EE291E Hybrid Systems and Intelligent Control
Lecture 1: Introduction

Claire J. Tomlin

Hybrid systems are dynamical systems with interacting continuous-time dynamics (modeled, for example, by differential equations) and discrete-event dynamics (modeled, for example, by automata). The hybrid behavior arises in many different contexts:

- **Continuous systems with a phased operation:**
  - bouncing ball
  - walking robots
  - biological cell growth and division
- **Continuous systems controlled by discrete logic:** the revolution in digital technology has fueled a need for design techniques that can guarantee safety and performance specifications of *embedded systems*, or systems that couple discrete logic with the analog physical environment.
  - thermostat
  - chemical plants with valves, pumps
  - control modes for complex systems, eg. intelligent cruise control in automobiles, aircraft autopilot modes
- **Coordinating processes:** systems which are comprised of many interacting subsystems (called multiple agent, or multi-agent systems) typically feature continuous controllers to optimize performance of individual agents, and coordination among agents to compete for scarce resources, resolve conflicts, etc.
  - air and ground transportation systems
  - swarms of micro-air vehicles

In such systems, the continuous-time model describes such behavior as the motion of the mechanical systems, linear circuit behavior, chemical reactions; and the discrete-event model describes behavior like collisions in mechanical systems, circuit switching, valves and pumps in chemical plants.

Examples

Example 1: Bouncing Ball (Figure 1) [1]

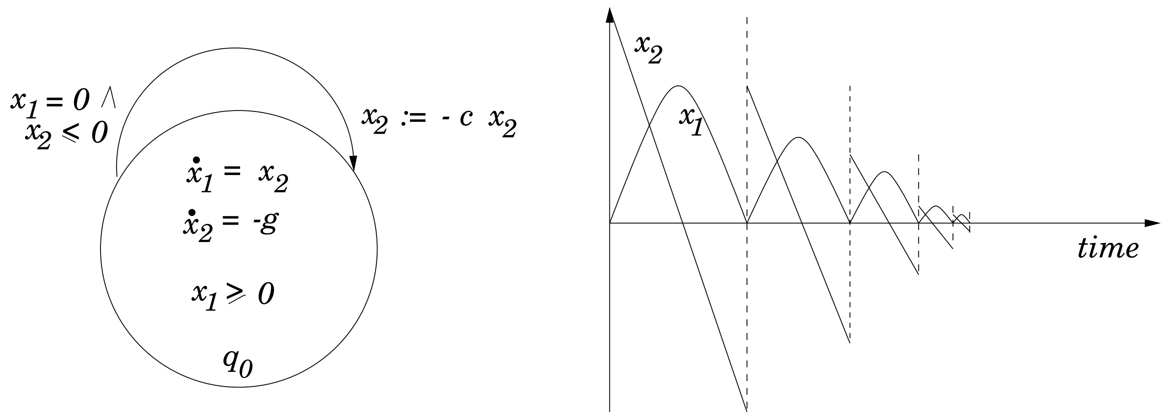


Figure 1: Bouncing ball

- x_1: vertical position of the ball; x_2 velocity of the ball
- g acceleration due to gravity
- $c \in [0, 1]$ coefficient of restitution
- continuous changes between bounces; discrete changes at bounce times
- easy to show the following of the *hybrid automaton* model of the bouncing ball:
  - it is **non-blocking** (meaning from any initial condition there exists at least one state trajectory (solution))
  - $x_1 \geq 0$ is an **invariant** (a property that is always true) of the hybrid automaton
  - if $c < 1$ the hybrid automaton is **Zeno** (takes an infinite number of discrete transitions in finite time)

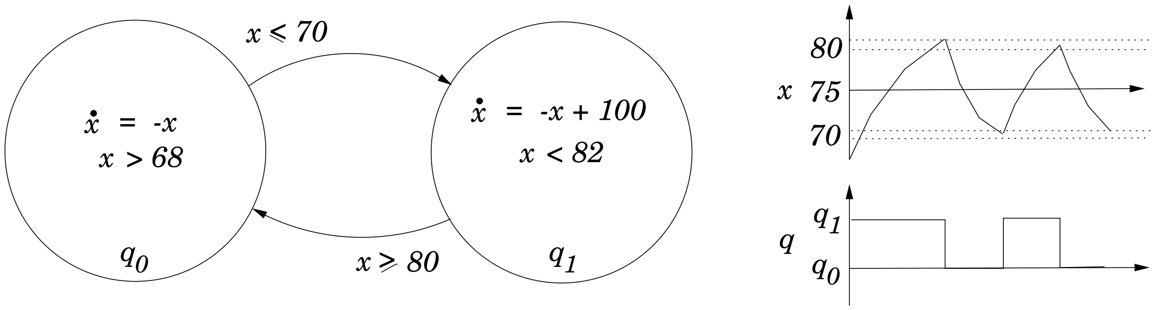


Figure 2: System consisting of a thermostat and the heating of a room

Example 2: Thermostat (Figure 2)

- temperature in a room (x) is controlled by switching a heater on and off
- thermostat regulates x around 75° by turning the radiator on when the temperature is between 68 and 70 and turning the radiator off when the temperature is between 80 and 82
- easy to show that the hybrid automaton model of the thermostat is **non-deterministic**, meaning that for a given initial condition there are a whole family of state trajectories (solutions)

Example 3: Control of an Inverted Pendulum (Figures 3 and 4) [2]

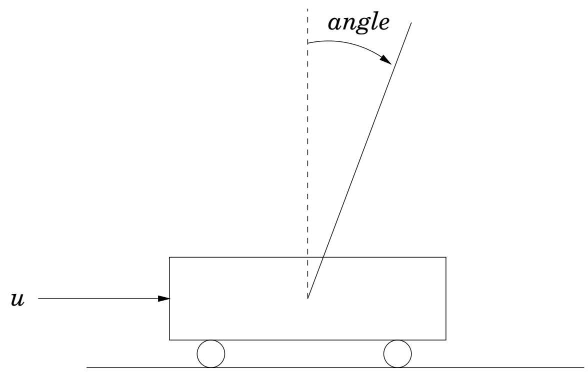


Figure 3: Inverted pendulum on a cart

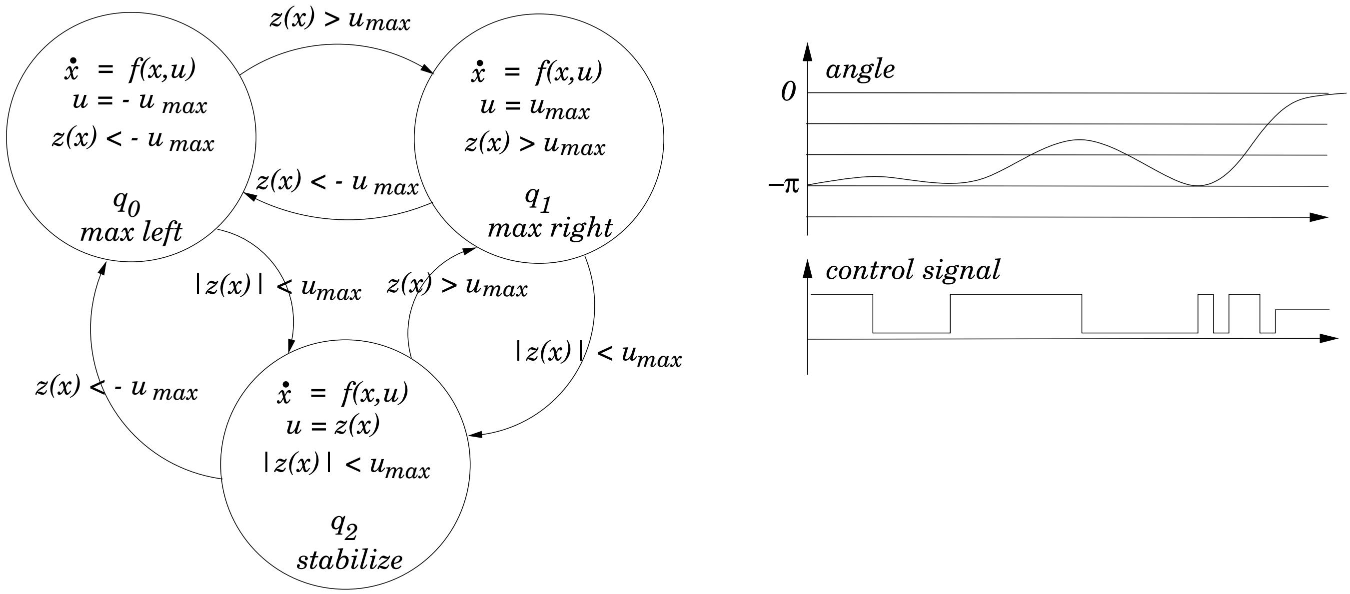


Figure 4: Hybrid control strategy and results for inverted pendulum

- regulate the inverted pendulum to an upright position using the linear acceleration of the pivot as control input u
- $f(x,u) = [x_2, g \sin x_1 - u \cos x_1]^T$, $z(x) = k[x_2^2/2 + g(\cos x_1 - 1)]\text{sgn}(x_2 \sin x_1)$ where x_1 is the pendulum angle and x_2 is the angular velocity
- g is the acceleration due to gravity, $k > 0$

Example 4: Autopilot of a Commercial Jet (Figure 5) [3, 4]

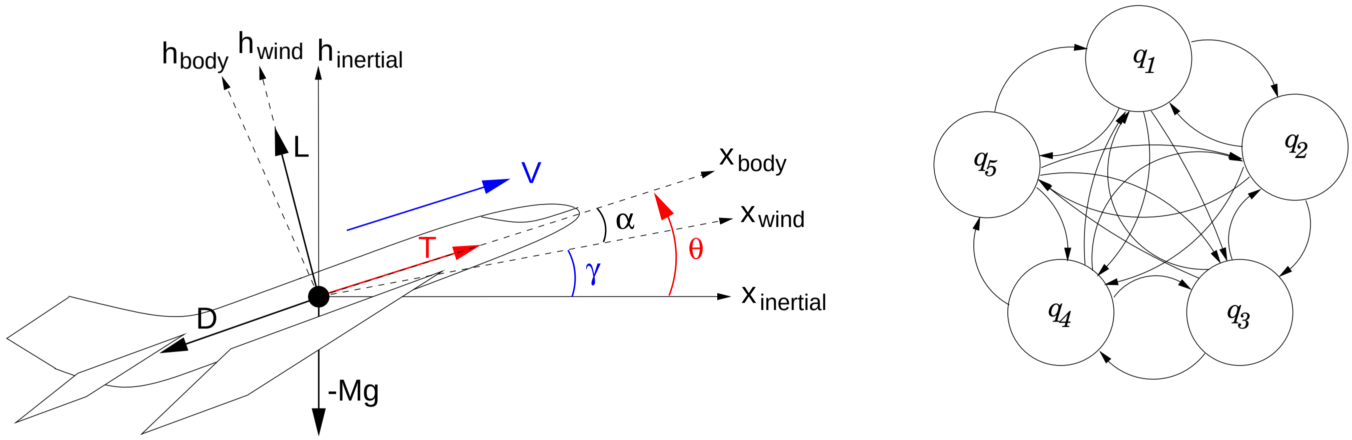


Figure 5: A planar aircraft in flight with attached axes about its center of mass; hybrid automaton of the autopilot.

- Dynamic equations of motion:

$$M \begin{bmatrix} \ddot{x} \\ \ddot{h} \end{bmatrix} = R(\theta) \left[R^T(\alpha) \begin{bmatrix} -D \\ L \end{bmatrix} + \begin{bmatrix} u_1 \\ 0 \end{bmatrix} \right] + \begin{bmatrix} 0 \\ -Mg \end{bmatrix} \quad (1)$$

  where $R(\alpha)$ and $R(\theta)$ are standard rotation matrices, M is the mass of the aircraft, g is gravitational acceleration, control inputs are thrust $u_1 = T$ and pitch $u_2 = \theta$, lift L, drag D

- The *flight path angle* γ and the *angle of attack* α are defined as: $\gamma = \tan^{-1}(\frac{\dot{h}}{\dot{x}})$, $\alpha = \theta - \gamma$; the groundspeed is V

- The system may be discretized into five flight modes, depending on the state variables being controlled:
  - q_1**:** (Speed, Flight Path), in which the thrust T is between its specified operating limits ($T_{min} < T < T_{max}$), the control inputs are T and θ, and the controlled outputs are the speed and the flight path angle of the aircraft $y = (V, \gamma)^T$;
  - q_2**:** (Speed), in which the thrust saturates ($T = T_{min} \vee T = T_{max}$) and thus it is no longer available as a control input; the only input is θ, and the only controlled output is V;
  - q_3**:** (Flight Path), in which the thrust saturates ($T = T_{min} \vee T = T_{max}$); the input is again θ, and the controlled output is γ;
  - q_4**:** (Speed, Altitude), in which the thrust T is between its specified operating limits ($T_{min} < T < T_{max}$), the control inputs are T and θ, and the controlled outputs are the speed and the vertical position of the aircraft $y = (V, h)^T$;
  - q_5**:** (Altitude), in which the thrust saturates $T = T_{min} \vee T = T_{max}$; the input is θ, and the controlled output is h.

Example 5: Automated Highway Systems (Figure 6) [5]

- **Large-scale multi-agent distributed system:** objective is to increase throughput of California's highway system without building new highways, while improving the system safety
- **Challenge 1:** It is unclear whether individual vehicle controllers which minimize travel time of each vehicle leads to maximum throughput of the system
- **Challenge 2:** Controllers which maximize throughput (high speed and close following) are in conflict with controllers designed for safety (low speed and large spacing between vehicles)
- **Solution [5]:** Organize vehicles in tightly spaced groups (called *platoons*) traveling quickly; continuous state for each vehicle (position, velocity, acceleration . . .), discrete state for each vehicle (position in the platoon and lane number)
- **Challenge 3:** State explosion as number of vehicles increase
- **Solution:** coordinate vehicles using communication protocols (Figure 6)

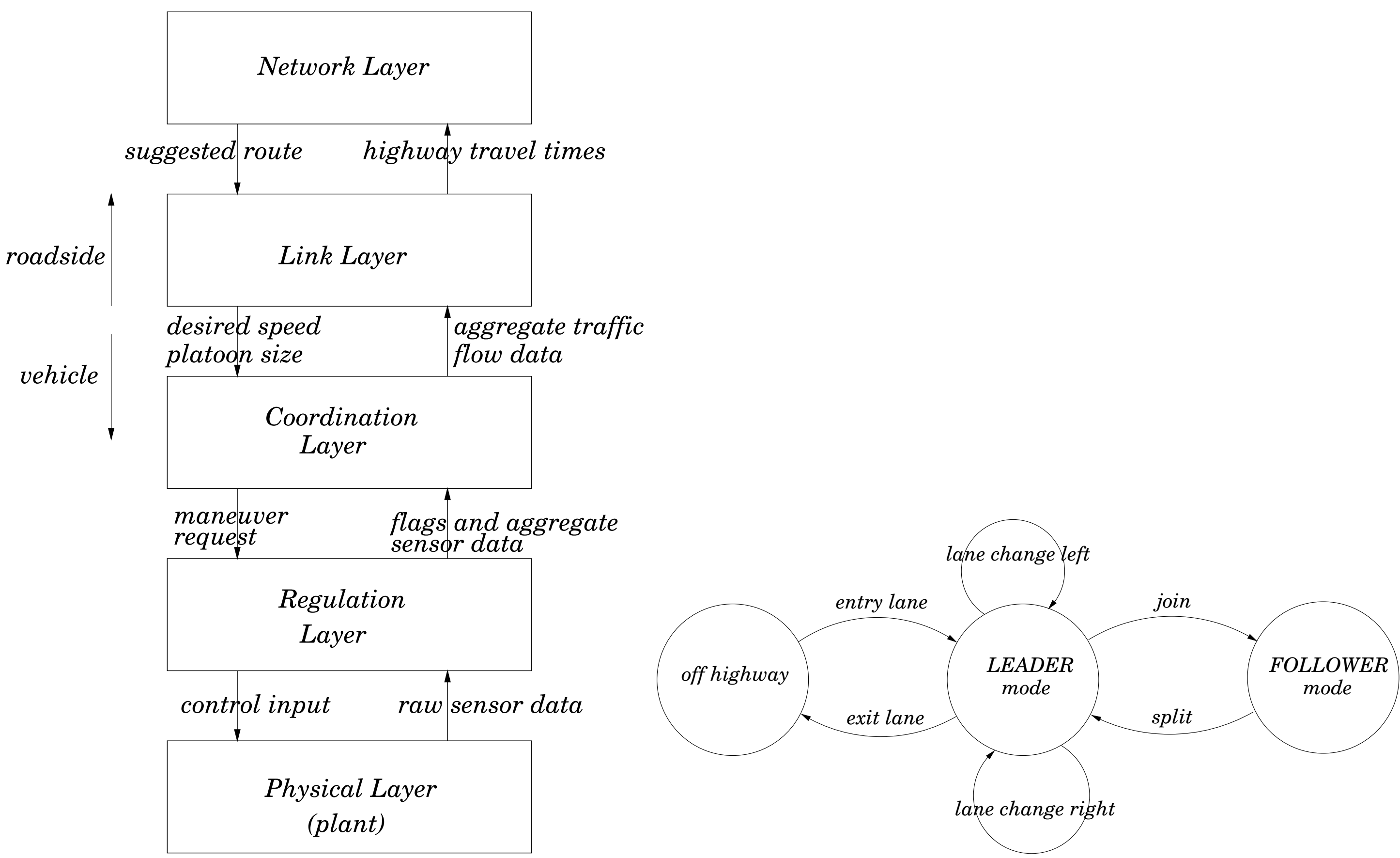


Figure 6: (a) AHS Control Hierarchy, and (b) Discrete state of an AHS vehicle.

Example 6: Free Flight (Figure 7) [6, 7, 8, 9]

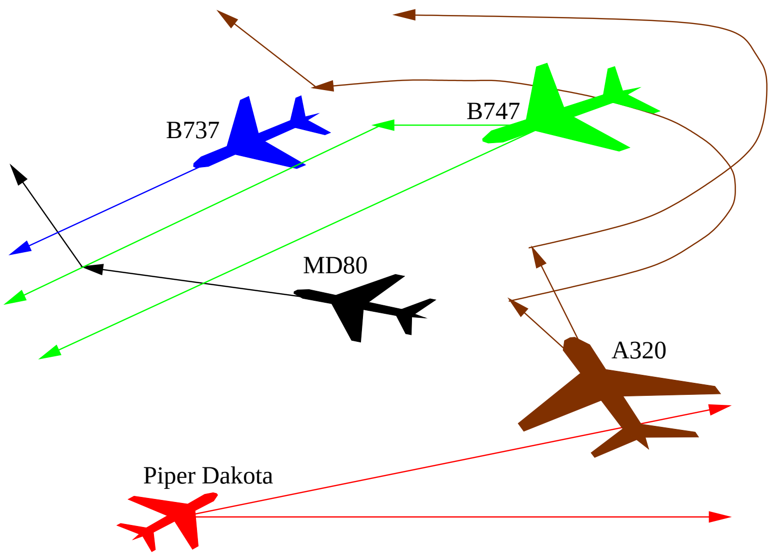
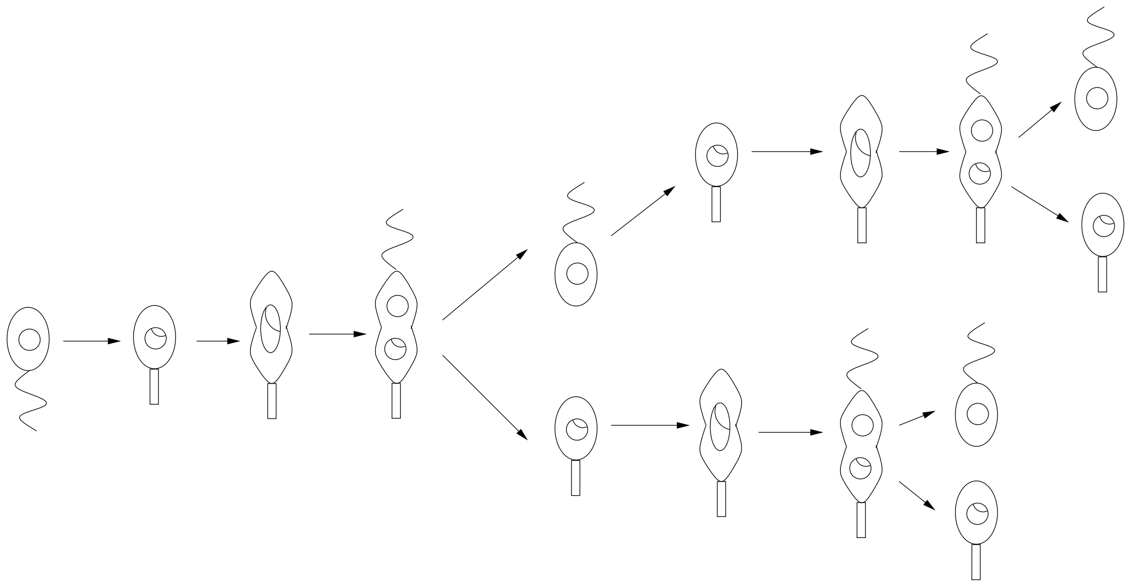


Figure 7: Free flight example

- **Challenge:** Each aircraft flies along a route which ensures that the flight time is short, the fuel consumption is minimized, and inclement weather is avoided, and at the same time will maintain safe distance from other aircraft
- **Solution:** Dynamic route structure and "protocol-based" maneuvers for conflict resolution between aircraft

Example 7: Biological cell growth and division (Figure 8) [10, 11, 12]



from:
H. McAdams and L. Shapiro "Systems Approach to Analysis of Genetic Regulatory Networks"
Stanford University School of Medicine, 2000

Figure 8: Asymmetric cell division in Caulobacter cell.

- The dynamics that govern the spatial and temporal increase or decrease of protein concentration inside a single cell are continuous differential equations derived from biochemistry, yet the activation or deactivation of these continuous dynamics are triggered by discrete switches which encode protein concentrations reaching given thresholds
- Caulobacter cell division: undergoes several mode changes, from a swarmer cell with a flagellum, to an immobile cell with stalk, through an asymmetric division process which creates a new swarmer and a new immobile cell

Hybrid Automaton

We summarize the dynamics of hybrid systems with a model called a *hybrid automaton*, which contains the following entities.

- discrete state $q \in Q$, where Q is a finite collection of variables
- continuous state $x \in \mathbb{R}^n$
- discrete inputs $\sigma \in \Sigma$, where Σ is a finite collection of variables
- continuous inputs $v \in V \subseteq \mathbb{R}^v$
- initial state $Init \subseteq Q \times \mathbb{R}^n$
- continuous dynamics $\dot{x} = f(q, x, v)$
- invariant $Inv \subseteq Q \times \mathbb{R}^n \times \Sigma \times V$ (this defines combinations of states and inputs for which continuous evolution is allowed)
- discrete dynamics $R : Q \times \mathbb{R}^n \times \Sigma \times V \to 2^{Q \times X}$

Example: bouncing ball

- $Q = \{q_0\}$
- $x = (x_1, x_2) \in \mathbb{R}^2$
- $\Sigma = \emptyset$
- $V = \emptyset$
- $Init = \{q_0\} \times \{x \in \mathbb{R}^2 : x_1 \geq 0\}$
- $\dot{x} = f(q_0, x) = [x_2, -g]^T$
- $Inv = \{q_0\} \times \{x \in \mathbb{R}^2 : x_1 \geq 0\}$
- $R(q_0, \{x : x_1 = 0 \wedge x_2 \leq 0\}) = (q_0, (x_1, -cx_2))$ where $c \in [0, 1]$

References

[1] K. J. Johansson, M. Egerstedt, J. Lygeros, and S. Sastry. On the regularization of zeno hybrid automata. *Systems and Control Letters*, 38:141–150, 1999.

[2] K. J. Astrom and K. Furuta. Swinging up a pendulum by energy control. *Automatica*, 1999.

[3] C. J. Tomlin, J. Lygeros, and S. Sastry. A game theoretic approach to controller design for hybrid systems. *Proceedings of the IEEE*, 88(7):949–970, July 2000.

[4] M. K. Oishi, I. M. Mitchell, A. M. Bayen, and C. J. Tomlin. Hybrid system verification: application to user-interface design. *IEEE Transactions on Control Systems Technology*, 16(2):229–244, 2008.

[5] Pravin Varaiya. Smart cars on smart roads: problems of control. *IEEE Transactions on Automatic Control*, AC-38(2):195–207, 1993.

[6] C. J. Tomlin, I. Mitchell, and R. Ghosh. Safety verification of conflict resolution maneuvers. *IEEE Transactions on Intelligent Transportation Systems*, 2(2):110–120, 2001. June.

[7] A. M. Bayen, P. Grieder, H. Sipma, C. J. Tomlin, and G. Meyer. Delay predictive models of the National Airspace System using hybrid control theory. In *Proceedings of the American Control Conference*, pages 767–772, Anchorage, AL, May 2002.

[8] I. Hwang and C. J. Tomlin. Multiple aircraft conflict resolution under finite information horizon. In *Proceedings of the American Control Conference*, Anchorage, AL, May 2002.

[9] R. Teo and C. J. Tomlin. Computing danger zones for provably safe closely spaced parallel approaches. *Journal of Guidance, Control, and Dynamics*, 26(3), May–June 2003.

[10] D. Hung, H. H McAdams, and L. Shapiro. Regulation of the caulobacter cell cycle. *Microbial Development*, 1999. ASM Press.

[11] R. Ghosh and C. J. Tomlin. Lateral inhibition through delta-notch signaling: a piecewise affine hybrid model. In M. D. Di Benedetto and A. Sangiovanni-Vincentelli, editors, *Hybrid Systems: Computation and Control*, LNCS 2034, pages 232–246. Springer Verlag, 2001.

[12] K. Amonlirdviman, N. A. Khare, D. R. P. Tree, W.-S. Chen, J. D. Axelrod, and C. J. Tomlin. Mathematical modeling of planar cell polarity to understand domineering nonautonomy. *Science*, 307(5708):423–426, January 2005.