Scheduling in distributed optimization

Pierre Coucheney Stéphane Durand Bruno Gaujal Corinne Touati

Univ. Versailles, Ens de Lyon, Univ. Grenoble Alpes and Inria

Lyon – July 2014

Outline

Distributed Optimization

Consider a function $F : \{0, A\}^N \to \mathbb{R}$, to be optimized in a distributed way.
N is the number of dimensions (agents)
$\{0, A\}$ is the action space of each agent (w.n.l.g.).

Distributed Optimization

Consider a function $F : \{0, A\}^N \to \mathbb{R}$, to be optimized in a distributed way.
N is the number of dimensions (agents)
$\{0, A\}$ is the action space of each agent (w.n.l.g.).

State $x = (x_1, x_2, \dots, x_N)$

Distributed Optimization

Consider a function $F : \{0, A\}^N \to \mathbb{R}$, to be optimized in a distributed way.
N is the number of dimensions (agents)
$\{0, A\}$ is the action space of each agent (w.n.l.g.).

State $x = (x_1, x_2, \ldots, x_N)$ is a global optimum if $F(x) = \max_{y \in \{0,A\}^N} F(y)$.

Distributed Optimization

Consider a function $F : \{0, A\}^N \to \mathbb{R}$, to be optimized in a distributed way.
N is the number of dimensions (agents)
$\{0, A\}$ is the action space of each agent (w.n.l.g.).

State $x = (x_1, x_2, \ldots, x_N)$ is a global optimum if $F(x) = \max_{y \in \{0,A\}^N} F(y)$.

x is a local optimum if $\forall i$, $F(x) = \max_{\alpha \in \{0,A\}} F(\alpha, x_{-i})$.

Assumption (A)

We assume that for all i and for all x,

$$\operatorname*{argmax}_{\alpha \in \{0,A\}} F(\alpha, x_{-i}) \quad \text{is unique.}$$

Example in dimension $N = 2$

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example in dimension $N = 2$

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example in dimension $N = 2$

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Asynchronous Greedy Algorithm (AGA)

Asynchronous Greedy Algorithm (AGA)

1. Pick one agent i (with a given distribution over all agents)
2. Agent i chooses the action that maximizes F
3. Go back to 1.

Example in dimension 2 (with 2 agents)

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example in dimension 2 (with 2 agents)

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example in dimension 2 (with 2 agents)

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example in dimension 2 (with 2 agents)

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example in dimension 2 (with 2 agents)

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example in dimension 2 (with 2 agents)

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example in dimension 2 (with 2 agents)

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example in dimension 2 (with 2 agents)

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Convergence to Local Optima

Theorem

Algorithm AGA converges in finite time a.s. to a local optimum of F.

Convergence to Local Optima

Theorem

Algorithm AGA converges in finite time a.s. to a local optimum of F.

Proof. Each time one coordinate is changed, the value increases (so it must converge to a local optimum).

General Greedy Algorithm

AGA is distributed (each agent acts independently of the others) but requires a time coordination between them. At each step a **single** agent must be selected. In distributed systems this requires an election mechanism, that may be costly.

General Greedy Algorithm

AGA is distributed (each agent acts independently of the others) but requires a time coordination between them. At each step a **single** agent must be selected. In distributed systems this requires an election mechanism, that may be costly.
An alternative is to let several agents act *simultaneously*.
Let $\mathcal{R}$ be a family of revision sets (sets of agents that can act simultaneously).

Greedy Algorithm (GA)

1. Pick one revision set S (with a given distribution).
2. Each agent in S chooses the action that maximizes F.
3. Go back to 1.

Example (continued)

Revision sets: $\{1, 2\}$ (both agents always play together).

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example (continued)

Revision sets: $\{1, 2\}$ (both agents always play together).

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example (continued)

Revision sets: $\{1, 2\}$ (both agents always play together).

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example (continued)

Revision sets: $\{1, 2\}$ (both agents always play together).

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example (continued)

Revision sets: $\{1, 2\}$ (both agents always play together).

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example (continued)

Revision sets: $\{1, 2\}$ (both agents always play together).

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example (continued)

Revision sets: $\{1, 2\}$ (both agents always play together).

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Example (continued)

Revision sets: $\{1, 2\}$ (both agents always play together).

1	3	1	0	4	2	1	0
4	1	9	0	0	3	2	0
5	1	3	3	4	1	1	2
7	3	1	4	6	2	1	1
0	4	0	3	0	2	0	3
2	3	0	0	5	4	1	1
5	0	1	5	0	1	6	0
2	2	8	0	1	0	0	1

Separable Families

Let $\mathcal{R}$ be a family of sets and consider the following elimination process:

As long as there is a singleton (say $\{k\}$) in $\mathcal{R}$, remove k from all sets in $\mathcal{R}$.

$\mathcal{R}$ is *separable* if the elimination process reduces $\mathcal{R}$ to the empty set.

Example:
$\mathcal{R}_1 = \{1\}, \{1, 2, 3\}, \{2, 4\}, \{1, 4\}$ is separable
but
$\mathcal{R}_2 = \{1\}, \{1, 2, 3\}, \{2, 4\}, \{3, 4\}$ is not separable

$\mathcal{R}_3 =$ all the sets obtained when each agent i decides to play with probability p_i is separable (and fully distributed).

Separability and Convergence to Local Optima

Theorem

The algorithm GA converges to a local optimum for all functions F satisfying (A) if and only if the revision set is separable.

Proof.

1) By contradiction.

If $\mathcal{R}$ is separable, and GA does not converge to a local optimum, let x be the state with maximal value visited by GA.

Separability and Convergence to Local Optima

Theorem

The algorithm GA converges to a local optimum for all functions F satisfying (A) if and only if the revision set is separable.

Proof.

1) By contradiction.

If $\mathcal{R}$ is separable, and GA does not converge to a local optimum, let x be the state with maximal value visited by GA.

From x, let us select agents in the order of the elimination for separability (this happens with a positive probability).

Separability and Convergence to Local Optima

Theorem

The algorithm GA converges to a local optimum for all functions F satisfying (A) if and only if the revision set is separable.

Proof.

1) By contradiction.
If $\mathcal{R}$ is separable, and GA does not converge to a local optimum, let x be the state with maximal value visited by GA.
From x, let us select agents in the order of the elimination for separability (this happens with a positive probability).
The value increases (impossible) or x is a local optimum (impossible).

Proof (continued)

2) By construction.
Assume GA always converges to a local optimum under $\mathcal{R}$.

Proof (continued)

2) By construction.
Assume GA always converges to a local optimum under $\mathcal{R}$.
Actions of each agent: $\{0, \ldots, A\}$ where $p = A + 1$ is a prime number $> N$.

Proof (continued)

2) By construction.
Assume GA always converges to a local optimum under $\mathcal{R}$.
Actions of each agent: $\{0, \ldots, A\}$ where $p = A + 1$ is a prime number $> N$.
We want to *minimize* $F(x) = \sum_i x_i \mod p$.

Its optimal value is 0.

Proof (continued)

2) By construction.
Assume GA always converges to a local optimum under $\mathcal{R}$.
Actions of each agent: $\{0, \dots, A\}$ where $p = A + 1$ is a prime number $> N$.
We want to *minimize* $F(x) = \sum_i x_i \mod p$.

Its optimal value is 0.
Assume agent i is selected at the next round (maybe with many others) at state x with value $k = F(x)$.

Proof (continued)

2) By construction.
Assume GA always converges to a local optimum under $\mathcal{R}$.
Actions of each agent: $\{0, \ldots, A\}$ where $p = A + 1$ is a prime number $> N$.
We want to *minimize* $F(x) = \sum_i x_i \mod p$.

Its optimal value is 0.
Assume agent i is selected at the next round (maybe with many others) at state x with value $k = F(x)$.
Its best action is $(x_i - k) \mod p$.

Proof (continued)

2) By construction.
Assume GA always converges to a local optimum under $\mathcal{R}$.
Actions of each agent: $\{0, \ldots, A\}$ where $p = A + 1$ is a prime number $> N$.
We want to *minimize* $F(x) = \sum_i x_i \mod p$.

Its optimal value is 0.
Assume agent i is selected at the next round (maybe with many others) at state x with value $k = F(x)$.
Its best action is $(x_i - k) \mod p$.
If m players play simultaneously, the new state has value $(1 - m)k \mod p$.

Proof (continued)

2) By construction.
Assume GA always converges to a local optimum under $\mathcal{R}$.
Actions of each agent: $\{0, \ldots, A\}$ where $p = A + 1$ is a prime number $> N$.
We want to *minimize* $F(x) = \sum_i x_i \mod p$.

Its optimal value is 0.
Assume agent i is selected at the next round (maybe with many others) at state x with value $k = F(x)$.
Its best action is $(x_i - k) \mod p$.
If m players play simultaneously, the new state has value $(1 - m)k \mod p$.
Since p is prime, we can only reach $F = 0$ when $m = 1$: $\mathcal{R}$ must contain a singleton.

Proof (continued)

2) By construction.
Assume GA always converges to a local optimum under $\mathcal{R}$.
Actions of each agent: $\{0, \ldots, A\}$ where $p = A+1$ is a prime number $> N$.
We want to *minimize* $F(x) = \sum_i x_i \mod p$.

Its optimal value is 0.
Assume agent i is selected at the next round (maybe with many others) at state x with value $k = F(x)$.
Its best action is $(x_i - k) \mod p$.
If m players play simultaneously, the new state has value $(1-m)k \mod p$.
Since p is prime, we can only reach $F = 0$ when $m = 1$: $\mathcal{R}$ must contain a singleton.
The rest holds by induction on N.

Randomized Algorithm

The problem with the Greedy Algorithm is that convergence to local optima may not be good enough.

Randomized Algorithm

The problem with the Greedy Algorithm is that convergence to local optima may not be good enough.
To give a chance to agents to exit from a local optimum, we randomize their choices. The choice $Q_i(x)$ of agent i under state x is:

Randomized Algorithm

The problem with the Greedy Algorithm is that convergence to local optima may not be good enough.
To give a chance to agents to exit from a local optimum, we randomize their choices. The choice $Q_i(x)$ of agent i under state x is:

$$\mathbb{P}(Q_i(x) = a) = \frac{e^{\theta F(a,x_{-i})}}{\sum_b e^{\theta F(b,x_{-i})}}$$

Randomized Algorithm

The problem with the Greedy Algorithm is that convergence to local optima may not be good enough.
To give a chance to agents to exit from a local optimum, we randomize their choices. The choice $Q_i(x)$ of agent i under state x is:

$$\mathbb{P}(Q_i(x) = a) = \frac{e^{\theta F(a, x_{-i})}}{\sum_b e^{\theta F(b, x_{-i})}}$$

Randomized Algorithm (RA)

1. Pick one revision set S (with a given distribution ρ).
2. Each agent i in S chooses action $Q_i(x)$
3. Go back to 1.

Randomized Algorithm (II)

The evolution of the state x is Markovian. The transition matrix has two parts: first choose the revision set S, then choose the new action for each agent in S.

Randomized Algorithm (II)

The evolution of the state x is Markovian. The transition matrix has two parts: first choose the revision set S, then choose the new action for each agent in S.

The (irreducible) transition matrix P is

$$P_{x,y} = \sum_{S \supseteq \mathrm{Diff}(x,y)} \rho(S) \prod_{i \in S} \frac{e^{\theta F(y_i, x_{-i})}}{\sum_{\alpha \in A} e^{\theta F(\alpha, x_{-i})}}.$$

Randomized Algorithm (II)

The evolution of the state x is Markovian. The transition matrix has two parts: first choose the revision set S, then choose the new action for each agent in S.

The (irreducible) transition matrix P is

$$P_{x,y} = \sum_{S \supseteq \mathrm{Diff}(x,y)} \rho(S) \prod_{i \in S} \frac{e^{\theta F(y_i, x_{-i})}}{\sum_{\alpha \in A} e^{\theta F(\alpha, x_{-i})}}.$$

Let $\pi(\theta)$ be the (unique) stationary measure of P.

Randomized Algorithm (II)

The evolution of the state x is Markovian. The transition matrix has two parts: first choose the revision set S, then choose the new action for each agent in S.
The (irreducible) transition matrix P is

$$P_{x,y} = \sum_{S \supseteq \mathrm{Diff}(x,y)} \rho(S) \prod_{i \in S} \frac{e^{\theta F(y_i, x_{-i})}}{\sum_{\alpha \in A} e^{\theta F(\alpha, x_{-i})}}.$$

Let $\pi(\theta)$ be the (unique) stationary measure of P.
A state x is stochastically stable if $\lim_{\theta \to \infty} \pi_x(\theta) > 0$

Randomized Algorithm (II)

The evolution of the state x is Markovian. The transition matrix has two parts: first choose the revision set S, then choose the new action for each agent in S.

The (irreducible) transition matrix P is

$$P_{x,y} = \sum_{S \supseteq \text{Diff}(x,y)} \rho(S) \prod_{i \in S} \frac{e^{\theta F(y_i, x_{-i})}}{\sum_{\alpha \in A} e^{\theta F(\alpha, x_{-i})}}.$$

Let $\pi(\theta)$ be the (unique) stationary measure of P.

A state x is stochastically stable if $\lim_{\theta \to \infty} \pi_x(\theta) > 0$

When $\theta \to \infty$, (RA) $\to$ (GA),

however $\pi_x(\theta) \not\to \pi_x(\infty)$, (the stable states of (GA)), but selects a subset.

Convergence to Local Optima for (RA)

Theorem (Convergence to local optimal)

If the revision set $\mathcal{R}$ is separable, then the stochastically stable states are local optima.

proof. Use the explicit form of the stationary distribution and compute equivalents w.r.t. θ.
Tree Theorem: Let $\mathcal{T}_x$ be the set of spanning in-trees of the transition graph, with root in x. The stationary distribution π is proportional to the sum of the probability weights of all the spanning trees T in $\mathcal{T}_x$:

$$\pi_x \propto \sum_{T \in \mathcal{T}_x} \prod_{(y,z) \in T} P_{y,z}.$$

Convergence to Global Optima

Theorem (Convergence to global optima for asynchronous revisions)

If the revision family is only made of all the singletons, then the only stochastically stable states are the global optima.

Theorem (Convergence to global optimal for two players)

If the revision family is $\{1\}, \{2\}, \{1, 2\}$*, then the only stochastically stable states are the global optima.*

Example 1: 2 agents, no convergence

$F =$

$1 \backslash 2$	a	b
a	1	0.5
b	0	1

Revision set: $\{1, 2\}$

Example 1: 2 agents, no convergence

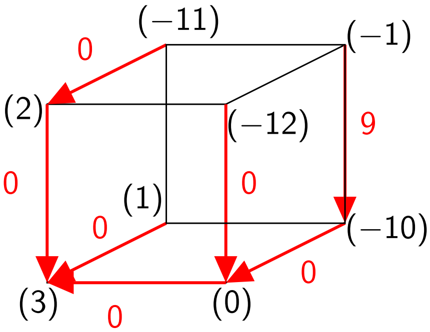
$F =$

$1\backslash 2$	a	b
a	1	0.5
b	0	1

Revision set: $\{1, 2\}$

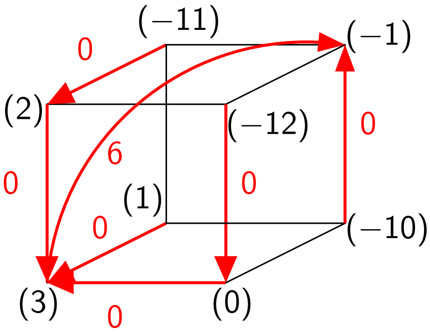
$$\pi(((a,a),(a,b),(b,a),(b,b)) \to (1/4, 1/4, 1/4, 1/4).$$

Example 2: 3 agents, convergence to LO



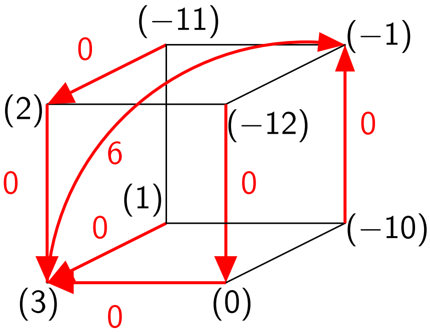
Revision set: $\{1\}, \{2\}, \{3\}, \{1, 2, 3\}$.

Example 2: 3 agents, convergence to LO



Revision set: $\{1\}, \{2\}, \{3\}, \{1, 2, 3\}$.

Example 2: 3 agents, convergence to LO



Revision set: $\{1\}, \{2\}, \{3\}, \{1, 2, 3\}$.

Unique stable state: $(1, 1, 1)$ (not global optimum).

Examples 3: 2 agents, convergence to LO

$F =$

1\2	a	b	c
a	11	0	5
b	5	10	8

Separable revision set: $\{2\}, \{1, 2\}$.

Examples 3: 2 agents, convergence to LO

$F =$

$1\backslash 2$	a	b	c
a	11	0	5
b	5	10	8

Separable revision set: $\{2\}, \{1, 2\}$.

Unique stable state: (b, b), not global optimum.

Example 4: 2 agents, no convergence

$F =$

1\2	a	b
a	1	1
b	1	0

Revision set $\{1, \}, \{2\}, \{1, 2\}$.

Example 4: 2 agents, no convergence

$$F = \begin{array}{|c||c|c|} \hline 1\backslash 2 & a & b \\ \hline a & 1 & 1 \\ b & 1 & 0 \\ \hline \end{array}$$

Revision set $\{1, \}, \{2\}, \{1, 2\}$.

$\pi((a, a), (a, b), (b, a), (b, b)) \rightarrow (36/79, 20/79, 20/79, 3/79)$