Inversion of Linear Systems

Shreyas Sundaram

Coordinated Science Laboratory

and

Department of Electrical and Computer Engineering

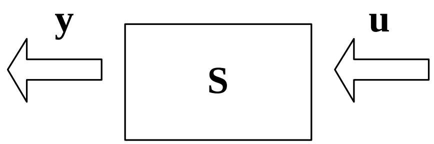
University of Illinois at Urbana-Champaign

Outline

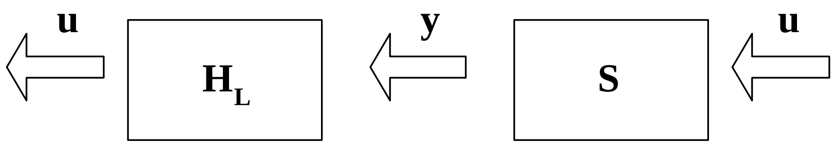
- Motivation
- Approaches to inversion
  - Massey-Sain algorithm
  - Moylan's algorithm
- Partial invertibility
- Future work

What is Inversion?

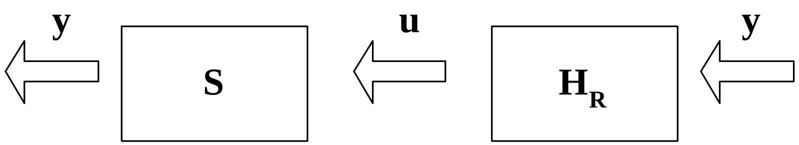
- Given the system



- **Left Inverse:** Outputs u when y is the input



- **Right Inverse:** Given a desired y, outputs the necessary u



- Here, we will focus on linear time-invariant systems

Why Study Inversion?

Inverse systems are useful in a variety of situations

- Left inverses:
  - Communication systems: determining input to a dynamical channel
  - Fault detection and identification
  - Coding theory
- Right inverses:
  - Feed-forward control
  - Disturbance decoupling

Approach 1: Transfer Function Matrix

- For SISO systems, inversion is easily handled through a transfer function approach

$$\frac{Y(z)}{U(z)} = \mathcal{S}(z) \Rightarrow \frac{U(z)}{Y(z)} = \frac{1}{\mathcal{S}(z)}$$

  - Inverse may be non-causal!
- For MIMO systems, problem is more complicated

$$\begin{bmatrix} Y_1(z) \\ Y_2(z) \\ \vdots \\ Y_p(z) \end{bmatrix} = \underbrace{\begin{bmatrix} \mathcal{S}_{11}(z) & \mathcal{S}_{12}(z) & \cdots & \mathcal{S}_{1m}(z) \\ \mathcal{S}_{21}(z) & \mathcal{S}_{22}(z) & \cdots & \mathcal{S}_{2m}(z) \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{S}_{p1}(z) & \mathcal{S}_{p2}(z) & \cdots & \mathcal{S}_{pm}(z) \end{bmatrix}}_{\mathcal{S}(z)} \begin{bmatrix} U_1(z) \\ U_2(z) \\ \vdots \\ U_m(z) \end{bmatrix}$$

  - Left inverse: find $\mathcal{H}_L(z)$ such that $\mathcal{H}_L(z)\mathcal{S}(z) = \mathbf{I}_m$ (requires $p \geq m$)
  - Right inverse: find $\mathcal{H}_R(z)$ such that $\mathcal{S}(z)\mathcal{H}_R(z) = \mathbf{I}_p$ (requires $m \geq p$)

Approach 2: State-Space Methods

State-space description of system $\mathcal{S}$ is

$$x[k+1] = Ax[k] + Bu[k]$$
$$y[k] = Cx[k] + Du[k]$$

where $x \in \mathbb{R}^n$, $y \in \mathbb{R}^p$ and $u \in \mathbb{R}^m$

- Will focus on left-invertibility in rest of talk
- **Definition:** System $\mathcal{S}$ has an inverse with delay L if $u[k]$ can be uniquely determined from $y[k], y[k+1], \cdots, y[k+L]$ (and perhaps $x[k]$)
  - Smallest such L is called "inherent delay" of system
- What are the conditions on matrices A, B, C, D for the system to be invertible?

Massey-Sain Algorithm (1)

- Appeared in IEEE Transactions on Automatic Control, vol. 14, 1969
- Find expressions for output in terms of input:

$$\begin{aligned} y[k] &= Cx[k] + Du[k] \\ y[k+1] &= Cx[k+1] + Du[k+1] \\ &= CAx[k] + CBu[k] + Du[k+1] \end{aligned}$$

- Continuing in this way, we get

$$\underbrace{\begin{bmatrix} y[k] \\ y[k+1] \\ y[k+2] \\ \vdots \\ y[k+L] \end{bmatrix}}_{Y_{[k,L]}} = \underbrace{\begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^L \end{bmatrix}}_{\mathcal{O}_L} x[k] + \underbrace{\left[\begin{array}{c|cccc} D & 0 & 0 & \cdots & 0 \\ CB & D & 0 & \cdots & 0 \\ CAB & CB & D & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{L-1}B & CA^{L-2}B & CA^{L-3}B & \cdots & D \end{array}\right]}_{M_L} \underbrace{\left[\begin{array}{c} u[k] \\ \hline u[k+1] \\ u[k+2] \\ \vdots \\ u[k+L] \end{array}\right]}_{U_{[k,L]}}$$

Massey-Sain Algorithm (2)

$$M_L = \left[\begin{array}{c|cccc} D & 0 & 0 & \cdots & 0 \\ CB & D & 0 & \cdots & 0 \\ CAB & CB & D & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{L-1}B & CA^{L-2}B & CA^{L-3}B & \cdots & D \end{array}\right]$$

- Notice that

$$\text{rank}(M_L) \leq \text{rank}(M_{L-1}) + m$$

- **Theorem:** System $\mathcal{S}$ has an inverse with delay L if and only if

$$\text{rank}(M_L) - \text{rank}(M_{L-1}) = m$$

Massey-Sain Algorithm (3)

$$M_L = \left[\begin{array}{c|cccc} D & 0 & 0 & \cdots & 0 \\ CB & D & 0 & \cdots & 0 \\ CAB & CB & D & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{L-1}B & CA^{L-2}B & CA^{L-3}B & \cdots & D \end{array}\right]$$

Proof of Theorem:

- If $\text{rank}(M_L) - \text{rank}(M_{L-1}) < m$, there exists a $U_{[k,L]}$ in the null-space of M_L, with $u[k] \neq 0$, which is indistinguishable from the all-zero input
- If $\text{rank}(M_L) - \text{rank}(M_{L-1}) = m$, the first m columns are linearly independent of each other, and of the rest of the columns in M_L.
  - There exists a matrix $\mathcal{K}$ such that

$$\mathcal{K} M_L = \left[\begin{array}{c|c} \mathbf{I}_m & \mathbf{0} \end{array}\right]$$

$$\Rightarrow \mathcal{K} Y_{[k,L]} = \mathcal{K}\mathcal{O}_L x[k] + u[k]$$

Massey-Sain Algorithm (4)

Construction of inverse:

- Input is given by

$$u[k] = -\mathcal{K}\mathcal{O}_L x[k] + \mathcal{K}Y_{[k,L]} \tag{1}$$

- Substitute into state-transition equation

$$\begin{aligned} x[k+1] &= Ax[k] + Bu[k] \\ &= Ax[k] - B\mathcal{K}\mathcal{O}_L x[k] + B\mathcal{K}Y_{[k,L]} \\ &= \left(A - B\mathcal{K}\mathcal{O}_L\right) x[k] + B\mathcal{K}Y_{[k,L]} \end{aligned} \tag{2}$$

- Equations (1) and (2) together form the state-space model of the inverse

Example of Massey-Sain Algorithm (1)

$$x[k+1] = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} x[k] + \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} u[k]$$

$$y[k] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x[k] + \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} u[k]$$

- Test for invertibility:

$$M_0 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \qquad \text{rank}(M_0) = 1$$

$$M_1 = \left[\begin{array}{cc|cc} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{array}\right], \qquad \text{rank}(M_1) = 3$$

- Since rank(M_1) − rank$(M_0) = 2$, system is invertible with delay 1

Example of Massey-Sain Algorithm (2)

- Find $\mathcal{K}$ such that $\mathcal{K}M_1 = \left[\begin{array}{c|c} \mathbf{I}_2 & \mathbf{0}_2 \end{array}\right]$

$$\Rightarrow \mathcal{K} = \begin{bmatrix} 0 & 0 & 1 & -1 \\ 0 & 1 & -1 & 1 \end{bmatrix}$$

- Inverse is given by

$$\begin{aligned} x[k+1] &= \left(A - B\mathcal{K}\begin{bmatrix} C \\ CA \end{bmatrix}\right)x[k] + B\mathcal{K}\begin{bmatrix} y[k] \\ y[k+1] \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} x[k] + \begin{bmatrix} 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} y[k] \\ y[k+1] \end{bmatrix} \\ u[k] &= -\mathcal{K}\begin{bmatrix} C \\ CA \end{bmatrix} x[k] + \mathcal{K}\begin{bmatrix} y[k] \\ y[k+1] \end{bmatrix} \\ &= \begin{bmatrix} 1 & -1 \\ -1 & 0 \end{bmatrix} x[k] + \begin{bmatrix} 0 & 0 & 1 & -1 \\ 0 & 1 & -1 & 1 \end{bmatrix}\begin{bmatrix} y[k] \\ y[k+1] \end{bmatrix} \end{aligned}$$

Massey-Sain Algorithm: Upper Bound

- When can we stop iterating?
- **Theorem:** If system is not invertible for $L = n$, it is not invertible at all
- Willsky later tightened upper bound to $L = n - q + 1$, where q is the nullity of D

Moylan's Algorithm (1)

- Appeared in IEEE Transactions on Automatic Control, vol. 22, 1977
- Define the matrix

$$M(\lambda) = \begin{bmatrix} A - \lambda I & B \\ C & D \end{bmatrix}$$

- **Theorem:** $\mathcal{S}$ is invertible if and only if rank$(M(\lambda)) = n + m$ for some real λ
- **Proof of Necessity:**
  - Assume rank$(M(\lambda)) < n + m$ for all λ
  - Then for any λ_i, there exist x_i and u_i such that

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x_i \\ u_i \end{bmatrix} = \begin{bmatrix} \lambda_i x_i \\ 0 \end{bmatrix}$$

  - It is possible to find scalars α_i such that $\sum_{i=0}^{n} \alpha_i x_i = 0$
  - The input $u[k] = \sum_{i=0}^{n} \alpha_i \lambda_i^k u_i$ results in $y[k] = 0$ for all k
  - Thus, the system is not invertible

Moylan's Algorithm (2)

Proof of sufficiency follows from the following construction:

- Consider the more general system

$$x[k+1] = Ax[k] + Bu[k] + v[k]$$
$$y[k] = Cx[k] + Du[k]$$

- Suppose D has rank $r < p$
  - There exists a non-singular $p \times p$ matrix Q_1 such that

$$Q_1 D = \begin{bmatrix} D_0 \\ 0 \end{bmatrix}, \quad Q_1 C = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix}$$

  - There exists a non-singular $(p-r) \times (p-r)$ matrix Q_2 such that

$$Q_2 C_2 = \begin{bmatrix} \tilde{C}_2 \\ 0 \end{bmatrix}$$

Moylan's Algorithm (3)

- Let

$$\tilde{y}[k] = \underbrace{\begin{bmatrix} I & 0 \\ 0 & Q_2 \end{bmatrix} Q_1}_{Q} y[k]$$

- This gives

$$\begin{bmatrix} \tilde{y}_1[k] \\ \tilde{y}_2[k] \\ \tilde{y}_3[k] \end{bmatrix} = \begin{bmatrix} C_1 \\ \tilde{C}_2 \\ 0 \end{bmatrix} x[k] + \begin{bmatrix} D_0 \\ 0 \\ 0 \end{bmatrix} u[k]$$

- Define similarity transformation $\tilde{x}[k] = Tx[k]$ such that

$$\tilde{C}_2 T^{-1} = \begin{bmatrix} 0 & I \end{bmatrix}$$

Moylan's Algorithm (4)

- New system:

$$\begin{bmatrix}\tilde{x}_1[k+1]\\ \tilde{x}_2[k+1]\end{bmatrix} = \begin{bmatrix}\tilde{A}_{11} & \tilde{A}_{12}\\ \tilde{A}_{21} & \tilde{A}_{22}\end{bmatrix}\begin{bmatrix}\tilde{x}_1[k]\\ \tilde{x}_2[k]\end{bmatrix} + \begin{bmatrix}\tilde{B}_1\\ \tilde{B}_2\end{bmatrix} u[k] + \begin{bmatrix}\tilde{v}_1[k]\\ \tilde{v}_2[k]\end{bmatrix}$$

$$\begin{bmatrix}\tilde{y}_1[k]\\ \tilde{y}_2[k]\\ \tilde{y}_3[k]\end{bmatrix} = \begin{bmatrix}\tilde{C}_{11} & \tilde{C}_{12}\\ 0 & I\\ 0 & 0\end{bmatrix}\begin{bmatrix}\tilde{x}_1[k]\\ \tilde{x}_2[k]\end{bmatrix} + \begin{bmatrix}D_0\\ 0\\ 0\end{bmatrix} u[k]$$

  - Note that $\tilde{y}_2[k] = \tilde{x}_2[k]$
- Define

$$z_1[k] = \tilde{y}_1[k] - \tilde{C}_{12}\tilde{y}_2[k]$$

$$z_2[k] = \tilde{y}_2[k+1] - \tilde{A}_{22}\tilde{y}_2[k] - \tilde{v}_2[k]$$

$$w[k] = \tilde{v}_1[k] + \tilde{A}_{12}\tilde{y}_2[k]$$

$$q[k] = \tilde{x}_1[k]$$

Moylan's Algorithm (5)

- System can be written as

$$q[k+1] = \hat{A}q[k] + \hat{B}u[k] + w[k]$$
$$z[k] = \hat{C}q[k] + \hat{D}u[k]$$

where

$$\hat{A} = \begin{bmatrix} \tilde{A}_{11} \end{bmatrix}, \quad \hat{B} = \begin{bmatrix} \tilde{B}_1 \end{bmatrix}$$
$$\hat{C} = \begin{bmatrix} \tilde{C}_{11} \\ \tilde{A}_{12} \end{bmatrix}, \quad \hat{D} = \begin{bmatrix} D_0 \\ \tilde{B}_2 \end{bmatrix}$$

- This system has $\hat{p} \leq p$ outputs and $\hat{n} \leq n$ states
- If rank$(\hat{D}) = m$, inverse is given by $\hat{D}^{\dagger}z[k] = \hat{D}^{\dagger}\hat{C}q[k] + u[k]$
- If rank$(\hat{D}) < m$, repeat procedure on new system
  - What if $\hat{p} < m$?

Moylan's Algorithm (6)

- Define

$$\hat{M}(\lambda) = \begin{bmatrix} \hat{A} - \lambda I & \hat{B} \\ \hat{C} & \hat{D} \end{bmatrix}$$

$$= \begin{bmatrix} \tilde{A}_{11} - \lambda I & \tilde{B}_1 \\ \tilde{C}_{11} & D_0 \\ \tilde{A}_{12} & \tilde{B}_2 \end{bmatrix}$$

- If rank$(M(\lambda)) = n + m$ for $\lambda = \lambda_i$, then

$$\text{rank}(\hat{M}(\lambda_i)) = \hat{n} + m$$

- This implies $\hat{p} \geq m$!

Moylan's Algorithm (7)

The following can be proved in a similar manner:

- System $\mathcal{S}$ has a stable inverse if and only if rank$(M(\lambda)) = n + m$ for all $|\lambda| > 1$
- System $\mathcal{S}$ is invertible with unknown initial state if and only if rank$(M(\lambda)) = n + m$ for all λ

Partial Invertibility (1)

What if we only want to invert some of the inputs?

- Suppose $u[k] = \begin{bmatrix} u_1[k] \\ u_2[k] \end{bmatrix}$, with $u_1[k] \in \mathbb{R}^{m_1}$, $u_2[k] \in \mathbb{R}^{m_2}$
  - Partition B and D as $B = \begin{bmatrix} B_1 & B_2 \end{bmatrix}$, $D = \begin{bmatrix} D_1 & D_2 \end{bmatrix}$
- System $\mathcal{S}$ becomes

$$
\begin{aligned}
x[k+1] &= Ax[k] + B_1 u_1[k] + B_2 u_2[k] \\
y[k] &= Cx[k] + D_1 u_1[k] + D_2 u_2[k]
\end{aligned}
$$

- What are the conditions on the system such that $u_1[k]$ is invertible?

Partial Invertibility (2)

Suppose $x[k]$ is unknown

- Can we invert $u_1[k]$ based only on $y[k], y[k+1], \ldots, y[k+L]$, for some L?
- The response of system $\mathcal{S}$ over $L+1$ time units is given by

$$\begin{bmatrix} y[k] \\ y[k+1] \\ \vdots \\ y[k+L] \end{bmatrix} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^L \end{bmatrix} x[k] + \begin{bmatrix} D_1 & 0 & \cdots & 0 \\ CB_1 & D_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{L-1}B_1 & CA^{L-2}B_1 & \cdots & D_1 \end{bmatrix} \begin{bmatrix} u_1[k] \\ u_1[k+1] \\ \vdots \\ u_1[k+L] \end{bmatrix}$$

$$+ \begin{bmatrix} D_2 & 0 & \cdots & 0 \\ CB_2 & D_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{L-1}B_2 & CA^{L-2}B_2 & \cdots & D_2 \end{bmatrix} \begin{bmatrix} u_2[k] \\ u_2[k+1] \\ \vdots \\ u_2[k+L] \end{bmatrix}$$

Partial Invertibility (3)

- Previous expression can be written more compactly as

$$\underbrace{\begin{bmatrix} y[k] \\ y[k+1] \\ y[k+2] \\ \vdots \\ y[k+L] \end{bmatrix}}_{Y_{[k,L]}} = \underbrace{\left[\begin{array}{c|cccccc} D_1 & D_2 & 0 & \cdots & 0 & C \\ CB_1 & CB_2 & D & \cdots & 0 & CA \\ CAB_1 & CAB_2 & CB & \cdots & 0 & CA^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ CA^{L-1}B_1 & CA^{L-1}B_2 & CA^{L-2}B & \cdots & D & CA^L \end{array}\right]}_{J_L} \underbrace{\begin{bmatrix} u_1[k] \\ \hline u_2[k] \\ u[k+1] \\ \vdots \\ u[k+L] \\ x[k] \end{bmatrix}}_{I_{[k,L]}}$$

- Partition J_L as $J_L = \left[\begin{array}{c|c} \mathbf{\Gamma}_L & \mathbf{\Psi}_L \end{array}\right]$
- **Theorem:** $u_1[k]$ is invertible with delay L and unknown state $x[k]$ if and only if

$$\text{rank}(J_L) - \text{rank}(\mathbf{\Psi}_L) = m_1 \quad .$$

Partial Invertibility (4)

Proof of Theorem:

- If rank(J_L) – rank$(\mathbf{\Psi}_L) < m_1$, there exists a $I_{[k,L]}$ in the null-space of J_L, with $u_1[k] \neq 0$, which is indistinguishable from the all-zero input
- If rank(J_L) – rank$(\mathbf{\Psi}_L) = m_1$, the first m_1 columns are linearly independent of each other, and of the rest of the columns in J_L.
  - There exists a matrix $\mathcal{K}$ such that

$$\mathcal{K} J_L = \left[\begin{array}{c|c} \mathbf{I}_{m_1} & \mathbf{0} \end{array} \right]$$

$$\Rightarrow \mathcal{K} Y_{[k,L]} = u_1[k]$$

Example: Fault Detection (1)

- Consider a model of the F-8 aircraft (Teneketzis et al.)

$$\dot{x}(t) = \underbrace{\begin{bmatrix} -0.01357 & -32.2 & -46.3 & 0 \\ 0.00012 & 0 & 1.214 & 0 \\ -0.0001212 & 0 & -1.214 & 1 \\ 0.00057 & 0 & -9.01 & -0.6696 \end{bmatrix}}_{A} x(t) + \underbrace{\begin{bmatrix} -0.433 \\ 0.1394 \\ -0.1394 \\ -0.1577 \end{bmatrix}}_{B_u} u(t) + \underbrace{\begin{bmatrix} -46.3 \\ 1.214 \\ -1.214 \\ -9.01 \end{bmatrix}}_{B_d} d(t)$$

$$y(t) = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}}_{C} x(t)$$

where $u(t)$ is the elevator deflection in radians, and $d(t)$ is the wind disturbance

- We wish to detect failures in the elevator
  - Invert $u(t)$ and compare to the specified value

Example: Fault Detection (2)

- Perform test for inversion:

$$J_0 = \left[\begin{array}{c|cc} D_u & D_d & C \end{array} \right], \qquad \text{rank}(J_0) - \text{rank}(\boldsymbol{\Psi}_0) = 0$$

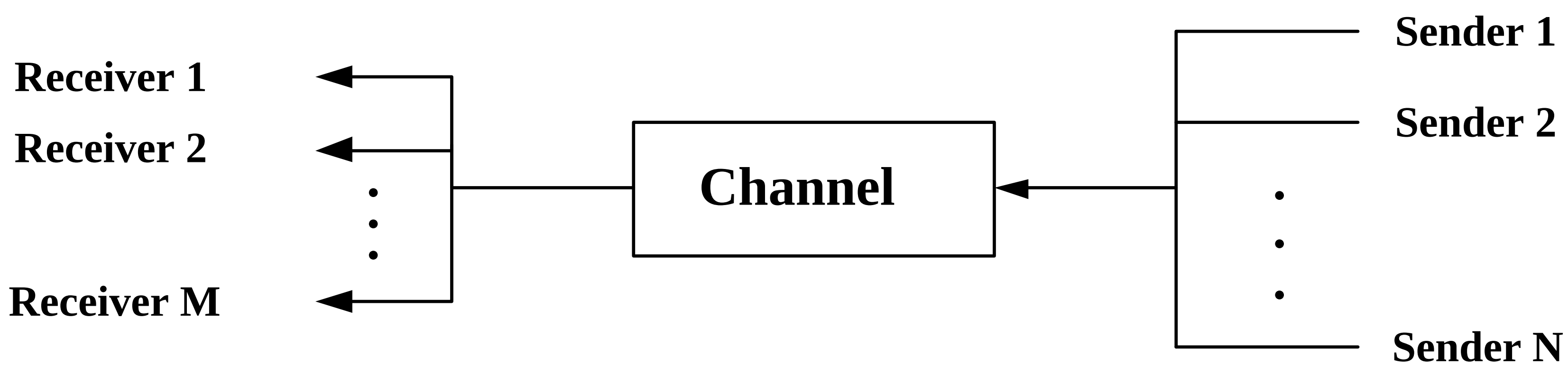
$$J_1 = \left[\begin{array}{c|cccc} D_u & D_d & 0 & 0 & C \\ CB_u & CB_d & D_u & D_d & CA \end{array} \right], \qquad \text{rank}(J_1) - \text{rank}(\boldsymbol{\Psi}_1) = 1$$

- Find $\mathcal{K}$ such that $\mathcal{K} J_1 = \left[\begin{array}{c|cccc} 1 & 0 & 0 & 0 & 0 \end{array} \right]$
- Input is given by

$$u(t) = \underbrace{\begin{bmatrix} -0.0018 & -6.5936 & -0.2048 & -7.8097 \end{bmatrix}}_{\mathcal{K}} \begin{bmatrix} y(t) \\ \dot{y}(t) \end{bmatrix}$$

- Thus $u(t)$ is invertible with “delay” 1

Example: Multiplexing in Communication Channels



- Suppose multiple users broadcast through a dynamic channel
  - Is it possible for Receiver i to only decode messages from a certain subset of senders?

Future Work

- Partial inversion:
  - Finish proof of upper bound on L for partial inversion with unknown state
  - Study construction of partial inverses for systems with known initial conditions
  - Develop partial inverses of minimum dimension
- Investigate invertibility of hybrid/switched systems
- Design system inverses that are robust to parameter variations