Multi-Input Ground Vehicle Control Using Quadratic Programming Based Control Allocation

John Plumlee

Directed by
David M. Bevly & A. Scottedward Hodel

Auburn University
Outline

- Motivation and Contributions
- The Problem and Objectives
- Vehicle Models
- Controller Design
- Control Allocation
- Background
- Simulation Results
- Conclusions
- Future Work
Motivation

- Multiple inputs are becoming more economically feasible
  - rear steer, differential braking, active suspension, active stabilizer, individual torque control of each wheel.

- Highway vehicles, Military applications

- Vehicle Stability Control systems – obstacle avoidance, rollover prevention
Recent Publications


Contributions

- A control strategy to provide intelligent combinations of input commands to a ground vehicle in order to accomplish multiple objectives.

- A method for the application of Control Allocation techniques to a ground vehicle with coupled dynamics

- Demonstrate ability to maintain successful tracking in the event of a failure.
The Problem

We want to track a yaw rate trajectory with a 4 wheeled ground vehicle while minimizing the sideslip angle.

Question: But if there are multiple inputs available, How do you choose which to use and how much?

Answer: Control Allocation!!
Control Allocation (CA)

- CA involves generating a set of effector commands that produce a desired control effect while minimizing the control effort.

- A CA approach is generally used for a redundant set of effectors.

- CA allows for reconfiguration in the event of an effector failure.
Vehicle Model (nonlinear)

Assumptions made:

- The vehicle runs at constant velocity
- No longitudinal weight transfer (pitch is neglected)
- Constant roll center.
- A Pacejka tire model was used to represent nonlinear tire behavior.
Vehicle Model (linear)

The vehicle model in state space form:

\[
\begin{bmatrix}
\dot{\beta} \\
\dot{r}
\end{bmatrix} =
\begin{bmatrix}
-\frac{C_o}{mV} & -\frac{C_1}{mV^2} & -1 \\
-\frac{C_1}{I_z} & -\frac{C_2}{VI_z}
\end{bmatrix}
\begin{bmatrix}
\beta \\
r
\end{bmatrix} + Bu
\]

Linearized about constant velocity
Vehicle Model (linear)

3-Input vehicle model:

\[
Bu = \begin{bmatrix}
\frac{C_{af}}{mV} & 0 & 0 \\
\frac{C_{af}}{a} \frac{T_f}{I_z} & \frac{T_f}{2I_z} & \frac{T_r}{2I_z} \\
\end{bmatrix}
\begin{bmatrix}
\delta \\
\Delta F_{xf} \\
\Delta F_{xr}
\end{bmatrix}
\]

\[\delta\] = steering angle
\[\Delta F_{xf}\] = differential braking force on front axle
\[\Delta F_{xr}\] = differential braking force on rear axle
4-Input vehicle model:

\[
Bu = \begin{bmatrix}
\frac{C_{\alpha f}}{mV} & \frac{C_{\alpha r}}{mV} & 0 & 0 \\
\frac{C_{\alpha f}}{2I_z} & -b & \frac{T_f}{2I_z} & \frac{T_r}{2I_z}
\end{bmatrix}
\begin{bmatrix}
\delta_f \\
\delta_r \\
\Delta F_{xf} \\
\Delta F_{xr}
\end{bmatrix}
\]

- \( \delta_f \) = front steering angle
- \( \delta_r \) = rear steering angle
- \( \Delta F_{xf} \) = differential braking force on front axle
- \( \Delta F_{xr} \) = differential braking force on rear axle
Vehicle Model (linear)

6-Input vehicle model:

\[
Bu = \begin{bmatrix}
\frac{C_{af}}{mV} & \frac{C_{ar}}{mV} & 0 & 0 & 0 & 0 \\
\frac{C_{af}}{I_z} & -b \frac{C_{ar}}{I_z} & -\frac{T_f}{2I_z} & \frac{T_f}{2I_z} & -\frac{T_r}{2I_z} & \frac{T_r}{2I_z} \\
\end{bmatrix}
\begin{bmatrix}
\delta_f \\
\delta_r \\
F_{xfR} \\
F_{xfL} \\
F_{xrR} \\
F_{xrL}
\end{bmatrix}
\]

\[\delta_f = \text{front steering angle}\]
\[\delta_r = \text{rear steering angle}\]
\[F_x = \text{force at each wheel}\]
**Vehicle Model**

**Effector limitations**

Front and rear steering angle: $\pm 0.5 \, rad$

Longitudinal wheel force:

\[
F_{tot} = \mu F_z
\]

\[
F_{x}^{\pm} = \sqrt{F_{tot}^2 - F_y^2}
\]

where $\mu = 0.8$
Controller Design Approach

Controller has 2 main tasks:

1. Generate Control Effort \rightarrow State Feedback
2. Generate Effector Commands \rightarrow Control Allocation

Controller Diagram:
- ref
- State Feedback
- Control Allocation
- Ground Vehicle

\( \bar{u} \rightarrow u \)

“virtual” command
LQR gains are designed for the following modified system:

\[
\begin{bmatrix}
\dot{\beta} \\
\dot{r} \\
\dot{\psi}
\end{bmatrix} =
\begin{bmatrix}
-\frac{C_o}{mV} & -\frac{C_1}{mV^2} - 1 & 0 \\
-\frac{C_1}{I_z} & -\frac{C_2}{VI_z} & 0 \\
0 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
\beta \\
r \\
\psi
\end{bmatrix} +
\begin{bmatrix}
1 & 0 \\
0 & 1 \\
0 & 0
\end{bmatrix}
\begin{bmatrix}
\bar{u}_\beta \\
\bar{u}_r
\end{bmatrix}
\]

This model is used solely for generating the virtual command effort.

The gain matrix \( K \) is applied to the error vector to produce the overall desired effect.

\[
\begin{bmatrix}
\bar{u}_\beta \\
\bar{u}_r
\end{bmatrix} = -K_{2 \times 3}
\begin{bmatrix}
\beta_{error} \\
r_{error} \\
\psi_{error}
\end{bmatrix}
\]
Physical effector commands are generated by the Control Allocation routine.

Given the virtual command $\bar{u}$, solve for the effector commands $u$.

$$Bu = \bar{u}$$

$$u^- \leq u \leq u^+$$
Control Allocation background
Traditional Approaches

- Direct CA involves the calculation of the Attainable Set of virtual commands
  1. Determine AS
  2. Scale Desired Effect Vector to boundary of AS
  3. Solve for AS and virtual command vector intersection
  4. Scale by inverse of Step 2 scaling ($u$, effector commands)

- Generalized Inverse
  - weighted pseudo inverse
  \[ u = Q^{-1}B^T \left[ BQ^{-1}B^T \right]^{-1} \bar{u} \]
Control Allocation background
Optimization Techniques

Least Squares

\[
\begin{align*}
\min_u & \quad \frac{1}{2} u^T Qu + c^T u \\
\text{s.t.} & \quad Bu = \bar{u} \\
& \quad -\infty \leq u \leq +\infty
\end{align*}
\]

Linear Programming

\[
\begin{align*}
\min_u & \quad c^T u \\
\text{s.t.} & \quad Bu = \bar{u} \\
& \quad -u \leq u \leq +u
\end{align*}
\]

Quadratic Programming

\[
\begin{align*}
\min_u & \quad \frac{1}{2} u^T Qu + c^T u \\
\text{s.t.} & \quad Bu = \bar{u} \\
& \quad -u \leq u \leq +u
\end{align*}
\]

\(Q\) and \(c\) are weighting matrices

No inequality constraints

Linear cost function

Nonlinear cost function and inequality constraints
Command Generation: slack variable

Solution: The coupled dynamics make it impossible to perfectly satisfy both parts of the virtual command. Therefore, we need to make a compromise between the infeasible portions of the yaw rate tracking and real effectors. We can do this by adding a pseudo-input \( \nu \).

\[
\begin{bmatrix}
\dot{\beta} \\
\dot{r}
\end{bmatrix} = \begin{bmatrix}
-\frac{C_o}{mV} & -\frac{C_1}{mV^2} & -1 \\
-\frac{C_1}{I_z} & -\frac{C_2}{VI_z}
\end{bmatrix} \begin{bmatrix}
\beta \\
r
\end{bmatrix} + \begin{bmatrix}
\frac{C_{af}}{mV} \\
\frac{C_{af}}{B} \\
\frac{a}{I_z} \\
\frac{C_{af}}{I_zV}
\end{bmatrix} \begin{bmatrix}
u \\
\frac{\kappa \beta}{mV} \\
-\kappa \beta \\
-\kappa \beta
\end{bmatrix} \begin{bmatrix}
0 \\
0 \\
\frac{T_f}{2I_z} \\
\frac{T_r}{2I_z}
\end{bmatrix} \begin{bmatrix}
\delta_f \\
\delta_r \\
\Delta F_{xf} \\
\Delta F_{xr}
\end{bmatrix}
\]

Reasoning: The CA law could use \( \nu \) to satisfy the infeasible portions of the yaw rate tracking while leaving the yaw rate tracking up to the real effectors.
Command Generation:
slack variable

Plan: Choose the effectiveness values $\kappa_\beta$ and $\kappa_r$ so that the DC gain from $\nu$ to $r$ is zero.

Step 1: Calculate the Transfer Function.
Step 2: Set the numerator equal to zero.
Step 3: Choose $\kappa_\beta = 1$ and solve for $\kappa_r$.

\[
\frac{R(s)}{N(s)} = \kappa_\beta \frac{mI_z V^2 s^{\kappa_\beta}}{mI_z V^2 s^2 + \kappa_\beta \frac{I_0 V}{C_0} - \kappa_\beta \frac{C_1 m V^2}{m V^2 - C_1}}
\]

\[
\kappa_r = \frac{C_0 I_z}{C_1 m V^2 - C_1}
\]
The QP problem is set up as follows:

\[
\begin{align*}
\min u, \nu : & \quad \frac{1}{2} \begin{bmatrix} u \\ \nu \end{bmatrix} \begin{bmatrix} Q & 0 \\ 0 & Q_\nu \end{bmatrix} \begin{bmatrix} u \\ \nu \end{bmatrix} + c^T \begin{bmatrix} u \\ \nu \end{bmatrix} \\
\text{subject to:} & \quad B \begin{bmatrix} \frac{1}{C_1 m V} & C_0 I_z \end{bmatrix} \begin{bmatrix} u \\ \nu \end{bmatrix} = \begin{bmatrix} \bar{u}_\beta \\ \bar{u}_r \end{bmatrix} \\
& \quad \begin{bmatrix} u^- \\ \nu^- \end{bmatrix} \leq \begin{bmatrix} u \\ \nu \end{bmatrix} \leq \begin{bmatrix} u^+ \\ \nu^+ \end{bmatrix}
\end{align*}
\]

A large penalty is placed on \( \nu \) to reduce its use.

Nonrestrictive inequality constraints
Simulation

- The vehicles are simulated traveling at constant velocities of 45, 55, and 65mph.

- Scenarios examined:
  1) Nominal – no failures experienced.
  2) Failure – steering angle jams at 2.25s.

- Penalty on pseudo effector: $Q_v = 10^3, 10^7$.

- The desired yaw rate trajectory simulates a double lane change maneuver.
Weighting the pseudo effector
3-Input, Nominal, 55mph
Rear Steering Issue
4-input, Nominal, 55mph
Nominal Case Performance
4-input 55mph QP

Yaw Rate vs. Time

Steering Input vs. Time

Sideslip Angle vs. Time

Differential braking force vs. Time
Failure Case Performance
3-input, 55mph

**Yaw Rate vs. Time**
- Actual
- Desired
- Error

**Steering Input vs. Time**
- Yaw Rate (degrees/second)
- Steering Angle (degrees)

**Sideslip Angle vs. Time**
- Yaw Rate (degrees/second)
- Sideslip Angle (degrees)

**Differential braking force vs. Time**
- Front
- Rear
- front limit
- rear limit
Failure Case Performance
4-input, 55mph

- **Yaw Rate vs. Time**
  - Actual
  - Desired
  - Error

- **Steering Input vs. Time**
  - Front
  - Rear

- **Sideslip Angle vs. Time**

- **Differential braking force vs. Time**
  - Front
  - Rear
  - front limit
  - rear limit
Failure Case Performance
6-input, 55mph

- Yaw Rate vs. Time
- Steering Input vs. Time
- Sideslip Angle vs. Time
- Wheel Force vs. Time
Conclusions

- The addition of the pseudo effector successfully allows the sideslip angle to vary so that opposing commands would be minimized in the presence of conflicting objectives.

- Large $Q_v$ produced better sideslip and yaw rate tracking in the nominal case at high speeds.

- Small $Q_v$ gave the best tracking of both objectives for the failure case.

- The proposed QP based CA strategy provides an intelligent reconfiguration of control effort in the event of a failure while respecting the effector limits.
Future Work

- Study controller performance for maneuvers other than a lane change.
- Investigate ways to alter the frequency content of the QP solution in the presence of noisy state estimates.
- Simulate driver in the loop. Evaluate performance.
- Development of an interior point algorithm for finding a fast solution to small scale QP problems.
- Actual implementation
Questions

Vehicle Dynamics Lab Reminder: I am Dr Bevly's 1st student to graduate. The difficulty of your questions will be setting the standard for your own defense.
Tire Force Estimation


  - Can give online estimates of $\mu$ and $F_y$ using Extended Kalman filtering and Bayesian Hypothesis Selection


  - Presents a method for estimating the maximum tire-road friction during braking. Also gives a method for estimating $F_z$ as an alternative to direct measurement.
Fault Detection

  
  - Uses nonlinear observer design techniques to construct a complete fault detection system for vehicle sensors including steering angle sensor, and brake pressure sensor.

  
  - Brake pressure sensor and braking actuator failure detection.
Performance with incorrect parameters, 4-input

- Controller designed with 2000 Honda Accord parameters
- 2001 Chevy Blazer parameters were used for simulation
- Sideslip error is significantly larger
In reality, exact state estimates are not available.
1. Use Kalman filter to get rid of some noise
2. Set up QP problem to minimize high frequency commands.
Commands w/ noisy state estimates
Frequency Weighted Sign Preserving QP

Steering Input vs. Time

Differential braking force vs. Time

Yaw Rate vs. Time

Sideslip Angle vs. Time
Pseudo effector command
4-input, failure, Qv=1e3
Friction circle
4-input, 55mph, failure