.

1 Review of wired optimization

$$\min \sum_{\phi} U\left(f_{\phi}\right)$$

$$\sum_{\{\phi : l \in P(\phi)\}} f_{\phi} \leq R_{l}$$

where R_l is the capacity of link l.

$$L\left(f, \mu\right) = \sum_{\phi} U\left(f_{\phi}\right) + \sum \mu_{l} \left(\sum_{\{\phi : l \in \mathcal{P}(\phi)\}} f_{\phi} - R_{l} \right)$$

$$\begin{aligned} q\left(\mu\right) &= \inf_{f_{\phi}} \sum_{\phi} U\left(f_{\phi}\right) + \sum \mu_{l} \left(\sum_{\{\phi : l \in \mathcal{P}(\phi)\}} f_{\phi} - R_{l} \right) \\ &= \inf_{f_{\phi}} \sum_{\phi} U\left(f_{\phi}\right) + \sum_{l} \mu_{l} \sum_{\{\phi : l \in \mathcal{P}(\phi)\}} f_{\phi} - \sum \mu_{l} R_{l} \\ &= \sum_{\phi} \inf_{f_{\phi}} \left(U\left(f_{\phi}\right) + f_{\phi} \sum_{\{l : l \in P(\phi)\}} \mu_{l} \right) - \sum \mu_{l} R_{l} \end{aligned}$$

Let

$f_{\phi}^{*}\left(\mu\right) = \arg\min_{f_{\phi}} U\left(f_{\phi}\right) + f_{\phi} \sum_{\{l : l \in P(\phi)\}} \mu_{l}$ (a local problem/can be distribu solve the dual problem

$$\begin{aligned} & \max q(\mu) \\ = & \max \underbrace{\sum_{\phi} U\left(f_{\phi}^{*}(\mu)\right)}_{\text{how can this be distributed}} + \sum_{l} \mu_l \underbrace{\sum_{\{\phi : l \in \mathcal{P}(\phi)\}} f_{\phi}^{*}(\mu)}_{\text{flow across link } l} - \sum \mu_l R_l \end{aligned}$$

E.g., steepest descent

$$\nabla q(\mu) = \left[\begin{array}{c} \sum_{\{\phi:1\in\mathcal{P}(\phi)\}} f_{\phi}^{*}(\mu) - R_1, \sum_{\{\phi:2\in\mathcal{P}(\phi)\}} f_{\phi}^{*}(\mu) - R_2, \\ \dots, \sum_{\{\phi:L\in\mathcal{P}(\phi)\}} f_{\phi}^{*}(\mu) - R_L \end{array} \right]$$

"luckily" the gradient is separable.

$$\begin{aligned} \mu_l(k+1) &= \left[\mu_l(k) + s_k \frac{d}{d\mu_l} q(\mu)\right]^{+} \\ &= \left[\mu_l(k) + s_k \left(\sum_{\{\phi:l\in\mathcal{P}(\phi)\}} f_{\phi}^{*}(\mu) - R_l\right)\right]^{+} \end{aligned}$$

which can be distributed

.

2 Wireless Network optimization

Let f_ϕ be the flow along path ϕ and $P(\phi)$ be the path, i.e., $l \in P(\phi)$ means that link l is along flow ϕ

Let $R(v,l)$ be the flow across link l when assignment v is used.

A schedule is convex sum of assignments. Thus, the rate over link l for a schedule $\{\alpha_v\}$ is $\sum_{v \in V} \alpha_v R(v,l)$, where $\sum \alpha_v = 1$and V is the set of all considered assignments.

Recall that if we don't consider all assignments, then we get a sub optimal solution. However...

$$\min \sum_\phi U(x_\phi)$$

$$\text{s.t.} \sum_{\{\phi : l \in P(\phi)\}} f_\phi \leq \sum_{v \in V} \alpha_v R(v,l) \text{ for all } l$$

$$\sum \alpha_v \leq 1$$

.

3 Lagrangian

$$L(x,\alpha,\mu,\lambda) = \sum_{\phi} U(x_\phi) + \sum \mu_l \left(\sum_{\{\phi: l\in\mathcal{P}(\phi)\}} f_\phi - \sum_{v\in V} \alpha_r R(v,l) \right) + \lambda \left(\sum_{v\in V} \alpha_v - 1 \right)$$

$$q(\mu,\lambda) = \inf_{x,\alpha} \sum_{\phi} U(x_\phi) + \sum_{l} \mu_l \left(\sum_{\{\phi: l\in\mathcal{P}(\phi)\}} f_\phi - \sum_{v\in V} \alpha_r R(v,l) \right) + \lambda \left(\sum_{v\in V} \alpha_v - 1 \right)$$

There are several approaches. One is as follows

After some manipulations, the dual function is found to be

$$q(\mu,\lambda) = \inf_{f,\alpha\geq 0} \sum_{\phi\in\Phi} U(f_\phi) w_\phi - \lambda + \sum_{l=1}^{L} \mu_l \sum_{\{\phi: l\in\mathcal{P}(\phi)\}} f_\phi - \sum_{v\in V} \alpha_v \left(\sum_{l=1}^{L} R(v,l)\mu_l - \lambda \right).$$

.

4 λ

We immediately note that if $\sum_{l=1}^{L} R(v,l)\,\mu_l - \lambda > 0$ for some v, then $q(\mu,\lambda) = -\infty$. Hence, we restrict the domain of q, to be such that $\sum_{l=1}^{L} R(v,l)\,\mu_l - \lambda \leq 0$. On the other hand, when solving the dual problem, an objective is to maximize q with respect to λ. It is not hard to see that this is equivalent to minimizing λ over the domain $\sum_{l=1}^{L} R(v,l)\,\mu_l - \lambda \leq 0$. Thus,

$$\lambda^* = \max_{v \in V} \sum_{l=1}^{L} R(v,l)\,\mu_l. \tag{1}$$

5 Dual again

Therefore, we can rewrite the dual function as[1],

$$\begin{aligned} q(\mu) &= \inf_{f\geq 0} \sum_{\phi\in\Phi} U(f_\phi) + \sum_{l=1}^{L} \mu_l \sum_{\{\phi : l\in\mathcal{P}(\phi)\}} f_\phi - \max_{v\in V} \sum_{l=1}^{L} R(v,l)\,\mu_l, \\ &= \left(\sum_{\phi\in\Phi} \left(\inf_{f_\phi} U(f_\phi) + f_\phi \sum_{\{l : l\in\mathcal{P}(\phi)\}} \mu_l \right) \right) - \max_{v\in V} \sum_{l=1}^{L} R(v,l)\,\mu_l \end{aligned}$$

As in wired case,

$$f_\phi^*(\mu) = \arg\min_{f_\phi} U(f_\phi) + f_\phi \sum_{\{l : l\in\mathcal{P}(\phi)\}} \mu_l$$

Then

$$q(\mu) = \left(\sum_{\phi\in\Phi} U\left(f_\phi^*(\mu)\right) + \sum_{l=1}^{L} \mu_l \sum_{\{\phi : l\in\mathcal{P}(\phi)\}} f_\phi^*(\mu) \right) - \max_{v\in V} \sum_{l=1}^{L} R(v,l)\,\mu_l$$

Before we got "lucky" that the gradient was such that this could be solved in a distributed fashion. Here, we are not so lucky.

We proved last time

$$\frac{d}{d\mu_l} \left(\sum_{\phi\in\Phi} U\left(f_\phi^*(\mu)\right) + \sum_{l=1}^{L} \mu_l \sum_{\{\phi : l\in\mathcal{P}(\phi)\}} f_\phi^*(\mu) \right) = \sum_{\{\phi : l\in\mathcal{P}(\phi)\}} f_\phi^*(\mu),$$

(which looks hard to believe, but is true).

[1] There are other, more straightforward ways to arrive at (??). However, these methods do not provide the important expression (1).

What about

$$\frac{d}{d\mu} \max_{v \in V} \sum_{l=1}^{L} R(v,l)\,\mu_l \neq \max_{v \in V} \frac{d}{d\mu_l} \sum_{l=1}^{L} R(v,l)\,\mu_l = \max_{v \in V} \frac{d}{d\mu_l} \sum_{l=1}^{L} R(v,l)\,.$$

.

Example of max function

$$h(x) = \max(-2x, 3x) = \begin{cases} -2x \text{ for } x < 0 \\ 3x \text{ for } x \geq 0 \end{cases}$$

for most x, this is differentiable,

$$h'(x) = \begin{matrix} -2 \text{ for } x < 0 \\ 3 \text{ for } x > 0 \end{matrix}$$

At $x = 0$ there is a confusion. On the one hand, both gradients meet the requirement that the the function is above the linear approximation. In fact, the convex combinations of these two gradients meets this requirement

$$\begin{aligned} L(x) &= h(0) + (\alpha(-2) + (1-\alpha)(3))\, x \\ L(x) &\leq h(x) \end{aligned}$$

The subdifferential is

$$\left\{ g : g = \alpha_1(-2) + \alpha_2(-3)\,,\ \sum_{i=1,2} \alpha_i = 1 \right\}$$

In general, a subgradient (of a convex function) at x_0 is a linear function $g(x)$ such that $g(x_o) = f(x_o)$ and $g(x) \leq f(x)$. The convex combination of such functions is the subdifferential. In most cases (many all?), the subdifferential has a finte set of extreme points g_i, so

$$\partial f = \left\{ g : g = \sum \alpha_i g_i,\ \sum \alpha_i = 1 \right\}$$

7 Linear approximation and the subgradient

The most useful thing about derivatives is that they yield linear approximations of the function. Subdifferntials yield linear approximations, but it is more complicated.

Example. Directional derivative of $f(x) = \max_i \left(m_i^T (x - x_o)\right)$ for $x, m \in R^n$ Let $Df(d)$ be the derivative in direction d. Claim, at $x = x_o$

$$Df(d) = \max_i m_i d$$

$$\begin{aligned} f(x_o + hd) &= \max_i m_i (x_o + hd - x_o) = h \max_i m_i d \\ f(x_o + hd) &= f(x_o) + hDf(d). \end{aligned}$$

Example. $f(x) = \max_i g_i(x)$ with $g_i'(x_o) = m_i$. Suppose that at x_o $g_j(x_o) = \max_i g_i(x_o)$ for all $j \in J$. Then

$$Df(d) = \max_{j \in J} \nabla g_j(x_o) \cdot d = \max_{j \in J} m_i$$

so small h

$$\begin{aligned} f(x_o + hd) &= \max_i g_i(x_o + hd) = \max_{j \in J} g_j(x_o + hd) \\ &= \max_{j \in J} g_j(x_o) + h \nabla g_j(x_o) \cdot d \\ &= f(x_o) + h \max_{j \in J} \nabla g_j(x_o) \cdot d = f(x_o) + hDf(d) \end{aligned}$$

Example,

$$f(x) = \max(-2x, 3x, 12x - 11)$$

at $x = 0$, $Df(1) = 3$, $Df(-1) = -2$, at $x = 1$, $Df(1) = 12$, $Df(-1) = 3$.

8 Direction of descent

Important note. The subgradient at $x = 0$ in the positive direction is positive. If the function was differentiable, this would imply that the direction of descent is the negative direction. But this is not necessarily the case. We would need to check if it is. Thus, the trick of finding the direction in increase and going in the opposite direction does not necessarily work.

Two options.

1. $x_{k+1} = x_k - s_k g$ where g is any vector in the sub differential. It turns out that this approach works. However, unlike the differentiable case $f(x_{k+1})$ might be larger than $f(x_k)$. So we cannot use a line search to determine s_k. Also, s_k cannot be fixed. For convergence, $s_k \to 0$ but $\sum (s_k)^2 = \infty$. If s_k is fixed, then x_k will oscillate near to the optimal point. The size of the oscillation depends on the problem.

2. There does exist directions of descent. The differential is $\sum_{j\in J} \alpha_j g_j$ with $\sum_{j\in J} \alpha_j = 1$. The direction of steepest descent is the subgradient that solves

$$\min \left\| \sum_{j\in J} \alpha_j g_j \right\|^2$$

$$\sum_{j\in J} \alpha_j = 1$$

which can be solved. Also, at the optimal point, there exist α such that $\sum_{j\in J} \alpha_j g_j = \vec{0}$.

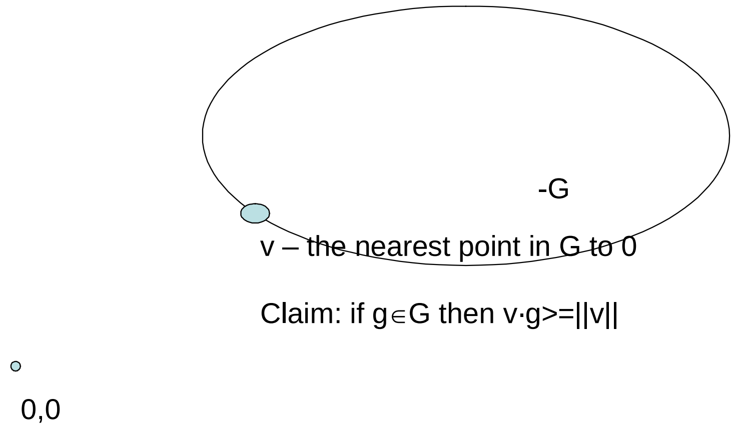

Figure 1:

9 Direction of descent

Let G be the subdifferential. Proof that if $v \in -G$ and $||v||$ is the min of all $g \in -G$, then v is direction of steepest descent

From pic $v \cdot g \geq ||v||^2$.

$$Df\left(x_o, v\right) = \max_{g \in G} g \cdot v = -\min_{g - G} g \cdot v = -||v||^2$$

So in the direction v the grad is v. Also, not that this is a direction of descent. This is just like differentiable functions.

But is it the direction of steepest descent? Let's search for the direction of steepest descent, i.e.,

$$\min_{d} Df\left(x_o, d\right)$$

Let $||d|| = 1$, be some direction

$$
\begin{aligned}
Df\left(x_o, d\right) &= \max_{g \in G} g \cdot d = -\min_{g \in -G} g \cdot d \\
&= -\min_{g - G} \left(||g|| ||d|| \cos\left(\theta_{g,d}\right)\right) \\
&= -\min_{g - G} \left(||g|| \cos\left(\theta_{g,d}\right)\right) \\
&\geq \min_{g \in -G} ||g|| = -||v||
\end{aligned}
$$

$$q(\mu) = \left(\sum_{\phi \in \Phi} U\left(f_\phi^*(\mu)\right) + \sum_{l=1}^{L} \mu_l \sum_{\{\phi : l \in \mathcal{P}(\phi)\}} f_\phi^*(\mu) \right) - \max_{v \in V} \sum_{l=1}^{L} R(v,l)\, \mu_l$$

Before we got "lucky" that the gradient was such that this could be solved in a distributed fashion. Here, we are not so lucky.

Consider

$$-\max_{v \in V} \sum_{l=1}^{L} R(v,l)\, \mu_l$$

Note that this is a min, as oppose to what we discussed above. This needs a supergradient. Find V^* such that

$$v^* \in V^* \text{ implies that } -\sum_{l=1}^{L} R(v^*,l)\, \mu_l = -\max_{v \in V} \sum_{l=1}^{L} R(v,l)\, \mu_l$$

Then the supergradients are $\{R(v,:) : v \in V^*\}$.

We can apply either of the options above.

HOWEVER, in either case, we must compute V^*. To do this, you must be aware of μ_l for all l. This cannot be distributed.

Recall

$$\mu_l(k+1) = \mu_l(k) + s_k \left(\sum_{\{\phi : l \in \mathcal{P}(\phi)\}} f^*_\phi(\mu) - g_l \right)$$

where g is the sub (super)gradient in the superdifferential. Example, if any of the supergradients is used

$$\mu_l(k+1) = \mu_l(k) + s_k \left(\sum_{\{\phi : l \in \mathcal{P}(\phi)\}} f^*_\phi(\mu) - R\left(v^{\%}, l\right) \right)$$

where

$$-\sum_{l=1}^{L} R\left(v^{\%}, l\right) \mu_l(k) = -\max_{v \in V} \sum_{l=1}^{L} R(v, l)\, \mu_l$$

Thus, $v^{\%}$ is a particular assignment of the perhaps many in V^*. (Are there many?? yes, we'll see).

What happens of two routers use different vs, e.g., $v^{\%}$ and $v^{\#}$. Then there may be collisions, or, in effect, some other v is used... In this case, convergence is a bit tricky. There is a paper that says convergence may still occurs, but....

11 $V^*(\mu^*)$

We'll see that at convergence, $V^*(\mu^*)$ is the set of assignments that are best. A schedule should multiple between these assignments.

Intuition: Economic point of view. μ_l^* is the optimal price for each link. the network seeks to get the most \$, so if provides bit-rates to maximize \$. Thus, it seeks to maximize $\sum_{l=1}^{L} R(v,l)\,\mu_l^*$. Which ones are the best are used, and the others are neglected. The network has no preference between the $v \in V^*$. It uses them to meet the demand on each link, which is $\sum_{\{\phi:l\in\mathcal{P}(\phi)\}} f_\phi^*(\mu)$.

Steepest descent

$$\mu_l(k+1) = \mu_l(k) + s_k\left(\sum_{\{\phi:l\in\mathcal{P}(\phi)\}} f_\phi^*(\mu(k)) - g_l^{++}(\mu(k))\right)$$

where $\left[\sum_{\{\phi:l\in\mathcal{P}(\phi)\}} f_\phi^*(\mu(k)) - g_l^{++}(\mu(k)), ...,\right]$ is the direction of steepest descent, i.e., $g_l^{++}(\mu(k)) = \sum \alpha_v R(v,l)$ where $\sum \alpha_v = 1$ and $v \in V^*$.

At optimality, the direction of steepest descent is the zero vector i.e,

$$\sum_{\{\phi:l\in\mathcal{P}(\phi)\}} f_\phi^*(\mu(k)) - g_l^{++}(\mu(k)) = 0,$$

thus, there is a linear combination of $R(v,l)$ that exact achieves $\sum_{\{\phi:l\in\mathcal{P}(\phi)\}} f_\phi^*(\mu(k))$ and the vs that achieve this are the ones in V^*. Only the assignments in V^* are needed.

12 Optimal Schedule

Finding the α

$$\min \sum_{\phi} U\left(f_{\phi}\right)$$
$$\text{s.t.} \sum_{\{\phi:l\in P(\phi)\}} f_{\phi} \leq \sum_{v\in V} \alpha_v R\left(v,l\right) \text{ for all } l$$
$$\sum \alpha_v \leq 1$$

We know f_{ϕ} :

$f_{\phi}^{*}\left(\mu\right) = \arg\min_{f_{\phi}} U\left(f_{\phi}\right)+f_{\phi} \sum_{\{l:l\in P(\phi)\}} \mu_l$ (a local problem/can be distribu

So the optimization problem is

$$\min \sum_{\phi} U\left(f_{\phi}^{*}\left(\mu^{*}\right)\right)$$
$$\text{s.t.} \sum_{\{\phi:l\in P(\phi)\}} f_{\phi}^{*}\left(\mu\right) \leq \sum_{v\in V} \alpha_v R\left(v,l\right) \text{ for all } l$$
$$\sum \alpha_v \leq 1$$

all we need is α (the schedule) i.e., find a such that $\sum_{\{\phi:l\in P(\phi)\}} f_{\phi}^{*}\left(\mu\right) \leq \sum_{v\in V} \alpha_v R\left(v,l\right)$ for all l and $\sum \alpha_v \leq 1$

$$\min \sum \alpha_v$$
$$st. \sum_{\{\phi:l\in P(\phi)\}} f_{\phi}^{*}\left(\mu\right) \leq \sum_{v\in V^{*}} \alpha_v R\left(v,l\right)$$

LP problem.

How many dimensions? is $\#V^* = 2^L$. No, #V$^* \leq L$ which is quite small (100's or maybe 1000s)

13 size of V^*

Consider the space of bit-rates,

$$\begin{aligned} & Co\left\{R(v,:) : v \in V\right\} \\ : = & \left\{\sum_{v\in V} \alpha_v R(v,:) : \sum \alpha_v \leq 1,\, \alpha_v \geq 0\right\} \end{aligned}$$

.

$R(v,:) \in R^L$, and so $Co(R(v,:)) \subset R^L$

Also, $Co(R(v,:))$ is a polytope and the extreme points are $R(v,:)$.

The optimal schedule is $\sum_{v\in V} \alpha_v^* R(v,:)$ is a point on the edge of the polytope. Any point on the edge of the polytope is define by at most L extreme points. Hence, V^* has at most L points

14 Summary of wireless optimization difficulties

Three difficulties facing wireless optimization

1. The number of independent set is large, the number of assignments is 2^L.
2. The subgradient has problematic convergence.
3. The determining of a single subgradient (a descent direction or not) requires global coordination to select the best over all 2^L assignments.

In the offline problem (mesh net), 2 and 3 are not so bad. But one is a problem.

On the other hand, we just showed that the optimal schedule only has L points, so why do we bother with 2^L.

15 Finding a small V

On the one hand, we can pick all assignments (independent sets, but what is independent, we could always go to smaller bit-rate)

Consider $V_{\#} \subsetneq V$, i.e., a subset of all assignments.

Suppose that $\mu^*_{\#}$ is the optimal link costs for this set.

Is there a better assignment V?

From economic interpretation, better means brings on more \$, so $v_{\$} \in V$ and $v_{\$} \notin V_{\#}$ but

$$-\sum_{l=1}^{L} R\left(v_{\$}, l\right) \mu^*_{\#,l} < -\max_{v \in V_{\#}} \sum_{l=1}^{L} R\left(v, l\right) \mu^*_{\#,l}$$

Thus, once we find $\mu^*_{\#}$ we can search for $v_{\$}$. Note that the above eq guides this search.

Goal, maximize $\sum_{l=1}^{L} R\left(v_{\$}, l\right) \mu^*_{\#,l}$. This is the maximum weighted independent set problem. In theory, the maximum weighted independent set problem is no easier than the independent set problem (set weights to 1). we find that in realistic networks, the heuristic techniques work well.

Compare to other techniques: find all independent sets or find all cliques. Here we just find the maximum weighted independent set.

Algorithm

0. Pick some V^0. k=0
1. Solve for optimal μ^k
2. Find better assignment $v_{\$}$
3. if search seceded, goto 1
4. otherwise stop

If the search for the maximum independent set fails, then the current solution might not be optimal. But we find that it is very close (0.5%)

16 Power control

If power control is permitted, then the set of assignments is larger than 2^L. The space is $[0,1]^L$, the L-dim unit cube.

We can use a similar approach as above.

$$\max_{\vec{p}} \sum_l \mu_l \log_2 \left(1 + \frac{H_{l,l}p_l}{\sum H_{k,l}p_k + N}\right).$$

difficulty: in general, this maximization is not concave. It is concave in the high SNR limit, i.e.,

$$\max_{\vec{p}} \sum_l \mu_l \log_2 \left(\frac{H_{l,l}p_l}{\sum H_{k,l}p_k + N}\right).$$

This never occurs in real networks. E.g., if a router has an incoming link and an outgoing link, if they both transmit at the same time, the incoming link will have very low SNR.

However, within an independent set (where independent means high SNR over each link), then the above is concave