Project Report 1–8

# Guiding Selfish Learners Toward Social Optimum

**Georgy Noarov** GNOAROV@PRINCETON.EDU and **Rohan Rao** ROHANR@PRINCETON.EDU
*Princeton University*

## Abstract

We consider the setting where a group of $n$ selfish agents (learners) are each trying to achieve a learning objective. Each agent has a choice between using a private sample as their training set, or sharing the private sample with others so as to train their classifier, just as other collaborators, on the pooled data. As a counterweight to this benefit of collaboration, we assume that it also imposes a bulk privacy risk on the agents who choose this option. We model this setting as an instance of a fair cost-sharing game where the agents are trying to minimize their costs, which include their classifiers' error rates as well as privacy risks.

We show how to gently but efficiently guide agents towards collaborating. To do so, we apply to this game instance a protocol from Balcan et al. (2013). Agents are allowed, at each step of the protocol, to either perform best-response to the current state, or temporarily commit to collaborating. Under some natural assumptions, we show that after $O(n^3)$ iterations, agents arrive at a state with social cost $O(OPT)$, where $OPT$ is the social cost of collaboration. This guarantee holds even if a constant fraction of all players is allowed to opt-out of the protocol and instead either always collaborate or always non-collaborate.

## 1. Introduction

Theoretical machine learning and game theory are quite interconnected. This is not surprising. On the one hand, some games require the use of machine learning protocols in order to guarantee convergence of players to a social welfare-optimal or equilibrium state. For example, in certain games there may exist states from which it is impossible to get to an equilibrium or socially optimal state via any polynomially long sequence of best responses (Chien and Sinclair (2011)). Thus, if a benevolent *central authority* desires to make players quickly converge to such a state, one must resort to e.g. communicating the desired behavior of the system to the players. Each player would then try, using a private learning algorithm, to decide whether it makes sense to follow the proposed strategy by observing other players' reactions to the proposal. In particular, Balcan et al. (2013) provides such a protocol to make players reach a socially good state in fair network cost-sharing games.

On the other hand, certain machine learning frameworks also benefit from game-theoretic results, especially if such frameworks allow learners to have incentives. A prime example of this is *collaborative machine learning*. For an agent, it may be hard to make a long-term decision on whether to collaborate or not, because whether the system will eventually converge to the collaborative state or not will depend on individual players' decisions at all time steps. Therefore, it is natural to model such a setting as a game. Furthermore, since collaboration is generally socially more optimal than non-collaboration, convincing players to collaborate plays an important role. Thus, one might want to apply to this game a pro-

tocol of the kind mentioned above. This would close the above circle of ideas: a specially devised learning protocol would now help improve the results of an entirely different kind of learning - collaborative machine learning - by steering selfish learners towards collaboration.

Recent research has provided fresh perspectives on incentives in collaborative machine learning. Bonawitz et al. (2019) describe the following design of multiple distributed collaborative learning systems. It involves an application which collects user information on a centralized server in order to train a global algorithm, and then broadcasts the trained algorithm back out to the users' local application instances. Bonawitz et al. (2019), Yang et al. (2019), Konečný et al. (2016) all describe similar infrastructures which discuss secure, distributed ways for users to contribute their information to a collective pool, so as to obtain a better-trained algorithm that would improve their experience. The collaborative learning model described above, rather than requiring users to consent to the pooling of their data, allows them to choose not to pool their data at the expense of only being able to use an algorithm trained on their own data. In context, the above-mentioned central authority that cares about the average quality of the users' algorithms can be interpreted as the creator of the app. This setting models the dynamics of recent and expected legislation regarding increased protections of consumer data (such as the right to be forgotten, which gives users the right to revoke consent of the usage of their data Yang et al. (2019)).

### 1.1. Results

Inspired by these ideas, we propose a new, privacy-aware collaborative PAC learning model with incentives. We then represent it as a fair cost-sharing game where players' costs reflect Occam's Razor-type upper[1] bounds on individual or collaborative performance of players' PAC classifiers. As the model is privacy-aware, it is also assumed that collaboration incurs some privacy risk, which is also factored into these costs. By utilizing the protocol from Balcan et al. (2013), we then obtain guarantees on how well one can guide agents towards collaboration in terms of social cost. Specifically, we show that after $O(n^3)$ iterations of the protocol, the players will have converged to a state whose social cost is within a constant factor from the optimal social cost $OPT$ (which is achieved in the all-collaborate state). Note that the proof in Balcan et al. (2013) only guarantees convergence to social cost within a $\log n$ factor of $OPT$, where $n$ is the number of players; we improve on this by tailoring our analysis of the protocol to the specific network that we are using.

We emphasize differences from previous research. The idea of using Occam's Razor upper bounds to model the cost of collaboration/non-collaboration, as well as the interpretation of it in terms of proportional cost sharing, is original. Furthermore, to the best of our knowledge, this paper is the first to construct a protocol that would lead players in a collaborative learning setting to converge to the collaborative state. The paper most closely related to ours is Redko and Laclau (2019). They also reduce a collaborative learning setting (viewed from an empirical risk minimization (ERM) perspective) to the same fair cost sharing game as we do, but they exclude incentives and privacy considerations from the picture, and do not attempt to lead players into the collaboration. Instead, they use the

1. Why do we consider upper and not, say, lower bounds on classifiers' errors? In short, for privacy-aware learning, what matters is the worst-case guarantees. One can think of face recognition software on modern IPhones. It should only unlock the screen when it sees the owner of the phone. The guarantee that IPhone users are after is a low upper bound on the error rate.

reduction as a means to algebraically upper bound the factor by which the sum of players' empirical risks is smaller in the all-collaborate state compared to the no-collaboration state[2].

## 2. Definitions and the Model

**Fair Cost-Sharing Games** Formally, a fair cost sharing game consists of the following ingredients. There is a directed and connected network $G = (V, E)$ with edge cost $c_e > 0$ for every edge $e \in E$. There are $n$ selfish players indexed by $i \in [n]$. Player $i \in [n]$ wants to get from his source $s_i \in V$ to his destination $t_i \in V$, and takes a directed path $P_i$ from $s_i$ to $t_i$ through the network $G$. Thus, his strategy set $S_i$ consists of all such paths in $G$. Consider any strategy profile (which we also call "state") $P = (P_1, \ldots, P_n)$ where each player has selected a path between his source and destination. The cost of path $P_i = \{e_{i,1}, ..., e_{i,len(P_i)}\}$ is defined as $\text{cost}(P_i) = \sum_{j \in [len(P_i)]} \frac{c_{e_{i,j}}}{n_{e_{i,j}}}$, where $n_e = |\{i \in [n] : e \in P_i\}|$ is the number of players on edge $e \in E$ in the strategy profile $P$.

Given profile $P$, each player pays $\text{cost}_i(P) := \text{cost}(P_i)$. The goal of each player is to minimize his own cost. The social cost of a state $P$ is defined as $SC(P) := \sum_{i \in [n]} \text{cost}_i(P)$. We assume that players move one at a time, in an arbitrary order. For a strategy profile $P$ and any $P_i' \in S_i$ for any $i \in [n]$, we denote by $(P_{-i}, P_i')$ the strategy profile obtained by changing Player $i$'s strategy to $P_i'$ and keeping all other players' strategies fixed. A move by Player $i$ is called a best-response to the current strategy profile $P$ if he switches to path $P_i' = \text{argmin}_{P_i \in S_i} \text{cost}_i((P_{-i}, P_i'))$. A state $P$ is a Nash Equilibrium (NE) if for each Player $i \in [n]$, his best response to $P$ is his strategy $P_i$ in $P$.

Anshelevich et al. (2004) introduced an important tool that we will use to track improvements in social cost when applying the protocol from Balcan et al. (2013) to a fair cost-sharing game instance. The *potential function* $\Phi$ of a fair cost-sharing game is a function from $\mathcal{S}$ to $\mathbb{R}_+$, where $\mathcal{S}$ is the set of all states of the game, such that for any state $P$, any Player $i$, and any strategy $P_i' \in S_i$, the unilateral deviation of Player $i$ from $P_i$ to $P_i'$ induces potential difference equal to his change in cost: $\Phi((P_{-i}, P_i')) - \Phi(P) = \text{cost}_i((P_{-i}, P_i')) - \text{cost}_i(P)$. Furthermore, we have

**Lemma 1 (Anshelevich et al. (2004))** *At any state $S$ of an (unweighted) $n$-player cost sharing game,* $\text{cost}(S) \leq \Phi(S) \leq H_n \text{cost}(S)$, *where* $H_n = 1 + \frac{1}{2} + \ldots + \frac{1}{n}$.

**Modeling Pooling vs. Non-Pooling of Data: A PAC-learning Approach** We consider the following setting. Suppose there are $n$ agents, and Agent $i \in [n]$ has access to a PAC-learner $A_i$ with a finite hypothesis space $\mathcal{H}$ and parameter $\delta := \delta_i$ (the bound on the probability of high error) and set $S_i$ of $m$ training samples from a common distribution $\mathcal{D}$. Each agent can either train $A_i$ on $S_i$, which we denote $N$ (for 'Non-Collaborate'), or it can join a collaboration, which we denote $C$ (for 'Collaborate'). In the latter case, letting $\mathcal{F} \in [n]$ be the set of indices of the agents who have chosen $C$, the algorithm $A_i$ is trained on the *pooled* dataset $S_{pool} = \bigcup_{j \in \mathcal{F}} S_j$.

2. Specifically, they define the social cost of any state to be the sum of players' empirical risks in that state, and to upper bound the mentioned factor, they use an algebraic bound on the so-called *Price of Stability* of weighted fair cost sharing games. The interested reader is referred to Chen and Roughgarden (2008).

Now we introduce incentives into this model. We define the cost incurred by any agent who has chosen strategy $N$ as the Occam's Razor bound on the probability $err_{\mathcal{D}}(h_{A_i})$ of its classifier $A_i$ (trained on $S_i$) making an error. Precisely, the cost of an Agent $i \in [n]$ who is using strategy $N$ is set to be $c_i = \frac{\ln|\mathcal{H}|+\ln\frac{1}{\delta_i}}{m}$. In order to define the cost of collaboration, we suppose that agents who choose $C$ subject themselves to *bulk privacy risk* $\Pi > 0$. This risk is proportionally shared between the collaborators. Indeed, one can think of a malicious adversary who breaks into the database storing the pooled samples $S_{pool}$ with some probability $\alpha > 0$, and steals the batch $S_j$ of a random $j \in \mathcal{F}$. Then, $\Pi = \alpha$, and the probability that any particular collaborator's batch is stolen is $\frac{\alpha}{|\mathcal{F}|} = \frac{\Pi}{|\mathcal{F}|}$. Furthermore, we consider the probability that on a test sample $x \sim \mathcal{D}$, at least one collaborator's classifier makes an error. Applying Occam's Razor along with the union bound, we see that this probability is at most $\sum_{j\in\mathcal{F}} \frac{\ln|\mathcal{H}|+\ln\frac{1}{\delta_j}}{|S_{pool}|} = \sum_{j\in\mathcal{F}} \frac{\ln|\mathcal{H}|+\ln\frac{1}{\delta_j}}{|\mathcal{F}|m} \leq c_{pool}$, where $c_{pool} := \frac{\ln|\mathcal{H}|+\ln\frac{1}{\min_{i\in[n]}\delta_i}}{m}$. Now we can define the *cost of collaboration* by $c^* = c_{pool}+\Pi$. Then, if Agent $i$ is using strategy $C$, its cost is given by $\frac{c^*}{|\mathcal{F}|}$. This means that Agent $i$ gets a proportional share $\frac{\Pi}{|\mathcal{F}|}$ of the bulk privacy risk, and also that its classifier's error bound is $\frac{\ln|\mathcal{H}|+\ln\frac{1}{\delta_i}}{|S_{pool}|}$, which is at most the 'proportional share' $\frac{c_{pool}}{|\mathcal{F}|}$ of the overall collaborators' error probability $c_{pool}$.

We represent this model by a fair cost sharing network game. We define the Opt In Opt Out network (OIOO) as in the below figure, where all players are trying to reach the "L" node: Each player $a_i$ has two choices - to pool their samples (choose $C$) or not to pool

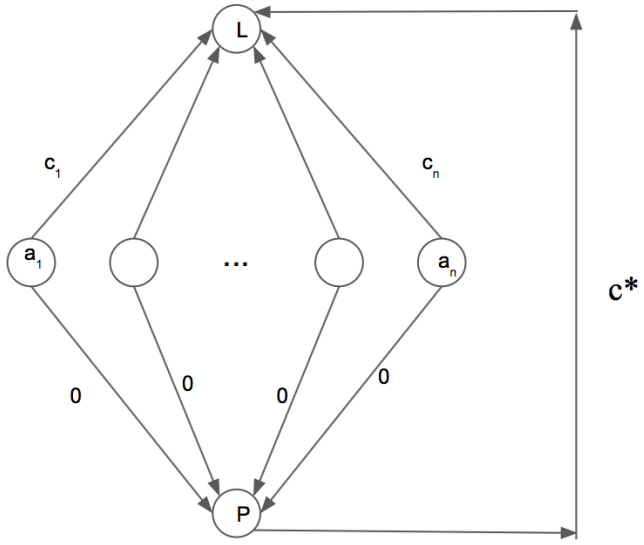


Figure 1: Fair Cost Sharing Network for OIOO

their samples (choose $N$). These correspond to the two paths Player $a_i$ can take to "L". The costs of each path are identical to that of our problem formulation.

Finally, we model the players decision making in this game as following the Learn-then-Decide Protocol as defined in the next section.

## 3. Game Dynamics

**Definition 2 (Learn-then-Decide Protocol, Balcan et al. (2013))** *This protocol is composed of an exploration phase, and an exploitation phase. In each round of the exploration phase, a player is chosen to move and flips a coin to decide whether to follow the proposed behavior $\mathcal{B}$ with probability $\geq \beta$ or follow their best response. After some time $T^*$, players switch to an exploitation phase where players commit arbitrarily to follow $\mathcal{B}$ or perform their best response from then on.*

In our learning setting we claim that it is natural for players to be following the protocol from Definition 2, as the protocol is actually a reasonable model of decision-making in this context. During the exploration phase of the Learn-then-Decide protocol, on every round an agent is chosen to either best respond to the current state with probability $\leq 1-\beta$, or try the proposed action with probability $\geq \beta$. $\beta$ can be thought of as an intrinsic parameter for a decision-maker describing how likely they are to take the suggested behavior when asked. Thus, we can think of the protocol as just modeling agent behavior rather than being forcibly imposed upon players.

**Remark 3** *Theorem 5 below is interesting because of the left hand side, and possible to prove because of the right hand side of the following inequality: $\sum_{i\in[n]} c_i > c^* > \max_i c_i$. The left side here is justified because already $c_j$ covers $c_{pool}$ where $j = \operatorname{argmax}_i c_i$, and as the number $n$ of players grows, the rest of the sum of the $c_i$s can be reasonably expected to cover also the bulk privacy risk $\Pi$ (taking into account that many of the error rate bounds $c_i$ can be assumed to be of the same order as $c_{pool}$). As for the right side of the inequality, it is obvious since $c^* = c_{pool} + \Pi = \max_i c_i + \Pi$.*

**Remark 4** *In our formulation of the game, we observe that if everyone chooses not to collaborate, the social welfare becomes $\sum_i c_i$. Because $c^* \geq \max_i c_i$, if all the $c_i$'s are the same, and $c^* = c_i + \Pi$ where $\Pi$ is a fixed constant, then asymptotically $\sum_i c_i = O(nc^*) = O(nOPT)$. Thus, the Non-Collaboration state has social cost a linear factor away from the $OPT$! This motivates the application of Theorem 5 to our game.*

**Theorem 5** *Suppose the proposed state $\mathcal{B}$ is the All-Collaborate state. After at most $O(n^3)$ steps, the social welfare of the game becomes at most $O(OPT)$, where $OPT = c^*$ is the social cost in $\mathcal{B}$. Moreover, this result still holds true if only a fixed constant fraction $\lambda \in (0, 1]$ of the $n$ players participate in the protocol while the rest arbitrarily commit to a single strategy at the beginning of the game and stay fixed.*

**Proof** We begin with introducing some notation. We define $\text{cost}_{i,t}$ as the cost player $i$ incurs at time $t$. We define $\text{cost}_t = \sum_i \text{cost}_{i,t}$. Now we proceed to our argument. First, we observe that by the coupon collector's problem, after time $T_0 = 2n \log n$, each player has been selected to move at least once w.h.p. Since $2n \log n = O(n^3)$ and the event happens w.h.p., we assume from now on that we start at such a state and index it by 0. Observe that

$\Phi_0 \leq n(1+\log n)c^*$. This is because if all players non-collaborate, $\text{cost}_0 = \sum_{i=1}^n c_i \leq nc^*$ since $c^* \geq \max_i c_i$, and otherwise, at least one player $j \in [n]$ collaborates, so $\text{cost}_0 \leq \sum_{i\in[n], i\neq j} c_i + c^* \leq nc^*$. Thus, $\Phi_0 \leq H_n \text{cost}_0 \leq (1+\log n)\cdot nc^*$ by Lemma 1.

Now, note that for all steps $[T_0, T_0+n^3]$, w.h.p. at least $\frac{\beta}{2}n$ players will have strategy $C$. This is because we can apply the following fact:

**Lemma 6 (Chernoff Bound for Sum of Bernoulli Variables)** *If $X_i \sim$ Bernoulli$(p_i)$ $\forall i \in [k]$, then letting $X = \sum_{i=1}^k X_i$, and $E[X] = \sum_{i=1}^k p_i$, it holds $\Pr[X \leq (1-\delta)E[X]] \leq e^{-\delta^2 E[X]/2}$ for any $\delta \in (0,1)$.*

For a fixed $t$, this lemma is applied to variables $X_i \sim \text{Bernoulli}(\beta)$ for $i \in [n]$, each $X_i$ corresponding to the event that after the last time Player $i$ was chosen to move before the current step and was asked to accept the proposed strategy. In fact, sometimes also the other option, to best respond, will also see the player switch to the proposed strategy, which only increases the probability of ending up in it. The total number of players on edge $c^*$ is $\sum_{i\in[n]} X_i$, so by Lemma 6 $\Pr[\sum_{i\in[n]} X_i \leq \frac{1}{2}\beta n] \leq e^{-(\frac{1}{2})^2\beta n/2} = e^{-\beta n/8}$. Union bounding over all $n^3$ events implies that w.h.p. for all $t \in [T_0, T_0+n^3]$, $\sum_{i\in[n]} X_i > \frac{\beta}{2}n$. If we assume this, then any chosen player $i$'s cost after its deviation, $E[\text{cost}_{i,t+1}]$ at any time $t$ is at most the expected cost of switching to the proposed strategy (best responding improves the cost by at least as much), which is at most $\frac{c^*}{\frac{\beta n}{2}}$. Hence, the expected social cost will be at most $\frac{1}{n}\sum_{i\in[n]} E[\text{cost}_{i,t+1}] \leq \frac{2}{\beta}\frac{c^*}{n}$.

Now, assume that until at least time $t^* = n^3$, the social cost never drops below $\frac{4}{\beta}c^*$. Then, for a random player $i \in [n]$ if chosen to move, its expected cost before the move is at least $\frac{4}{\beta}\frac{c^*}{n}$. Hence, the expected drop in the potential at any time $t$, if player $i_t \in [n]$ is chosen to move, is

$$E[\Phi_t - \Phi_{t+1}] = E[\text{cost}_{i_t,t}] - E[\text{cost}_{i_{t+1},t+1}] \geq \frac{4}{\beta}\frac{c^*}{n} - \frac{2}{\beta}\frac{c^*}{n} = \frac{2}{\beta}\frac{c^*}{n}. \tag{3.1}$$

Denote this last expression by $Q/n$. The idea here will be to show that after $n^3$ steps the social cost must drop below $\frac{4}{\beta}c^*$. We will define $\Delta_T = \max(\Phi_T - \Phi_{T-1} + \frac{Q}{n}, -2Q)$. Let $X_T$ be a stochastic process, where $X_T = \Phi_0 + \sum_{i=1}^T \Delta_i$ and we stop the process if $cost_T < 2Q$. From our construction, we know that the expected decrease in potential is at least $Q/n$ which tells us that $E[X_T|X_1, ..., X_{T-1}] \leq X_{T-1}$, and we also have that $|X_T - X_{T-1}| \leq 2Q$. This lets us apply Hoeffding-Azuma bounds for supermartingales, to conclude that after $n^3$ steps w.h.p we will have $X_T - X_0 \leq \frac{1}{2}n^2 Q$. By the way we have defined $X_T$ we know that the inequality

$$\Phi_T \leq \Phi_0 + (X_T - X_0) - \frac{TQ}{n}$$

holds which would imply that $\Phi_T$ would be negative. This is impossible so our process must have terminated prior to this point. Thus in $O(n^3)$ steps we achieve a social welfare of the game that is no greater than $\frac{4}{\beta}c^* = O(c^*)$

Moreover, if $(1-\lambda)n$ players fix their strategies (to either $C$ or $N$), the only difference in the past proof is that with high probability $\frac{\lambda\beta}{2}n$ players are collaborating at any time from $T_0$ to $T_0+n^3$. The rest of the proof follows with $\lambda\beta$ instead of $\beta$. ■

**Theorem 7 (Convergence of Protocol)** *For OIOO in the Learn then Decide model, a polynomial number of exploration steps $T^*$ is sufficient so that the expected total cost at any time $T' \geq T^*$ is $O(\log n \cdot OPT)$*

**Proof Sketch:** This proof is a modified version of one provided in Balcan et al. (2013). The idea is that if we can guarantee the existence of a $T_1$ where $cost_{T_1} = O(OPT)$, which is within a polynomial amount of rounds before $T^*$ (the time where the exploration phase ends and right before the exploitation phase begins), then the expected value of the potential at time $T^*$ is $O()$. From this, Balcan et al. (2013) proves that this means the expected total cost of the game any time after the decision-making time $T^*$ is $O(\log nOPT)$.

Thus the Protocol helps us avoid the situation in Remark 4 where the game could be in a state far from $OPT$. Note that players could decide whether to accept, at $T^*$, $\mathcal{B}$ or reject it using any learning algorithm of their choice - i.e. they could apply a two-experts algorithm to the protocol execution until time $T^*$.

## 4. Discussion and Interpretation, Conclusions

In this paper, we have presented a collaborative learning model, which is privacy-aware in the sense that agents in this model care about worst-case performance guarantees of their classifiers and the bulk privacy cost, rather than e.g. average-case performance. Of course, this model has some limitations that we hope future research can address.

Thus, we have assumed that the training sample sizes for all agents are equal: $m_i = m, \forall i \in [n]$. If this were not true, and the $m_i$ were at completely different scales, we would have to make our OIOO game instance *weighted*, that is, each player $i$ would carry weight $w_i$ inversely proportional to their sample size $m_i$ (cf. Redko and Laclau (2019) for a discussion on this). Then, we would run into difficulty proving the bound on the convergence of our protocol. Indeed, the proof of Theorem 5 assumes the existence of an *exact* potential function for our game. However, as shown more generally in Monderer and Shapley (1996), the only class of potential games with an exact potential are *unweighted* congestion games[3], which include unweighted, but exclude weighted fair cost sharing games. Weighted fair cost sharing games only have an approximate potential, which is given in Chen and Roughgarden (2008). However, that potential's drop on a single-player deviation is not equal to the decrease in that player's cost but rather weighs the old and the new cost of that player differently by a certain factor. This does not allow to lower-bound the potential drop through a player's cost difference as in Equation 3.1.

Also, the model assumes that there always exists a consistent hypothesis with any input. In the privacy context, this is a reasonable assumption. Without this assumption, the upper bound on the error of the classifier would not be inversely proportional to the sample size anymore. Thus, we would not be able to use *fair cost sharing* games to analyze the model. Instead, we would anticipate such analysis to use, more generally, a congestion game (see above for a definition).

3. A *congestion game*, more generally than a fair cost sharing game, allows the cost of any player taking any given edge to be an arbitrary positive-valued function of the number of players using that edge.

## References

Elliot Anshelevich, Anirban Dasgupta, Jon Kleinberg, Eva Tardos, Tom Wexler, and Tim Roughgarden. The price of stability for network design with fair cost allocation. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '04, pages 295–304, Washington, DC, USA, 2004. IEEE Computer Society. ISBN 0-7695-2228-9. doi: 10.1109/FOCS.2004.68. URL http://dx.doi.org/10.1109/FOCS.2004.68.

Maria-Florina Balcan, Avrim Blum, and Yishay Mansour. Circumventing the price of anarchy: Leading dynamics to good behavior. *SIAM Journal on Computing*, 42(1):230–264, 2013.

Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloe Kiddon, Jakub Konečný, Stefano Mazzocchi, H. Brendan McMahan, Timon Van Overveldt, David Petrou, Daniel Ramage, and Jason Roselander. Towards federated learning at scale: System design, 2019.

Ho-Lin Chen and Tim Roughgarden. Network design with weighted players. *Theory of Computing Systems*, 45(2):302, Jul 2008. ISSN 1433-0490. doi: 10.1007/s00224-008-9128-8. URL https://doi.org/10.1007/s00224-008-9128-8.

Steve Chien and Alistair Sinclair. Convergence to approximate nash equilibria in congestion games. *Games and Economic Behavior*, 71(2):315 – 327, 2011. ISSN 0899-8256. doi: https://doi.org/10.1016/j.geb.2009.05.004. URL http://www.sciencedirect.com/science/article/pii/S0899825609001110.

Jakub Konečný, H. Brendan McMahan, Felix X. Yu, Peter Richtarik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. In *NIPS Workshop on Private Multi-Party Machine Learning*, 2016. URL https://arxiv.org/abs/1610.05492.

Dov Monderer and Lloyd S. Shapley. Potential games. *Games and Economic Behavior*, 14(1):124 – 143, 1996. ISSN 0899-8256. doi: https://doi.org/10.1006/game.1996.0044. URL http://www.sciencedirect.com/science/article/pii/S0899825696900445.

Ievgen Redko and Charlotte Laclau. On Fair Cost Sharing Games in Machine Learning. In *AAAI Conference on Artificial Intelligence*, Honolulu, United States, January 2019. URL https://hal.archives-ouvertes.fr/hal-02051399.

Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning. *ACM Transactions on Intelligent Systems and Technology*, 10(2):1–19, Jan 2019. ISSN 2157-6904. doi: 10.1145/3298981. URL http://dx.doi.org/10.1145/3298981.