

Lecture 6: Simultaneous Protocols

Lecturer: *Matt Weinberg*Scribe: *Gavriel Hirsch*

These notes may contain errors, and are not intended for use as a primary source. I have included references at the end of the notes for those who wish detail surrounding skipped steps.

1 Logistics

First homework will be posted soon.

2 Simultaneous Communication

This is the last class where we'll see algorithms that aren't concerned with incentives. Today specifically we'll see a *simultaneous* algorithm for two players that does as well as Feige's algorithm from last week (3/4). We'll wait until a later lecture to see why it's interesting to study simultaneous algorithms. To be clear, a simultaneous protocol proceeds as follows:

DEFINITION 1 *A simultaneous protocol between Alice and Bob is where Alice and Bob each send a single message, simultaneously. Then, a referee sees both messages and must produce a solution.*

For interactive protocols (which don't care about the number of rounds of interaction), it's not particularly interesting to distinguish between search (find an allocation maximizing welfare) and decision (does there exist an allocation with welfare at least C ?). The main reason for this is that a simple black-box reduction from decision to search requires only one extra round of communication (solve search, then ask Alice and Bob to evaluate the solution). Note that there's not a simple reduction in the other direction, but all known protocols for decision proceed by first solving search, then evaluating. To be extra clear:

DEFINITION 2 (SEARCH PROBLEM) *The referee outputs a (possibly randomly) set X . A protocol is an α -approximation if $v^A(X) + v^B(\bar{X}) \geq \alpha \max_Y \{v^A(Y) + v^B(\bar{Y})\}$.*

DEFINITION 3 (DECISION PROBLEM) *The referee is also given a challenge, C . A protocol is an α -approximation if the referee says "yes" whenever there exists an X with $v^A(X) + v^B(\bar{X}) \geq C$, and "no" whenever there does not exist an X with $v^A(X) + v^B(\bar{X}) \geq \alpha C$ (it is allowed to behave arbitrarily in between).*

This class we'll see good algorithms for the search problem, and next class we'll see lower bounds on the decision problem. For this class, we'll focus on a restricted class of XOS functions called Binary XOS.

DEFINITION 4 (BINARY ADDITIVE) *A binary additive function is an additive function where the value for each item is 0 or 1.*

DEFINITION 5 (BINARY XOS) *A binary XOS function is the maximum over a set of binary additive functions.*

We'll begin with the following lemma, which certainly exploits the binary nature of binary XOS.

LEMMA 1

Let v^A be binary additive, and v^B be binary XOS. Then the welfare maximizing allocation between v^A and v^B is achieved by giving to Alice all items he values at 1.

PROOF: Let S be the set of items that Alice values at 1 and (T, \bar{T}) be a potential allocation where Alice gets T . We'll modify T into S without decreasing the welfare. First, take all the items in $T \setminus S$ and give them to Bob. This doesn't decrease Alice's welfare but might increase Bob's. Now, take all items in $S \setminus T$ and give them to Alice. This increases Alice's welfare by exactly $|S \setminus T|$ and decreases Bob's by at most $v^B(S \setminus T)$ (by subadditivity), which is at most $|S \setminus T|$. Explicitly, using subadditivity we have

$$v^B(\bar{T} \cup (T \setminus S)) = v^B((\bar{T} \setminus \bar{S}) \cup (\bar{T} \setminus S) \cup (T \setminus S)) \leq v^B(\bar{T} \setminus \bar{S}) + v^B((\bar{T} \setminus S) \cup (T \setminus S))$$

so Bob's welfare is decreased by

$$v^B(\bar{T} \cup (T \setminus S)) - v^B((\bar{T} \setminus S) \cup (T \setminus S)) \leq v^B(\bar{T} \setminus \bar{S}) \leq |S \setminus T|$$

Note that a stronger lemma statement is possible: if we happen to know that $v^A(T) \geq v^B(T)$ for all $T \subseteq S$ and $v^A(\bar{S}) = 0$, then the welfare maximizing allocation is achieved by giving Alice S . \square

We'll use the lemma to prove the following approximation guarantee: it's actually possible to do something non-trivial with polynomial simultaneous communication.

PROPOSITION 2

The following protocol guarantees a 2/3 approximation for the search problem.

Let $a(\cdot)$ denote the binary additive clause of Alice's maximizing $a(M)$, and let $a_1(\cdot), a_2(\cdot)$ be the two clauses maximizing $SW(a_1, a_2)$ (interpretation one: think of clauses as sets, maximize the union. Interpretation two: maximize the number of items for which either a_1 or a_2 values it at 1). Randomly pick one of $\{a, a_1, a_2\}$ and give all of the items it values at one to Alice, then give the rest to Bob.

PROOF: By Lemma 1, the welfare achieved by the protocol is:

$$\frac{1}{3} (SW(a, v^B) + SW(a_1, v^B) + SW(a_2, v^B))$$

Because $SW(x, y)$ maximizes welfare among all partitions, it is at least as big as any particular partition. Let (T, \bar{T}) denote the welfare optimal allocation between v^A and v^B , and (S, \bar{S}) denote the welfare optimal allocation between a_1 and a_2 :

$$\geq \frac{1}{3} (a(T) + v^B(\bar{T}) + a_1(S) + v^B(\bar{S}) + a_2(\bar{S}) + v^B(S))$$

a_1, a_2 maximize welfare among all partitions among all pairs of Alice's binary additive functions (let a^* denote the additive clause for which $a^*(T) = v^A(T)$):

$$\geq \frac{1}{3} (a(T) + v^B(\bar{T}) + a(\bar{T}) + a^*(T) + v^B(\bar{S}) + v^B(S))$$

Also, v^B is subadditive and a is additive:

$$\begin{aligned} &\geq \frac{1}{3} (a(T) + v^B(\bar{T}) + a(\bar{T}) + a^*(T) + v^B(M)) \\ &\geq \frac{1}{3} (a(M) + 2v^B(\bar{T}) + a^*(T)) \end{aligned}$$

Now, a was chosen to maximize $a^*(M)$, so:

$$\begin{aligned} &\geq \frac{1}{3} (a^*(M) + a^*(T) + 2v^B(\bar{T})) \\ &\geq \frac{1}{3} (2a^*(T) + 2v^B(\bar{T})) \\ &= \frac{2}{3} (v^A(T) + v^B(\bar{T})) \end{aligned}$$

□

Note that this protocol is somewhat heavily tied to the search problem: we were crucially able to claim that we knew a welfare-maximizing allocation between a binary additive function and a binary XOS one without knowing how good that allocation is. A similar protocol guarantees a $3/5$ approximation for decision.

PROPOSITION 3

The following protocol guarantees a $3/5$ approximation for decision.

Alice sends the same information as before (a, a_1, a_2) . Bob sends the same type of information as Alice (b, b_1, b_2) . Find the additive functions $a' \in \{a, a_1, a_2\}, b' \in \{b, b_1, b_2\}$ reported by Alice and Bob, respective, maximizing $SW(a', b')$. If the welfare exceeds $3C/5$, say yes. Otherwise, say no.

PROOF: Assume that it's possible to get welfare at least C , and let $a^*(S) + b^*(\bar{S}) \geq C$. Let (A, \bar{A}) denote the optimal allocation between a_1 and a_2 , and (B, \bar{B}) for b_1, b_2 . Then the welfare known to the protocol above is at least:

$$\begin{aligned} &\frac{1}{5} (SW(a, b_1) + SW(a, b_2) + SW(a, b) + SW(a_1, b) + SW(a_2, b)) \\ &\geq \frac{1}{5} (a(\bar{B}) + b_1(B) + a(B) + b_2(\bar{B}) + SW(a, b) + a_1(A) + b(\bar{A}) + a_2(\bar{A}) + b(A)) \\ &\geq \frac{1}{5} (a(M) + [b_1(B) + b_2(\bar{B})] + SW(a, b) + [a_1(A) + a_2(\bar{A})] + b(M)) \\ &\geq \frac{1}{5} (a(M) + b^*(\bar{S}) + b(S) + a(S) + b(\bar{S}) + a^*(S) + a(\bar{S}) + b(M)) \end{aligned}$$

$$\begin{aligned}
&\geq \frac{1}{5} (a(M) + b^*(\bar{S}) + b(M) + a^*(S) + a(M) + b(M)) \\
&\geq \frac{1}{5} (3b^*(\bar{S}) + 3a^*(S))
\end{aligned}$$

□

Next we'll show that it's actually possible to get the tight 3/4-approximation for search with a simultaneous protocol.

DEFINITION 6 (SUMMARIES OF BINARY XOS VALUATIONS) *For a binary XOS valuation v , define its (k, α) -summary (b_1, \dots, b_k) as $\arg \max_{b_1, \dots, b_k \in \{a_1, \dots, a_t\}} \sum_{i=1}^m (x_i - \alpha \cdot x_i^2)$, where a_1, \dots, a_t are the clauses of v and $x_i = \frac{1}{k} \sum_{j=1}^k b_j(\{i\})$.*

Consider the special case when all additive functions value exactly $m/2$ items at 1. Then this is asking for the k clauses that minimize $\sum_i x_i^2$. As k goes to ∞ , we're looking for a distribution over clauses that minimize the expected number of items in the intersection of two random draws from that distribution. We'll now proceed under the assumption that all clauses value exactly $m/2$ items at 1 to keep the math simpler, but this captures the main ideas.

LEMMA 4

Let $k \rightarrow \infty$ and fix some $A \subseteq M$. If there exists a^ , some additive function of Alice's with value 1 for exactly the items in A , then $\sum_{i \in A} x_i \geq \sum_{i=1}^m x_i^2$.*

PROOF: Let Alice's summary induce \vec{x} . We could make a new distribution by replacing an ε fraction of the current distribution uniformly with a^* . This would yield a new \vec{x}' satisfying $x'_i = (1 - \varepsilon)x_i$ for all $i \notin A$ and $x'_i = (1 - \varepsilon)x_i + \varepsilon$ for all $i \in A$. This new distribution is not equal to Alice's summary, so it must not decrease $\sum_i x_i^2$. Therefore we have:

$$\begin{aligned}
&\sum_i (x'_i)^2 - x_i^2 \geq 0. \\
\implies &\sum_{i \notin A} ((1 - \varepsilon)^2 - 1)x_i^2 + \sum_{i \in A} ((1 - \varepsilon)^2 - 1)x_i^2 + 2\varepsilon(1 - \varepsilon)x_i + \varepsilon^2 \geq 0. \\
\implies &\sum_i \varepsilon^2 x_i^2 + \sum_{i \in A} 2\varepsilon(1 - \varepsilon)x_i + \varepsilon^2 \geq \sum_i 2\varepsilon x_i^2.
\end{aligned}$$

Dividing by ε and taking a limit as $\varepsilon \rightarrow 0$ yields:

$$\sum_{i \in A} 2x_i \geq \sum_i 2x_i^2$$

as desired. □

Now, we show how to make use of this to get a 3/4 approximation in the case defined:

THEOREM 1 ([?])

Let $k \rightarrow \infty$. Then the protocol defined below gets a 3/4 approximation for search when valuations are Binary XOS.

Alice randomly draws $a(\cdot)$ from her $(k, 1/2)$ -summary. Award Alice all items which $a(\cdot)$ values at 1, and give the rest to Bob.

PROOF: We'll provide the proof for the case that all clauses of Alice and Bob have exactly $m/2$ items they value at 1, and Alice has an additive function which values exactly the items in S , and Bob has one which values exactly the items in \bar{S} . So the target is to get welfare at least $3m/4$.

In this case, observe that Bob's welfare is at least the number of items in \bar{S} that he gets from the protocol. Moreover, observe that he gets such an item whenever it is not valued by Alice's randomly drawn additive function. Since Alice's function values item i with probability x_i , Bob's value is at least $\sum_{i \notin S} 1 - x_i$.

Alice's value is easier to compute: she gets $m/2$ always. So we wish to lower bound:

$$\begin{aligned}
& m/2 + \sum_{i \notin S} 1 - x_i \\
&= m - \sum_{i \notin S} x_i \\
&\geq m - \sum_i x_i + \sum_{i \in S} x_i \\
&\geq m/2 + \sum_i x_i^2 \\
&\geq 3m/4
\end{aligned}$$

The second-to-last line follows by the lemma, and the fact that Alice has an additive function which values all items in S at 1 by hypothesis. The final line follows by observing that $\sum_i x_i = m/2$, so the sum of squares is minimized when all $x_i = 1/2$. \square

Finally, we'll show how to use similar techniques to get a $23/32$ approximation for the decision problem. We'll stay restricted to the same case.

THEOREM 2 ([?])

Let $k \rightarrow \infty$. Then the protocol defined below gets a $23/32 = 3/4 - 1/32$ approximation for search when valuations are Binary XOS.

Let $\ell \rightarrow \infty$. Alice randomly draws $a(\cdot)$ from her $(k, 1/3)$ -summary, and Bob randomly draws $b(\cdot)$ from his $(k, 1/3)$ -summary. Count the number of items which at least one of them values at 1. Repeat this process ℓ times. If the mean of these counts is at least $23m/32$, output yes. Otherwise, output no.

PROOF: Clearly, whenever the correct answer is no, the algorithm will always output no. Now consider that the algorithm is supposed to output yes (that is, there exists an S such that Alice values S at $m/2$ and Bob values \bar{S} at $m/2$). Then let \vec{x}, \vec{y} denote the vectors induced by the summaries, for Alice and Bob respectively.

Observe that our protocol will produce, in expectation, $m - \vec{x} \cdot \vec{y}$ ones per draw. This is because Alice and Bob each contribute $m/2$ ones, but some of them will be for the same

item (and not actually contribute. Each i is in the intersection with probability $x_i y_i$, so the total expected intersection is $\vec{x} \cdot \vec{y}$. So now we wish to understand how bad $\vec{x} \cdot \vec{y}$ can possibly be, subject to constraints on \vec{x} and \vec{y} both describing valid summaries for Alice and Bob (given that this good partition S exists).

By the lemma we know that we must have $\sum_i x_i^2 \leq \sum_{i \in S} x_i$ and $\sum_i y_i^2 \leq \sum_{i \notin S} y_i$, and we have ourselves the following non-convex optimization problem:

- Maximize $\sum_i x_i y_i$.
- Such that: $\sum_i x_i^2 \leq \sum_{i \in S} x_i$.
- Such that: $\sum_i y_i^2 \leq \sum_{i \notin S} y_i$.
- Such that $\sum_i x_i = m/2$.
- Such that $\sum_i y_i = m/2$.

Observe that we can rewrite $2x_i y_i = x_i^2 + y_i^2 - (x_i - y_i)^2$. We'll now use Lagrangian multipliers to upper bound how big our solution can possibly be. Put a Lagrangian multiplier of $1/2$ on each of the first two constraints. Recall that we want to add something that is guaranteed (by feasibility conditions) to be ≥ 0 to the objective, as this can only increase it. So our Lagrangian relaxation is now:

- Maximize $\sum_{i \in S} x_i/2 + \sum_{i \notin S} y_i/2 - \sum_i (x_i - y_i)^2/2$.
- Such that $\sum_i x_i = m/2$.
- Such that $\sum_i y_i = m/2$.

Next, put a multiplier of $1/4$ on each of the remaining constraints. This makes our new problem simply to maximize:

$$\sum_{i \in S} x_i/2 + \sum_{i \notin S} y_i/2 - \left(\sum_i (x_i - y_i)^2/2 + x_i/4 + y_i/4 \right) + m/4.$$

Let's now take partial derivatives to see what potential optima are. The partial with respect to x_i , $i \in S$ is $1/4 - (x_i - y_i)$. For $i \notin S$, it's $-1/4 - (x_i - y_i)$. Similarly, the partial with respect to y_i , $i \in S$ is $-1/4 - (y_i - x_i)$, and for $i \notin S$ is $1/4 - (y_i - x_i)$.

Therefore, the only potential local optima have $x_i = y_i - 1/4$ for $i \notin S$, and $x_i = y_i + 1/4$ for $i \in S$. After making this substitution in, we get:

$$\begin{aligned} & \left(\sum_{i \in S} x_i/4 + y_i/4 + 1/16 + \sum_{i \notin S} x_i/4 + y_i/4 + 1/16 \right) - \left(\sum_i 1/32 + x_i/4 + y_i/4 \right) + m/4 \\ &= (m/4 + m/16) - (m/32 + m/4) + m/4 = m/4 + m/32 = 9m/32 \end{aligned}$$

So this proves that no local optimum is better than $9m/32$. To see that there is indeed a global optimum (which is necessarily a local optimum), observe that this is just a sum

of separable 2-variable optimizations (there is no interaction across different i). So now consider any fixed $i \in S$, and fixed y_i . Then the optimum is achieved with single-variate optimization, and is at $x_i = y_i + 1/4$, yielding $x_i/2 - (x_i - y_i)^2/2 - x_i/4 - y_i/4 = 1/32$. Similar work holds for any $i \notin S$, so summing over all i the optimum is $m/32 + m/4 = 9m/32$ for all i . \square

It turns out that $23/32$ is tight for the described relaxation of the above scheme: consider $x_i = 5/8$ for all $i \in S$, and $3/8$ for all $i \notin S$. $y_i = 3/8$ for all $i \in S$, and $5/8$ for all $i \notin S$. It is possible to come up with a Binary XOS function which induces this summary (see [?] for construction. It simply sets all of the inequalities as equalities and confirms that a solution exists).

3 Possible Project Topics

(Disclaimer: I suspect that all of these projects are quite challenging, and only make sense if you're excited about them and want to spend a decent amount of effort).

- Achieve an approximation ratio better than $23/32$ for the decision problem (with simultaneous protocols).
- Achieve an approximation ratio better than $23/32$ for search with a deterministic protocol (with simultaneous protocol).
- Achieve an approximation ratio better than $23/32$ for search with a randomized protocol for all of XOS (with simultaneous protocol).
- Prove a lower bound better than $3/4 - 1/108 = 20/27$ for problem 1 (see next class for a lower bound of $3/4 - 1/108$).
- Prove a lower bound better than $3/4$ for problems 2 or 3.