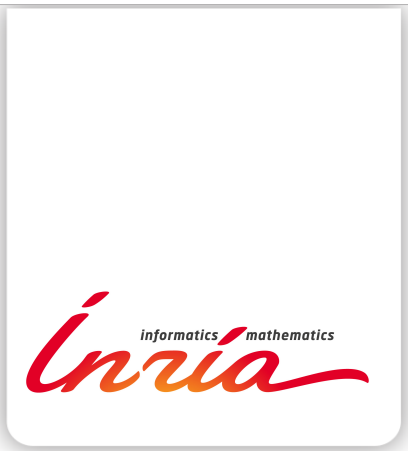


Complexity and Optimality of the Best Response Algorithm in Random Potential Games

Stéphane Durand, Bruno Gaujal

RESEARCH REPORT
N° 8925
June 2016
Project-Team Polaris

ISSN 0249-6399 ISRN INRIA/RR--8925--FR+ENG



Complexity and Optimality of the Best Response Algorithm in Random Potential Games

Stéphane Durand, Bruno Gaujal

Project-Team Polaris

Research Report n° 8925 — June 2016 — ?? pages

Abstract: In this paper we compute the worst-case and average execution time of the Best Response Algorithm (BRA) to compute a pure Nash equilibrium in finite potential games. Our approach is based on a Markov chain model of BRA and a coupling technique that transform the average execution time of this discrete algorithm into the solution of an ordinary differential equation. In a potential game with N players and A strategies per player, we show that the worst case complexity of BRA (number of moves) is exactly NA^{N-1}, while its average complexity over random potential games is equal to $e^{\gamma}N + O(N)$, where γ is the Euler constant. We also show that the effective number of states visited by BRA is equal to $\log N + c + O(1/N)$ (with $c \leqslant e^{\gamma}$), on average. Finally, we show that BRA computes a pure Nash Equilibrium faster (in the strong stochastic order sense) than any local search algorithm over random potential games.

Key-words: Potential Games, Best Response, Average Complexity, Coupling

RESEARCH CENTRE
GRENOBLE – RHÔNE-ALPES

Inovallée
655 avenue de l'Europe Montbonnot
38334 Saint Ismier Cedex

Complexité et optimalité de l'algorithme de meilleure réponse dans les jeux de potentiel randomisés

Résumé : Dans cet article, nous calculons le temps d'exécution en moyenne et en pire cas de l'algorithme de meilleure réponse (AMR) pour calculer un équilibre de Nash pur dans les jeux de potentiel finis. Notre approche utilise une chaîne de Markov pour modéliser le comportement de AMR et un couplage qui transforme le temps d'exécution de AMR en la solution d'une équation différentielle. Pour un jeu de potentiel avec N joueurs et A stratégies par joueur, le nombre d'itérations de AMR est exactement NA^{N-1} dans le pire cas et $e^{\gamma}N + O(N)$ dans le cas moyen, où γ est la constante d'Euler. Nous montrons aussi que le nombre de changements d'état est $\log N + c + O(1/N)$ (avec $c \leqslant e^{\gamma}$), en moyenne. Finalement, nous montrons aussi que AMR calcule un équilibre de Nash plus vite que tout algorithme de recherche local, pour l'ordre stochastique fort.

Mots-clés : Jeux de Potentiel, Meilleure Réponse, Complexité en Moyenne, Couplage

1 Introduction

The question of computing Nash Equilibria (NE) in games is a central question in algorithmic game theory and has been investigated of many papers. The most classical result is in [**?**], showing that the problem of computing NE in finite games is PPAD complete.

Potential games have been introduced in [**?**] and have proven very useful, especially in the context of routing problems in networks, first mentioned in [**?**] and exhaustively studied ever since, in the transportation as well as computer science literature, see for example [**?**, **?**, **?**]. They have also been heavily investigated in the context of distributed optimization (see for example [**?**]). In [**?**, **?**] the authors show that the computation of NE for general potential games is PLS complete (Polynomial Local Search complete). As for PPAD, this complexity class is believed to be different from P.

The best response dynamics is one of the most basic tool in game theory. The original proof of the existence of a Nash Equilibrium by Nash [**?**] can be seen as the proof of existence of a fixed point of the best response correspondence (best response is called *countering* in [**?**]). It has been well-known for a long time that the Best Response Algorithm converges in finite time to a pure NE in potential games [**?**]. So BRA is a natural candidate for computing Nash equilibria.

In this paper, we analyze the performance of BRA over potential games with N players, each with A possible strategies. It is well known that the convergence time of BRA over potential games can be exponential in the number of players (see for example [**?**]). Here, we confirm this by showing that the worst case complexity of BRA (number of plays) is exactly NA^{N-1}. Special cases, such as graphical potential games have been analyzed in [**?**] by showing an equivalence between the potential of such games and Markov fields. In other special cases such as scheduling congestion games with identical tasks, it has been show that BRA takes at most N steps before finding a NE [**?**]. Extensions with positive and negative externalities also have a linear complexity [**?**].

However the *average* complexity of BRA over all potential games has attracted surprisingly little attention. Random (non potential) games with two players have been studied in [**?**]: With two IID utility matrices of size $A \times A$, the computation of a NE is $O(A^3 \log\log A)$ with high probability using a rather sophisticated algorithm.

Our main contribution is to show that for potential games with N players, $\mathbb{E}[M_{BRA}]$, the average number of strategy profiles visited by BRA before convergence, is $\mathbb{E}[M_{BRA}] = \log(N) + C + O(1/N)$ (where $C \leqslant e^{\gamma}$, γ being the Euler constant). We also show that the average number of comparisons performed by the algorithm is equal to $e^{\gamma}(A-1)(N-1) + o(AN)$. This could be intuitively explained by the fact that random potential games have a lot of pure NE [**?**]. In our framework, potentials are IID random variables so that, on average, one action profile out of $(A-1)N+1$ profiles is a NE while in the worst case, a potential game may have a single pure NE. This is only a partial explanation, however. This does not explain the fact that the complexity does not depend on the number of actions, nor the value of the constant factor, $e^{\gamma} \approx 1.78$.

We further show that the Best Response Algorithm computes a pure Nash Equilibrium faster than any algorithm based on player's local information, not only in average but also in the strong stochastic order sense.

1.1 Coupling and Markovian Analysis

The main idea of our approach is to see the evolution of BRA in a random environment as a dynamical system, whose behavior can be computed using differential equations. This will allow us to compute the exact asymptotics of the average complexity in N and A, not only $O(.)$ bounds. Second moments of T_{BRA} and of M_{BRA} can also be computed by the same approach

(in Appendix **??**).

The first step (§ 4.2) is to construct an approximation of the behavior of BRA over a potential game. This approximation is called IFA in the paper, for Intersection-Free Approximation because it discards strategies already explored by BRA. We show that the execution time of BRA is smaller than the execution time of its IFA approximation for the strong stochastic order. This is done by constructing a non-trivial *coupling* between both executions. This powerful technique is exploited to our great benefit here.

The second and most important step (§ 4.4) is to consider one run of the IFA approximation of BRA as a trajectory of a Markov chain over the continuous space of potentials. Doing so, the average complexity is transformed into the average hitting time of an absorbing state of the Markov chain. The theory of Markov chains implies that this average hitting time satisfies a Poisson differential equation. Thus, the average complexity of BRA is given by the solution of a system of ordinary differential equations. This system happens to have a solution in closed form whose asymptotics in N and A can be computed by taking integrals over initial states.

As for the proof of optimality of BRA among all local search algorithms (§ **??**), our approach is based once again on a coupling argument. While using coupling techniques is more classical in this context (comparison of algorithms), this particular case retains some originality because the coupling used here is not built off-line but is being constructed on the fly while the algorithm runs.

2 Best Response Algorithm and Potential Games

We consider a game with a finite number N of players, each with A strategies.

Definition 1 (**N-player game).** *A game is a tuple* $\mathfrak{G} \stackrel{\text{def}}{=} \mathfrak{G}(\mathcal{N}, \mathcal{A}, u)$ *with*

- *a finite set of* **players** $\mathcal{N} = \{1, \ldots, N\}$*;*
- *a finite set* $\mathcal{A}_k$ *of pure* **strategies** *for each player* $k \in \mathcal{N}$*.*
  The set of **strategy profiles** *or* **states** *of the game is* $\mathcal{A} \stackrel{\text{def}}{=} \mathcal{A}_1 \times \mathcal{A}_2 \times \cdots \mathcal{A}_N$*.*
- *The players'* **payoff** *functions* $u_k : \mathcal{A} \to \mathbb{R}$*.*

We define the *best response correspondence* $\mathbf{br}_k(x)$ as the set of all strategies that maximizes the payoff for player k under profile $x = (x_1, \ldots, x_N)$: $\mathbf{br}_k(x) \stackrel{\text{def}}{=} \left\{ \underset{\alpha \in \mathcal{A}_k}{\operatorname{argmax}}\, u_k(\alpha; x_{-k}) \right\}$.

A *Nash equilibrium* (NE) is a fixed point of this correspondence, i.e. a profile x^* such that $x^*_k \in \mathbf{br}_k(x^*)$ for every player k.

Definition 2 (**Potential games and its generalizations).** *A game is an (exact)* potential game *[?] if it admits a function (called the potential)* $\Phi : \mathcal{A} \to \mathbb{R}$ *such that for any player* k *and any* unilateral *deviation of* k *from strategy profile* x *to* x'*:* $u_k(\alpha, x_{-k}) - u_k(\alpha', x_{-k}) = \Phi(\alpha, x_{-k}) - \Phi(\alpha', x_{-k})$.

A game is a generalized ordinal potential game *[?] (or G-potential game for short) if there exists a potential function* $\Phi : \mathcal{A} \to \mathbb{R}$ *such that, for any player* k *and any state* x*,* $u_k(\alpha, x_{-k}) > u_k(\alpha', x_{-k}) \Rightarrow \Phi(\alpha, x_{-k}) > \Phi(\alpha', x_{-k})$.

A game is a best-response potential game *[?] (or BR-potential game for short) if there is* $\Phi : \mathcal{A} \to \mathbb{R}$ *such that for any player* k *and strategy profile* x*,* $\mathbf{br}_k(x) = \left\{ \underset{\alpha \in \mathcal{A}_k}{argmax}\, \Phi(\alpha, x_{-k}) \right\}$.

As shown in [**?**], exact potential games are BR-potential games, but there exist G-potential games that are not BR-potential games. In the following, we will consider the most general case (*i.e.* all games that are either BR-potential or G-potential games) and call them potential games for simplicity.

We consider a general version of the *Best Response Algorithm* (BRA) with uniform choice over all possible best responses when ties occur and where the next player is selected according to a *revision function* $R(.)$, that may depend of the whole past of the algorithm. We assume that this function is *weakly fair*: each player appears infinitely often in the sequence of plays induced by R, almost surely. This revision function can be deterministic (for example, round-robin: $R(t) = t \mod N$) or random (for example, Bernoulli where the next player is chosen according to an probability distribution ρ (the *revision law*): $\forall k \in \mathcal{N}, \mathbb{P}(R(t) = k) = \rho_k$). In that case, weak fairness implies that the probability of choosing any player k is strictly positive ($\forall k \in \mathcal{N}, \rho_k > 0$).

Algorithm 1: Best Response Algorithm (BRA)

Input: Game utilities ($u_k(\cdot)$); Initial state ($x := x(0)$);
Weakly fair revision function R;
List of satisfied customers, initially empty: $L := \emptyset$;

repeat
  Pick next player $k := R(t)$; $t := t + 1$;
  if $x_k \notin \mathbf{br}_k(x)$ **then**
    Update strategy for player k to any $x_k \in \mathbf{br}_k(x)$;
    $L := \emptyset$;
  $L := L \cup \{k\}$;
until *size(L)* $= N$;

It is well known (see [**?**]) that for any potential game $\mathfrak{G}$, Algorithm 1 converges in finite time, almost surely, to a Nash Equilibrium of $\mathfrak{G}$.

3 Worst Case Complexity

In this section, we analyze the time complexity of BRA. More precisely, we consider three measures (related to each other). The first one is T_{BRA}, the number of iterations (or the number of times that the function **br** is called) before BRA reaches a Nash equilibrium. A related measure is the total number of comparisons used by BRA (denoted C_{BRA}). One should expect that $C_{BRA} \approx (A-1)T_{BRA}$. Finally, another interesting quantity is the number of different states visited by BRA (denoted M_{BRA}). This is called the number of *moves* done by BRA before convergence to a Nash equilibrium (NE). Of course, $M_{BRA} \leqslant T_{BRA}$.

These quantities depend on the game over which BRA is run, on the initial state x^0 and on the infinite sequence of revision players R. It should also be clear that they are functions of the game only through the potential Φ, so we denote by $T_{BRA}(\Phi, x^0, R)$ the number of steps before convergence of Algorithm BRA for a game with potential Φ, starting in state x^0, under the condition that the sequence of players is given by R.

In the worst case, for some weakly fair revision functions R, $T_{BRA}(\Phi, x^0, R)$ can be unbounded because the revision sequence induced by R can be arbitrarily bad: one player might appear too few times to guarantee convergence in any bounded time. When R is the round-robin function, the time for convergence is finite but can still be very large, as shown in the following theorem.

Theorem 1. *In the worst case,* $T_{BRA}(\Phi, x^0, \textit{round-robin}) = NA^{N-1}$.

It is well known that the worst case complexity of BRA is exponential in the number of players (see for example [**?**]). The version of this result given here (for round robin revision and generalized potential games) is only given for the record (the proof is given in Appendix **??**).

4 Average Complexity of BRA

4.1 Randomization

In the following we will randomize over the potential games over which BRA is used. Since the behavior of BRA only depends on the potential function, we randomize directly over the potential Φ.

We consider a randomization over all games, uniformly over all possible orders for the potentials. On one hand, this is the classical average complexity approach when no additional information is known about the games (the same approach is used in [**?**] for 2 player games for example). This yields IID potential for all profiles, as explained below. On the other hand, some may argue that uniformly random games are not generic in some sense and a good performance of BRA on average does not necessarily translate in good performances for "real word" games. In any case, this is a first step that must be taken in absence of additional information about specific games that one may want to study.

There are several equivalent ways to do this randomization. The first one is based on the fact that the complexity of the algorithm does not depend on the actual values of the potential of the states but only on the comparisons between them. When two potentials are equal, a strict order between them is chosen uniformly. Therefore, the natural randomization is to consider the linear extensions (total orders) of all possible partial orders over the set $\mathcal{A}$ and pick one uniformly. The number of total orders on $\mathcal{A}$ is the number of permutations on $\mathcal{A}$, namely $(A^N)!$.

The second (equivalent) randomization is the following: The potentials of all states x are chosen independent, identically distributed according to an arbitrary distribution F admitting a density w.r.t. the Lebesgue measure.

Both randomizations are equivalent. Indeed, take any k states $x_1, \ldots, x_k$ in $\mathcal{A}$. In both cases, $\mathbb{P}(\Phi(x_1) > \Phi(x_2) > \cdots > \Phi(x_k)) = 1/k!$. Now, since F is increasing, F^{-1} is well-defined and we get $\mathbb{P}(\Phi(x) > \Phi(x')) = \mathbb{P}(F^{-1}(\Phi(x)) > F^{-1}(\Phi(x')))$. Note that $F^{-1}(\Phi(x))$ is uniformly distributed on $[0, 1]$. Therefore, with no loss of generality, one can assume that the potential of all the states are i.i.d., uniformly distributed on $[0, 1]$. This randomization is used in the following.

4.2 Intersection-Free Approximation

The direct analysis of the behavior of BRA over a random potential is difficult because, over time, more and more states have been visited by the algorithm. Thus, its behavior is non-homogeneous in time. To avoid this difficulty, we consider a new model, called the *Intersection-Free Approximation* (IFA) in the following. Under the Intersection-Free Approximation, every time a new player (say k) has to compute its best response in a state (say x), it compares $\Phi(x)$ with the potential of its $A - 1$ other possible strategies, as for the real BRA. Here however, we assume that those $A - 1$ states have not yet been visited during the previous steps of the algorithm. Note that under the real behavior of BRA, it could happen that some of these states have already been compared at a previous step of the algorithm, by another player (this will be called an *intersection* in the following). Under the Intersection-Free Approximation, the states visited by the algorithm are always "new" states, never compared before with any other states.

More formally, the algorithm BRA under IFA can be written as follows.

Algorithm 2: BRA algorithm under IFA

Input: Initial state ($x(0)$); Revision function R;
Set of satisfied players, initially empty $L := \emptyset$.

repeat
  Pick next player $k := R(t)$; $t := t + 1$;
  if $k \notin L$ **then**
    Generate IID potentials $\Phi(\alpha, x_{-k}), \alpha \in \mathcal{A}_k \setminus \{x_k\}$ unif. on $[0, 1]$;
    Compute best response: $\alpha_k := \underset{\beta \in \mathcal{A}_k}{\text{argmax}}\, \Phi(\beta; x_{-k})$;
    if $\alpha_k = x_k$ **then**
      $L := L \cup \{k\}$
    else
      $L := \{k\}$; $x_k := \alpha_k$;
until $size(L) = N$;

Let us recall that C_{BRA} (resp. T_{BRA}, M_{BRA}) is the number of comparisons (resp. number of steps, number of moves) taken by BRA before convergence and let us define C_{IFA} (resp. T_{IFA}, M_{IFA}) to be the number of comparisons (steps, moves) of BRA under the intersection-free approximation. By definition, the worst case complexity of IFA under a round-robin revision sequence is infinite. However, its average complexity is the same as for BRA, as shown by the following lemma.

Lemma 1 (BRA and IFA are asymptotically equivalent)**.** *Under the foregoing notations and using a round-robin revision function, the following comparisons hold, where $\leqslant_{st}$ is the strong stochastic order:*

1. $C_{BRA} \leqslant_{st} C_{IFA}$ *(equivalently, [?]* $\forall t \in \mathbb{R}$, $\mathbb{P}(C_{BRA} > t) \leqslant \mathbb{P}(C_{IFA} > t)$*).*
2. *If I is the total number of intersections in BRA, then* $T_{BRA} \leqslant_{st} T_{IFA} + \frac{I}{A-1}$.
3. $\mathbb{E}[T_{BRA}] = \mathbb{E}[T_{IFA}] + o(1)$ *and* $\mathbb{E}[C_{BRA}] = \mathbb{E}[C_{IFA}] + o(1)$,
4. $\mathbb{E}[M_{BRA}] = \mathbb{E}[M_{IFA}] + o(1)$.

The proof of the lemma is postponed to Appendix **??**. It is based on the construction of a *coupling* between the executions of BRA with and without IFA. The assumption that the revision function is round-robin for BRA and for IFA does not play a big role in the proof, and it could be removed. However, the following section, asserting the optimality of round-robin implies that extending the proof to more general revision functions has a limited interest.

4.3 Round-Robin and Other Revision Sequences

As for the worst case analysis, the revision sequence influences the average time complexity of the algorithm. We show that on average round-robin is asymptotically the best one.

Lemma 2 (**Asymptotic optimality of round-robin**)**.** *For any revision function R,* $\mathbb{E}\left[T_{BRA}(\Phi, x^0, \textit{round-robin})\right] \leqslant \mathbb{E}\left[T_{BRA}(\Phi, x^0, R)\right] + \epsilon(N)$*, where the expectation is taken over all potentials Φ and all initial states x^0 and $\epsilon(N)$ goes to zero when N goes to infinity.*

The proof is given in appendix **??**. It uses the comparison with IFA. In the rest, we focus on round-robin revision functions and omit it in the notations, unless specified otherwise.

4.4 Complexity Analysis

We will be analyzing the intersection-free approximation of the behavior of BRA, under a round-robin revision sequence, with no further reference to this.

Let us consider the intersection-free approximation and let y be the potential of the current state x: ($y \stackrel{\text{def}}{=} \Phi(x)$). Let k be the number of players that have already played best response without changing the profile. This number of "satisfied" players can replace the explicit set L used in Algorithm 2 when the revision sequence is round-robin. The evolution at the next step of BRA under IFA is as follows. The kth player computes its best response. The player has $A-1$ new strategies whose potential must be compared with the current potential (y). As mentioned before, we can assume that the potentials of those $a \stackrel{\text{def}}{=} A-1$ strategies are IID, uniformly distributed in $[0,1]$.

With probability y^a none of the new strategies beat the current choice. The state remains at y, one more player is satisfied and it is the turn of the $k+1$-st player to try its best response.

With probability $1-y^a$, one of the new strategies is the best response. The current state moves to a new state where the number of satisfied players is set back to 1 and the potential increases to a value larger than $u > y$ with probability $1-u^a$.

Let Y_t be the potential at step t ($Y_t \in [0,1]$) and K_t be the current number of consecutive players whose best response did not change the current potential ($K_t \in \{1,2,\ldots,N\}$) (number of satisfied players). The previous discussion says that the couple (Y_t, K_t) is a discrete-time, continuous-space Markov chain whose kernel is:

$$\mathbb{P}\Big((Y_{t+1},K_{t+1}) = (y,k+1)\Big|(Y_t,K_t)=(y,k)\Big) = y^a,$$

and, for any $u > y$,

$$\mathbb{P}\Big((Y_{t+1},K_{t+1}) \in ([u,1],1)\Big|(Y_t,K_t)=(y,k)\Big) = 1-u^a.$$

All the other transitions have a null probability.

Let $m(y,k)$ be the number of moves of IFA before convergence when the current state of the Markov chain is equal to (y,k).

With probability y^a, the next player does not change its choice so that $m(y,k) = m(y,k+1)$. With probability density au^{a-1} the next player finds a new best response with potential u so that one move is taken and $m(y,k) = 1 + m(u,1)$.
Let $M(y,k) = \mathbb{E}[m(y,k)]$. The previous one step analysis of $m(y,k)$ makes $M(y,k)$ satisfy a forward Poisson equation:

$$M(y,k) = y^a M(y,k+1) + \int_y^1 au^{a-1}(M(u,1)+1)du.$$

By definition, the boundary conditions are: $\forall y, M(y,N) = 0$ (the current state is NE when all players agree on this) and $\forall k, M(1,k) = 0$ (the potentials are all bounded by 1, so a state with potential 1 is guaranteed to be a NE).

By setting $B(y) \stackrel{\text{def}}{=} \int_y^1 au^{a-1}(M(u,1)+1)du$, we get the following system of integral equations

$$\begin{cases} M(y,1) & = y^a M(y,2) + B(y), \\ M(y,2) & = y^a M(y,3) + B(y), \\ \vdots & = \vdots \\ M(y,N-2) & = y^a M(y,N-1) + B(y), \\ M(y,N-1) & = B(y). \end{cases} \tag{1}$$

Successive substitution of $M(y,2),\ldots,M(y,N-1)$ in the first equality yields $M(y,1) = B(y)H(y)$ where $H(y) \stackrel{\text{def}}{=} 1 + y^a + \cdots + y^{a(N-2)}$. Differentiating w.r.t. y, one gets an ordinary differential equation in $M(y,1)$:

$$\frac{dM(y,1)}{dy} + (ay^{a-1}H - \frac{1}{H}\frac{dH}{dy})M(y,1) = -ay^{a-1}H.$$

The equation is of the form $\dot{f} + gf = h$. Using the boundary condition $M(1,1) = 0$, its generic solution is

$$M(y,1) = e^{-Q(y)} \int_y^1 au^{a-1}H(u)e^{Q(u)}du. \tag{2}$$

where

$$Q(y) \stackrel{\text{def}}{=} \int_0^y \left(au^{a-1}H(u) - \frac{1}{H(u)}\frac{dH(u)}{du} \right) du = -\log(H(y)) + \int_0^y au^{a-1}H(u)du. \tag{3}$$

The average number of profile changes in the execution of the algorithm starting from an arbitrary profile is $\mathbb{E}[M_{IFA}] = \int_0^1 M(y,1)dy$. Since $M(y,1)$ is decreasing in y, $\mathbb{E}[M_{IFA}]$ is upper-bounded by $M(0,1)$. Using $Q(0) = 0$, $H(0) = 1$ and replacing Q and H by their values,

$$\begin{aligned} M(0,1) &= \int_0^1 \exp\left(\sum_{i=0}^{N-2} \frac{u^{a(i+1)}}{i+1} \right) au^{a-1}du \quad = \int_0^1 \exp\left(\sum_{i=0}^{N-2} \frac{v^{i+1}}{i+1} \right) dv \quad \text{(with } v = u^a\text{)} \\ &= \int_0^{1-\frac{1}{N}} \exp\left(\sum_{i=1}^{N-1} \frac{v^i}{i} \right) dv + \int_{1-\frac{1}{N}}^1 \exp\left(\sum_{i=1}^{N-1} \frac{v^i}{i} \right) dv \\ &\leqslant \int_0^{1-\frac{1}{N}} \exp\left(\sum_{i=1}^{\infty} \frac{v^i}{i} \right) dv + \frac{1}{N} \exp\left(\sum_{i=1}^{N-1} \frac{1}{i} \right) \qquad (4) \\ &= \int_0^{1-\frac{1}{N}} \frac{dv}{1-v} + e^\gamma + O(1/N) \quad = \log(N) + e^\gamma + O(1/N). \qquad (5) \end{aligned}$$

Furthermore, this bound is tight, up to an additive constant (see Appendix **??**).

Let us now consider the average number of comparisons made by BRA under the intersection-free assumption. Let $C(y,k)$ be the average number of comparisons starting in a state with potential y and k players have played without changing their strategy. The Poisson equation for $C(y,k)$ is :

$$C(y,k) = y^a(C(y,k+1) + a) + \int_y^1 au^{a-1}(C(u,1) + a)du,$$

with the boundary conditions $C(1,1) = a(N-1)$ and $C(y,N) = 0$.

The solution of this differential system can be obtained in closed form, using a similar approach as for $M(y,1)$.

$$C(y,1) = a\left(\sum_{i=0}^{N-2} y^{ai}\right)\exp\left(-\sum_{i=1}^{N-1}\frac{y^{ai}-1}{i}\right).$$

The average number of comparisons is $\mathbb{E}[C_{IFA}] = \int_0^1 C(y,1)dy$.
For all $y < 1$,

$$\begin{aligned} C(y,1) &= a\left(\sum_{i=0}^{\infty} y^{ai}\right)\exp\left(\sum_{i=1}^{N-1} 1/i\right)\exp\left(-\sum_{i=1}^{\infty} y^{ia}/i\right) + o(aN) && (6)\\ &= a\frac{1}{1-y^a}(N-1)e^{\gamma}(1-y^a) + o(aN) + O(1) && (7)\\ &= a(N-1)e^{\gamma} + o(aN), && (8)\end{aligned}$$

where γ is the Euler constant ($\gamma \approx 0.5772...$). Therefore, the same equality holds for the integral, equal to $\mathbb{E}[C_{IFA}]$.

The results of this section, together with Lemma 1, lead to the following theorem, the main result of the section.

Theorem 2 (**Average complexity of BRA).** *Under the round-robin revision sequence, the average complexity of BRA over a potential game satisfies:*
(i) Average number of moves: $\mathbb{E}[M_{BRA}] = \log(N) + c + O(1/N)$*., where* $c \leqslant e^{\gamma}$
(ii) Average number of comparisons : $\mathbb{E}[C_{BRA}] = e^{\gamma}AN + o(AN)$.
(iii) Average number of steps: $\mathbb{E}[T_{BRA}] = e^{\gamma}N + o(N)$.

The average complexity $\mathbb{E}[T_{BRA}]$ can be split into two parts: The number of plays before reaching a NE and the number of plays needed to check if a state is indeed a NE. This last part takes exactly $N-1$ steps in Algorithm 1: The players have to play one by one to fill up set L. This means that a NE equilibrium is reached on average as soon as $e^{\gamma} - 1 \approx 78\%$ of the players have played once. The second moments of the number of steps and the number of moves under IFA can be computed similarly (see Appendix **??**). In both cases, the standard deviations are of the same order as the means.

5 Optimality of BRA

In this section, we prove that BRA finds a Nash equilibrium faster than any local search algorithm (defined in Section **??**), in the strong stochastic order sense.

By definition a *Local Search Algorithm* can only access the payoff matrix, one player at a time. This access is often called a *query* in the literature. Once the payoff of a strategy profile has been obtained, it is stored in memory and can re-used later by the algorithm without an additional query.

In addition to queries, a local search algorithm can use any arithmetic operation, draw random variables and choose a strategy for all players.

Any local search algorithm can be written in the following form, based on the history of the execution, $\mathcal{H}_t$, that corresponds to the amount of information gathered by the algorithm up to

step t.

Algorithm 3: A general local search algorithm

Initial storage reduced to the initial profile: $\mathcal{H}_0 := \{(x(0)\}$.
repeat
  Select next player: $k := R(\mathcal{H}_t)$;
  Query payoff vector of k under current state: $u_k(\cdot, x_{-k}(t))$;
  Store the new visited states and their payoffs in memory:
  $\mathcal{H}_{t+1} := \mathcal{H}_t \cup \{((\alpha, x_{-k}(t)), u_k(\alpha, x_{-k}(t)))_{\alpha \in \mathcal{A}_k}\}$;
  jump to next state $x(t+1) := J(\mathcal{H}_{t+1})$;
  Set $stop := 1$ if the current state is a NE;
  $t := t + 1$;
until *stop*;

The functions J and R used in the inner loop are arbitrary functions that choose the next state as well as the next player to play, according to the whole history of the process. These functions can be deterministic or random. Testing if $x(t+1)$ is NE is not detailed. Notice, however, that it can only be done when all the payoff vectors for all the players in state $x(t+1)$ have been stored in memory.

The complexity of a local search algorithm A is defined as the total number of its payoff vector queries (denoted T_A).

Theorem 3 (**Optimality of BRA).** *Let A be any local search algorithm that computes a Nash Equilibria in potential games. Under the foregoing randomization, and choosing the starting point x^0 uniformly among all states, $\forall t \geqslant 0$, $\mathbb{P}(T_{BRA} \geqslant t | R = R_A) \leqslant \mathbb{P}(T_A \geqslant t)$, where R_A is the revision sequence constructed in A.*

The proof is reported in Appendix **??**. Combining this theorem with Lemma 2 establishes the optimality of BRA with round-robin.

6 Conclusion and Perspectives

The best response algorithm is one on the most basic object in game theory. In this paper, we prove it has a linear complexity on average over uniformly randomized potential games. Furthermore, BRA is optimal in the class of local search algorithms when one has no information about the structure of the potential game.

Does all this make BRA the perfect algorithm to compute NE in general? We believe that the answer is no because BRA suffers from several drawbacks. First, it does not tolerate simultaneous plays. Second, it requires to know the entire payoff vector of a player before choosing its strategy. Other drawbacks include high sensitivity on the order of play and on noisy perturbations on the payoffs. Designing algorithms that do not suffer from these drawbacks is the object of our future investigations.

A Worst Case Analysis, Proof of Theorem 1

Proof. Let us construct a game for which there exists a starting point that makes the best response algorithm take NA^{N-1} steps before reaching a NE. The game is built by induction on the number of players N.

With 1 player, the game takes one step (when the starting point is not a NE). With two players, it is possible to construct a game where the sequence of visited states is $(0,0),(1,0),(1,1)\cdots(i,i),(i+1,i),(i+1,i+1)\cdots(A-1,A-1),(0,A-1)$, by choosing the respective potentials of these states to be $0,1,\cdots(i,i)\to 2i,(i+1,i)\to 2i+1$ while the potential of all the other states is -1.

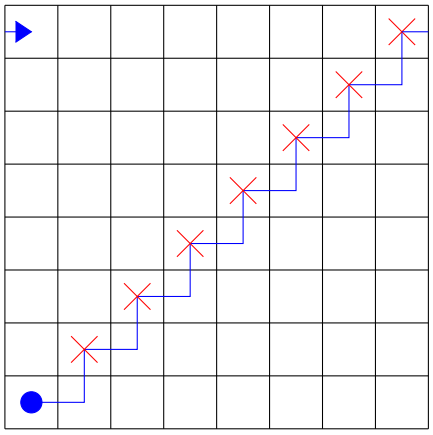


Figure 1: States visited by a round-robin sequence with two players.

This takes $2A$ steps before reaching a NE. For 3 players, let us assume that during the first $3A$ steps, the behavior is the same as for two players (with one additional dimension): on players 2 and 1 turns, they add 1 (mod A) to their strategy. The visited states are $(0,0,0),(1,0,0),(1,1,0),\cdots,(i,i,0),(i+1,i,0),(i+1,i+1,0),\ldots,(A-1,A-1,0),(0,A-1,0)$. In this sequence, at player 3's turns, namely in states of the form $(i,i,0)$, its best response is to stay in the same state (states marked with a red cross in Figure **??**. Once state $(0,A-1,0)$ is reached, player 2's best response is to stay still and player 3 prefers to move to its next state: $(0,A-1,1)$. From this point on, players 1, 2 and 3 repeat the same stair pattern as before. Note that the staircase is shifted by one unit to the right (mod (A,A)). The states where player 3 does not move are now of type $(i+1,i,1)\text{mod}(A,A)$. This construction is repeated with staircases of type $(i+j,i,j)$ for all $0\leqslant j\leqslant A-1$. The total number of steps is $3A^2$. Again, this can be obtained by choosing as potential -1 for all non visited states, and the order of visit for the visited states.

Let us generalize this construction for N players. During the first NA^{N-2} steps, the sequence of visited states is the same as with $N-1$ players. At each turn of player N, its strategies is to stay still. Once all these steps have been taken, The construction is shifted by one and repeated with sequences of type $(i_2+i_3+\cdots+i_{N-1}+j,i_2,\ldots,i_{N-1},j)$ mod $(A,\ldots,A)$. In total the number of steps is NA^{N-1}. Therefore the worst case complexity is bounded from below by NA^{N-1}.

Let us now show that this lower bound is tight. If the algorithm has not reached a NE then one player (say ℓ) has not played more than once in each "line" (set of states of type $(\cdot,x_{-\ell})$. There are A^{N-1} such lines and under round-robin revision sequences, player ℓ turns come every N steps, so that the number of steps before convergence is bounded from above by NA^{N-1}. □

B Intersection-Free Approximation, Proof of Lemma 1

Proof. Let us consider an infinite sequence $U = (U_1, U_2, \ldots)$ of independent random variables, uniformly distributed in $[0, 1]$. We couple an execution of the original BRA with an execution of IFA as follows. In both executions, each new potential used by the algorithm is the next value in the sequence U.

1. Under this coupling, we will show an almost sure comparison, $C_{BRA} \leq C_{IFA}$. If the t^{th} state visited by IFA is a Nash Equilibrium, this implies that all the values in the interval $(U(t+1), U(t+2), \ldots, U(t+(A-1)N)$ are below $U(t)$.
After $t-1$ comparisons, assume that BRA has not stopped yet. Then $U(t)$ is used while a player computes its best response. There are at most $A-2$ values left to be compared by the current player. The state obtained after this player has finished computing its best response is a Nash equilibrium if at most $N-1$ next players agree. In total they will explore less that $(A-1)(N-1)$ values in the sequence U. We already know that they will all be below $U(t)$.
So if the t^{th} value explored by IFA is a Nash equilibrium and if BRA did not find a Nash equilibrium before this instant, then the t^{th} comparison by BRA is on a Nash Equilibrium, thus $C_{BRA} \leq C_{IFA}$.

2. Now, let us consider the number of comparisons per step: IFA makes exactly $A-1$ comparisons to compute the best response **br** at step k, while BRA makes $A-1-I_k$ comparisons to do the same (where I_k is the number of intersections for the current step k).

So in total,

$$T_{BRA} = \frac{C_{BRA} + I}{A-1} \leq \frac{C_{IFA} + I}{A-1} = T_{IFA} + \frac{I}{A-1}, \tag{9}$$

where $I \stackrel{\text{def}}{=} I_1 + \cdots + I_{T_{BRA}}$ is the total number of intersections.

3. We will first show that the expected number of intersections in a trajectory containing T steps (with $T > N$) is upper-bounded by $\frac{(T-N)(T-N+1)}{2(A-1)^2 A^{N-2}}$. The potentials compared during one step form a one dimensional line in the state space $\mathcal{A}^N$. This line which can be defined by $N-1$ coordinates. There is one intersection when there exists a previously explored line with $N-2$ coordinates in common. During a trajectory following the round-robin scheme, each step replaces one coordinate of the current state with a new value, therefore, an intersection between two lines can only happen when the coordinates of both starting points differ according to the direction of the second line, meaning at least a full round (N steps) separated the lines. These allows us to see the second intersecting line as randomly chosen, independent of the first one.

All this implies that the probability that the line explored at step t creates an intersection with a previous one is either 0 (if t is too small) or $\frac{1}{A^{N-2}}$.

The expectation that a new intersection is formed at step t is $\frac{N-1}{N}(t-N)\frac{1}{A^{N-2}}$. Adding these expectations up to $t = T_{BRA}$, which is a stopping time of the process, yields by Wald formula

$$\begin{aligned}
\mathbb{E}I &\stackrel{\text{def}}{=} \mathbb{E}\sum_{t=1}^{T_{BRA}} I_t && (10)\\
&= \sum_{t=1}^{\mathbb{E}T_{BRA}} \frac{N-1}{N}(t-N)\frac{1}{A^{N-2}} \quad \text{(Wald Formula)} && (11)\\
&= \frac{N-1}{N}\frac{(\mathbb{E}T_{BRA}-N)(\mathbb{E}T_{BRA}-N+1)}{2A^{N-2}}. && (12)
\end{aligned}$$

By taking expectations in (**??**) we get

$$\mathbb{E}T_{BRA} \leqslant \mathbb{E}T_{IFA} + \frac{1}{A-1}\frac{N-1}{N}\frac{(\mathbb{E}T_{BRA}-N)(\mathbb{E}T_{BRA}-N+1)}{2A^{N-2}}.$$

At this point, we can solve this inequality by studying this polynomial in $\mathbb{E}T_{BRA}$. A more direct approach is to bound shamelessly T_{BRA} by C_{IFA}, to get

$$\begin{aligned}\mathbb{E}T_{BRA} &\leqslant \mathbb{E}T_{IFA} + \frac{1}{A-1}\frac{N-1}{N}\frac{(\mathbb{E}C_{IFA}-N)(\mathbb{E}C_{IFA}-N+1)}{2A^{N-2}}\\ &\leqslant \mathbb{E}T_{IFA} + \frac{e^{2\gamma}A^2N^2}{2A^{N-2}} + o(A^{-N+4}).\end{aligned}$$

This last inequality uses the bound on $\mathbb{E}C_{IFA}$ given in Equation (**??**) and ends the proof.

The reverse inequality follows from Markov's inequality. Let I be the number of intersections during the execution of BRA. By Markov's inequality, $\mathbb{E}[I] \geqslant \mathbb{P}(I \geqslant 1)$. Therefore,

$$\begin{aligned}\mathbb{E}[T_{BRA}] &= \mathbb{P}(I=0)\mathbb{E}[T_{BRA}|I=0] + \mathbb{P}(I\geqslant 1)\mathbb{E}[T_{BRA}|I\geqslant 1]\\ &\geqslant \mathbb{P}(I=0)\mathbb{E}[T_{BRA}|I=0]\\ &\geqslant (1-\mathbb{E}[I])\mathbb{E}[T_{BRA}|I=0]\\ &= (1-\mathbb{E}[I])\mathbb{E}[T_{IFA}].\end{aligned}$$

Now, $\mathbb{E}[I]\mathbb{E}[T_{IFA}]$ goes to zero when N goes to infinity (see (**??**)). The same derivation can be used for the number of comparisons.

4. We consider the previous sequence $(U_n)_{n\in\mathbb{N}}$ of potentials and we focus on the first T steps in both algorithms.
Let $(B_i)_{i\in\mathbb{N}}$ be the sequence defined by $B_i = 1 \Leftrightarrow U_i > \max_{j<i} U_j$. The proportion of 1's in B is decreasing.

The number M^T_{IFA} of moves for IFA in the first T steps is the number of blocks of size $A-1$ in B containing at least one 1, while the number of moves for BRA M^T_{BRA}, is the number of blocks containing at least one 1, of size $A-1$ minus the number of intersections. The probability of having at least a one in a block of B from i to $j-1$ is equal to $1-\frac{i}{j}$. In expectation, $\mathbb{E}[M^T_{IFA}] = \sum_{t=1}^{T}(1-\frac{t}{t+1})$ and $\mathbb{E}[M^T_{BRA}] = \sum_{t=1}^{T}(1-\frac{At-I_t}{A(t+1)-I_{t+1}})$ where I_t is the expected number of intersections before step t. From the previous calculation, $I_t = \frac{(t-N)(t-N+1)}{2(A-1)A^{N-2}} = O\left(\frac{t^2}{A^{N-1}}\right)$. Therefore, by replacing T by T_{BRA}, using Wald formula and bounding $\mathbb{E}[T_{BRA}]$ by $O(AN)$, we get $\mathbb{E}[M_{BRA}] = \mathbb{E}[M_{IFA}] + O\left(\frac{N^2}{A^N}(\mathbb{E}[T_{BRA}])^2\log(\mathbb{E}[T_{BRA}])\right)$ so that $\mathbb{E}[M_{BRA}] = \mathbb{E}[M_{IFA}] + o(1)$. □

C Optimality of Round-Robin, Proof of Lemma 2

Let us denote by $T_{BRA}(R)$ (resp. $T_{IFA}(R)$) the number of steps taken by BRA (resp. IFA) under revision function R, and drop the explicit mention of the dependence on the initial state, that does not play any role under our randomization.

We are going to show that

$$\begin{aligned}\mathbb{E}[T_{BRA}(\text{round-robin})] &\leqslant \mathbb{E}[T_{IFA}(\text{round robin})] + \epsilon_1(N) &(13)\\ &\leqslant \mathbb{E}[T_{IFA}(R)] + \epsilon_1(N) &(14)\\ &\leqslant \mathbb{E}[T_{BRA}(R)] + \epsilon_1(N) + \epsilon_2(N), &(15)\end{aligned}$$

where $\epsilon_1(N)$ and $\epsilon_2(N)$ go to 0 when N goes to infinity.

The first inequality (**??**) was proved in Lemma 1. The second one (**??**) comes from the following observation. Under IFA, the only difference between round-robin and any other revision sequence is when a player in L is chosen under R. This play is useless and never happens under round-robin. Therefore, for any revision sequence, R, $T_{IFA}(\text{round-robin}) \leqslant T_{IFA}(R)$ on all trajectories.

The last inequality (**??**) follows from Markov's inequality. Let I be the number of intersection during the execution of BRA under revision sequence R. By Markov's inequality, $\mathbb{E}[I] \geqslant \mathbb{P}(I \geqslant 1)$. Therefore,

$$\begin{aligned}\mathbb{E}[T_{BRA}(R)] &= \mathbb{P}(I=0)\mathbb{E}[T_{BRA}(R)|I=0] + \mathbb{P}(I \geqslant 1)\mathbb{E}[T_{BRA}(R)|I \geqslant 1]\\ &\geqslant \mathbb{P}(I=0)\mathbb{E}[T_{BRA}(R)|I=0]\\ &\geqslant (1-\mathbb{E}[I])\mathbb{E}[T_{BRA}(R)|I=0]\\ &= (1-\mathbb{E}[I])\mathbb{E}[T_{IFA}(R)].\end{aligned}$$

Now, $\mathbb{E}[I]\mathbb{E}[T_{IFA}(R)]$ goes to 0 when N goes to infinity (see Appendix **??**). This finishes the proof.

D Other Algorithms

This analysis of BRA leads naturally to consider variants of BRA that may improve on the convergence time. Let us consider the following variant of the BRA, called Full-BRA in the following.

- At each step *all* the players are active and they all compute their best response.
- The player whose best response has the largest potential wins and only this player actually plays its best response.

This algorithm should use less moves than BRA before convergence because, at each step, the increase of the potential is larger under Full-BRA.

Its average time complexity can be computed in the same way as before, and the computation is even simpler.

Theorem 4. *The number T_F of steps and the number of moves M_F, needed for the Full-BRA Algorithm, under IFA, to reach a NE are equal and constant on average: $\mathbb{E}[M_F] = \mathbb{E}[T_F] = e - 1 + o(1)$. The number of comparisons C_F before convergence is on average $\mathbb{E}[C_F] = e(A-1)(N-1) + o(AN)$.*

Before giving the proof of the theorem, let us comment on this result. It should be rather striking that the number of steps T_F does not depend on the number of players nor on the number of strategies per player. However, notice that its number of comparisons is larger than for BRA.

Proof. First, it should be clear by definition of Full-BRA that the the number of moves is equal to the number of steps. Indeed, a step that does not provoke a move implies that a NE has been found.

Let us consider now a new discrete time Markov chains $Y(t)$ whose state at step t is the potential of the profile reached by Full-BRA at step t, under the intersection-free approximation.

The transitions probabilities are the following (using $b \stackrel{\text{def}}{=} (A-1)(N-1)$):

$$\mathbb{P}(Y(t+1) \in [z,1] | Y(t) = y) = 1 - z^b, \quad z > y \tag{16}$$

$$\mathbb{P}(Y(t+1) = y | Y(t) = y) = y^b. \tag{17}$$

The average number of steps before convergence starting with a potential y, $S_F(y)$ satisfies the Poisson equation:

$$S_F(y) = \int_y^1 b u^{b-1} S_F(u) + 1) du,$$

with the boundary condition $S_F(0) = 1$.

By differentiating w.r.t. y, one gets $S'_F(y) = -by^{b-1}(S_F(y) - 1)$, whose solution is

$$S_F(y) = e^{y^b} - 1.$$

The expected number of steps is $\mathbb{E}T_F = \int_0^1 S_F(y)dy$. By replacing y by one in the integral, we get $\mathbb{E}[T_F] \leqslant e - 1$. More over, this bound is tight because for all y, $S_F(y)$ goes to $e-1$ when N or A go to infinity. Therefore,

$$\mathbb{E}[T_F] = e - 1 + o(1).$$

As for the average number of comparisons performed by the algorithm. It satisfies

$$\mathbb{E}[C_F] = (A-1)(N-1)(\mathbb{E}[T_F] + 1) = e(A-1)(N-1) + o(AN).$$

□

At this point, we have shown that BRA converges after $\log(N)$ moves while Full-BRA converges after a constant number of steps. However, Full-BRA makes more comparisons than BRA on average (eAN against $e^\gamma AN$, so approximately AN more comparisons for Full-BRA). This gives the impression that both algorithms are very efficient to compute NE in large games. However a natural question is whether this is simply because there are so many NE on average that any algorithm will be efficient. For that let us consider the following naive random algorithm (denoted rand) to find a NE:

- Pick a profile at random and check if this is a NE.
- Else, pick again.

Using the same approach, the average number of steps (or moves) T_{rand}, starting with potential y, of this random algorithm satisfies $T_{\text{rand}}(y) = (1 - y^c) \int_0^1 (M_{\text{rand}}(u) + 1) du$, with $c \stackrel{\text{def}}{=} (A-1)N$. The average number of steps is therefore

$$\mathbb{E}[T_{\text{rand}}] = \int_0^1 T_{\text{rand}}(y)dy = (1 + \mathbb{E}[T_{\text{rand}}]) \int_0^1 (1 - y^c)dy = c = (A-1)N.$$

This is linear in A and N, while the average number of comparisons can also be computed. Here, as soon as a state is found with a larger potential than the picked profile, this profile is rejected as not being a NE. Therefore, the number of comparisons per step is a random variable, smaller than $(A-1)N$. Its average value over all steps is

$$\mathbb{E}[C_{\text{rand}}] = \mathbb{E}[T_{\text{rand}}] \int_{y=0}^{1} \left(\sum_{i=0}^{c} y^i \right) dy = c \sum_{i=1}^{c} \frac{1}{i} = AN \log(AN) + O(AN).$$

So this simple algorithm takes more steps than the BR approaches in terms of number of comparisons. Also the quality of the NE (the potential of the NE) obtained by the BR approaches is better than for random NE. Let us assume that the potential Φ of all states is uniformly distributed in $[0,1]$ (unlike for the variables M, C and T, the potential of a NE depends on the distribution used to generate Φ).

In that case, the average potential of a random NE_{rand} is

$$\mathbb{E}[\Phi(\text{NE}_{\text{rand}})] = \int_0^1 (1-t^c)dt = \frac{c}{c+1} \approx 1 - \frac{1}{AN}.$$

Meanwhile, the average potential of a NE found by the BRA and Full-BRA is given by the following computation (only done for Full-BRA, but BRA is treated the same way).

Let us denote by $D_F(y,h)$ the probability that the potential of the NE found by Full-BR is larger than h, starting with a profile with potential y. $D_F(y,h)$ satisfies the Poisson equation,

$$D_F(y,h) = (1-h^b) + \int_y^h (b-1)u^b D_F(u,h)du.$$

Differentiating w.r.t. y yields

$$D_F(y,h) = (1-h^b)\exp(h^b - y^b).$$

The average potential of the Nash equilibrium reach by Full-BRA, (NE_F) is therefore

$$\mathbb{E}[\Phi(\text{NE}_F)] = \int_0^1 \left(y + \int_y^1 (1-h^b)\exp(h^b-y^b)dh \right) dy.$$

This is bounded from below by the average potential of a NE starting with potential 0. $\mathbb{E}[\Phi(\text{NE}_F)] \geqslant \int_0^1 (1-h^b)\exp(h^b)dh$.

Note that this compares favorably with the average potential of a random NE (the gap to 1 is at least twice as small) since by bounding $\exp(y)$ by $(1+y)$,

$$\mathbb{E}[\Phi(\text{NE}_F)] \geqslant 1 - \frac{1}{2b+1} \approx 1 - \frac{1}{2AN}.$$

E Numerical Experiments

The expected values computed in the previous section might only be theoretical results with little practical interest if the distribution of the execution time has a large variance.

To test this we have run several simulations of BRA over random potential games and we computed the standard deviations and confidence intervals. To run these experiments with a large number of players, we do not generate the potentials of all states (there are A^N states). Instead, we generate the potentials on the fly, only when they are needed. Hopefully, as suggested

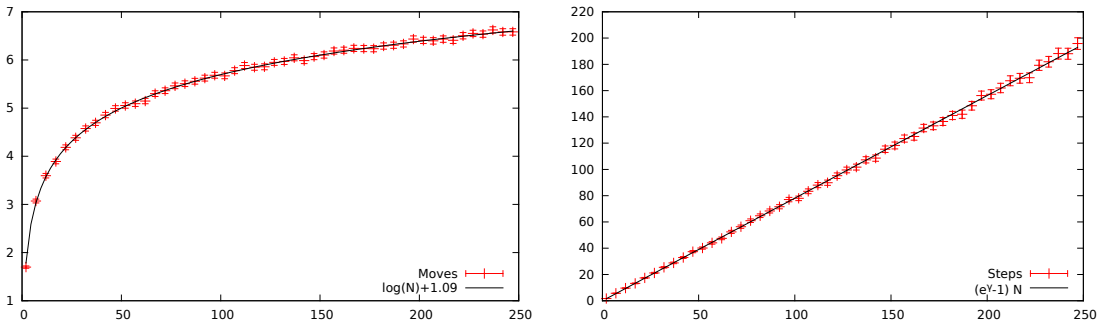
by Theorem **??**, BRA should converge fast, so without exploring too many states. Of course, the following simulations are exact runs of BRA and do not use the intersection-free approximation: the potentials of all visited states are kept in memory in case some states would be visited several times by the algorithm.

The following numerical experiments are also used to test our claim that while IFA is an upper-bound of BRA, this bound should be very tight. We compare the theoretical results obtained for IFA in Section 4 with the experimental performance of BRA obtained here.

Finally, the numerical experiments are also used to assess the performance of alternative revision sequences, such as Bernoulli sequences, instead of round-robin.

The following figures correspond to simulations on games, all with $A = 30$. The number of players N ranges from 0 to 250. All these games have potentials chosen uniformly and independently in $[0, 1]$. For each value of N, algorithm BRA is run 5000 times. The error-bars correspond to confidence interval at 95%.

E.1 Number of Moves, Number of Steps



(a) Average number of moves. The best fit function is $\log(N) + 1.09$

(b) Average number of steps before reaching a NE. The predicted value is $(e^\gamma - 1)N$.

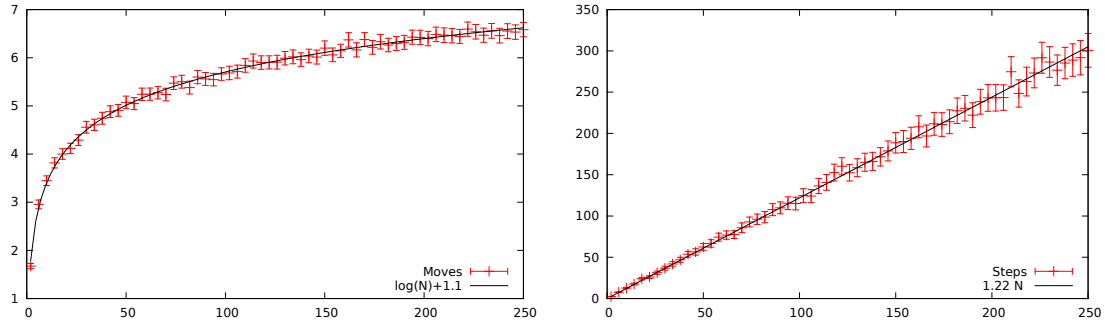
Figure 2: Simulation results for Round-Robin BRA

As one can see in Figure **??** the average number of moves for BRA with round-robin revisions is approximately $\log N + 1.09$, while Theorem **??** predicts that it should less than $\log N + e^\gamma$, with $e^\gamma \approx 1.7$. The confidence intervals are very small. This is a numerical evidence of the fact that the distribution of M_{BRA} is tight around its mean. Also, the confidence intervals do not grow with N, meaning that the variance barely depends on N.

Figure **??** displays the average number of steps for BRA under round-robin to reach an equilibrium. The best fit function is $(e^\gamma - 1)N$ while exactly N additional steps are needed to check that the current state is a NE, so that the total number of steps is $e^\gamma N$. This is exactly what Theorem **??** says. Here also, the very tight confidence intervals do not grow with N and provide numerical evidence that the variance is small (and the distribution is not spread).

E.2 Bernoulli Revision Sequences

We have run BRA with Bernoulli random revision sequences: at each step, the next player is chosen randomly, each with probability $1/N$.



(a) Average number of moves.
The best fit function is $\log(N) + 1.1$.

(b) Average number of steps before reaching a NE.
The best fit function is $1.22N$.

Figure 3: Simulation results for BRA, with a Bernoulli random revision sequence.

Its average performance has not been analyzed theoretically. It is rather direct to show that the average number of steps under a Bernoulli revision sequence can be bounded by the average number of steps under round-robin multiplied by $\log N$ by using a coupon collector argument: it takes on average $N \log N$ steps before all N players have played once. However this bound is obviously not tight.

The average number of steps for BRA with a random Bernoulli revision sequence before reaching an equilibrium is displayed in Figure **??**. The empirical best fit function is $1.22N$. The total number of steps is equal to this value plus the verification time for NE, the latter is given by the coupon collector, whose average duration is $N \log N + \gamma N + o(N)$. Therefore, the empirical number of steps is equal to $1.22N + (N \log N + \gamma N)$. This is to be compared with the linear behavior obtained for round-robin where the average number of steps can be decomposed into $(e^{\gamma} - 1)N + N \approx 0.78N + N$. Under round robin, a NE is reached on average when 78% of the players have played once. As for Bernoulli revisions, the simulation means that an equilibrium is reached on average before the size of the list L of satisfied players becomes bigger that $1 - e^{-1.22}N \approx 0.7N$.

In Figure **??**, the standard deviation increases with N. This growth comes from the fact that the variance also depends on the revision sequence. The variance of the last step (verification that the last state is indeed a NE) is not displayed in Figure **??**. This is simply the standard deviation of the coupon collector, known to be equal to $\frac{\pi N}{\sqrt{6}} + o(N)$. This brings an additional inefficiency in this case.

The average number of moves for BRA with a random revision sequence obtained by simulation is displayed in Figure **??**. With IFA, the number of moves is exactly the same under all revision sequences. As expected, without IFA, these simulations show that the expected number of moves is very close to the round-robin case, just slightly larger: best fit is $\log N + 1.1$ here instead of $\log N + 1.09$ for round-robin.

E.3 Simulations of Full-BRA

As seen in Figure **??**, the average number of steps for full-BRA remains constant, equal to e. Again the upper-bound predicted by Theorem **??** happens to be tight, and our claim about IFA and BRA being close is further confirmed. The number of comparisons is $e(A-1)N$ in this case, as under IFA. This confirms the fact that on average Full-BRA uses slightly more comparisons

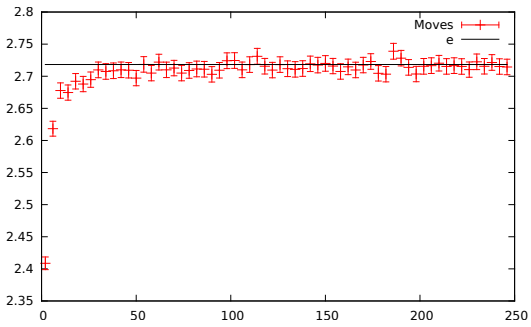


Figure 4: Average number of steps for full-BRA.

that BRA. Note that confidence intervals are even smaller (this could have been predicted since each step uses more random variables and some averaging takes place), and again, they do not grow with N.

F Tightness of the Bound on M_{IFA} in Equation (??)

The bound given in Equation (**??**) is tight when N goes to infinity, up to an additive constant.

In the derivation of (**??**), the only inequality is (**??**). Let us show this inequality can be bounded by a constant.

$$\left[\int_0^{1-\frac{1}{N}} \exp(\sum_{i=1}^{\infty}\frac{u^i}{i})du + \int_{1-\frac{1}{N}}^{1} \exp(\sum_{i=1}^{N-1}\frac{u^i}{i})du\right] - M(0,1), \tag{18}$$

$$= \left[\int_0^{1-\frac{1}{N}} \exp(\sum_{i=1}^{\infty}\frac{u^i}{i})du + \int_{1-\frac{1}{N}}^{1} \exp(\sum_{i=1}^{N-1}\frac{u^i}{i})du\right] - \int_0^{1} \exp(\sum_{i=0}^{N-1}\frac{u^i}{i})du, \tag{19}$$

$$= \int_0^{1-\frac{1}{N}} \left(\exp(\sum_{i=1}^{\infty}\frac{u^i}{i}) - \exp(\sum_{i=1}^{N-1}\frac{u^i}{i})\right) du, \tag{20}$$

$$\leqslant \int_0^{1-\frac{1}{N}} \exp\left(\sum_{i=1}^{\infty}\frac{u^i}{i}\right)\left(\sum_{i=N}^{\infty}\frac{u^i}{i}\right) du, \quad \text{[by convexity of exp]} \tag{21}$$

$$= \int_0^{1-\frac{1}{N}} \frac{1}{1-u}\int_0^{u}\frac{v^{N-1}}{1-v}dvdu, \tag{22}$$

$$= \int_0^{1-\frac{1}{N}} \frac{v^{N-1}}{1-v}\int_v^{1-\frac{1}{N}}\frac{1}{1-u}dudv, \tag{23}$$

$$= \int_0^{1-\frac{1}{N}} \frac{v^{N-1}}{1-v}(\log(N)+\log(1-v))dv, \tag{24}$$

$$= \sum_{k=0}^{N-2}\int_{k/N}^{(k+1)/N} \frac{v^{N-1}}{1-v}(\log(N)+\log(1-v))dv, \tag{25}$$

$$\leqslant \sum_{\ell=2}^{N}\frac{\log(\ell)}{\ell+1}\left(1-\frac{\ell-1}{N}\right)^{N-1}, \quad [\ell = N-k] \tag{26}$$

$$\leqslant \sum_{\ell=2}^{N}\frac{\log(\ell)}{\ell+1}\exp\left(-\ell+1+\frac{\ell-1}{N}\right) \tag{27}$$

$$\leqslant \sum_{\ell=2}^{\infty}\frac{\log(\ell)}{\ell+1}\exp\left(\frac{-\ell+1}{2}\right). \tag{28}$$

This series converges to a finite constant smaller than 0.39.

The inequality (**??**) comes from the fact that the positive function $f_1(v) := \frac{v^{N-1}}{1-v}$ is increasing in v while $f_2(v) := \log(N) + \log(1-v)$ is decreasing in v, and positive as long as $v < 1 - 1/N$. Therefore, an upper bound of the product $f_1(v)f_2(v)$, is obtained under the form of a staircase function with stairs of size $1/N$ by replacing the value of v by $\frac{k+1}{N}$ in f_1 and by $\frac{k}{N}$ in f_2. Namely, for all $\frac{k}{N} \leqslant v \leqslant \frac{k+1}{N}$, $f_1(v)f_2(v) \leqslant f_1(\frac{k+1}{N})f_2(\frac{k}{N})$.

Finally, the definition of M_{IFA} implies that $M(0,1) - 1 \leqslant \mathbb{E}M_{IFA} \leqslant M(0,1)$.

G Second Moments

In this part, we compute $\mathbb{E}[m^2]$ and $\mathbb{E}[\tau^2]$ where m and τ are the random variables denoting respectively the number of moves and the number of steps of IFA.

Using the same notations as in Section 4.4, the number of moves squared satisfies
$m^2(y,k) = m^2(y,k)$ w.p. $P((y,k),(y,k+1))$ and
$m^2(y,k) = (1+m(u,1))^2$ w.p. $P((y,k),(u,1))$.

Let $S(y,k) := \mathbb{E}m^2(y,k)$. The previous equalities imply that, using the first moment M,

$$S(y,k) = y^a S(y,k+1) + \int_y^1 au^{a-1}(S(u,1) + 2M(u,1) + 1)du.$$

Using the same approach as in Section 4.4, and the function $H(y) = 1 + \cdots + y^{a(N-2)}$ defined in Section 4.4, one gets

$$S(y,1) = G(y)H(y) \tag{29}$$

where $G(y) \stackrel{\text{def}}{=} \int_y^1 au^{a-1}(S(u,1) + 2M(u,1) + 1)du$. Its solution is

$$S(y,1) = \exp(-Q(y)) \int_y^1 au^{a-1}(2M(u,1) + 1)H(u)\exp(Q(u))du.$$

where Q is defined in (**??**). Since $M(0,1) = \log(N) + c + o(1/N)$, this implies that

$$\begin{aligned} S(0,1) &= (2\log(N) + 2c + o(N^{-1})) \left[\exp(-Q(y)) \int_y^1 au^{a-1}H(u)\exp(Q(u))du\right], \\ &= (2\log(N) + 2c + o(1/N))(\log(N)) + c + o(1/N)) \end{aligned}$$

because the term between brackets is exactly $M(0,1)$ (see Equation (**??**)). Therefore, the standard deviation of the number of moves is $\sqrt{S(0,1) - M(0,1)^2} = \log(N) + O(\sqrt{\log(N)})$.

The computation of the standard deviation of the number of steps follows the same path but is more cumbersome. It is easier to get a bound on the second moment. Let $U(y,k) \stackrel{\text{def}}{=} \mathbb{E}\tau^2(y,k)$. The one step analysis of the behavior of $\tau(y,k)$ yields the following formula for U, using the first moment T:

$$U(y,k) = y^a(U(y,k+1) + 2T(y,k+1) + 1) + \int_y^1 au^{a-1}(U(u,1) + 2T(u,1) + 1)du.$$

Using this recurrence equation and the recurrence equation satisfied by T, namely

$$T(y,k) = y^a(T(y,k+1) + 1) + \int_y^1 au^{a-1}(T(u,1) + 1)du,$$

as well as the inequality $T(y,k) \leqslant T(0,1) = e^\gamma Na + o(N)$, one gets

$$\begin{aligned} U(y,1) &\leqslant \left(y^a(2e^\gamma Na + 1) + o(N) + \int_y^1 au^{a-1}(U(u,1) + 1)du\right) H(y), \\ &\leqslant 2e^\gamma N^2 a + o(N^2). \end{aligned}$$

This gives a standard deviation $\sqrt{U(0,1) - T^2(0,1)} = O(N)$.

H Optimality of BRA, Proof of Theorem ??

The proof holds in several steps.

We first introduce a subclass of local search algorithms called *no-jump algorithms*. We will first prove that BRA is optimal within the class of no-jump algorithms before proving it for the entire class of local search algorithms.

Definition 3 (**No-Jump algorithm).** *A no-jump algorithm is a local search algorithm that only reads a potential vector starting from a state visited during the previous read.*

The profiles visited by a no-jump algorithm form a "walk" over the potential matrix along the dimensions of the matrix, with no "jumps" to unknown states. It should be clear that BRA is a no-jump algorithm.

Any no-jump algorithm can be written in the following form, similar to any local search up to the function J, replaced by a more constraint function W, that imposes the next state to belong to the newly explored states.

Algorithm 4: A general no-jump algorithm

Initial storage reduced to the initial profile: $\mathcal{H}_0 := \{(x(0), \Phi(x(0))\}$
repeat
  Select next player: $k := R(\mathcal{H}_t)$;
  Read payoff vector of k under current state $u_k(\cdot, x_{-k}(t))$;
  Store the new payoffs in memory: $\mathcal{H}_{t+1} := \mathcal{H}_t \cup \{((\alpha, x_{-k}(t)), u_k(\alpha, x_{-k}(t)))_{\alpha \in \mathcal{A}_k}\}$;
  walk to next state $x(t+1) := W(\mathcal{H}_{t+1})$; (in $\{(\alpha, x_{-k}(t))_{\alpha \in \mathcal{A}_k}\}$)
  Set $stop := 1$ if the current state is a NE
  $t := t + 1$;
until *stop*;

To compare two algorithms other potential games, we assume that the potentials are chosen independently and uniformly between 0 and 1. As shown in the randomization section 4.1, any uniform randomization over the potentials is equivalent to this one in terms of convergence time.

Let us consider one run of algorithm **??** starting with state $x(0) = (x_1(0), \ldots, x_N(0))$ and using the sequence R of players. After the first step (involving player, say ℓ), the new state becomes $x(1)$ where only the ℓ-th coordinate may have changed. At this point, we add a flag to all the visited states, of the form $(\alpha, x_{-\ell}(1))$, whose potential is smaller (or equal to) the potential of $x(1)$. The flagging goes on in the same manner with BRA. This brings the definition of *traces* that capture the whole history of the algorithm in a compact form.

Definition 4 (**Trace).**

- *A trace is the sequence S_i of the flagged states at each step of the algorithm, together with a distribution F of the potential of the current state x. Let $H(R, i, \mathfrak{G}, x^0) = (S_i, x, F)$ denote the trace on game $\mathfrak{G}$, starting in x^0, after the strategies of the first i plays in R. It should be clear that the past behavior of the algorithm up to step i is entirely characterized by its trace up to step i.*

- *Two traces (S, x, F) and (S', x', F') are comparable, denoted $(S, x, F) \prec (S', x', F')$ if they have the same final state, $(x = x')$ and the sequences of flagged states are included in one another ($\forall i, S_i \subset S'_i$) and the distribution of potential of the final states are comparable: $\forall u \in [0, 1], F(u) \leqslant F'(u)$. Two traces are also comparable when the final state of the second trace is a Nash equilibrium.*

The goal of this comparison is to determine which trace is "closer" to a NE.

Definition 5 (**Transform).** *A measure-preserving transform θ is a bijection on games that preserves the probability measure on games: If G is any measurable set of games (or equivalently for us a measurable set of potentials), then $\mathbb{P}(G) = \mathbb{P}(\theta(G))$.*

Definition 6 (**m-class**). *A set of games forms an m-class if they are equivalent up to step m: $\mathfrak{G}_1 \equiv_m \mathfrak{G}_2$ if there exists $x^0, x^{0'}$, and a transform θ such that $\forall i \leqslant m, H(R, i, \mathfrak{G}_1, x^0) = \theta H(R, i, \mathfrak{G}_2, x^{0'})$.*

Lemma 3. *Let us consider a revision sequence that used player ℓ at step m while the current step is $m' > m$, and two arbitrary distinct strategies of ℓ, namely α_1 and α_2.*

There exists a measure-preserving transform θ such that for any game $\mathfrak{G}$ and for any state z,

- *If $z_\ell \neq \alpha_1$ and $z_\ell \neq \alpha_2$, z is flagged in $\mathfrak{G}$ iff z is flagged in $\theta(\mathfrak{G})$.*
- *If $z_\ell = \alpha_1$, z is flagged before m in $\mathfrak{G}$ iff $(z_{-\ell}, \alpha_2)$ is flagged before m in $\theta(\mathfrak{G})$*
- *If $z_\ell = \alpha_2$, z is flagged before m in $\mathfrak{G}$ iff $(z_{-\ell}, \alpha_1)$ is flagged before m in $\theta(\mathfrak{G})$*
- *After m' steps, the current state in $\mathfrak{G}$ and $\theta(\mathfrak{G})$ is the same.*

In other words, the set of all the flagged states of the "hyperplane" of coordinates α_1 and α_2 at time m can be switched for the purpose of trace comparisons, after the next jump along ℓ.

Proof. Let us consider a game $\mathfrak{G}$ and let x^m be the current state at step m.
If $x_\ell^{m'} \neq \alpha_1$ and $x_\ell^{m'} \neq \alpha_2$, then we can construct a new game by permuting the coordinates α_1 and α_2 in ℓ. This transform is a valid candidate for θ. Since every change is a permutation of independent random variable, the measure is preserved.

If $x_\ell^{m'} = \alpha_1$. We build a new game by temporary adding a new coordinate along ℓ, assume it has never been visited before m and carry x^m. Every flagged state in this plane has been flagged by a line of the form $(z_{-\ell}, \cdot)$ and the corresponding point of any other plane, including $(\cdot, \alpha_1)$ is flagged as well, so the part of the trace between the jumps is valid. We exchange the plane $(\cdot, \alpha_1)$ and $(\cdot, \alpha_2)$, then bring back the new plane on α_1, flagging any point that was flagged in either $(\cdot, \alpha_1)$ or the new plane. This transformation preserves the potential and their comparisons, so it preserves the measure.

In all cases, we have built a new game with permutations of the potential, preserving the final state and exchanging two hyperplanes. □

We are now ready for the proof of the main lemma

Lemma 4. *Let us consider two traces H and H' that are comparable at step m: $H_m = (S_m, f, y) \prec (S'_m, f', y) = H'_m$. Then there exists a coupling such that they remain comparable steps by steps when both are completed using BRA, until convergence, of BRA on the two games.*

Proof. The proof holds by induction on the number of players.

With a single player a trace for BRA is either empty (already an equilibrium) or one step, with the case equilibrium being shorter than any other. If two trace are comparable, then after a step under BRA in both the final step only have flagged states as neighbor and they are still comparable.

Let us now assume that the property holds for $\ell - 1$ players. In the first m steps of sequence R, player ℓ appears k times. We use now an induction on k. If $k = 0$, we are back to the case with $\ell - 1$ players, where the property holds by assumption. Let us now assume the property holds for all sequence when player ℓ appears $k - 1$ times and consider its k-th appearance. We can use the recurrence hypothesis to build a transform θ up to this step. For any class of θ, we consider the coordinate chosen at this step.
We further constraint the transform θ by associating, whenever possible, the traces where the new coordinate along ℓ is the same. We know that the potential and flagged state are both higher

in the second trace, and that any choice not stay immobile nor flagged is equivalent, hence we can focus on the cases where the trace do not coincide, are of the form: new coordinate for the first trace, immobile for the second.

If both traces keep still, we can ignore this step, and use the second induction hypothesis (on k) on the sequence without this instance of ℓ. The result will be preserved when reinserting these two empty steps.

If both trace lead to new states, we can use the lemma 10 and a permutation to ensure that it is the same new states, regarding flagged state, hence we still have comparability. We can then use the first induction hypothesis on the sequence from this step to the step m to obtain the result.

If only the first trace stays still, we can use Lemma **??** between the planes of the two current states of the second trace in order to have comparable flagged states : We obtain a new pair of games in dimension $\ell - 1$, such that the set of flagged state of the first is included in the second, and the law of the potential of the first is below that of the second. We can apply the first induction hypothesis on the reduced game.

We can remark that at any point, the inclusion of the sets of flagged states impose that if the second trace starting point is a detected equilibrium, then so is the corresponding point of the first one.

In the second and third cases, we can apply the lemma on the induced game and obtain coupling where either the first trace has converged globally before the second, the first converged and the second continue, or both continued and the induction assumptions are preserved.

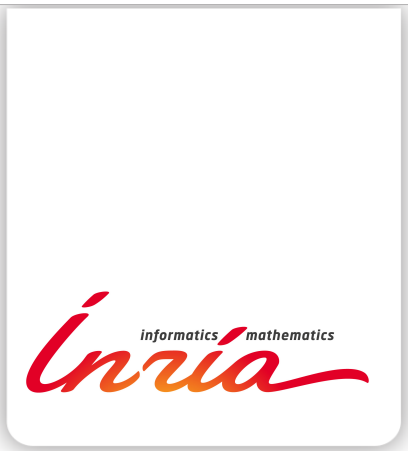
□

Proof of Theorem ?? for no-jump algorithms.
Let us consider two algorithms. The first one uses the choice function W up to step $m + 1$ and the best response function **br** for the remaining steps. The second one uses the choice function W up to step m and the best response function **br** for the remaining steps.

From step $m + 1$ on, the comparison between the algorithms can be seen as a comparison between two ordered traces, under BRA. After step m, the traces will remain comparable by Lemma **??**. The first one being smaller than the second one (sequence of flagged states as well as potential of the end point). The previous lemma **??** says both traces will remain comparable, using the appropriate transform, under BRA. This implies that the second trace will reach a local maximum before the first one.

Proof of Theorem ?? for all local search algorithms.

Consider any local search algorithm. By modeling the payoff vectors as vertices of a graph with arcs when the vertices directed by the order of visit, we obtain a graph that we can divide and cover by no-jump algorithms. The algorithm can then be seen as a group of no-jump algorithm that may sometimes merge or fork. In case of fork the new trace can always improved (in terms of trace comparison) by starting at the last point of the first one instead of in the middle. This means that all the no-jump algorithms composing the local search algorithm can be compared trace-wise with a single one.



RESEARCH CENTRE
GRENOBLE – RHÔNE-ALPES

Inovallée
655 avenue de l'Europe Montbonnot
38334 Saint Ismier Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399