OFDMA Downlink Resource Allocation using Limited Cross-Layer Feedback

Prof. Phil Schniter

Joint work with
Mr. Rohit Aggarwal, Dr. Mohamad Assaad, Dr. C. Emre Koksal

September 20, 2010
Setup:

- Single-antenna downlink with $K$ mobile users
- OFDMA with $N$ subchannels
- Channels are Markov time-varying with impulse responses of length $L$
- ACK/NAK feedback on each transmitted data packet (i.e., ARQ)
The OFDMA Resource Allocation Problem:

- At each time $t$, we want to assign the “best” users (*multiuser diversity*) to their “best” subchannels (*frequency diversity*).
- We also want to optimize the powers and data-rates of assigned users.
- For this, we need channel state information (CSI) at the transmitter.
- Feedback of receiver CSI can be very costly! (Many users, subchannels.)
- Feedback of data-packet ACK/NAKs is free! (Provided by link layer.)

*Is it possible to do near-optimal resource allocation using only ACK/NAK feedback from previously transmitted data packets?*

*Can we get **enough** Tx-CSI from such limited cross-layer feedback?*
A Much Simpler Problem:

- Simplified scenario: $K = 1$ user, $N = 1$ subchannel, $L = 1$ channel tap.
  \[
  \Rightarrow \textit{Point-to-point communication over a time-varying channel.}
  \]
- The only parameter to optimize is the transmission rate $r_t$ (bits/packet).
  \[
  \Rightarrow \textit{Rate adaptation via ARQ feedback.}
  \]
- Though it may sound simple, this is still an interesting problem!
  - Notice that $r_t$ effects \textbf{short-term} throughput:
    \[
    r_t \text{ too high } \Rightarrow \text{ the packet will not be correctly decoded.}
    \]
    \[
    r_t \text{ too low } \Rightarrow \text{ the channel will be underutilized.}
    \]
  - But $r_t$ also effects \textbf{long-term} throughput:
    Recalling that more accurate CSI leads to higher throughput... 
    \[
    r_t \text{ very high } \Rightarrow \text{ likely NAK } \Rightarrow \text{ nothing learned about CSI.}
    \]
    \[
    r_t \text{ very low } \Rightarrow \text{ likely ACK } \Rightarrow \text{ nothing learned about CSI.}
    \]
    \[
    r_t \text{ such that } \Pr\{\text{ACK}\} = \frac{1}{2} \text{ is most informative about CSI.}
    \]
Rate Adaptation based on ARQ Feedback:

Goal: Maximize total expected goodput over \( T \) packets:

\[
[r_1^*, \ldots, r_T^*] = \operatorname{argmax}_{[r_1, \ldots, r_T]} \left\{ \sum_{t=1}^{T} G(\gamma_t, r_t) \right\}, \quad G(\gamma, r) = (1 - \epsilon(\gamma, r))r.
\]

- If our knowledge of the SNR \( \gamma_t \) was decoupled from the rates \( \{r_\tau\}_{\tau \neq t} \), then it would be relatively easy to calculate the optimal \( \{r_t^*\}_{t=1}^{T} \):

\[
r_t^* = \operatorname{argmax}_{r_t} E\{G(\gamma_t, r_t)\} = \operatorname{argmax}_{r_t} \int G(\gamma_t, r_t)p(\gamma_t)dt.
\]

- But really, our knowledge of SNR \( \gamma_t \) will depend on the ARQ feedbacks received up to time \( t \), which are strongly influenced by rates \( \{r_\tau\}_{\tau=1}^{t-1} \).

The current rate allocation affects not only immediate goodput, but also future SNR knowledge (and hence future goodput)!

\( \Rightarrow \) A classical tradeoff between exploitation and exploration.
Optimal Rate Adaptation:

Optimal rate adaptation, i.e.,

\[ r^*_t = \argmax_{r_{t+1}} E \left\{ G(r_{t+1}, \gamma_{t+1}) + \sum_{\tau=t+2}^{T} G(r^*_\tau, \gamma_\tau) \biggm| \hat{e}_t, r_t \right\} \]

can be recognized as a dynamic program.

Denoting the optimal expected future-goodput by

\[ \bar{G}^*_t \triangleq E \left\{ \sum_{\tau=t+1}^{T} G(r^*_\tau, \gamma_\tau) \biggm| \hat{e}_t, r_t \right\}, \]

we can write the Bellman equation as

\[ \bar{G}^*_t(\hat{e}_t, r_t) = \max_{r_{t+1}} \left\{ E\{ G(r_{t+1}, \gamma_{t+1}) \mid \hat{e}_t, r_t \} \right. \]

\[ \left. + E\{ \bar{G}^*_t([\hat{e}_t, \hat{e}_{t+1}], [r_t, r_{t+1}]) \mid \hat{e}_t, r_t \} \right\} \]

with the first expectation over \( \gamma_{t+1} \) and the second expectation over \( \hat{e}_{t+1} \).
Optimal Rate Adaptation (cont.):

The solution to this dynamic program is a partially observable Markov decision process (POMDP), which

- is intractable when the SNRs $\{\gamma_t\}$ are continuous random variables,
- can be solved numerically when the SNRs $\{\gamma_t\}$ are discrete, but at great cost: both complexity and memory increase exponentially with the horizon length $T$ and the number of states used for $\gamma_t$. ("PSPACE-complete")

$\Rightarrow$ Not practical to implement!
Greedy Adaptation:

- What if we maximize only the short-term reward?
- This corresponds to the greedy scheme

$$r_{t+1}^g \triangleq \arg\max_{r_{t+1}} E\{G(r_{t+1}, \gamma_{t+1}) \mid \hat{e}_t, r_t\},$$

which should be much easier to implement.

*How good is this greedy scheme (w.r.t the optimal POMDP)?*
The Causal Genie:

Consider the optimal rate under non-degraded causal error-rate feedback:

\[ r_{t+1}^{cg} \triangleq \arg\max_{r_{t+1}} \mathbb{E} \left\{ G(r_{t+1}, \gamma_{t+1}) + \sum_{\tau=t+2}^{T} G(r_{\tau}^{cg}, \gamma_{\tau}) \mid \epsilon_t, r_t \right\}. \]

For a given rate \( r_t \), the error \( \epsilon_t \) uniquely determines the SNR \( \gamma_t \), so that

\[ r_{t+1}^{cg} = \arg\max_{r_{t+1}} \mathbb{E} \left\{ G(r_{t+1}, \gamma_{t+1}) + \sum_{\tau=t+2}^{T} G(r_{\tau}^{cg}, \gamma_{\tau}) \mid \gamma_t \right\}. \]

Since future causal-genie rates \( \{r_{\tau}^{cg}\}_{\tau \geq t+2} \) will be chosen based on perfect SNR knowledge, they will not depend on \( r_{t+1} \). Thus,

\[ r_{t+1}^{cg} = \arg\max_{r_{t+1}} \mathbb{E} \left\{ G(r_{t+1}, \gamma_{t+1}) \mid \gamma_t \right\}. \]

**Optimal adaptation under non-degraded error-rate feedback is greedy!**
The Causal Genie is an Upper Bound on the POMDP:

Since \( E\{G(r_t^*, \gamma_t) \mid \hat{e}_{t-d}, r_{t-d}\} \)

\[
\leq \max_{r_t \in \mathcal{R}} E\{G(r_t, \gamma_t) \mid \hat{e}_{t-d}, r_{t-d}\}
\]

\[
\ldots \text{since } r_t^* \text{ is not necessarily short-term optimal}
\]

\[
= \max_{r_t \in \mathcal{R}} E \{ E\{G(r_t, \gamma_t) \mid \hat{e}_{t-d}, r_{t-d}, \epsilon_{t-d}\} \mid \hat{e}_{t-d}, r_{t-d}\}
\]

\[
\leq E \{ \max_{r_t \in \mathcal{R}} E\{G(r_t, \gamma_t) \mid \hat{e}_{t-d}, r_{t-d}, \epsilon_{t-d}\} \mid \hat{e}_{t-d}, r_{t-d}\}
\]

\[
\ldots \text{since } \max_{r_t} E\{f(r_t)\} \leq E\{\max_{r_t} f(r_t)\} \text{ for any } f(\cdot)
\]

\[
= E \{ \max_{r_t \in \mathcal{R}} E\{G(r_t, \gamma_t) \mid \gamma_{t-d}\} \mid \hat{e}_{t-d}, r_{t-d}\}
\]

\[
= E\{G(r_t^c, \gamma_t) \mid \hat{e}_{t-d}, r_{t-d}\},
\]

summing and averaging both sides gives

\[
E \left\{ \sum_{t=1}^{T} G(r_t^*, \gamma_t) \right\} \leq E \left\{ \sum_{t=1}^{T} G(r_t^c, \gamma_t) \right\}.
\]
Recap:

So far we’ve shown that, in the point-to-point problem,

1. Optimal rate adaptation is very difficult to implement/analyze.
2. There exists a suboptimal greedy scheme which is easy to implement.
3. The causal-genie scheme (i.e., the optimal scheme based on non-degraded feedback) is easy to implement, since it’s greedy.
4. The causal genie upper bounds the performance of the optimal scheme.

So, if we can show that the greedy scheme is close to the causal genie, then greedy must be close to optimal.
Numerical Examples – Setup:

- Modulation: Uncoded square-QAM, $p = 100$, packet error rate

$$\epsilon(r_t, \gamma_t) = 1 - \left(1 - 2 \left(1 - \frac{1}{\sqrt{2^{r_t/p}}}\right) Q\left(\sqrt{\frac{3\gamma_t}{2^{r_t/p} - 1}}\right)\right)^{2p}. $$

- ARQ feedback: $\hat{\epsilon}_t \in \{0, 1\}$

$$\Pr(\hat{\epsilon}_t = f | \epsilon_t) = \begin{cases} 
\epsilon_t & \text{for } f = 0 \quad ("NAK") \\
1 - \epsilon_t & \text{for } f = 1 \quad ("ACK")
\end{cases}$$

- SNR variation:
  - Rayleigh-fading channel gain (via Gauss-Markov process):

$$g_{t+1} = (1 - \alpha) g_t + \alpha w_t, \quad \{w_t\} \sim \text{i.i.d } \mathcal{CN}(0, 1)$$

- SNR:

$$\gamma_t = C |g_t|^2.$$  

- Parameters $(\alpha, C')$ control the mean and coherence time of $\gamma_t$. For a given $\alpha$, we assume $C'$ is chosen so that $\mathbb{E}\{\gamma_t\} = 25$ dB.
Steady-state goodput versus fading-rate parameter $\alpha$:

Greedy rate adaptation extracts most of the goodput gain available from the use of causal ARQ feedback!

Best fixed-rate: $r_{fr}^{t+1} = \arg\max_r G(r_{t+1}, \gamma_{t+1}) \int G(r_{t+1}, \gamma_{t+1}) p(\gamma_{t+1}) d\gamma_{t+1}$

Greedy: $r_{g}^{t+1} = \arg\max_r \int G(r_{t+1}, \gamma_{t+1}) p(\gamma_{t+1}|\hat{\epsilon}_t, r_t) d\gamma_{t+1}$

Causal Genie: $r_{cg}^{t+1} = \arg\max_r \int G(r_{t+1}, \gamma_{t+1}) p(\gamma_{t+1}|\epsilon_t, r_t) d\gamma_{t+1}$

Non-causal Genie: $r_{ng}^{t+1} = \arg\max_r G(r_{t+1}, \gamma_{t+1})$
Back to the OFDMA Problem:

- Single-antenna downlink with \( K \) mobile users
- OFDMA with \( N \) subchannels
- Channels are Markov time-varying with impulse responses of length \( L \)
- ACK/NAK feedback on each transmitted data packet (i.e., ARQ)
Formal Objective:

At each time $t$ and subchannel $n$, choose each user $k$’s next...

- rate $r_{n,k,t+1} \in \mathcal{R}$, for some $M$-ary rate alphabet $\mathcal{R}$,

- power $p_{n,k,t+1} \geq 0$,

based on ARQ feedbacks $F_{1:t}$, to maximize the total future utility

$$
\bar{G}_{t+1:T} = \sum_{\tau=t+1}^{T} \sum_{k=1}^{K} \sum_{n=1}^{N} \mathbb{E} \left\{ \sum_{n=1}^{N} U\left( \left(1 - \epsilon(r_{n,k,\tau}, \gamma_{n,k,\tau}, p_{n,k,\tau})\right) r_{n,k,\tau} \right) \right\} \left| F_{1:t} \right|
$$

subject to the power constraint $\sum_{n,k} p_{n,k,\tau} \leq P_{\text{con}}, \ \forall \tau$,

and subject to $\leq 1$ (user,rate) per (time,subchannel).

Here, $\epsilon(r, \gamma, p)$ is packet error rate, $U(\cdot)$ is a concave utility function, and $F_{1:t} \in \{0, 1, \emptyset\}^{tNK}$ collects all past ARQ feedbacks.
Greedy Resource Allocation:

Using the indicator $I_{n,m,k,t} \in \{0, 1\}$ to denote time-$t$ assignment of subchannel $n$ to user $k$ at MCS index $m$, the time-$t$ greedy allocation is

$$\max \sum_k \mathbb{E} \left\{ \sum_{n,m} U \left( I_{n,k,m,t+1} (1 - a_m e^{-b_m p_{n,k,m,t+1}}) \gamma_{n,k,t+1} r_m \right) \right\} \bigg| F_{1:t}$$

subject to

$$\sum_{n,k,m} I_{n,k,m,t} p_{n,k,m,t} \leq P_{\text{con}}, \quad \forall t,$$

and

$$\sum_{k,m} I_{n,k,m,t} \leq 1, \quad \forall n, \quad \forall t,$$

where

- $\gamma_{n,k,t}$ is SNR of user $k$ on subchannel $n$ at time $t$,
- $(a_m, b_m, r_m)$ determines data rate and error rate for MCS index $m$,
- $F_{1:t}$ collects all ARQ feedbacks collected from times 1 to $t$.

A mixed-integer programming problem!
Greedy Resource Allocation:

Exact solution:


Proposed approximate solution:

- Relax problem by allowing each (subcarrier,time) slot to be shared among multiple (user,rate) pairs.
  - Can reformulate as a convex optimization problem.
  - Can solve dual problem using an iterative algorithm with exponential convergence rate at $O(NKM)$ per iteration.

- Byproducts of above algorithm yield a near-exact solution to the original mixed-integer problem.

Approximate Greedy Algorithm — Sketch:

Say that we relax the binary indicators to $\tilde{I}_{n,m,k,t} \in [0, 1]$.

Then the KKT conditions become (suppressing the time-$t$ notation):

\begin{align*}
\forall n, k, m, \quad \mu &= a_mb_mr_m \mathbb{E}\{\gamma_{n,k} e^{-b_m p_{n,k,m} \gamma_{n,k}} \mid F\} \\
\forall n, k, m, \quad \lambda_n &= r_m \mathbb{E}\{1 - a_m e^{-b_m p_{n,k,m} \gamma_{n,k}} \mid F\} - \mu p_{n,k,m}
\end{align*}

(1) (2)

where $\{\lambda_n\}_{n=1}^N$ and $\mu$ are Lagrange multipliers. A practical alg is then:

1. Initialize search region $[\mu, \bar{\mu}]$ with midpoint $\mu = (\mu + \bar{\mu})/2$.

2. For each subchannel $n$,
   - For each $(k, m)$ . . .
     - calculate $p_{n,k,m}$ by solving (1) and forcing a non-negative result.
     - plug $p_{n,k,m}$ into (2) and calculate the corresponding $\lambda_n(k, m)$.
   - Find $(k^*, m^*) = \text{arg max}_{(k,m)} \lambda_n(k, m)$.
   - Set $I_{n,k^*,m^*} = 1$ and $I_{n,k,m} |_{(k,m) \neq (k^*,m^*)} = 0$.

3. If $\sum_n p_{n,k^*,m^*} > P_{\text{con}}$, set $[\mu, \bar{\mu}] \leftarrow [\mu, \bar{\mu}]$, else set $[\mu, \bar{\mu}] \leftarrow [\mu, \mu]$. Goto 2.
### Approximate Greedy Algorithm — Example Performance:

<table>
<thead>
<tr>
<th>$N$</th>
<th>$K$</th>
<th>$M$</th>
<th>greedy goodput (brute force)</th>
<th>approximation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>9</td>
<td>5.9884</td>
<td>5.988</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>9</td>
<td>6.3501</td>
<td>6.3499</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>9</td>
<td>10.3251</td>
<td>10.3249</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>9</td>
<td>10.9778</td>
<td>10.9774</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>9</td>
<td>14.0573</td>
<td>14.0571</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>9</td>
<td>14.9653</td>
<td>14.9651</td>
</tr>
</tbody>
</table>

The practical approximation typically yields 99.99% of the goodput attained by the exact greedy scheme!

We have also derived performance bounds on our approximation.
Tracking the SNR distribution:

The greedy allocator tracks the SNR by updating the SNR distributions

\[ \{ p(\gamma_{n,k,t+1} \mid F_{1:t}), \ \forall \text{ users } k \text{ and subchannels } n \} \]

The SNR evolves as follows:

- Markov evolution of time-domain channel taps:
  \[ h_{l,k,t+1} = (1 - \alpha) h_{l,k,t} + \alpha w_{l,k,t}, \quad w_{l,k,t} \sim \mathcal{CN}(0, 1), \]

- subchannel gains as a function of time-domain channel taps:
  \[ H_{n,k,t} = \sum_{l=0}^{L-1} h_{l,k,t} e^{-j \frac{2\pi}{N} nk}, \]

- subchannel SNRs as a function of subchannel gains:
  \[ \gamma_{n,k,t} = C |H_{n,k,t}|^2. \]
Tracking the SNR distribution (cont.):

SNR tracking can be done as follows:

\[ p(\gamma_{n,k,t+1} \mid F_{1:t}) = \int_{h_{k,t+1}} p(\gamma_{n,k,t+1} \mid h_{k,t+1}) p(h_{k,t+1} \mid F_{1:t}) \]  
(approx of) Dirac delta

\[ p(h_{k,t+1} \mid F_{1:t}) = \int_{h_{k,t}} p(h_{k,t+1} \mid h_{k,t}) p(h_{k,t} \mid F_{1:t}) \]  
Markov prediction

\[ p(h_{k,t} \mid F_{1:t}) = \frac{p(f_{k,t} \mid h_{k,t}) p(h_{k,t} \mid F_{1:t-1})}{\int_{h'_{k,t}} p(f_{k,t} \mid h'_{k,t}) p(h'_{k,t} \mid F_{1:t-1})} \]  
(Bayes rule)

\[ p(f_{k,t} \mid h_{k,t}) = \prod_{n=1}^{N} p(f_{n,k,t} \mid \gamma_{n,k,t}(h_{k,t})) \]

\[ p(f_{n,k,t} = f \mid \gamma_{n,k,t}) = \begin{cases} \sum_{m} I_{n,k,m,t} a_{m} e^{-b_{m} p_{n,k,m,t} \gamma_{n,k,t}} & f = 0 \\ \sum_{m} I_{n,k,m,t} (1 - a_{m} e^{-b_{m} p_{n,k,m,t} \gamma_{n,k,t}}) & f = 1 \\ 1 - \sum_{m} I_{n,k,m,t} & f = \emptyset \end{cases} \]
Tracking the SNR distribution (cont.):

Thus, for each user $k$,

1. measure feedbacks $f_{k,t}$ across all subchannels,

2. compute $p(f_{n,k} \mid \gamma_{n,k,t}(h_{k,t}))$ on $h$-lattice using error-rate rules (6)-(7),

3. compute $p(h_{k,t} \mid F_{1:t})$ on $h$-lattice by updating previous posterior via (5),

4. compute $p(h_{k,t+1} \mid F_{1:t})$ on $h$-lattice via Markov-prediction step (4),

5. compute $p(\gamma_{k,t+1} \mid F_{1:t})$ on $\gamma$-lattice via $h$-to-$\gamma$ conversion step (3).

This costs $O(KNQ_H^L + KLQ_H^{L+1} + KNQ_\gamma Q_H^L)$, where

$Q_H = \text{number of grid points used per dimension of } h$-lattice,

$Q_\gamma = \text{number of grid points used per dimension of } \gamma$-lattice.
Numerical Experiments:

Setup:

- $K = 2$ users
- $N = 2$ subchannels
- $L = 2$ time-domain channel taps
- $E\{\gamma_{n,k,t}\} = 25dB$ mean subchannel SNR
- $\rho = 0.33$ subchannel correlation

Plots show

- goodput versus fading rate $\alpha$
- goodput versus time $t$
- power/rate/user versus time $t$
Steady-state goodput versus $\alpha$:

- **Global Genie**
- **Greedy Algorithm**
- **Round Robin (no feedback)**
**Goodput for $\alpha = 0.001$:**

2 users, 2 subcarrier, $\alpha = 1e^{-3}$, 200 packets

- genie-CSI avg = 11.0324
- tracked-CSI avg = 10.574
- prior-CSI avg = 7.3398
Allocations for $\alpha = 0.001$:
Summary:

- Goal: From only ARQ feedback, optimize OFDMA users, powers, and rates to maximize finite-horizon expected goodput under an instantaneous total-power constraint.
- The optimal resource allocator is a POMDP, which is computationally impractical.
- We settle for greedy resource allocation, found to be near-optimal for practical fading rates.
- Greedy allocation is a mixed-integer programming problem, but we can solve it almost exactly with $O(NKM)$ complexity.
- To maintain CSI, we track the SNR distribution (conditioned on past ACK/NAK feedback) of each user at each subcarrier.
- Preliminary experiments for 2 users and 2 subchannels indicates that our practical algorithm performs relatively close to a genie-aided upper bound on the optimal POMDP.
Thanks for listening!