Component Selection in Geometric Programming

Edward Burnell

November 2016

1 Motivation

We find it useful to look at models of engineering systems as consisting of two types of equations: those describing a family of possible designs (parametrization), and those that describe the physics and performance of a parametrization. An electrochemical cell, for instance, could be modeled as a simple voltage source with a fixed energy density, or as a series of chemical reaction interfaces and rates; each model offers different design variables, and requires different models to tie those variables to the battery's weight and energy.

If new parametrizations open the design space up to new possibilties, so do new types of parametrization. For systems where the physics is unknown or diffcult to model, existing designs can be parametrized implicitly by fitting their observed physics. **?** Airfoil geometry, for instance, is difficult to parametrize in a convex fashion, but a convex fit can be made to the lift and drag of a family of airfoils, such that any point allowed by that fit is an implicit interpolation between airfoils. **?**

What, however, is to be done with subsystems for which neither detailed parametrization or implicit interpolation will do? This could be because the properties of existing designs are not smooth, because the desired outcome of a model is not an design point fit to observed physics, but the selection of a particular option. When you want to buy a small battery pack from a list of options, a model which selects an implied battery may be of as little use as one that tells you the optimal anode surface area.

In this paper we present a method for making efficient optimal selections via integer programming on continuous GP relaxations of the selection problem. The proximity of the interpolated design to each available design is an explicit variable in the model, and so each GP solve selects the globally optimal blend of available options. A heuristic motivated by engineering design process is then used to move from blended relaxations to the globally optimal selection.

2 Formulation of the selection problem

We consider all formulations in the context of choosing a battery pack with three properties: mass m, maximum power output P, and total energy E.

2.1 LP formulation

Say we have N different battery packs to select from, and to represent our choice we construct the vector S , to have exactly one element of value 1, while the rest are 0; the index of this 1 represents our choice of battery.

$$S_i \in \{0, 1\} \ \forall i$$

$$\sum_{i=1}^{N} S_i = 1$$

With this vector and the lists of the selection option's masses m_{opt}, powers P_{opt}, and energies E_{opt}, we can determine the properties of our selection with inner products:

$$S^T m_{opt} = m$$

$$S^T P_{opt} = P$$

$$S^T E_{opt} = E$$

The sharing of S across properties ensures that each property corresponds to the same selection.

2.2 LP Mixtures

Relaxing this model to be LP-compatible is simple: let S be continuous.

$$0 \leq S_i \leq 1 \ \forall i$$

But what does it mean for a value in the selection vector to be between 0 and 1? Instead of selecting a single battery, it can now select 70% of one battery and 30% of another: an interpolated battery. In Figure 1 you can see the triangle of interpolations this opens up in an example dataset with $N = 3$. When comparing mass and power, or mass and energy, this interpolation is optimistic: the best answer is always to select a mixture between the lightest and heaviest batteries. For the power / energy tradeoff, however, the interpolation is pessimistic, and will always select a spot between the small and medium, or medium and large batteries.

2.3 Adapting the LP formulation in GP

To formulate the above as a GP model we'll need to turn equalities of sums into inequalities with a monomial greater than a posynomial. Additionally, GP variables must be strictly greater than zero. If we were to turn the LP formulation above directly into a GP we would thus end up with

$$\sum_{i=1}^{N} S_i \leq 1$$

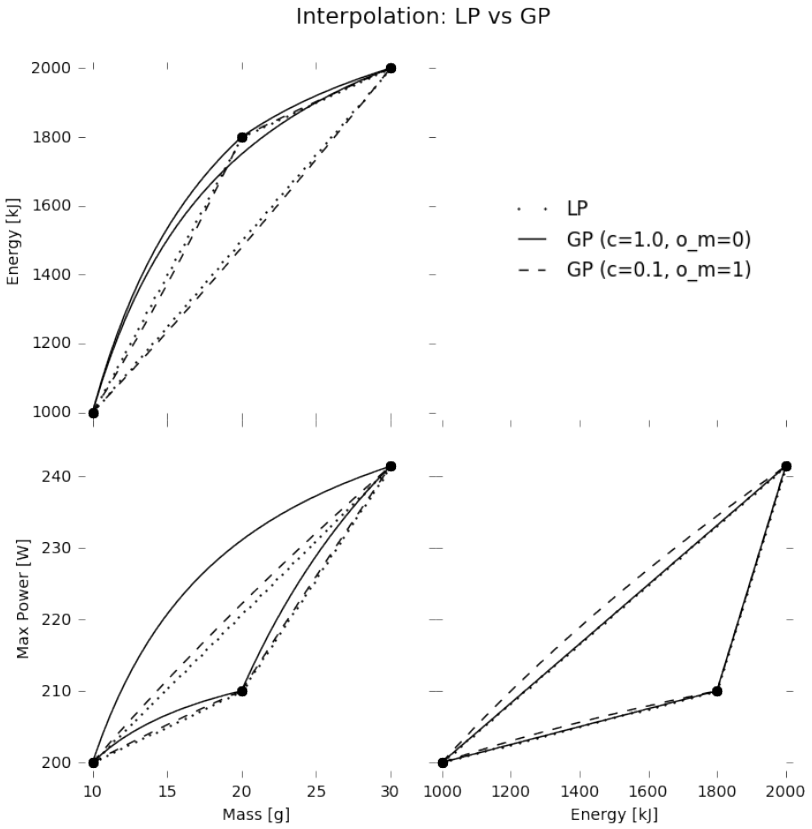

Figure 1: Linear interpolation between the smallest and largest battery is optimistic in power per mass, pessimistic in energy per mass, and optimistic in power per energy.

$$\frac{\epsilon}{n-1} \leq S_i \leq 1 - \epsilon \; \forall i$$

and

$$S^T m_{opt} \leq m$$
$$S^T P_{opt} \leq P$$

but this formulation will not work. For P, the optimization's push to increase it is unconstrained; P can happily go to $+\infty$ without worrying about the value of $S^T m_{opt}$. For m, the optimization's push to reduce it will keep it equal to $S^T m_{opt}$, but every value of S will happily go to its lower bound, such that $m \leftarrow \epsilon$. Another transformation will fix this mismatch of optimization pressures.

2.4 Exponentiaion

Let $\gamma_i = e^{S_i}$ and exponentiate both sides of equations above.

$$1 \leq \gamma_i \leq e \ \forall i$$

$$e^{\Sigma S_i} = \Pi \gamma_i = e$$

Where the constraints bounding the values and sum of S are now monomials equality constraints.

To avoid exponents that are united quantities, we'll define

$$|P_{opt}| = \left(\frac{P_{opt}}{\max(P_{opt})} + o_P \right)^{c_P}$$

and similarly constrain

$$|P|^{\frac{1}{c_P}} \geq \frac{P}{\max(P_{opt})} + o_P$$

This lets us use only the dimensionless and offset variables $|P|$ and $|P_{opt}|$ in exponents. The mixing of option properties then also becomes a monomial,

$$e^{S^T |P_{opt}|} = \Pi \gamma_i^{|P_{opt}|_i} = e^{|P|}$$

and we can approximate e^P with M terms of its taylor series expansion

$$e^{|P|} \geq 1 + \sum_{k=1}^{M} \frac{|P|^k}{k!}$$

to arrive at a GP-compatible inequality.

$$\Pi \gamma_i^{|P_{opt}|_i} \geq 1 + \sum_{k=1}^{M} \frac{|P|^k}{k!}$$

2.5 GP mixtures

The constraint

$$\Pi \gamma_i^{|P_{opt}|_i} \geq 1 + \sum_{k=1}^{M} \frac{|P|^k}{k!}$$

has the right direction for P; the optimization's push to increase P will keep the inequality tight. However,

$$\Pi \gamma_i^{|m_{opt}|_i} \geq 1 + \sum_{k=1}^{M} \frac{|m|^k}{k!}$$

will not work as a constraint, for m will quite happily go to 0.

The normal geometric programming technique of inverting a variable to flip its pressures applies here, for we can fix this constraint interpolating the value of $\frac{1}{m}$ instead:

$$\Pi\gamma_i^{|m_{opt}|_i^{-1}} \geq 1 + \sum_{k=1}^{M} \frac{1}{k!\,|m|^k}$$

As can be seen in the solid line of Figure 1, thhis leads to an unfortunately optimistic interpolation between mass and the other parameters, but this can be rectified by setting the values of o and c.

2.6 Approaching linear interpolation

Note that the model remains GP compatible if and only if $o \geq 0$, $c \geq 0$.

As can be seen in Figure 2, high values of c favour larger parameter values. Since in the GP formulation we can only mix parameters we want more of (whether it's P or $\frac{1}{m}$), we will generally want to set c as low as possible.

The effect of a positive o, on the other hand, is to squish the log difference between the smallest and largest options for a property, and so is only useful for inverted terms like m. As seen in Figure 4, larger values of o_m approach a linear fit.

As an extremely large o_m and small c both lead to numerical issues in the solver, we reccomend using values near $c = 0.1, o_m = 1$.

2.7 Using exponent choice as a fit

In some cases, the optimism of a fit can be decreased by changing a particular c, as in Figure **??**. This causes interpolations of the corresponding property with unwanted properties such as m to become more optimistic, but as long as those comparisons were already pessimistic this can increase the overall pessimism of the fit considerably.

These fits can be enhanced by using negative values of c and o in an SP selection framework, but such models can be numerically tricky and are generally more quickly solved by the integer programming heuristic described below.

2.8 Summary

Our final constraint set, then is (with the chose of ϵ for the power we fit of each variable, and do this transformation in the definition of $|P|$ etcetera for numerical reasons)

$$\gamma_i = e^{S_i}$$

$$1 \leq \gamma_i \leq e \ \forall i$$

$$\Pi\gamma_i = e$$

$$|P_{opt}| = \left(\frac{P_{opt}}{\max(P_{opt})} + o_P\right)^{c_P}$$

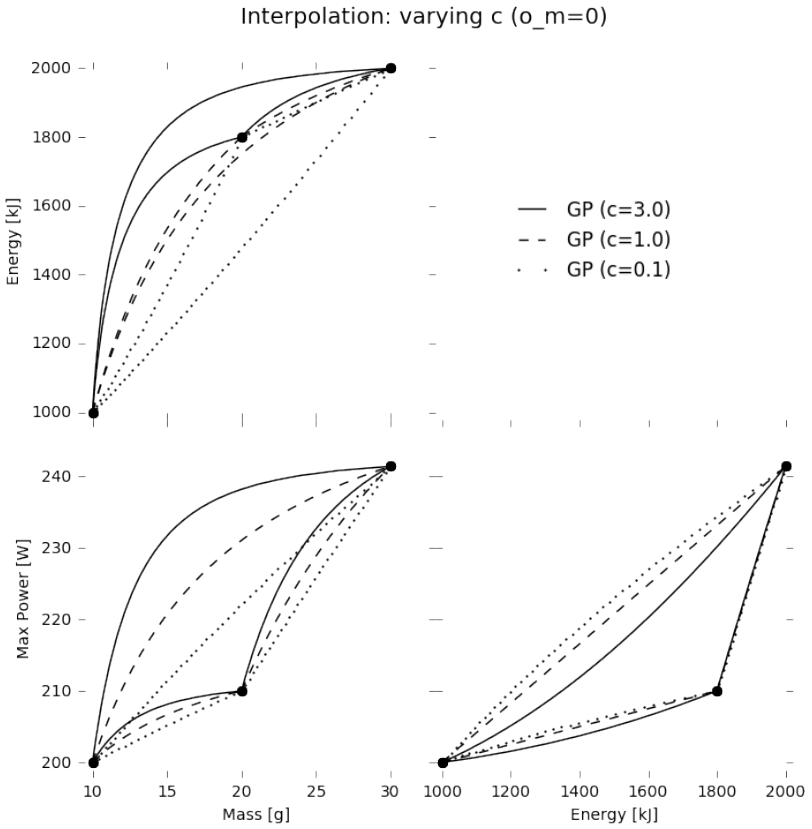

Figure 2: At larger values of c interpolations between the smallest and largest battery are more optimistic in power and energy per mass, and pessimistic in power per energy.

$$|P|^{\frac{1}{c_P}} \geq \frac{P}{\max(P_{opt})} + o_P$$

$$\Pi(e^{S_i})^{|P_{opt}|_i} \geq 1 + \sum_{k=1}^{M} \frac{|P|^k}{k!}$$

$$|E_{opt}| = \left(\frac{E_{opt}}{\max(E_{opt})} + o_E \right)^{c_E}$$

$$|E|^{\frac{1}{c_E}} \geq \frac{E}{\max(E_{opt})} + o_E$$

$$\Pi(e^{S_i})^{|E_{opt}|_i} \geq 1 + \sum_{k=1}^{M} \frac{|E|^k}{k!}$$

$$|m_{opt}| = \left(\frac{m_{opt}}{\max(m_{opt})} + o_m \right)^{c_m}$$

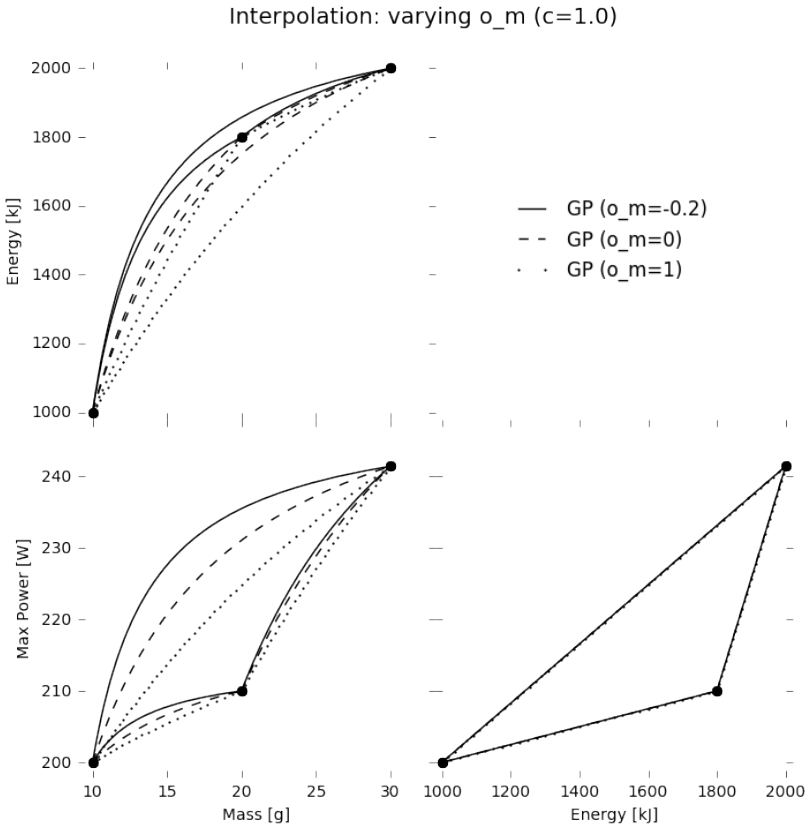

Figure 3: Larger values of o_m approach a linear fit.

$$|m|^{\frac{1}{c_m}} \geq \frac{m}{\max(m_{opt})} + o_m$$

$$\Pi(e^{S_i})^{|m_{opt}|_i^{-1}} \geq 1 + \sum_{k=1}^{M} \frac{1}{k!\,|m|^k}$$

3 Binary selection

For now the best way to solve this to an final part selection seems to be to, starting with the most certain selection among those in the problem, fixing that selection and then re-solving, continuing until all selections have been fiexed and then going to the top level and starting with the second-best selection, etc. Searches can be stopped if a fully fixed solution has been found that is better than them.

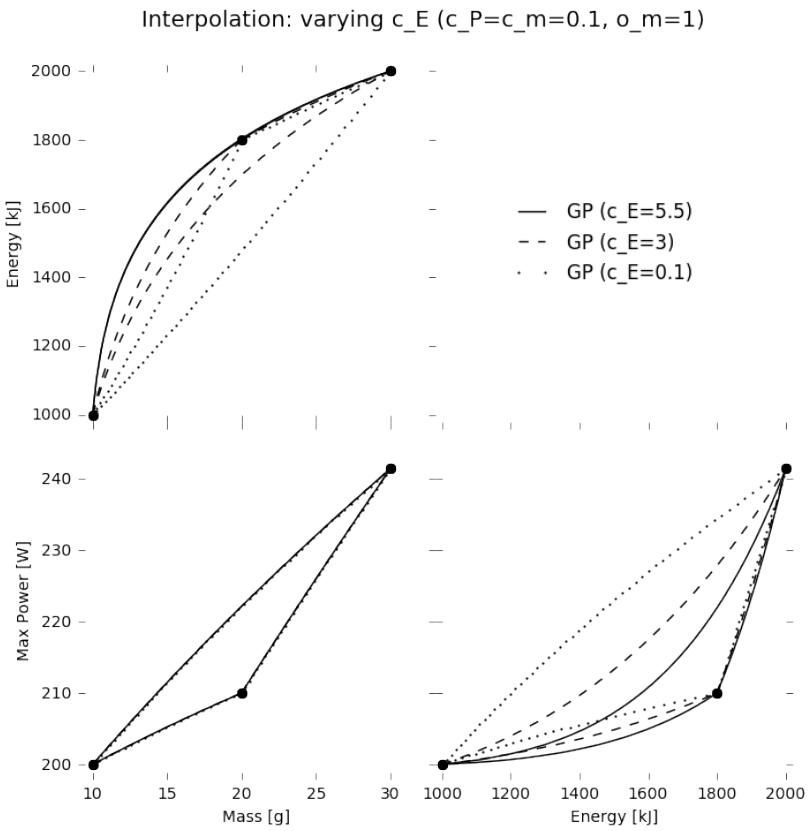

Figure 4: Larger values of c_E make energy per mass more optimistic, but power per energy more pessimistic. This could be considered an improvement to the overall fit.