APPROXIMATION ALGORITHMS FOR MAXIMUM NON-CLASHING TEACHING

Anonymous authors
Paper under double-blind review

ABSTRACT

Non-clashing teaching is a canonical theoretical setting in machine teaching, and its fundamental problem is to compute the Non-Clashing Teaching Dimension (NCTD). Given a collection of candidate targets, a teacher selects at most k labeled examples for each target. If two targets can be distinguished by one of the selected examples, we call the pair non-clashing. NCTD asks for the minimum budget k needed to make every pair of targets non-clashing. However, even deciding whether $k = 1$ suffices is NP-hard, indicating the computational difficulty of directly optimizing k. Motivated by fixed budgets that may not permit complete separation, we study Maximum Non-Clashing Teaching (MNCT), which maximizes the number of non-clashing target pairs for a given k. Using linear programming, we develop a technique based on configurations that achieves an approximation ratio of $1-(1-\frac{1}{2\lceil k/r\rceil})^{2\lceil k/r\rceil}$ for any fixed $1 \leq r \leq k$, and provides a tradeoff between running time and approximation quality. Consequently, for every fixed k, setting $r = k$ yields a $3/4$-approximation. These approximation guarantees extend naturally to the weighted setting in which target pairs carry nonnegative weights. We also establish several complexity results, including APX-hardness for weighted MNCT even when $k = 1$. Finally, we give exact polynomial-time algorithms for some special cases.

1 INTRODUCTION

Unlike the usual machine-learning perspective, machine teaching asks how a teacher can select a small set of labeled examples enabling a learner to identify a given target. Its foundations lie in early work on teachability and teaching dimensions (Shinohara and Miyano, 1991; Goldman and Kearns, 1995). Subsequent research extended this framework to education, human–computer interaction, learning from demonstrations, and inverse reinforcement learning (Zhu, 2015; Zhu et al., 2018; Thomaz and Cakmak, 2009; Akgun et al., 2012; Ho et al., 2016; Brown and Niekum, 2019).

Because the teacher selects examples, the teacher and learner may agree in advance that particular examples represent a target. The learner could then identify the target from this convention, without using example labels to rule out other candidates. A small teaching set may therefore reflect a prearranged code rather than the intrinsic difficulty of teaching the target. To rule out such encodings, Goldman and Mathias introduced anti-collusion conditions (Goldman and Mathias, 1996). Building on this principle, Kirkpatrick et al. (2019) introduced non-clashing teaching, later developed further by Fallat et al. (2023). Its central idea is that differences between candidate targets should be witnessed directly by the teaching examples, rather than conveyed through an agreed-upon encoding scheme.

On a finite example domain $\mathcal{X}$, the teacher assigns each candidate target $C \subseteq \mathcal{X}$ a teaching set $T(C) \subseteq \mathcal{X}$ of at most k labeled examples. Two targets C, C' are non-clashing if an example selected for either receives different labels under them, i.e., $(T(C) \cup T(C')) \cap (C \triangle C') \neq \varnothing$. In the unrestricted setting, positive and negative examples are allowed; the minimum budget making all pairs non-clashing is the Non-Clashing Teaching Dimension (NCTD). Restricting selections to examples belonging to the target defines the positive setting and the corresponding minimum budget, the positive NCTD (NCTD^+).

Research on NCTD has investigated its combinatorial properties, computational complexity, and algorithms and teaching-budget bounds for different structured concept classes (Kirkpatrick et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

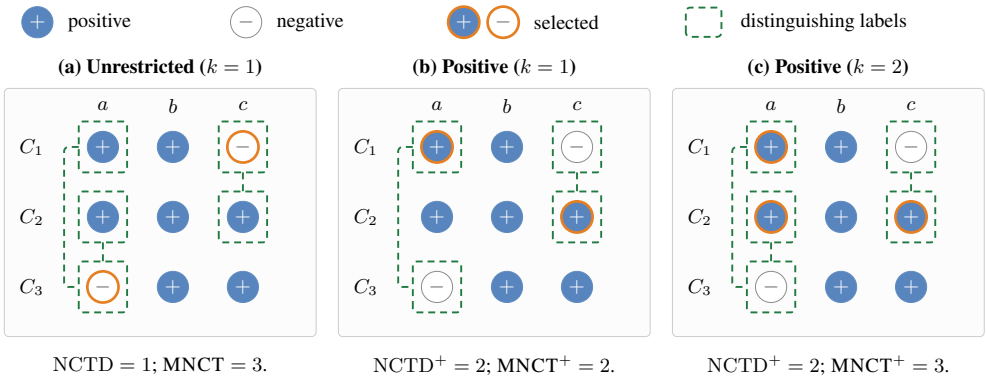

Figure 1: Blue and white points indicate positive and negative labels. Orange outlines mark selected examples; each dashed connection joins two boxed labels witnessing separation.

2019; Fallat et al., 2023; Simon, 2023; Chalopin et al., 2024; Ganian et al., 2025; Bhore et al., 2026). These studies ask for the minimum budget making every target pair non-clashing, or whether a given budget k suffices. Existing results show that even deciding whether $k = 1$ suffices is NP-hard, demonstrating the computational difficulty of directly optimizing NCTD.

In many teaching settings, the budget k is fixed by annotation costs, communication resources, or demonstration costs. When the budget is insufficient for complete separation, distinguishing more target pairs remains valuable since it reduces the pairwise ambiguity left by the teaching map. Motivated by this observation, we introduce Maximum Non-Clashing Teaching (MNCT): given a budget k, select at most k examples for each target so as to maximize the number of non-clashing target pairs. As with NCTD, we study both unrestricted MNCT and positive MNCT (MNCT$^+$).

A simple example illustrates these four quantities. Let $\mathcal{X} = \{a, b, c\}$ and consider the three targets $C_1 = \{a, b\}$, $C_2 = \{a, b, c\}$, and $C_3 = \{b, c\}$, shown in Figure 1. In Figure 1(a), C_1 selects the negative example c, C_3 selects the negative example a, and C_2 selects nothing. All three pairs are non-clashing, so NCTD $= 1$ and MNCT $= 3$. In Figure 1(b), selecting the positive example a for C_1, the positive example c for C_2, and nothing for C_3 makes $\{C_1, C_3\}$ and $\{C_1, C_2\}$ non-clashing, while $\{C_2, C_3\}$ remains clashing. Since both C_1 and C_3 are subsets of C_2, neither C_1 nor C_3 can distinguish itself from C_2 using its own positive examples. Thus, C_2 must select both c and a to separate these two pairs. Budget one cannot separate all three pairs, so MNCT$^+$ $= 2$. In Figure 1(c), adding the positive example a to $T(C_2)$ makes the remaining pair non-clashing, establishing NCTD$^+$ $= 2$.

1.1 Related Work

Kirkpatrick et al. (2019) introduced non-clashing teaching and established its optimality among batch teaching models satisfying the Goldman–Mathias anti-collusion criterion. Fallat et al. (2023) studied its combinatorial properties, complexity, and connections to VC dimension and sample compression, while Simon (2023) investigated its relationship with recursive teaching dimension (Doliwa et al., 2014). For graph-structured concept classes, Chalopin et al. (2024) initiated the study of non-clashing teaching for graph balls. More recently, two ICLR works provided direct inspiration for the present study: Ganian et al. (2025) developed complexity and algorithmic results for graph balls, while Bhore et al. (2026) established hardness and tractability results for closed neighborhoods and showed that closed-neighborhood subfamilies can represent arbitrary finite binary concept classes.

Approximation algorithms trade exact optimality for polynomial time while guaranteeing that the returned solution stays within a stated factor of the optimum. These studies primarily minimize the budget needed for complete separation. For NCTD$^+$, the hardness of deciding whether the optimum is one rules out a $(2 - \varepsilon)$-approximation for any fixed $\varepsilon > 0$ unless P $=$ NP (Kirkpatrick et al., 2019; Ganian et al., 2025). Another relaxation requires separation only for a prescribed subset of pairs, as in positive teaching maps for sufficiently distant graph balls (Chalopin et al., 2024). In contrast, our objective fixes the budget and maximizes the number of separated pairs.

Table 1: Summary of the main results. Here, $q = \lceil k/r \rceil$.

Type	Setting	Result
Approximation	MNCT, MNCT$^+$; $1 \le r \le k$	$(1-(1-1/(2q))^{2q})$-approximation
	MNCT, MNCT$^+$; fixed k	$3/4$-approximation
	MNCT$^+$; $k = 1$	Tight $3/4$ LP guarantee
Hardness	MNCT$^+$; general k	No $(1-1/e+\varepsilon)$-approximation
	Weighted MNCT; $k = 1$	APX-hard
	Weighted MNCT$^+$; $k = 1$, size-two targets	NP-hard
Exact	MNCT$^+$; size-two targets	Polynomial-time solvable
	MNCT$^+$; laminar classes	Polynomial-time solvable

Since one example may separate several target pairs, MNCT and MNCT$^+$ have a coverage objective under a partition matroid enforcing per-target budgets. This is a classical setting for monotone submodular maximization under matroid constraints (Nemhauser et al., 1978; Fisher et al., 1978; Calinescu et al., 2011); in particular, the algorithm of Calinescu et al. (2011) achieves a $(1-1/e)$-approximation, the baseline for our analysis. Our improved guarantees exploit the additional structure that each target pair can be separated only by examples selected for one of its two targets.

1.2 Our Contributions

We introduce MNCT and MNCT$^+$, the fixed-budget optimization counterparts of NCTD and NCTD$^+$. The main results in this paper are summarized in Table 1.

For both versions and $1 \le r \le k$, our LP-rounding framework achieves an approximation ratio of $1-(1-1/(2q))^{2q}$, where $q = \lceil k/r \rceil$, in time $|\mathcal{X}|^{O(r)}\,\mathrm{poly}(|\mathcal{C}|, k)$. It groups up to r examples into a configuration. The case $r = 1$ provides a polynomial-time algorithm for every positive budget, while $r = k$ yields a $3/4$-approximation for every fixed k. The algorithms are deterministic, and all approximation guarantees extend to nonnegative target-pair weights.

For MNCT$^+$ with general k, no polynomial-time $(1-1/e+\varepsilon)$-approximation exists for any fixed $\varepsilon > 0$ unless P = NP. Weighted MNCT is APX-hard even for $k = 1$. We also construct positive budget-one instances showing that the natural LP has integrality gap $3/4$, so its $3/4$ guarantee is tight.

Finally, for MNCT$^+$, we give exact polynomial-time algorithms for size-two and laminar concept classes. The weighted size-two case is NP-hard even with target-pair weights in $\{0, 1\}$.

Due to limited space, proofs of lemmas and theorems marked with "*" are deferred to the appendix.

2 Preliminaries and Problem Definition

Let $\mathcal{X}$ be the finite set of examples available to the teacher. Each target is represented by its set of positive examples, so a target is a subset $C \subseteq \mathcal{X}$. Examples in C are positive and all remaining examples are negative. The possible targets form a finite concept class $\mathcal{C} = \{C_1, \ldots, C_m\}$ whose members are pairwise distinct. The input explicitly specifies which examples belong to each target. We let $\binom{\mathcal{C}}{2}$ denote the set of all unordered pairs of distinct targets in $\mathcal{C}$, so $|\binom{\mathcal{C}}{2}| = \binom{m}{2}$.

When the true target is C, let $T(C)$ denote the set of examples shown by the teacher, with each example labeled according to C. We call $T(C)$ the *teaching set* of C, and a rule that assigns a teaching set to every target a *teaching map*. In the positive version, the teaching set may contain only examples belonging to the target, so $T(C) \subseteq C$. In the unrestricted version, $T(C)$ may also contain negative examples of C. A teaching map T satisfies budget k if $|T(C)| \le k$ for every $C \in \mathcal{C}$.

Consider two distinct targets C and C'. We say that C and C' are *separated* by a teaching map T if some example in $T(C) \cup T(C')$ receives different labels under the two targets. Because targets are represented by their positive sets, this condition is equivalent to $(T(C) \cup T(C')) \cap (C \triangle C') \neq \varnothing$, where $C \triangle C'$ is the symmetric difference of the two targets. Let $\mathrm{sep}(T)$ denote the number of unordered target pairs separated by T. For any positive integer q, we define $[q] \coloneqq \{1, \ldots, q\}$.

Definition 2.1 (Positive Maximum Non-Clashing Teaching)**.** The positive *Maximum Non-Clashing Teaching* problem, denoted as MNCT$^+$, takes as input a finite example set $\mathcal{X}$, a finite concept class $\mathcal{C}$, and an integer $k \geq 1$. A feasible solution is a teaching map T satisfying budget k and $T(C) \subseteq C$ for every $C \in \mathcal{C}$. The objective is to maximize $\mathrm{sep}(T)$.

The unrestricted *Maximum Non-Clashing Teaching* problem, denoted as MNCT, has the same input and objective. The only difference is that teaching examples may be selected from the entire example set, so $T(C) \subseteq \mathcal{X}$. A teaching set may therefore contain both positive and negative examples. We may assume $k \leq |\mathcal{X}|$, since if $k > |\mathcal{X}|$, setting $T(C) = C$ is optimal for both versions.

Let $\mathrm{OPT}_k^+(\mathcal{C})$ and $\mathrm{OPT}_k(\mathcal{C})$ denote the optimum values of MNCT$^+$ and MNCT, respectively. If $\mathrm{sep}(T) = \binom{m}{2}$, then T separates every target pair and is therefore a non-clashing teaching map. Consequently, $\mathrm{NCTD}^+(\mathcal{C})$ is the smallest integer k for which $\mathrm{OPT}_k^+(\mathcal{C}) = \binom{m}{2}$, and $\mathrm{NCTD}(\mathcal{C})$ is the smallest integer k for which $\mathrm{OPT}_k(\mathcal{C}) = \binom{m}{2}$.

For any $\alpha \in [0, 1]$, an α-approximation for MNCT$^+$ is an algorithm that, on every instance, outputs a feasible teaching map T satisfying $\mathrm{sep}(T) \geq \alpha\, \mathrm{OPT}_k^+(\mathcal{C})$. For a randomized algorithm, the inequality is required in expectation. The approximation ratio for MNCT is defined analogously, with $\mathrm{OPT}_k(\mathcal{C})$ in place of $\mathrm{OPT}_k^+(\mathcal{C})$.

In the weighted variants, the input also specifies a nonnegative weight $w_{C,C'}$ for every unordered target pair. The objective $\mathrm{sep}_w(T)$ is the total weight of the pairs separated by T. The feasible teaching maps and per-target budget are unchanged.

3 A Basic LP-Rounding Algorithm

In this section, we present a basic approximation algorithm that applies to both MNCT$^+$ and MNCT. The algorithm first solves a linear program and then uses its solution to select at most k examples at random for each target, using randomized rounding (Raghavan and Thompson, 1987). A pair is separated whenever either target selects an example that distinguishes the two. Bounding the probability of this event yields an approximation guarantee depending on k.

3.1 Linear programming relaxation

To handle both the positive and unrestricted versions, define the set of examples available to each target $C \in \mathcal{C}$ by $A(C) = C$ for MNCT$^+$ and $A(C) = \mathcal{X}$ for MNCT. For two distinct targets C and C', let $Q_C(C') = A(C) \cap (C \triangle C')$. This is the set of all examples that C is allowed to select and that distinguish C from C'. In the positive version, $Q_C(C') = C \setminus C'$, whereas in the unrestricted version, $Q_C(C') = C \triangle C'$.

The two sides of a target pair may have different available distinguishing examples. In the positive version, for example, if $C \subsetneq C'$, then $Q_C(C') = \varnothing$ while $Q_{C'}(C) = C' \setminus C$. Only C' can then separate the pair using a positive example.

For every target $C \in \mathcal{C}$ and example $x \in A(C)$, a variable $y_{C,x} \in [0, 1]$ relaxes the binary choice of whether C selects x. For every unordered target pair $\{C, C'\} \in \binom{\mathcal{C}}{2}$, a variable $z_{C,C'}$ represents the value credited to that pair by the linear program. Consider the following LP:

$$\max \quad \sum_{\{C,C'\} \in \binom{\mathcal{C}}{2}} z_{C,C'} \tag{A}$$

$$\text{s.t.} \quad \sum_{x \in A(C)} y_{C,x} \leq k \qquad \forall\, C \in \mathcal{C}, \tag{1}$$

$$z_{C,C'} \leq \sum_{x \in Q_C(C')} y_{C,x} + \sum_{x \in Q_{C'}(C)} y_{C',x} \qquad \forall\, \{C, C'\} \in \tbinom{\mathcal{C}}{2}, \tag{2}$$

$$0 \leq y_{C,x} \leq 1 \qquad \forall\, C \in \mathcal{C},\ x \in A(C), \tag{3}$$

$$0 \leq z_{C,C'} \leq 1 \qquad \forall\, \{C, C'\} \in \tbinom{\mathcal{C}}{2}. \tag{4}$$

Let LP* denote the optimum value of the LP for the version under consideration.

Constraint (1) ensures that the sum of the selection variables for each target is at most k. For a pair $\{C, C'\}$, constraint (2) sums the selection variables corresponding to distinguishing examples at the two targets and uses this sum to bound the value credited to the pair. The bounds $y_{C,x} \leq 1$ reflect that an integral teaching map selects each example at most once for a target, while $z_{C,C'} \leq 1$ ensures that every target pair contributes at most one unit to the objective.

The LP has $O(|\mathcal{C}||\mathcal{X}| + |\mathcal{C}|^2)$ variables and constraints. Any feasible teaching map induces an integral LP solution of the same value by setting $y_{C,x} = 1$ for selected examples and $z_{C,C'} = 1$ for separated pairs, with all other variables zero.

Lemma 3.1. *The LP for either version has polynomial size and can be solved in polynomial time. For the positive version,* $\mathrm{LP}^* \geq \mathrm{OPT}_k^+(\mathcal{C})$*, and for the unrestricted version,* $\mathrm{LP}^* \geq \mathrm{OPT}_k(\mathcal{C})$.

3.2 Randomized rounding and approximation analysis

After computing an optimal LP solution, perform k independent random trials for every target C, with all trials for different targets also mutually independent. In each trial, select an example $x \in A(C)$ with probability $y_{C,x}/k$, and select nothing with the remaining probability $1 - \sum_{x \in A(C)} y_{C,x}/k$. Constraint (1) guarantees that this is a valid probability distribution.

Let $T(C)$ be the set obtained by removing duplicates from the outcomes of all trials for C. Since only k trials are performed, $|T(C)| \leq k$. In the positive version, every candidate example belongs to C, so the resulting teaching map also satisfies the positive restriction.

Fix a pair of targets C, C'. Let $a = \sum_{x \in Q_C(C')} y_{C,x}$ and $b = \sum_{x \in Q_{C'}(C)} y_{C',x}$. The k trials for C all miss $Q_C(C')$ with probability $(1 - a/k)^k$, and similarly for C'. Independence yields the following result.

Lemma 3.2. *The randomized rounding separates* $\{C, C'\}$ *with probability* $1 - (1 - a/k)^k(1 - b/k)^k$.

Let $\rho_k = 1 - (1 - 1/(2k))^{2k}$. By (2) and (4), $z_{C,C'} \leq \min\{1, a + b\}$. The next lemma establishes a uniform rounding guarantee for this pair.

Lemma 3.3 (*). *For all* $a, b \in [0, k]$*, let* $s = a + b$*. Then* $1 - (1 - a/k)^k(1 - b/k)^k \geq \rho_k \min\{1, s\}$.

Theorem 3.4 (*). *For every finite concept class and every integer* $k \geq 1$*, both* MNCT^+ *and* MNCT *admit a deterministic polynomial-time* ρ_k*-approximation, where* $\rho_k = 1 - (1 - 1/(2k))^{2k}$.

When $k = 1$, we have $\rho_1 = 3/4$. For $k \geq 1$, $\rho_k > 1 - 1/e$, and $\rho_k \to 1 - 1/e$ as $k \to \infty$.

4 An Improved Algorithm Based on Configurations

The basic algorithm in the previous section makes k independent choices for each target, selecting at most one example in each choice. In this section, we instead select a *configuration* of up to r examples at a time and make $q = \lceil k/r \rceil$ independent choices for each target. Increasing r improves the guarantee whenever q decreases, at the cost of considering larger configurations.

Fix an integer $r \in [1, k]$, and let $q = \lceil k/r \rceil$. Each target has q *slots*, each assigned one configuration. The first $q - 1$ slots have *capacity* r, and the last has capacity $k - r(q - 1)$. Let κ_j denote the capacity of slot j. Then $\sum_{j=1}^{q} \kappa_j = k$.

4.1 Configuration linear program

We reuse the definitions of $A(C)$ and $Q_C(C')$ in Section 3. For slot j of target C, the feasible configurations are all sets of at most κ_j available examples, including the empty set: $\mathcal{F}_{C,j} = \{S \subseteq A(C) : |S| \leq \kappa_j\}$. For each target C, the first $q - 1$ slots have the same set of feasible configurations, since they all have capacity r. The LP uses one variable for each feasible configuration in each slot.

For every $C \in \mathcal{C}$, $j \in [q]$, and $S \in \mathcal{F}_{C,j}$, introduce a variable $\lambda_{C,j,S}$ representing the probability of choosing configuration S for slot j of target C during rounding. For each target pair $\{C, C'\}$, the variable $z_{C,C'}$ represents its value in the LP. Consider the following configuration LP:

$$\max \sum_{\{C,C'\} \in \binom{\mathcal{C}}{2}} z_{C,C'} \tag{B}$$

$$\text{s.t.} \quad \sum_{S \in \mathcal{F}_{C,j}} \lambda_{C,j,S} = 1 \qquad \forall\, C \in \mathcal{C},\ j \in [q], \tag{5}$$

$$z_{C,C'} \le \sum_{j=1}^{q} \sum_{\substack{S \in \mathcal{F}_{C,j} \\ S \cap Q_C(C') \neq \varnothing}} \lambda_{C,j,S} + \sum_{j=1}^{q} \sum_{\substack{S \in \mathcal{F}_{C',j} \\ S \cap Q_{C'}(C) \neq \varnothing}} \lambda_{C',j,S} \quad \forall\, \{C, C'\} \in \tbinom{\mathcal{C}}{2}, \tag{6}$$

$$\lambda_{C,j,S} \ge 0 \qquad \forall\, C, j,\ S \in \mathcal{F}_{C,j}, \tag{7}$$

$$0 \le z_{C,C'} \le 1 \qquad \forall\, \{C, C'\} \in \tbinom{\mathcal{C}}{2}. \tag{8}$$

Let CLP_r^* denote the optimum value of the configuration LP for the version under consideration.

Constraint (5) makes the configuration probabilities for each slot sum to one. A configuration chosen by C separates $\{C, C'\}$ if it intersects $Q_C(C')$, and similarly for C'. Constraint (6) bounds the value credited to the pair by the sum, over all slots of both targets, of the probabilities of choosing a configuration that separates it.

Each slot has at most $\sum_{\ell=0}^{r} \binom{|\mathcal{X}|}{\ell} = |\mathcal{X}|^{O(r)}$ feasible configurations. Since the slot capacities sum to k, every feasible $T(C)$ can be partitioned into $T_1(C), \ldots, T_q(C)$ with $|T_j(C)| \le \kappa_j$. Setting $\lambda_{C,j,T_j(C)} = 1$, all other configuration variables to zero, and $z_{C,C'} = 1$ exactly for separated pairs yields an integral configuration-LP solution with value $\mathrm{sep}(T)$.

Lemma 4.1. *The configuration LP can be constructed and solved in $|\mathcal{X}|^{O(r)}\,\mathrm{poly}(|\mathcal{C}|, k)$ time. For the positive version, $\mathrm{CLP}_r^* \ge \mathrm{OPT}_k^+(\mathcal{C})$, and for the unrestricted version, $\mathrm{CLP}_r^* \ge \mathrm{OPT}_k(\mathcal{C})$.*

4.2 Rounding configurations

After solving the configuration LP optimally, independently select a configuration $S_{C,j}$ for every target C and slot j according to the distribution $\{\lambda_{C,j,S}\}_{S \in \mathcal{F}_{C,j}}$. Define $T(C) = \bigcup_{j=1}^{q} S_{C,j}$.

Since $|S_{C,j}| \le \kappa_j$, $|T(C)| \le \sum_{j=1}^{q} \kappa_j = k$. In the positive version, every configuration in $\mathcal{F}_{C,j}$ is a subset of C, so the rounded teaching map also satisfies the positive restriction.

Fix a pair $\{C, C'\}$. For every $j \in [q]$, define

$$a_j = \sum_{\substack{S \in \mathcal{F}_{C,j} \\ S \cap Q_C(C') \neq \varnothing}} \lambda_{C,j,S}, \qquad b_j = \sum_{\substack{S \in \mathcal{F}_{C',j} \\ S \cap Q_{C'}(C) \neq \varnothing}} \lambda_{C',j,S}.$$

Thus, a_j and b_j are the probabilities that the configurations chosen for slot j of C and C', respectively, separate the pair. Let $s = \sum_{j=1}^{q} (a_j + b_j)$ and $\rho_q = 1 - (1 - 1/(2q))^{2q}$.

Lemma 4.2 (*). *The probability that configuration rounding separates $\{C, C'\}$ is at least $\rho_q z_{C,C'}$.*

Theorem 4.3 (*). *For every finite concept class and integers $1 \le r \le k$, both MNCT^+ and MNCT admit a deterministic algorithm with approximation ratio $\rho_q = 1 - (1 - 1/(2q))^{2q}$ and running time $|\mathcal{X}|^{O(r)}\,\mathrm{poly}(|\mathcal{C}|, k)$.*

When $r = 1$, we have $q = k$, and the guarantee becomes ρ_k from Section 3. Setting $r = 2$ still yields a polynomial-time algorithm and, for every $k \ge 2$, yields ratio $1 - (1 - 1/(2\lceil k/2 \rceil))^{2\lceil k/2 \rceil}$, which is strictly larger than the guarantee of the basic algorithm. When $r = k$, the ratio achieves $3/4$, with polynomial running time for fixed k.

Corollary 4.4. *For every fixed positive integer k, both MNCT^+ and MNCT admit a polynomial-time $3/4$-approximation.*

Remark 4.5. *Both LP-rounding algorithms extend to the weighted variants with the same guarantees. Replace the LP objective by $\sum_{\{C,C'\} \in \binom{\mathcal{C}}{2}} w_{C,C'} z_{C,C'}$, keeping the constraints and rounding procedures unchanged. The same approximation guarantee follows by multiplying each pair's separation-probability bound by its nonnegative weight and summing over all pairs. Moreover, the same derandomization applies to $\mathrm{sep}_w(T)$, as explained in Appendix B.*

4.3 TIGHTNESS OF THE LP FOR $k = 1$

When $k = 1$, each target has one slot whose configuration contains at most one example, so the configuration LP is equivalent to the basic LP from Section 3. We now construct a family of positive instances showing that the $3/4$ guarantee relative to the LP optimum is tight.

For an integer $a = 2h \geq 4$, let $G_a = K_{a,a}$ with bipartition $L = \{u_1, \ldots, u_a\}$ and $R = \{v_1, \ldots, v_a\}$. Consider the positive instance consisting of all distinct closed neighborhoods in G_a, and let $U_i = N[u_i] = \{u_i\} \cup R$ and $V_j = N[v_j] = \{v_j\} \cup L$. The concept class is $\mathcal{C}_a = \{U_1, \ldots, U_a, V_1, \ldots, V_a\}$.

Under $k = 1$, a target U_i may select its own vertex u_i or a vertex of R. The former helps distinguish U_i from other U-type targets, whereas the latter helps distinguish it from V-type targets. The choices for a target V_j are similar. An integral solution must choose between these two roles, while the LP can assign positive fractions to both types of examples. This creates the following integrality gap.

Lemma 4.6 (*). *The constructed instance satisfies* $\mathrm{OPT}_1^+(\mathcal{C}_a) = \binom{2a}{2} - \binom{a}{2} = a(3a-1)/2$.

Lemma 4.7 (*). *For the same instance,* $\mathrm{LP}^* = 2a(a-1)$.

We define the integrality gap as the infimum of $\mathrm{OPT}_1^+(\mathcal{C})/\,\mathrm{LP}^*$ over instances with $\mathrm{LP}^* > 0$. The rounding analysis in Theorem 3.4 shows $\mathrm{OPT}_1^+(\mathcal{C}) \geq (3/4)\,\mathrm{LP}^*$ for every positive instance, so the integrality gap is at least $3/4$. Lemmas 4.6 and 4.7 show that this ratio approaches $3/4$ for the family $\mathcal{C}_a$, establishing the matching upper bound.

Theorem 4.8. *The LP* (A) *for* MNCT^+ *with* $k = 1$ *has integrality gap* $3/4$.

5 HARDNESS OF APPROXIMATION

5.1 INAPPROXIMABILITY FOR GENERAL BUDGETS

We transfer the approximation threshold of maximum coverage to the positive model. The construction uses center targets to encode coverage solutions and replicated element targets to measure their values. Marker examples make all targets distinct while contributing only a controlled additive error.

Consider an instance of MAX k-COVER. Let U be a universe of n elements, let $S_1, \ldots, S_m \subseteq U$ be the available sets, and let k be the number of sets allowed. Let M denote the maximum number of elements covered by at most k sets. We may assume $1 \leq k \leq m$ and $M \geq 1$, since the remaining cases are trivial. For a fixed $0 < \varepsilon < 1/e$, set $L = \lceil 4k/\varepsilon \rceil$ and $R = \lceil 4n(k + Ln)/\varepsilon \rceil$.

Introduce set examples $Y = \{x_1, \ldots, x_m\}$ and marker examples $P = \{p_1, \ldots, p_R\}$ and $Q = \{q_{u,\ell} : u \in U,\ 1 \leq \ell \leq L\}$, all mutually distinct. Let $\mathcal{X} = Y \cup P \cup Q$, and create R center targets and Ln element targets by $A_i = \mathcal{X} \setminus \{p_i\}$ and $B_{u,\ell} = \{x_j : u \notin S_j\} \cup P \cup \big(Q \setminus \{q_{u,\ell}\}\big)$. The concept class $\mathcal{C}$ consists of these targets, and the teaching budget remains k. Each marker is absent from exactly one target and belongs to every other target, so all targets are distinct. The instance has $R + Ln$ targets and $m + R + Ln$ examples and can be constructed in polynomial time for fixed ε. The construction yields the following result.

Theorem 5.1 (*). *For every fixed* $\varepsilon > 0$*, unless* $\mathrm{P} = \mathrm{NP}$*, there is no deterministic polynomial-time* $(1 - 1/e + \varepsilon)$*-approximation for* MNCT^+ *when* k *is part of the input.*

5.2 APX-HARDNESS FOR WEIGHTED MNCT

Figure 5.2 illustrates how variables and clauses are mapped to targets and weighted target pairs.

We now show that weighted MNCT remains APX-hard when each target selects at most one example. The reduction uses variable targets whose choices encode a truth assignment and clause targets that contribute one additional separated pair precisely when the corresponding clause is satisfied. Auxiliary targets enforce the two truth choices through pairs of weight six, while the clause pairs have weight one.

We reduce from MAX-E3SAT(5), the APX-hard restriction of maximum satisfiability in which every clause contains three distinct variables and every variable occurs in exactly five clauses (Feige, 1998). Let Φ have variables $x_1, \ldots, x_n$ and clauses $\varphi_1, \ldots, \varphi_m$, so $3m = 5n$. Let I_j be the set of the three

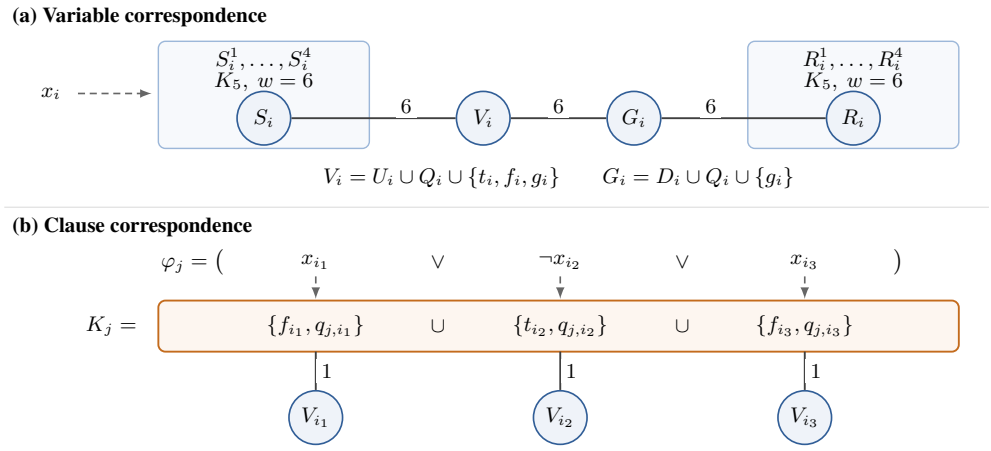

Figure 2: The APX-hardness construction. Dashed arrows show the construction correspondence, and solid edges carry target-pair weights. Each blue block represents a weight-six clique on its five targets, with internal edges omitted. In (b), each literal contributes its occurrence example and its falsifying truth example to the single target K_j.

variable indices in φ_j, and let $\mathrm{sat}_\Phi(\tau)$ be the number of clauses satisfied by an assignment τ. Denote the maximum of this quantity by OPT_Φ.

For each variable x_i, introduce sets $U_i = \{u_i^0, \ldots, u_i^4\}$ and $D_i = \{d_i^0, \ldots, d_i^4\}$, truth examples t_i, f_i, and an auxiliary example g_i. For each occurrence of x_i in clause φ_j, introduce an example $q_{j,i}$, and set $Q_i = \{q_{j,i} : i \in I_j\}$. All introduced examples are distinct, and $|Q_i| = 5$. The example domain is the union of these sets and examples over all variables.

The gadget for x_i contains twelve targets. Its two main targets are $V_i = U_i \cup \{t_i, f_i, g_i\} \cup Q_i$ and $G_i = D_i \cup \{g_i\} \cup Q_i$. The remaining targets are $S_i = U_i$, $R_i = D_i$, and the eight singleton targets $S_i^p = \{u_i^p\}$ and $R_i^p = \{d_i^p\}$ for $p \in \{1, 2, 3, 4\}$. For every clause φ_j, introduce a target K_j containing $q_{j,i}$ for each $i \in I_j$, together with f_i for each positive literal x_i in the clause and t_i for each negative literal $\neg x_i$. Thus, K_j contains the truth example that falsifies each of its three literals, as well as the three occurrence examples.

For every variable gadget, assign weight six to the six pairs $\{S_i^p, S_i^q\}$ with $1 \le p < q \le 4$, the four pairs $\{S_i, S_i^p\}$, and the pair $\{V_i, S_i\}$. Assign the same weight to the corresponding eleven pairs on the R side, namely the six pairs $\{R_i^p, R_i^q\}$, the four pairs $\{R_i, R_i^p\}$, and the pair $\{G_i, R_i\}$. Finally, give $\{V_i, G_i\}$ weight six. These are the twenty-three heavy pairs of the gadget. Give weight one to $\{V_i, K_j\}$ whenever $i \in I_j$, and weight zero to every other pair. Set the teaching budget to $k = 1$.

Let $\mathcal{I}_\Phi$ denote the resulting weighted instance. It has $18n$ examples and $12n + m$ targets and can be constructed in polynomial time. Target sizes, disjoint variable domains, and distinct occurrence examples ensure that all targets are distinct.

Theorem 5.2 (*). *Weighted* MNCT *is APX-hard even when* $k = 1$.

6 Structured Concept Classes

We now study the positive model on two structured concept classes. For size-two targets, a minimum-cost flow formulation yields a polynomial-time algorithm in the unweighted setting, whereas the weighted setting is NP-hard. We also obtain a polynomial-time algorithm for laminar concept classes.

Assume now that every target contains exactly two examples. View $\mathcal{X}$ as the vertex set of a simple graph H, and view each target $\{u, v\}$ as an edge uv. Under budget $k = 1$, adding a positive example cannot reduce the number of separated pairs, so every target may be assumed to select exactly one example. Orienting each edge toward its selected endpoint establishes a one-to-one correspondence between teaching maps and orientations of H.

Disjoint edges always give separated targets. Two adjacent edges remain unseparated exactly when both are oriented toward their common endpoint, and each such pair has a unique common endpoint.

Lemma 6.1. *Let D be an orientation of H, and let $d_D^-(v)$ denote the indegree of a vertex v. The number of unseparated target pairs in the corresponding teaching map is $\sum_{v \in V(H)} \binom{d_D^-(v)}{2}$.*

Thus, for $k = 1$, the problem is equivalent to finding an orientation minimizing a separable convex function of the indegrees. We reduce this problem to minimum-cost flow. Construct a network $\mathcal{F}_H$ with a source s, a sink t, one edge node p_e for every $e \in E(H)$, and one vertex node p_v for every $v \in V(H)$. Add an arc of capacity one and cost zero from s to every p_e. If $e = uv$, add arcs of capacity one and cost zero from p_e to both p_u and p_v. Finally, add $\deg_H(v)$ parallel unit-capacity arcs from p_v to t, with costs $0, 1, \ldots, \deg_H(v) - 1$. The required flow value is $|E(H)|$.

The network has $O(|V(H)| + |E(H)|)$ nodes and $O(|E(H)|)$ arcs. Since all capacities and the required flow value are integral, an integral minimum-cost flow exists and can be found in polynomial time (Ahuja et al., 1993). Call a feasible integral flow of value $|E(H)|$ *normalized* if, at every vertex node, it uses the cheapest available outgoing arcs. Replacing the outgoing arcs used at a vertex node converts any feasible integral flow into a normalized one without increasing its cost.

Route each edge node's unit of flow to its head in an orientation. A vertex of indegree ℓ uses its ℓ cheapest outgoing arcs, of total cost $0 + 1 + \cdots + (\ell - 1) = \binom{\ell}{2}$. Conversely, each normalized integral flow determines the head of every edge by the endpoint receiving its unit of flow, with the same cost.

Lemma 6.2. *Orientations of H are in one-to-one correspondence with normalized integral flows of value $|E(H)|$ in $\mathcal{F}_H$. Moreover, the orientation and the flow both have cost $\sum_{v \in V(H)} \binom{d_D^-(v)}{2}$.*

For $k = 1$, Lemmas 6.1 and 6.2 identify the minimum number of unseparated pairs with the minimum flow cost. For $k \geq 2$, selecting both examples of each target separates all pairs. Thus, we obtain the following result.

Theorem 6.3. *If every target contains exactly two examples, then* MNCT$^+$ *can be solved exactly in polynomial time. In particular, when $k = 1$ and $\mu(H)$ is the minimum cost of a flow in $\mathcal{F}_H$, $\mathrm{OPT}_1^+(\mathcal{C}) = \binom{|\mathcal{C}|}{2} - \mu(H)$.*

This polynomial-time result does not extend to arbitrary target-pair weights.

Theorem 6.4 (*). *Weighted* MNCT$^+$ *is NP-hard even when $k = 1$, every target contains exactly two examples, and all target-pair weights belong to $\{0, 1\}$.*

A concept class $\mathcal{C}$ is *laminar* if every two targets $A, B \in \mathcal{C}$ satisfy $A \subseteq B$, $B \subseteq A$, or $A \cap B = \varnothing$.

Theorem 6.5 (*). *When $\mathcal{C}$ is a laminar concept class,* MNCT$^+$ *can be solved exactly in polynomial time for every budget k. Moreover, when $k \geq 2$, all target pairs can be separated.*

7 Conclusion

We studied Maximum Non-Clashing Teaching (MNCT) and its positive version (MNCT$^+$) under a fixed example budget, asking how many target pairs can be made non-clashing for a given budget. For arbitrary finite concept classes, we developed a budget-dependent approximation framework and, using configurations, obtained a $3/4$-approximation for every fixed k, together with complementary inapproximability results. For the positive version with $k = 1$, our tightness result shows that the $3/4$ guarantee is tight with respect to the natural LP, while our results for structured concept classes show polynomial-time solvability for size-two targets and laminar classes but NP-hardness for weighted size-two targets. This optimization perspective extends the traditional minimum-teaching-budget problem to resource-constrained settings and allows different teaching maps to be compared quantitatively under a common objective. Future work may determine the remaining optimal approximation thresholds, particularly for the unrestricted version and for individual fixed budgets, characterize broader concept-class structures that admit exact algorithms or stronger approximation guarantees, and investigate how the number of non-clashing target pairs reflects teaching quality and resource efficiency in practical teaching tasks.

Reproducibility statement

This work is purely theoretical and involves no experiments or datasets. All definitions and assumptions are stated explicitly, and complete proofs of the theoretical results are provided in the main text and appendices.

AI use statement

Generative AI tools were used to assist with literature search and language editing. The research topic, problem formulation, mathematical models, algorithms, theoretical results, and proofs were conceived and developed by the authors without generative AI assistance. The authors reviewed and verified all AI-assisted material, including bibliographic information and edited text, and take full responsibility for the final manuscript.

References

Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, 1993.

Baris Akgun, Maya Cakmak, Jae Wook Yoo, and Andrea L. Thomaz. Trajectories and Keyframes for Kinesthetic Teaching: A Human-Robot Interaction Perspective. In *Proceedings of the 7th ACM/IEEE International Conference on Human-Robot Interaction*, pages 391–398, 2012. doi: https://doi.org/10.1145/2157689.2157815.

Sujoy Bhore, Liana Khazaliya, and Fionn Mc Inerney. Non-Clashing Teaching in Graphs: Algorithms, Complexity, and Bounds. In *Proceedings of the 14th International Conference on Learning Representations*, 2026. https://proceedings.iclr.cc/paper_files/paper/2026/file/b47c9f5cdfaad52e14f6d6fbb4553719-Paper-Conference.pdf.

Daniel S. Brown and Scott Niekum. Machine Teaching for Inverse Reinforcement Learning: Algorithms and Applications. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7749–7758, 2019. doi: https://doi.org/10.1609/aaai.v33i01.33017749.

Gruia Calinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a Monotone Submodular Function Subject to a Matroid Constraint. *SIAM Journal on Computing*, 40(6):1740–1766, 2011. doi: https://doi.org/10.1137/080733991.

Jérémie Chalopin, Victor Chepoi, Fionn Mc Inerney, and Sébastien Ratel. Non-Clashing Teaching Maps for Balls in Graphs. In *Proceedings of the 37th Conference on Learning Theory*, Proceedings of Machine Learning Research, volume 247, pages 840–875, PMLR, 2024. https://proceedings.mlr.press/v247/chalopin24a.html.

Thorsten Doliwa, Gaojian Fan, Hans Ulrich Simon, and Sandra Zilles. Recursive Teaching Dimension, VC-Dimension and Sample Compression. *Journal of Machine Learning Research*, 15(89):3107–3131, 2014. https://www.jmlr.org/papers/v15/doliwa14a.html.

Shaun M. Fallat, David G. Kirkpatrick, Hans Ulrich Simon, Abolghasem Soltani, and Sandra Zilles. On Batch Teaching Without Collusion. *Journal of Machine Learning Research*, 24(40):1–33, 2023. https://www.jmlr.org/papers/v24/22-0330.html.

Uriel Feige. A Threshold of $\ln n$ for Approximating Set Cover. *Journal of the ACM*, 45(4):634–652, 1998. doi: https://doi.org/10.1145/285055.285059.

Marshall L. Fisher, George L. Nemhauser, and Laurence A. Wolsey. An Analysis of Approximations for Maximizing Submodular Set Functions—II. In *Polyhedral Combinatorics*, Mathematical Programming Studies, volume 8, pages 73–87, North-Holland, 1978. doi: https://doi.org/10.1007/BFb0121195.

Robert Ganian, Liana Khazaliya, Fionn Mc Inerney, and Mathis Rocton. The Computational Complexity of Positive Non-Clashing Teaching in Graphs. In *Proceedings of the 13th International Conference on Learning Representations*, 2025. https://proceedings.iclr.cc/pape

r_files/paper/2025/hash/6e1caf62e7074b95aaa8904d44e02054-Abstract-Conference.html.

Sally A. Goldman and Michael J. Kearns. On the Complexity of Teaching. *Journal of Computer and System Sciences*, 50(1):20–31, 1995. doi: https://doi.org/10.1006/jcss.1995.1003.

Sally A. Goldman and H. David Mathias. Teaching a Smarter Learner. *Journal of Computer and System Sciences*, 52(2):255–267, 1996. doi: https://doi.org/10.1006/jcss.1996.0020.

Mark K. Ho, Michael L. Littman, James MacGlashan, Fiery Cushman, and Joseph L. Austerweil. Showing versus Doing: Teaching by Demonstration. In *Advances in Neural Information Processing Systems 29*, pages 3027–3035, 2016. https://proceedings.neurips.cc/paper/2016/hash/b5488aeff42889188d03c9895255cecc-Abstract.html.

Richard M. Karp. Reducibility among Combinatorial Problems. In *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972. doi: https://doi.org/10.1007/978-1-4684-2001-2_9.

David Kirkpatrick, Hans U. Simon, and Sandra Zilles. Optimal Collusion-Free Teaching. In *Proceedings of the 30th International Conference on Algorithmic Learning Theory*, Proceedings of Machine Learning Research, volume 98, pages 506–528, PMLR, 2019. https://proceedings.mlr.press/v98/kirkpatrick19a.html.

George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An Analysis of Approximations for Maximizing Submodular Set Functions—I. *Mathematical Programming*, 14:265–294, 1978. doi: https://doi.org/10.1007/BF01588971.

Christos H. Papadimitriou and Mihalis Yannakakis. Optimization, Approximation, and Complexity Classes. *Journal of Computer and System Sciences*, 43(3):425–440, 1991. doi: https://doi.org/10.1016/0022-0000(91)90023-X.

Prabhakar Raghavan. Probabilistic Construction of Deterministic Algorithms: Approximating Packing Integer Programs. *Journal of Computer and System Sciences*, 37(2):130–143, 1988. doi: https://doi.org/10.1016/0022-0000(88)90003-7.

Prabhakar Raghavan and Clark D. Thompson. Randomized Rounding: A Technique for Provably Good Algorithms and Algorithmic Proofs. *Combinatorica*, 7(4):365–374, 1987. doi: https://doi.org/10.1007/BF02579324.

Ayumi Shinohara and Satoru Miyano. Teachability in Computational Learning. *New Generation Computing*, 8(4):337–347, 1991. doi: https://doi.org/10.1007/BF03037091.

Hans U. Simon. Tournaments, Johnson Graphs and NC-Teaching. In *Proceedings of the 34th International Conference on Algorithmic Learning Theory*, Proceedings of Machine Learning Research, volume 201, pages 1411–1428, PMLR, 2023. https://proceedings.mlr.press/v201/simon23a.html.

Andrea L. Thomaz and Maya Cakmak. Learning about Objects with Human Teachers. In *Proceedings of the 4th ACM/IEEE International Conference on Human-Robot Interaction*, pages 15–22, 2009. doi: https://doi.org/10.1145/1514095.1514101.

Xiaojin Zhu. Machine Teaching: An Inverse Problem to Machine Learning and an Approach Toward Optimal Education. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, volume 29, 2015. doi: https://doi.org/10.1609/aaai.v29i1.9761.

Xiaojin Zhu, Adish Singla, Sandra Zilles, and Anna N. Rafferty. An Overview of Machine Teaching. *arXiv preprint arXiv:1801.05927*, 2018. https://arxiv.org/abs/1801.05927.

A Omitted Proofs

A.1 Proof of Lemma 3.3

Proof. We have $(1-a/k)^k(1-b/k)^k \le (1-s/(2k))^{2k}$. The function $h_k(s) = 1-(1-s/(2k))^{2k}$ is increasing and concave on $[0, 2k]$, with $h_k(0) = 0$ and $h_k(1) = \rho_k$. Thus, $h_k(s) \ge \rho_k s$ for $s \le 1$ and $h_k(s) \ge \rho_k$ for $s \ge 1$, proving the lemma. □

A.2 Proof of Theorem 3.4

Proof. We first analyze the randomized rounding. For each target pair $\{C, C'\} \in \binom{\mathcal{C}}{2}$, let $\xi_{C,C'}$ be the indicator that the pair is separated by the rounded teaching map. Then $\mathrm{sep}(T) = \sum_{\{C,C'\}\in\binom{\mathcal{C}}{2}} \xi_{C,C'}$.

Fix $\{C, C'\}$ and use the definitions of a and b in Section 3.2. Since $z_{C,C'} \le \min\{1, a+b\}$, by Lemmas 3.2 and 3.3, $\Pr[\xi_{C,C'} = 1] \ge \rho_k z_{C,C'}$. Thus, $\mathbb{E}[\mathrm{sep}(T)] \ge \rho_k \sum_{\{C,C'\}\in\binom{\mathcal{C}}{2}} z_{C,C'} = \rho_k \,\mathrm{LP}^*$.

By Lemma 3.1, $\mathrm{LP}^* \ge \mathrm{OPT}^+_k(\mathcal{C})$ in the positive version and $\mathrm{LP}^* \ge \mathrm{OPT}_k(\mathcal{C})$ in the unrestricted version. Therefore, the randomized rounding yields a ρ_k-approximation for both problems.

Finally, the algorithm can be derandomized in polynomial time using the method of conditional expectations (Raghavan, 1988), with the same approximation guarantee (see Appendix B). □

A.3 Proof of Lemma 4.2

Proof. The pair remains unseparated exactly when none of the configurations chosen by either target contains a distinguishing example. Its separation probability is therefore $1-\prod_{j=1}^q(1-a_j)\prod_{j=1}^q(1-b_j)$. Since $a_j, b_j \in [0,1]$ and their sum is s, $\prod_{j=1}^q(1-a_j)\prod_{j=1}^q(1-b_j) \le (1-s/(2q))^{2q}$. The same one-variable argument used in Lemma 3.3 shows that $1-(1-s/(2q))^{2q} \ge \rho_q \min\{1,s\}$. On the other hand, the pair constraint of the configuration LP and the bound $z_{C,C'} \le 1$ imply $z_{C,C'} \le \min\{1,s\}$. Combining the inequalities proves the result. □

A.4 Proof of Theorem 4.3

Proof. We first analyze the randomized rounding. For each pair $\{C, C'\}$, let $\eta_{C,C'}$ indicate whether configuration rounding separates the pair. By Lemma 4.2, $\Pr[\eta_{C,C'} = 1] \ge \rho_q z_{C,C'}$. Summing over all target pairs and using linearity of expectation yields $\mathbb{E}[\mathrm{sep}(T)] \ge \rho_q \sum_{\{C,C'\}\in\binom{\mathcal{C}}{2}} z_{C,C'} = \rho_q \,\mathrm{CLP}^*_r$. Together with Lemma 4.1, this proves a randomized ρ_q-approximation for both the positive and unrestricted versions. The algorithm can also be derandomized within the stated running time using the conditional-expectation procedure in Appendix B. □

A.5 Proof of Lemma 4.6

Proof. Adding a selected example cannot reduce the number of separated pairs, so we may assume that every target selects exactly one example. Let p_U be the number of targets U_i that select their own vertex u_i, and let p_V be the number of targets V_j that select their own vertex v_j.

If two U-type targets do not select their own vertices, then both select examples from R. Every example in R belongs to both targets, so the pair cannot be separated. There are at least $\binom{a-p_U}{2}$ such unseparated pairs. Symmetrically, at least $\binom{a-p_V}{2}$ pairs of V-type targets remain unseparated.

Now consider a cross pair U_i, V_j. If both targets select their own vertices, then the chosen examples are u_i and v_j, and both belong to $U_i \cap V_j$. The pair is therefore still unseparated. There are $p_U p_V$ such pairs. Let $\ell = p_U + p_V - a \in \mathbb{Z}$. Every teaching map therefore leaves at least

$$\binom{a-p_U}{2} + \binom{a-p_V}{2} + p_U p_V = \binom{a}{2} + \frac{\ell(\ell+1)}{2} \ge \binom{a}{2}$$

pairs unseparated.

We next construct a teaching map attaining this lower bound. Let half of the targets on each side select their own vertices and the remaining half select vertices on the opposite side. Partition L and

R into sets L_0, L_1 and R_0, R_1, each of size h. For $i \in [h]$, set $T(U_i) = \{u_i\}$ and $T(V_i) = \{v_i\}$. For $s \in [h]$, set $T(U_{h+s}) = \{v_{h+s}\}$. Choose a cyclic permutation σ of $[h]$ with no fixed point, and set $T(V_{h+s}) = \{u_{h+\sigma(s)}\}$.

The unseparated pairs consist of the $\binom{h}{2}$ pairs among $U_{h+1}, \ldots, U_{2h}$, the $\binom{h}{2}$ pairs among $V_{h+1}, \ldots, V_{2h}$, and the h^2 cross pairs in which both targets select their own vertices.

Every remaining same-side pair has at least one target that selects its own vertex and is therefore separated. If one target in a cross pair selects its own vertex while the other selects a vertex on the opposite side, then the latter selected vertex does not belong to the former target, so the pair is also separated. Finally, consider U_{h+s} and V_{h+t}. If this pair were unseparated, then both $s = t$ and $\sigma(t) = s$ would have to hold, implying $\sigma(t) = t$. Since σ has no fixed point, no additional unseparated pairs arise. The three classes above are therefore exactly all unseparated pairs, and their total number is $2\binom{h}{2} + h^2 = \binom{a}{2}$. This attains the lower bound. □

A.6 Proof of Lemma 4.7

Proof. We first construct a fractional solution of this value. For every U_i, allocate half of its unit budget to its own vertex and distribute the other half uniformly over the vertices of R. Thus, set $y_{U_i,u_i} = 1/2$ and $y_{U_i,v_j} = 1/(2a)$ for every $j \in [a]$. Symmetrically, for every V_j, set $y_{V_j,v_j} = 1/2$ and $y_{V_j,u_i} = 1/(2a)$ for every $i \in [a]$. The selection variables for each target sum to one.

For two distinct U-type targets, the selection variables for their respective private vertices are both $1/2$, so the corresponding z variable can be set to one. The same holds for two distinct V-type targets. For a cross pair U_i, V_j, the selection variables for U_i sum to $(a-1)/(2a)$ over $R \setminus \{v_j\}$, and those for V_j have the same sum over $L \setminus \{u_i\}$. Hence, we may set $z_{U_i,V_j} = (a-1)/a$. The objective value of this fractional solution is $2\binom{a}{2} + a^2(a-1)/a = 2a(a-1)$.

We now prove that this value is also an upper bound. For every target $C \in \mathcal{C}_a$ and every selectable example $x \in C$, the operation (C, x) separates exactly $a-1$ target pairs from the side of C. If $C = U_i$ and $x = u_i$, the operation appears in the pair constraints for U_i and the other $a-1$ U-type targets. If $x = v_j$, it appears in the pair constraints for U_i and all V-type targets except V_j. The case of a target V_j is symmetric. Summing the pair constraints counts every $y_{C,x}$ exactly $a-1$ times. Applying the budget constraints then yields

$$\sum_{\{C,C'\} \in \binom{\mathcal{C}_a}{2}} z_{C,C'} \le (a-1) \sum_{C \in \mathcal{C}_a} \sum_{x \in C} y_{C,x} \le 2a(a-1),$$

so the fractional solution above is optimal. □

A.7 Proof of Theorem 5.1

Proof. We reduce from Max k-Cover, for which no polynomial-time algorithm achieves ratio $1-1/e+\eta$ for any fixed $\eta > 0$ unless P = NP (Feige, 1998). It suffices to consider $\varepsilon < 1/e$. Suppose that a deterministic polynomial-time α-approximation for MNCT$^+$ exists, where $\alpha = 1 - 1/e + \varepsilon$. Use the construction in Section 5.1, with its parameters L and R and coverage optimum M.

Consider any feasible positive teaching map T. For each center A_i, let $J_i = \{j : x_j \in T(A_i)\}$ and $c_i = |\bigcup_{j \in J_i} S_j|$. Since $|J_i| \le k$, these indices specify a feasible coverage solution. All set examples belong to every center, so a set example selected by an element target cannot separate it from a center. Thus, $\{A_i, B_{u,\ell}\}$ is witnessed by a selected set example exactly when $u \in \bigcup_{j \in J_i} S_j$. Such examples then separate exactly Lc_i pairs between A_i and the element targets, and they separate no pair of centers.

A marker selected for a target can separate it only from the unique target lacking that marker. Since the total number of selected examples is at most $k(R + Ln)$, markers witness at most this many pairs. All remaining pairs lie among the Ln element targets. Consequently,

$$\mathrm{sep}(T) \le L \sum_{i=1}^{R} c_i + \Delta, \qquad \Delta = k(R + Ln) + \binom{Ln}{2}.$$

Conversely, let every center select the set examples corresponding to an optimal coverage solution, and let every element target select nothing. This separates exactly LRM pairs, so $\mathrm{OPT}_k^+(\mathcal{C}) \ge LRM$.

Our parameter choices also give

$$\frac{\Delta}{LR} \le \frac{k}{L} + \frac{kn + Ln^2/2}{R} \le \frac{\varepsilon}{4} + \frac{\varepsilon}{4} = \frac{\varepsilon}{2}.$$

Apply the assumed approximation algorithm and choose an index i^* maximizing c_i. The resulting coverage solution is computable in polynomial time, and

$$c_{i^*} \ge \frac{\mathrm{sep}(T) - \Delta}{LR} \ge \alpha M - \frac{\varepsilon}{2} \ge \left(1 - \frac{1}{e} + \frac{\varepsilon}{2}\right) M,$$

where the second inequality uses $\mathrm{sep}(T) \ge \alpha \mathrm{OPT}_k^+(\mathcal{C}) \ge \alpha LRM$, and the last uses $M \ge 1$. This yields a polynomial-time $(1 - 1/e + \varepsilon/2)$-approximation for MAX k-COVER, contradicting its inapproximability unless $\mathrm{P} = \mathrm{NP}$. □

A.8 PROOF OF THEOREM 5.2

Proof. We use the instance $\mathcal{I}_\Phi$ constructed in Section 5.2. First, we show that if every heavy pair of the gadget for x_i is separated, then $T(V_i)$ is either $\{t_i\}$ or $\{f_i\}$, and that every feasible teaching map can be transformed in polynomial time, without decreasing its weighted value, into one that separates all heavy pairs.

Consider the four singleton targets $S_i^1, \ldots, S_i^4$. If none selects its own example, then each singleton target can separate at most one of the six pairs among them. Indeed, an example distinguishes two singleton targets only if it is one of their two elements. A singleton not selecting its own element can therefore help only the pair with the singleton whose element it selects. Four such choices cannot separate all six pairs. Hence some S_i^p selects u_i^p.

Since u_i^p belongs to both S_i^p and S_i, separating $\{S_i, S_i^p\}$ requires S_i to select an example in $U_i \setminus \{u_i^p\}$. This example also belongs to V_i, so separating $\{V_i, S_i\}$ forces V_i to select from $\{t_i, f_i, g_i\} \cup Q_i$. Applying the same argument to the four singleton targets on the R side forces R_i to select an example in D_i and G_i to select from $\{g_i\} \cup Q_i$. Every possible selected example of G_i then belongs to both G_i and V_i. Among the remaining choices available to V_i, only t_i and f_i lie in $V_i \triangle G_i$. Thus, $\{V_i, G_i\}$ can be separated only when V_i chooses one of these two truth examples.

Conversely, either truth choice for V_i extends to a teaching map separating all twenty-three heavy pairs. Set $T(G_i) = \{g_i\}$, $T(S_i) = \{u_i^0\}$, and $T(R_i) = \{d_i^0\}$. Let each singleton target select its own element. These choices separate all heavy pairs.

Now take any feasible teaching map. If a variable gadget has an unseparated heavy pair, replace the choices of all twelve targets in that gadget by the choices just described, using either truth example for V_i. This gains at least six units on heavy pairs. The only positive-weight pairs connecting these targets to the rest of the instance are the five unit-weight clause pairs incident with V_i, so at most five units can be lost. The replacement therefore increases the weighted value and does not affect heavy pairs in any other gadget. Repeating it for every gadget with an unseparated heavy pair completes the transformation in polynomial time.

Now suppose that every variable target selects a truth example, and let τ assign x_i true exactly when $T(V_i) = \{t_i\}$. We show that, with these choices fixed, the maximum total weight contributed by the three pairs $\{V_i, K_j\}$ with $i \in I_j$ is three if τ satisfies φ_j and two otherwise.

The truth example selected by V_i is outside K_j exactly when the corresponding literal of φ_j is satisfied. Thus, V_i separates its pair with K_j precisely in this case.

The three incident variable targets are pairwise disjoint, and every example in K_j belongs to exactly one of them. Thus, a positive choice by K_j separates its pairs with the other two targets, whereas a negative choice separates at most one incident pair.

If all three literals are false under τ, none of the variable targets separates its clause pair, so at most two of the three pairs can be separated. Choosing any occurrence example $q_{j,i}$ attains two. If a literal belonging to x_i is true, let K_j select $q_{j,i}$. This separates the two pairs involving the other variables, while the selected truth example of V_i separates the remaining pair. The total is then three.

Let $B = 138n + 2m$, and let $\mathrm{OPT}_w(\mathcal{I}_\Phi)$ denote the optimum weighted value of the constructed instance. For any assignment τ, its truth choices and the auxiliary choices described above separate

all $23n$ heavy pairs, contributing $138n$. Choosing each clause target as above produces a teaching map of value $B + \mathrm{sat}_\Phi(\tau)$.

Conversely, apply the normalization above to any feasible map T, and let τ be the assignment given by its truth choices. Optimizing the clause choices as above produces a map $\widehat{T}$ with $\mathrm{sep}_w(\widehat{T}) = B + \mathrm{sat}_\Phi(\tau) \geq \mathrm{sep}_w(T)$ in polynomial time. Together, these two constructions show that

$$\mathrm{OPT}_w(\mathcal{I}_\Phi) = B + \mathrm{OPT}_\Phi. \tag{9}$$

A uniformly random assignment satisfies each clause with probability $7/8$, so $\mathrm{OPT}_\Phi \geq 7m/8$. Since $n = 3m/5$, equation (9) implies $\mathrm{OPT}_w(\mathcal{I}_\Phi) \leq (3427/35)\mathrm{OPT}_\Phi < 98\mathrm{OPT}_\Phi$. For the recovered assignment, $\mathrm{sep}_w(\widehat{T}) \geq \mathrm{sep}_w(T)$ also implies $\mathrm{OPT}_\Phi - \mathrm{sat}_\Phi(\tau) \leq \mathrm{OPT}_w(\mathcal{I}_\Phi) - \mathrm{sep}_w(T)$. These are the two inequalities of an L-reduction (Papadimitriou and Yannakakis, 1991), with constants 98 and 1. The APX-hardness of MAX-E3SAT(5) therefore proves the theorem. □

A.9 Proof of Theorem 6.4

Proof. We reduce from MAXIMUM INDEPENDENT SET (Karp, 1972). Let $G = (V, E)$ be a simple graph with $n = |V|$ and $m = |E|$, and let $\alpha(G)$ denote its maximum independent-set size. Introduce one common example z and three private examples x_v, a_v, b_v for each $v \in V$. For each vertex v, create four targets

$$C_v = \{z, x_v\}, \qquad A_v = \{x_v, a_v\}, \qquad B_v = \{a_v, b_v\}, \qquad D_v = \{b_v, x_v\}.$$

Set $k = 1$. For every v, assign weight one to the five pairs $\{A_v, B_v\}$, $\{B_v, D_v\}$, $\{D_v, A_v\}$, $\{C_v, A_v\}$, and $\{C_v, D_v\}$. Also assign weight one to $\{C_u, C_v\}$ for every edge $uv \in E$. All remaining target pairs have weight zero. This gives $4n$ distinct targets of size two over $3n + 1$ examples, and the construction is polynomial in the size of G.

Adding a positive example cannot decrease the objective, so every target may be assumed to select exactly one example. Fix the choice of C_v. If $T(C_v) = \{z\}$, the choices $T(A_v) = \{a_v\}$, $T(B_v) = \{b_v\}$, and $T(D_v) = \{x_v\}$ separate all five local positive-weight pairs. If $T(C_v) = \{x_v\}$, these same auxiliary choices separate exactly four of the five pairs, with only $\{C_v, D_v\}$ remaining unseparated. Separating all five is impossible in this case. Indeed, separating $\{C_v, A_v\}$ and $\{C_v, D_v\}$ forces $T(A_v) = \{a_v\}$ and $T(D_v) = \{b_v\}$. Then separating $\{A_v, B_v\}$ requires $T(B_v) = \{b_v\}$, while separating $\{B_v, D_v\}$ requires $T(B_v) = \{a_v\}$, a contradiction. Thus, the maximum local contribution is five when C_v selects z and four when it selects x_v.

Let $S = \{v \in V : T(C_v) = \{z\}\}$, and let $E(S)$ be the edge set induced by S in G. For distinct u, v, the targets C_u and C_v intersect only at z, so their pair is unseparated exactly when both $u, v \in S$. Hence the pairs corresponding to edges of G contribute $m - |E(S)|$. The auxiliary choices for different vertices can be optimized independently, since all other pairs involving auxiliary targets have weight zero. For any fixed S, the maximum weighted value is therefore $4n + m + |S| - |E(S)|$.

For every $S \subseteq V$, one can obtain an independent set by repeatedly deleting an endpoint of an edge in $G[S]$. Each deletion removes at least one remaining edge, so at most $|E(S)|$ vertices are deleted. Consequently, $|S| - |E(S)| \leq \alpha(G)$. Equality is attained when S is a maximum independent set. Denote the constructed instance by $\mathcal{I}_G$ and its optimum weighted value by $\mathrm{OPT}_w(\mathcal{I}_G)$. Then,

$$\mathrm{OPT}_w(\mathcal{I}_G) = 4n + m + \alpha(G). \tag{10}$$

Thus, G has an independent set of size at least ℓ if and only if $\mathcal{I}_G$ admits a teaching map of weighted value at least $4n + m + \ell$. This polynomial-time reduction proves the theorem. □

A.10 Proof of Theorem 6.5

Adding a positive example cannot make a previously separated pair unseparated. Thus, when $k = 1$, we may assume that every nonempty target selects exactly one example; the empty target can only select the empty set.

If $A \subsetneq B$, then every positive example selected for the smaller target A also belongs to B and therefore cannot separate the two targets. Whether the pair is separated depends only on whether B selects an example outside A. For every nonempty target $B \in \mathcal{C}$ and every $x \in B$, define

$d_B(x) = \big|\{A \in \mathcal{C} : A \subsetneq B,\ x \in A\}\big|$. This is the number of proper subtargets of B that remain unseparated from B when B selects x.

Disjoint targets are separated, so counting each unseparated comparable pair at its larger target yields the following decomposition.

Lemma A.1. *For every nonempty target B, let $x_B \in B$ be its selected example. The number of unseparated target pairs is $\sum_{B\in\mathcal{C},\, B\neq\varnothing} d_B(x_B)$.*

Proof of Theorem 6.5. For $k = 1$, independently choose an example minimizing $d_B(x)$ for each nonempty target B, as justified by Lemma A.1. The resulting minimum number of unseparated target pairs is $\sum_{B\in\mathcal{C},\, B\neq\varnothing} \min_{x\in B} d_B(x)$. All values $d_B(x)$ can be computed in polynomial time by enumerating targets, their examples, and their proper subtargets.

For $k \geq 2$, let $\mathcal{M}(B)$ be the family of inclusion-wise maximal proper subtargets of a nonempty target B that belong to $\mathcal{C}$. These maximal subtargets are pairwise disjoint, and every proper subtarget is contained in one of them. If $B \setminus \bigcup_{A\in\mathcal{M}(B)} A$ is nonempty, select one example from this set. It belongs to no proper subtarget of B. Otherwise, the maximal proper subtargets cover B, and there are at least two of them. Select one example from each of two distinct maximal proper subtargets. Every proper subtarget lies inside a single maximal one and hence cannot contain both selected examples.

Applying this rule to every nonempty target selects at most two examples per target. Comparable targets are separated by an example selected by the larger one, while disjoint targets are separated by any example selected by either nonempty target. Hence all target pairs are separated. □

B Derandomization by Conditional Expectations

We now explain how to turn the randomized rounding procedures in Sections 3 and 4 into deterministic algorithms. Both procedures consist of finitely many mutually independent random choices. The objective is the final number of separated target pairs, or their total weight in the weighted variants.

Suppose that the next unfixed random choice has several possible outcomes. Conditioned on all choices fixed so far, the weighted average of the conditional expectations associated with these outcomes equals the current conditional expectation. At least one outcome therefore does not decrease the conditional expectation. Selecting such an outcome repeatedly until all random choices are fixed yields a deterministic teaching map whose objective value is at least the expected value of the original randomized algorithm.

For the basic algorithm, one random choice is one trial for a target, and its possible outcomes are any candidate example or the empty outcome. After some trials have been fixed, a target pair that is already separated has conditional success probability one. Otherwise, suppose that the two targets have s and t trials remaining, respectively, and retain the definitions of a and b from Section 3. The conditional probability that neither target ever selects a distinguishing example is $(1-a/k)^s(1-b/k)^t$. Thus, the total conditional expectation associated with every candidate outcome can be computed in polynomial time, and an outcome maximizing it can be selected.

For the configuration LP algorithm, one random choice selects a configuration for one slot of a target. If a pair is not yet separated, its final failure probability is the product of the failure probabilities for all unfixed slots of its two targets. We compute the conditional expectation for each feasible configuration in the current slot. Each slot has at most $|\mathcal{X}|^{O(r)}$ feasible configurations, so this procedure preserves the running time $|\mathcal{X}|^{O(r)}\,\mathrm{poly}(|\mathcal{C}|, k)$.

For the weighted variants, compute each conditional expectation by multiplying each pair's conditional separation probability by $w_{C,C'}$ and summing over all pairs. The probability calculations above are unchanged. Fixing an outcome that does not decrease this sum preserves the expected value of $\mathrm{sep}_w(T)$ and hence the weighted approximation guarantee. For rational input weights, these calculations remain polynomial in the input size.