An Exploration of "Gauss's Principle With Inequality Constraints for Multiagent Navigation and Control" by Boyang Zhang and Henri P. Gavin

Gabriel Dimonde

March 30, 2025

1 Section II - Preliminaries

Here, we review Section II of "Gauss's Principle With Inequality Constraints for Multiagent Navigation and Control" by Boyang Zhang and Henri P. Gavin.

Next, I define variables and what their meanings are relating to multiagent navigation and control as if I were to define variables for a MATLAB or Octave program. Please keep in mind that the intention, first of all, is to build a GPLC control framework based pendulum, so the variables set here and their explanations are meant both to cover their generality and to provide understanding required for simulation of a simple pendulum.

N is the number of agents:

$$N = 1 \tag{1}$$

It equals one in this case, due to the requirements of a pendulum (an ideally single mass) and for defining the size of required matrices / vectors.

It is worth noting that the following matrix and vector sizes are dependent on N, but also the degree of freedom (DOF) of each mass particle (or agent). For the rest of this summary DOF is understood to be two, and is not included as a variable in matrix and vector dimensionality specifications, it will be included as "2".

M is a Symmetric Positive Definite (SPD) Mass matrix, of size 2N x 2N. A symmetric matrix requires that it be equal to its transpose. So, $M = M^T$. A symmetric matrix M is a positive definite matrix if the real number $x^T M x$ is always positive for $x \in \mathbb{R}^{2N}$.

$$M = \begin{bmatrix} m & 0 \\ 0 & m \end{bmatrix} \tag{2}$$

So, in this case, M obviously satisfies the requirement that it be equal to its transpose. As for the proof that M, in its current configuration, can also be positive definite, we must consider what m would result in the real number $x^\top M x$ as always positive for $x \in \mathbb{R}^{2N}$.

For $N = 1$, $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$. Evaluating the real number expression $x \top M x$ symbolically results in:

$$m(x_1^2 + x_2^2) \tag{3}$$

showing that for $x_1 \in \mathbb{R}$ and $x_2 \in \mathbb{R}$ Expression 3 > 0 when $m > 0$.

a is the unconstrained accelerations related to the net forces due to internal and external actions through Newton's Second Law: $a \in \mathbb{R}^{2N}$

f is the net forces to the internal and external actions through Newton's Second Law: $f = Ma$

$q(t)$ is the coordinate position vector that varies with time: $q(t) \in \mathbb{R}^{2N}$

$\ddot{q}(t)$ is the second derivative of q with respect to time. $\ddot{q}(t) \in \mathbb{R}^{2N}$

Now, switching contexts to quadratic programming (QP) methods. The objective of QP is to find an n-dimensional vector $\ddot{q}(t)$ that will minimize Z:

$$Z = \frac{1}{2}\ddot{q}^\top M\ddot{q} - f^\top\ddot{q} \tag{4}$$

subject to constraints:

$$\mathbf{A}\ddot{\mathbf{q}} \preccurlyeq \mathbf{b} \tag{5}$$

Generally, the constraints to be imposed on the above minimization would enforce geometric relationships among coordinate positions $q(t)$ and velocities $\dot{q}$. Their form, in general, would be:

$$\mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{0} \tag{6}$$