Outcome, external & hybrid Sampling for Regret Minimization for Kuhn & Leduc poker

Felix Cammaerts, r0663453
felix.cammaerts@student.kuleuven.be

April 29, 2021

Abstract

This paper discusses the work done to research the performance of outcome and external sampling of Monte Carlo CFR minimizing algorithms introduced in Lanctot, Waugh, Zinkevich, and Bowling (2009). We firstly look at the difference in performance between vanilla CFR and Monte Carlo CFR using either external or outcome sampling for the games of Kuhn poker and Leduc poker. Secondly we introduced a hybrid sampling mechanism consisting of a probability factor α which decides in every iteration whether outcome or external sampling is used. This is also applied on the games of Kuhn poker and Leduc poker. ***Keywords:*** *openspiel; CFR; Monte Carlo CFR; outcome sampling; external sampling*

1 Introduction

In Lanctot, Waugh, Zinkevich, and Bowling (2009) an extension of Counterfactual regret minimization (CFR) is made by introducing a family of Monte Carlo CFR minimizing algorithms (MCCFR). In that paper two sampling approaches are used namely outcome and external sampling. In Lanctot (2013) there is also a third sampling approach named Public Chance Sampling, however we will not consider that sampling approach here. Both outcome and external sampling have shown promising results for the games of One card poker, Goofspiel, Latent Tic Tac Toe, Princess and Monster and Bluff when they have been compared to the results of vanilla CFR as researched in Lanctot, Waugh, Zinkevich, and Bowling (2009) and Lanctot, Waugh, and Bowling (2009). We will check if these improved results also apply for Kuhn and Leduc poker. As One card poker is a generalization of Kuhn poker, it could be expected that this is the case. However it could be that vanilla CFR still outperforms MCCFR in this particular case as it is possible that Kuhn poker is better solved using vanilla CFR. For instance think about how a genetic algorithm is able to solve a lot of problems decently well but is completely outperformed by specific algorithms for specific problems, such as Newton Raphson for convex optimization (Lingaraj (2016)).

Once we have these results we will experiment with a hybrid sampling method consisting of both outcome and external sampling. This is achieved by using a probability factor α at each iteration.

2 Methodology

To experiment we use the OpenSpiel framework Lanctot et al. (2019) in which we wrote all our code in C++, this ensured that the execution time of our scripts was fast as C is

in general faster than Python. This also meant we would not be bound by the Python bindings which means that we will have easier access to all functionality of the framework. The graphs are still plotted in Python using the Matplotlib framework.

2.1 Hybrid sampling

We introduce a new sampling scheme that we will call hybrid sampling. This sampling scheme consists of a parameter α, if this factor is set to 0.5 then both external and outcome sampling have an equal chance of being selected. If α is set to 1 then every iteration will be done using outcome sampling, if α is equal to 0 then every iteration is done using external sampling. It is important to notice that a value of 0.5 for α does not mean that external and outcome sampling will be selected exactly the same amount of times, α is only the threshold for a random number generator, this means that it is possible that more external or outcome sampling iterations will occur.

2.2 Hypothesis

Our hypothesis consists of two parts:

- Kuhn and Leduc poker have a better NashConv and exploitability when using external and outcome sampling when compared to vanilla CFR.
- Our proposal for a hybrid sampling scheme also improves the NashConv and exploitability metrics for Kuhn and Leduc poker compared to the pure external and outcome sampling schemes.

2.3 Benchmarks & evaluation metrics

As mentioned in our hypothesis, we will be using the NashConv and exploitability metrics to evaluate our algorithms. The optimal value for both of these metrics is 0. We will be plotting them in function of the amount of touched nodes, which means that we will be able to see the evolution of these metrics during learning time. The fact that we use the amount of touched nodes means that we are evaluating them in a application independent manner as mentioned in Lanctot, Waugh, Zinkevich, and Bowling (2009). We will also compare the total evaluation time for the different methods.

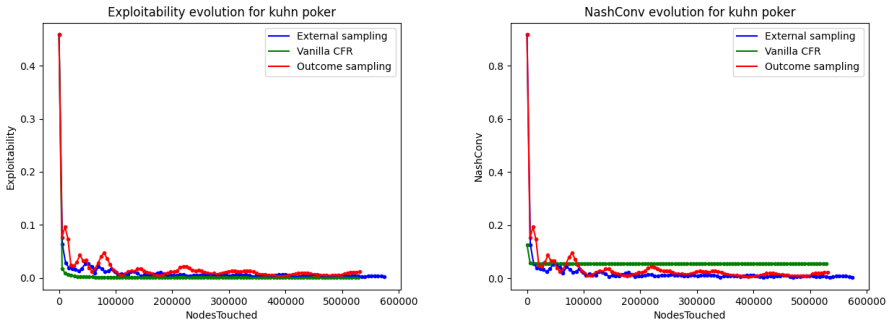
2.4 Parameter tuning

It is also possible to tune the α parameter of hybrid sampling. A lower value for α will mean that outcome sampling will be selected more often, while a higher value of α means that external sampling will be selected more often.

3 Results

3.1 Vanilla CFR vs outcome and external sampling

As can be seen in Figure 1 for both the exploitability and NashConv metrics vanilla CFR is fast to converge. For the exploitability both outcome and external sampling have not reached 0 exploitability after 600.000 touched nodes while vanilla CFR was already able to do this after around 30.000 touched nodes. The NashConv for vanilla CFR converges to $\frac{1}{18} = 0.055$ which is equal to the NashConv of Kuhn poker. Both outcome and external sampling are unable to do so in the same amount of nodes touched and end up having a lower NashConv.

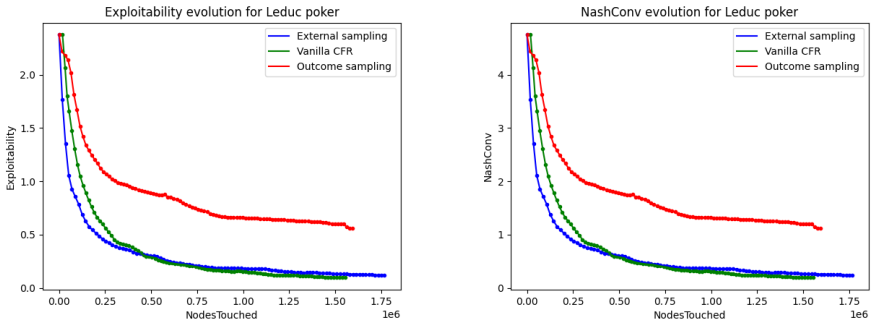


(a) Exploitability of vanilla CFR, external sampling and outcome sampling plotted in function of the amount of touched nodes.

(b) NashConv of vanilla CFR, external sampling and outcome sampling plotted in function of the amount of touched nodes.

Figure 1: Results for Kuhn poker of vanilla CFR vs outcome and external sampling.

For Leduc poker as shown in Figure 2 we see that the different methods bring along very different results. The NashConv and exploitability of both vanilla CFR and external sampling have almost converged to the same value for NashConv while outcome sampling is still far away from that mutual convergence point.



(a) Exploitability of vanilla CFR, external sampling and outcome sampling plotted in function of the amount of touched nodes.

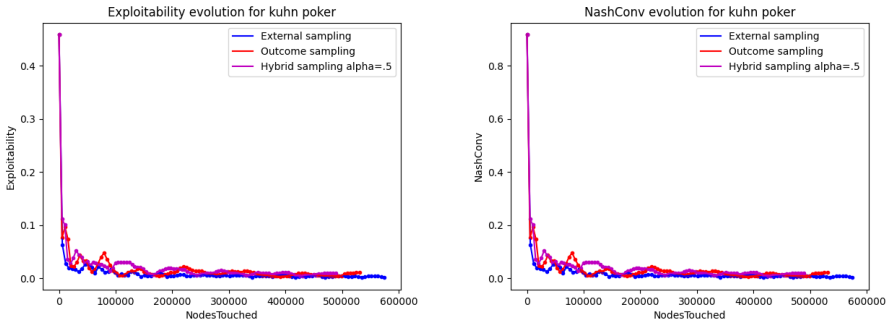
(b) NashConv of vanilla CFR, external sampling and outcome sampling plotted in function of the amount of touched nodes.

Figure 2: Results for Leduc poker of vanilla CFR vs outcome and external sampling.

3.2 Hybrid sampling vs outcome and external sampling

We now compare the results of our newly proposed sampling scheme, hybrid sampling, against the results of outcome and external sampling.

For the game of Kuhn poker all 3 sampling schemes evolve in the same manner with seemingly little difference. As can be seen in Figure 3.

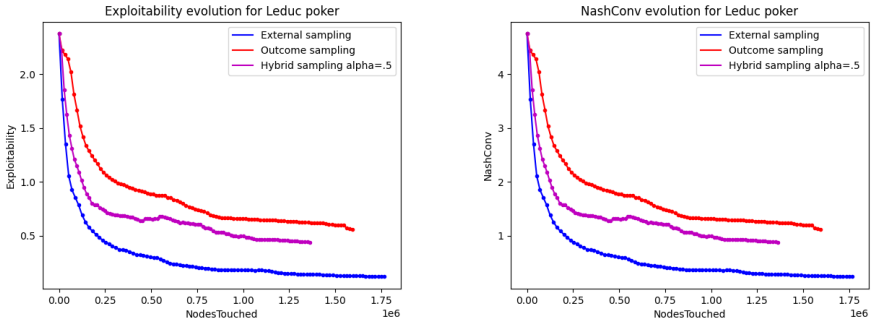


(a) Exploitability of hybrid sampling, external sampling and outcome sampling plotted in function of the amount of touched nodes.

(b) NashConv of hybrid sampling, external sampling and outcome sampling plotted in function of the amount of touched nodes.

Figure 3: Results for Kuhn poker of hybrid sampling vs outcome and external sampling.

When looking at Leduc poker it however becomes clear that hybrid sampling converges slower than external sampling and faster than outcome sampling.



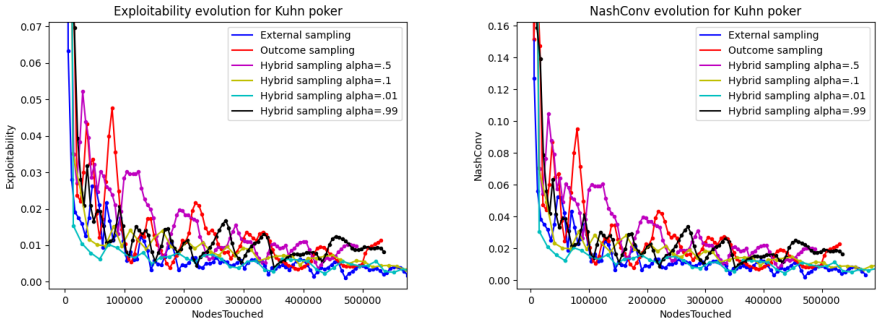
(a) Exploitability of hybrid sampling, external sampling and outcome sampling plotted in function of the amount of touched nodes.

(b) NashConv of hybrid sampling, external sampling and outcome sampling plotted in function of the amount of touched nodes.

Figure 4: Results for Leduc poker of hybrid sampling vs outcome and external sampling.

3.3 Optimal value for α

We can now also see how different values of α influence the evolution of the exploitability and NashConv metrics. For Kuhn poker we see in Figure 5 that in the beginning of learning the value of 0.01 yields the lowest NashConv and exploitability. This is later however overtaken by the pure external sampling scheme.



(a) Exploitability of hybrid sampling, external sampling and outcome sampling, for different values of α.

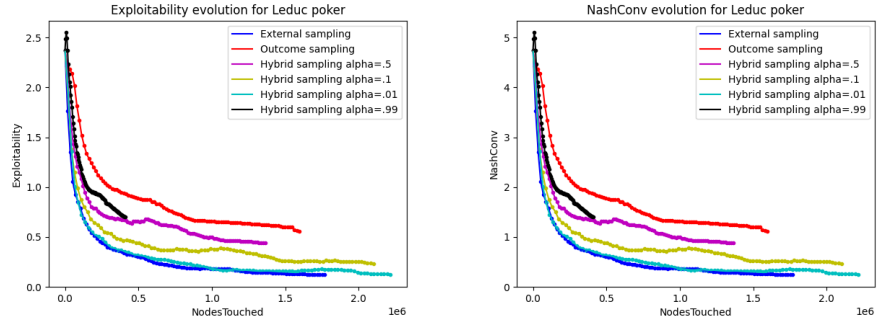
(b) NashConv of hybrid sampling, external sampling and outcome sampling, for different values of α.

Figure 5: Results for Kuhn poker of hybrid sampling vs outcome and external sampling.

However if we take a look at Leduc poker we see that all hybrid schemes remain between the pure sampling schemes during learning for the NashConv and exploitability (Figure 6). It is also very clear how higher α values let the hybrid scheme be closer to outcome sampling whilst for lower values of α, they are closer to external sampling.

3.4 Runtime

Lastly we compare the different run times for each of the different schemes. The results can be found in table 1. We see that Leduc poker always has a longer running time than Kuhn poker which is to be expected as Leduc poker is a more complex game consisting of more nodes. For Kuhn poker pure outcome sampling has the fastest run time per iteration, while the longest run time for Kuhn poker is when using pure external sampling. When looking at Leduc poker the opposite is true however: the run time per itaretion is shorter for external sampling than for outcome sampling. In the hybrid case a lower value for α brings around a faster run time per iteration for Leduc poker. The worst run time per iteration for Leduc poker is found when using vanilla CFR.



(a) Exploitability of hybrid sampling, external sampling and outcome sampling, for different values of α.

(b) NashConv of hybrid sampling, external sampling and outcome sampling, for different values of α.

Figure 6: Results for Leduc poker of hybrid sampling vs outcome and external sampling.

Sampling scheme	α	Game	Running time (s)	Iterations	Time per iteration
External	-	Kuhn poker	2.22944	40,000	0.00005
External	-	Leduc poker	130.238	40,000	0.00326
Hybrid	0.01	Kuhn poker	4.47133	100,000	0.00004
Hybrid	0.01	Leduc poker	129.778	100,000	0.00130
Hybrid	0.1	Kuhn poker	4.29696	100,000	0.00004
Hybrid	0.1	Leduc poker	129.398	100,000	0.00130
Hybrid	0.5	Kuhn poker	3.32898	100,000	0.00003
Hybrid	0.5	Leduc poker	123.786	100,000	0.01238
Hybrid	0.99	Kuhn poker	2.2022	100,000	0.00002
Hybrid	0.99	Leduc poker	116.207	100,000	0.01162
Outcome	-	Kuhn poker	2.29383	100,000	0.00002
Outcome	-	Leduc poker	1222.7253	100,000	0.01227
- (vanilla)	-	Kuhn poker	3.30197	10,000	0.00003
- (vanilla)	-	Leduc poker	1120.57	10,000	0.11206

Table 1: Running time of external, outcome and hybrid sampling as well as vanilla CFR. For hybrid sampling different values for α are showcased.

4 Discussion

We now have a look at our hypotheses again and compare them to our results. The first hypothesis stated that the NashConv and exploitability metrics improve with external and outcome sampling compared to vanilla CFR. As for Kuhn poker there is no big difference in these metrics for the sampling schemes and CFR, so we reject the hypothesis. However for Leduc poker the early iterations of external sampling yield better results than vanilla CFR, unfortunately we cannot say the same about outcome sampling. We therefor also reject this hypothesis as it states that the sampling schemes perform better than both pure sampling schemes.

Our second hypothesis stated that a hybrid sampling scheme improves the NashConv and exploitability metrics for the two games compared to outcome and external sampling. For Kuhn poker we do not reject this hypothesis as a value of 0.01 for α gives a scheme that in the early stages has a lower NashConv and exploitability than external and outcome sampling. However we do reject the hypothesis for Leduc poker as no value of α was able to reduce the NashConv and exploitability metrics compared to external and outcome sampling.

A value of 0.99 for α in hybrid sampling gives a faster execution time for Kuhn poker than the pure external sampling scheme, however this also reduces the NashConv and exploitability, leading to a less optimal solution. For Leduc poker the values of 0.01 and 0.1 seem to be promising to obtain a solution of decent quality (still close to the NashConv/exploitability of external sampling) while significantly decreasing the computation time per iteration.

5 Conclusion

This paper had two main purposes:

- Compare the performance of external and outcome sampling to vanilla CFR for the games of Kuhn and Leduc poker.
- Compare the performance of a newly introduced hybrid scheme which is a blend of external and outcome sampling to the pure sampling schemes.

For Kuhn poker no significant improvement was found when using external or outcome sampling, for Leduc poker an improved performance was found when using external sampling, however this only holds true in the early iterations of learning.

The newly proposed hybrid scheme uses both outcome and external sampling in different iterations depending on a parameter α, the higher the value of α, the higher the chance of outcome sampling being selected as sampling scheme and vice versa. When applied to Leduc poker all values for α give a result somewehere between the two pure sampling schemes. For Kuhn poker a slight improvement over external and outcome sampling is found with a value of 0.01 for α in the early iterations.

In further research the new sampling scheme could be applied to larger games than the two games used in this paper. This is likely to yield more interesting results as the pure sampling schemes were introduced to provide a significant improvement in these larger games.

I spent 28 hours on the project.

References

Lanctot, M. (2013). *Monte Carlo sampling and regret minimization for equilibrium computation and decision-making in large extensive form games* (Unpublished doctoral dissertation). University of Alberta, University of Alberta, Computing Science, 116 St. and 85 Ave., Edmonton, Alberta T6G 2R3.

Lanctot, M., Lockhart, E., Lespiau, J.-B., Zambaldi, V., Upadhyay, S., Pérolat, J., ... Ryan-Davis, J. (2019). *Openspiel: A framework for reinforcement learning in games.*

Lanctot, M., Waugh, K., & Bowling, M. (2009). Monte Carlo sampling for regret minimization in extensive games..

Lanctot, M., Waugh, K., Zinkevich, M., & Bowling, M. (2009). Monte Carlo sampling for regret minimization in extensive games. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, & A. Culotta (Eds.), *Advances in neural information processing systems 22* (pp. 1078–1086).

Lingaraj, H. (2016, 10). A study on genetic algorithm and its applications. *International Journal of Computer Sciences and Engineering*, *4*, 139-143.