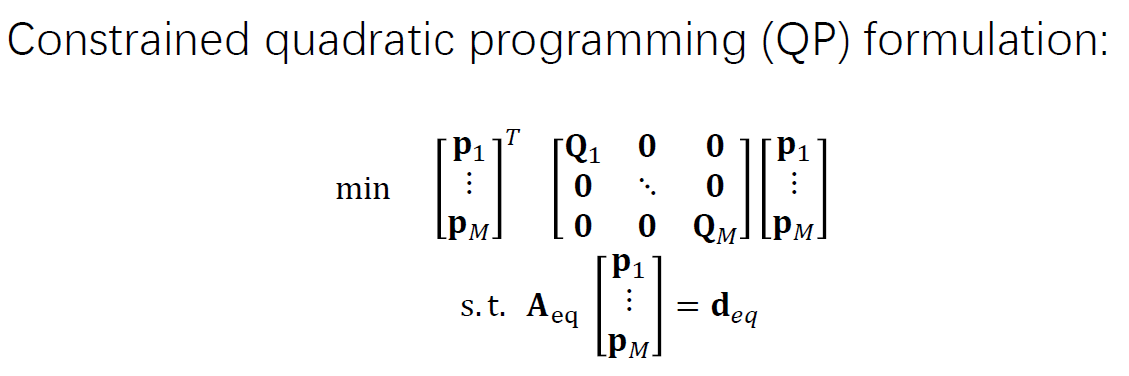
# QP Solver

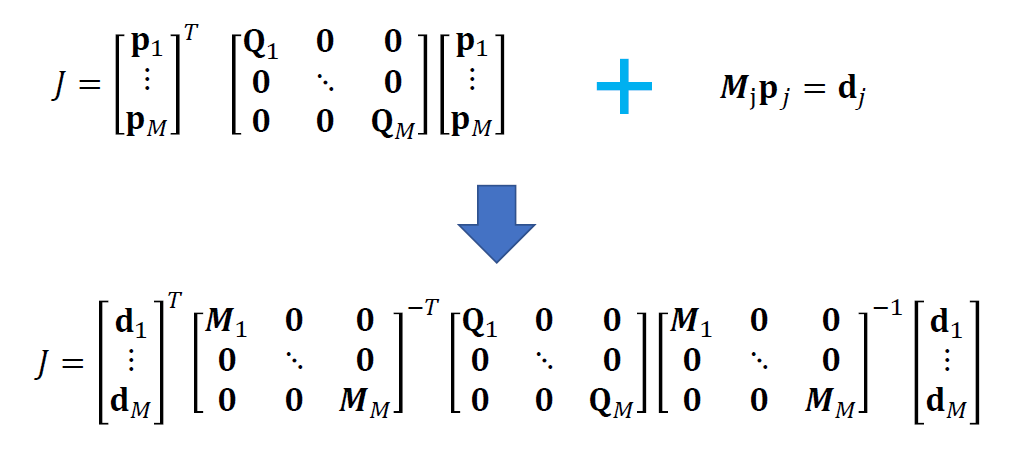
Key idea is to write the problem into an constrained quadratic programming formulation, like:



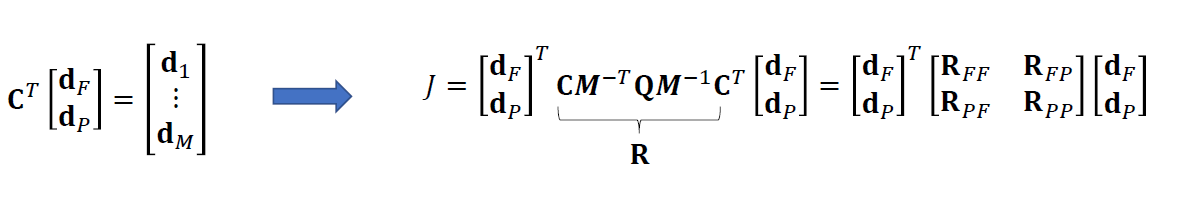
Derivative constraints and continuity constraints are put into Aeq and deq. Then it is solved by the QP solver in MATLAB.

# Closed-form Solver

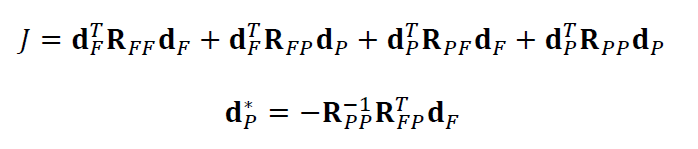
The problem of finding an optimal polynomial can be transformed into finding optimal v, a at each waypoint, by the way of decision variable.



The next step is separating free and constrained variables in order to get a closed form solution.



Finally, it is solving a optimal problem.



To understand it, we formulate a simple problem as three waypoints with only p and v two variables in each segment. Therefore, we have:

# Simulation Results and Analysis

Minimum snap trajectory generated by QP solver and closed-form solution are shown in the following picture.



Analysis of two solutions:

|  |  |
| --- | --- |
| QP solver | Expensive in iteration for a numerical solution |
| Closed-form solution | Expensive in matrix operation |

# Implementation Notes

The difficulty lies in making the matrix correct. Patience is important!