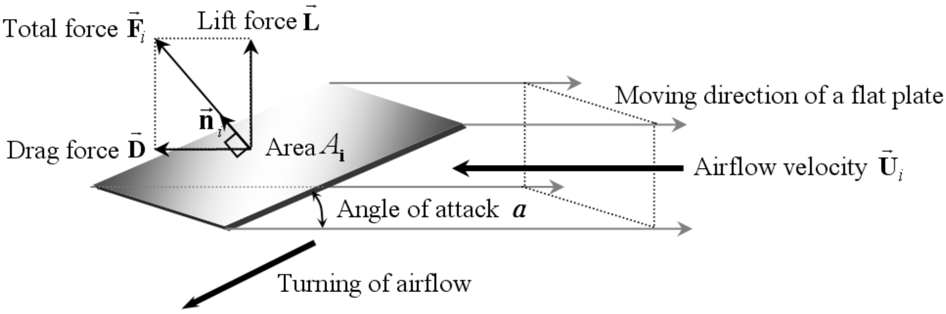


Figure 3. Forces acting on a flat plate.

of the airfoils, as shown in Figure 2. When a slanted flat plate moves in the air, collisions between the plate and the air molecules cause momentum changes. These momentum changes act as action-reaction forces, and thus, the air flows are directed to downward while the plate moves upward, as shown in Figure 3. According to Newton's calculation[5], the magnitude of the lift force L and the drag force D on the flat plate can be calculated as follows:

$$\begin{aligned} L &= \rho \mathbf{U}^2 A \sin^2 \alpha & (7) \\ D &= \rho \mathbf{U}^2 A \sin \alpha \, (1 - \cos \alpha) & (8) \end{aligned}$$

where ρ is the airflow density, $\mathbf{U}$ is the relative airflow velocity, A is the area of the plate, and α is the angle of attack.

The direction of the sum of the lift and drag forces coincides with the normal vector of the plate, and this summed force acts as the sum of aerodynamic forces acting on the plate. We actually regard the set of slices of the blades as flat plates, and use the above force equations to get the sum of the lift and drag forces on a specific slice[12].

When considering the collisions between the slice and the air molecules, the force $\mathbf{F}_i$ and torque $\boldsymbol{\tau}_i$ acting on the i-th slice can be calculated as follows:

$$\begin{aligned} \mathbf{F}_i &= \kappa \left(\rho \mathbf{U}_i \cdot \mathbf{n}_i A_i \mathbf{U}_i \cdot \mathbf{n}_i \right) \mathbf{n} & (9) \\ \boldsymbol{\tau}_i &= \mathbf{F}_i \times \mathbf{y}_i & (10) \end{aligned}$$

where i is the index of the slice, $\mathbf{n}_i$ is the normal vector of the slice, A_i is the surface area of the slice, and $\mathbf{y}_i$ is the vector from the rotor to the slice. A user-controllable constant κ is used for reflecting the viscosity, friction, and power losses for the rotor and blades. Letting $\kappa = 1$, we get an inviscous, frictionless air condition and the entire engine power. The relative airflow velocity $\mathbf{U}_i$ is the sum of the relative airflow velocity $\mathbf{U}^{\text{body}}$ due to the helicopter body movement and that of $\mathbf{U}_i^{\text{slice}}$ due to the rotation of the rotor, as follows:

$$\mathbf{U}_i = \mathbf{U}^{\text{body}} + \mathbf{U}_i^{\text{slice}},$$

where $\mathbf{U}_i^{\text{slice}}$ can be calculated with the rotor angular velocity $\boldsymbol{\omega}$ as follows:

$$\mathbf{U}_i^{\text{slice}} = \mathbf{y}_i \times \boldsymbol{\omega}.$$

In this configuration, the mass flux for the airflow acting on the slice is calculated as $\rho \mathbf{U}_i \cdot \mathbf{n}_i A_i$ and the momentum change, in other words, the magnitude of the force as $\rho \mathbf{U}_i \cdot \mathbf{n}_i A_i \mathbf{U}_i \cdot \mathbf{n}_i$.

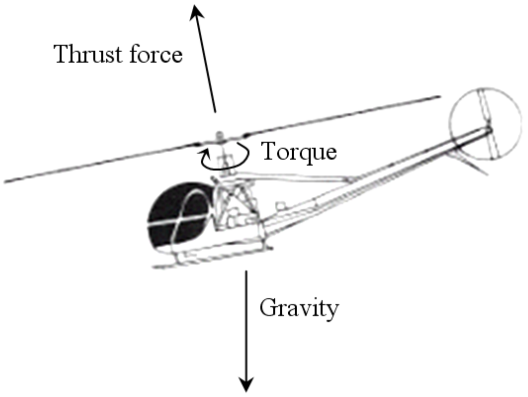


Figure 4. Forces acting on the helicopter.

Comparing the above derivations to those from the blade element theory, Equations (9) and (10) are discretized and simplified forms of the infinitesimal thrust force and torque of Equations (3) and (4) or their simplified ones, Equations (5) and (6). The lift and drag coefficients C_{L} and C_{D} in Equations (1) and (2) are determined by the blade shape and the angle of attack, as already mentioned in the previous section. Actually, our blade shapes and the angle of attack are reflected by the term $\mathbf{U}_i \cdot \mathrm{n}_i$, which is the relative airflow velocity with respect to the angle of attack, in our discretized calculations.

Using Equation (9), we have no need to separate the lift force and the drag force, since the drag forces on the opposite blades are counterbalanced. Thus, we can simply sum up the forces of Equation (9) to finally get the total lift force. In a similar way, the lift forces are canceled out and only the drag forces contribute to the torque, in Equation (10). Conclusively, Equations (9) and (10) correspond to the force and torque calculations in the blade element theory. The total force and torque on the whole rotor and blades can be calculated through simply summing up those at each slice.

3.2 Forces acting on a helicopter

After calculating the force and torque at each slice, we can use them as the same way in the traditional rigid body simulations[1, 12]. As shown in Figure 4, the total thrust and torque on the blades are applied to the helicopter body, through rotor connections.

With the rotating blades, the torque acts as a kind of resistance to the blade motion. Thus, the angular velocity of the rotor will be decreased. To overcome this situation and additionally to simulate the real-world engine power, we also apply the following torque:

$$\boldsymbol{\tau}_{\mathrm{engine}}(t) = \psi(t)\mathrm{R}(t),$$

where $\psi(t)$ is the magnitude of the torque, corresponding to the real world engine rotation speed. $\mathbf{R}(t)$ is the direction of rotor, at the specific time t, and would be controlled by the user.

For simplicity, we actually canceled out the rotating component of the constraint forces with the thrust forces generated by the tail rotor. In this way, we generated natural motions without simulating the tail rotor. Overall physical simulations are performed using traditional rigid body simulation method[1].

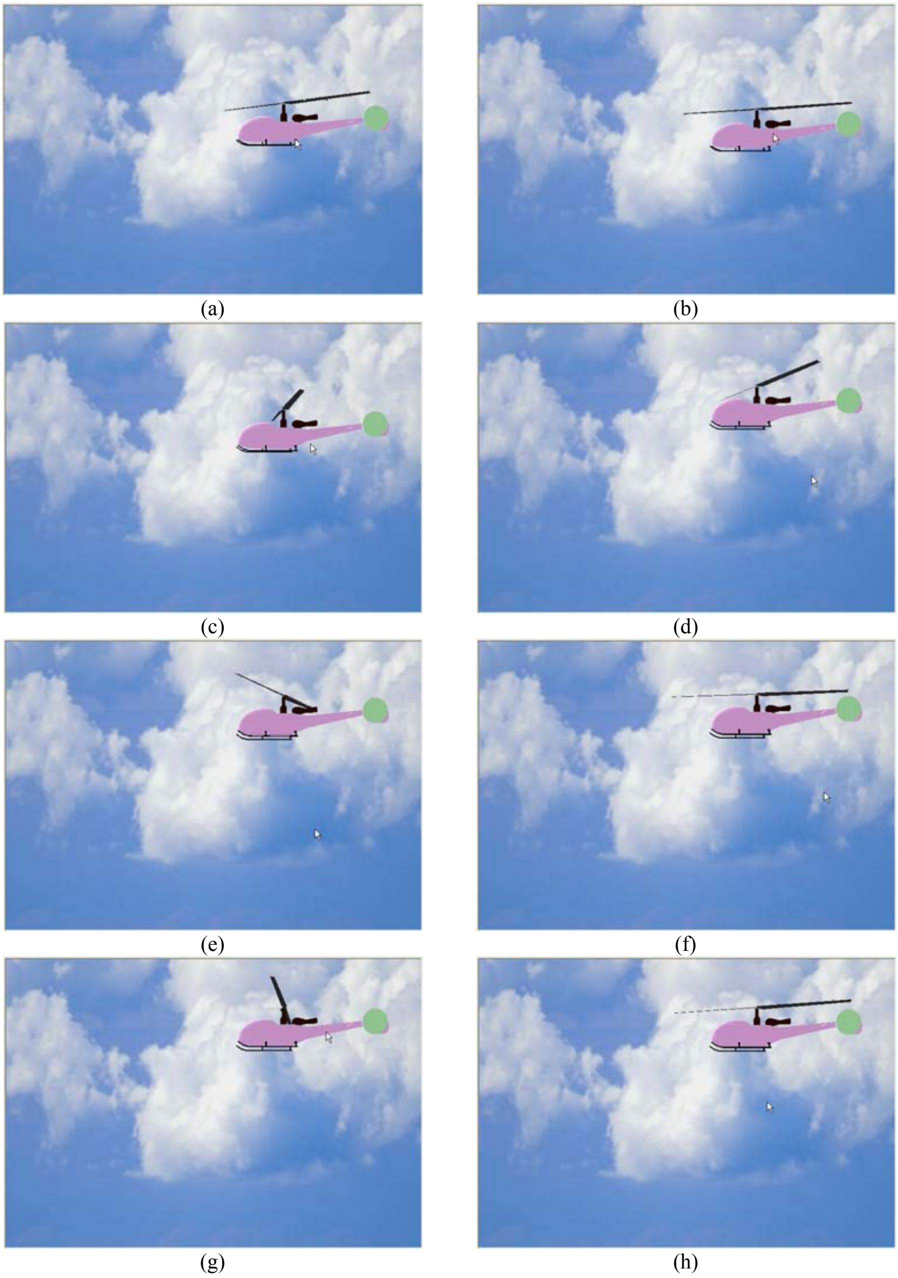
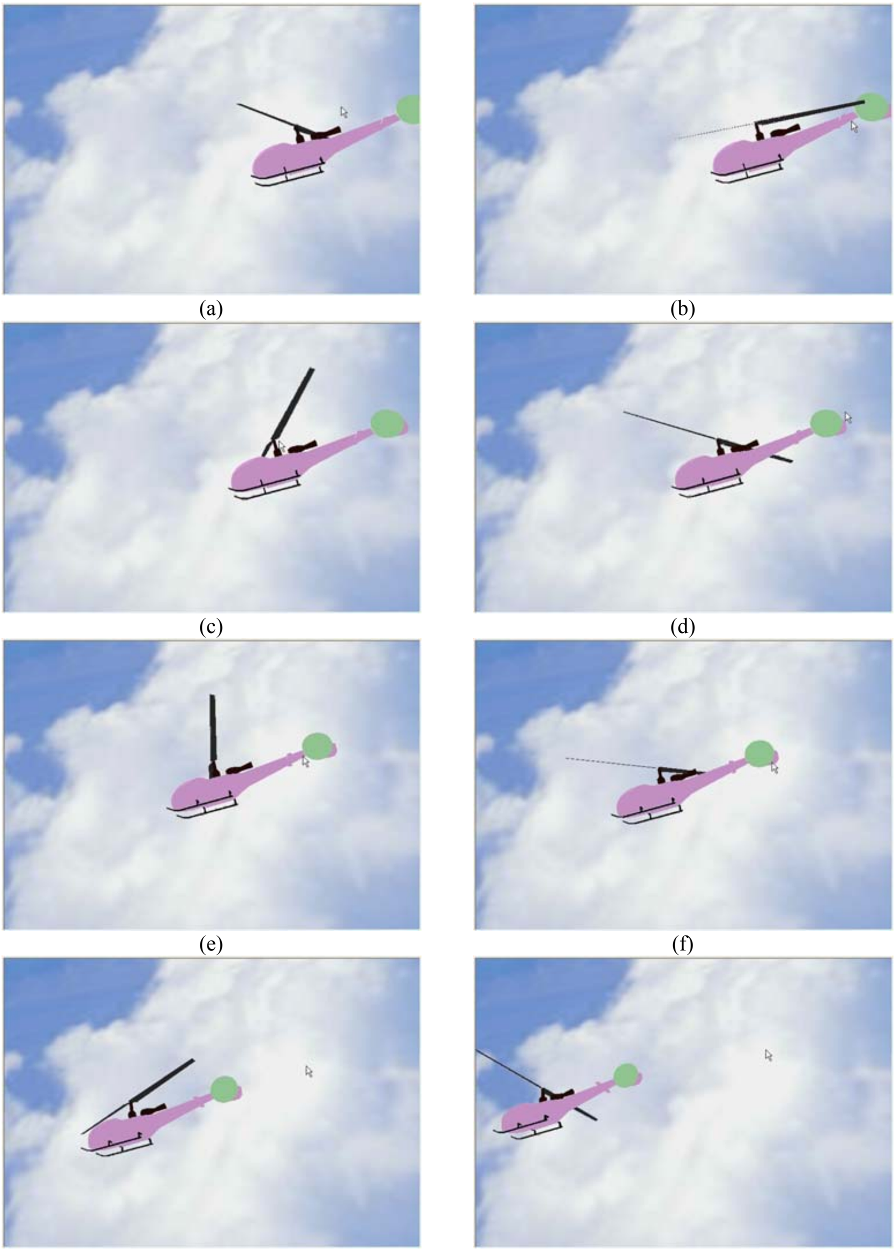


Figure 5. A helicopter hovering in the sky.

4 Experimental Results

A prototype system is implemented using Visual C++ and DirectX libraries on the Microsoft Windows-based PC. We used an Intel Core 6600 2.4GHz CPU with 2G byte memory and GeForce 7950 GPU with 512M byte video RAM. Our prototype implementation displays the helicopter motions according to the current configuration of the main rotor and the engine power. Users can interactively control helicopter motions through keyboards or mice.



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 6. Forward flight of a helicopter.

Figure 5 is a sequence of images for a helicopter hovering. Figure 6 is for a forward flight. In both cases, our system shows much sufficient processing speed for real-time control. As a strength test, we simulated a crowd of 1,000 helicopters simultaneously, as shown in Figure 7. Even in this case, the motions look natural with respect to the rotor and engine power changes.

Table 1 shows our experimental results with $n = 36$ slices for the rotor blades, with respect to a set of varying number of helicopters. Figure 8 is the graph representation of the results in Table 1. Figure 8 shows a linear dependency between the elapsed time and

Figure 7. A crowd of 1,000 helicopters.

Table 1. Processing time for $n = 36$ slices.

	number of helicopters					
	1	100	200	400	800	1,000
simulation time (msec)	0.026	2.106	4.191	8.320	16.647	20.801
rendering time (msec)	0.156	5.519	10.760	21.724	42.894	55.882
total time (msec)	0.182	7.625	14.951	30.044	59.541	76.683
frames per second	5,494.505	131.148	66.885	33.285	16.795	13.041

the number of helicopters. Most of the processing time was spent on the rendering of the final results on the 1280×1024 screen, while less than 27% was used for the simulation. As shown in Table 1, our system achieves more than 33.28 frames per second even for a crowd of 400 helicopters, which will be sufficient for real time processing in most cases. Since our prototype implementation is not fully optimized, it would be more accelerated in the next version.

5 Conclusion and Future Work

We presented a dynamics model of rotor blades for helicopter simulation. Our model is based on the traditional rigid body simulations, rather than fluid simulation methods, and achieved real-time simulations. Based on Newton's thrust force calculation method, we simplified the aerodynamics equations, while preserving aerodynamics properties.

Due to the remarkable processing speed of our prototype implementation, we expect that our approach can be also used for visually simulating fluid-rigid body interactions, such as flight simulation and submarine simulation, especially for game programs and casual simulators. To implement higher level helicopter motions such as blade flapping, the

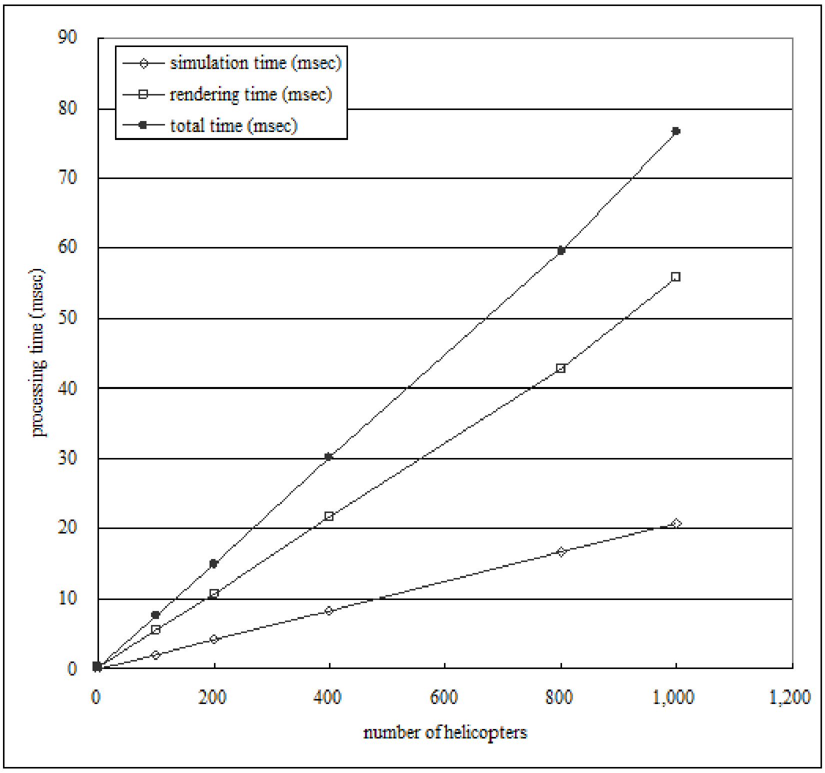


Figure 8. Processing time versus number of helicopters.

rotor blades should be modeled as articulated multi-bodies, and further more, interactions between fluids and rigid bodies should be more precisely calculated. However, these precise calculations may introduce serious drawbacks in processing speed, as already shown in traditional helicopter engineering methods.

As a demonstration, we presented a crowd of 1,000 helicopters. For more realistic and spectacular scenes, our system would be equipped with a kind of crowd simulation module in near future. We also need to extend our results to various types of helicopters. From the rendering point of view, we need to introduce motion blur techniques especially for the rotating blades. Currently, we focused on the typical two-blade single rotor helicopters. Our implementation would be extended to support dual rotors, coaxial rotors, intermeshing rotors, transverse rotors, etc.

Acknowledgements

This work was supported by the IT R&D program of MKE/KEIT. [10041145, Self-Organized Softwareplatform(SOS) for welfare devices]

References

[1] Baraff, D.: Analytical methods for dynamic simulation of non-penetrating rigid bodies. In: SIGGRAPH ’89. pp. 223–232 (1989)

[2] Batty, C., Bertails, F., Bridson, T.: A fast variational framework for accurate solid-fluid coupling. In: SIGGRAPH ’07. p. 100 (2007)

[3] Bertin, J.J.: Aerodynamics for Engineering. Prentice-Hall, 4th ed. edn. (2001)

[4] Carlson, M., Mucha, P.J., Turk, G.: Rigid fluid: Animating the interplay between rigid bodies and fluid. In: SIGGRAPH ’04. pp. 377–384 (2004)

[5] Craig, G.M.: Stop Abusing Bernoulli!: How Airplanes Really Fly. Regenerative Press (1998)

[6] Ghiringhelli, G.L., Mantegazza, P., Masarati, P., Nixon, M.W.: Multi-body analysis of the 1/5 scale wind tunnel model of the V-22 tiltrotor. In: American Helicopter Society 55th Annual Forum (1999)

[7] Goldstein, H., Poole, C.P., Safko, J.L.: Classical mechanics. Addison-Wesley, 3rd. edn. (2002)

[8] Heo, N., Ko, H.S.: Detail-preserving fully-eulerian interface tracking framework. In: SIGGRAPH Asia ’10 (2010)

[9] Kerlick, G.D.: Visualization for aerodynamic design of helicopter rotor blades. In: Proc. of the 6th IEEE Conf. on Visualization. p. 351 (1995)

[10] Masarati, P., Attolico, M., Nixon, M.W., Mantegazza, P.: Real-time multibody analysis of wind-tunnel rotorcraft models for virtual experiment purposes. In: AHS 4th Decennial Specialists’ Conf. on Aeromechanics (2004)

[11] Mirtich, B.: Fast and accurate computation of polyhedral mass properties. J. of Graphics Tools 1(2), 31–50 (1996)

[12] Moore, P.M., Wilhelms, J.: Collision detection and response for computer animation. In: SIGGRAPH ’88. pp. 289–298 (1988)

[13] Prouty, R.W.: Helicopter Aerodynamics. Phillips Publishing Company (1985)

[14] Seddon, J., Newman, S.: Basic Helicopter Aerodynamics: An Account of First Principles in the Fluid Mechanics and Flight Dynamics of the Single Rotor Helicopter. American Inst. of Aeronautics & Ast., 2nd ed. edn. (2001)

[15] Stam, J.: Stable fluids. In: SIGGRAPH ’99. pp. 121–128 (1999)

[16] Stam, J.: Real-time fluid dynamics for games. In: Proc. of the Game Developer Conference (2003)

[17] Takahashi, T., Ukei, H., Kunimatsu, A., Fujii, H.: The simulation of fluid-rigid body interaction. In: SIGGRAPH ’02. p. 266 (2002)

[18] Theodore, C., Celi, R.: Flight dynamic simulation with refined aerodynamic and flexible blade modeling. In: Proc. 56th Annual Forum of the American Helicopter Society (2000)

[19] Thurey, N., Wojtan, C., Gross, M., Turk, G.: A multiscale approach to mesh-based surface tension flows. In: SIGGRAPH ’10 (2010)

[20] Yuksel, C., House, D.H., Keyser, J.: Wave particles. In: SIGGRAPH ’07. p. 99 (2007)