One Constraint to Rule Them All: How to simplify optimizations under constant variable sum, with applications for maximum likelihood

Doug McErlean
Advisor: Sorin Istrail

05/01/2013

Abstract

We begin by considering the general problem of optimizing a function of non-negative real variables subject to a set of constant-sum constraints on a disjoint partitioning of the variable set. Such problems are widely applicable, and in particular subsume global maximum likelihood estimation problems over one or more sets of multinomial parameters. We will give an intuitive interpretation of how the Lagrangian constrains the partial derivatives at stationary points, and derive a function we call the Ψ-mapping, the fixed points of which are exactly the stationary points under a set of reasonable assumptions. Next we will demonstrate a number of interesting properties of this mapping, including the fact that its codomain composes convexly under addition and multiplication of two objective functions. Finally, we will show how this mapping simplifies in common polynomial-like and homogeneous cases, and what these simplified forms tell us about the space of candidate stationary points. Specific applications will also be explored for the haplotype phasing problem, in which the resulting Ψ coincides will the known iterative equation for the Expectation Maximization algorithm.

1 Introduction

In order to effectively tackle a hard problem in mathematics or computer science, no matter what the context or problem statement, we need to bring to bear both proven facts and creative intuition. The more you can prove about a problem, the more of a glimpse you get at the loose trappings of its structure, until with a great intuitional leap the observations crystallize into a deep and powerful insight. With this insight might come more proofs, or simply more elegant ones, leading to improved insight, leading to more proven facts, and so on. It is this ability to iterate between observations, formal proofs, and fluid understanding that allows for true solutions to be obtained. Yet for a large, useful, and incredibly common class of optimization problems, we seem to have largely thrown in the towel.

Maximum likelihood problems have been commonplace ever since their introduction by Fisher in the early 20th century (a rigorous introduction to the general theory of likelihoods can be found in [1]). In the most general case, maximum likelihood optimizations attempt to find whatever parameters give a chosen model the highest probability of generating some observed data set. Examples are widespread throughout the literature; we will focus in particular on the haplotype phasing problem later on in this thesis. Yet despite their ubiquity, we know very little about truly solving these problems. There are general frameworks for finding local optima of these functions, including the Expectation Maximization framework as famously unified by Dempster, Laird, and Rubin in [2]. But even though EM and processes like it can generally converge to a local solution, they give very little insight or intuition into how we should be thinking about a particular problem and what sorts of solutions might be feasible. In short, they are relatively nearsighted solvers; once formulated for a particular problem domain, EM simply works as a black box, giving solutions but without revealing any deeper answers.

The goal of this work is to take a longer view of maximum likelihood in general. Even though we can't hope to fully solve the broadest possible case, we're more interested in what headway we *can* make and what details are most important to make progress. In short, we're looking for some relatively low-hanging fruit, which has been neglected due to the use of black-box solvers, in order to at the very least get our theoretical foot in the door on these sorts of problems. If we can prove a few small lemmas about the general problem, then perhaps we can gain some intuition that we can capitalize on to prove more lemmas, and so on. Only having proven all we can in this generalized setting will we be properly equipped to focus back in on a particular problem, probing its structure for features which our general theory can exploit.

It is for this very reason that this thesis has two main components: first, a continuous treatment of a general problem; then, a discrete treatment of a very specific problem. Performing our theoretical inquiry in this fashion has led us to an avalanche of new results about our chosen haplotype phasing problem, as well as some deep intuitional leaps regarding what EM is doing in a fairly general setting. Our hope is to reveal both proven results and helpful intuition, in order to begin to catalyze future work both on haplotype phasing and other maximum likelihood problems as well.

2 Continuous Analysis

The first thing we wish to do in our analysis is solve the most general class of optimization problems we can, to the greatest extent we can. We can always narrow our focus later on, in order to simplify unruly formulas or exploit some particular fact about a class of optimizations. However, starting out with a broad view of the possible problems will give us a better idea of where to go next. We can't expect a heavily generalize problem to be fully solveable, but we might be able to discover through enough creative exploration what level of specificity is actually required to make which claims. Thus this section, which focuses on the analysis of optimizations in the continuous setting, is about finding out what we need to know in order to solve a problem, as opposed to figuring out any particular solution or algorithm.

2.1 General Problem

For some mapping $\Pi : \{1, \ldots, V\} \to \{1, \ldots, C\}$ and constants $c_1, \ldots, c_C > 0$, we wish to maximize a function $f : \mathbb{R}^V_{\geq 0} \to \mathbb{R}$ subject to the following set of disjoint and complete constant-sum constraints:

$$\forall i \in \{1, \ldots, C\} : \; c_i = \sum_{\Pi(u)=i} x_u \tag{1}$$

In other words, Π maps each variable to a single variable set, and each such set is constrained to have a particular constant sum. We will quickly discover classes of f for which this problem is greatly simplified, but we will first explore the most general case.

This class of problems was selected as a starting point because it is generalizes many common applications. The haplotype phasing problem, which we'll see later on, falls into this category, but so does the hidden markov model learning problem, and a great deal of other optimizations which in general look somewhat like maximum likelihood problems. More specifically, any attempt to find a maximum likelihood assignment for one or more multinomial distributions will fall into this category. Again, we'll narrow our analysis more and more as we proceed, but only as we discover which details of the problem are actually helpful or necessary to simplify our task. Until we've found what looks like a good path to take, we're simply trying to keep our eyes open.

2.2 Stationary Points

We first wish to gain an intuitive understanding of what the critical points of this optimization look like. Of course, any point where the objective function is not differentiable will be a critical point, but for most practical applications we will be able to neglect this case. What we're most interested in is what the partial derivatives of f will look like at stationary points, where the gradient of the Lagrangian is the zero vector.

For this particular optimization, the Lagrangian is quite simple, and so the desired result is simple as well:

$$\mathcal{L}(\vec{x}, \vec{\lambda}) = f(\vec{x}) - \sum_i \lambda_i ((\sum_{\Pi(u)=i} x_u) - c_i) \tag{2}$$

$$\begin{aligned} 0 &= \frac{\partial \mathcal{L}}{\partial x_u} \\ &= \frac{\partial f}{\partial x_u} - \lambda_{\Pi(u)} \\ \lambda_{\Pi(u)} &= \frac{\partial f}{\partial x_u} \end{aligned} \tag{3}$$

So at the stationary points of our optimization the partial derivatives with respect to the variables in any particular partition must be the same, but not necessarily equal to zero! However, our domain is closed, so we must also check the boundaries for candidate optima. In order to apply the intuition we

have just gained, we can separate the problem into interior and boundary cases.

In the interior of the domain, we wish to maximize $f(x_1, \ldots, x_V)$ such that $x_u > 0$ for all u, and so the partial derivatives must be equal within each group as stated. But along the boundary where $x_u = 0$, we wish to maximize $g(x_1, \ldots, x_{u-1}, x_{u+1}, x_V) = f(x_1, \ldots, x_{u-1}, 0, x_{u+1}, x_V)$ such that $x_v > 0$ for all $v \neq u$. In this boundary case, the partial derivative with respect to x_u is excluded from the equality condition, because it does not appear in the resulting modified Lagrangian. For further boundaries and corners, this logic holds inductively. At the risk of slightly abusing notation, we will therefore use the following definition of "stationary point" from now on:

Definition: Stationary point A stationary point of the optimization is any point where there exist constants $\lambda_1, \ldots, \lambda_C$ such that, for all $u \in \{1, \ldots, V\}$, either $x_u = 0$ or $\frac{\partial f}{\partial x_u} = \lambda_{\Pi(u)}$.

In other words, within each variable group defined by Π, the partial derivatives of the non-zero variables must all be equal.

2.3 Radial Functions

We see that, so long as $\lambda_{\Pi(u)} \neq 0$, the following holds true at a stationary point, by our new definition:

$$\begin{aligned}
\frac{x_u}{c_{\Pi(u)}} &= \frac{x_u}{\sum_{\Pi(v)=\Pi(u)} x_v} \\
&= \frac{\lambda_{\Pi(u)} x_u}{\sum_{\Pi(v)=\Pi(u)} \lambda_{\Pi(v)} x_v} \\
&= \frac{\lambda_{\Pi(u)} x_u}{\sum_{\Pi(v)=\Pi(u)} \lambda_{\Pi(v)} x_v} \\
&= \frac{x_u \frac{\partial f}{\partial x_u}}{\sum_{\Pi(v)=\Pi(u)} x_v \frac{\partial f}{\partial x_v}}
\end{aligned} \tag{4}$$

Furthermore, if this condition holds for all $u \in \{1, \ldots, V\}$, then we must be at a stationary point, because we have that

$$\begin{aligned}
\forall u, v \; s.t. \; \Pi(u) = \Pi(v) \wedge x_u > 0 \wedge x_v > 0 : \; \frac{x_u}{x_v} &= \frac{x_u \frac{\partial f}{\partial x_u}}{x_v} \\
1 &= \frac{\frac{\partial f}{\partial x_u}}{\frac{\partial f}{\partial x_v}} \\
\frac{\partial f}{\partial x_v} &= \frac{\partial f}{\partial x_u}
\end{aligned}$$

which is exactly our definition of a stationary point. However, this insight is really only useful if we can guarantee that (4) gives reasonable outputs, meaning that it should be defined and non-negative. This leads us to the following definition and lemma:

Definition: i-radial A function f with the constraint set $\Pi, c_1, \ldots, c_C$ is i-radial over some neighborhood S if for all points $\vec{x}$ in S, the ratio $\frac{x_u \frac{\partial f}{\partial x_u}}{\sum_{\Pi(u)=i} x_u \frac{\partial f}{\partial x_u}}$ is defined and non-negative throughout S, for all u such that $\Pi(u) = i$.

The idea behind this definition is that in an i-radial optimization, the objective function f is never stationary with respect to a radial change in the ith variable set. Such motion (increasing all variables in the set in proportion to their values) would of course violate the constant-sum constraint, but the point is that the gradient of f should not lie strictly in the constant-sum hyperplane. Additionally, this motion should be consistent across different variables, meaning that if increasing one variable raises the objective, increasing another should not lower it.

Lemma 2.1. *If a function f and constraint set $\Pi, c_1, \ldots, c_C$ are i-radial for all $i \in 1, \ldots, C$ (**fully radial**) over some neighborhood S, then within S the set of points satisfying equation (4) for all $u \in \{1, \ldots, V\}$ are exactly the stationary points of the optimization.*

Proof. So long as the denominator of equation (4) is non-zero for every variable group, we have already proven both directions of the implication in the discussion above. But our requirement for an i-radial optimization is exactly that this denominator will be non-zero for all u such that $\Pi(u) = i$. □

This definition also leads to some intuitional insight about the meaning of equation (4): the ratio of a variable to the sum of the variables in its group is the fraction of the radial change of f contributed by changing that variable. This is most easily-seen when we make the observation that $x_u \frac{\partial f}{\partial x_u} = \frac{\partial f}{\partial \ln x_u}$.

Of course, forcing our optimization to be radial in any of the variable groups is quite a strong statement. In fact, it typically constrains us to the following two cases:

Definition: i-inward and i-outward A function f with the constraint set $\Pi, c_1, \ldots, c_C$ is i-inward (resp. i-outward) over some neighborhood S if it is i-radial over S and $\frac{\partial f}{\partial x_u} \leq 0$ (resp. ≥ 0) throughout S, for all u such that $\Pi(u) = i$.

Lemma 2.2. *If f is continuously-differentiable over some neighborhood S, then the optimization problem $(f, \Pi, \vec{c})$ is i-radial over S if and only if it is i-inward or i-outward over S.*

Proof. This follows simply from the intermediate value theorem. The variable values x_u are clearly continuous over any neighborhood, so $\sum_{\Pi(u)=i} x_u \frac{\partial f}{\partial x_u}$ will be continuous over S. If this sum were positive at one point in S and negative in another, then its value would have to be zero at some point along any path connecting the two points, including paths exclusively traveling through S. So it must have consistent sign throughout S in order for the optimization to be i-radial. But in addition, $\frac{\partial f}{\partial x_u}$ for $\Pi(u) = i$ must either be zero or have the same sign as $\sum_{\Pi(u)=i} x_u \frac{\partial f}{\partial x_u}$, at any given point in S. This implies that if the optimization is i-radial, it must also be i-inward or i-outward. The reverse implication is definitional, as i-inward and i-outward optimizations are explicitly i-radial. □

Despite these restrictions, there are many interesting and useful cases in which f is fully-radial throughout the entire constrained domain. From now on we will primarily be concerning ourselves on with such cases, in order to explore and exploit the properties of equation (4). For convenience and compactness, we will express this system of equations in terms of a set of vector-valued mappings:

$$\hat{\Psi}^i_{f,\Pi,\vec{c}}(\vec{x}) = \frac{c_i}{\sum_{\Pi(v)=i} x_v \frac{\partial f}{\partial x_v}} \sum_{\Pi(u)=i} x_u \frac{\partial f}{\partial x_u} \vec{e}_u \tag{5}$$

$$\Psi^i_{f,\Pi,\vec{c}}(\vec{x}) = \begin{cases} \hat{\Psi}^i_{f,\Pi,\vec{c}}(\vec{x}) & : \sum_{\Pi(u)=i} x_v \frac{\partial f_p}{\partial x_u} \neq 0 \\ \vec{0} & : \sum_{\Pi(u)=i} x_v \frac{\partial f_p}{\partial x_u} = 0 \end{cases} \tag{6}$$

$$\Psi_{f,\Pi,\vec{c}}(\vec{x}) = \sum_{i=1}^{C} \Psi^i_{f,\Pi,\vec{c}}(\vec{x}) \tag{7}$$

Clearly, we have that $\vec{x} = \Psi_{f,\Pi,\vec{c}}(\vec{x}) = \hat{\Psi}_{f,\Pi,\vec{c}}(\vec{x})$ at all stationary points of any fully-radial optimization f, Π, $\vec{c}$. So the stationary points are fixed points of this mapping. The reason for the piecewise convention will become clear in the next section, in which we will be creating an i-radial objective by combining multiple component functions, which may not themselves be i-radial.

2.4 Additive Composition of Ψ

Let $f(\vec{x}) = \sum_p f_p(\vec{x})$. Then we observe the following:

$$R_i(\vec{x}) = \{p : 0 \neq \sum_{\Pi(u)=i} x_u \frac{\partial f_p}{\partial x_u}\}$$

$$\begin{aligned}
[\hat{\Psi}^i_{f,\Pi,\vec{c}}]_u &= c_i \mathbb{1}(\Pi(u) = i) \frac{x_u \frac{\partial f}{\partial x_u}}{\sum_{\Pi(v)=\Pi(u)} x_v \frac{\partial f}{\partial x_v}} \\
&= c_i \mathbb{1}(\Pi(u) = i) \frac{\sum_p x_u \frac{\partial f_p}{\partial x_u}}{\sum_{\Pi(v)=\Pi(u)} \sum_j x_v \frac{\partial f_j}{\partial x_v}} \\
&= c_i \mathbb{1}(\Pi(u) = i) \sum_{p \in R_i(\vec{x})} \frac{\frac{x_u \frac{\partial f_p}{\partial x_u}}{\sum_{\Pi(s)=\Pi(u)} x_s \frac{\partial f_p}{\partial x_s}} \sum_{\Pi(t)=\Pi(u)} x_t \frac{\partial f_p}{\partial x_t}}{\sum_j \sum_{\Pi(v)=\Pi(u)} x_v \frac{\partial f_j}{\partial x_v}} \\
&= \sum_p \frac{\sum_{\Pi(t)=\Pi(u)} x_t \frac{\partial f_p}{\partial x_t}}{\sum_j \sum_{\Pi(v)=\Pi(u)} x_v \frac{\partial f_j}{\partial x_v}} [\Psi_{f_p,\Pi,\vec{c}}]_u
\end{aligned}$$

This value is defined so long as the total function f is i-radial, and we see that it is a weighted sum of the Ψ values for the component f_p, with weights conveniently summing to one. We even have that zero weights occur for precisely those $\hat{\Psi}$ values which we have defined to be zero, meaning it gives a reasonable potential value for $\vec{x}$. However, it is possible for some of these weights to be negative, which is potentially somewhat undesirable. Thus we make the following new definition:

Definition: i-consistent A function $f = \sum_p f_p$ with the constraint set $\Pi, c_1, \ldots, c_C$ is i-consistent over some neighborhood S if it is i-radial over S, and in addition for all points $\vec{x}$ in S and for all p, $\frac{\sum_{\Pi(u)=i} x_u \frac{\partial f_p}{\partial x_u}}{\sum_{\Pi(u)=i} x_u \frac{\partial f}{\partial x_u}}$ is non-negative.

Lemma 2.3. *If a function $f = \sum_p f_p$ with the constraint set $\Pi, c_1, \ldots, c_C$ is i-consistent over some neighborhood S, then for every point $\vec{x}$ in S, $\hat{\Psi}^i_{f,\Pi,\vec{c}}(\vec{x})$ is a convex combination of the non-zero component $\Psi^i_{f_p,\Pi,\vec{c}}(\vec{x})$ values.*

Proof. We make the following substitutions using the result from the discussion above:

$$\omega^i_{f,\Pi,\vec{c}} = \frac{1}{\sum_j \sum_{\Pi(v)=i} x_v \frac{\partial f_j}{\partial x_v}} \sum_{\Pi(u)=i} x_u \left[\frac{\partial f_1}{\partial x_u}, \frac{\partial f_2}{\partial x_u}, \ldots, \frac{\partial f_p}{\partial x_u}, \ldots\right]^T \tag{8}$$

$$\Psi^i_{f,\Pi,\vec{c}} = \sum_p [\omega^i_{f,\Pi,\vec{c}}]_p \Psi^i_{f_p,\Pi,\vec{c}} \tag{9}$$

By the definition of i-consistency, we know that $\frac{\sum_{\Pi(u)=i} x_u \frac{\partial f_p}{\partial x_u}}{\sum_{\Pi(u)=i} x_u \frac{\partial f}{\partial x_u}} \geq 0$ throughout S, for all p. But by construction each such value is exactly the pth component of the weight vector $\omega^i_{f,\Pi,\vec{c}}$, so we know that this vector has only non-negative components. Furthermore, we see that zero components occur exactly when the corresponding f_p is not i-radial and thus excluded from the combination, because the numerator of this fraction is precisely what must be non-zero for an i-radial f_p. Finally, the sum of these weights is $\frac{\sum_p \sum_{\Pi(u)=i} x_u \frac{\partial f_p}{\partial x_u}}{\sum_{\Pi(u)=i} x_u \frac{\partial f}{\partial x_u}}$, which is one by inspection. □

This is a fascinating property, but we have introduced yet another even more complicated prerequisite for its use! In order for this compositional rule to be of any practical value we'll need to find an interesting class of problems which are i-consistent. Fortunately, there is a loose sufficient condition we can place on the components f_p in order to guarantee that their sum will be i-consistent:

Definition: Strongly i-consistent A function $f = \sum_p f_p$ with the constraint set $\Pi, c_1, \ldots, c_C$ is strongly i-consistent over some neighborhood S if it is both i-consistent and either i-outward or i-inward over S.

Lemma 2.4. *A function $f = \sum_p f_p$ with the constraint set $\Pi, c_1, \ldots, c_C$ which is continuously-differentiable over some neighborhood S is i-consistent over S if and only if it is strongly i-consistent over S.*

Proof. The proof follows directly from Lemmas 2.2. If f is i-consistent over S, then it must be i-radial over S, and for a continuously-differentiable f to be i-radial over S it must be either i-inward or i-outward over S. Either case, combined with the i-consistency of f, yields strong i-consistency. The opposite direction is definitional, as strong i-consistency explicitly requires i-consistency. □

2.5 Multiplicative Composition of Ψ

For neighborhoods in which our objective function is strictly positive and therefore has a well-defined real logarithm, we observe the following interesting invariance:

$$\begin{aligned}
[\hat{\Psi}^i_{\ln f,\Pi,\vec{c}}]_u &= c_i \mathbb{1}(\Pi(u) = i) \frac{x_u \frac{\partial \ln f}{\partial x_u}}{\sum_{\Pi(v)=\Pi(u)} x_v \frac{\partial \ln f}{\partial x_v}} \\
&= c_i \mathbb{1}(\Pi(u) = i) \frac{x_u \frac{\frac{\partial f}{\partial x_u}}{f(\vec{x})}}{\frac{1}{f(\vec{x})} \sum_{\Pi(v)=\Pi(u)} x_v \frac{\partial f}{\partial x_v}} \\
&= c_i \mathbb{1}(\Pi(u) = i) \frac{x_u \frac{\partial f}{\partial x_u}}{\sum_{\Pi(v)=\Pi(u)} x_v \frac{\partial f}{\partial x_v}} \\
&= [\hat{\Psi}^i_{f,\Pi,\vec{c}}]_u
\end{aligned} \tag{10}$$

Combining this result with the results of the previous section gives us the following, for $f = \prod_p f_p$:

$$\begin{aligned}
LR_i(\vec{x}) &= \{p : 0 \neq \sum_{\Pi(u)=i} x_u \frac{\partial \ln f_p}{\partial x_u}\} \\
[\hat{\Psi}^i_{f,\Pi,\vec{c}}]_u &= [\hat{\Psi}^i_{\ln f,\Pi,\vec{c}}]_u \\
&= c_i \mathbb{1}(\Pi(u) = i) \frac{x_u \frac{\partial \ln f}{\partial x_u}}{\sum_{\Pi(v)=\Pi(u)} x_v \frac{\partial \ln f}{\partial x_v}} \\
&= c_i \mathbb{1}(\Pi(u) = i) \frac{\sum_p x_u \frac{\partial \ln f_p}{\partial x_u}}{\sum_{\Pi(v)=\Pi(u)} \sum_j x_v \frac{\partial \ln f_j}{\partial x_v}} \\
&= c_i \mathbb{1}(\Pi(u) = i) \sum_{p \in LR_i(\vec{x})} \frac{\frac{x_u \frac{\partial \ln f_p}{\partial x_u}}{\sum_{\Pi(s)=\Pi(u)} x_s \frac{\partial \ln f_p}{\partial x_s}} \sum_{\Pi(t)=\Pi(u)} x_t \frac{\partial \ln f_p}{\partial x_t}}{\sum_j \sum_{\Pi(v)=\Pi(u)} x_v \frac{\partial \ln f_j}{\partial x_v}} \\
&= \sum_p \frac{\sum_{\Pi(t)=\Pi(u)} x_t \frac{\partial \ln f_p}{\partial x_t}}{\sum_j \sum_{\Pi(v)=\Pi(u)} x_v \frac{\partial \ln f_j}{\partial x_v}} [\Psi_{\ln f_p,\Pi,\vec{c}}]_u \\
&= \sum_p \frac{\sum_{\Pi(t)=\Pi(u)} x_t \frac{\partial f_p}{\partial x_t}}{f_p(\vec{x}) \sum_j \frac{1}{f_j(\vec{x})} \sum_{\Pi(v)=\Pi(u)} x_v \frac{\partial f_j}{\partial x_v}} [\Psi_{f_p,\Pi,\vec{c}}]_u
\end{aligned}$$

Lemma 2.5. *If a function $f = \prod_p f_p$ is non-zero over some neighborhood S and the optimization of $\ln f = \sum_p \ln f_p$ with the constraint set $\Pi, c_1, \ldots, c_C$ is i-consistent over S, then $\hat{\Psi}^i_{f,\Pi,\vec{c}}(\vec{x})$ is a convex combination of the non-zero component $\Psi^i_{f_p,\Pi,\vec{c}}(\vec{x})$ values.*

Proof. This follows directly from Equation (10) and Lemma 2.3. □

The equations defining this convex combination are similar, but not identical, to those from the previous section. Following from the discussion above, we have the following new weight vector:

$$\omega^i_{f,\Pi,\vec{c}} = \frac{1}{\sum_j \frac{1}{f_j(\vec{x})} \sum_{\Pi(v)=i} x_v \frac{\partial f_j}{\partial x_v}} \sum_{\Pi(u)=i} x_u \left[\frac{\frac{\partial f_1}{\partial x_u}}{f_1(\vec{x})}, \frac{\frac{\partial f_2}{\partial x_u}}{f_2(\vec{x})}, \ldots, \frac{\frac{\partial f_p}{\partial x_u}}{f_p(\vec{x})}, \ldots \right]^T \tag{11}$$

2.6 Positive Homogeneous Functions and Elasticity

Given that the Ψ-mapping is somewhat complex in its full generality, we'd like to find a useful class of problems for which it simplifies into a more tractable form. As it happens, one simplification is possible for a very broad class of problems, including the polynomials and many polynomial-like objective functions:

Definition: Positive Homogeneous Over some neighborhood S, we say that a function f is positive (k, H)-homogeneous with respect to some real number k and some set of variable indices H if the following holds for all $\vec{x} \in S$ and for all $\alpha > 0$:

$$f(\alpha^{\mathbb{1}(1\in H)} x_1, \alpha^{\mathbb{1}(2\in H)} x_2, \ldots, \alpha^{\mathbb{1}(V\in H)} x_V) = \alpha^k f(x_1, x_2, \ldots, x_V) \tag{12}$$

In other words, multiplying every variable in the set H by some positive factor α multiplies f by α^k.

In a neighborhood S where our objective f satisfies this definition, we can exploit Euler's homogeneous function theorem, which states that the positive (k, H)-homogeneous functions over S are equivalently those functions for which the following equality holds throughout S:

$$\sum_{u\in H} x_u \frac{\partial f}{\partial x_u} = k f(\vec{x}) \tag{13}$$

This equality yields the following important fact about what is required for a positive homogeneous function to also be partially or fully radial:

Lemma 2.6. *If f is positive $(k_i, \{x_u : \Pi(u) = i\})$-homogeneous and the optimization of f, Π, $\vec{c}$ is i-radial, both over some neighborhood S, then $k_i \neq 0$ and $f(\vec{x})$ is either everywhere positive or everywhere negative throughout S.*

Proof. The proof is similar to that of Lemma 2.2. By definition, the optimization will be i-radial over S if and only if $\sum_{\Pi(u)=i} x_u \frac{\partial f}{\partial x_u} = k_i f(\vec{x})$ is non-zero throughout S. We note that in order to satisfy the definition of positive homogeneity, f must be continuous over this neighborhood, and so the intermediate value theorem tells us that if f changes sign in this neighborhood then $f(\vec{x})$ and therefore $k_i f(\vec{x})$ will be zero somewhere in S. □

For variable groups and neighborhoods where these conditions hold, we can greatly simplify equations (4) and (5). In particular, for any neighborhood over which f is positive $(k_i, \{x_u : \Pi(u) = i\})$-homogeneous for $k_i \neq 0$, we have the following, for all u such that $\Pi(u) = i$:

$$\frac{x_u}{c_i} = \frac{x_u \frac{\partial f}{\partial x_u}}{k_i f(\vec{x})} \tag{14}$$

$$\hat{\Psi}^i_{f,\Pi,\vec{x}}(\vec{x}) = \frac{c_i}{k_i f(\vec{x})} \sum_{\Pi(u)=i} x_u \frac{\partial f}{\partial x_u} \vec{e}_u \tag{15}$$

We can also introduce a yet more intuitive notion of what these simplified equations mean by noting that $\frac{x_u \frac{\partial f}{\partial x_u}}{f} = \frac{\partial \ln f}{\partial \ln x_u}$. This value, which is seen far more commonly in economics than in pure mathematical contexts, is known as the **elasticity** of f with respect to x_u. The astute reader will note that if f is a

monomial function, this value coincides with the multiplicity of x_u in f. Indeed, elasticity is constant if and only if $f = x_u^m g(\vec{x})$, for some constant m and some function g which is constant with respect to x_u.

In more complex contexts, however, we can note that elasticity is in some sense still an approximation to multiplicity. So long as the elasticity in every variable is defined at a point, we can always form a monomial function with the same elasticity in each variable as our original function. For each homogeneous variable group, the corresponding components of Ψ are in proportion to the elasticities of their respective variables. Thus Ψ maps each homogeneous variable group to the exact point which maximizes this **locally co-elastic monomial**, since as a monomial function its elasticities are constant throughout the domain. For $\vec{x}$ to be a stationary point, then, means that it is a point which maximizes its associated locally co-elastic monomial.

2.7 Ψ Composition for Positive Homogeneous f

As with the equations for Ψ itself, positive homogeneity greatly simplifies the rules for composing multiple Ψ mappings.

For $f = \sum_p f_p$, over neighborhoods where every f_p is positive $(k_i, \{x_u : \Pi(u) = i\})$-homogeneous, we can substitute into Equation (8) to get the following, where applicable:

$$\begin{aligned}
\omega^i_{f,\Pi,\vec{c}} &= \frac{1}{\sum_j \sum_{\Pi(v)=i} x_v \frac{\partial f_j}{\partial x_v}} \sum_{\Pi(u)=i} x_u \left[\frac{\partial f_1}{\partial x_u}, \frac{\partial f_2}{\partial x_u}, \ldots, \frac{\partial f_p}{\partial x_u}, \ldots\right]^T \\
&= \frac{1}{\sum_j k_i f_j(\vec{x})} \left[k_i f_1(\vec{x}), k_i f_2(\vec{x}), \ldots, k_i f_p(\vec{x}), \ldots\right]^T \\
&= \frac{1}{k_i f(\vec{x})} \left[k_i f_1(\vec{x}), k_i f_2(\vec{x}), \ldots, k_i f_p(\vec{x}), \ldots\right]^T \\
&= \frac{1}{f(\vec{x})} \left[f_1(\vec{x}), f_2(\vec{x}), \ldots, f_p(\vec{x}), \ldots\right]^T
\end{aligned} \tag{16}$$

Clearly these weights are defined over any neighborhood where f is non-zero, so long as k_i is also non-zero. However, while the weights will sum to one, some of them may be negative. We therefore give a simple equivalent condition for this to define a convex combination, as opposed to just a linear one:

Lemma 2.7. *If $f = \sum_p f_p$ and every f_p is $(k_i, \{x_u : \Pi(u) = i\})$-homogeneous over some neighborhood S, then the optimization f, Π, $\vec{c}$ will be (strongly) i-consistent if and only if $k_i \neq 0$, f is non-zero over S, and either every f_p is everywhere non-negative, or every f_p is everywhere non-positive, throughout S.*

Proof. Recall the definition of i-consistency, which requires that the ratio $\frac{\sum_{\Pi(u)=i} x_u \frac{\partial f_p}{\partial x_u}}{\sum_{\Pi(u)=i} x_u \frac{\partial f}{\partial x_u}} = \frac{k_i f_p(\vec{x})}{k_i f(\vec{x})}$ must be defined and non-negative throughout S. This clearly requires that k_i and f be non-zero, and additionally requires that any component values $f_p(\vec{x})$ which are non-zero must have the same sign as $f(\vec{x})$, for any given point $\vec{x}$ in S. Since f is positive homogeneous (as the sum of homogeneous functions), it must by definition be continuous, and so it cannot change sign within S, or else the intermediate value theorem gives us that it would be zero at some point within S. So i-consistency implies that $k_i \neq 0$ and that f and therefore all f_p have the same sign as each other at all points in S, and that they never change sign within S. In other words, the optimization will be strongly i-consistent. The converse follows because if the given conditions hold, f will clearly take on the same unchanging sign as its components f_p, at all points within S, and will be non-zero by assumption, satisfying the definitions of both i-consistency and strong i-consistency. □

For $f = \prod_p f_p$, over neighborhoods where the f_p are positive $(k_{i,p}, \{x_u : \Pi(u) = i\})$-homogeneous,

we can substitute into Equation (11) to get the following, where applicable:

$$\begin{aligned}\omega^i_{f,\Pi,\vec{c}} &= \frac{1}{\sum_j \frac{1}{f_j(\vec{x})} \sum_{\Pi(v)=i} x_v \frac{\partial f_j}{\partial x_v}} \sum_{\Pi(u)=i} x_u \left[\frac{\frac{\partial f_1}{\partial x_u}}{f_1(\vec{x})}, \frac{\frac{\partial f_2}{\partial x_u}}{f_2(\vec{x})}, \ldots, \frac{\frac{\partial f_p}{\partial x_u}}{f_p(\vec{x})}, \ldots\right]^T \\ &= \frac{1}{\sum_j \frac{1}{f_j(\vec{x})} k_{i,j} f_j(\vec{x})} \left[\frac{k_{i,1} f_1(\vec{x})}{f_1(\vec{x})}, \frac{k_{i,2} f_2(\vec{x})}{f_2(\vec{x})}, \ldots, \frac{k_{i,p} f_p(\vec{x})}{f_p(\vec{x})}, \ldots\right]^T \\ &= \frac{1}{\sum_j k_{i,j}} \left[k_{i,1}, k_{i,2}, \ldots, k_{i,p}, \ldots\right]^T \end{aligned} \tag{17}$$

Clearly, this derivation is only valid for neighborhoods over which f and therefore all the components f_p are non-zero. In addition, $\sum_p k_{i,p}$ must be non-zero. However, once again it is possible to have a valid negative or zero weight. So again we give a simple equivalent condition for this to be a truly convex combination:

Lemma 2.8. *If $f = \prod_p f_p$ and the f_p are $(k_{i,p}, \{x_u : \Pi(u) = i\})$-homogeneous over some neighborhood S, then the optimization $\ln f$, Π, $\vec{c}$ will be (strongly) i-consistent if and only if every f_p is non-zero throughout S, $0 \neq \sum_p k_{i,p}$, and either every $k_{i,p}$ is non-negative or every $k_{i,p}$ is non-positive.*

Proof. We can see in the derivation above that the radial derivatives in question, $\sum_{\Pi(u)=i} x_u \frac{\partial \ln f_p}{\partial x_u} = \frac{k_{i,p} f_p(\vec{x})}{f_p(\vec{x})}$ are defined if and only if the f_p are non-zero, and take on the same sign as their respective homogeneous orders, $k_{i,p}$. Their sum, meanwhile, is $\sum_p \frac{k_{i,p} f_p(\vec{x})}{f_p(\vec{x})} = \sum_p k_{i,p}$ wherever they are defined. The definition of i-consistency requires that these values all be either zero or of the same sign as their non-zero sum, which is in this case a fixed constant. Thus these conditions imply not just i-consistency, but strong i-consistency, because this sign never changes. The converse is clearly true as well, because strong i-consistency guarantees that these values are defined, not of mixed sign, and that their sum is non-zero. □

It is clear that the positive homogeneous functions will have extremely interesting properties in constant-sum optimizations, as they greatly simplify both Ψ and its compositional rules. These simplifications let us build up manner of interesting objective functions, while maintaining relatively simple equations for their stationary points. For example, we can give an alternative derivation of the constant-sum maximum for a monomial function:

Example: Monomial Maximization Suppose we wish to maximize a monomial $f = z \prod_u x_u^{y_u}$, for $z > 0$ and $y_u > 0$, subject to constraints given by Π,$\vec{c}$. We can of course neglect the constant coefficient z when looking for the maximizing arguments $\vec{x}$. Now consider the component optimizations $f_u = x_u^{y_u}$, and note that each is positive $(y_u, \{x_v : \Pi(v) = \Pi(u)\})$-homogeneous. We can also neglect points where it is zero, because the maximum of f is necessarily positive and therefore cannot occur at such points. Finally, we note that $\hat{\Psi}^i_{f_u,\Pi,\vec{c}} = c_i \mathbb{1}(\Pi(u) = i)\vec{e}_u$, or in other words that f_u is maximized by setting x_u to its maximum allowed value, $c_{\Pi(u)}$. Applying the multiplicative composition rule from Equation (17) above, we end up with

$$\begin{aligned}\hat{\Psi}^i_{f,\Pi,\vec{c}} &= \frac{1}{\sum_{\Pi(u)=i} y_u} \sum_{\Pi(u)=i} y_u \Psi^i_{f_u,\Pi,\vec{c}} \\ &= \frac{1}{\sum_{\Pi(u)=i} y_u} \sum_{\Pi(u)=i} y_u c_i \mathbb{1}(\Pi(u) = i)\vec{e}_u \\ &= \frac{c_i}{\sum_{\Pi(u)=i} y_u} \sum_{\Pi(u)=i} y_u \vec{e}_u \end{aligned} \tag{18}$$

In other words, within each variable group, the variables chosen to be in proportion to their exponents. We note that wherever it is defined, $\hat{\Psi}$ is *always* this same constant vector, and wherever it isn't defined f is zero and therefore of no interest.

Of course, we have neglected the valid case under Lemma 2.8 in which, rather than being all positive, the exponents are all negative. However, this being the inverse of a monomial of the positive variety, its single Ψ-point will be a global minimum rather than a maximum, and its value will approach infinity towards the boundaries. Thus it will be of no interest in the case of global maximization.

2.8 Increasing Posynomial Functions

Having built up Ψ for monomials (and by closure, products of monomials) in Equation (18) above, a natural next step is to try adding them together with positive coefficients (to satisfy i-consistency), forming a so-called "posynomial" function:

Definition: Posynomial A posynomial function is defined as any f of the form $f(\vec{x}) = \sum_i z_i \prod_j x_j^{y_{i,j}}$, for positive real z_i and real $y_{i,j}$.

Of course, Lemma 2.8 precludes us from using monomials with mixed-sign exponents if we want Ψ to give us valid positive values for the variables, and we've already discussed in the previous section why monomials with negative exponents are of little interest. Thus we constrain ourselves to the case of all positive exponents, and henceforth when we use the term posynomial we will actually mean the following:

Definition: Increasing posynomial We define an increasing posynomial to be a posynomial such that each of its component monomials has only positive exponents.

We then have the following lemmas, which will allow us to argue that we can neglect most cases where such a function is not i-consistent:

Lemma 2.9. *An increasing posynomial function f, with constraint set given by Π, $\vec{c}$, is (strongly) i-consistent over exactly the union of the neighborhoods over which at least one component optimization m_p, Π, $\vec{c}$ is i-radial.*

Proof. Since all coefficients, exponents, and variables are non-negative, we know that $\frac{\partial m_p}{\partial x_u} \geq 0$ and $m_p(\vec{x}) \geq 0$, for all non-negative $\vec{x}$ and for all p. Within this region specifically, there is at least one component m_p where $\sum_{\Pi(u)=i} x_u \frac{\partial m_p}{\partial x_u}$ is strictly positive, and since this value too can never be negative, the overall radial motion $\sum_{\Pi(u)=i} x_u \frac{\partial f}{\partial x_u}$ of f must also be strictly positive. This is sufficient not only for i-consistency but strong i-consistency, since the radial motion for the component monomials is either zero or of the same sign as that of the overall function. Outside of this region, $\sum_{\Pi(u)=i} x_u \frac{\partial m_p}{\partial x_u}$ is zero for all m_p, meaning that f is radially stationary and therefore not i-consistent. □

Lemma 2.10. *An increasing posynomial function f is non-zero over exactly the union of neighborhoods over which at least one component optimization m_p, with constraint set given by Π, $\vec{c}$, is i-radial for some i.*

Proof. This follows from the fact that the ith radial derivative $\sum_{\Pi(u)=i} x_u \frac{\partial m_p}{\partial x_u} = m_p(\vec{x}) \sum_{\Pi(u)=i} y_{p,u}$ is zero if and only if $m_p(\vec{x})$ is zero or m_p contains no variables from the ith variable group, and it is strictly positive otherwise. So as long as some m_p and some i have $0 < \sum_{\Pi(u)=i} x_u \frac{\partial m_p}{\partial x_u}$, there is at least one non-zero m_p, and since the monomials are non-negative, f is also non-zero. In the converse case, if $\sum_{\Pi(u)=i} x_u \frac{\partial m_p}{\partial x_u}$ is zero for all i and p, then every m_p is either zero or doesn't contain variables from any of the variable groups. But this latter case is nonsensical, so all m_p must be zero and therefore f must be as well. □

So if an increasing posynomial is not i-consistent in *any* of its variable groups, then it is not i-radial either, and therefore f is zero. Since zero is a lower bound on f, and a strict one for all non-trivial functions, such cases do not interest us for the purposes of maximization.

However, it may be the case that f is sometimes i-radial for some i, but not others. In this case, as mentioned in the proof of Lemma 2.10 above, it must be the case that every monomial m_p which contains a variable from a non-radial i is zero. Such a situation turns out not to be an issue in the

practical applications we explore later. But even if it were, we note that in such cases any properly-constrained values of variables from the non-radial groups will produce the same value for f, and so we can treat those groups as if they don't exist at all. This is a similar claim to the one we made while discussing stationary points: we just treat this as a boundary problem in which some of the variables can be neglected.

Having covered all cases of interest, we can then substitute the definition of a monomial into Equation (8) to get:

$$\omega_f = \frac{1}{\sum_j m_j(\vec{x}) \sum_v y_{j,v}} \left[m_1(\vec{x}) \sum_t y_{1,t}, m_2(\vec{x}) \sum_t y_{2,t}, \ldots, m_\ell(\vec{x}) \sum_t y_{\ell,t}, \ldots \right]^T \tag{19}$$

We note here the astonishing intuition that, since the Ψ-value for each monomial is fixed at the point which maximizes it, stationary points of increasing posynomials must fall within the convex hull of these monomial max points!

Furthermore, if f is also positive $(k_i, \{u : \Pi(u) = i\})$-homogeneous, then this simplifies further, back down to Equation (16):

$$\omega^i_{f,\Pi,\vec{c}} = \frac{1}{f(\vec{x})} [m_1(\vec{x}), m_2(\vec{x}), \ldots, m_\ell(\vec{x}), \ldots]^T \tag{20}$$

In addition, such homogeneity allows us to simplify Equation (18) for m_p to:

$$\hat{\Psi}^i_{m_p,\Pi,\vec{c}} = \frac{c_i}{k_i} \sum_{\Pi(u)=i} y_{p,u} \vec{e}_u \tag{21}$$

Combining these two equations together, we observe the stunningly simple formula for $\hat{\Psi}$ of a homogeneous increasing posynomial:

$$\hat{\Psi}^i_{f,\Pi,\vec{c}} = \frac{c_i}{k_i f(\vec{x})} \sum_p m_p(\vec{x}) \sum_{\Pi(u)=i} y_{p,u} \vec{e}_u \tag{22}$$

2.9 Variable Bounds at Stationary Points

Recall the amazing fact that, so long as we satisfy i-consistency for all our variable groups, we can compose Ψ convexly under addition and multiplication of objective functions. What this means, specifically, is that the codomain of Ψ, in which all stationary points of the optimization must fall, will be some subset of the convex hull of the Ψ-codomains for the component functions. This in turn means that we can perform a weak composition on bounds applying on or between variables in the component functions:

Lemma 2.11. *Let $\vec{A}$ and $\vec{B}$ be vectors of coefficients and let a and b be constants, such that the bound $\sum_{\Pi(u)=i} A_u[\Psi^i_{f_p,\Pi,\vec{c}}(\vec{x})]_u + a \leq \sum_{\Pi(u)=i} B_u[\Psi^i_{f_p,\Pi,\vec{c}}(\vec{x})]_u + b$ holds over a neighborhood S for some set of component functions f_p, so long as $\Psi^i_{f_p,\Pi,\vec{c}}(\vec{x}) \neq \vec{0}$. If throughout S $\hat{\Psi}^i_{f,\Pi,\vec{c}}(\vec{x})$ is a convex combination of the non-zero $\Psi^i_{f_p,\Pi,\vec{c}}(\vec{x})$ values, then the bound $\sum_{\Pi(u)=i} A_u[\Psi^i_{f,\Pi,\vec{c}}(\vec{x})]_u + a \leq \sum_{\Pi(u)=i} B_u[\Psi^i_{f,\Pi,\vec{c}}(\vec{x})]_u + b$ holds for the overall function f as well.*

Proof. This is clear from the definition of a convex combination. A linear bound which holds over some set of points will hold for any convex combination of them, and we have explicitly stated that Ψ values of $\vec{0}$, where the bound may not hold, are excluded from the combination. □

Depending on the structure of our objective function f, this lemma can be incredibly powerful or completely useless. Its utility depends entirely on what we can know about the Ψ-values of the component functions. Here we will consider the case of extreme usefulness, in which our f is an increasing posynomial. We know from Equation (18) that $\hat{\Psi}$ of each component monomial m_p is a fixed point, making the codomain of $\Psi^i_{f,\Pi,\vec{c}}$ some subset of a higher-dimensional simplex between these points!

As a result of this insight, we can place bounds between linear combinations of variables from the same variable group simply by looking at their "multiplicity fractions" in the component monomials. More formally, we have the following lemma:

Lemma 2.12. *Let $\vec{A}$ and $\vec{B}$ be vectors of coefficients and let a and b be constants, such that the bound $\sum_{\Pi(u)=i} A_u \frac{c_i y_{p,u}}{\sum_{\Pi(v)=i} y_{p,v}} + a \leq \sum_{\Pi(u)=i} B_u \frac{c_i y_{p,u}}{\sum_{\Pi(v)=i} y_{p,v}} + b$ holds over some neighborhood S, for component monomials m_p of an increasing monomial function $f = \sum_p m_p$. If the optimization of f with constraint set given by Π, $\vec{c}$ is i-consistent over S, then the bound $\sum_{\Pi(u)=i} A_u x_u + a \leq \sum_{\Pi(u)=i} B_u x_u + b$ holds for any stationary point $\vec{x}$ in S.*

Proof. This is a simple application of Lemmas 2.3 and 2.11 and Equation (18). Since f is i-consistent, it satisfies the convex combination property, and since its components are monomials their $\hat{\Psi}$ values yield the substitution above. Therefore, by Lemma 2.11, the image of S under $\Psi^i_{f,\Pi,\vec{c}}$ obeys the bound given, and any stationary point within S must be a fixed point within this image. □

These bounds are already quite useful, but so far we've only been putting constraints on the stationary points of our optimization. We may be able to gain more analytical power by considering not just the first derivatives of f, but also the second, in order to place constraints on the actual maxima of f. Of course these second derivatives will be quite complex for arbitrary f, and will tend not to give much useful information. However, if we place one more modest restriction on an increasing posynomial f, a very powerful constraint suddenly emerges:

Lemma 2.13. *Let x_u and x_v be two variables such that $\Pi(u) = \Pi(v) = i$, for the constraint partition Π and let f be an increasing posynomial. If for all p, $y_{p,u}, y_{p,v} \notin (0,1)$ and one or both of $y_{p,u}, y_{p,v}$ is zero, then there exists a **global** maximum for the optimization f,Π,$\vec{c}$ such that at least one of the variables x_u or x_v is zero.*

Proof. First, note that since all variables and coefficients are non-negative and no exponents for x_u or x_v are in the open interval $(0,1)$, both $\frac{\partial^2 f}{\partial x_u^2}$ and $\frac{\partial^2 f}{\partial x_v^2}$ are non-negative. Additionally, since no single monomial contains both x_u and x_v, $\frac{\partial^2 f}{\partial x_u \partial x_v}$ is exactly zero. We see then that the value $(\frac{\partial}{\partial x_u} - \frac{\partial}{\partial x_v})^2 f = (\frac{\partial}{\partial x_v} - \frac{\partial}{\partial x_u})^2 f = \frac{\partial^2 f}{\partial x_u^2} - 2\frac{\partial^2 f}{\partial x_u \partial x_v} + \frac{\partial^2 f}{\partial x_v^2}$ is non-negative. For any stationary point $\vec{x}$ where both x_u and x_v are non-zero, we claim that at one or both of the points $\vec{x}^+ = \vec{x} + x_v(\vec{e}_u - \vec{e}_v)$ or $\vec{x}^- = \vec{x} + x_u(\vec{e}_v - \vec{e}_u)$, f takes on a value greater than or equal to its value $f(\vec{x})$ at $\vec{x}$. If this claim holds, then we can "drop a variable" from any global maximum with non-zero x_u and x_v simply by moving to one of these points as appropriate, noting that $\vec{x}^+_v = \vec{x}^-_u = 0$. But we have already proven above that f is either linear or concave up along the $\vec{e}_u - \vec{e}_v$ direction, so the claim will certainly hold. □

We note that a polynomial f with positive coefficients satisfies the requirements of the above lemma, meaning that for a broad class of practical optimization problems we can determine variable combinations which we simply need not consider.

3 Discrete Analysis

Our work so far has focused entirely on the continuous setting, building up a framework for analyzing the most general classes of objective functions that we can wrangle into a useful theory. However, the bounds we've derived in the general case require our optimization to be highly structured in order to best apply them. When considering the case of an arbitrary function, it is highly unlikely that such helpful structure exists, and we can always come up with a degenerate case where the lemmas we've proven above simply aren't helpful at all. But this is really something to be expected; the more general the problem, the harder it is to completely solve it. In order to make use of what we've proven so far, we need discrete, problem-specific knowledge. The more we can constrain what a problem instance might look like, the more constraints we can place on the solutions to that problem. Thus having spent the previous section doing a continuous analysis of a general problem, we will now refocus our attention on performing a discrete analysis of a specific problem. It is in this duality that real solutions can be found. The continuous world can teach us how to exploit structure, but applying this knowledge requires us to explore that structure using the power of discrete math.

3.1 Haplotype Phasing Problem

The main problem we have chosen to analyze is a biological problem called haplotype phasing, both for its importance in practical applications and its role in inspiring much of the work we've already done. The continuous theory we've already built up began as a series of gradually loosening generalizations around this problem, in an attempt to see which structural properties might prove most fruitful. Most of the remainder of this work will be dedicated to a concrete analysis of the haplotype phasing problem and its properties.

The motivation for the phasing problem requires a brief primer on a few biological concepts. Humans are diploid organisms, meaning that in every cell we have two copies of each strand of DNA, one from each parent. Typically we make the infinite sites assumption, which states that there is probability zero of more than one point mutation ever affecting a particular DNA position. In other words, although every strand of DNA is made up of four possible bases, we assume that only two, the so-called major and minor alleles, are ever present at a particular location. These alleles are thus representable by zeros and ones, respectively, making each DNA strand a bit string in our representation.

In order to do many important statistical analyses, we require knowledge of the individual bit strings for both copies of a DNA strand, which are called the haplotypes. But in practice, it is very difficult to sequence these strands separately, and so instead DNA sequencing typically outputs a genotype, which we can represent as the position-wise sum of the two component haplotypes (zero for two copies of the major allele, two for two copies of the minor allele, and one for one of each). Given a multiset of n genotypes of length m, we then wish to output frequencies for each of the 2^m correctly-sized haplotypes such that the likelihood of observing the given genotypes is maximized. This particular formulation of the problem is called maximum likelihood haplotype phasing [3]. There are many other formulations of the problem ([3] gives a survey), including the so-called parsimony phasing problem studied in [4] where the goal is to explain the genotype sample using as few distinct haplotypes as possible. However, we will constrain ourselves to the maximum likelihood setting in this work.

Unfortunately, an unpublished manuscript by Hubbel gave a proof that efficiently solving this problem, or more specifically solving for the actual likelihood which the best set of frequencies yields, would allow to efficiently solve the NP-Complete Max-Clique problem. Thus it is considered unlikely that this problem can be efficiently solved in the worst case, which is to be expected since we are trying to solve for a maximizing assignment of exponentially many variables. However, biology is inherently messy and stochastic, and it is unlikely that the worst possible case will need to be solved in practice. In particular, Hubbel's proof involved representing a general graph as a phasing problem, which introduces many structural properties that a random problem instance is unlikely to have. It is our hope that, in the average case, it becomes possible to limit both the size of the good solutions and the complexity of finding them.

To show that our continuous results apply, we must first define the phasing problem as the maximization of an objective function subject to some fixed-sum constraints on a partition of its variables:

Definition: Maximum Likelihood Haplotype Phasing Let G be an n-by$-m$ matrix representing our unique input genotypes, such that G_{ij} is the jth position from the ith unique genotype. Let $\vec{p}$ be a

vector of n positive integers such that p_i is the number of times the ith unique genotype appear in our genotype sample. Let x_u be the frequency of the haplotype represented by the length-m binary string u. Then our problem is defined as follows:

$$\begin{aligned} c_1 &= c = 1 \\ \Pi(u) &= 1 \\ g_i(\vec{x}) &= \sum_{u=0^m}^{1^m} \sum_{v=0^m}^{1^m} x_u x_v \prod_{j=1}^{m} \mathbb{1}(u_j + v_j = G_{ij}) \qquad (23) \\ f(\vec{x}) &= \prod_{i=1}^{n} g_i(\vec{x})^{p_i} \qquad (24) \end{aligned}$$

where 0^m is the length-m string of all zeros, and analogously for 1^m. Here g_i gives the likelihood of a particular genotype, and our overall objective is to maximize their product.

Note that if a pair $x_u x_v$ appears in g_i, then $x_u x_s$ will not be in g_i for any other $s \neq v$, and $x_u x_v$ will not be in g_j for any other $j \neq i$. In other words, a particular haplotype only "mates" at most once in a genotype, and a particular pair of haplotypes only "mate" in up to one unique genotype. We call each such pair an "explanation," as it is a pair of haplotypes which potentially explains the given genotype.

3.2 Ψ is EM for Phasing

Since there is only one variable partition for the phasing problem, we will drop the leading index for definitions such as i-radial and i-consistent, saying "radial" instead of "1-radial',' etc. We can then start building the Ψ-mapping for f, in order to see what structure we may be able to exploit.

We first note that each genotype likelihood g_i is a positive 2-homogeneous increasing posynomial, making it consistent wherever it (and therefore f) is non-zero. Thus we can apply Equation (22) to get:

$$\hat{\Psi}_{g_i} = \frac{1}{2g_i(\vec{x})} \sum_{u=0^m}^{1^m} \sum_{v=0^m}^{1^m} x_u x_v (\vec{e}_u + \vec{e}_v) \prod_{j=1}^{m} \mathbb{1}(u_j + v_j = G_{ij}) \qquad (25)$$

We also see that f itself is a positive $2n$-homogeneous product of these 2-homogeneous functions, and therefore that it satisfies the conditions of Lemma 2.8. This means that f is also consistent whenever it is non-zero, and so we can apply Equation (17) to get:

$$\begin{aligned} \hat{\Psi}_f &= \frac{1}{2n} \sum_{i=1}^{n} 2p_i \Psi_{g_i} \\ &= \frac{1}{2n} \sum_{i=1}^{n} \frac{p_i}{g_i(\vec{x})} \sum_{u=0^m}^{1^m} \sum_{v=0^m}^{1^m} x_u x_v (\vec{e}_u + \vec{e}_v) \prod_{j=1}^{m} \mathbb{1}(u_j + v_j = G_{ij}) \qquad (26) \end{aligned}$$

This is exactly the iterative update formula used by the Expectation Maximization algorithm for phasing! This may not seem particularly exciting at first, as we've rederived an algorithm which is already extremely well-known. But the reader should consider the fact that, unlike the classical derivation of EM, this one was done completely agnostic to the language of probability. All we used to reach this point was some simple calculus and a few clever substitutions. This suggests that we may be able to gain some new insight into exactly how, why, and how well EM performs on the phasing problem. Such intuition often paves the way for new methods and algorithms which would not otherwise have been considered.

3.3 Absolute Frequency Bounds

Recall from Lemma 2.12 that we can translate linear bounds on the "multiplicity fractions" of variables into linear bounds on the variables themselves. One consequence of this is that we can place loose absolute bounds on each haplotype frequency according to its minimum and maximum possible multiplicity:

Definition: Haplotype Degree For a haplotype represented by $u \in \{0,1\}^m$, its "degree" d_u is given as

$$d_u = \sum_i p_i \sum_{v=0^m}^{1^m} (1 + \mathbb{1}(u = v))$$

In other words, the degree of a haplotype is the number of (not-necessarily unique) genotypes it has a mate in, where self-mating is counted twice.

Definition: Haplotype Necessity Let A be the set of all $1 \leq i \leq n$ such that there do not exist distinct $1 \leq j_1 \neq j_2 \leq m$ with $G_{ij_1} = G_{ij_2} = 1$. In other words, A is the set of indices for genotypes which have at most one heterozygous position. Then the the "necessity" t_u of haplotype u is given as

$$d_u = \sum_{i \in A} p_i \sum_{v=0^m}^{1^m} (1 + \mathbb{1}(u = v))$$

In other words, the necessity of a haplotype is its degree if we ignore genotypes with multiple heterozygous positions.

Lemma 3.1. *If $\vec{x}$ is a stationary point of the phasing optimization such that $f(\vec{x}) \neq 0$, then $\frac{t_u}{2n} \leq x_u \leq \frac{d_u}{2n}$, for all $u \in \{0,1\}^m$.*

Proof. Consider the value $[\hat{\Psi}_{g_i}]_u$ as given by Equation (25):

- If u does not have a mate in the ith genotype, meaning that x_u does not appear in g_i at all, then $[\hat{\Psi}_{g_i}]_u$ is exactly zero.
- If u has a mate $v \neq u$ in the ith genotype, meaning that x_u appears in the pair $x_u x_v$ in g_i, then $0 \leq [\hat{\Psi}_{g_i}]_u \leq \frac{1}{2}$. These bounds follow by applying the argument from Lemma 2.12 and noting that the multiplicity fraction of x_u in g_i is between 0 and $\frac{1}{2}$, inclusive. In addition, if this genotype has only one heterozygous position, meaning that $g_i(\vec{x}) = 2x_u x_v$, then this lower bound is raised to $\frac{1}{2}$.
- If u mates with itself in the ith genotype, meaning that $g_i(\vec{x}) = x_u^2$, then $[\hat{\Psi}_{g_i}]_u$ is exactly 1, the fixed multiplicity fraction of x_u.

But Equation (26) gives us that $[\hat{\Psi}_f]_u$, and therefore x_u at a non-zero stationary point, is simply the average of these $[\hat{\Psi}_{g_i}]_u$ values, weighted by the genotype multiplicities p_i. This then gives us exactly the bounds described. □

Of course, most phasing problems will have all the haplotype necessity values t_u as zeros, and the upper bound based on degree is typically quite loose as well. Nevertheless, the behavior of these bounds is quite intuitive, so it is satisfying to see a rigorous proof that they hold.

3.4 Mating Graph Analysis

In order to obtain better insights into the phasing problem, we'll have to delve deeper into its structure than simple absolute variable counts. We need to start taking into account which haplotypes mates with each other and where, and what if any general structure lies beneath these connections. To that end, we define and analyze the mating graph in order to better understand and exploit these more fundamental structural constraints.

Definition: Mating Graph The mating graph M_G for genotype matrix G has a vertex for each haplotype ($V = \{0,1\}^m$) and a labeled edge between two haplotypes for the genotype in which they "mate," if any. In other words, an edge (u, v) with label i is in the edge set E if and only if $u_j + v_j = G_{ij}$, for all $1 \leq j \leq m$. Note that since the rows of G are unique, there can be at most one such i for any u and v.

Having defined this graph, we'd like to get an intuitive sense of what it looks like. Two important properties, which will become especially crucial in a later lemma, are what the cycles and connected components of this graph look like. As it turns out, these properties are often constrained in very interesting ways. In order to explore them, however, it will be helpful to define a more succinct way of determining which edges might be in the mating graph:

Definition: Transition Matrix Let G' be a matrix such that $G'_{ij} = g_{ij} \bmod 2$. We define the transition matrix T to be the matrix containing the unique non-zero rows of G'.

Lemma 3.2. *If the haplotype u mates with a distinct haplotype $v \neq u$ in the ith genotype, then the form of $u \oplus v$ form a row of the transition matrix T, where $\oplus$ is the XOR operator.*

Proof. If u mates with v in the ith haplotype, then it must be by definition that $G_{ij} = u_j + v_j$, for all $1 \leq j \leq m$, and since $u \neq v$ we know that $G_{ij'} = 1$ for at least one $1 \leq j' \leq m$. As a result, $G_{ij'} \bmod 2 \neq 0$, and so one of the rows of T must be the ith row of G, elementwise modulo 2. So we have for some ℓ that $T_{i\ell} = u_j + v_j \bmod 2$ for all j, or in other words that $T_{i\ell} = u_j \oplus v_j$ for all j. □

From here, we can prove two very interesting facts about the mating graph:

Lemma 3.3. *Let r be the rank of the transition matrix T over $\mathbb{Z}_2$, the integers modulo two. In other words, r is the number of linearly independent rows of T. Then the number of connected components in the mating graph is at least 2^{m-r}.*

Proof. From Lemma 3.2 we know that haplotypes u and v can only be neighbors in the mating graph if the bits of $u \oplus v$ form a row of the transition matrix T. This is not a sufficient condition, but it is a necessary one. We then see that haplotypes s and t can only be connected if there exists some linear combination of rows of T which together combine to $s \oplus t$. Since we're working modulo 2, the only effective coefficients are zero or one, and so the set of unique linear combinations is exactly the power set of the independent rows. Thus the size of this set, 2^r, is an upper bound on the size of any connected component of M_G. Since every one of the 2^m haplotypes is a vertex in this graph, we therefore have at least 2^{m-r} connected components. □

Lemma 3.4. *If the transition matrix T has linearly independent rows, then the mating graph has no odd cycles.*

Proof. By an application of the argument from Lemma 3.3, we see that there can be a path from a haplotype u back to itself only if there is some linear combination of rows of T which sum to the zero vector. Of course, we can trivially do this by using non-zero even coefficients for some subset of the rows, but such a cycle would by necessity be even in length. The only way to have an odd cycle would be to have some odd-sized subset of the rows of T sum to the zero vector, which would imply linear dependence. □

Although we did not have time to explore further, these structural regularities within the mating graph are entirely new insights, and look like very promising topics for future work. The latter proof in particular may allow for the derivation of an average-case sparsity condition on the mating graph, showing in general that it has this regular structure except in very rare cases. We note in particular that if the condition for Lemma 3.4 holds, then every connected component of the graph is bipartite. This will become a very useful fact by the end of this section.

In the mean time, of course, we'd also like to develop some bounds not just on the abstract structure of the graph but on the frequencies themselves. Before doing so, however, we'd like to eliminate self-loops from our analysis. By the logic of Lemma 3.1 they give a constant rather than variable frequency contribution, and they tend to complicate graph properties unnecessarily. We'd also like to imbue the graph with information about the current haplotype frequencies, in order to better unify some of the upcoming analyses. Thus we make the following definitions:

Definition: Mated Frequency We define the mated frequency $\bar{x}_u$ of a haplotype u to be $x_u - \sum_i p_i \sum_{v=0^m}^{1^m} \frac{\mathbb{1}(u=v)}{n}$. In other words, the mated frequency is just the regular frequency if we neglect the contributions of self-mating.

Definition: Mate Frequency Graph We define the mate frequency graph $M'_G(\vec{x})$ to be the original mating graph M_G with self-loops removed and each vertex u augmented by assigning it a value equal to the mated frequency $\bar{x}_u$ at $\vec{x}$.

Lemma 3.5. *If $\vec{x}$ is a stationary point of the phasing optimization such that $f(\vec{x}) \neq 0$, then each vertex in the mate frequency graph $M'_G(\vec{x})$ has a value less than or equal to the total value of its non-self neighbors. In other words, a haplotype's mated frequency is upper bounded by the sum of its distinct mates' mated frequencies.*

Proof. Rather than consider f in its factored form as we have so far, we instead think of it as a single sum of monomials of degree $2n$. We see that each such monomial is a product of variable pairs from the g_i functions, and since there is only one such pair for any self-mating genotypes, each such pair will be in every monomial. Among the other pairs, each instance of x_u must by definition be accompanied by an instance of one of its potential mates x_v, which in turn must not come from a self-mating. If we let s_t be the number of self-matings for haplotype t, then we have that in jth monomial, $y_{j,u} - 2s_u \leq \sum_{v \in mates(u)}(y_{j,v} - 2s_v)$, for all j. Diving through by $2n$ and applying Lemma 2.12, we see that $x_u - \frac{s_u}{n} \leq \sum_{v \in mates(u)}(x_v - 2s_v)$, and therefore that $\bar{x}_u \leq \sum_{v \in mates(u)} \bar{x}_v$. □

Lemma 3.6. *If $\vec{x}$ is a stationary point of the phasing optimization such that $f(\vec{x}) \neq 0$ and a connected component C of the mate frequency graph $M'_G(\vec{x})$ can be bipartitioned into two vertex sets B_1 and B_2 (neglecting self-loops), then the sum of the vertex values over B_1 is exactly equal to the sum of the vertex values over B_2. In other words, the total mated frequency of haplotypes on one side of the bipartition is exactly the same as the total for the other side.*

Proof. The argument is similar to that of Lemma 3.5. Again we think of f in its fully-multiplied form. But this time we note that, neglecting the self-mating pairs, each instance of an x_u for some $u \in B_1$ is accompanied by an instance of some x_v for $v \in B_2$, *and the converse also holds*! So we have both that $\sum_{u \in B_1}(y_{j,u} - 2s_u) \leq \sum_{v \in B_2}(y_{j,v} - 2s_v)$, and vice versa, for all j. Together with Lemma 2.12, this gives us that $\sum_{u \in B_1} \bar{x}_u = \sum_{v \in B_2} \bar{x}_v$. □

Combining this lemma with Lemmas 3.3 and 3.4, we see that if the transition matrix is indeed linearly independent we have at least 2^{m-n} bipartite connected components, all of which with equal halves. These constraints could prove very helpful in the formulation of an approximate or exact algorithm for phasing, and are certainly a topic we are interested in pursuing. For the time being, though, they're just incredibly interesting!

3.5 Singleton Exclusion

So far we've derived a lot of bounds on frequencies, but we have yet to consider the question of which haplotypes we can simply leave out. This is perhaps the most important question of all, because there are exponentially many variables and we'd like to eliminate as many of them as we can *before* performing any computation on them. In the phasing problem the most likely candidates for exclusion are the so-called "singletons," i.e. haplotypes which appear in only a single genotype. Previous lemmas tell us that such haplotypes have low upper bounds on their frequencies, and that they must have frequency no greater than that of their single mate, and so they seem like relatively safe candidates for exclusion.

Definition: Singleton Haplotype A haplotype u is a singleton if it has only one neighbor in the mating graph, and it is not a self-loop.

Lemma 3.7. *Let u and v be two singleton haplotypes which are mated together in the ith genotype. In other words, the edge (u, v) is in the mating graph with label i, and neither u nor v has any other incident edges that aren't self-loops. If any other pair of haplotypes s and t also mate in the ith haplotype, meaning that (s, t) is in the mating graph with label i, then there exists a **global** maximum for this phasing problem such that $x_u = x_v = 0$.*

Proof. Given any point $\vec{x}$ where x_u or x_v is non-zero, we will construct a point $\vec{x^*}$ such that $f(\vec{x^*}) \geq f(\vec{x})$, but $x^*_u = x^*_v = 0$. To do this, we simply set $x^*_s = x_s + x_u$, $x^*_t = x_t + x_v$, and $x^*_u = x^*_v = 0$. We then note that

$$\begin{aligned} 2x^*_s x^*_t + 2x^*_u x^*_v &= 2(x_s + x_u)(x_t + x_v) \\ &= 2x_s x_t + 2x_s x_u + 2x_t x_v + 2x_u x_v \\ &\geq 2x_s x_t + 2x_u x_v \end{aligned}$$

and so our new $\vec{x^*}$ yields an equal or improved value for g_i. Since u and v are singletons, reducing them does not reduce any other g_j, while increasing x_s and x_t might. So dropping this singleton pair can only improve the overall likelihood. □

This lemma in particular is quite powerful, as it allows us to proactively drop all singleton-singleton pairings from consideration, no questions asked. While this thesis is intended to be a purely theoretical treatment of phasing and optimization, this is one of the results which shows the most potential for future practical work on the haplotype phasing problem.

We can do even better than just dropping singleton-singleton pairs, though. In fact, we can get rid of most if not all of the singleton haplotypes:

Lemma 3.8. *Let u and v be two singleton haplotypes which mate with s and t, respectively, in the ith genotype. In other words, the edges (u, s) and (v, t) are in the mating graph with label i, and there are no other edges incident on u or v. Then there exists a* ***global*** *maximum for this phasing problem where one or both of x_u or x_v are zero.*

Proof. The proof is similar to the one for Lemma 3.7. Given any point $\vec{x}$ where neither x_u nor x_v are zero, we will construct a point $\vec{x^*}$ such that $f(\vec{x^*}) \geq f(\vec{x})$, but x_u or x_v is zero. To do this, we simply set $x^*_u = x_u + x_v$ and $x^*_v = 0$ if $x_s \geq x_t$, or $x^*_v = x_u + x_v$ and $x^*_u = 0$ otherwise. We then see that $2x^*_u x^*_s + 2x^*_v x^*_t = 2(x_u + x_v)\max(x_s, x_t) \geq 2x_u x_s + 2x_v x_t$, and so our new $\vec{x}^*$ yields an equal or improved value for g_i. Since u and v are singletons, reducing them does not reduce any other g_j, and no other variables are changed by this transformation. So dropping one of the singletons can only improve the overall likelihood. □

This lemma has the potential to greatly reduce the number of haplotypes we must consider, but there is an important caveat that we have to keep in mind. Specifically, although this lemma tells us that we need not use more than one singleton-based explanation per genotype, we don't know which, if any, of those explanations we might have to preserve. So we need not consider using arbitrary subsets of such explanations anymore, but we still must consider the possibility that any single one of them might be non-zero. It is only those explanations which are made up of two singletons that we can safely remove with impunity.

4 Conclusion

By first generalizing and then slowly re-narrowing our problem domain, we have managed to discover a set of properties and a corresponding theoretical framework which has yielded new insights into a large class of continuous optimizations. Rather than focusing on finding a solution, we first took a step back and explored which problem details actually mattered, and which we could leave to vary freely. In this way, we managed to preserve maximum generality from the start, learning a great deal about which types of specificity we can exploit and how to do so.

We then turned our attention from this continuous analysis to an in-depth discrete analysis of the haplotype phasing problem, and found that it fit our model so well that we actually re-derived the EM algorithm! This derivation, however, was built from first principles, and so we were able to gain deep insights into the phasing problem which are not apparent at face value from the EM formula. We discovered surprisingly regular and disconnected structure in the haplotype mating graph, opening the door to future work on average-case sparsity conditions for the phasing problem. Finally, we derived a pair of rules which have the potential to eliminate the vast majority of variables from an average phasing problem. How best to exploit these rules in a practical setting is again a target for future work.

Overall, we were able to derive an impressive array of new results in both the continuous and discrete settings, with the potential to go even broader with our continuous results and deeper with our discrete ones in upcoming research.

Acknowledgements

I would like to thank Professor Sorin Istrail for his constant advice and encouragement, for putting up with many an excited email that turned out to be a flawed or just completely unchecked proof, and of course for serving as my thesis advisor. I would also like to thank Professor Eli Upfal for serving on my thesis committee, and all my classmates and professors at Brown for giving me four of the best years of my life. My friends and family back home also deserve a special thank you for all the love and support they've given me over the years. As this is a thesis in computer science, I'd also very much like to thank Josh Merlis, my high school CS teacher, who got me started on the path I'm on today. Finally, since I am writing this thesis while enrolled at a not inexpensive institution of higher learning, I'd like to thank my grandmother again for helping to pay the bills, not to mention letting me repeatedly borrow a very nice old car.

References

[1] Edwards, A.W.F. 1992. Likelihood, Expanded Edition. Johns Hopkins University Press, Baltimore.

[2] Dempster, A.P.; Laird, N.M.; Rubin, D.B. (1977). "Maximum Likelihood from Incomplete Data via the EM Algorithm". Journal of the Royal Statistical Society. Series B (Methodological) 39 (1): 1-38. JSTOR 2984875. MR 0501537.

[3] Bjarni V Halldorsson, Vineet Bafna, Nathan J. Edwards, Ross Lippert, Shibu Yooseph, Sorin Istrail, "Combinatorial problems arising in SNP and haplotype analysis", In Fourth International Conference on Discrete Mathematics and Theoretical Computer Science, (CDMTCS03), Springer, pp. 26-47, 2003.

[4] Roded Sharan, Bjarni V Halldorsson, Sorin Istrail, "Islands of Tractability for Parsimony Haplotyping", In IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 3, no. 3, pp. 303-311, 2006.