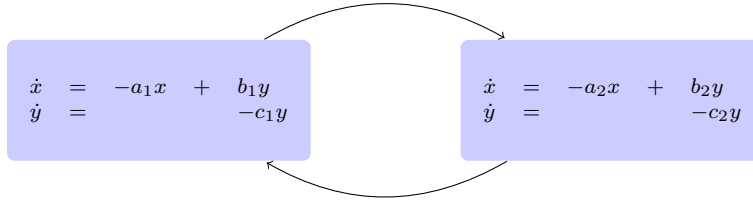


Typeset your solutions using \LaTeX zip your writeup (.pdf) and code in a single file called `nedid-584-F19.zip` and upload this file through Compass.

Problem 2 (25 points). In the following hybrid automaton (switched system), $a_1, a_2, c_1 \in \mathbb{R}$ and $b_1, b_2 \in \mathbb{R}$ are constant parameters of the model.



(a) Assuming, $c_2 \in \mathbb{R}$ is a constant show that the system is globally asymptotically (exponentially) stable under arbitrary switching.

(b) For any arbitrary, possibly negative, c_2 , show that the system is globally asymptotically (exponentially) stable under additional assumption about the switching speed (dwell time or average dwell time).

(a) Considering quadratic Lyapunov functions $V(x) = x^T P x$, where $P = \begin{bmatrix} p & 0 \\ 0 & 1 \end{bmatrix}$. For matrix $A = \begin{bmatrix} -a & b \\ 0 & -c \end{bmatrix}$, if the system is asymptotically stable, we must have $A^T P + P A = \begin{bmatrix} -2ap & bp \\ bp & -2c \end{bmatrix} \prec 0$, which is equivalent to $-2ap < 0$ and $4acp - b^2 p^2 > 0$. For the two modes of the system, we can choose p such that $0 < p < \min \left\{ \frac{4a_1 c_1}{b_1^2}, \frac{4a_2 c_2}{b_2^2} \right\}$. Then $V(x) = x^T P x$ is a common Lyapunov function for both modes and $\dot{V}(x) < 0$ holds for all x . Therefore the system is GAS.

(b) If c_2 is negative, then according to (a), the system is GAS without any need of extra assumptions. Therefore, here we only consider the case where $c_2 < 0$.

For mode 1 (in this paragraph, we omit the subscript for a, b, c which are originally a_1, b_1, c_1), we can define a Lyapunov function $V(x) = x^T P x$, where $P = \begin{bmatrix} p & 0 \\ 0 & 1 \end{bmatrix}$. We want to find positive values p and λ_1 such that $A^T P + P A + \lambda_1 P \prec 0$. Here, $A^T P + P A + \lambda_1 P \prec 0$ is equivalent to $\dot{V}(x) < -\lambda_1 V(x)$ for all x and implies that $\dot{V}(x) < 0$ for all x . Next we show that it is possible

to find such p and λ_1 . $A^T P + PA + \lambda_1 P = \begin{bmatrix} (-2a + \lambda_1)p & bp \\ bp & (-2c + \lambda_1)p \end{bmatrix} \prec 0$ is equivalent to $(-2a + \lambda_1)p < 0$ and $(-2a + \lambda_1)(-2c + \lambda_1)p - b^2 p^2 > 0$. First, we find a small enough λ_1 such that $-2a + \lambda_1$ and $-2c + \lambda_1$ are both negative. Then, we choose a $p \in (0, \frac{(-2a + \lambda_1)(-2c + \lambda_1)}{b^2})$. Then these two inequilities hold.

For mode 2 (in this paragrah, we omit the subscript for a, b, c which are orginally a_2, b_2, c_2), we show that it is possible to find a $\lambda_2 > 0$ such that $(V)(x) < \lambda_2 V(x)$ for all x . It is equivalent to $A^T P + PA - \lambda_2 P = \begin{bmatrix} (-2a - \lambda_2)p & bp \\ bp & (-2c - \lambda_2)p \end{bmatrix} \prec 0$, which is equivalent to $-(2a + \lambda_2)p < 0$ and $(2a + \lambda_2)(2c + \lambda_2)p - b^2 p^2 > 0$. We can just choose λ_2 large enough such that $\frac{(2a + \lambda_2)(2c + \lambda_2)}{b^2} > p$.

Then, we know that in mode 1, $V(x)$ decreases and upper bounded by an exponetially decreasing function $V(x(t)) = \exp(-\lambda_1 t)V(x(0))$, and in mode 2, $V(x)$ increases and upper bounded by an exponetially increasing function $V(x(t)) = \exp(\lambda_2 t)V(x(0))$. Denote \bar{T}_1 and \bar{T}_2 as the average dwell time in model 1 and 2 respectively. If $\frac{\bar{T}_1}{\bar{T}_2} > \frac{\lambda_2}{\lambda_1}$, then the system is GAS. Next, we give a proof.

In an execution, denote the dwell time in mode i as $T_{i,1}, T_{i,2}, T_{i,3}, \dots$ and $T_i = \sum_{t=1}^{\infty} T_{i,t}$. Then,

$$\begin{aligned} V(x(T_1 + T_2)) &< V(x(0)) \prod_{t=1}^{\infty} \exp(-\lambda_1 T_{1,t}) \prod_{t=1}^{\infty} \exp(\lambda_2 T_{2,t}) \\ &= V(x(0)) \exp(-\lambda_1 \sum_{t=1}^{\infty} T_{1,t} + \lambda_2 \sum_{t=1}^{\infty} T_{2,t}) = V(x(0)) \exp(-\lambda_1 T_1 + \lambda_2 T_2) \end{aligned}$$

Because the average dwell time \bar{T}_1 and \bar{T}_2 satisfy that $\lambda_1 \bar{T}_1 > \lambda_2 \bar{T}_2$, as $T_1 + T_2$ goes to ∞ , we have $-\lambda_1 T_1 + \lambda_2 T_2$ goes $-\infty$, and thus $V(x(T_1 + T_2))$ goes to 0. Therefore the system is GAS.

Problem 2 (25 points). Automaton \mathcal{A} models a distributed graph coloring algorithm running on N processes with identifiers $0, \dots, N-1$ communicating over a given graph G . Each process i corresponds to a vertex of G and the set neighbors of i is denoted by $Nbrs(i)$. The maximum degree of G is D , that is, $\max_{i \in [N]} Nbrs(i) \leq D$. Each process i has a single state variable $color[i]$ which can take values from the set $\{0, \dots, D\}$. Initially, $color[i]$ is set arbitrarily. has a set of actions $recolor(i : [N])$ and the corresponding transitions are defined as follows: $recolor(i:[N])$ pre $j:Nbrs(i): color[i] = color[j]$ eff $color[i] := \text{choose } c:[D+1] \text{ — } A \text{ } j: Nbrs[i], colors[j] \neq c$ Show that an equilibrium state for \mathcal{A} is an assignment of the $color[i]$ variables such that each process has a color that is distinct from its neighbors. Show that all *fair* executions of \mathcal{A} converge to this equilibrium. [A fair execution is one in which every process gets a turn to recolor, infinitely often. That is, no process i for which there is a neighbor j with identical color is blocked from taking the action for ever.]

Solution If there exist a process i in a color that is the same as one of its neighbor j , then when i gets a turn to recolor, it will change its color, and thus the state of the system will change. Therefore such a state is not an equilibrium. Therefore an equilibrium state for the automaton is an

assignment of the $color[i]$ variables such that each process has a color that is distinct from its neighbors.

We define the ranking function f such that $f(color)$ is the number of processes that have colors same as one of its neighbors, i.e. $f(color) = \sum_{i=0}^{N-1} \mathbb{I}_i$, where $\mathbb{I}_i = \begin{cases} 1, & \text{if } \exists j \in Nbrs(i) : color[i] = color[j], \\ 0, & \text{otherwise.} \end{cases}$

After every recolor action, there are two possible changes on the ranking value. If the pre condition is not satisfied, the ranking value doesn't change. If the pre condition is satisfied for process i , then after this action, process i has a color different with all its neighbors, and thus \mathbb{I}_i changes from 1 to 0. For each neighbor $j \in Nbrs(i)$, if $\mathbb{I}_j = 0$ before this action, then after action $\mathbb{I}_j = 0$. If $\mathbb{I}_j = 1$ before this action, then after action $\mathbb{I}_j = 0$ or 1. Therefore, ranking value decreases at least one after a action where the pre condition is satisfied. In a fair execution, ranking function cannot stay at a non-zero value forever. After every process has been visited, the ranking function converges to 0, which means the system reaches the equilibrium.