# Double Oracle Algorithm for Computing Equilibria in Continuous Games

**Lukáš Adam, Rostislav Horčík, Tomáš Kasl, Tomáš Kroupa**
Artificial Intelligence Center, Faculty of Electrical Engineering, Czech Technical University in Prague
{adamluk3, xhorcik, kasltoma, tomas.kroupa}@fel.cvut.cz

### Abstract

Many efficient algorithms have been designed to recover Nash equilibria of various classes of finite games. Special classes of continuous games with infinite strategy spaces, such as polynomial games, can be solved by semidefinite programming. In general, however, continuous games are not directly amenable to computational procedures. In this contribution, we develop an iterative strategy generation technique for finding a Nash equilibrium in a whole class of continuous two-person zero-sum games with compact strategy sets. The procedure, which is called the double oracle algorithm, has been successfully applied to large finite games in the past. We prove the convergence of the double oracle algorithm to a Nash equilibrium. Moreover, the algorithm is guaranteed to recover an approximate equilibrium in finitely-many steps. Our numerical experiments show that it outperforms fictitious play on several examples of games appearing in the literature. In particular, we provide a detailed analysis of experiments with a version of the continuous Colonel Blotto game.

## Introduction

Action spaces of games appearing in AI applications are often prohibitively large. Consequently, one has to strive for efficiently computable approximations of equilibria, possibly with provable bounds on convergence rates (Gilpin, Peña, and Sandholm 2012). A number of algorithms applied in AI like regret matching (Hart and Mas-Colell 2000), the double oracle algorithm (McMahan, Gordon, and Blum 2003) or the policy-space response oracle (Lanctot et al. 2017; Muller et al. 2019) overcome the problem with the cardinality by selecting 'good' strategies iteratively. The recent advances in algorithmic game theory have led to the development of algorithms for approximately solving extremely large finite games, such as variants of poker (Moravčík et al. 2017; Brown and Sandholm 2019) or multidimensional resource allocation problems (Behnezhad et al. 2017).

Completely new problems arise from considering games with infinite strategy spaces, in which the strategies are vectors of real numbers corresponding to physical parameters (Archibald and Shoham 2009) or to the setting of classifiers (Yasodharan and Loiseau 2019). The first theoretical obstacle is that the existence of mixed strategy equilibria is guaranteed only for infinite games whose utility functions satisfy additional conditions such as continuity (Glicksberg 1952; Fan 1952). On top of that, some well-understood classes of infinite games have only optimal strategies with uncountable supports; see (Roberson 2006) for an in-depth discussion of infinite Colonel Blotto games.

Computational procedures for finding (approximate) equilibria of infinite games exist for rather special kinds of utility functions. Two-person zero-sum polynomial games are solvable by semidefinite programming (Parrilo 2006; Laraki and Lasserre 2012). Games with piecewise-linear utility functions and their equilibria were analyzed in (Kroupa and Majer 2014). Approximate equilibria of separable games can be computed under particular assumptions (Stein, Ozdaglar, and Parrilo 2008). For some games the best response can be approximated by neural nets (Kamra et al. 2018, 2019). One of the important iterative procedures for finite games, Brown-Robinson learning process known as fictitious play (Brown 1951; Robinson 1951), has been recently applied to infinite games (Ganzfried 2020). However, the dynamics of best response strategies generated by fictitious play was analyzed only in special cases; cf. (Hofbauer and Sorin 2006; Perkins and Leslie 2014). To the best of our knowledge, not much is known about the convergence of fictitious play for general zero-sum continuous games as defined below.

This paper deals with continuous games, which we define as two-person zero-sum games with continuous utility functions over compact strategy sets. Finding equilibria in such games is a much harder problem than solving finite games. Computational techniques exist only for special utility functions mentioned above (polynomial, convex/concave, separable, etc.) In this paper we propose an iterative algorithm that can be applied to all continuous games. Namely we extend the double oracle algorithm (McMahan, Gordon, and Blum 2003) to such games. This algorithm is an iterative strategy generation technique based on (i) the solution of subgames by LP solvers and (ii) the expansion of subgames' strategy sets using the best response strategies obtained thus far. Our main result, Theorem 1, is the convergence of this algorithm for any continuous game. We point out that this is a non-trivial generalization of Theorem 1 from (McMahan, Gordon, and Blum 2003). The crucial issues to be overcome are the following.

- Computing equilibria in continuous games is inherently an infinite-dimensional optimization problem.
- Equilibria in continuous games may be mixed strategies with uncountable supports, hence not computable structures.
- The convergence of the algorithm and the approximation of equilibria necessarily involve weak convergence in the spaces of mixed strategies.

We discuss the results of numerical experiments, which show that the double oracle algorithm converges faster than fictitious play on several examples (polynomial game, Townsend function, and a version of the Colonel Blotto game). Our experiments' codes are available at `https://github.com/sadda/Double_Oracle`.

## Basic Notions

This section summarizes basic notions and results related to continuous zero-sum games and their equilibria; see (Karlin 1959) or (Stein, Ozdaglar, and Parrilo 2008) for details.

### Continuous Games

Player 1 and Player 2 select strategies from nonempty compact sets $X \subseteq \mathbb{R}^m$ and $Y \subseteq \mathbb{R}^n$, respectively. The utility function of Player 1 is a continuous function $u\colon X \times Y \to \mathbb{R}$. The utility function of Player 2 is $-u$. The triple $\mathcal{G} = (X, Y, u)$ is called a *continuous game.* Note that some authors use the term 'continuous game' in a somewhat different sense, allowing utility functions to be discontinuous functions over metric spaces of strategies.

A continuous game $\mathcal{G}$ is a *finite game* if $X$ and $Y$ are finite, and $\mathcal{G}$ is an *infinite game* otherwise. We will need the notion of subgame. When $X' \subseteq X$ and $Y' \subseteq Y$ are nonempty compact sets, we define the *subgame* $\mathcal{G}' = (X', Y', u)$ of $\mathcal{G}$ by the restriction of $u$ to $X' \times Y'$, which is denoted by the same letter.

The concept of mixed strategy in continuous games allows a player to randomize with respect to any probability measure. We will spell out the definitions related to mixed strategies only for Player 1. Their counterparts for Player 2 are completely analogous. A *mixed strategy* of Player 1 is a Borel probability measure $p$ over $X$. The set of all mixed strategies of Player 1 is denoted by $\Delta_X$. The *support* of a mixed strategy $p \in \Delta_X$ is the set

$$\operatorname{spt} p \coloneqq \bigcap \{K \subseteq X \mid K \text{ compact}, \ p(K) = 1\}.$$

Every mixed strategy $p \in \Delta_X$ can be classified as one of the following types depending on the size of its support.

1. *Pure strategy* $p$. This means that $\operatorname{spt} p = \{x\}$ for some $x \in X$. Equivalently, $p$ is equal to Dirac measure $\delta_x$.
2. *Finitely-supported mixed strategy* $p$. The support $\operatorname{spt} p$ is finite. Hence, $p$ can be written as a convex combination
$$p = \sum_{x \in \operatorname{spt} p} p(x) \cdot \delta_x.$$
3. *Mixed strategy* $p$ *with infinite support* $\operatorname{spt} p$.

Put $\Delta \coloneqq \Delta_X \times \Delta_Y$. If players implement a mixed strategy profile $(p, q) \in \Delta$, the expected utility of Player 1 is

$$U(p, q) \coloneqq \int_{X \times Y} u(x, y) \, \mathrm{d}(p \times q). \tag{1}$$

This yields a function $U\colon \Delta \to \mathbb{R}$, which can be effectively evaluated in important special cases. For example, when both $\operatorname{spt} p$ and $\operatorname{spt} q$ are finite,

$$U(p, q) = \sum_{x \in \operatorname{spt} p} \ \sum_{y \in \operatorname{spt} q} p(x) \cdot q(y) \cdot u(x, y).$$

If Player 1 employs a pure strategy given by $x \in X$ and Player 2 uses a mixed strategy $q \in \Delta_Y$, we will use the short notation $U(x, q) \coloneqq U(\delta_x, q)$.

### Equlibria in Continuous Games

A mixed strategy profile $(p^*, q^*) \in \Delta$ is an *equilibrium* in a continuous game $\mathcal{G}$ if, for all $(p, q) \in \Delta$,

$$U(p, q^*) \leq U(p^*, q^*) \leq U(p^*, q). \tag{2}$$

Every continuous game has at least one equilibrium; see (Glicksberg 1952). Define the *lower/upper value* of $\mathcal{G}$ by

$$\underline{v}(\mathcal{G}) \coloneqq \max_{p \in \Delta_X} \min_{q \in \Delta_Y} U(p, q) \quad \text{and}$$
$$\overline{v}(\mathcal{G}) \coloneqq \min_{q \in \Delta_Y} \max_{p \in \Delta_X} U(p, q).$$

Proposition 1 gives several conditions for equilibrium, which will be used throughout the paper without further references. Its proof is omitted since it is completely analogous to the case of finite games.

**Proposition 1.** *Let $\mathcal{G} = (X, Y, u)$ be a continuous game and $(p^*, q^*) \in \Delta$. The following assertions are equivalent.*

1. *The strategy profile $(p^*, q^*)$ is an equilibrium.*
2. *$U(x, q^*) \leq U(p^*, q^*) \leq U(p^*, y)$, for all $(x, y) \in X \times Y$.*
3. *$\min_{y \in Y} U(p^*, y) = \underline{v}(\mathcal{G})$ and $\max_{x \in X} U(x, q^*) = \overline{v}(\mathcal{G})$.*
4. *$\underline{v}(\mathcal{G}) = U(p^*, q^*) = \overline{v}(\mathcal{G})$.*

Hence, the equality $\underline{v}(\mathcal{G}) = \overline{v}(\mathcal{G})$ holds for every continuous game $\mathcal{G}$, and $v(\mathcal{G}) \coloneqq \underline{v}(\mathcal{G})$ is called the *value* of $\mathcal{G}$.

Bounds on the size of supports of equilibrium strategies are known only for particular classes of continuous games; see (Stein, Ozdaglar, and Parrilo 2008). There are examples of games whose equilibria are almost any sets of finitely-supported mixed strategies (Rehbeck 2018). Moreover, some continuous games possess only equilibria with uncountable supports (Roberson 2006).

In many applications it is enough to find an *$\epsilon$-equilibrium* $(p^*, q^*)$ for some $\epsilon \geq 0$, that is,

$$U(p, q^*) - \epsilon \leq U(p^*, q^*) \leq U(p^*, q) + \epsilon \tag{3}$$

for all $(p, q) \in \Delta$. Note that this is a natural extension of (2). According to Proposition 2, whose proof is in Appendix, we can always recover an approximate equilibrium $(p^*, q^*)$ with finite supports and such that $U(p^*, q^*)$ is arbitrarily close to the value of game $v(\mathcal{G})$.

**Proposition 2.** *Let $\mathcal{G}$ be continuous game and let $\epsilon > 0$.*

- *There exists an $\epsilon$-equilibrium $(p^*, q^*)$ of $\mathcal{G}$ such that both $\operatorname{spt} p^*$ and $\operatorname{spt} q^*$ are finite.*
- *Every $\epsilon$-equilibrium $(p^*, q^*)$ of $\mathcal{G}$ satisfies the inequality $|U(p^*, q^*) - v(\mathcal{G})| \leq \epsilon$.*

**Algorithm 1** Double Oracle Algorithm

**Input:** Continuous game $\mathcal{G} = (X, Y, u)$, nonempty finite subsets $X_1 \subseteq X$, $Y_1 \subseteq Y$, and $\epsilon \geq 0$

Let $i := 0$
**repeat**
    Increase $i$ by one
    Find an equilibrium $(p_i^*, q_i^*)$ of subgame $(X_i, Y_i, u)$
    Find some $x_{i+1} \in \beta_1(q_i^*)$ and $y_{i+1} \in \beta_2(p_i^*)$
    Let $X_{i+1} := X_i \cup \{x_{i+1}\}$ and $Y_{i+1} := Y_i \cup \{y_{i+1}\}$
    Let $\underline{v}_i := U(p_i^*, y_{i+1})$ and $\overline{v}_i := U(x_{i+1}, q_i^*)$
**until** $\overline{v}_i - \underline{v}_i \leq \epsilon$

**Output:** $\epsilon$-equilibrium $(p_i^*, q_i^*)$ of game $\mathcal{G}$

## Double Oracle Algorithm

The double oracle algorithm uses the notion of best response strategies. For every mixed strategy $q \in \Delta_Y$ of Player 2, the *best response set* of Player 1 is

$$\beta_1(q) := \left\{ x \in X \mid U(x, q) = \max_{x' \in X} U(x', q) \right\}.$$

Analogously, for any $p \in \Delta_X$, let

$$\beta_2(p) := \left\{ y \in Y \mid U(p, y) = \min_{y' \in Y} U(p, y') \right\}.$$

Note that best response strategies are defined to be pure, without any loss of generality; see Proposition 4. Moreover, by compactness and continuity, $\beta_1(q)$ and $\beta_2(p)$ are always nonempty compact sets.

The double oracle algorithm proceeds as follows. In every iteration $i$, finite strategy sets $X_i$ and $Y_i$ are constructed, and an equilibrium $(p_i^*, q_i^*)$ of finite subgame $(X_i, Y_i, u)$ is computed by the linear programming. The best responses $x_{i+1}$ and $y_{i+1}$ to $q_i^*$ and $p_i^*$, respectively, are found, and then added to the strategy sets $X_i$ and $Y_i$. This is repeated until the terminating condition $U(x_{i+1}, q_i^*) - U(p_i^*, y_{i+1}) \leq \epsilon$ is satisfied. The resulting strategy profile is guaranteed to be an $\epsilon$-equilibrium.

The main result of this paper is the convergence of Algorithm 1. This result generalizes Theorem 1 from (McMahan, Gordon, and Blum 2003), which applies only to finite games. We will first discuss several important properties of the algorithm. At each step $i$ we have

$$\underline{v}_i \leq U(p_i^*, q_i^*) \leq \overline{v}_i, \tag{4}$$

which follows from

$$U(p_i^*, q_i^*) = \max_{x \in X_i} U(x, q_i^*) \leq \max_{x \in X} U(x, q_i^*) = \overline{v}_i,$$

and similarly from the analogous inequality for $\underline{v}_i$. The same bounds hold even for the value of game by Lemma 2:

$$\underline{v}_i \leq v(\mathcal{G}) \leq \overline{v}_i.$$

The standard stopping condition of the double oracle algorithm for finite games is $X_{i+1} = X_i$ and $Y_{i+1} = Y_i$. Herein we use the terminating condition $\overline{v}_i - \underline{v}_i \leq \epsilon$ for two main reasons.

- It is more general. Lemma 1 says that if $X_{i+1} = X_i$ and $Y_{i+1} = Y_i$, then $\overline{v}_i - \underline{v}_i = 0$.
- It enables us to control the quality of approximation. Assume that the algorithm terminates at step $i$. Then $(p_i^*, q_i^*)$ is an $\epsilon$-equilibrium by Theorem 1 and Proposition 2 guarantees that $v(\mathcal{G})$ is known precisely up to $\epsilon$.

**Theorem 1.** *Let $\mathcal{G} = (X, Y, u)$ be a continuous game.*

1. *If $\mathcal{G}$ is a finite game and $\epsilon = 0$, Algorithm 1 converges to an equilibrium of $\mathcal{G}$ in finitely-many iterations.*
2. *If $\mathcal{G}$ is an infinite game and $\epsilon = 0$, then any weakly convergent subsequence of Algorithm 1 converges to an equilibrium of $\mathcal{G}$ in a possibly infinite number of iterations. Moreover, such a weakly convergent subsequence exists.*
3. *If $\mathcal{G}$ is an infinite game and $\epsilon > 0$, Algorithm 1 converges to a finitely supported $\epsilon$-equilibrium of $\mathcal{G}$ in finitely-many iterations.*

*Proof.* Let $\epsilon \geq 0$. Assume that the terminating condition

$$\overline{v}_i - \underline{v}_i = U(x_{i+1}, q_i^*) - U(p_i^*, y_{i+1}) \leq \epsilon$$

is satisfied. We will show that $(p_i^*, q_i^*)$ is an $\epsilon$-equilibrium of $\mathcal{G}$. For every $p' \in \Delta_X$,

$$U(p', q_i^*) - \epsilon \leq \max_{p \in \Delta_X} U(p, q_i^*) - \epsilon = \max_{x \in X} U(x, q_i^*) - \epsilon$$
$$= U(x_{i+1}, q_i^*) - \epsilon \leq U(p_i^*, y_{i+1}) \leq U(p_i^*, q_i^*),$$

where the second relation follows from Proposition 4 (see Appendix), the third and the fifth from the definition of best response, and the fourth inequality from the terminating condition. We can derive a similar inequality for Player 2. This proves that $(p_i^*, q_i^*)$ is an $\epsilon$-equilibrium of $\mathcal{G}$. Note that for $\epsilon = 0$, this means that $(p_i^*, q_i^*)$ is an equilibrium of $\mathcal{G}$.

*Item 1.* Let $\mathcal{G}$ be a finite game and $\epsilon = 0$. After finitely-many iterations, necessarily $X_{i+1} = X_i$ and $Y_{i+1} = Y_i$. Lemma 1 implies that the terminating condition of Algorithm 1 is satisfied with $\epsilon = 0$ and the first paragraph of this proof implies that $(p_i^*, q_i^*)$ is an equilibrium of $\mathcal{G}$.

*Item 2.* Let $\mathcal{G}$ be an infinite game and $\epsilon = 0$. If Algorithm 1 terminates at step $i$, then the first paragraph implies that $(p_i^*, q_i^*)$ is an equilibrium of $\mathcal{G}$. In the opposite case, the algorithm generates an infinite sequence $(p_1^*, q_1^*), (p_2^*, q_2^*), \ldots$ Consider any weakly convergent subsequence[1] of this sequence. Without loss of generality, this and all other subsequences in this proof will be denoted with the same indices as the original sequences. Therefore, $p_i^* \Rightarrow p^*$ for some $p^* \in \Delta_X$ and $q_i^* \Rightarrow q^*$ for some $q^* \in \Delta_Y$, where the symbol $\Rightarrow$ denotes the weak convergence (see Appendix). We need to show that $(p^*, q^*)$ is an equilibrium of $\mathcal{G}$. First, assume that $y \in \bigcup_{i=1}^{\infty} Y_i$. Then $y \in Y_{i_0}$ for some $i_0$, hence $y \in Y_i$ for each $i \geq i_0$. Since $(p_i^*, q_i^*)$ is an equilibrium of subgame $(X_i, Y_i, u)$, we get

$$U(p_i^*, q_i^*) \leq U(p_i^*, y) \to U(p^*, y),$$

where the convergence follows from (11). Since $U(p_i^*, q_i^*) \to U(p^*, q^*)$ due to (10), this implies

$$U(p^*, q^*) \leq U(p^*, y). \tag{5}$$

[1] At least one such subsequence exists by Proposition 3.

The previous inequality holds for all $y \in \mathrm{cl}(\bigcup_{i=1}^{\infty} Y_i)$, by continuity of $U$.

Fix now an arbitrary $y \in Y$. The definition of $y_{i+1}$ yields

$$U(p_i^*, y_{i+1}) \le U(p_i^*, y) \to U(p^*, y), \tag{6}$$

where the limit holds due to (11). Since $y_i \in Y_i$, compactness of $Y$ provides a convergent subsequence $y_i \to \hat{y}$ such that $\hat{y} \in \mathrm{cl}(\bigcup_{i=1}^{\infty} Y_i)$. This allows us to use (5) to obtain

$$U(p_i^*, y_{i+1}) \to U(p^*, \hat{y}) \ge U(p^*, q^*). \tag{7}$$

Combining (6) and (7) yields $U(p^*, q^*) \le U(p^*, y)$.

Similarly, we get $U(x, q^*) \le U(p^*, q^*)$ for all $x \in X$. Hence, $(p^*, q^*)$ is an equilibrium of $\mathcal{G}$ by Proposition 1.

*Item 3.* As in item 2., we consider the subsequence $y_i \to \hat{y}$ such that $\hat{y} \in \mathrm{cl}(\bigcup_{i=1}^{\infty} Y_i)$. Then from (7) and from the inequality $U(p_i^*, y_{i+1}) \le U(p_i^*, q_i^*)$, we derive $U(p_i^*, y_{i+1}) \to U(p^*, q^*)$. Completely analogously, we get $U(x_{i+1}, q_i^*) \to U(p^*, q^*)$. Then

$$U(x_{i+1}, q_i^*) - U(p_i^*, y_{i+1}) \to 0.$$

This means that for any $\epsilon > 0$, there exists some $j$ such that

$$U(x_{i+1}, q_i^*) - U(p_i^*, y_{i+1}) \le \epsilon, \qquad \text{for all } i \ge j.$$

Therefore, the terminating condition is satisfied in the $j$-th step of Algorithm 1. It follows from the first paragraph of this proof that $(p_j^*, q_j^*)$ is a finitely-supported $\epsilon$-equilibrium of $\mathcal{G}$. □

Since best response strategies are not necessarily unique, the sequence generated by Algorithm 1 may fail to converge for some continuous game. Hence, it is inevitable to consider a weakly convergent subsequence of iterates in Theorem 1. Such a continuous game is shown in Example 1. Another feature of the double oracle algorithm is that the sequence $\overline{v}_1 - \underline{v}_1, \overline{v}_2 - \underline{v}_2, \ldots$ has nonnegative terms and converges to zero, but it is not necessarily monotone. This behavior can be demonstrated even for some finite games.

## Numerical Experiments

We present two classes of games. The first class contains one-dimensional strategy spaces and the second class consists of certain Colonel Blotto games. The equilibrium of each finite subgame is found by solving a linear program. The best responses were computed by selecting the best point of a uniform discretization for the one-dimensional problems and by using a mixed-integer linear programming reformulation for the Colonel Blotto games. The examples were implemented in Python with solvers `scipy.optimize` and `mip`. All computations were performed on a laptop with Intel Core i5 CPU and 8GB RAM and no GPU was involved. Randomness is present only in the initialization of one-dimensional examples when a random pair of pure strategies is found.

We compare the double oracle algorithm with fictitious play. Its extension from finite to infinite games was recently formulated in (Ganzfried 2020).

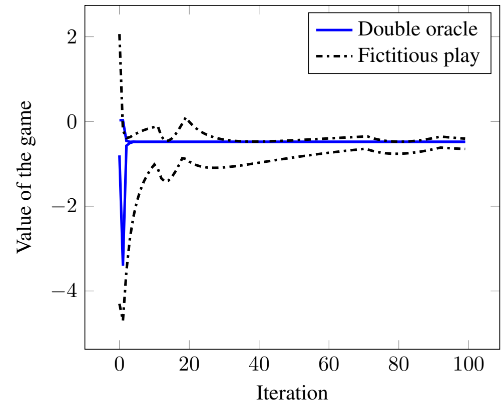


Figure 1: Convergence to the value of game $\mathcal{G}_1$

### One-dimensional Examples

We consider a polynomial game $\mathcal{G}_1$ from (Parrilo 2006) with the strategy spaces $X = Y = [-1, 1]$ and the utility function

$$u_1(x, y) = 5xy - 2x^2 - 2xy^2 - y.$$

In the equilibrium, Player 1 has the pure strategy $x^* = 0.2$ and Player 2 has the mixed strategy $q^* = 0.78\delta_1 + 0.22\delta_{-1}$. The value of game is $-0.48$. Figure 1 shows the convergence of upper/lower estimates of the value of game. Note that the fictitious play is much slower to converge than the double oracle algorithm.

The utility function $u_2$ in our second example (game $\mathcal{G}_2$) is based on (Townsend 2014). Specifically,

$$u_2(x, y) = -\cos^2((x - 0.1)y) - x\sin(3x + y)$$

is defined on $X = [-2.25, 2.5]$ and $Y = [-2.5, 1.75]$; see Figure 2. The convergence to the value is depicted on Figure 3. Once again the double oracle algorithm converges fast, while fictitious play is rather slow to converge. In Figure 4 we show the optimal strategies of Player 1. The double oracle algorithm converged to a mixed strategy supported by four points, the fictitious play seems to reach in limit a continuous distribution whose peaks are those points. Note that the vertical axis is rescaled to account for the difference between discrete and continuous distributions.

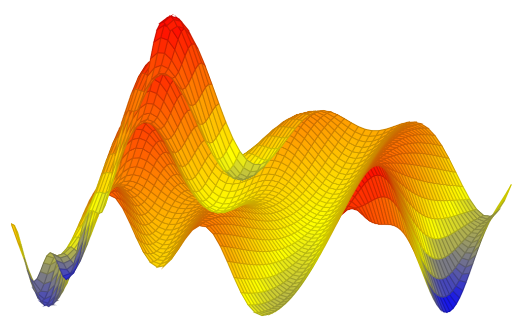

Figure 2: Townsend function $u_2$

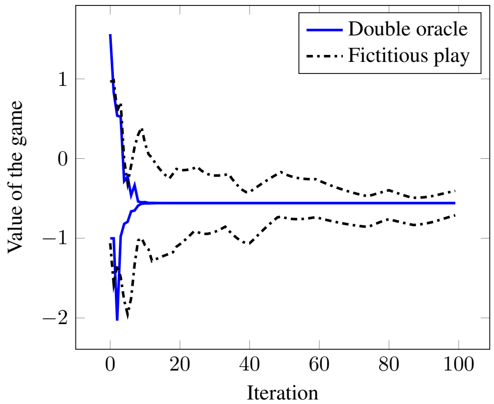


Figure 3: Convergence to the value of game $\mathcal{G}_2$

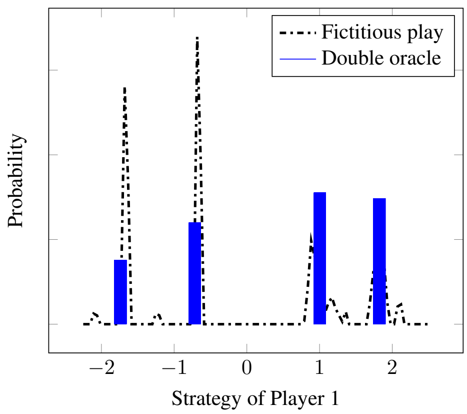


Figure 4: Mixed strategies in game $\mathcal{G}_2$

## Colonel Blotto Game

We consider a continuous variant of the Colonel Blotto game. Two players simultaneously allocate forces across $n$ battlefields. Both strategy spaces $X$ and $Y$ equal to

$$\left\{ \boldsymbol{x} := (x^1, \dots, x^n) \in \mathbb{R}^n_+ \mid x^1 + \dots + x^n = 1 \right\}.$$

The utility function of Player 1,

$$u(\boldsymbol{x}, \boldsymbol{y}) := \sum_{j=1}^n a^j \cdot l(x^j - y^j),$$

captures the total excess of the first army over the second army. The result on a battlefield $j$ is $a^j \cdot l(x^j - y^j)$, where $a^j > 0$ is a weight of battlefield $j$ and $l(x^j - y^j)$ measures the performance of the first army on a battlefield $j$. The standard choice is the signum function $l(z) = \operatorname{sgn}(z)$; see (Gross and Wagner 1950) or (Roberson 2006). This paper assumes that each player must allocate a sufficiently higher proportion of forces than the opponent to win the battle on a single battlefield. Specifically, we consider

$$l(z) = \begin{cases} -1 & \text{if } z \le -c, \\ \frac{1}{c} z & \text{if } z \in [-c, c], \\ 1 & \text{if } z \ge c, \end{cases} \qquad \text{for some } c > 0. \tag{8}$$

When $c \to 0$, we recover the classical infinite colonel Blotto game since (8) approaches $\operatorname{sgn}(z)$ in the limit.

We will show how to compute best response strategies in case of (8). Assume that Player 2 employs strategies $(\boldsymbol{y}_1, \dots, \boldsymbol{y}_k)$ with probabilities $(q_1, \dots, q_k)$, where $\boldsymbol{y}_i := (y_i^1, \dots, y_i^n) \in Y$. Then any best response strategy of Player 1 is a solution to

$$\max_{\boldsymbol{x} \in X} \sum_{i=1}^k q_i \sum_{j=1}^n a^j \cdot l(x^j - y_i^j). \tag{9}$$

Since $l$ is a piecewise affine function, this nonlinear optimization problem can be reformulated as a mixed-integer linear problem. In Appendix we derive its equivalent form

$$\begin{aligned} \max_{\boldsymbol{x},\boldsymbol{s},\boldsymbol{t},\boldsymbol{z},\boldsymbol{w}} \quad & \sum_{i=1}^k q_i \sum_{j=1}^n a^j \left( s_{ij} - t_{ij} - 1 \right) \\ \text{s.t.} \quad & \boldsymbol{x} \in X, \\ & s_{ij} \ge 0, \ s_{ij} \ge \tfrac{1}{c}(x^j - y_i^j + c), \\ & s_{ij} \le \tfrac{1}{c}(x^j - y_i^j + c) + M_l^s(1 - z_{ij}), \\ & s_{ij} \le M_u^s z_{ij}, \\ & t_{ij} \ge 0, \ t_{ij} \ge \tfrac{1}{c}(x^j - y_i^j - c), \\ & t_{ij} \le \tfrac{1}{c}(x^j - y_i^j - c) + M_l^t(1 - w_{ij}), \\ & t_{ij} \le M_u^t w_{ij}, \\ & s_{ij} \in \mathbb{R}, \ t_{ij} \in \mathbb{R}, \ z_{ij} \in \{0, 1\}, \ w_{ij} \in \{0, 1\}, \end{aligned}$$

where $M_l^s = M_u^t = \frac{1}{c} - 1$ and $M_l^t = M_u^s = \frac{1}{c} + 1$. The best response of Player 2 is obtained by solving an analogous MILP. Note that the MILP defined above is necessarily different from the one formulated in (Ganzfried 2020).

For the numerical results we consider three battlefields ($n = 3$) with equal weights ($\boldsymbol{a} = (1, 1, 1)$). We observed that the choice of initial strategy sets $X_1$ and $Y_1$ is crucial. Indeed, setting

$$X_1 = Y_1 = \{(1, 0, 0), (0, 1, 0), (0, 0, 1)\}$$

provides much faster convergence than starting from a random point. The reason lies in the left-hand side of Figure 5, which shows the optimal solution produced by the double oracle algorithm for $c = \frac{1}{32}$. The optimal strategies are equidistant on the grid with distance $c$. This is a sensible result as the best response of Player 1 to the strategy $(y_1, y_2, y_3)$ of Player 2 is $(y_1 + c, y_2 + c, y_3 - 2c)$. Since $X_1$ and $Y_1$ already belong to the grid, all the iterates stay in it. However, they may not converge within this set when initial strategies are chosen at random.

The previous observation inspired us to start with both $X_1$ and $Y_1$ as the whole grid. It turned out that the double oracle converged in one iteration (the initial point was already an equilibrium) to the strategies depicted in Figure 6. The left-hand side shows the results for $c = \frac{1}{16}$, while the right-hand side corresponds to $c = \frac{1}{32}$. These results are close to the hexagonal solutions obtained in (Gross and Wagner 1950) and (Roberson 2006).

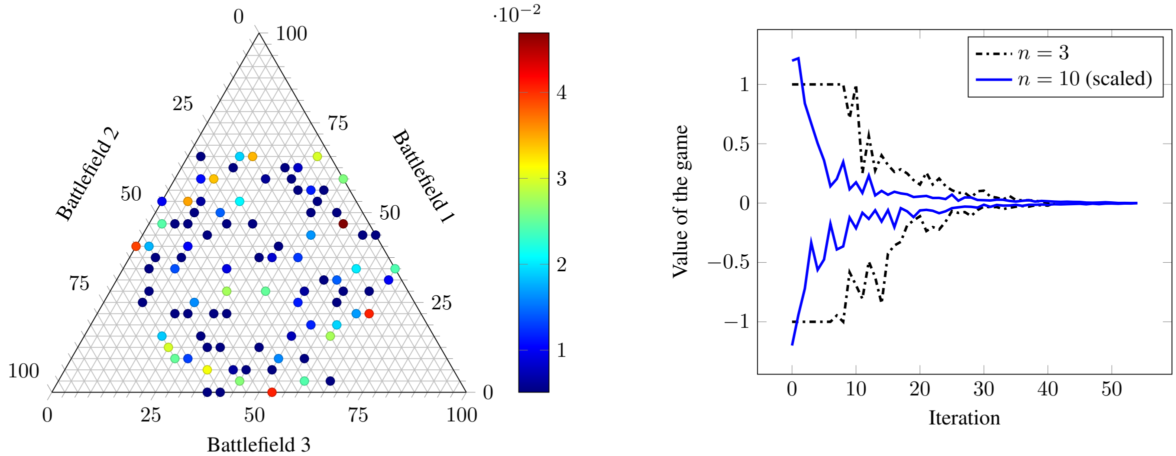


Figure 5: The optimal strategy for $c = \frac{1}{32}$ when started from three corner points (left). The convergence of the double oracle algorithm for $n = 3$ and $n = 10$ (scaled by $\frac{1}{50}$ for demonstration purposes) battlefields (right).

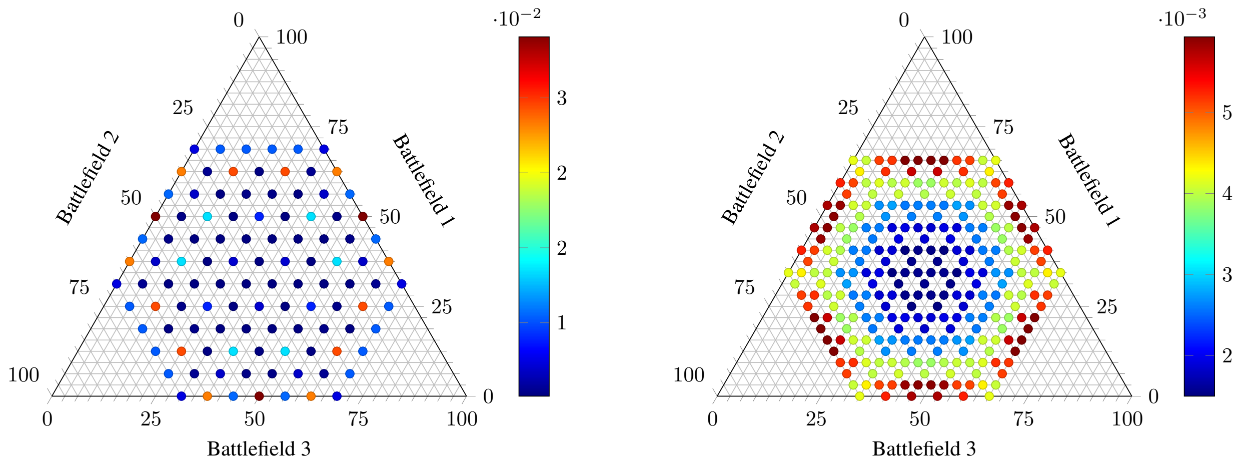


Figure 6: The optimal strategies for $c = \frac{1}{16}$ (left) and $c = \frac{1}{32}$ (right) produced by the double oracle algorithm when started from the grid. Both solutions are symmetric.

The right-hand side of Figure 5 shows the convergence of the double oracle algorithm for $n = 3$ with $\boldsymbol{a} = (1, 1, 1)$ and for $n = 10$ with $\boldsymbol{a} = (3, 4, \dots, 12)$. In both cases we put $c = \frac{1}{16}$. It appears that the convergence is influenced by $c$ more than by the number of battlefields $n$.

## Conclusions

We extended the double oracle algorithm from finite to continuous games. We proved that the algorithm recovers a finitely-supported $\epsilon$-equilibrium in finitely many iterations and converges to an equilibrium in a possibly infinite number of iterations. We showed that the double oracle algorithm performs better than fictitious play on selected examples. It is evident that the convergence of this algorithm depends on the size of constructed subgames and the best response calculation in each iteration. An important open problem is to analyze the speed of convergence of the double oracle algorithm.

## Acknowledgments


This material is based upon work supported by, or in part by, the Army Research Laboratory and the Army Research Office under grant number W911NF-20-1-0197. The authors acknowledge the support by the project *Research Center for Informatics* (CZ.02.1.01/0.0/0.0/16_019/0000765). L. Adam was partially supported from the project GA18-21409S of the Grant Agency of the Czech Republic.


## References


Archibald, C.; and Shoham, Y. 2009. Modeling Billiards Games. In *Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems*, volume 1, 193–199.

Behnezhad, S.; Dehghani, S.; Derakhshan, M.; HajiAghayi, M.; and Seddighin, S. 2017. Faster and Simpler Algorithm for Optimal Strategies of Blotto Game. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, 369–375.

Billingsley, P. 1968. *Convergence of Probability Measures*. New York: John Wiley & Sons Inc.

Brown, G. W. 1951. Iterative Solution of Games by Fictitious Play. In *Activity analysis of production and allocation*, 374–376. John Wiley & Sons.

Brown, N.; and Sandholm, T. 2019. Superhuman AI for Multiplayer Poker. *Science* 365(6456): 885–890.

Fan, K. 1952. Fixed-point and Minimax Theorems in Locally Convex Topological Linear Spaces. *Proceedings of the National Academy of Sciences of the United States of America* 38(2): 121.

Ganzfried, S. 2020. Algorithm for Computing Approximate Nash equilibrium in Continuous Games With Application to Continuous Blotto. *arXiv preprint arXiv:2006.07443* .

Gilpin, A.; Peña, J.; and Sandholm, T. 2012. First-order Algorithm With $\mathcal{O}(\ln(1/\epsilon))$ Convergence for $\epsilon$-equilibrium in Two-person Zero-sum Games. *Mathematical Programming* 133(1-2): 279–298.

Glicksberg, I. L. 1952. A Further Generalization of the Kakutani Fixed Point Theorem, With Application to Nash Equilibrium Points. *Proceedings of the American Mathematical Society* 3: 170–174.

Gross, O.; and Wagner, R. 1950. A Continuous Colonel Blotto game. Technical report, RM-408.

Hart, S.; and Mas-Colell, A. 2000. A Simple Adaptive Procedure Leading to Correlated Equilibrium. *Econometrica* 68(5): 1127–1150.

Hofbauer, J.; and Sorin, S. 2006. Best Response Dynamics for Continuous Zero-sum Games. *Discrete and Continuous Dynamical Systems–Series B* 6(1): 215.

Kamra, N.; Gupta, U.; Fang, F.; Liu, Y.; and Tambe, M. 2018. Policy Learning for Continuous Space Security Games Using Neural Networks. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 1103–1112.

Kamra, N.; Gupta, U.; Wang, K.; Fang, F.; Liu, Y.; and Tambe, M. 2019. DeepFP for Finding Nash Equilibrium in Continuous Action Spaces. In *International Conference on Decision and Game Theory for Security*, 238–258. Springer.

Karlin, S. 1959. *Mathematical Methods and Theory in Games, Programming and Economics. Vol. 2: The Theory of Infinite Games*. Addison-Wesley Publishing Company.

Kroupa, T.; and Majer, O. 2014. Optimal Strategic Reasoning With McNaughton Functions. *International Journal of Approximate Reasoning* 55(6): 1458–1468.

Lanctot, M.; Zambaldi, V. F.; Gruslys, A.; Lazaridou, A.; Tuyls, K.; Pérolat, J.; Silver, D.; and Graepel, T. 2017. A Unified Game-Theoretic Approach to Multiagent Reinforcement Learning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems*, 4190–4203.

Laraki, R.; and Lasserre, J. B. 2012. Semidefinite Programming for Min–max Problems and Games. *Mathematical programming* 131(1-2): 305–332.

McMahan, H. B.; Gordon, G. J.; and Blum, A. 2003. Planning in the Presence of Cost Functions Controlled by an Adversary. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, 536–543.

Moravčík, M.; Schmid, M.; Burch, N.; Lisý, V.; Morrill, D.; Bard, N.; Davis, T.; Waugh, K.; Johanson, M.; and Bowling, M. 2017. Deepstack: Expert-level Artificial Intelligence in Heads-up No-limit Poker. *Science* 356(6337): 508–513.

Muller, P.; Omidshafiei, S.; Rowland, M.; Tuyls, K.; Perolat, J.; Liu, S.; Hennes, D.; Marris, L.; Lanctot, M.; Hughes, E.; et al. 2019. A Generalized Training Approach for Multiagent Learning. *arXiv preprint arXiv:1909.12823* .

Parrilo, P. 2006. Polynomial Games and Sum of Squares Optimization. In *Decision and Control, 2006 45th IEEE Conference on*, 2855–2860.

Perkins, S.; and Leslie, D. S. 2014. Stochastic Fictitious Play With Continuous Action Sets. *Journal of Economic Theory* 152: 179–213.

Rehbeck, J. 2018. Note on Unique Nash Equilibrium in Continuous Games. *Games and Economic Behavior* 110: 216–225.

Roberson, B. 2006. The Colonel Blotto Game. *Economic Theory* 29: 1–24.

Robinson, J. 1951. An Iterative Method of Solving a Game. *Annals of Mathematics* (54): 296–301.

Stein, N. D.; Ozdaglar, A.; and Parrilo, P. A. 2008. Separable and Low-rank Continuous Games. *International Journal of Game Theory* 37(4): 475–504.

Townsend, A. 2014. Constrained Optimization in Chebfun. URL http://www.chebfun.org/examples/opt/ConstrainedOptimization.html.

Yasodharan, S.; and Loiseau, P. 2019. Nonzero-sum Adversarial Hypothesis Testing Games. In *Advances in Neural Information Processing Systems*, 7310–7320.


# Appendix

## Weak Convergence of Measures

We will summarize a necessary background in weak topology on the space of probability measures (Billingsley 1968). A sequence of mixed strategies $(p_i)$ in $\Delta_X$ *weakly converges* to $p \in \Delta_X$ if

$$\lim_{i\to\infty} \textstyle\int_X f(x)\, \mathrm{d}p_i = \int_X f(x)\, \mathrm{d}p$$

for every continuous function $f\colon X \to \mathbb{R}$, and we denote this by $p_i \Rightarrow p$. Endowed with the topology corresponding to weak convergence, the convex set of mixed strategies $\Delta_X$ is a compact space. Analogously, $\Delta_Y$ becomes a compact set and so is the set $\Delta = \Delta_X \times \Delta_Y$. Then the definition (1) warrants that $U$ is a continuous function on $\Delta$. Note that compactness of $\Delta$ and continuity of $U$ imply the existence of all maximizers/minimizers throughout the paper.

**Proposition 3.** *The space $\Delta$ is weakly sequentially compact, that is, every sequence in $\Delta$ contains a weakly convergent subsequence.*

Since $U$ is continuous, the definition of weak convergence immediately implies the following two statements:

- If $p_i \Rightarrow p$ in $\Delta_X$ and $q_i \Rightarrow q$ in $\Delta_Y$, then
$$U(p_i, q_i) \to U(p, q). \tag{10}$$
- If $p_i \Rightarrow p$ in $\Delta_X$ and $y_i \to y$ in $Y$, then
$$U(p_i, y_i) \to U(p, y). \tag{11}$$

Finally, it can be shown that the optimal value of utility function in response to the opponent's mixed strategy is attained for some pure strategy.

**Proposition 4.** *For any $p \in \Delta_X$ we have*
$$\min_{y\in Y} U(p, y) = \min_{q\in\Delta_Y} U(p, q).$$

## Proofs and Additional Results

*Proof of Proposition 2.* The existence of an $\epsilon$-equilibrium follows from Theorem 1. To prove the second part, assume that $(p^*, q^*)$ is an $\epsilon$-equilibrium. Then (3) implies
$$\max_{p\in\Delta_X} U(p, q^*) - \epsilon \le U(p^*, q^*) \le \min_{q\in\Delta_Y} U(p^*, q) + \epsilon. \tag{12}$$

Let $(\hat{p}, \hat{q})$ be an equilibrium of $\mathcal{G}$. Then
$$U(\hat{p}, \hat{q}) \le U(\hat{p}, q^*) \le \max_{p\in\Delta_X} U(p, q^*) \le U(p^*, q^*) + \epsilon,$$
where the first inequality follows from (2) and the third from (12). In a similar way, we can show
$$U(\hat{p}, \hat{q}) \ge U(p^*, \hat{q}) \ge \min_{q\in\Delta_Y} U(p^*, q) \ge U(p^*, q^*) - \epsilon.$$
Combining these two relations with $U(\hat{p}, \hat{q}) = v(\mathcal{G})$ implies the second statement of Proposition 2. □

**Lemma 1.** *Assume $X_{i+1} = X_i$ and $Y_{i+1} = Y_i$ in some step $i$ of Algorithm 1. Then $U(p_i^*, y_{i+1}) = U(x_{i+1}, q_i^*)$.*

*Proof.* The condition $X_{i+1} = X_i$ implies $x_{i+1} \in X_i$. Then
$$U(p_i^*, q_i^*) = \max_{x\in X_i} U(x, q_i^*) = \max_{x\in X} U(x, q_i^*) = U(x_{i+1}, q_i^*),$$
where the first equality follows from Proposition 1 applied to the subgame $(X_i, Y_i, u)$, the second from $x_{i+1} \in X_i$, and the third from the definition of iterate $x_{i+1}$.

Similarly, we can show $U(p_i^*, q_i^*) = U(p_i^*, y_{i+1})$, which means $U(p_i^*, y_{i+1}) = U(x_{i+1}, q_i^*)$. □

**Lemma 2.** *The inequality*
$$\underline{v}_i \le v(\mathcal{G}) \le \overline{v}_i$$
*holds in every step $i$ of Algorithm 1.*

*Proof.* Let $(p^*, q^*)$ be an equilibrium of $\mathcal{G}$. Then
$$\begin{aligned}\underline{v}_i &= U(p_i^*, y_{i+1}) = \min_{y\in Y} U(p_i^*, y) = \min_{q\in\Delta_Y} U(p_i^*, q)\\ &\le U(p_i^*, q^*) \le U(p^*, q^*) = v(\mathcal{G}).\end{aligned}$$
The second inequality can be obtained analogously. □

**Example 1.** Let $X := [0, 1]$, $Y := [0, 1]$, and consider any continuous function $u : X \times Y \to \mathbb{R}$ for which the double oracle algorithm produces an infinite number of iterates $(x_1, y_1), (x_2, y_2), \ldots$ for $\epsilon = 0$. Put $\tilde{X} := [0, 1] \cup [2, 3]$ and define $\tilde{u} : \tilde{X} \times Y \to \mathbb{R}$ by
$$\tilde{u}(x, y) = \begin{cases} u(x, y) & \text{if } x \in [0, 1],\\ u(x-2, y) & \text{if } x \in [2, 3].\end{cases}$$
Since $u$ is continuous, $(\tilde{X}, Y, \tilde{u})$ is a continuous game. Since $\tilde{u}(x, y) = \tilde{u}(x+2, y)$, the extrema of marginal functions are not unique. Considering $\tilde{y}_i = y_i$, the double oracle algorithm may produce the sequence of iterations
$$\tilde{x}_i = \begin{cases} x_i & \text{if } i \text{ is odd},\\ x_i + 2 & \text{if } i \text{ is even}.\end{cases}$$
This sequence is obviously not convergent. However, there exists a convergent subsequence and its limit is an equilibrium by Theorem 1.

## Best Response for Colonel Blotto Game

Function $l$ from (8) can be written as
$$l(z) = \max\left\{\tfrac{1}{c}(z + c), 0\right\} - \max\left\{\tfrac{1}{c}(z - c), 0\right\} - 1.$$
With each $i, j$ in (9) we associate auxiliary variables $s_{ij}$ and $t_{ij}$ and the contraints ensuring $l(x^j - y_i^j) = s_{ij} - t_{ij} - 1$. The constraints on $s_{ij}$ and $t_{ij}$ follow from Lemma 3.

**Lemma 3.** *Let $a > 0$, $b \in \mathbb{R}$, $M_l > 0$, $M_u > 0$ and $f(x) := \max\{a(x-b), 0\}$. For every $x$ such that $a(x-b) \in [-M_l, M_u]$ there are a unique $s \in \mathbb{R}$ and a possibly non-unique $z \in \{0, 1\}$ solving the system*
$$\begin{aligned} &s \ge 0, && s \le a(x-b) + M_l(1-z),\\ &s \ge a(x-b), && s \le M_u z.\end{aligned}$$
*Moreover, it holds $f(x) = s$.*

*Proof.* The proof is based on the well-known big-M method for the deactivation of constraints. The claim follows from the following implications,
$$\begin{aligned} a(x-b) < 0 &\implies z = 0 \implies s = 0,\\ a(x-b) > 0 &\implies z = 1 \implies s = a(x-b).\end{aligned}$$
If $a(x-b) = 0$, then $s = 0$ is unique, whereas $z$ may have either value. □

Since $x^j, y_i^j \in [0, 1]$, we have
$$\begin{aligned} \tfrac{1}{c}(x^j - y_i^j + c) &\in [-\tfrac{1}{c} + 1, \tfrac{1}{c} + 1],\\ \tfrac{1}{c}(x^j - y_i^j - c) &\in [-\tfrac{1}{c} - 1, \tfrac{1}{c} - 1],\end{aligned}$$
which gives the bounds in Lemma 3.