Stable Matching Algorithm Bias

CONTENTS

01 Problem & Algorithm

The Stable Matching Problem

A Centralized Assignment Market

Market Definition

Two disjoint sets of size **N**: **Students (S)** and **Establishments (E)**. Each agent holds a strict, complete preference list over the other set.

Objective: Stability

Find a matching μ with no blocking pairs to prevent "justified envy" and ensure no incentive to deviate.

Mathematical Definition of a Blocking Pair

A pair **(s, e)** is blocking if:

1. Student **s** prefers **e** over their current assignment **μ(s)**.
2. Establishment **e** prefers **s** over their current assignment **μ(e)**.

The Gale-Shapley Algorithm

Guaranteeing Stability via Deferred Acceptance

1. Propose

Unmatched agents send offers to their next favorite choice.

2. Tentative Accept

Recipients hold the best offer and reject all others.

3. Iterate

Rejected agents move down their list and propose again.

Experimental Scenarios

DA-S: Student-Proposing (Optimal for Students)

DA-E: Establishment-Proposing (Optimal for Establishments)

Implementation & Validation

Software Architecture

`ExperimentRunner` class generates preferences.
`StableMarriageAlgorithm` class solves matching using FIFO queues.

Validation: Mathematical Stability Verification

A dedicated `verify_stability()` function checks every non-matched pair **(s, e)** for blocking conditions by comparing ranks.

Result: Across all simulations (n=50, 100, 500)

0 Blocking Pairs detected. Implementation is correct.

02 Satisfaction Metrics

Satisfaction Metrics: Quantifying Happiness

Beyond Stability: Measuring Fairness

Normalized Satisfaction Score

Maps average rank to a 0-100 scale for fair comparison across different market sizes (N).

$$Score = 100 \times \left(1 - \frac{AvgRank}{N-1}\right)$$

100 = Perfect Match (1st choice), 0 = Worst Case (last choice).

Egalitarian Cost (Social Welfare)

Sum of all ranks. Measures total system "friction". Lower is better.

Group Equality Score

Absolute difference in satisfaction scores. High values indicate bias.

Cost Analysis & System Efficiency

Which scenario is fairer and more efficient?

1. Egalitarian Cost (Global Efficiency)

The sum of all ranks, representing total system "friction". Lower is better.

Student-Proposes	Establishment-Proposes
31,487	47,305

DA-S is 33% more efficient.

2. Group Inequality

The absolute difference in satisfaction scores between groups.

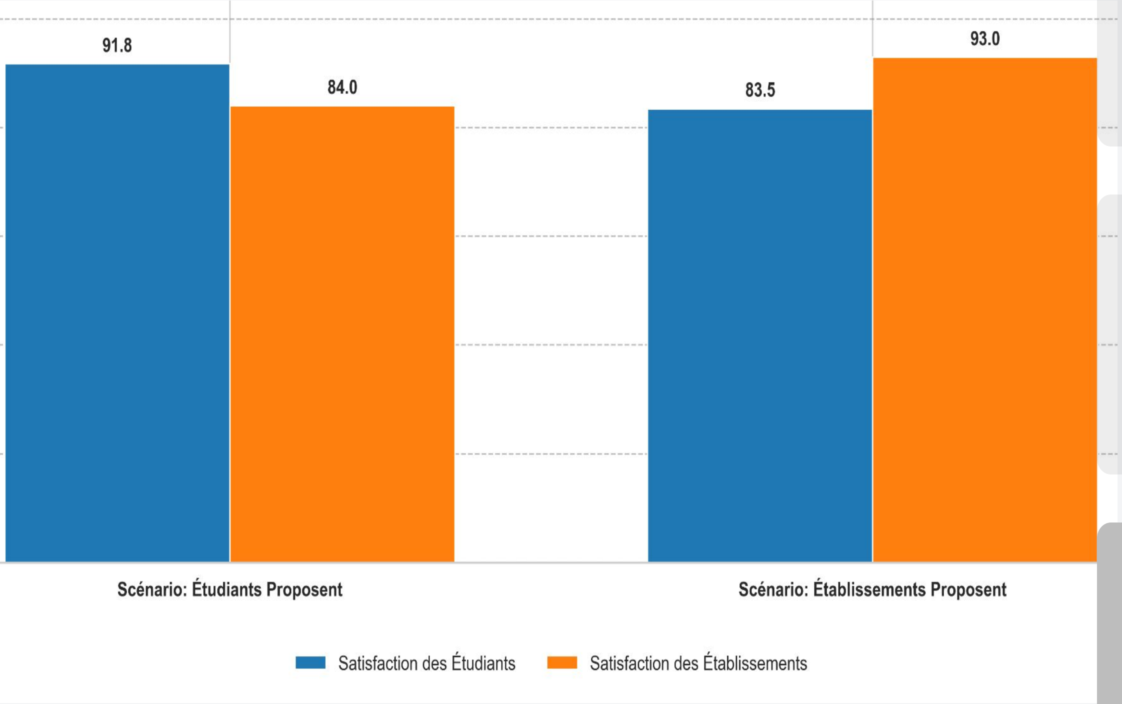
Student-Proposes	Establishment-Proposes
9.77	17.13

DA-S is objectively fairer.

03 Experimental Results

Bias Emerges at n=50

Early Signs of Non-Neutrality



Student-Proposing (DA-S)

Student Satisfaction: **91.8**

Establishment Satisfaction: **84.0**

Establishment-Proposing (DA-E)

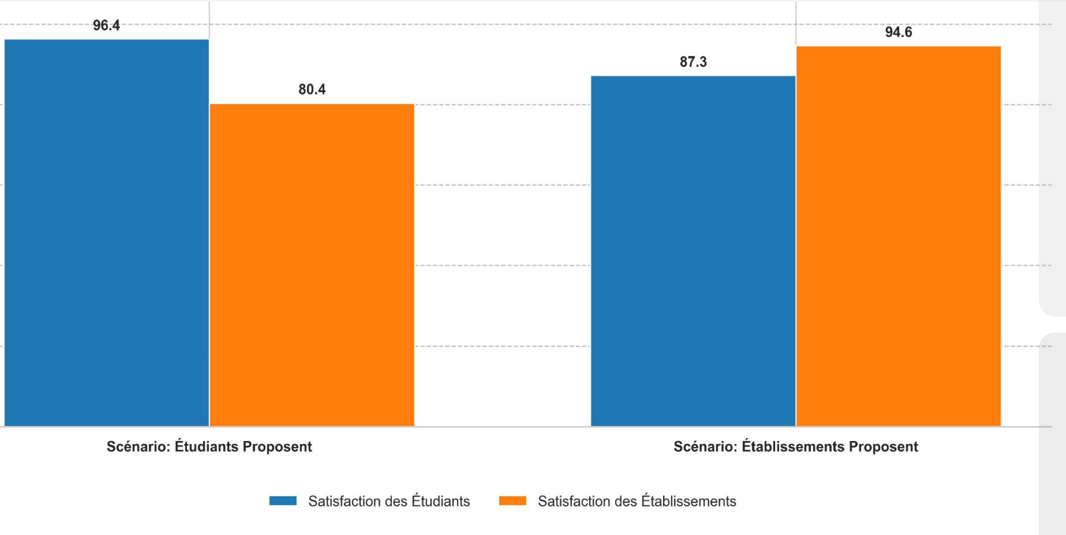
Student Satisfaction: **83.5**

Establishment Satisfaction: **93.0**

An eight-point gap appears immediately, showing the passive side consistently achieves lower welfare.

The Structural Gap Widens at n=100

The "Proposer Advantage" becomes more pronounced.



Key Observations

The **equality gap nearly doubles** to 15.1 points in the DA-S scenario.

- Proposers can "climb" their lists with more options available.
- Receivers are stuck with the "best of what's available," regressing to the mean.

Satisfaction Scores (n=100)

DA-S: Students Propose

Students: **96.1**, Establishments: **81.0**

DA-E: Establishments Propose

Students: **87.3**, Establishments: **94.6**

The Scale Effect

Massive Asymmetry at n=500

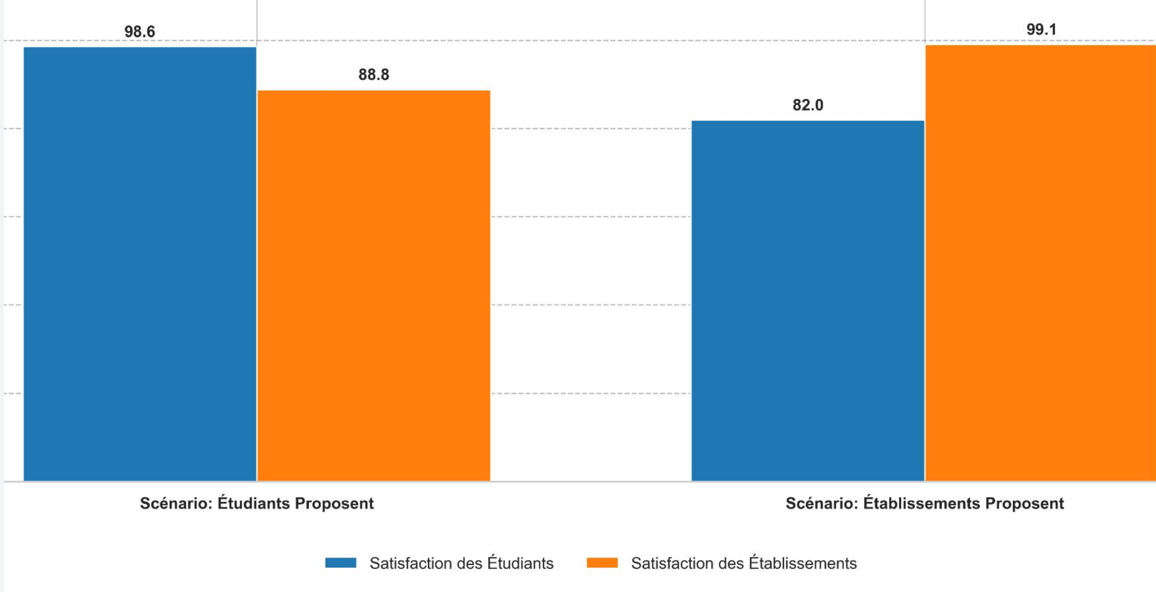
DA-S: Student Satisfaction

98.6 (near-perfect)

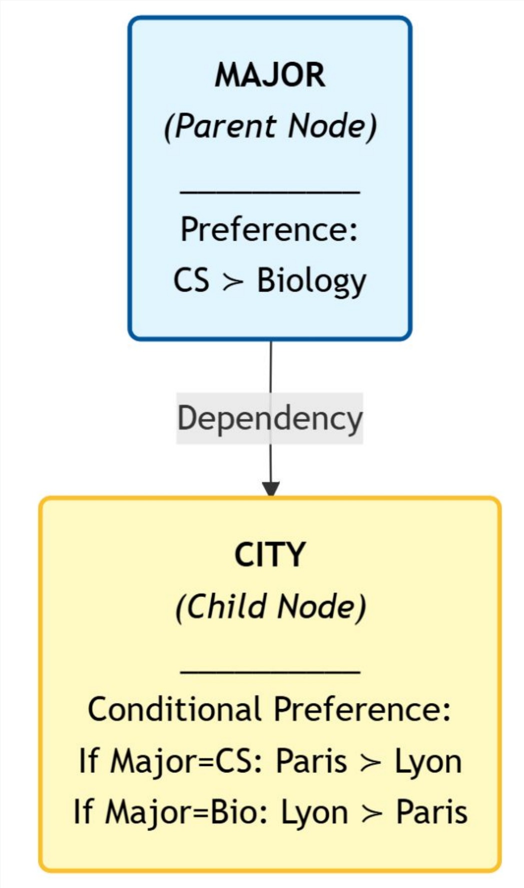
DA-S: Establishment Satisfaction

88.8 (stagnates)

Conclusion: In large systems, the proposer dictates the outcome. Stability is achieved at the expense of equity.



04 Extension & Conclusion



Extension: Compact Preferences

Using CP-Nets for Cognitive Feasibility

The Problem

Ranking **N=500** items is cognitively infeasible for human users.

Proposed Solution: CP-Nets

Users express **ceteris-paribus** rules on attributes instead of a full list.

Rule 1: "I prefer CS to Biology ."

Rule 2: "IF CS, THEN I prefer Paris to Lyon ."

Conclusion & Key Findings

1. Implementation

Successful $O(n^2)$ algorithm with rigorous stability verification. Confirmed correctness across all tests.

2. Proven Bias

The algorithm is inherently unfair. The proposing side captures surplus satisfaction, and the gap grows with market size (from 8% to 15%).

3. Recommendation

For student-centric systems like Parcoursup, the **Student-Proposing** model is mandatory to maximize welfare and fairness.

Thank you!