50th Annual
Air Traffic Control Association
Conference Proceedings

Fall 2005

October 30 - November 2, 2005
Gaylord Texan Resort & Conference Center
Grapevine, Texas

SESSION #5: AUTOMATION

SESSION #6: AIR TRAFFIC OPERATIONS

Cellular Automata Modeling of En Route and Arrival Self-Spacing for Autonomous Aircrafts

Charles Kim, Assistant Professor
Khalid Abubakar and Obinna Obah, Graduate Students
Department of Electrical and Computer Engineering
Howard University
2300 6th St. NW, Washington, DC 20059

Charles Kim is an assistant professor in the Department of Electrical and Computer Engineering at Howard University. His research interests include performance evaluation of air traffic management and operation, situational awareness monitoring for large network systems, and engineering education.

Dr. Kim received his Ph.D. degree in Electrical Engineering from Texas A&M University, and is currently conducting FAA sponsored research on Traffic Management Advisor (TMA) performance metrics model in which he and his graduate students focus on the comparison analysis of Pre-TMA and Post-TMA arrival traffic patterns and quality of services of NAS.

Khalid Abubakar and Obinna Obah are graduate students in the Department of Electrical and Computer Engineering at Howard University.

1. Introduction

The Next Generation Air Transportation System of the Joint Planning and Development Office (JPDO) envisions a shift from a radar-based system to a space-based system for air traffic control and navigation. Some demonstration of limited realization of the vision has been already reported with actual deployment, notably, in Colorado and Alaska. Under the space based navigation and air traffic flow management control, each aircraft would be aware of its position by GPS based equipment. In addition, on-board transponder-like equipment would broadcast one's position to nearby air space. When an aircraft is aware of its position and those of other aircrafts in its navigation space, the navigation, especially in the en-route domain and near a terminal space, can be done by automated, self-spacing approach. The self-spacing navigation of the so-called autonomous aircrafts would reduce the burden of sector controllers in the voice communication between controllers and pilots.

The objective of the paper is to present a modeling approach of the autonomous aircraft navigation in the en-route domain and terminal arrival air space using cellular automata (CA) approach. This paper reports the results of the modeling and simulation of the self-spacing gate-to-gate navigation using "directional" cellular automata, which is a dynamical system in which space and time are discrete. A cellular automaton consists of a regular grid of cells, each of which can be in one of a finite number of possible states, updated synchronously in discrete time steps according to a local, identical interaction rule. The state of a cell is determined by the previous states of a surrounding neighborhood of cells. The application of cellular automata in physical simulations can provide some interesting results that will help track the trajectories of a given system dynamics.

In the air traffic control and navigation, the National Airspace System (NAS) can be modeled in a CA as a gird system with divided airspaces as cells. The status of a cell, then, is determined by the presence or absence of an aircraft. With a simple rule of interaction, the aircraft's next position is moved to a neighboring cell that is both empty and closest to the destination, which assures self-spacing navigation with shortest path. The dynamics of the aircraft movement in the discrete space and time simulate the navigation of autonomous aircrafts in the en-route domain and terminal arrival, with free maneuvering and separation assurance. This paper demonstrates the modeling of multiple aircrafts from multiple departure airports to multiple arrival airports using "directional" cellular automaton, and the simulation results using Matlab in PC platform.

2. Cellular Automata

John von Neumann, in the 1950's, conceived cellular automaton concept as an ideal structure for modeling self- reproducing "machines", as reported by A.W. Burks [1]. It is a dynamical system where space, time, and variables are discrete. Neumann's cellular automaton theory describes a universe consisting of a homogeneous array of "cells". Each cell is endowed with a finite number of states, and evolves in discrete time according to a uniform local transition rule. The rule can be seen as a function whose argument is the state at time t of itself and the neighboring cells, and whose value is the next state of the considered cell at time t+1.

This concept finds a wide range of application in the field of biological sciences, where the famous "Conway Life" rule invented by John Conway in 1970 still reflects the biology-motivated origin of cellular automata [2]. On the other hand, Stephen Wolfram performed extensive calculations for a class of one-dimensional cellular automaton [3]. Recently, there has been renewed interest in cellular automata due to developments in dynamical systems theory. Cellular automata provide eminently usable models for many investigations in natural sciences, combinatorial mathematics, and computer science [4]. According to Kauffman [5], studies of large, randomly assembled cellular automata have demonstrated that such systems can spontaneously crystallize enormously ordered dynamical behavior.

The idea of CA application is borne out of the current investigation by NASA to explore new concepts of operations for future air transportation systems to improve capacity while maintaining current levels of safety [6]. Directional cellular automata, which guides a movement in the cell space from a departure to a destination, to the air traffic control is well suited for modeling and simulation of, especially, the self-awareness enabled and position broadcasting autonomous flights. One example of such autonomous navigation is the subject of Traffic Alert and Collision System (TCAS). TCAS intends to exchange information using transponders between aircrafts in danger of collision and provide the best advisory resolution information that allows the pilots to maneuver appropriately to maintain safe separation [7]. In addition, the Radio Technical Commission for Aviation (RTCA), proposed an advanced collision avoidance scheme known as Automatic Dependent Surveillance-Broadcast (ADS-B) [8]. This method, which is to be completely implemented by the year 2020, will allow aircrafts within some acceptable defined distances to share information on their path trajectories, and other aircraft specific information to support free flight. ADS-B unlike TCAS supports a forward planning technique to eliminate last minute collision avoidance contingency [9].

3. Directional Cellular Automata for Autonomous Navigation Modeling

In this paper, we present the idea of using a two state directional CA as navigational simulation tool within the NAS by employing 2-dimensional (2D) examples. This 2D approach can easily be extended to a 3D system that uses 3-dimensional array, in which all alternative paths an aircraft can take can be covered. For this 2D system, the layout of the National Airspace System (NAS) is modeled as a system consisting of a homogeneous array of **"**cells" of size MxN. M and N should be selected carefully by considering minimum separation distance, aircraft speed, size of restricted zones, and so on. When needed, cells can reproduce themselves based on the location of the cell. Since airplanes fly slowly at and around departure and arrival, it is even possible to dynamically create another 2D array within a cell so that more than one aircraft can use the cell near airports. However, this "fractal" CA is not the scope of the paper.

In a CA system, each cell takes either state "1" or state "0", and evolves in discrete time and space according to a uniform local CA rule. The cell with "1" state indicates it is occupied by an aircraft, and one with "0" is empty and ready to be occupied. In this model, aircraft position, restricted zone, and severe weather area are defined in the cell space as state "1." An aircraft occupying a defined "cell location" in a given discrete "time step" interacts with neighboring sites by a simple directional CA hierarchical search rule to achieve two major objectives:

- Navigate through an optimal trajectory path in no conflict scenario to maintain pre-planned flight route.
- Maneuver properly in a conflict situation to resolve it while maintaining the best optimal trajectory path.

The aircraft based on its defined position in the air space (cell) has one optimal route to navigate from its departure cell to its destination cell in "no conflict" (NC) situation. In an event that the aircraft is faced with one or several conflict situations, the embedded directional **CA** algorithm conducts an optimal path search to enable the aircraft maneuver through the best of the several alternative routes along its forward path. This is achieved by evolving dynamic interactions between the given aircraft and the status of the neighboring cells using a simple directional CA rule. The methodology adopted here is that the status of the cells in the neighborhood of the aircraft will first be ascertained in a hierarchical order. Action is then taken to advance the aircraft one step to the next free cell, which optimizes its trajectory to the destination cell.

4. CA Modeling and Simulation

CA Model and Rules: By modeling the NAS as a 2D array of size MxN, where M and N are chosen conveniently to suit the many properties of the NAS like the size of restricted zones and aircraft size and speed, we define in mathematical terms the spatial coordinates of aircraft's departure and destination cell references as follows:

$\mathbf{X_i}$: x coordinate of the departure cell,
$\mathbf{X_f}$: x coordinate of the destination cell,
Where, X_i, X_f = 1,2,3,……N
$\mathbf{Y_i}$: y coordinate of the departure cell
$\mathbf{Y_f}$: y coordinate of the destination cell.
Where, Y_i, Y_f = 1,2,3,……M

We can then write a very simple directional rule in terms of the variables defined above.

CA Rule:
If $((X_f - X_i) > 0)$, Move right (or East).

If $((Y_f - Y_i) > 0)$, Move down (or South).
If $((X_f - X_i) < 0)$, Move left (or West).
If $((Y_f - Y_i) < 0)$, Move up (or North).
If $((X_f - X_i) = 0)$, No horizontal move.
If $((Y_f - Y_i) = 0)$, No Vertical move.

The combination of the above stated base rule using Boolean combinatorial logic directs the navigation of the aircraft in discrete time and space by recommending the next cell along the optimal path.

For example**,** if both "Down" and "Right" moves are selected, then optimal path direction would be southeast (or diagonal from left top corner to right bottom corner). Then, based on the status of the best recommended cell, the next best cell might be selected as the next occupation. Once a cell is occupied, its status becomes “1”, while the status of the vacated cell flags back to “0”, and is ready to be occupied by another aircraft. Using this rule, a simulation program is developed and tested to confirm the CA's capability of self-spacing and optimal path navigation under conflict condition.

This simple **CA** local rule applies across the entire cells in a synchronous fashion in the cell space. The navigating aircrafts cruise their ways to destination points via available optimal paths. In the event that two aircrafts make separate requests to move into a particular free cell along their respective paths at the same time in space, consideration has to be given to the size of aircrafts involved, for example, to allow for a prioritized free cell allocation to a flight. A more detailed explanation of this concept follows.

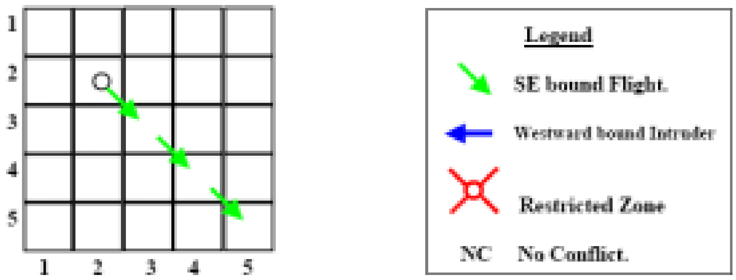

Fig. 1 Flight Optimal Path in No Conflict situation

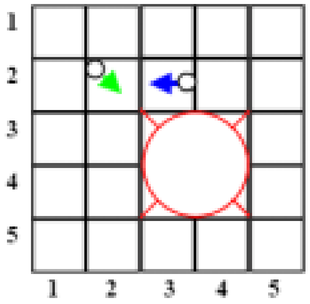

Fig. 2 Flight faced with Conflict situation.

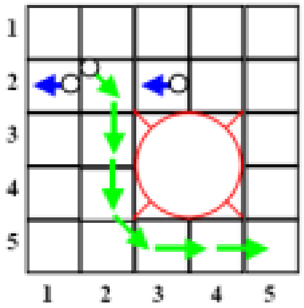

Fig. 3 Flight Path after Conflict resolution.

A simple illustration of the two dimensional **CA** concept using one aircraft scenario is shown in the Figures 1-3. As illustrated in Figure 1, the aircraft departs from CELL (2, 2) and is destined to arrive at CELL (5, 5) with a direct path. For the southeast navigating aircraft alone, it requires only 3 simulation time steps to arrive at its destination cell in no conflict situation via the optimal path. When, as illustrated in Figure 2, flight restrictions are placed at CELL (3, 3), CELL (4, 3), CELL (3, 4) and CELL (4, 4), and the second flight is approaching at the CELL (3,2), the flight path of the first one must be changed to resolve the conflict.

When faced with another aircraft and restricted fly zones, it takes 5 simulation time steps for the aircraft to maneuver its way to the destination port through the optimal path. As can be seen from Figure 3, the aircraft in the first simulation time step, occupies CELL (2, 3), while avoiding both the intruder aircraft and the restricted fly zone. After that, the movement in the vertical downward direction continues until the algorithm searches for a free cell, which leads to optimal path at CELL (3, 5). A second maneuvering is done from there to direct the aircraft heading toward its destination port of CELL (5, 5).

In the event that the entire cells in the aircraft's route at a given time are occupied, the CA resolution of the worst case scenario prompts the aircraft either to adopt a holding pattern within the cell or maneuver backwards, until the conflict is cleared. Using the CA approach of the same local rule applied to all the cells, a very simple CA algorithm can solve seeming complex navigation problem.

5. Example Simulations of CA-based Navigation

To better appreciate the advantage of the CA approach, we simulate two scenarios in a bid to thoroughly assess the performance of the algorithm. In the simulations, navigations are very successful. This is a good indication of the effectiveness of the CA algorithm in securing self-spacing ability as well as maintaining flight optimal path trajectory for autonomous aircrafts. The simulation results of the respective cases are described as follows.

<u>Scenario 1-Single Flight Maneuvering Through Randomly Placed Restricted Zones:</u> This scenario assumes that a southeastward flight departed from CELL (2,2) in the airspace faces with randomly distributed hazardous conditions (bad weather conditions, etc) along the assumed pre-planned flight route to the destination at CELL (10,10). The simulation objective is to examine how efficient the aircraft's maneuvering is in conflict situations as illustrated in Figure 4.

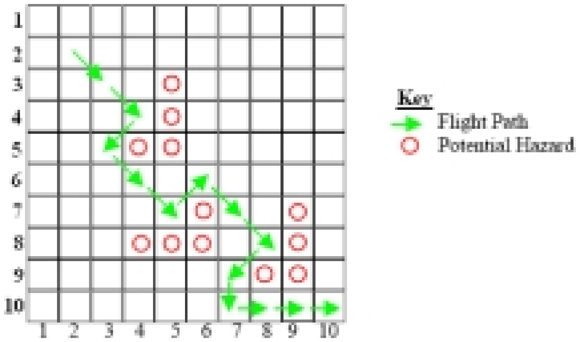

Fig.4 Simulated Trajectory Paths of Flight in Conflict Resolution

The aircraft departed from CELL (2,2) moves along the optimal diagonal path in two successive time steps before facing an obstruction in CELL (5,5). At the point of conflict, the hierarchical search pattern of the CA algorithm illustrated in Figure 5, first searches through the diagonal path to ascertain the status of the neighboring cell labeled "**a**".

Knowing that the diagonal CELL (5, 5) along the optimal trajectory path is occupied automatically transfers the search mode to CELL (5, 4) and CELL (4, 5) labeled as "**b**" and "**c**" respectively in the search order. At this stage, sine both CELL (5, 4) and CELL (4, 5) are occupied, it prompts the algorithm to move to the next level of search, involving CELL (5,3) and CELL (3,5) which correspond in Figure 5 to "**d**" and "**e**", respectively. CELL (3, 5), being the next free cell to be captured in the search algorithm, becomes the next available cell to be occupied. Conflict is then resolved by the aircraft maneuvering from CELL (4, 4) to CELL (3, 5) avoiding collision with the first set of distributed potential hazards.

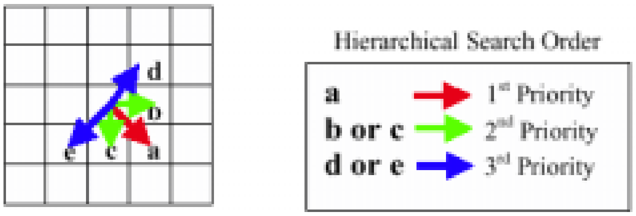

Fig.5. Hierarchical Search in CA model

The best possible path after resolving conflict is the route of (3, 5) → (4, 6) →(5, 7). However, the aircraft on reaching CELL (5, 7) is again faced with an obstruction. The search order of CA algorithm is again activated to determine the next free cell that moves the aircraft closer to the point of destination. This sequence of search is repeatedly applied at the points of conflict to successfully navigate the aircraft from the departure port of CELL (2, 2) to the destination port of CELL (10, 10) along the path traced in Figure 4.

<u>Scenario 2 – Four Autonomous Aircrafts Heading for Different Directions</u>: This case simulates multiple aircrafts configured to cross others paths in the airspace. This simulation also successful: all the flights maneuver successfully without collision to their respective destinations. As illustrated in Figure 6, flight 1 is southeast bound from CELL (2,2) for CELL (20, 20), flight 2 is heading in the northeast direction from CELL (2,19) to CELL (16,4). At the same time, flights 3 and 4 are navigating in the westward and northward directions, from CELL (19,11) to CELL (2,11) and from CELL (12,17) to CELL (12,3), respectively.

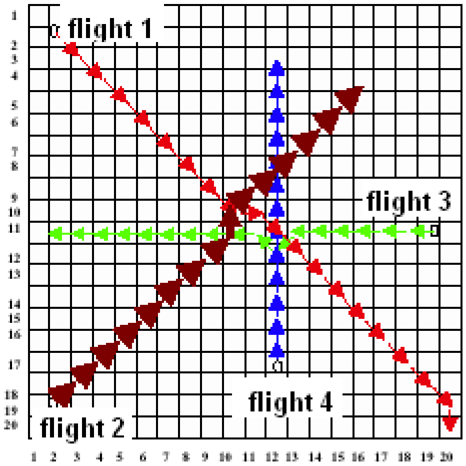

Fig. 6. Simulated Trajectories of the four flights

In each time step, the movements of all the four flights in this scenario are determined sequentially, starting with flight 1 through flight 4. Both flight 1 and flight 2 navigate from their respective departure cells along the diagonal paths, which are optimal routes defined by the directional CA algorithm in no conflict situation for these northeast and southeast bound flights. On the other hand flight 3 and flight 4 have their headings towards westward and northward directions, respectively.

The first point of conflict is met in the time step 6 between flights 3 and 4 occupying CELL (13, 11) and CELL (12, 11), respectively. At that point, flight 3 being the first to move in the time step 7 has to maneuver to CELL (12,12) to avoid collision with flight 4, while the vertical movement of flight 4 continues at the same time step and in the remaining time steps to arrive at its destination CELL(12,3), without encountering any other conflict. See Figure 7 for the maneuvering illustration of flights 3 and 4.

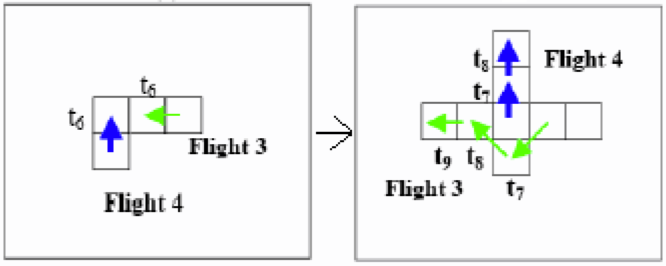

Fig. 7. Maneuvering Illustration of Flight 3 and 4.

In the time step 8, flights 1, 2, and 3 occupying CELL (10, 10), CELL (10, 11) and CELL (11, 11), respectively, have a conflict situation. Flight 1 being the first to move in the time step 9 and according to the CA search pattern, maneuvers to occupy CELL (11, 10) which is free and as well leads to its optimal path. Flight 2 in time step 9 moves to occupy CELL (10, 10) vacated by flight 1. Flight 3 then moves from CELL (11, 11) to occupy CELL (10, 11), vacated by flight 2.

The heading of flight 3 in the time step 9, being the optimal direction that leads to the destination cell continues until the flight arrives its destination cell. Flights 1 and 2 in the time step 10, have to maneuver to direct their headings along the optimal diagonal paths that lead to their destination cells, by occupying CELL (12,11) and CELL (11,9), respectively. See Figure 8 for the maneuvering illustration of flights 1, 2, and 3.

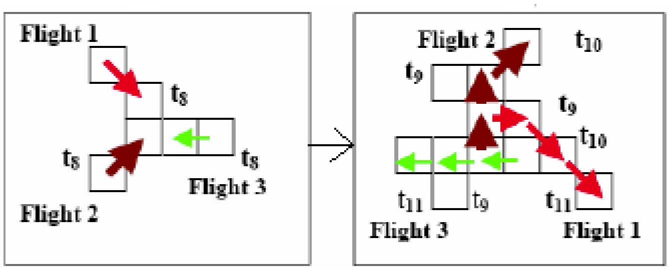

Fig. 8. Self-Spacing Maneuvering Details of flight 1, 2, and 3.

6. Discussions and Conclusions

The examples presented here are simple and small size ones, however, an increase in cell arrays size can bring an equivalent increase in the volume of aircrafts in space, hazardous points, and no fly zones for the simulation runs. The worst-case scenario is observed when an aircraft is obstructed on every side from which it has to maneuver its way out. The aircraft has to move backward and forward, based on the CA worst-case rule, a situation that could be regarded as a holding pattern until all obstructions are cleared. However, this situation is not very likely in the real world situation.

Under CA-based self-spacing and automatic navigation scheme, the layout of the National Airspace System (NAS) could be modeled as a system consisting of a homogeneous array of "cells" of size MxN. The value of M and N determines the capacity of the NAS. Making the cells too big would be a waste of space (i.e., an aircraft holds much bigger space than it needs to satisfy all safety conditions). On the other hand, making the cell size too small might endanger safety since, considering the high speed of aircraft, there would not be enough time to resolve conflict.

The directional CA concept is proposed solely for the autonomous flights which are or would be equipped with Global Positioning System and a transponder for information exchange between aircrafts. The simulation results show that directional cellular automata algorithm could be an effective modeling and simulation tool of autonomous aircrafts in en route and arrival navigation with self-spacing and automatic maneuvering.

7. Acknowledgment

The authors would like to acknowledge the partial financial support of the research funded by Federal Aviation and Administration and the assistance of its project manager, Kelvin Streety.

8.References

[1] A.W. Burks, "Essays on Cellular Automata", University of Illinois Press, Illinois, p.xv, 1968.
[2] M.Gardener, "Wheels, Life, and other Mathematical Amusements", Scientific American, 223:4(1970) 120; 224: 2(1971) 112; 224:3(1971)106.
[3] S.Wolfram, "Theory and applications of Cellular Automata", vol.1, Advances Series on Complex Systems. World Scientific, Singapore, 1986.
[4] Tommasco Toffoli, "Cellular Automata Mechanics", Tech. Rep. No. 208, logic of Computers Group, CCS Dept., The University of Michigan, November 1977.
[5] S.Kaufmann, "Emergent Properties in Random Complex Automata", Proceedings of an Interdisciplinary Workshop, Los Alamos, New Mexico, USA, March 7-11, 1983.
[6] Richard Barhydt, et al, "Regaining Lost Separation in a Piloted Simulation of Autonomous Aircraft Operation"; 5th USA/EUROPE Air Traffic Management R & D Seminar, Budapest, June 23-27, 2003.
[7] R. Y. Gazit, August 1996, Aircraft Surveillance and Collision Avoidance using GPS, Ph.D.Thesis, Stanford University
[8] RTCA Special Committee 147, May 1977, Minimum Aviation System Performance Standards for Traffic Alert and Collision Avoidance System II (TCAS II) Airborne Equipment, RTCA, RTCA/DO-185A.
[9] Robert Holdsworth, et al, "In-flight Path Planning Replacing Pure Collision Avoidance using ADS-B. IEEE AES Systems Magazine, Feb. 2001.