Kinematics Modelling and ADAMS MATLAB/Simulink Co-Simulation for Automated Aerobridge Docking Process

Huang, YunYun; Wang, Jianliang; Tham, Desmond Mark

2017

Huang, Y., Wang, J., & Tham, D. M. (2017). Kinematics Modelling and ADAMS
MATLAB/Simulink Co-Simulation for Automated Aerobridge Docking Process. 2017 3rd
International Conference on Control, Automation and Robotics (ICCAR), 304-309.

<https://hdl.handle.net/10356/82113>

<https://doi.org/10.1109/ICCAR.2017.7942708>

© 2017 IEEE. Personal use of this material is permitted. Permission from IEEE must be
obtained for all other uses, in any current or future media, including
reprinting/republishing this material for advertising or promotional purposes, creating new
collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted
component of this work in other works. The published version is available at:
[<https://doi.org/10.1109/ICCAR.2017.7942708>].

Downloaded on 26 Apr 2023 20:43:54 SGT

Kinematics Modelling and ADAMS MATLAB/Simulink Co-Simulation for Automated Aerobridge Docking Process

YunYun Huang, Jianliang Wang

School of Electrical and Electronic Engineering
Nanyang Technological University, Singapore
50 Nanyang Avenue, Singapore 639798
e-mail: huangyy@ntu.edu.sg; ejlwang@ntu.edu.sg

Desmond Mark Tham

ST Engineering-NTU Corporate Lab
Singapore Technologies Dynamics Pte Ltd
249 Jalan Boon Lay, Singapore 619523
e-mail: thamdmr@stengg.com

Abstract—The objective of this work is first to present the kinematics modelling of an aerobridge for the application of automated aerobridge docking process to the aircraft, which will solve the manpower shortage issues as well as prevent the damage to aircraft due to human error in manual operation. As the aerobridge is a complex system which involves five active DOFs and four free DOFs, we have proposed an alternative solution of kinematics modelling for this higher order systems. And then the numerical solutions for the inverse kinematics equations are obtained using MATLAB and the kinematic modelling of the aerobridge is validated successfully through the motion simulation using ADAMS. The control schemes of the aerobridge docking process are proposed based on (1) Home position to Pre-set position, and (2) Pre-set position to Target position, which are developed in MATLAB/Simulink. Co-simulation model is established based on ADAMS/Control and MATLAB/ Simulink, which is used for modelling & design of the control systems, and also for visualizing and validating the aerobridge docking process.

Keywords- robotics; kinematics modelling; co-simulation; control; industrial automation

I. INTRODUCTION

For the past 50 years more and more industrial robots have been used to handle the dirty and dangerous tasks in manufacturing plants, such as welding, painting, assembly, pick and place for printed circuit boards, packaging and labeling, palletizing, and product inspection etc [1].

Industrial automation deals primarily with the automation of manufacturing, quality control and material handling processes. Industrial automation is to replace the decision making of humans and replace human operators in tasks that involve hard physical or monotonous work [2], and it's being deployed in many new application areas.

Thanks to the availability of increased computing power, and advances in sensor systems, the development of robotic systems and the precise control of motion, has progressed with a renewed vigor in recent years [3]. There are also great promise for Virtual Reality (VR) used as a tool for CAD, robot programming, plant process simulation for robotics applications [4].

Now robots with sensors, vision, and computer control are doing tasks that required human skills in the past [5]. In the coming decade, significant progress can be expected in

manufacturing robotics and automation, automotive, service, and health care robotics, demonstrating the utility of robotic systems and, as a result, helping their societal acceptance [3]. They can be programmed to carry out a great number of different movements and operate with the speed and efficiency of automatic special purpose machines [6].

Currently, the docking of aerobridges to aircraft is manually performed by qualified licensed operators. With the manpower crunch and expansion of airport, there will be further strain on existing manpower resources to meet operational demands especially during peak arrival and departure hours.

Incidents have occurred due to human error, where the aerobridge structure came into contact with the aircraft body, resulting in damage to the aircraft, which is very costly and causing flight delays.

The aim of this project is to develop a robust solution, an automated passenger boarding bridge (Aerobridge) to address the following challenges: (1) Manpower shortage to operate aerobridges, and (2) Injury to personnel or damage to aircraft or equipment due to human error in manual operation.

In the following sections, the forward and inverse kinematics modeling of the aerobridge and validation using motion simulation is presented in Section II. The control system of the aerobridge docking process is designed in MATLAB/Simulink and is described in Section III. In Section IV, ADAMS MATLAB/Simulink co-simulation is presented followed by the conclusion and future works of this paper.

II. KINEMATIC MODELLING OF AEROBRIDGE

In order to fully automate the aerobridge docking process, we have to build up an accurate model of the aerobridge system according to the mechanical structure and dimensions. The kinematic model is used to calculate the position of the aerobridge related to its input changes (e.g., drive column and driving wheels). The design of aerobridge is complex. It has five degrees of freedom (DOFs) in order to provide the motion of the cabin end effector (see Fig. 1).

The aerobridge has one vertical linear motion at drive column, two active rotational motions at cabin and cabin floor, and one differential drive of two driving wheels. There are one free linear motion between two tunnels, and two free

rotational motions at rotunda, and one free rotational motion at differential drive of two driving wheels.

It is well known that a general transformation between two joints requires four parameters by Denavit & Hartenberg [7]. These parameters known as the Denavit-Hartenberg (DH) parameters have become the standard for describing robot kinematics. The DH method that uses four parameters is the most common method for describing robot kinematics. However, for complex systems like aerobridge with higher order DOFs, the DH method cannot be used. In order to compute the position of the cabin end-effector from specified values for the joint parameters, the kinematic equations of aerobridge has to be derived.

Figure 1. Kinematic Modelling of Aerobridge

A. Forward Kinematic Modelling

Currently, the existing aerobridge is designed for manual operation by qualified licensed operators. The aerobridge operates from Home position to Pre-set position, which is 0.5 meter away from the aircraft door, followed by the final 0.5 meter alignment to reach the Target position, which is manually operated by an aerobridge operator.

The origin of the coordinate system XYZ is at the rotunda base. The coordinate systems are illustrated in Figure 1. The correct docking means the cabin floor that makes contact with the aircraft floor and the cabin floor must be paralleled to the aircraft floor, this allows passengers to slowly transition from level aircraft floor to sloping aerobridge floor. So we consider the Target position of the end-effector is at the center of cabin floor (X_c, Y_c, Z_c) positions and direction angle of cabin floor θ_{cd} . The joint variables of the aerobridge are height of drive column H_{dc} , wheel rotation angle ϕ_w , wheel moving distance d_w and cabin rotation angle θ_c . The additional rotation at the cabin floor can be obtained as the rotation center is the same as the center of cabin floor.

Figure 2. Forward kinematics modelling of aerobridge docking

As shown in Fig. 2, the differential drive of the two driving wheels are assumed move in a straight path, which is the shortest distance for docking process. And path planning will be considered based on aerobridge layouts and locations when multiple aerobridges are working together at the same time.

The kinematic equations are derived between the intermediate joint variables (including the free joint variables) and four control joint variables ($H_{dc}, \phi_w, d_w, \theta_c$), and the cabin position parameters ($X_c, Y_c, Z_c, \theta_{cd}$) can then be derived in terms of the four control joint variables ($H_{dc}, \phi_w, d_w, \theta_c$).

The Maple software [8] can be used to solve all the equations and find the relationship between joint variables and cabin positions. For aerobridge moving from Home position to Pre-set position, the forward kinematic equations of aerobridge are derived to describe the motion of cabin end effector from the joint parameters as follows.

$$\begin{aligned} X_c &= f_1(H_{dc}, d_w, \phi_w, \theta_c) \\ Y_c &= f_2(H_{dc}, d_w, \phi_w, \theta_c) \\ Z_c &= f_3(H_{dc}, d_w, \phi_w, \theta_c) \\ \theta_{cd} &= f_4(H_{dc}, d_w, \phi_w, \theta_c) \end{aligned} \quad (1)$$

For fixed aerobridge dimensions, we can calculate the positions of cabin end effector for the given joint variables.

B. Inverse Kinematic Modelling

Control variables can be obtained based on the inverse kinematic model to determine the joint parameters that achieve the desired aerobridge motion from Home position to Pre-set position.

$$\begin{aligned} H_{dc} &= F_1(X_c, Y_c, Z_c, \theta_{cd}) \\ d_w &= F_2(X_c, Y_c, Z_c, \theta_{cd}) \\ \phi_w &= F_3(X_c, Y_c, Z_c, \theta_{cd}) \\ \theta_c &= F_4(X_c, Y_c, Z_c, \theta_{cd}) \end{aligned} \quad (2)$$

Due to the nonlinearity of the forward kinematic equations in (1), closed form solution may be difficult to attain (e.g., using Maple or MATLAB to get symbolic inverse solutions like in (2)), so numerical method has been considered for calculating the control joint variables. In order to find the numerical solutions for the inverse kinematics equations, the initial position parameters of the aerobridge are substituted into the forward kinematic equations, and MATLAB is used to calculate the joint variables that provide the desired position of the aerobridge.

```
[Hdc 1, dw 1, phiw 1, thetac 1]=solve(Xc_1==13000,Yc_1==500,Zc_1==22500,thetacd_1==0.1)
Hdc_1 =
4134.091546622799183267045618752
dw_1 =
17743.80124107337076137530989718
phiw_1 =
-8.3533955254457892017570788069142
thetac_1 =
-0.46590108246832821819296905046287
```

Figure 3. An example of numerical solutions for inverse kinematics

For example, given the Pre-set position at $X=13000\text{mm}$, $Y=-500\text{mm}$, $Z=22500\text{mm}$, $\theta_{cd}=0.1\text{rad}$, by solving the four forward kinematic equations in (1), the four control variables H_{dc} , ϕ_w , d_w , θ_c can be found, and the numerical solutions are shown in Fig. 3.

C. Validation of Kinematic Results by Motion Simulation

To validate our forward kinematic equations and inverse kinematic solution, the motion simulation is created in ADAMS/View with the aerobridge model/structure. The coordinate of the simulation model is consistent with that of the actual aerobridge model.

Figure 4. Snapshot of motion simulation of aerobridge using ADAMS

Given a Pre-set position of aircraft door, the joint variables can be calculated based on the inverse kinematic equations in MATLAB. Then all the joint parameters are input into the ADAMS to provide the motion of the aerobridge. Fig. 4 shows the snapshot of the motion simulation at the final position with four different views, while the white color path indicates the wheel motion trajectory.

Figure 5. Plot of cabin positions during motion simulation in ADAMS

Fig. 5 shows the cabin position during the ADAMS simulation, at the time $t=5$ sec, the cabin reaches the Pre-set position $X=13000\text{mm}$, $Y=-500\text{mm}$, $Z=22500\text{mm}$, $\theta_{cd}=0.1\text{rad}=5.7^\circ$, which is the same as the input data for the inverse kinematic equation as mentioned in Section II B. Therefore, it validates the solution to the inverse kinematics as well as the kinematic modelling of the aerobridge.

III. CONTROL SYSTEM OF AEROBRIDGE DOCKING PROCESS IN MATLAB/SIMULINK

The aerobridge will be operated from Home position to Pre-set position first (which is 0.5 meter away from the

aircraft door) in a straight path. Additional sensors (e.g., vision and distance sensors) will be used to determine the relative distance between the current position of aerobridge and the closed aircraft door. Then through the inverse kinematic model and closed-loop control algorithm, the aerobridge will be operated to perform the final 0.5 meter alignment to the aircraft door (from Pre-set position to Target position). As the aerobridge is moving slowly during the docking process, we assume the dynamic properties of aerobridge is negligible.

A. Home Position to Pre-set Position

The control system for aerobridge operation from Home position to Pre-set position is based on the inverse kinematics model. As it is an open loop system, no feedback is involved as shown in Fig. 6. Optimal path planning will be obtained based on aerobridge locations and even for multiple aerobridges working together.

Figure 6. Control block diagram from home position to pre-set position

Given the aircraft door position, the Pre-set position for the aerobridge docking can be obtained, and the control variables for each joint are calculated based on the inverse kinematics model, which is programmed as an m-file (Inverse_Kinematics_1) of MATLAB function in Simulink (as shown in Fig. 7).

Figure 7. Inverse kinematics model in Simulink

B. Pre-set Position to Target Position

As it is an open loop system from Home position to Pre-set position, the aerobridge will reach at the Pre-set position

with certain position errors. Furthermore, for an arrival aircraft reaching the Target position, there will be some position error due to its big size. So we have no exact position information for both aerobridge at Pre-set position and the aircraft at Target position.

However, we can measure the relative distance between the aerobridge and the aircraft door using sensors, e.g., lateral distance, longitudinal distance, vertical distance and cabin alignment. With the error measurement, we can use the inverse Jacobian matrix for the Cartesian space to joint space mapping, as the control scheme may still require the inverse Jacobian for calculating the joint velocity from the operational space velocity input within the control block [9].

The closed-loop control of the aerobridge from Pre-set position to Target position is shown in Fig. 8. The inverse Jacobian model [10] is used for the final 0.5 meter alignment based on real-time feedback of distance sensor measurement data (lateral distance in X axis, vertical distance in Y axis, longitude distance in Z axis and cabin alignment). Aircraft door detection will be done using vision approach, which is currently under development by our team [11], and distance sensors will be used to determine the relative distances between the aircraft door and the aerobridge.

Figure 8. Control block diagram from pre-set position to target position

Based on the forward kinematic equations in (1), we take first-order partial derivatives, and get (3) using the Jacobian matrix. We can then find the inverse Jacobian matrix in (4)

$$\begin{bmatrix} \Delta X_c \\ \Delta Y_c \\ \Delta Z_c \\ \Delta \theta_{cd} \end{bmatrix} = \begin{bmatrix} \frac{\partial X_c}{\partial H_{dc}} & \frac{\partial X_c}{\partial d_w} & \frac{\partial X_c}{\partial \phi_w} & \frac{\partial X_c}{\partial \theta_c} \\ \frac{\partial Y_c}{\partial H_{dc}} & \frac{\partial Y_c}{\partial d_w} & \frac{\partial Y_c}{\partial \phi_w} & \frac{\partial Y_c}{\partial \theta_c} \\ \frac{\partial Z_c}{\partial H_{dc}} & \frac{\partial Z_c}{\partial d_w} & \frac{\partial Z_c}{\partial \phi_w} & \frac{\partial Z_c}{\partial \theta_c} \\ \frac{\partial \theta_{cd}}{\partial H_{dc}} & \frac{\partial \theta_{cd}}{\partial d_w} & \frac{\partial \theta_{cd}}{\partial \phi_w} & \frac{\partial \theta_{cd}}{\partial \theta_c} \end{bmatrix} \begin{bmatrix} \Delta H_{dc} \\ \Delta d_w \\ \Delta \phi_w \\ \Delta \theta_c \end{bmatrix} = [J] \begin{bmatrix} \Delta H_{dc} \\ \Delta d_w \\ \Delta \phi_w \\ \Delta \theta_c \end{bmatrix} \quad (3)$$

$$\begin{bmatrix} \Delta X_c \\ \Delta Y_c \\ \Delta Z_c \\ \Delta \theta_{cd} \end{bmatrix} = [J] \begin{bmatrix} \Delta H_{dc} \\ \Delta d_w \\ \Delta \phi_w \\ \Delta \theta_c \end{bmatrix} \Rightarrow \begin{bmatrix} \Delta H_{dc} \\ \Delta d_w \\ \Delta \phi_w \\ \Delta \theta_c \end{bmatrix} = [J^{-1}] \begin{bmatrix} \Delta X_c \\ \Delta Y_c \\ \Delta Z_c \\ \Delta \theta_{cd} \end{bmatrix} \quad (4)$$

The inverse Jacobian matrix is calculated from the kinematic model which is programmed as an m-file of MATLAB function in Simulink (as shown in Fig. 9). As the

aerobridge is moving slowly with the maximum speed of 0.4-0.5 meter per second, we assume that the aerobridge is first order system, and the time constant τ can be obtained through experiments. A PID controller is designed for the aerobridge docking process from Pre-set position to Target position.

Figure 9. Inverse Jacobian model in Simulink

IV. ADAMS-MATLAB CO-SIMULATION

In order to develop robust control algorithms which enable precise and reliable aerobridge docking to aircraft, ADAMS-MATLAB/Simulink Co-simulation is used for modelling and design of the control systems. The co-simulation technique based on ADAMS and SIMULINK cooperation can be a useful tool for improving of the development cycle [12], e.g., the development of mechatronic systems [13], Hardware-in-the-loop simulation based on MATLAB and ADAMS [14], and dynamic modeling and kinematic simulation of industrial robot [15].

The ADAMS tool allows virtual prototyping and virtual testing -- the ability to build and test our complex system in the computer before committing to physical prototyping and testing at airport. The mechanical system of the aerobridge is produced in ADAMS, the ADAMS inputs and outputs are identified, and then ADAMS and MATLAB are integrated in a co-simulation system.

Figure 10. Steps of ADAMS-MATLAB Co-simulation

As shown in Fig. 10, the outputs describe the variables that go to the controls application, the inputs describe the

variables that come back into ADAMS and therefore complete a closed-loop between ADAMS and the control applications.

First, the 3D model of the aerobridge is built in ADAMS/View, the fixed joint is located at rotunda base, and all the rotational joints and linear joints are assigned accordingly. The input and output of the aerobridge model in ADAMS/View is determined as state variables. The information exchange between the ADAMS/Controls toolkit and the control system in MATLAB/Simulink is passed by the state variables of the virtual prototype of the aerobridge.

A. Home Position to Pre-set Position

The control system for the aerobridge operation from Home position to Pre-set position is based on the inverse kinematics model. As it is an open loop system, no feedback is involved.

Figure 11. Mechanical subsystem structure and Simulink block diagram (from Home Position to Pre-set Position)

Figure 12. Co-simulation results in Simulink (from Home Position to Pre-set Position)

Given the aircraft door position, the Pre-set position for the aerobridge docking can be obtained, and the control variables for each joint are calculated based on the inverse kinematics model, which is programmed as an m-file with

MATLAB function in Simulink (as shown in Fig. 11). However, it is not be included in the main functions of our Simulink blocks due to processing time issue. So the inverse kinematics model has a separate Simulink block and it will call the main Simulink block using the “sim” function when the inverse kinematics results have been obtained.

The Pre-set position of the aerobridge and the measured data of cabin end effector position from ADAMS are displayed in Fig. 11 with the cabin end effector moving from Home position to Pre-set position based on the control of joint variables of the aerobridge. Fig. 12 shows the co-simulation results of cabin position (X_c , Y_c , Z_c positions and angle of cabin floor θ_{cd}) for the aerobridge moving from Home position to Pre-set position.

B. Pre-set Position to Target Position

The control system for the aerobridge operation from Pre-set position to Target position is based on the inverse Jacobian model and the closed-loop control system (blue box: Pre-set to Target Position as shown in Fig. 13) is developed based on the sensor feedback from the measurement data in ADAMS/View. The relative distance between cabin to aircraft door is measured in ADAMS/View, and is feedback to MATLAB/Simulink (see Fig. 13).

Figure 13. Mechanical subsystem structure and Simulink block diagram (from Home Position to Target Position)

Figure 14. Co-simulation results in ADAMS/postprocessor (from Pre-set Position to Target Position)

The co-simulation results can also be obtained from ADAMS/Postprocessor as shown in Fig. 14, where the cabin position in Z axis is increased from 24500mm to 25000mm and the direction angle of cabin floor is also adjusted to 0 degree (parallel to aircraft door) while the aerobridge performs the final 0.5 meter alignment for the docking process based on the inverse Jacobian model and closed-loop feedback from sensors.

V. CONCLUSION AND FUTURE WORKS

A. Summary

In this paper, the kinematics modelling of the aerobridge docking process is presented. As the aerobridge is a complex system which involves five active DOFs and four free DOFs and the DH method cannot be used. So we have proposed the alternative solution for higher order systems where the forward kinematic equations of the aerobridge has to be derived instead. And then the numerical solutions for the inverse kinematics equations are obtained using MATLAB and the kinematic modelling of the aerobridge is validated successfully through the motion simulation using ADAMS, while the mechanical structure is created and used for validation in ADAMS/View.

The control schemes of the aerobridge docking process are proposed based on (1) Home position to Pre-set position, and (2) Pre-set position to Target position. The control system for the aerobridge operation from Home position to Pre-set position is based on the inverse kinematics model. The closed-loop control of the aerobridge from Pre-set position to Target position is presented in Section III. The inverse Jacobian model is used for the final 0.5 meter alignment based on real-time feedback of distance sensor measurement data.

Because we are not able to access the actual aerobridge at the development stage of the project, ADAMS-MATLAB/Simulink Co-simulation is a good tool to be used for modelling & design of the control systems, and also for visualizing and validating the aerobridge docking process.

B. Future Works

As the induction motors are widely used in industrials because they are rugged, reliable and economical, and their speed control may be required in many applications. However, the induction motor control is complex due to its nonlinear characteristics. We will study and incorporate the induction motor control into our control system in the near future.

And path planning will also be considered and studied based on the aerobridge layouts and locations when multiple aerobridges are working together at the same time.

Integration between the door detection using vision approach as well as real-time distance sensors feedback and

proposed control schemes of the aerobridge docking process will be carried out and demonstrated on a mobile platform before implementing on the actual aerobridge.

ACKNOWLEDGMENT

The research was partially supported by the ST Engineering - NTU Corporate Laboratory through the NRF corporate lab @ university scheme.

REFERENCES

- [1] HaiPeng Wang, N. Li, X. Fang and J. Hofschulte, "An architecture for robot application simulation systems and its industry practice," 2011 IEEE 5th International Conference on Robotics, Automation and Mechatronics (RAM), Qingdao, 2011, pp. 143-147.
- [2] Lamb, Frank (2013). Industrial Automation: Hands on. pp. 1-4.
- [3] R. Madhavan, A. P. del Pobil and E. Messina, "Performance Evaluation and Benchmarking of Robotic and Automation Systems [TC Spotlight]," in IEEE Robotics & Automation Magazine, vol. 17, no. 1, pp. 120-122, March 2010.
- [4] G. C. Burdea, "Invited review: the synergy between virtual reality and robotics," in IEEE Transactions on Robotics and Automation, vol. 15, no. 3, pp. 400-410, Jun 1999.
- [5] A. C. Critchlow, "An Overview of Robotics Engineering", IEEE Potentials, pp. 12-16, 1983.
- [6] Bruno Siciliano, Oussama Khatib, Springer Handbook of Robotics, ISBN: 978-3-540-23957-4, (2008).
- [7] Denavit, Jacques; Hartenberg, Richard Scheunemann (1955). "A kinematic notation for lower-pair mechanisms based on matrices". *Trans ASME J. Appl. Mech.* 23: 215-221.
- [8] K. M. Heal, Maple V Learning Guide. New York Springer, 1998.
- [9] C. C. Cheah, X. Li, Task-Space Sensory Feedback Control of Robot Manipulators, 2015, Springer.
- [10] Simon, C. P. and Blume, L. E. Mathematics for Economists. New York: W. W. Norton, 1994.
- [11] Andonovski Bojan and Jianliang Wang, "Development of a Novel Feature Detection -Based Method for Aircraft Door Position Identification using Vision Approach" 2017 13th IEEE International Conference on Control & Automation (ICCA), Ohrid, 2017 (*In Preparation*)
- [12] D. L. Zhu, J. Y. Qin, Y. Zhang, H. Zhang, M. M. Xia, "Research on Co-simulation Using ADAMS and MATLAB for Active Vibration Isolation System", Proceedings of the 2010 International Conference on Intelligent Computation Technology and Automation (ICICTA 10), Vol. 2. IEEE Computer Society, Washington, DC, USA, pp. 1126-1129, 2010.
- [13] T. Brezina, Z. Hadas and J. Vetiska, "Using of Co-simulation ADAMS-SIMULINK for development of mechatronic systems," 14th International Conference Mechatronika, Trencianske Teplice, 2011, pp. 59-64.
- [14] X. Li, M. Ding, M. Wang, C. Yang and C. Wang, "Hardware-in-the-loop simulation system of VIS based on Matlab and Adams," 2015 IEEE International Conference on Mechatronics and Automation (ICMA), Beijing, 2015, pp. 1391-1396.
- [15] F. Cheraghpour, M. Vaezi, H. E. Shoori Jazeh and S. A. A. Moosavian, "Dynamic modeling and kinematic simulation of Stäubli® TX40 robot using MATLAB/ADAMS co-simulation," 2011 IEEE International Conference on Mechatronics, Istanbul, 2011, pp. 386-391.