A mostly complete *Algorithmic Game Theory* Summary*

Written by Philip KALUĐERČIĆ†

Last updated for the Winter Semester 2022/23, last rendered February 22, 2023

1 Basic Definitions

Def. 1. A (finite, cost-minimisation) game $\mathcal{G}$ consists of *finite set of players* $N = [n] := \{1, 2, \ldots, n\}$, set of *strategies* $S_i \neq \emptyset$ and *cost function* $C_i : S_1 \times \cdots \times S_n \to \mathbb{R}$ for each player i. We distinguish

Zero-sum iff the social cost is 0.

Bimatrix iff $n = 2$ where $A, B \in \mathbb{R}^{|S_1| \times |S_2|}$ describe the costs of strategies $A_{ij} = C_A(s_{1,i}, s_{2,j})$, and equiv. for B_{ij}.
For a zero-sum, bimatrix game $A = -B$ holds, meaning a single matrix suffices to describe the costs.

Payoff-Max. where players attempt to maximise the "cost", referred to as "payoff".

Def. 2. Strategy profile of a (pure) Game $\mathcal{G}$ is a n-tuple $\vec{s} = (s_1, \ldots, s_n) \in \vec{S} = S_1 \times \cdots \times S_n$ of strategies for each player i.

Def. 3. Social Cost $C(\vec{s})$ of a strategy profile $\vec{s}$ is defined as
$$C(\vec{s}) := \sum_{i \in N} C_i(\vec{s}).$$

Def. 4. A strat. profile consisting of prob. distributions[1]
$$\vec{\sigma} = (\sigma_1, \ldots, \sigma_n) \in \Delta(S_1) \times \cdots \times \Delta(S_n).$$

Def. 5. A (weakly) dominant *strategy* $s_i \in S_i$ minimises the cost regardless of the strategies of other players, s.t.[2]
$$\forall i \in N, \vec{s}_i' \in \vec{S}_i.\, C_i(s_i, \vec{s}_{-i}') \leq C_i(s_i', \vec{s}_{-i}').$$

Def. 6. Pure Nash equilibrium is a strategy profile $\vec{s} = (s_1, \ldots, s_n)$ where no player has an incentive to deviate, i.e.
$$\forall i \in N, s_i' \in S_i.\, C_i(\vec{s}) \leq C_i(s_i', \vec{s}_{-i}).$$

Def. 7. (Mixed) Nash equilibrium is a mixed strategy profile $\vec{\sigma} = (\sigma_1, \ldots, \sigma_n)$ where no player has an incentive to deviate, i.e.[3]
$$\forall i \in N, \sigma_i' \in \Delta(S_i).\, C_i(\vec{\sigma}) \leq C_i(\sigma_i', \vec{\sigma}_{-i}).$$
⇝ Thm. 1 Every game has at least one NE.

2 Congestion Games

Def. 8. A potential games has a *(exact) potential function* $\Phi : \vec{S} \to \mathbb{R}$, s.t.
$$\forall i \in N, \vec{s} \in \vec{S}, s_i' \in S_i.\, C_i(s_i', \vec{s}_{-i}) - C_i(\vec{s}) = \Phi(s_i', \vec{s}_{-i}) - \Phi(\vec{s})$$
⇝ Thm. 2 Every potential game has at least one pure Nash equilibrium.

Def. 9. Congestion Game $\mathcal{G} = (N, E, \{S_i\}_{i \in N}, \{c_e\}_{e \in E})$ is a potential game $(N, \{S_i\}_{i \in N}, \{C_i\}_{i \in N})$ with a set of resources E and a nondecreasing set of cost functions $c_e : \mathbb{N} \to \mathbb{R}_{\geq 0}$ representing the number of players using it (*congestion*). The cost of a strategy profile for a player $i \in N$ is
$$C_i(\vec{s}) = \sum_{e \in s_i} c_e(n_e(\vec{s})),$$
where $n_e(\vec{s}) = |\{i \mid e \in s_i\}|$ and the social cost is
$$C(\vec{s}) = \sum_{i=1}^{n} C_i(\vec{s}) = \sum_{e \in E} n_e(\vec{s}) c_e(n_e(\vec{s})).$$
We distinguish

Network/Routing on a graph $G = (V, E)$ we define $\{S_i\}$ to be all the paths between two nodes $o_i, d_i \in V$ (origin and destination of each player i).

Polynomial of degree d (or "linear" if $d = 1$) where each cost function c_e is defined by a polynomial in
$$C_d := \left\{ x \mapsto \sum_{j=0}^{d} a_j x^j \;\middle|\; a_j \in \mathbb{R}_{\geq 0} \forall j \in \{0, \ldots, d\} \right\}$$

Singleton Strategies s_i are singleton sets.

Symmetric All players have the same set of actions.

All congestion games are potential games, via the "Rosenthal's potential": ⇝ Thm. 3
$$\Phi(\vec{s}) = \sum_{\vec{e} \in E} \sum_{j=1}^{n_e(\vec{s})} c_e(j).$$

Def. 10. Price of Anarchy of a Potential game $\mathcal{G}$ with a set of pure Nash equilibria $\mathrm{PNE}(\mathcal{G})$ is ⇝ Thm. 4
$$\mathrm{PoA}(\mathcal{G}) = \frac{\max_{s \in \mathrm{PNE}(\mathcal{G})} C(\vec{s})}{C(\vec{s}^*)},$$
where $\vec{s}^* \in \mathrm{argmin}_{s \in S} C(\vec{s})$.

Def. 11. Price of Stability, as with Price of Anarchy, is ⇝ Thm. 5
$$\mathrm{PoS}(\mathcal{G}) = \frac{\min_{s \in \mathrm{PNE}(\mathcal{G})} C(\vec{s})}{C(\vec{s}^*)}.$$

Def. 12. Better-response dynamics is a method for calculating a PNE: ⇝ Thm. 6

Algo. 1 $\mathsf{BR}(\mathcal{G}, \vec{s} \in \vec{S})$ **returns** a PNE

```
while s⃗ is not a PNE do
    Choose a i ∈ N, s'_i ∈ S_i, s.t. C_i(s'_i, s⃗_{-i}) < C_i(s⃗)
    s⃗ ← (s'_i, s⃗_{-i})
```

Variations include "*best*-response dyn.", which chooses the best deviation or "*maximum-gain* best-response dyn." which chooses the best relative deviation.

For a Singleton and symmetric, network game, a PNE can be computed in polynomial time. ⇝ Thm. 7

Def. 13. α-approximate PNE given $\alpha \geq 1$, we relax the equilibrium condition ⇝ Thm. 8 ⇝ Thm. 9
$$\forall i \in N, s_i' \in S_i.\, C_i(\vec{s}) \leq \alpha C_i(s_i', \vec{s}_{-i}).$$

3 Computation of Pure Nash Equilibria

Def. 14. Local search problem $\Pi = (\mathcal{I}, S, N, f)$ consisting of a set of instances $\mathcal{I}$, each $I \in \mathcal{I}$ having a set of feasible solutions S_I, a neighbourhood function $N_I : S_I \to 2^{S_I}$ and a objective function $f_I : S_I \to \mathbb{Z}$.

Def. 15. Polynomial local search (PLS), a local search problem, for which given an instance $I \in \mathcal{I}$ and a feasible solution $x \in S_I$, an initial feasible solution ($\mathcal{A}_\Pi$), an objective value $f_I(x)$ ($\mathcal{B}_\Pi$) and the verification that x is a local optimum *or* compute a local neighbour $y \in N(x)$ with $f_I(y) < f_I(x)$ can all be calculated in polynomial time.

---

*Based on the lecture "Algorithmic Game Theory" (2022/23), as held by Yiannis Giannakopoulos at the University of Erlangen–Nuremberg. The numbering used in this document for definitions and theorems differs from the script used in the lecture.

†https://gitlab.cs.fau.de/oj14ozun/agt-summary, the source for this document should be accessible as a PDF attachment.

[1] $\Delta(A) = \left\{ p \in [0,1]^A \mid \sum_{a \in A} p(a) = 1 \right\}$.

[2] $(y, \vec{x}_{-i}) := (x_1, \ldots, x_{i-1}, y, x_{i+1}, \ldots, x_n)$.

[3] Notation for the expected cost $C_i(\vec{\sigma}) := \mathbb{E}_{\vec{s} \sim \vec{\sigma}}[C_i(\vec{s})]$.

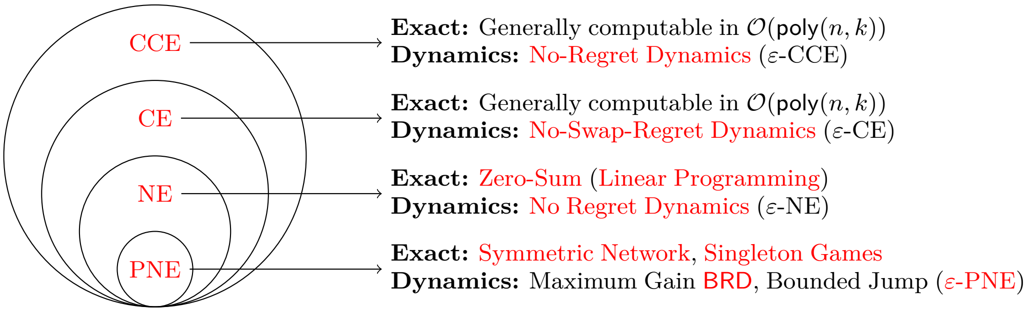

Figure 1: Overview of types of equiliria and their methods of computation.

Def. 16. A best-response (di) graph of a congestion game G with a potential function Φ consists of vertices corresponding to strategy profiles, and directed edges pointing from a profile to another with a single-player beneficial deviation.

The skin of this graph (vertex with out-degree of 0) is the NE of G, as it has no beneficial deviations. An algorithm like best-response dynamics can therefore be interpreted as local search on the best-response graph, optimising Φ.

Def. 17. PLS-reduction $\Pi \leq_{\mathsf{PLS}} \Pi'$ is given for two PLS problems $\Pi = (\mathcal{I}, S, N, f)$, $\Pi' = (\mathcal{I}', S', N', f')$, and polynomial-time commutable functions h with $\forall I \in \mathcal{I}.\, h(I) \in \mathcal{I}'$ and g with

$$\forall I \in \mathcal{I} \forall x' \in S'_{h(I)}.\, g(x') \in S_I.$$

In this case it is said that Π' is "as hard as" Π.

⇝ Thm. 10

One PLS-complete problem is PNE-Congestion, i.e. the computation of a PNE for a congestion game. .

4 Mixed Nash Equilibria

⇝ Thm. 11

Def. 18. The value of a zero-sum game $A \in \mathbb{R}^{m\times n}$ is

$$V^A := \vec{x}^{*\top} A \vec{y}^* = \max_{\vec{x}\in\Delta^m} \min_{\vec{y}\in\Delta^n} \vec{x}^\top A \vec{y} = \min_{\vec{y}\in\Delta^n} \max_{\vec{x}\in\Delta^m} \vec{x}^\top A \vec{y},$$

where $(\vec{x}^*, \vec{y}^*)$ is an optimal strategy profile, that can be computed in polynomial time

Def. 19. Fictitious play is a dynamic, given a repeated bimatrix game, where players maintain a pair of mixed strategies $(\vec{x}^t, \vec{y}^t)$ determined by the empirical distribution of their respective opponents past play:

$$\vec{x} = \frac{1}{t}\sum_{\tau=1}^{t} e^m_{i_\tau}, \qquad \vec{y} = \frac{1}{t}\sum_{\tau=1}^{t} e^n_{j_\tau},$$

where for a time step t, $e^m_{i_t} \in \Delta^m$, $e^n_{j_t} \in \Delta^n$, s.t.

$$e^m_{i_t} \in \mathsf{BR}_1(\vec{y}^{t-1}) \qquad e^n_{j_t} \in \mathsf{BR}_2(\vec{x}^{t-1})$$

and x^0, y^0 are respectively m or n dimensional zero-vectors.

Def. 20. Duality gap of a mixed strategy profile $(\vec{x}, \vec{y})$ for a zero-sum game $A \in \mathbb{R}^{m\times n}$ is

$$\Psi_A(\vec{x}, \vec{y}) := \max_{\vec{x}'\in\Delta^m} \vec{x}'^\top A\vec{y} - \min_{\vec{y}'\in\Delta^n} \vec{x}^\top A \vec{y}.$$

Karlin's conjecture states that $\Psi_A(\vec{x}^t, \vec{y}^t) \in \mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$.

5 Correlated Equilibria

Def. 21. Correlated equilibria has a strategy profile $\vec{\sigma} \in \Delta(\vec{S})$, where $\vec{S} = S_1 \times \cdots \times S_n$ (Cartesian product) for a Game $\mathcal{G}$ s.t.

$$\forall i \in N, \delta_i : S_i \to S_i.\, \mathop{\mathbb{E}}_{\vec{s}\sim\vec{\sigma}}[C_i(\vec{s})] \leq \mathop{\mathbb{E}}_{\vec{s}\sim\vec{\sigma}}[C_i(\delta_i(s_i), \vec{s}_{-i})],$$

where δ is called a *swap function* or equivalently

$$\forall i \in N, s_i, s_i' \in S_i.\, \mathop{\mathbb{E}}_{\vec{s}_{-i}\sim\vec{\sigma}_{-i|s_i}}[C_i(s_i, \vec{s}_{-i})] \leq \mathop{\mathbb{E}}_{\vec{s}_{-i}\sim\vec{\sigma}_{-i|s_i}}[C_i(s_i', \vec{s}_{-i})]$$

⇝ Thm. 12

This is a generalisation of (mixed) Nash equilibria, in the sense that $\Delta(\vec{S})$ is a distribution over strategy profiles, while NEs consist of $\Delta(S_i)$ distributions over strategies for i. CEs can be computed in polynomial time via LP.

Def. 22. Coarse Correlated Equilibria has a strategy profile $\vec{\sigma} \in \Delta(\vec{S})$ for a Game $\mathcal{G}$ s.t.

$$\forall i \in N, s_i' \in S_i.\, \mathop{\mathbb{E}}_{\vec{s}\sim\vec{\sigma}}[C_i(\vec{s})] \leq \mathop{\mathbb{E}}_{\vec{s}\sim\vec{\sigma}}[C_i(s_i', \vec{s}_{-i})].$$

This is a generalisation of correlated equilibria, considering only constant swap functions. It differs from NE, in that we quantify over $s_i \in S_i$ instead of $\sigma_i' \in \Delta(S_i)$.

Def. 23. Online learning algorithm $\mathcal{A} = \{p^t\}$ calculates a probability distribution p^t of a set of actions A with $|A| = k$ in a dynamic decision-making setting. Knowing this, an adversary sets a cost function $c^t : A \to [0;1]$ that an action a^t drawn from p^t is applied onto, for every time step $t = 1, 2, \ldots, T$, where $T \in \mathbb{N}_{\geq 1}$ is a time horizon. The regret of a sequence of action $\vec{a} = \{a^t\}_{t\in[T]}$ is defined as

$$\mathcal{R}(\vec{a}, \vec{c}) := \frac{1}{T}\left[\sum_{t=1}^{T} c^t(c^t) - \min_{a\in A}\sum_{t=1}^{T} c^t(a)\right].$$

If the "expected regret" $\mathcal{R}^T_{\mathcal{A}} := \sup_{\vec{c}} \mathbb{E}_{\vec{a}\sim\vec{p}}[\mathcal{R}(\vec{a}, \vec{c})] \to 0$, we say $\mathcal{A}$ has *no regret.*

Examples include:

⇝ Thm. 13

"Follow the Leader" A deterministic algorithm that chooses an action based on the expected cost so far. As all deterministic algorithms, it has a regret of at least $1 - \frac{1}{k}$.

⇝ Thm. 14

Multip. Weight Update A randomised algorithm, given $\eta \in [0; \frac{1}{2}]$ maintains a set of weights $\{w^t(a)\}_{a\in A}$ used at each time-step t to assign probabilities

$$p^t(a) = \frac{w^t(a)}{\sum_{\bar{a}A} w^t(\bar{a})},$$

and update the weights by $w^{t+1}(a) = w^t(a)(1-\eta)^{c^t(a)}$. This archives being "no-regret".

Def. 24. No-regret game dynamics, using a cost-min game $\mathcal{G}$ and a decision making process, where each player $i \in [N]$ uses a no-regret algorithm like MWU. A ε-approximate coarse correlated equilibria can be calculated using NRD:

Algo. 2 $\mathsf{NRD}(\mathcal{G}, \{A_i\}_{i\in N}, T)$ **returns** a cr. strat. profile $\vec{\sigma}$

```
1: for t = 1, ..., T do
2:     for i ∈ N do Compute p_i^t over S_i using A_i
3:     Define mixed profile σ^t = ∏_{i∈N} p_i^t
4:     for i ∈ N do Adversary presents player i with i
           c_i^t(s_i) := E_{s_{-i} ~ σ^t_{-i}} [C_i(s_i, s_{-i})], ∀s_i ∈ S_i
5: return σ = (1/T) ∑_{t=1}^T s^t
```

In a zero sum game the guality gap will at most be 2ε.

Def. 25. Swap-regret is a stricter benchmark than swap regret:

$$\bar{\mathcal{R}}(\vec{a},\vec{c}) \coloneqq \frac{1}{T}\left[\sum_{t=1}^{T} c^t\left(a^t\right) - \min_{\delta:A\to A}\sum_{t=1}^{T} c^t\left(\delta\left(a^t\right)\right)\right],$$

hence $\mathcal{R}(\vec{a},\vec{c}) \leq \bar{\mathcal{R}}(\vec{a},\vec{c})$. When used with no-regret dynamic, we converge to the set of correlated equilibria.

A Proof Sketches

NOTEME: The proofs in this section make no claim to be rigorous, just to convey an approximate approach strategy employed in attempting to prove the following theorems.

The document source is publicly available (see the frontpage), so any ideas on how to improve the following proofs are much appreciated.

Thm. 1. Every game has at least one NE.

Proof. Consider the function $\vec{f} = (f_1,\dots,f_n)$ point-wise defined on a payoff-maximisation game $\mathcal{G} = \left(N, \{S_i\}_{i\in N}, \{u_i\}_{i\in N}\right)$

$$f_i(\vec{\sigma}) \coloneqq \operatorname*{argmax}_{\bar{\sigma}\in\Delta(S_i)}\left(\mathop{\mathbb{E}}_{\vec{s}\in\bar{\sigma},\vec{\sigma}_{-i}}[u_i(\vec{s})] - \|\bar{\sigma}_i - \sigma_i\|^2\right)$$

The proof verifies that $\vec{f}$ must have a fixed-point and that it is a NE. *Brouwer's Fixed-Point Theorem* is used to prove the existence of a fixed-point, by exploiting the structure of the space $\Delta(S_i)\times\cdots\times\Delta(S_n)$ (being nonempty, compact and convex) assuming f is continuous. *Berge's Maximum Theorem* derives the necessary continuity from f being well-defined. □

Thm. 2. Every potential game $\mathcal{G} = (N, \{S_i\}_{i\in N}, \{c_i\}_{i\in N})$ has at least one PNE

Proof. Given the potential minimizer $\vec{s}^* \in \operatorname{argmin}_{\vec{s}\in\vec{S}}\Phi(\vec{s})$, any deviation $\forall i \in N, s_i' \in S_i$

$$\Phi\left(s_i', \vec{s}^*_{-i}\right) - \Phi\left(\vec{s}^*\right) \geq 0$$

using the definition of potential games, we can deduce the PNE condition

$$\Rightarrow C_i\left(s_i', \vec{s}^*_{-i}\right) - C_i\left(\vec{s}^*\right) \geq 0 \Rightarrow \underline{C_i\left(\vec{s}^*\right) \leq C_i\left(s_i', \vec{s}^*_{-i}\right)}$$

□

Thm. 3. Using *Rosenthal's potential* function

$$\Phi(\vec{s}) = \sum_{e\in E}\sum_{j=1}^{n_e(\vec{s})} c_e(j),$$

congestion games $\mathcal{G} = \left(N, E, \{S_i\}_{i\in N}, \{c_e\}_{e\in E}\right)$ (which in turn is a game $\left(N, \{S_i\}_{i\in N}, \{C_i\}_{i\in N}\right)$) are potential games, hence have at least one PNE.

Proof. For any deviation $(s_i', \vec{s}_{-i})$ from $\vec{s}$ the number of players using any resource will stay the same (if unaffected by the deviation), increase (if part of the deviation) or decrease (if part of the previous strategy profile).

Therefore, if we only consider the congestion difference when observing a single player $(C_i(s_i', \vec{s}_{-i}) - C_i(\vec{s}))$ and amongst all players

$$\Phi\left(s_i', \vec{s}_{-i}\right) - \Phi\left(\vec{s}\right) = \sum_{e\in E}\sum_{j=1}^{n_e'(s_i',\vec{s}_{-i})} c_e(j) - \sum_{e\in E}\sum_{j=1}^{n_e'(\vec{s})} c_e(j)$$

will differ by the same amount, and therefore satisfy the *potential game* condition. □

Thm. 4. The price of anarchy of a linear congestion game is exactly $\frac{5}{2}$.

Proof. This is derived from an upper and lower bound, via algebraic reformulations. □

Thm. 5. The price of stability of a polynomial congestion game $\mathcal{G}$ of degree d is at most $d+1$.

Proof. Using the *Potential Method*, that states

$$\exists\alpha,\beta > 0\,\forall\vec{s}.\,\alpha C(\vec{s}) \leq \Phi(\vec{s}) \leq \beta C(\vec{s}) \Rightarrow \mathrm{PoS}(\mathcal{G}) < \frac{\alpha}{\beta}$$

one can upper bound PoS by taking $\alpha = \frac{1}{d+1}$ and $\beta = 1$. □

Thm. 6. Better-response-dynamics find a pure Nash equilibrium for a congestion game in $nmc_{\max}$ iterations, where n is the number of players, m is the number of resources and $c_{\max} \coloneqq \max_e c_e(n)$

Proof. The number of iterations is bounded by the difference between $\min_{\vec{s}}\Phi(\vec{s}) \geq 0$ and

$$\max_{\vec{s}}\Phi(\vec{s}) = \max_{\vec{s}}\sum_{e}\sum_{j=1}^{n_e(\vec{s})} c_e(j) \leq \sum_{e} nc_{\max} = mnc_{\max}$$

□

Thm. 7. A PNE can be found in polynomial time for a singleton congestion game $\mathcal{G}$.

Proof. Via theorem 6, by that a $\tilde{\mathcal{G}}$ with modified costs $\tilde{c}_e$ s.t. $\tilde{c}_{\max}$ that is in $\mathcal{O}(\mathsf{poly}(n,m))$

$$\tilde{c}(j) = |\{c_e(j) \mid e\in E, j\in[n], c_e(j) < c_e(j)\}| - 1.$$

then proving the equivalence between $\mathcal{G}$ and $\tilde{\mathcal{G}}$

$$c_{e'}\left(n_{e'}(\vec{s})+1\right) < c_e\left(n_e(\vec{s})\right)$$
$$\iff \tilde{c}_{e'}\left(n_{e'}(\vec{s})+1\right) < \tilde{c}_e\left(n_e(\vec{s})\right)$$

which is given, since $\tilde{c}_e$ by construction preserves the order of cost values. □

Thm. 8. A PNE can be found in polynomial time for a symmetric, network congestion game $\mathcal{G}$.

Proof. Via reduction to Max-Cost Flow. For this to work a network corresponding to $\mathcal{G}$ has to be provided. Do this by retaining the vertices and replacing each edge $e$with $|N|$ parallel edges for each player (since symmetric), where each edge i has a constant capacity and a cost corresponding to $c_e(i)$. The network is constrained s.t. flows equal to the number of players are allowed. □

Thm. 9. A $(1+\varepsilon)$-PNE can be found in polynomial time on $\frac{1}{\varepsilon}$, α and the site of the input for a symmetric congestion game.

Proof. By proving that *maxinum-gain* best-response dynamics always find a $(1+\varepsilon)$-approx. PNE in

$$\left(1+\frac{1}{\varepsilon}\right)\alpha n\ln(nmc_{\max})$$

iterations, since every step makes an $(1+\varepsilon)$-improving move □

Thm. 10. Computing a PNE for a congestion game via Rosenthal's potential $\Phi(\vec{s})$ – PNE-Congestion – is PLS-complete.

Proof. A problem is PLS-complete if all problems in PLS can be reduced to it. This can be done by a reducing PNE-Congestion to Local-Max-Cut and vice versa. The former is given by theorem 8. Given a Local-Max-Cut problem we construct a game s.t. when given a cut and translated into a game, this corresponds to a PNE. □

Thm. 11. The value V^A of a zero sum game A can be computed in polynomial time.

Proof. Done by reduction to linear programming: We can compute $(\vec{x}^*, \vec{y}^*)$ using

$$\begin{aligned}
&\min . \vec{x}^* \\
&\text{s.t.} \sum_{i=1}^{m} x_i a_{i,j} \geq \vec{x}^* && \forall j \in [n], \\
&x_1 + \cdots + x_m = 1 \\
&x_i \geq 0 && \forall i \in [m]
\end{aligned}$$

and the dual problem

$$\begin{aligned}
&\min . \vec{y}^* \\
&\text{s.t.} \sum_{j=1}^{n} y_i a_{i,j} \geq \vec{y}^* && \forall i \in [m], \\
&y_1 + \cdots + y_n = 1 \\
&y_i \geq 0 && \forall i \in [n]
\end{aligned}$$

□

Thm. 12. A correlated equilibrium of a game can be computed in polynomial time.

Proof. Done by reduction to linear programming (which is solvable in polynomial time) with the variables being probabilities, that would calculate the set of CEs. We formulate constraints preventing deviation and ensuring the probability distribution is valid. □

Thm. 13. Any deterministic online algorithm for a decision making setting with $|A| = k$ actions, has regret of at least $1 - \frac{1}{k}$.

Proof. As a deterministic algorithm chooses a single action (assigning it a probability of 1, and 0 to the rest) based on previous experience, an adversary can predict the next step and use that to generate a pathological cost function returning a cost of 1. The total cost is therefore the number of time-steps T, and $\frac{T}{k}$ for a specific $a_j \in A$. This can be used to bound

$$\mathcal{R}_{\mathcal{A}}^T \geq \cdots \geq \frac{1}{T}\left(T - \frac{T}{k}\right) = 1 - \frac{1}{k},$$

for any T □

Thm. 14. The expected regret of MWU(η) is at most $2\sqrt{\frac{\ln k}{T}}$, for a set of actions A where $|A| = k$ and any time horizon $T \geq 4 \ln k$.

Proof. The proof involves demonstrating that

$$\underbrace{\sum_{t=1}^{T} \sum_{a \in A} p^t(a) c^t(a)}_{(\aleph)} \leq (1+\eta) \underbrace{\left(\min_{a \in A} \sum_{t=1}^{T} c^t(a) \right)}_{(\beth)} + \frac{\ln k}{\eta}$$

holds, where $(\aleph)$ and $(\beth)$ arise from the term for

$$\mathcal{R}_{\mathcal{A}}(\vec{c}) = \frac{1}{T}\left[(\aleph) - (\beth)\right]$$

where we are interested in the upper bound

$$\leq \frac{1}{T}\left[(1+\eta)(\beth) + \frac{\ln 4}{\eta} - (\beth)\right]$$

which is achieved by means of algebraic manipulations. □