Script for Mike Neely's talk "Distributed stochastic optimization via correlated scheduling."

Thursday, 3:30pm, first talk of "Scheduling and buffer management 4 session", Pier 4.

For convenience, I placed numbers on each slide to enumerate them.

If there is a crunch for time, you can skip to slide 22 with the simulation results.

SLIDE 1:

The title is "Distributed stochastic optimization via correlated scheduling." Mike Neely regrets not being able to be here. He had unexpected passport difficulties. Xiaojun Lin will present the talk in his place.

SLIDE 2: "Distributed sensor reports"

This paper looks at a problem of many distributed sensor devices that report observations to a fusion center.

The system operates over time slots t = {0, 1, 2, …}.
"Events" may or may not occur every slot.
Sensors may or may not observe the events.
Sensors use power to report their observations to a fusion center.

This slide illustrates a simple 2-sensor example:
w_i(t) = observation variable for sensor i on slot t.
w_i(t) = 0 if sensor i did not observe an event.
w_i(t) = 1 if sensor I did observe an event.

p_i(t) = reporting power variable on slot t for sensor i:
p_i(t)=0 means sensor i does not report.
p_i(t)=1 means sensor i reports and uses 1 unit of power.

SLIDE 3: "Distributed sensor reports"

The fusion center gets utility u(t).
u(t) is a general, possibly non-separable, function of the reports of each sensor.

An example function is shown here. In this example, the fusion center trusts sensor 1 more than sensor 2:
U(t) = 1 if sensor 1 observes an event and reports (no additional increase if sensor 2 redundantly reports).
U(t) = ½ if sensor 2 observes an event and reports, without sensor 1 reporting.
U(t)=0 if nobody reports anything.

Main idea: REDUNDANT REPORTS DO NOT BRING ADDED UTILITY.

The reporting decisions occur every slot. The goal is to design a distributed reporting algorithm to maximize time average utility subject to time average power constraints at each sensor.

SLIDE 4: "Main ideas for this example"
<<read slide>>

SLIDE 5: "Assumed structure"
<<Read slide>>

SLIDE 6: "Example plans"
<<Read example plan>>

SLIDE 7: "Common source of randomness"
Here is another type of "Plan" that uses an important concept: "A common source of randomness."
Consider 2 people. Common source of randomness is Boston Globe Newspaper:
-If first letter "T" then use plan 1.
-If first letter is "S" then use plan 2.

COMMON RANDOMNESS IS CRUCIAL TO ACHIEVE OPTIMALITY
FOR THE DISTRIBUTED REPORTING PROBLEM.

SLIDE 8: "Specific example"
Here is a specific 2-sensor example with specific numbers.

Specific probabilities for the observations w_i(t) are given here.
The power constraint at each sensor is 1/3.

Approach 1: Independent reporting.
-If w_1(t) =1, sensor 1 reports with probability theta_1
-If w_2(t) = 1, sensor 2 reports with probability theta_2

Optimizing theta_1, theta_2 over this class of distributed strategies gives u_bar = 0.44444.

SLIDE 9: "Approach 2: Correlated reporting"
To introduce correlated reporting, let us first define 3 "pure strategies"
<<read the slide>>

SLIDE 10: "Approach 2: Correlated reporting"
<<Read slide>>

SLIDE 11: "Summary of approaches"
Summary for this simple example:
1) independent reporting gives utility = 0.44444.
2) correlated reporting gives utility = 0.47917
3) It can be shown that the optimal centralized algorithm gives utility = 0.5.

SLIDE 12: "Summary of approaches"
It can be shown that this correlated reporting algorithm is optimal over all possible distributed strategies. Thus, there is a fundamental performance gap between the best distributed algorithm and the best Centralized algorithm.

SLIDE 13: "General distributed optimization"
That was a simple 2-sensor example with binary observations and binary reporting decisions. The general problem is this:

Maximize average utility.
N sensors.
(w_1(t), …, w_N(t)) = observation vector for slot t
(iid over slots, possibly correlated entries on each slot).
pi(w) = known probabilities for these observation vectors.
(a_1(t), …, a_N(t)) = action vector for slot t.
Utility U(t) and power P_k(t) are general functions of the observation vector w(t) and the action vector a(t).

SLIDE 14: "Pure strategies"
<<Read slide quickly>>

SLIDE 15: "Optimality theorem"
<<Read theorem on slide quickly>>

SLIDE 16: "LP and complexity reduction"
<<Read slide, note that the complexity reduction technique is in the paper, skipped here for brevity.>>

SLIDE 17: "Discussion of Theorem 1"
<<Read slide quickly>>

SLIDE 18: "ONline dynamic approach"
<<Read slide quickly>>

SLIDE 19: "Lyapunov optimization approach"
<<Read slide quickly >>

SLIDE 20: "Lyapunov optimization approach"
<<Read slide quickly>>

SLIDE 21: "Separable problems"
<<Read slide quickly>>

SLIDE 22: "Simulation (non-separable problem)"
<<REad slide>>
In this example, observations take one of 10 values. Actions are still binary.

SLIDE 23: "Utility versus V parameter"
<<Graph shows that utility converges to optimal as V gets larger"

SLIDE 24: "Average power versus time"
<<Graph shows that average power converges to its constraint slower with large V>>

SLIDE 25: "Adaptation to non-ergodic changes"
The timeline was chopped into 3 phases. The first and third phases use one probability distribution. The middle phase uses another. Nobody tells the algorithm the probability distributions or the time of phase changes.

The algorithm adapts well. Interestingly, average power has some disturbances around the phase changes.
Average utility quickly adapts to the utility associated with the new probabilities.

SLIDE 26: "Adaptation to non-ergodic changes"
<<This shows what the horizontal dashed lines are>>

SLIDE 27: CONCLUSIONS
<<Read slide>>