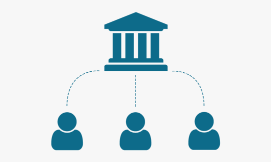
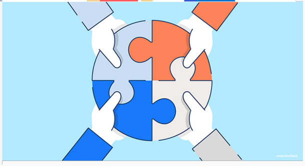
# EEF1-NN: Efficient and EF1 allocations through Neural Networks

**Shaily Mishra** Manisha Padala Sujit Gujar

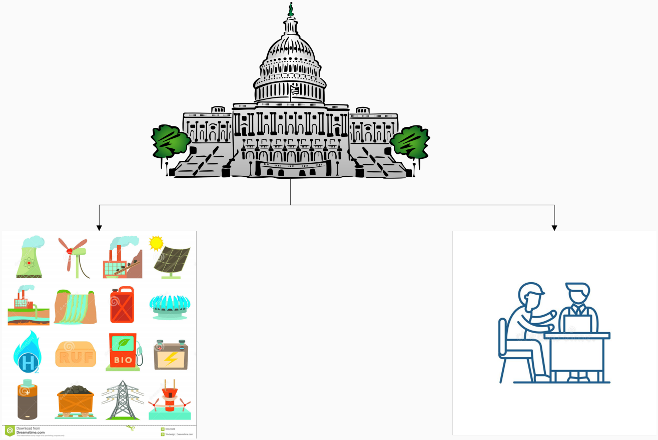
International Institute of Information Technology, Hyderabad

The 19th Pacific Rim International Conference on Artificial Intelligence (PRICAI 2022)

# Fair Division

# Fair Division



# Fair and Efficient Division

Fairness

Efficiency

## One of the most popular fairness - EnvyFreeness (EF)

# EF may not exist

# Does it exist?

$\Big\downarrow$ relaxation

??

**Envy-freeness up to one item (EF1)**

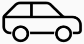
An allocation $A$ is said to be EF1 if envy of any agent can be eliminated by either removing some good from the envied agent's allocation or removing some chore from the agent's allocation

**Definition (Envy-free (EF) and relaxations)**

An allocation $A$ that satisfies $\forall i, j \in N$,

$$v_i(A_i) \geq v_i(A_j) \text{ is EF}$$
$$v_i(A_i \setminus \{k\}) \geq v_i(A_j \setminus \{k\}); \exists k \in \{A_i \cup A_j\} \text{ is EF1}$$

# EF1

| |  |  |  |
|---|---|---|---|
| Agent 1 | 6 | 6 | 4 |
| Agent 2 | 6 | 6 | 4 |

# EF1

| |  |  |  |
|---|---|---|---|
| Agent 1 | 6 | 6 | 4 |
| Agent 2 | 6 | 6 | 4 |

~~EF~~

EF1

**Complexity**

Finding EF1 Allocation is polynomial-time
[Lipton et al., 2004, Bhaskar et al., 2020]

# One of the most popular efficiency - Utilitarian Social Welfare USW

Sum of utilities of individual agents

# Our Goal

Maximizing USW within EF1 allocations, i.e., EEF1 - efficient and envy-free up to one item, which is an NP-hard problem, even when valuations are additive for two agents

# Key Results for EEF1

| Items | Valuation | Property | Results | Complexity |
|---|---|---|---|---|
| any | additive | $\epsilon$-Efficient and $\mathcal{F}$-Fair | [Bredereck et al., ] | Double exponential in terms of $n$ and $m$ |
| goods | additive | EEF1 | [Aziz et al., 2020] | Exponential in $n$ and polynomial in $m$ and $V$ ,where $V$ bounds the valuation per item |
| any | general | EF1 | [Lipton et al., 2004, Bhaskar et al., 2020] | polynomial in terms of $n$ and $m$ |
| goods | additive | EF1 with increased USW (WCRR) | [Lipton et al., 2004, Bhaskar et al., 2020] | polynomial in terms of $n$ and $m$ |
| any | additive | MUW | | polynomial in terms of $n$ and $m$ |

**Table 1:** Existing results for EEF1

# Our Approach

## Contribution

- We propose a neural network EEF1-NN inspired by U-Net to learn EEF1.

## Contribution

- We propose a neural network EEF1-NN inspired by U-Net to learn EEF1.
- We transform our valuations and augment them with additional channels to enhance the network's performance.

## Contribution

- We propose a neural network EEF1-NN inspired by U-Net to learn EEF1.
- We transform our valuations and augment them with additional channels to enhance the network's performance.
- We use a series of convolutional and up-convolutional layers to learn EEF1; EEF1-NN is generalized for any number of agents and items.

## Contribution

- We propose a neural network EEF1-NN inspired by U-Net to learn EEF1.
- We transform our valuations and augment them with additional channels to enhance the network's performance.
- We use a series of convolutional and up-convolutional layers to learn EEF1; EEF1-NN is generalized for any number of agents and items.
- We sample valuations from various distributions and report the expected fairness and efficiency achieved. Even for large instances, our network performs well.
- We show that, for our setting, bagging of networks improves performance.

# Formulation of Optimization Problem

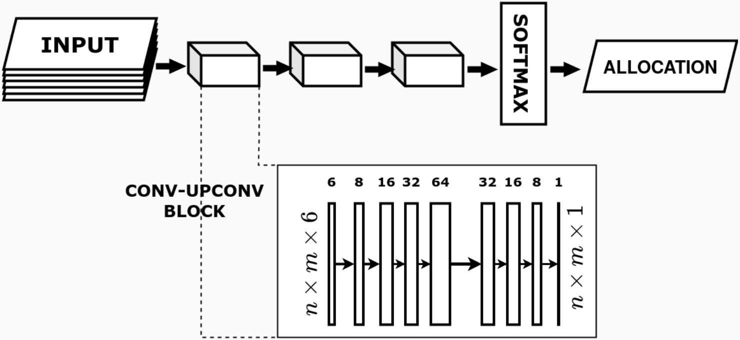
Our goal is to maximize the expected welfare w.r.t. to the expected fairness.

$$\begin{aligned} \text{minimize} \quad & -\mathbb{E}_v\left[sw(A,v)\right] = \mathbb{E}_v[\sum_{i \in N} v_i(A_i)] \\ \text{subject to} \quad & \mathbb{E}_v\left[\sum_{i \in N} envy_i(A,v)\right] = 0 \end{aligned}$$

# Lagrangian Loss Function

$$Loss(I_v^l, w, \lambda) = \left[ - sw(\mathcal{A}^w(I_v^l), v^l) + \lambda \frac{\sum_{i \in N} envy_i(\mathcal{A}^w(I_v^l), v^l)}{n} \right]$$

# EEF1-NN architecture



# Training Challenges

- **Ensuring Integral Allocations** : We set a temperature parameter $T$ in the softmax layer of the network to prevent getting stuck at such optima.
- Let $o_j = \{o_{j_1}, \ldots, o_{j_n}\}$ denote the output of our network before the final layer. The final allocation for agent $i$ is given by, $\mathcal{A}_i^w(\{j\}) = \text{softmax}(o_{j_i}) = \frac{e^{o_{j_i}/T}}{\sum_{k=1}^{n} e^{o_{j_k}/T}}$ which represents the probability of assigning item $j$ to all the agents.

## Training Challenges

- **Inefficient Local Optima** : To overcome this, we use the technique of Bootstrap Aggregation or Bagging.

# Experimentation Results - Ablation Study

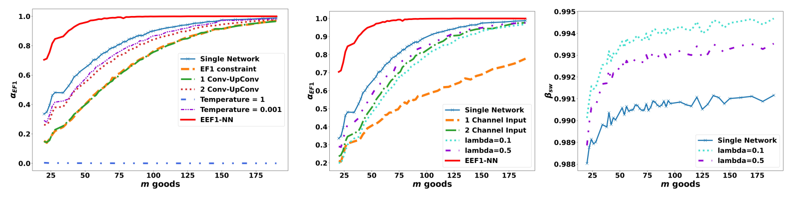



**Figure 1:** Ablation Study over hyper-parameters

# Experimentation Results

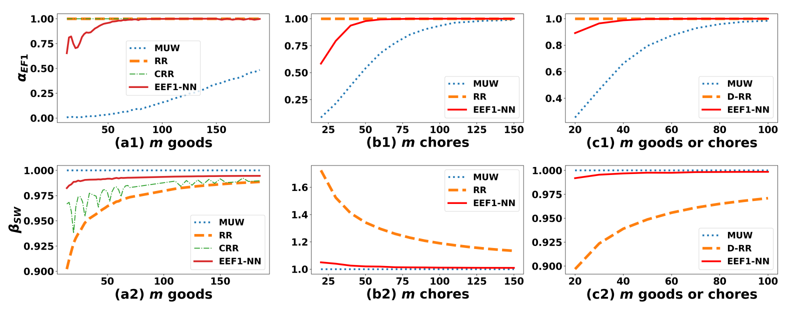



**Figure 2:** $n = 10$ uniform distribution - goods, chores, combination

# Experimentation Results

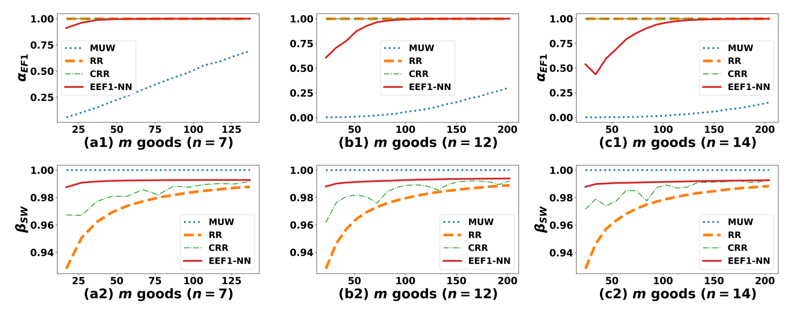



**Figure 3:** Scaling of EEF1-NN

# Experimentation Results

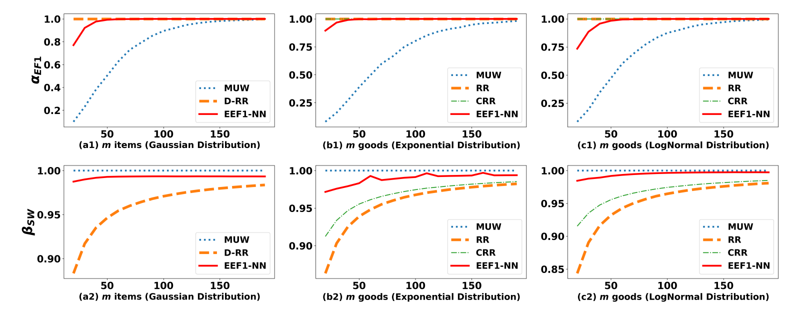



**Figure 4:** Different Distributions

## Analysis of Convergence to EEF1 Allocations (Uniform Distribution)

| $n$ | $(m)$ Goods | | | $(m)$ Chores | | |
|---|---|---|---|---|---|---|
| | EEF1-NN | RR | MUW | EEF1-NN | RR | MUW |
| 7 | 38 | 159 | 380 | 44 | 195 | 112 |
| 8 | 46 | 172 | 450 | 44 | 240 | 120 |
| 9 | 57 | 186 | 530 | 53 | 295 | 130 |
| 10 | 70 | 196 | 610 | 60 | 340 | 148 |
| 11 | 82 | 206 | 660 | 68 | 400 | 160 |
| 12 | 94 | 214 | 740 | 75 | 455 | 167 |
| 13 | 110 | 220 | 840 | 83 | 505 | 180 |
| 14 | 134 | 228 | 940 | 87 | 565 | 190 |

## To conclude

- We proposed a neural network EEF1-NN to find EEF1, an NP-hard problem.
- We designed architecture and input representation combined with other training heuristics to learn approximate EEF1 on average.
- We studied the effect of each proposed constituent on performance.
- Our experiments demonstrated the efficacy of EEF1-NN for different input distributions across various $n$ and $m$ over existing approaches.
- We believe that the path of amalgamating deep learning and resource allocation is worth exploring further with more complex objective functions.

# References

- Aziz, H., Huang, X., Mattei, N., and Segal-Halevi, E. (2020).
  **Computing fair utilitarian allocations of indivisible goods.**

- Bhaskar, U., Sricharan, A., and Vaish, R. (2020).
  **On approximate envy-freeness for indivisible chores and mixed resources.**
  *arXiv preprint arXiv:2012.06788*.

Bredereck, R., Kaczmarczyk, A., Knop, D., and Niedermeier, R.

**High-multiplicity fair allocation: Lenstra empowered by n-fold integer programming.**

In *Proceedings of the 2019 ACM Conference on Economics and Computation*, EC '19, page 505–523.

Lipton, R. J., Markakis, E., Mossel, E., and Saberi, A. (2004).

**On approximately fair allocations of indivisible goods.**

In *Proceedings of the 5th ACM Conference on Electronic Commerce*, EC '04, page 125–131.

# Thank You

# Questions?

Machine Learning Lab, IIIT

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD