# State-of-the-Art: Gale–Shapley Algorithm for Optional Subject Selection

## 1. Overview and Foundational Work

The stable matching problem, first formalized by **Gale and Shapley (1962)** in their seminal paper "College Admissions and the Stability of Marriage," provides the theoretical foundation for two-sided matching markets. Their deferred acceptance algorithm guarantees a stable matching where no pair of agents would prefer each other over their current assignments. This work earned Lloyd Shapley the Nobel Prize in Economics (2012) and has been extensively applied to real-world allocation problems.

The Gale-Shapley (GS) algorithm operates through iterative proposals and rejections, terminating in at most $O(n^2)$ steps while guaranteeing stability. As demonstrated by **Gale and Sotomayor (1985)** in their work "Some Remarks on the Stable Matching Problem," the algorithm produces the proposing-side optimal stable matching—meaning students in a student-proposing version receive their best possible stable assignment.

## 2. Alternative Allocation Methods

Beyond the Gale-Shapley algorithm, several other mechanisms have been developed for course allocation problems.

**Serial Dictatorship** represents the simplest approach, where students are ordered by some priority (typically GPA or seniority) and sequentially choose their most preferred available course. While computationally trivial and strategy-proof, this method completely ignores the preferences of lower-priority students and often produces Pareto-inefficient outcomes.

**Random Serial Dictatorship (RSD)** introduces fairness through randomized ordering but maintains the same efficiency limitations.

**Top Trading Cycles (TTC)**, developed by Shapley and Scarf, allows students to form trading chains where each receives a more preferred option, guaranteeing both Pareto efficiency and strategy-proofness. However, TTC can be computationally complex and may produce outcomes that lack stability in the traditional sense.

**Probabilistic Serial (PS)** mechanism, introduced by Bogomolnaia and Moulin (2001), treats course allocation as a continuous process where students simultaneously "eat" their preferred courses at equal rates until capacity is exhausted, resulting in fractional allocations that can be randomized into final assignments. This approach achieves ex-ante efficiency but sacrifices strategy-proofness.

**Auction-based mechanisms** represent another class of solutions, particularly relevant when course popularity varies significantly.

The **Course Bidding System**, implemented at Harvard Business School and Wharton, gives students virtual currency to bid on courses, creating market-clearing prices that allocate seats to highest bidders. While this approach naturally handles capacity constraints and reveals preference intensity (not just ordinal rankings), it introduces complexity for students and may disadvantage those less skilled at strategic budgeting.

## 3. Evaluation Methods

Stable matching mechanisms for course allocation are evaluated through several key approaches. **Theoretical analysis** proves formal properties like stability (no blocking pairs), strategy-proofness, and computational complexity—for example, Gale-Shapley guarantees stability in $O(n^2)$ time. **Simulation studies** use synthetic preference data to measure performance metrics such as percentage of students receiving top-choice courses and average satisfaction ranks. **Real-world deployments**, like the school choice systems studied by Abdulkadiroğlu et al. (2005), track actual satisfaction rates and participation levels across thousands of students. **Computational experiments** test scalability by measuring response times and memory usage with increasing numbers of students and courses. **Fairness evaluation** examines whether outcomes are equitable across different student groups, often using statistical measures to detect bias toward high-achieving students.

## 4. Resources and Tools Available

Several production-ready libraries implement the Gale-Shapley algorithm for stable matching. **matching** (Python, available via pip) provides clean implementations of both student-optimal and course-optimal versions with support for capacity constraints, making it directly applicable to university course allocation. **Matching.jl** (Julia) offers high-performance implementations with excellent documentation and examples for the hospital-residents variant, which naturally extends to courses with enrollment limits.

## 5. Relevant links

Remarks on the stable matching problem

New techniques for best-matching retrieval

Gale-Shapley Algorithm

Serial dictatorship

Matching.jl