General Model for Wireless Data Systems

We now formalize a model for wireless data networks to capture the properties discussed previously. We will also formulate the capacity region of such a system and see how that can be used to reason about optimal scheduling. In doing so, we will also introduce some key properties in linear programming.

Model

We desire a model that can help us answer the following question: *What is the optimal capacity of a wireless system, and how can we achieve it?*

To simplify, let's assume that each source/destination has a single **path** through the network, and that the path is given. A path is a collection of one or more connected links. A source/destination pair is called a **flow**. In other words, we assume each flow is using a single path.

Formally, each flow $s = 1, 2, ..., S$ has one path that consists of a subset of the links $l = 1, ..., L$ in the network. We specify the flow-link assignments through a matrix of binary values H_s^l:

$$\begin{cases} H_s^l = 1 & \text{if the path of flow s passes through link l,} \\ H_s^l = 0 & \text{otherwise.} \end{cases} \tag{1}$$

We further assume a set of possible transmission schemes is chosen, and therefore each node can only choose its action among a set. Formally, let $p_l(t)$ denote the **action** chosen by the transmitting node of link l at time t (each link has a transmitting and receiving node). We collect these actions across links in a vector $\vec{p}(t) = [p_1(t), ..., p_L(t)]$, which is the **global action vector**. Further, we let $\mathcal{H}$ denote the set of all feasible $\vec{p}(t)$, i.e., any chosen $\vec{p}(t) \in \mathcal{H}$.

Once the action $\vec{p}(t)$ is chosen, certain service rates for all links can be determined. We let $r_l(t)$ denote the **rate** for link l at time t (e.g., in bits/sec), and $\vec{r}(t) = [r_1(t), ..., r_L(t)]$.

Due to interference across links, in general, $r_l(t)$ depends not only on $p_l(t)$, but also on the actions chosen by other links. Further, it depends on the time-varying channel conditions. We let $K(t)$ denote the **global channel state** at time t, which we can think of as a categorical quantity: $K(t) = 1$ is one possible channel state configuration, $K(t) = 2$ is another, and so on. In other words, we express the rate as the following function:

$$\vec{r}(t) = g(\vec{p}(t), K(t)). \tag{2}$$

The function g is assumed to be given, and it is determined by the coding/modulation/transmission scheme that is chosen.

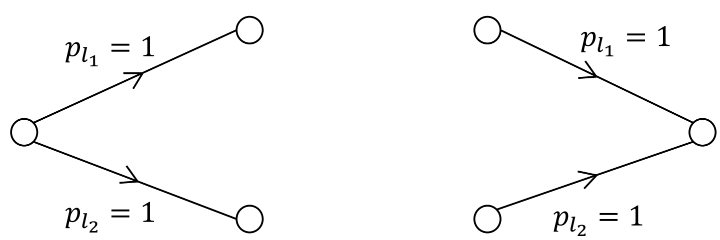
This is a very general model. In the following, we will walk through some examples to see how it encapsulates specific types of wireless networks.

Collision Channel

In a collision channel, all nodes share a common frequency band. In each timeslot, the action of a node is either to transmit ($p_l(t) = 1$) or not to transmit ($p_l(t) = 0$). We can achieve a constant rate if no collisions occur.

Example A: First consider a collision channel in which a node cannot transmit to two other nodes at the same time. Neither can a node receive from two other nodes at the same time. This may arise in a CDMA-based wireless network where a unique spreading code is used on each link (such as in Bluetooth), but each node only has one radio.

Visually, we must avoid the following primary conflicts:



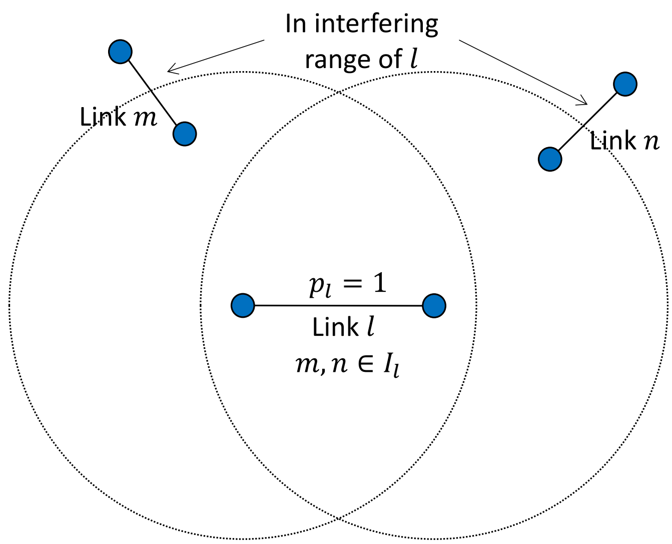
This determines the set $\mathcal{H}$.

Example B: Now consider a more general case where each link l has a set of interfering links, $\mathcal{I}_l$.

- If $p_{l'}(t) = 1$ for any of the interfering links $l' \in \mathcal{I}$, then $p_l(t) = 0$ and $r_l(t) = 0$.

- Otherwise, if $p_l(t) = 1$, then the link achieves a constant rate $r_l(t) > 0$.

Example A is a special case where interfering links are those sharing a common transmit or receive node. In general, links interfere with each other when they fall in a certain radius, as depicted below:



WiFi standards (IEEE 802.11) are commonly based on the implications of this collision channel interference model.

SINR-based System

The collision model is based on the assumption that interference must be avoided. As we saw in our discussion on voice systems, there are other ways to manage interference. In an SINR-based model, the actions are the power transmission levels of links. Letting i index the transmitter and j index the receiver of link $l = (i, j)$, we model the SINR as

$$\mathrm{SINR}_{ij}(t) = \frac{p_i(t) g_{ij}}{\sum_{k \neq i} p_k(t) g_{jk} + n_0}, \tag{3}$$

where $\mathrm{SINR}_{ij}(t)$ is the SINR level at the link from node i to node j, $p_i(t)$ is the transmission power of node i, g_{ij} is the signal attenuation experienced, and n_0 is the background noise. Here, g_{ij} includes path loss, fading, and coding/decoding factors: ideally, the g's will be as large as possible for desired links (i, j) and as small as possible for interfering links $(k, j), k \neq i$.

The function g determining the link rates in (2) will vary depending on the coding/modulation scheme in use. For example:

1. We could have constant rates when the SINR is above a threshold, i.e., $r_{ij}(t) = R$ if $\mathrm{SINR}_{ij}(t) > \gamma$ and $r_{ij}(t) = 0$ otherwise.
2. Allowing adaptive modulation and coding (AMC), the rate will approach the Shannon bound for a given SINR:
$$r_{ij}(t) = B \log_2(1 + \mathrm{SINR}_{ij}(t)) \tag{4}$$
3. On the other hand, suppose we have a CDMA scheme in place. In CDMA systems with moderate gains, we will have a small SINR, for which we can use a linear approximation of (4). Before despreading, we have:
$$r_{ij}(t) \approx B \cdot \mathrm{SINR}_{ij}(t) = B \cdot \frac{p_i(t) g_{ij}}{\sum_k p_k(t) g_{kj} + n_0} \tag{5}$$
(CDMA may maintain a certain target SINR η after de-spreading to get a symbol rate R_s. Assuming $n_0 = 0$, the spreading gain W can be designed as follows:
$$\frac{p_i g_{ij}}{\frac{1}{W} \cdot \sum_k p_k g_{kj}} = \eta \to W = \frac{\eta}{p_i g_{ij} / (\sum_k p_k g_{kj})}, \tag{6}$$
which leads to an effective rate of
$$\text{rate} = \frac{1}{W} \cdot R_s = \frac{R_s}{\eta} \cdot \frac{p_i g_{ij}}{\sum_k p_k g_{kj}}. \tag{7}$$

For this SINR-based model, $\mathcal{H}$ corresponds to the power constraints, which emerge from physical device/standard limitations. We also may specify some constraints in terms of minimum SINR levels.

Note also that time-varying channel gains can be incorporated into such SINR models easily. For simplicity, we will assume $K(t)$ takes one of the values in a finite set $\{1, ..., K\}$. We then define a probability mass function for $K(t)$ to give the proportion of time $K(t) = k$:

$$\lambda_k = P\{K(t) = k\} \geq 0,\ k = 1, ..., K; \qquad \sum_k \lambda_k = 1 \tag{8}$$

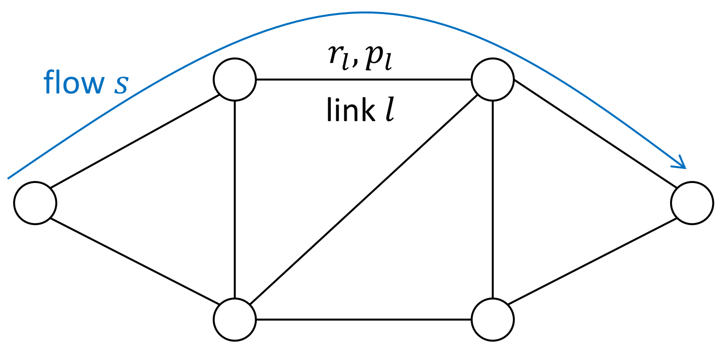
Aside from the collision channel and SINR-based system, we can also fit other protocols we have seen into this general model. For an FDMA/OFDM-based system, each link's action would be a vector over all available channels/sub-carriers, and $\mathcal{H}$ would have the restriction that no two links can share the same channel/sub-carrier at a particular time. In MIMO, each link's action would be a vector over all transmitting antennas.

Modeling the Offered Load

In summary, we have a general model $\vec{r}(t) = g(\vec{p}(t), K(t))$ for a wireless data system. This is defined over a feasible set of actions $\mathcal{H}$, with a probability λ_k of being in channel state k.

We will further assume that the channel state is independent and identically distributed (i.i.d.) over time, meaning the λ_k do not change. Additionally, we will assume the set $\{g(\vec{p}), \vec{p} \in \mathcal{H}\}$ of achievable rates is bounded.

Now, recall that there are S flows in the system, where each flow s has a path through the network:



We defined H_s^l in (1) as the flow-link assignments. Further, let $X_s > 0$ be the **average rate** of flow s. Since the arrivals may be random, X_s is formally defined as

$$X_s = \mathbb{E}[A_s(t)], \tag{9}$$

where $A_s(t)$ is the traffic arrival for flow s at time t, which is in general a random process. We can further collect these average rates as a vector $\vec{X} = [X_1, ..., X_s]$. The **offered load** on link l in the system is then

$$\sum_s H_s^l X_s, \tag{10}$$

i.e., the sum of the rates of flows assigned to link l.

Note that for cellular systems, we would typically have each flow being associated with a particular link. In this case, the offered load on the link is simply the average rate of that flow.

The Capacity Problem

Returning back to our original problem statement, we would like to know, *what is the largest set of $\vec{X}$ that the network can support*? And further, *how to choose the actions of the links to achieve this capacity*?

We will show that the set of possible $\vec{X}$ must satisfy:

$$\Big[\sum_s H_s^l X_s\Big] \in \Lambda = \sum_k \lambda_k \cdot \mathsf{Conv\_hull}\{g(\vec{p}, k) \mid \vec{p} \in \mathcal{H}\}, \tag{11}$$

where $\mathsf{Conv\_hull}\{\mathcal{A}\}$ stands for the **convex hull** of the set of points $\mathcal{A}$, and Λ is the **capacity region**. The convex hull of $\mathcal{A}$ is the smallest **convex set** containing all the points in $\mathcal{A}$, which we will discuss further later.

Moreover, we will develop a scheduling algorithm that can compute the action $\vec{p}$ of the links for any given $\vec{X}$ within the set Λ, without requiring knowledge of the λ_k.

Capacity Region

Let's take a closer look at how the capacity region in (11) arises.

Intuitively, due to interference, typically not all links can be active at the same time. To support the end-to-end rate $\vec{X}$, then, it is preferable to use a set of actions in an alternating manner.

Suppose that for channel state k, the network can use M actions,

$$\vec{p}_k^{\,1}, \vec{p}_k^{\,2}, ..., \vec{p}_k^{\,M} \in \mathcal{H}, \tag{12}$$

each for

$$\alpha_k^1, \alpha_k^2, ..., \alpha_k^M \geq 0 \tag{13}$$

fraction of the time, where $\sum_{m=1}^M \alpha_k^m = 1$ for all k. In vector form, $\vec{\alpha}_k = [\alpha_k^1, ..., \alpha_k^M]$ for each k. The **long-term rate** that can be supported at each link l is then

$$\sum_k \lambda_k \Big(\sum_m \alpha_k^m g_l(\vec{p}_k^{\,m}, k)\Big). \tag{14}$$

Using these actions, the network can support the offered load $\vec{X}$ as long as the above quantity is larger than $\sum_s H_s^l X_s$ for each link l:

$$\Big[\sum_s H_s^l X_s\Big] \leq \sum_k \lambda_k \Big(\sum_m \alpha_k^m g(\vec{p}_k^{\,m}, k)\Big). \tag{15}$$

Conversely, to support an offered load $\vec{X}$, there must exist

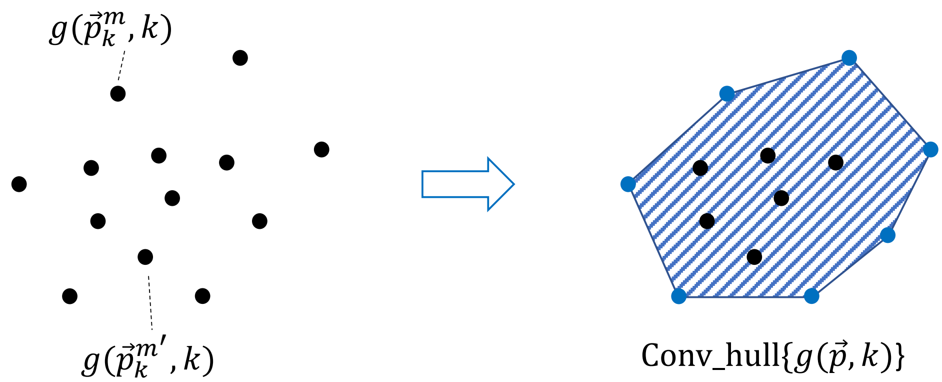
$$\vec{p}_k^1, \vec{p}_k^2, ..., \vec{p}_k^M \in \mathcal{H}, \sum_{m=1}^{M} \alpha_k^m = 1 \qquad \text{for each } k \tag{16}$$

such that (15) is true.

When we vary $\vec{p}$, $\vec{\alpha}$, the inner summation on the right hand side of (15) forms the convex hull of

$$\{g(\vec{p}, k),\ \vec{p} \in \mathcal{H}\}, \tag{17}$$

since it traces out all weighted combinations of $g(\vec{p}, k)$. Geometrically, the convex hull of a set of points is bounded by line segments connecting the outer-most points. In (17), the points are in L-dimensional space, where L is the number of links. In the case of $L = 2$ links, then, we can readily visualize the convex hull. In the following example, the shaded region on the right side is the convex hull:



Hence, the offered load $\vec{X}$ that can be supported by the network must belong to

$$\Lambda = \sum_k \lambda_k \cdot \mathsf{Conv\_hull}\{g(\vec{p}, k) \mid \vec{p} \in \mathcal{H}\}, \tag{18}$$

the weighted sum of convex hulls for $k = 1, ..., K$, which is in turn a convex set.

If we know $\vec{X}$, λ_k, and $g(\cdot, \cdot)$, we may be able to find the actions $\vec{p}_k^1, ..., \vec{p}_k^M$ and fractions $\alpha_k^1, ..., \alpha_k^M$ offline. However, prior knowledge of $\vec{X}$ and λ_k may not be available, or even if it is, it can be inaccurate. We therefore ask, *can we develop an adaptive scheduling scheme that does not require prior knowledge of $\vec{X}$ and λ_k, yet is able to support any $\vec{X} \in \Lambda$?*

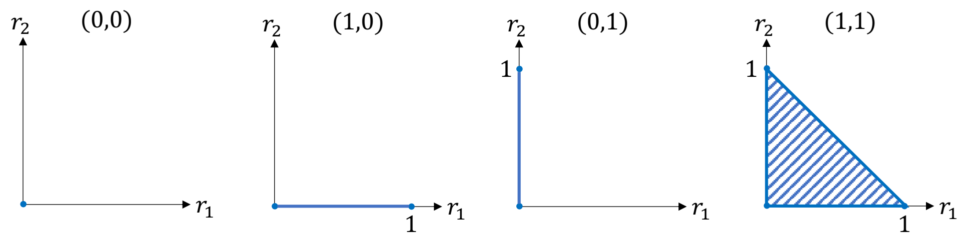
The answer is yes. We will present such a throughput-optimal scheme that is queue-length based.

Optimizing over Capacity Region

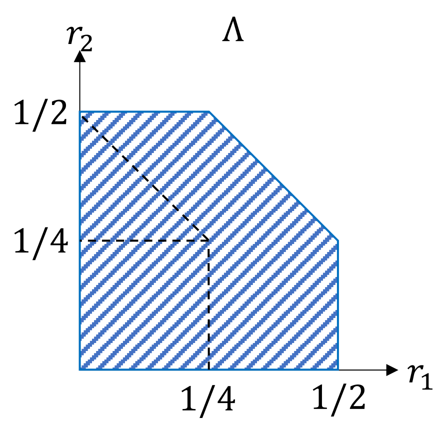
We will now, by way of example, illustrate the process of obtaining the capacity region, and then discuss optimizing a function over this region.

Suppose we have one base station and two users. The base station can only transmit to one user at a time. We also consider four channel states: $(0, 0)$, $(0, 1)$, $(1, 0)$, $(1, 1)$, where $(1, 0)$ is ON for user 1 and OFF for user 2. In this context, ON means that the link is available for communication.

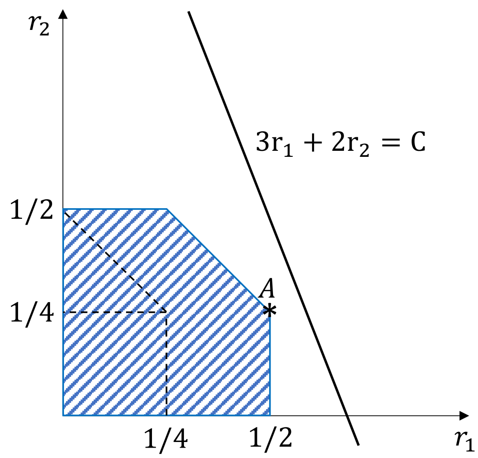
From the capacity region expression in (18), we start by obtaining the convex hull for each state. We will normalize the maximum rate of each user to $r = 1$. In each case, the convex hull is either a point, a line, or a triangle:



Now, suppose $\lambda_{00} = \lambda_{01} = \lambda_{10} = \lambda_{11} = 1/4$. The overall capacity region is obtained by multiplying each point in the convex hulls above by $1/4$ and adding them together. This gives the following for Λ:

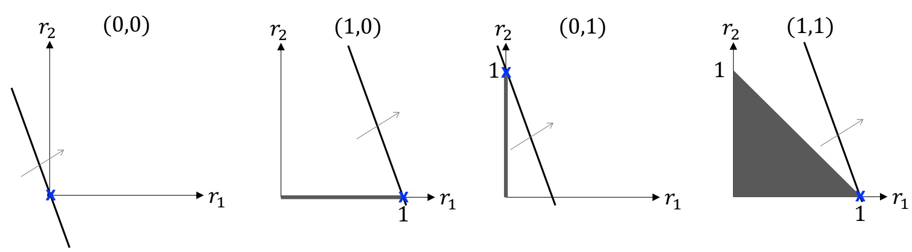


Now, suppose we want to find the point $(r_1, r_2) \in \Lambda$ that maximizes an objective function. Consider a weighted sum $3r_1 + 2r_2$. Visually, we can draw the line $3r_1 + 2r_2 = C$, and search for the point in the capacity region that gives us the maximum value of C, which corresponds to shifting the line to the right. That point is $A = (1/2, 1/4)$, as visualized below:



This example illustrates three important facts about optimizing weighted sum objective functions over the capacity region:

1. The point A can be seen as the average of the maximizing rate vector at each channel state:



   From left to right, we have $\frac{1}{4} \times (0,0) + \frac{1}{4} \times (1,0) + \frac{1}{4} \times (0,1) + \frac{1}{4} \times (1,1) = (1/2, 1/4)$.

   In general,

$$\begin{cases} \max\limits_{\vec{r}} & \sum_l w_l r_l \\ \text{subject to} & [r_l] \in \sum_k \lambda_k \cdot \mathsf{Conv\_hull}\{g(\vec{p}, k) \mid \vec{p} \in \mathcal{H}\} \end{cases} \tag{19}$$

$$= \sum_k \lambda_k \cdot \begin{cases} \max\limits_{\vec{r}} & \sum_l w_l r_l \\ \text{subject to} & [r_l] \in \textsf{Conv\_hull}\{g(\vec{p}, k) \mid \vec{p} \in \mathcal{H}\} \end{cases} \tag{20}$$

2. The maximizer of a weighted sum over a convex hull always occurs at the extreme points (i.e., those used to form the convex hull):

$$\begin{cases} \max\limits_{\vec{r}} & \sum_l w_l r_l \\ \text{subject to} & [r_l] \in \textsf{Conv\_hull}\{g(\vec{p}, k) \mid \vec{p} \in \mathcal{H}\} \end{cases} \tag{21}$$

$$= \quad \begin{cases} \max\limits_{\vec{r}} & \sum_l w_l r_l \\ \text{subject to} & [r_l] \in \{g(\vec{p}, k) \mid \vec{p} \in \mathcal{H}\} \end{cases} \tag{22}$$

3. For any point $\vec{r}^{\,0}$ in the capacity region, i.e., $\vec{r}^{\,0} \in \Lambda$,

$$\sum_l w_l r_l^0 \quad \leq \quad \begin{cases} \max\limits_{\vec{r}} & \sum_l w_l r_l \\ \text{subject to} & [r_l] \in \Lambda \end{cases} \tag{23}$$

These facts are also true for **linear optimization** problems more generally, which is the class of problems associated with optimizing a linear objective function over a set that can be classified as a **convex polytope**.