Safe Subgame Resolving for Extensive Form Correlated Equilibrium

Chun Kai Ling, Fei Fang
Carnegie Mellon University
chunkail@cs.cmu.edu, feif@cs.cmu.edu

Carnegie Mellon University

1. Motivation

- It is known that in 2-player general sum games, **extensive-form correlated equilibrium** (EFCE) can lead to higher social welfare (SW) [1, 2]. Players in the benchmark game *Battleship* can be incentivized by a centralized mediator to deliberately avoid shooting at their opponent, leading to peaceful outcomes.
- EFCE is a superset of CE. Players only receive recommendations for the information set they are currently in. Players who deviate from recommendations no longer get recommendations for the rest of the game.
- **Computationally difficult**. NP-hard to find SW maximizing EFCE. In games *without chance*, can be done in polynomial time, though quadratic in the size of the game tree. Example: Battleship on a board of size 3x2, time horizon of 4, and a single ship of size 2x1 has a correlation plan of size > 100M.

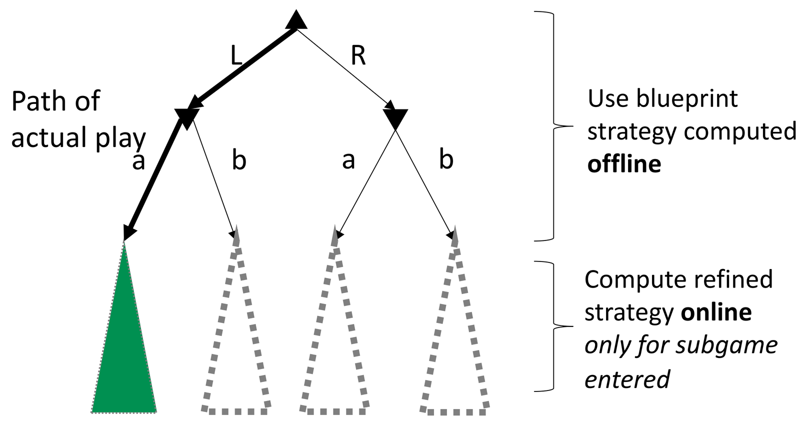
2. Subgame Resolving

- A crucial component of successful bots is **subgame resolving**, or search. In perfect information games (e.g, chess), one applies search *online* in actual play. Resolving is only initiated from states encountered in *actual play*
- Extends to imperfect information games (aka continual resolving, search). Notable success in *zero-sum* game solvers (*Libratus, DeepStack* [4, 5]).
- Limited success outside of zero-sum or cooperative games, with some initial work in applying general-sum Stackelberg extensive form games [6].
- Follow **blueprint** (typically from a simple abstraction of the original) strategy, computed offline in at the start of the game. Upon entering a subgame, a **refinement** is computed online *only for the subgame entered.*
- Refinements can be **unsafe**: Performing resolving based on initial state distributions (of the subgame) of the blueprint can be counterproductive.
  - Since players know that refinement would be performed upon entering subgame, they can respond to refinement *even before entering the subgame.*

Algorithm 1: Subgame Resolving

Input: EFG, blueprint ξ_0

```
while game is not over do
    if currently in some subgame j then
        if first time in subgame then
            (*) Refine ξ_0 → ξ̃_j
        end if
        Recommend action according to ξ̃_j
    else
        Recommend action according to ξ_0
    end if
end while
```

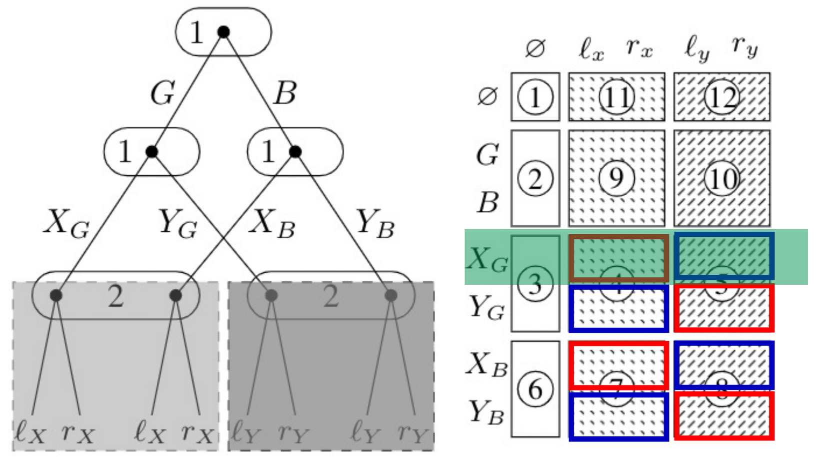


3. Our Contributions

- First resolving algorithm for EFCE
  - Polytope of correlation plans Ξ does not have a clear hierarchical structure.
  - We "divide" Ξ into subgames and show that there is sufficient independence between each partial correlation plan to perform refinements independently
- Define notions of safety for EFCE
  - We play the role of a mediator and seek to (i) improve social welfare and (ii) reduce exploitability. Refinements are safe if applying resolving to every subgame gives a refined strategy that outperforms the blueprint in SW and exploitability.
- Propose 2 algorithms to achieve safe resolving

4. Partial Correlation Plans and Refinements

- Assume game is 2 player, has no chance, and perfect recall
  - Ξ can be represented by a 2D grid indexed by sequence pairs (σ_1, σ_2), where sequence form constraints are obeyed by each row and column. [1]



$$\xi(\emptyset,\emptyset) = 1$$

$$\xi(\emptyset,\emptyset) = \xi(G,\emptyset) + \xi(B,\emptyset)$$
$$\xi(G,\emptyset) = \xi(X_G,\emptyset) + \xi(Y_G,\emptyset)$$
$$\xi(B,\emptyset) = \xi(X_B,\emptyset) + \xi(Yz_B,\emptyset)$$

Sequence form constraints for first col.

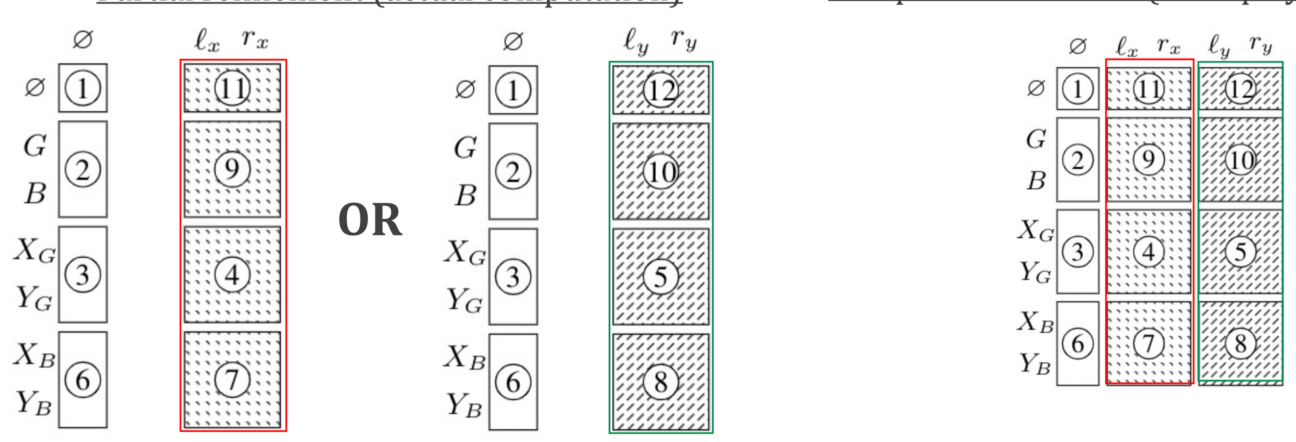
$$\xi(\emptyset,\emptyset) = \xi(\emptyset,\ell_x) + \xi(\emptyset,r_x)$$
$$\xi(\emptyset,\emptyset) = \xi(\emptyset,\ell_y) + \xi(\emptyset,r_y)$$

Sequence form constraints for first row.

If P1 was recommended X_G and is considering deviating to Y_G, it will consider probability that P2 plays ℓ_y, r_y (given in blue)

- Polytope of partial correlation plans Ξ_j
  - Contains σ_1, σ_2, both belonging to subgame j or occur before any subgame.
  - Valid refinement $\tilde{\xi}_j$: if σ_1, σ_2 occur before subgame, $\tilde{\xi}_j[\sigma_1, \sigma_2] = \xi_0[\sigma_1, \sigma_2]$, i.e., we cannot change what has happened in the past once inside a subgame.
  - Sequence form constraints are the same as Ξ (where the sequence pair exists)
- Partial correlation plans are close to independent from each other
  - For valid refinements of blueprint ξ_0, sequence form constraints do not "intersect"
- When performing refinements, we need to only consider valid refinements in Ξ_j
  - The fully refined strategy $\tilde{\xi} \in \Xi$ that players see when considering deviations is obtained by "piecing together" partial refinements.
  - Note: we *do not* explicitly compute full refinements. However, we will need to *reason* about the social welfare and exploitability of it in order to guarantee safety.

Partial refinement (actual computation) — Complete refinement (what players "see")



5. Refinement Algorithms

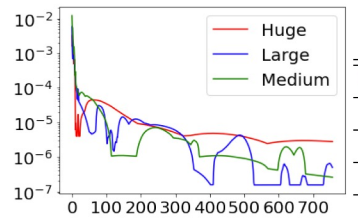
- Phase 1: compute bounds on player payoffs which guarantee safety
  - For each recommended sequence not belonging to a subgame, we compute (i) upper-bounds on how well a player does upon deviating and (ii) lower bounds on how much a player gets if it abides to this (and all future) recommendations
  - Obeying these bounds ensure that exploitability is no greater than the blueprint
  - Uses a method similar to prior work by Ling and Brown, used in Stackelberg games [6]
  - Blueprint satisfies these bounds trivially.
- Phase 2: Find valid refinement in Ξ_j which respects these bounds
  - Method 1: Builds off the Linear Programming method first proposed by Von Stengel [1]. Bounds are enforced by adding them directly as linear constraints. Having a higher social welfare follows by putting it as the objective to be maximized.
  - Method 2: Builds off a newer regret-minimization method based on self play [3] between deviator and mediator. Bounds enforced by expanding the set of deviators. Perform binary search to achieve a social welfare no worse than blueprint.
  - Both methods: Safety for deviating sequences *within* subgame is handled by the original

6. Experiments

- Evaluated our method on the *Battleship* benchmark game
  - Blueprints: (i) the uniform, independent blueprint, and (ii) a jittered alternative.
  - Subgames begin after 1st round of shooting.
- Experiment 1 (left). Maximize social welfare using LP solver
  - Significant improvement in social welfare for both blueprints
- Experiment 2 (right). Minimizing exploitability only using regret minimization

n, T, J	$\lvert\Xi_j\rvert$	γ	*Uniform* BP	*Uniform* Refined	*Jittered* BP	*Jittered* Refined
3, 2, 9	382	2	-3.70	-3.70	-3.55	-3.55
		5	-14.8	-14.8	-14.2	-14.2
4, 3, 16	3.2e3	2	-3.13	-2.95	-3.24	-3.10
		5	-12.5	-11.4	-13.0	-11.8
5, 3, 25	2.3e4	2	-1.92	-1.34	-1.95	-1.25
		5	-7.68	-4.80	-7.82	-4.32
6, 3, 36	1.2e5	2	-1.23	-.772	-1.25	-.627
		5	-4.94	-2.47	-4.99	-1.95

Table 1: Comparison of social welfare between blueprint (BP) and SW-maximizing safe refinement with ships of size 1. Social welfare is reported at a scale of 1e-2.



	Med.	Large	Huge
W, H	3,2	3,2	3,2
T	4	4	5
m	1	2	2
$\lvert\Xi\rvert$	3.89M	111M	360M

Figure 2: Left: Most violated incentive constraint of $\tilde{\xi}$ plot against iteration number. Right: Parameters of game.

7. Future Work

- Extensions to other forms of correlated behavior
- Explore possibility of learning values (as with *DeepStack*)

[1] Von Stengel, B.; and Forges, F. 2008. Extensive-form correlated equilibrium: Definition and computational complexity. Mathematics of Operations Research, 33(4): 1002–1022.
[2] Farina, G.; Ling, C. K.; Fang, F.; and Sandholm, T. 2019a. Correlation in Extensive-Form Games: Saddle-Point Formulation and Benchmarks. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E. and Garnett, R., eds., Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc.
[3] Farina, G.; Ling, C. K.; Fang, F.; and Sandholm, T. 2019b. Efficient Regret Minimization Algorithm for Extensive-Form Correlated Equilibrium. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E.; and Garnett, R., eds., Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc.
[4] Brown, N.; and Sandholm, T. 2018. Superhuman AI for heads-up no-limit poker: Libratus beats top professionals. Science, 359(6374): 418–424.
[5] Moravčík, M., Schmid, M., Burch, N., Lisý, V., Morrill, D., Bard, N., ... & Bowling, M. (2017). Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, *356*(6337), 508-513.
[6] Ling, C. K.; and Brown, N. 2021. Safe Search for Stackelberg Equilibria in Extensive-Form Games. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 35, 5541–5548.

This work was supported in part by NSF grant IIS-2046640 (CAREER) and a research grant from Lockheed Martin.