Binary MNW Paper

First Author[1], **Second Author**[2], **Third Author**[2,3] and **Fourth Author**[4]
[1]First Affiliation
[2]Second Affiliation
[3]Third Affiliation
[4]Fourth Affiliation
{first, second}@example.com, third@other.example.com, fourth@example.com

1 Introduction

2 Model

3 Properties of binary Maximum Nash Welfare

Theorem 1. *When valuations are binary, for a fixed allocation* $\mathbf{A}$*, the following are equivalent:*

1. $\mathbf{A}$ *is a Maximum Nash Welfare allocation.*
2. $\mathbf{A}$ *is a Leximin allocation.*
3. $\mathbf{A}$ *is a possible allocation for a welfare function that satisfies the Pigou-Dalton principle in the strong sense.*

   (Alternatively, we can define something like a Pigou-Dalton improvement and say no other allocation dominates $\mathbf{A}$*)*

Theorem 2. *For two Maximum Nash Welfare allocations* $\mathbf{A}^1$*,* $\mathbf{A}^2$*, the values for an agent* i *in each differ by at most* 1*. That is,* $|v_i(A_i^1) - v_i(A_i^2)| \leq 1$

4 Deterministic Setting

Theorem 3. *For an allocation problem with at least 2 agents and 6 goods, there is no deterministic, strategyproof mechanism that always outputs a Maximum Nash Welfare allocation.*

Theorem 4. MNW^{tie} *is deterministic, group-strategyproof, always outputs a Maximum Nash Welfare allocation (but doesn't allocate non-valued goods), and can be computed in polynomial time.*

5 Randomized Setting

Theorem 5. *There is a randomized mechanism that is group strategyproof, ex-ante envy-free, whose support contains only Maximum Nash Welfare allocations (which implies it is ex-post Pareto Efficient and ex-post EF1) that can be computed in polynomial time.*

6 Conclusion