Attention-Budget Auctions for Generative Content: Truthful Matching Under Trust and Compliance Constraints

Liz Lemma Future Detective

January 16, 2026

Abstract

Generative assistants in 2026 embed sponsored content inside free-form text, tool outputs, and multi-step answers. Unlike classic position auctions, insertion points are contextual and heterogeneous; moreover, platforms face binding constraints from disclosure rules, brand-safety policies, and user-trust objectives. We model ad insertion as a matching problem with a hard attention/trust budget: each advertiser-position pair yields predicted click probability and an additive disruption cost, and feasible allocations must respect both matching constraints and a global budget on disruption. Building on the non-separable, context-dependent view of position auctions in Balseiro et al. (2025), we study implementable mechanisms when the platform must optimize under such constraints. We (i) characterize the constrained winner determination problem as a budgeted matching variant, (ii) give efficient approximation algorithms and identify when near-optimal solutions reduce to a small family of candidate 'cost classes,' (iii) design a monotone constant-factor allocation rule under mild cost-structure assumptions (bounded cost spread or few cost magnitudes), enabling DSIC payments via the envelope theorem, and (iv) derive a price-of-trust bound interpreting the welfare/revenue loss from tighter compliance budgets via the budget's dual variable. The resulting mechanism provides an auditable knob—an explicit disruption budget—that regulators can verify and platforms can tune.

Table of Contents

3. 3. Constrained Winner Determination (WDP-B): Formulation as budgeted matching; computational complexity; LP relaxation and dual interpretation (shadow price of trust).

4. 4. Approximation Benchmarks: PTAS-style results for welfare maximization absent monotonicity (what is achievable algorithmically) and why truthfulness is nontrivial under budgets.

5. 5. A Monotone Constant-Factor Allocation Rule Under Structured Costs: Cost-bucketing + greedy maximal matching + deterministic tie-breaking; monotonicity proof; approximation factor.

6. 6. Mechanism Design: Envelope payments for welfare; virtual values for revenue under regularity; ε-IC from numerical integration/discretization (flag when needed).

7. 7. Price of Trust: Comparative statics in B, concavity/upper bounds, and dual-based welfare loss bounds; interpretation as 'trust tax' or 'compliance rent.'

8. 8. Extensions and Discussion: (i) cascade/MNL substitution (how the budget interacts with externalities), (ii) multiple budgets (disclosure + brand safety), (iii) auditing and governance implications.

9. 9. Conclusion and Open Problems: Removing structured-cost assumptions; robust/learning-based (p_{ij}, d_{ij}); dynamic trust budgets across turns.

1 Introduction

Large language models have shifted sponsored content from a separate "ad block" into the body of a generated response. In a conversational interface, the platform typically has many candidate places where a short sponsored insertion could plausibly appear: a product recommendation in a list, a cited source, a sidebar card, or a brief "sponsored" sentence inserted between paragraphs. These *generative insertion points* are not fixed ex ante in the same way as classical search slots; they are produced (and can be pruned) as part of response generation. In practice, the platform evaluates a set of candidates—which we can think of as potential advertiser–position pairs—and decides which, if any, to insert.

The central economic tension in this environment is not only about relevance and revenue, but also about *trust and compliance*. In 2026, disclosure rules, brand-safety requirements, and user-experience constraints are binding design parameters rather than afterthoughts. Many platforms now treat "ad load" in generative responses as a hard constraint: too many disclosures, too prominent a footprint, or too aggressive a placement can trigger user dissatisfaction, regulator scrutiny, or partner restrictions. Importantly, these harms are not well captured by the classical objective of maximizing expected click value subject to per-slot feasibility. Instead, the platform must manage a *global* budget of disruption across the entire response, reflecting that user trust is a cumulative resource and that compliance teams often approve experiences at the level of an interaction, not a slot.

This paper develops a mechanism-design formulation that makes this tradeoff explicit. We model each potential insertion as producing two primitives: a predicted click benefit and a predicted disruption cost. The click side corresponds to standard auction inputs (expected click-through conditional on placement), while the disruption side represents the expected "trust footprint" of that insertion—for example, the visual salience of disclosure, the sensitivity of the surrounding topic, the probability of user complaints, or the incremental risk of confusing sponsored and non-sponsored content. The platform faces a hard cap on total disruption in the interaction, which can be interpreted as an auditable product policy or a regulator-imposed constraint. The key question is then: *how should the platform allocate sponsored insertions and set payments when feasibility is jointly constrained by matching structure (at most one advertiser per position and vice versa), by a cap on the number of insertions, and by a global trust budget?*

A natural first guess is to treat this as "sponsored search with another constraint" and to reuse separability arguments. That instinct is misleading for two reasons. First, even under the simplest click model in which click probabilities are additive across positions, the trust budget couples decisions across edges: selecting a high-value insertion at one location may force the platform to forgo several low-disruption insertions elsewhere. This destroys

the convenient decomposability that often underlies both optimal allocation and incentive analysis. Second, the platform's candidate positions are not fixed slots with exogenous position multipliers; they are *endogenous candidates* generated by a model, where the click and disruption primitives depend on the local semantics of the response and on disclosure implementation. As a result, the usual "rank by a score and fill all slots" logic can be infeasible: the platform may have to leave some candidate positions empty, not because demand is low, but because inserting ads would exceed the permissible disruption footprint.

At a high level, our approach is to separate what is conceptually simple from what is computationally and incentive-theoretically delicate. Conceptually, the welfare objective remains linear in expected clicks, and the constraints are transparent: a matching structure plus a single knapsack-style constraint capturing trust. This immediately yields an interpretable policy object: the marginal value of trust. Computationally, however, the resulting optimization is a *budgeted matching* problem, which is substantially harder than the unconstrained assignment that appears in classical position auctions. Incentive-theoretically, even if one is willing to accept approximate welfare maximization, standard approximation schemes can fail the monotonicity property needed for dominant-strategy truthfulness. The main contribution of the paper is to show that in economically relevant regimes—where disruption costs have limited heterogeneity or can be meaningfully bucketed—there exist allocation rules that are simultaneously (i) polynomial-time, (ii) approximately welfare-optimal up to a constant factor, and (iii) monotone in bids, enabling DSIC mechanisms via envelope payments.

The paper is motivated by a pragmatic observation: disruption costs in real systems are often *coarsely structured*. Compliance teams do not typically provide a continuum of permissible footprints; instead, they define a small number of disclosure templates (e.g., light, standard, heavy), a small number of sensitive-topic levels, or a small number of UI surfaces where sponsorship is allowed. When combined, these features produce a modest number of disruption "classes." Even when raw predictions are continuous, product policy often requires rounding to a finite menu for auditability. This structure is precisely what allows us to recover monotonicity and tractable approximation: by bucketing edges into a small number of cost scales, we can apply deterministic greedy matching procedures within each scale and then combine them using a standard knapsack comparison argument. The resulting allocation is stable under bid increases, avoiding the non-monotonicities that arise in PTAS-style "guess-and-optimize" routines.

We also emphasize a policy-relevant byproduct of the formulation: the *shadow price of trust*. Relaxing the hard trust budget with a Lagrange multiplier yields a natural "tax" interpretation—each insertion is scored not only by its bid-weighted click contribution but also by its disruption cost times a

common multiplier. This multiplier can be read as the platform's marginal willingness to trade welfare (or revenue) for an additional unit of permissible disruption. In settings where the budget is externally set, the multiplier provides an economically meaningful diagnostic: when it is high, the system is trust-constrained and additional "trust capacity" (e.g., better disclosure UI, improved targeting to reduce disruption, or looser policy) would have large welfare consequences; when it is low, the system is effectively constrained by other limits (such as the cap on the number of insertions). While the exact multiplier is defined most cleanly in the linear relaxation, it offers a concrete bridge between mechanism design and auditable product governance.

Our formal results proceed in three steps, mirroring this logic. First, we show that the welfare-maximizing feasible allocation under the base-click model can be written as a matching problem with an additional knapsack constraint on disruption. This step is mostly bookkeeping, but it is important because it clarifies that the only source of coupling across positions, in the base model, is the trust budget. Second, we study the Lagrangian relaxation and interpret the dual variable as the price of trust. This yields comparative statics that align with platform intuition: as the trust budget increases, welfare rises and the shadow price weakly falls (in the relaxed problem), reflecting diminishing returns to additional permissible disruption. Third, we address incentives and computation. We show that, under structured disruption costs (few distinct values after rounding, or bounded spread), we can design deterministic monotone allocation rules with constant-factor approximation guarantees. Plugging these monotone rules into the standard envelope formula yields DSIC and individually rational mechanisms. Under regularity, the same construction can be applied to virtual values, producing an approximately revenue-optimal mechanism subject to the same trust budget.

It is worth underscoring what we *do not* claim. We do not claim that trust costs are truly exogenous or perfectly measurable; rather, we take the measurement process as an institutional primitive, much like click-through rate estimation in sponsored search. In practice, disruption costs will be learned, audited, and sometimes disputed. Our framework is useful precisely because it makes the dependence on these measurements explicit and because it yields interpretable sensitivity objects (e.g., the shadow price) that can be monitored as measurement pipelines evolve. We also do not claim that the base-click model captures all user behavior. In conversational settings, insertions may cannibalize organic engagement or affect the probability of later clicks (substitution and spillovers), and the order of presentation can matter. We view the base-click model as a deliberately transparent starting point: it isolates the mechanism-design consequences of a hard trust budget, and it provides a baseline against which richer behavioral models can be compared.

The relationship to the recent literature is closest to work on constrained

auctions and online advertising with additional feasibility constraints. In particular, **?** emphasize that modern ad allocation problems increasingly include non-standard constraints—pacing, budgets, and platform-side limits—and they develop tools for understanding welfare and revenue in such environments. Our setting is complementary in two ways. First, we focus on *generative* insertion points, where the platform's feasible actions resemble a matching over a set of candidate opportunities that are local to a generated response rather than a fixed slate of slots. This makes the trust constraint interaction-level and naturally modeled as a knapsack budget across candidate insertions. Second, we place incentive compatibility at the center of the analysis: our primary design requirement is DSIC in a single-parameter environment, which forces us to confront monotonicity failures that are often benign in purely algorithmic approximations. In that sense, our results can be read as identifying structural conditions (coarse cost classes or bounded spread) under which constrained allocation remains compatible with truthful mechanisms without sacrificing computational tractability.

A further distinction is interpretability. In many applied constrained-allocation systems, constraints are treated as engineering requirements, and the auction is tuned around them. Our formulation treats the trust budget as a first-class economic constraint and gives it a dual interpretation that can be reported, audited, and potentially regulated. This is especially relevant for disclosure compliance. A regulator might not dictate the platform's ranking rule, but it can set or verify a bound on permissible disruption (or require reporting of the implied shadow price). Conversely, a platform can use the shadow price internally to decide whether investments in safer ad formats or improved disclosure design would relax the constraint effectively (reducing d_{ij}) and thereby increase welfare or revenue without increasing ad load.

Finally, we note that generative interfaces raise a subtle but important normative issue: the constraint is not only about "how many" ads appear, but about *where* and *how* they appear. A single highly intrusive insertion may be worse than two mild ones, and this depends on context. Modeling disruption at the edge level, and then constraining it globally, accommodates this heterogeneity while keeping the mechanism analyzable. It also makes clear why classic separability fails: even if clicks decompose additively across insertions, trust does not. The platform is solving a problem of allocating a scarce, interaction-level resource (trust capacity) across heterogeneous opportunities.

The rest of the paper formalizes this setting and develops mechanisms that are both economically principled and operationally plausible. We begin by specifying the primitives—candidate positions, click and disruption predictions, and feasibility constraints—and we contrast the base-click benchmark with brief extensions that capture substitution effects. We then derive the welfare program, interpret the dual, and construct monotone approximation algorithms under structured costs, leading to DSIC mechanisms with

transparent performance guarantees.

2 Model

We fix a conversational context c (a user query together with any relevant conversational state) and study the platform's problem of inserting sponsored content into the generated response. Unlike classical position auctions in which the set of slots is fixed ex ante, a generative system typically produces a *menu of plausible insertion opportunities* as part of response construction. We take this menu as given within the interaction: there are m candidate insertion positions indexed by $j \in [m]$. A candidate position can be interpreted broadly—a sentence-level insertion between paragraphs, a short product card adjacent to a list item, or a "sponsored" citation embedded in a recommendation. The platform may also choose to leave a candidate empty.

There are n advertisers indexed by $i \in [n]$. Each advertiser has a single-parameter private value $v_i \geq 0$, interpreted as value per click (or per attributable action, normalized to a click). Advertisers are risk-neutral and have quasi-linear utility. The platform designs an allocation rule and a payment rule and commits to them before bids are submitted. We focus on dominant-strategy incentive compatibility (DSIC), which is operationally attractive in advertising settings because it does not require bidders to reason about others' beliefs or equilibrium selection in a fast-moving environment.

Predicted primitives: clicks and disruption. For each advertiser–position pair (i, j), the platform observes a pair of primitives

$$(p_{ij}, d_{ij}) \in [0, 1] \times \mathbb{R}_+.$$

The first component p_{ij} is a predicted *standalone* click probability if advertiser i is inserted at position j.[1] The second component d_{ij} is a predicted *disruption* or *trust/compliance* cost associated with placing advertiser i at position j. This cost is meant to summarize the incremental footprint of sponsorship in that local semantic and UI context: disclosure salience, topic sensitivity, brand-safety risk, likelihood of user dissatisfaction, or the probability that the insertion triggers a policy violation. In applications, both p_{ij} and d_{ij} may be learned from data, but for the mechanism-design analysis we treat them as known inputs to the platform at allocation time.

A key modeling choice is that disruption is treated as a *resource constraint* rather than as a term in the platform's objective. We impose a hard budget $B \geq 0$ on the total disruption in the interaction. This captures the fact that compliance and product policy are often expressed as requirements of the

[1] "Standalone" here means that p_{ij} abstracts from cross-effects of other sponsored insertions. We use this as the baseline model and return to interaction effects below.

form "the experience must not exceed a certain disclosure or risk footprint," and that such requirements are audited at the interaction level. Formally, the platform must choose an allocation whose total disruption does not exceed B. We think of B and the measurement protocol for d_{ij} as institutional primitives: they may be set by an internal governance process, by a regulator, or by contractual obligations with partners.

Allocations as matchings with caps. An allocation specifies which advertiser, if any, is inserted at each candidate position. We represent allocations by binary variables $x_{ij} \in \{0,1\}$, where $x_{ij} = 1$ means advertiser i is assigned to position j. The generative setting naturally imposes two matching-style constraints: each advertiser can be inserted at most once in the response, and each position can contain at most one advertiser. We also impose a cap $K \leq m$ on the total number of sponsored insertions in the response, reflecting an "ad load" limit that is separate from (and potentially less binding than) the disruption budget. The feasible set of matchings is

$$X \;=\; \Big\{x \in \{0,1\}^{n\times m} \;:\; \sum_{j=1}^{m} x_{ij} \leq 1 \;\forall i,\; \sum_{i=1}^{n} x_{ij} \leq 1 \;\forall j,\; \sum_{i=1}^{n}\sum_{j=1}^{m} x_{ij} \leq K\Big\}.$$

The trust/compliance constraint then restricts X further to the budget-feasible subset

$$X(B) \;=\; \Big\{x \in X \;:\; \sum_{i=1}^{n}\sum_{j=1}^{m} d_{ij}x_{ij} \leq B\Big\}.$$

The distinguishing feature of this environment is that the disruption budget couples decisions across positions: even when click values add across insertions, feasibility does not. In particular, a high-disruption placement can crowd out multiple low-disruption placements that would otherwise be permissible under the ad-load cap K.

Timing and information. The timing within an interaction is as follows. First, context c arrives and the platform determines the candidate positions $j \in [m]$ and associated primitives (p_{ij}, d_{ij}) for each advertiser. The platform commits to the mechanism (allocation rule and payment rule) and to the disruption accounting protocol and budget B. Second, advertisers submit bids b_i; under DSIC we interpret b_i as a truthful report of v_i in equilibrium, but we keep the bid notation to separate the reported type from the private value. Third, the platform computes an allocation $x(b) \in X(B)$ and generates the response with the corresponding sponsored insertions. Fourth, user clicks are realized. Finally, payments are assessed.

Advertisers observe their own values v_i and the mechanism, but do not observe others' values. The platform observes (p_{ij}, d_{ij}) and B. For the

theoretical development we treat (p_{ij}, d_{ij}) as common knowledge primitives, in the same spirit that classical models treat click-through rates or quality scores as known inputs to the auction; the mechanisms we study remain well-defined when these primitives are estimated, provided the estimation is not manipulable by individual bidders.

Base-click model and welfare. Our baseline behavioral model assumes that click probabilities are additive across positions and insensitive to the presence of other sponsored insertions. Under an allocation x, advertiser i's total click probability is

$$\pi_i(x) \;=\; \sum_{j=1}^{m} p_{ij} x_{ij}.$$

Given bids b, the induced interim "allocation" in the single-parameter sense is $y_i(b) = \pi_i(x(b))$, i.e., the total click probability assigned to advertiser i. Advertiser i's quasi-linear utility when her true value is v_i and the bid profile is b is

$$u_i(b; v_i) \;=\; v_i\, y_i(b) \;-\; t_i(b),$$

where $t_i(b)$ is the payment charged by the platform.

The platform's welfare objective, given a value profile v, is the expected total value from clicks:

$$W(x; v) \;=\; \sum_{i=1}^{n} v_i \pi_i(x) \;=\; \sum_{i=1}^{n} \sum_{j=1}^{m} v_i p_{ij} x_{ij}.$$

Thus, under the base-click model, welfare is linear in the allocation variables with edge weights $v_i p_{ij}$. This linearity is conceptually useful: it isolates the role of the trust budget as the sole source of interaction across candidate insertions in the baseline model. However, linearity does not imply that the allocation problem is easy, because the disruption budget introduces a knapsack-style constraint on top of matching structure.

Mechanism requirements: DSIC, monotonicity, and envelope payments. Because each advertiser has a single private parameter, DSIC reduces to the standard monotonicity-and-envelope structure. An allocation rule is DSIC-implementable (with appropriate payments) if $y_i(v_i, v_{-i})$ is non-decreasing in v_i for every fixed v_{-i}. Once monotonicity holds, payments are pinned down (up to a constant normalization) by the envelope formula:

$$t_i(v) \;=\; t_i(0, v_{-i}) + v_i y_i(v) - \int_0^{v_i} y_i(z, v_{-i})\, dz,$$

and we adopt the individually rational normalization $t_i(0, v_{-i}) = 0$. The substantive design challenge in our setting is that the trust budget makes the

winner determination problem a constrained combinatorial optimization, and many natural approximation algorithms for such problems are not monotone in bids. Our aim is therefore not only to allocate efficiently subject to B, but to do so with an allocation rule that respects monotonicity and can be coupled with envelope payments.

Interpretation of disruption costs and the role of structure. While our results treat d_{ij} as an input, it is helpful to be explicit about why d_{ij} is modeled at the edge level. In generative systems, the same advertiser can be more or less disruptive depending on where the insertion lands: a brief product mention in a shopping query may require minimal disclosure, whereas an insertion in a medical or political context may be heavily constrained or effectively disallowed. Edge-level disruption captures precisely this interaction between advertiser identity and local context.

At the same time, there is an important limitation: in reality, disruption is unlikely to be perfectly cardinal and additive. The total harm from two insertions may be more than the sum of their harms (e.g., compounding loss of trust), or less (e.g., disclosures share UI real estate). We nevertheless impose an additive budget as a transparent benchmark that supports auditing and policy communication. In later sections, when we interpret the shadow price of trust, the additivity assumption is what gives the dual variable a clean "marginal value" meaning.

A further pragmatic motivation for our later algorithmic conditions is that disruption assessments are often *coarsely categorized*. Compliance processes frequently map contexts and disclosure templates into a small menu (e.g., light/standard/heavy disclosure; low/medium/high sensitivity). Even when raw scores are continuous, they are often rounded for governance and reproducibility. This kind of structure is not merely a modeling convenience: it is what makes monotone approximate allocation feasible in polynomial time.

Beyond the base-click model: substitution and interaction effects (brief). The additive click model is intentionally conservative: it assumes the platform can estimate p_{ij} independent of other insertions and that user attention is not reallocated. Generative interfaces plausibly violate both assumptions. Two broad classes of extensions are natural.

First, *substitution or cannibalization* may occur when multiple sponsored insertions compete for the same user attention. One reduced-form way to capture this is to let the effective click probability at position j depend on the total number of insertions:

$$\pi_i(x) \;=\; \sum_{j=1}^{m} p_{ij}\, g\left(\sum_{i',j'} x_{i'j'}\right) x_{ij},$$

where $g(\cdot)$ is a nonincreasing "attention dilution" function. This preserves separability across advertisers conditional on the total sponsored load, but introduces an additional coupling term beyond the disruption budget. In such a model, the cap K becomes not only a policy constraint but also an endogenous welfare-relevant parameter.

Second, *trust spillovers* may affect user behavior even when the number of insertions is fixed. For example, a high-disruption insertion may reduce the likelihood that the user engages with subsequent content, including other sponsored items. A simple way to express this is to let the click probability be scaled by a function of total disruption:

$$\pi_i(x) \;=\; \sum_{j=1}^{m} p_{ij}\, h\left(\sum_{i',j'} d_{i'j'} x_{i'j'}\right) x_{ij},$$

where $h(\cdot)$ is nonincreasing. Here, the disruption budget B can be viewed as an institutional substitute for directly optimizing such spillovers: rather than requiring the platform to trade off continuous trust effects in the objective, policy imposes a hard constraint that keeps the system within a safe region where $h(\cdot)$ is not too small. This interpretation aligns with real governance practice: teams often prefer enforceable limits (with audit trails) over delicate objective-function tuning.

We do not attempt to solve the general interactive model in this paper. The base-click benchmark provides a clean separation between (i) linear welfare and (ii) a global feasibility constraint representing trust/compliance. This separation is what allows us to focus sharply on the computational and incentive consequences of the budget. In the next section, we formulate the platform's allocation problem under the base-click model as a constrained winner determination problem, and we show how the trust budget induces a budgeted matching structure with a natural dual interpretation.

3 Constrained Winner Determination Under a Trust Budget

Fix the context c and a bid profile $b \in \mathbb{R}^n_+$. The platform's immediate computational task is to choose which sponsored insertions to make—and where to place them—subject to both the matching constraints (at most one insertion per advertiser and per position) and the hard disruption budget. Because the click model is linear in x, this task takes the form of a *constrained winner determination problem* with a single knapsack-style coupling constraint.

WDP-B: an integer program with matching and a knapsack constraint. Let $w_{ij}(b) = b_i p_{ij}$ denote the welfare weight of assigning advertiser i to position j under bids b (equivalently, the reported value per click times

predicted click probability). The constrained winner determination problem is

$$\text{(WDP-B)} \qquad \max_{x \in X(B)} \sum_{i=1}^{n} \sum_{j=1}^{m} w_{ij}(b)\, x_{ij} \;=\; \max_{x \in X(B)} \sum_{i=1}^{n} \sum_{j=1}^{m} b_i p_{ij}\, x_{ij}. \quad (1)$$

We emphasize two structural features. First, (1) is a *bipartite matching* problem when the disruption budget is removed: if we drop $\sum_{i,j} d_{ij} x_{ij} \le B$, the remaining constraints define a standard assignment polytope with an additional cardinality cap $\sum_{i,j} x_{ij} \le K$, solvable in polynomial time (e.g., by reducing to a min-cost flow). Second, it becomes a *pure knapsack* problem when matching constraints are removed: if each advertiser could be selected independently (no position conflicts), we would simply pick up to K items under a budget. The generative setting forces us to confront both types of constraints at once.

Why the trust budget is the source of computational difficulty. Absent the disruption constraint, the feasible region is described by a totally unimodular matrix, so the natural LP relaxation is integral; in other words, we do not face fractional allocations in the classical assignment environment. The trust budget breaks this property: intersecting an integral matching polytope with a single knapsack inequality can create fractional extreme points and, more importantly, pushes the problem into the terrain of NP-hard "budgeted matching" (also called knapsack-matching).

A simple reduction makes this precise. Consider an instance in which each advertiser i is eligible for at most one position (say, a unique $j(i)$ with $p_{ij(i)} = 1$ and $p_{ij} = 0$ for $j \neq j(i)$), and set $K = m = n$ so the ad-load cap never binds. Then the matching constraints force $x_{ij(i)} \in \{0, 1\}$ independently, and (1) reduces to

$$\max \sum_{i=1}^{n} b_i x_i \quad \text{s.t.} \quad \sum_{i=1}^{n} d_i x_i \le B,\; x_i \in \{0, 1\},$$

which is exactly the 0–1 knapsack problem. Thus WDP-B is NP-hard in general, even under very sparse eligibility structure. In richer cases where multiple advertisers compete for the same positions, the problem remains hard for essentially the same reason: the platform must decide not only *which* advertisers to include but also *how* to resolve conflicts across positions under a global budget. This is why we treat (1) as the computational bottleneck in the mechanism.

Exact solution methods and what they buy us. In practice, a platform could attempt to solve (1) exactly via integer programming (branch-and-bound with cutting planes) or via specialized combinatorial methods for

budgeted matchings. Exact methods are useful in two ways: (i) they can be deployed at moderate scale for offline evaluation, policy calibration, or as an oracle in learning-to-rank pipelines; and (ii) they provide benchmarks for the welfare–trust frontier. However, exact solution is not a satisfying default for an online mechanism, both because worst-case running time can be large and because DSIC requires a deterministic, well-specified tie-breaking policy that is stable across inputs.[2] These considerations motivate studying relaxations and approximation algorithms, and in particular studying which approximations can be made compatible with monotonicity.

3.1 LP Relaxation and a Dual "Price of Trust"

The LP relaxation. A natural first step is to relax integrality and allow fractional assignments $x_{ij} \in [0, 1]$:

$$\begin{aligned} \max_{x \in [0,1]^{n \times m}} \quad & \sum_{i=1}^{n} \sum_{j=1}^{m} w_{ij}(b)\, x_{ij} \\ \text{s.t.} \quad & \sum_{j=1}^{m} x_{ij} \le 1 \qquad \forall i, \\ & \sum_{i=1}^{n} x_{ij} \le 1 \qquad \forall j, \\ & \sum_{i=1}^{n} \sum_{j=1}^{m} x_{ij} \le K, \\ & \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} x_{ij} \le B. \end{aligned} \tag{2}$$

We denote the optimal value by $W^{LP}(B; b)$ to emphasize the dependence on the budget B and bids b. When the budget constraint is removed, this LP is integral; with the budget constraint present, the relaxation can be loose, but it is still extremely informative: it yields (i) an upper bound on the integral optimum, and (ii) a dual certificate that admits a clean economic interpretation.

The dual program and interpretable multipliers. Associate dual variables $\alpha_i \ge 0$ with the advertiser constraints $\sum_j x_{ij} \le 1$, $\beta_j \ge 0$ with the position constraints $\sum_i x_{ij} \le 1$, $\lambda \ge 0$ with the cardinality cap $\sum_{i,j} x_{ij} \le K$,

[2] Tie-breaking is not an implementation footnote in DSIC environments: even when the objective is well-defined, arbitrary tie-breaking can create non-monotonicities in the induced allocation rule $y_i(\cdot)$ at measure-zero bid profiles, which complicates the payment definition and auditing.

and $\mu \geq 0$ with the trust budget $\sum_{i,j} d_{ij} x_{ij} \leq B$. The dual of (2) can be written as

$$\begin{aligned} \min_{\alpha,\beta,\lambda,\mu} \quad & \sum_{i=1}^{n} \alpha_i \; + \; \sum_{j=1}^{m} \beta_j \; + \; K\lambda \; + \; B\mu \\ \text{s.t.} \quad & \alpha_i + \beta_j + \lambda + \mu d_{ij} \; \geq \; w_{ij}(b) \qquad \forall (i,j), \\ & \alpha_i, \beta_j, \lambda, \mu \; \geq \; 0. \end{aligned} \tag{3}$$

Two points matter for our purposes. First, the constraint

$$\alpha_i + \beta_j + \lambda + \mu d_{ij} \; \geq \; w_{ij}(b)$$

is a "no-arbitrage" condition: the adjusted cost of using advertiser i and position j, plus the global per-insertion charge λ and the per-disruption charge μd_{ij}, must cover the edge value $w_{ij}(b)$. Second, the objective makes the policy parameters transparent. The multiplier λ prices the ad-load cap (how much welfare we gain from one more allowed insertion), while μ prices the disruption budget (how much welfare we gain from one more allowed unit of disruption). It is μ that we interpret as a *shadow price of trust*.

A Lagrangian view: "tax" disruption and solve a matching. The dual variable μ also emerges from a Lagrangian relaxation. If we move the budget constraint into the objective with penalty $\mu \geq 0$, we obtain the penalized problem

$$\max_{x \in X} \sum_{i,j} \Big(w_{ij}(b) - \mu d_{ij} \Big) x_{ij} \; + \; \mu B, \tag{4}$$

where X is the feasible set without the disruption constraint (but still with matching and the cap K). For a fixed μ, (4) is simply a maximum-weight matching (with a cardinality cap) under *modified weights*

$$\widetilde{w}_{ij}(\mu; b) \; = \; w_{ij}(b) - \mu d_{ij}.$$

This observation is operationally important. It says that if governance can agree on a trust price μ, then the platform can implement the allocation by running a standard assignment algorithm on *trust-adjusted* edge weights, effectively treating disruption as a linear "tax." Conversely, if governance insists on a hard budget B, then μ is the implicit exchange rate between welfare and trust required to rationalize the constrained choice.

The welfare–budget frontier and diminishing returns. Define the relaxed value function $B \mapsto W^{LP}(B; b)$. Standard LP sensitivity analysis implies that this function is nondecreasing and concave in B, capturing diminishing returns to additional permitted disruption: the platform spends

the first units of budget on the most cost-effective edges (high w_{ij} per unit d_{ij}), and later units buy progressively less. Moreover, any optimal dual solution $(\alpha^*, \beta^*, \lambda^*, \mu^*)$ at budget B furnishes a local bound on the slope: μ^* is a subgradient of $W^{LP}(B; b)$ with respect to B. In words, if governance were to loosen the budget marginally from B to $B + \Delta$, the relaxed welfare upper bound increases by approximately $\mu^* \Delta$ for small Δ.

We view this as more than a mathematical convenience. A recurring operational question in generative advertising is how to communicate the consequences of changing disclosure or risk policy. The scalar μ^* offers a compact, auditable statistic: it is the marginal welfare value of trust budget in the current context and bid environment. Reporting μ^* (or a robustified version of it) can therefore serve as a governance interface between product policy (which controls B and the accounting of d_{ij}) and auction design (which controls allocation and payments).

Complementary slackness as an "explainability" tool. The primal–dual pair (2)–(3) also yields a useful explanation vocabulary. Complementary slackness implies that if an edge (i, j) receives positive fractional assignment, then its adjusted value must be exactly tight:

$$w_{ij}(b) \;=\; \alpha_i^* + \beta_j^* + \lambda^* + \mu^* d_{ij}.$$

Thus, in the relaxed model, every chosen insertion can be "explained" as having value equal to the sum of (i) an advertiser-side scarcity term, (ii) a position-side scarcity term, (iii) an ad-load term, and (iv) a trust term proportional to d_{ij}. While we do not claim this decomposition is a complete interpretability solution, it is the correct economic accounting for why some edges are excluded: they fail to clear the combined scarcity prices.

From relaxation to mechanisms: what remains difficult. The LP and its dual do *not* resolve the full mechanism design problem. First, LP solutions can be fractional and must be rounded to obtain a valid matching; rounding can lose welfare and can interact poorly with incentive constraints if done naively. Second, even if one uses the Lagrangian form (4) to produce integral matchings for each μ, tuning μ to meet the hard budget can lead to nontrivial discontinuities: small bid changes can flip which μ (or which matching at a fixed μ) is selected, and such flips are precisely what can break monotonicity for approximation procedures. Third, the integrality gap of the relaxation can be meaningful, so dual prices should be interpreted as prices for the *relaxed* frontier unless accompanied by a rounding argument.

These caveats sharpen the role of the next section. The constrained winner determination problem (1) is computationally hard, but we can still ask: *how well can we approximate it in polynomial time* if we ignore incentive constraints? And, once we know the algorithmic frontier, *which approximation*

approaches can be made monotone so that envelope payments yield a DSIC mechanism? Section 4 takes up these questions by separating approximation benchmarks (what is achievable computationally) from the additional structure required for truthfulness under a hard trust budget.

4 Approximation Benchmarks and Why Truthfulness is Nontrivial

Our next step is to separate two questions that are easy to conflate in practice. The first is purely algorithmic: if we ignore incentives, how close can we get to the welfare optimum of WDP-B in polynomial time? The second is mechanistic: can we make such approximations compatible with dominant-strategy truthfulness under a hard trust budget? The short answer is that the algorithmic frontier is surprisingly strong (PTAS-style guarantees exist), but these procedures typically fail the monotonicity property required for DSIC payments. This section develops that benchmark and explains where the incentive difficulty comes from.

Approximation as a benchmark for the welfare–trust frontier. Even when the platform is not literally maximizing welfare (e.g., it maximizes revenue via virtual values), approximation results for welfare are the right first yardstick. They quantify how much computational loss we incur relative to the best feasible matching under the same budget B and cap K, holding fixed the primitives (p_{ij}, d_{ij}). Put differently, approximation guarantees describe the attainable frontier in the *engineering* sense: what can be implemented at scale if we temporarily ignore the strategic response of advertisers.

We write

$$\mathrm{OPT}(b; B) \;=\; \max_{x \in X(B)} \sum_{i,j} w_{ij}(b)\, x_{ij}$$

for the integral optimum. An algorithm $\mathcal{A}$ is a $(1-\varepsilon)$-approximation if it returns an allocation $\widehat{x} = \mathcal{A}(b)$ with value at least $(1-\varepsilon)\mathrm{OPT}(b; B)$. In the present setting this is the relevant notion because, absent truthfulness constraints, welfare approximation composes cleanly with standard ex post evaluation: one can compare achieved welfare and total disruption to offline optima or upper bounds from (2).

4.1 PTAS-style approximation for budgeted matching (ignoring incentives)

Why a PTAS is plausible despite NP-hardness. WDP-B is NP-hard by the knapsack reduction already discussed, so exact polynomial-time optimization is out of reach unless P = NP. Nonetheless, the structure of the problem—a matching constraint system plus a *single* knapsack budget—is

precisely the kind of hybrid feasibility where polynomial-time approximation schemes are often possible. Intuitively, one can "factor" the hardness into a small number of "heavy" decisions: the few edges that consume a nontrivial fraction of the budget. Once those are fixed (or guessed), what remains is a lower-impact residual instance in which discretization and rounding introduce only ε-level loss.

At a high level, PTAS frameworks for knapsack-coupled combinatorial problems proceed by combining two ideas:

1. *Guessing or enumerating heavy structure.* Identify a small set of edges whose costs or contributions are large relative to B or to the optimum value. Because there can be only $O(1/\varepsilon)$ such edges in any feasible solution, we can enumerate their identities in time polynomial in n, m for fixed ε.

2. *Solving the residual instance optimally (or near-optimally).* After removing conflicts caused by the guessed edges and reducing the remaining budget, the residual problem has bounded granularity: costs can be scaled and rounded so that dynamic programming, min-cost flow variants, or LP rounding becomes near-exact.

The tension is that matching constraints make the residual problem nonseparable across advertisers and positions, but they are also the reason why the residual subproblem is algorithmically well behaved: matchings admit polynomial-time optimization under many types of weight perturbations and side constraints.

One concrete PTAS template. To make the above intuition more concrete, consider the following stylized PTAS template (we present it as a benchmark rather than as an implementation recipe). Fix $\varepsilon > 0$ and define an edge (i, j) to be *budget-heavy* if $d_{ij} > \varepsilon B$. Any feasible allocation uses at most $1/\varepsilon$ budget-heavy edges, since their total disruption is at most B. We can therefore enumerate every feasible set H of at most $\lceil 1/\varepsilon \rceil$ edges that is *matching-feasible* (no shared advertiser or position) and respects the cap K.[3] For each such guess H, we (i) commit to inserting all edges in H, (ii) delete their incident advertisers and positions from the graph, (iii) reduce the remaining budget to $B(H) = B - \sum_{(i,j)\in H} d_{ij}$ and the remaining cardinality to $K(H) = K - |H|$, and then (iv) solve the residual problem on the remaining bipartite graph with *small* edge costs $d_{ij} \le \varepsilon B$.

In the residual instance, we can discretize costs by rounding each d_{ij} up to the nearest multiple of $\varepsilon^2 B/K$ (or another scale chosen to control the additive error). Because any feasible solution uses at most K edges, rounding

[3] One can also enrich the guessing step by including "value-heavy" edges, or by guessing the top few edges in the optimal solution under an appropriate profit density. The precise choice does not matter for our purposes; what matters is that the heavy part has bounded cardinality.

introduces at most $\varepsilon^2 B$ additional disruption, which can be absorbed by a small tightening of the budget or by a standard "repair" step. After rounding, the remaining costs take only $O(K/\varepsilon^2)$ distinct values, and the residual problem can be solved to near-optimality via pseudo-polynomial dynamic programming layered over a matching oracle (equivalently, one can view it as a min-cost flow with an additional discretized resource dimension). Taking the best solution over all guesses H yields an overall $(1-\varepsilon)$ approximation for fixed ε with running time polynomial in n, m (and typically quasi-polynomial or polynomial with a large constant in $1/\varepsilon$).

This is the content of Proposition 3 in the global summary: algorithmically, budgeted matching admits PTAS-style approximation benchmarks. The important point for our narrative is not the exact details of the scheme, but the *shape* of the guarantee: we can get arbitrarily close to $\mathrm{OPT}(b; B)$ if we are willing to accept an ε-dependent polynomial and to ignore incentive compatibility.

Alternative benchmark: Lagrangian search plus rounding. A second, operationally appealing benchmark derives from the Lagrangian perspective. For any $\mu \geq 0$, let $x(\mu) \in \arg\max_{x \in X} \sum_{i,j} (w_{ij}(b) - \mu d_{ij}) x_{ij}$ be an optimal matching under trust-adjusted weights. As μ increases, the algorithm penalizes disruption more heavily and the selected matching tends to use less budget. One can therefore perform a search over μ to find two neighboring matchings $x(\mu^-)$ and $x(\mu^+)$ whose disruption straddles B and then combine them (e.g., by randomized mixing or by selecting one and repairing) to obtain a near-feasible solution. This paradigm often performs extremely well in practice because each inner problem is a standard assignment instance. As a *benchmark*, it also links back to the dual interpretation: the approximation loss can be viewed as the cost of converting a soft trust price into a hard trust budget.

However, this benchmark already foreshadows the incentive issue. The mapping $b \mapsto x(\mu)$ can change discontinuously when small bid perturbations alter which matching maximizes trust-adjusted weights, and the outer search over μ introduces additional discontinuities when it selects different bracket points (μ^-, μ^+). Such discontinuities are not inherently problematic for welfare approximation, but they are precisely what can break monotonicity in bids.

4.2 Why PTAS-style procedures typically fail monotonicity

Monotonicity is the binding constraint for DSIC. In single-parameter environments, DSIC and IR reduce to a simple condition: each advertiser's allocation probability $y_i(b)$ must be nondecreasing in b_i holding b_{-i} fixed, and payments must follow the envelope formula. Under WDP-B, $y_i(b)$ is induced by a combinatorial choice among matchings. In principle, a welfare-

maximizing rule is monotone (it is an exact maximizer of a linear objective in b_i over a bid-independent feasible set). The difficulty is that we cannot compute the exact maximizer in polynomial time in general. Once we move to approximation algorithms, monotonicity is no longer automatic: a procedure can be near-optimal yet behave erratically as a function of bids.

This issue is not an artifact of our generative-ad setting; it is the mechanism-design analogue of a familiar algorithmic phenomenon. Approximation schemes often rely on *case distinctions* (which heavy edges are guessed, which price μ is chosen, which rounded instance is solved). These case distinctions are typically functions of the entire bid profile. From the perspective of a single advertiser, raising b_i can move the instance across a case boundary and thereby reduce (or eliminate) that advertiser's allocation.

Where non-monotonicity enters: guessing and global decisions. Consider the heavy-edge enumeration template. The algorithm evaluates many candidate heavy sets H and returns the best overall feasible solution among them. Whether advertiser i is served is determined not only by the relative order of edges incident to i, but by which global heavy set wins the final comparison. A small increase in b_i can make a particular candidate solution involving i appear more attractive, but it can also change the identity of the winning heavy set in a way that *crowds out* i through matching conflicts or through budget usage.

The same tension arises in Lagrangian search. The map $\mu \mapsto x(\mu)$ is typically piecewise constant, with jumps when two matchings swap optimality under trust-adjusted weights. If the algorithm selects μ by comparing objective values that depend on bids, then increasing b_i can shift the selected μ into a region where advertiser i is no longer chosen because the algorithm now prefers a different bundle of high-density edges that exhaust the budget.

A simple illustrative failure mode. We can illustrate the logic with a minimal thought experiment. Take $K = 1$ (at most one insertion) so matching constraints are trivial, and suppose the algorithm uses a common knapsack heuristic: it compares (a) the best single edge and (b) a density-based candidate (or, more generally, it compares solutions produced by different subroutines and returns the best). Let advertiser i have an edge with moderate value and low disruption, while advertiser k has an edge with slightly higher value but higher disruption. For some bids, the algorithm may pick i because the density heuristic prefers low disruption. If i slightly increases b_i, the algorithm might switch to the "best single edge" subroutine that now picks k (because k's value is still higher), thereby *reducing* i's allocation from 1 to 0. Welfare approximation is unaffected (the chosen solution is still near-optimal), but monotonicity is violated.

This is the core difficulty: approximation algorithms frequently combine

multiple candidate solutions and then take the maximum. The max operation over candidate solutions is benign for welfare but hostile to monotonicity because it creates bid-dependent regime switches.

Why deterministic tie-breaking is not enough. One might hope that careful, deterministic tie-breaking could resolve the issue. Deterministic tie-breaking is necessary for DSIC implementability, but it is not sufficient. The problem is not only ties; it is that the identity of the candidate solution being compared can change with bids. Even with a fixed, lexicographic rule within each subroutine, the outer selection among subroutines can flip. In other words, we can have strict inequalities throughout and still obtain non-monotone outcomes.

Randomization helps, but changes the incentive notion. A natural response is to randomize: mix two matchings (for example, the two bracket matchings from Lagrangian search) so that expected disruption meets the budget and expected welfare is high. Randomization can smooth discontinuities and is compatible with Bayesian incentive compatibility (BIC) under suitable constructions. But it does not, in general, deliver DSIC, and it complicates governance and auditing in the present application: a hard trust budget is most naturally interpreted as an ex post constraint, while randomized mixing enforces it only in expectation unless one adds additional machinery.

4.3 Implications for mechanism design under hard budgets

The "algorithmic" and "incentive" frontiers do not coincide. The upshot is that Proposition 3 should be read as a computational benchmark, not as a mechanism. It tells us that, absent incentives, near-optimal welfare under a hard trust budget is achievable in polynomial time (for fixed accuracy). But DSIC introduces an additional constraint that is qualitatively different from feasibility and approximation: it is a *global shape restriction* on how the allocation changes with bids. There is no general reason to expect a PTAS to satisfy it.

This gap matters for policy and practice. In a generative advertising system, the trust budget B is precisely the object likely to be audited: regulators and internal risk teams want a hard guarantee that the aggregate disruption does not exceed a specified threshold. If the platform also wants DSIC (for transparency, simplicity, or robustness), it cannot simply take the best-performing approximation heuristic. It must commit to an allocation rule whose dependence on bids is monotone and whose tie-breaking is fixed. Otherwise, the envelope payment formula is not well defined (or yields negative transfers in some regions), and strategic manipulation becomes possible exactly at the regime boundaries created by the approximation.

What kind of structure should we look for? The natural design lesson is that we should search for approximation algorithms whose decision logic is *order-based* and stable: increasing b_i should move edges incident to i earlier in a fixed ordering, rather than changing the set of cases the algorithm considers. In knapsack settings, this is precisely where structural assumptions on costs become valuable. If disruption costs live on a small number of scales (few buckets) or have bounded spread, then the budget constraint becomes "almost" a cardinality constraint within each scale. This opens the door to greedy maximal-matching rules with deterministic tie-breaking, which are both approximately efficient and monotone.

That is the direction we take next. Section 5 builds a monotone, constant-factor allocation rule under structured disruption costs, and then uses envelope payments to obtain a DSIC and IR mechanism under the same hard trust budget.

5 A Monotone Constant-Factor Allocation Rule Under Structured Costs

The previous section explains why "near-optimal" algorithms for WDP-B are not automatically usable in a DSIC mechanism: once the procedure makes global, bid-dependent case distinctions (e.g., which heavy set to guess, which Lagrange price to bracket), it becomes easy for a bidder to increase b_i and nonetheless end up with a smaller click allocation y_i. We now show that this pathology is not inevitable. Under mild structure on disruption costs—either a constant number of distinct magnitudes after rounding, or bounded spread—we can design an allocation rule whose *decision logic is order-based* and stable, and which therefore satisfies the monotonicity condition needed for DSIC payments.

5.1 Structured costs and bucketing

We study two related assumptions, each of which limits how "knapsack-like" the budget constraint is.

Case A (few cost magnitudes). After a publicly committed rounding scheme (auditable and context-dependent), assume that each edge cost belongs to a constant-sized set

$$d_{ij} \in \{\delta_1, \dots, \delta_L\}, \qquad L = O(1),$$

with $\delta_1 < \cdots < \delta_L$. The point is not that costs are literally discrete, but that a platform can credibly commit to a coarse cost scale (for disclosure intensity, brand-safety tiers, etc.), and these tiers become the mechanism's primitives.

Case B (bounded spread). Assume $d_{\max}/d_{\min} \le c$ for a constant c. In this case we define geometric buckets: let $\delta_k = 2^k d_{\min}$ for $k = 0, 1, \ldots, \lceil \log_2 c \rceil$, and assign each edge (i, j) to the smallest k with $d_{ij} \le \delta_k$. This reduces Case B to Case A with

$$L \le 1 + \lceil \log_2 c \rceil,$$

which is constant when c is constant. The bucketing is conservative: we treat an edge in bucket k as if it costs δ_k, i.e., we upper bound disruption inside each bucket.

In either case, let $E_k = \{(i, j) : d_{ij} \text{ is in bucket } k\}$ and define a *bucket-specific cardinality cap*

$$q_k \;=\; \min\Big\{K, \Big\lfloor \frac{B}{\delta_k} \Big\rfloor\Big\}.$$

Any matching that uses at most q_k edges from E_k is automatically budget-feasible, because its total disruption is at most $q_k \delta_k \le B$ (and also respects the global insertion cap K by construction). The central simplification is that within a single bucket, the hard knapsack constraint is replaced by a *pure size constraint.*

5.2 Greedy maximal matching within a bucket

Fix bids b and define weights $w_{ij}(b) = b_i p_{ij}$. For each bucket k, we run the following deterministic greedy routine on the bipartite graph with edge set E_k.

Greedy subroutine GreedyMatch(E_k, q_k)**.** Order edges $(i, j) \in E_k$ by decreasing weight $w_{ij}(b)$, breaking ties deterministically by a fixed lexicographic rule (e.g., increasing i, then increasing j). Initialize an empty matching $M_k = \emptyset$. Scan edges in this order; whenever an edge (i, j) is encountered with both endpoints currently unmatched and $|M_k| < q_k$, add it to M_k. Output the allocation $x^{(k)}(b)$ corresponding to M_k.

This is a weighted analogue of a maximal matching algorithm. Two properties are immediate: (i) it is polynomial time, and (ii) it produces a feasible allocation in $X(B)$ because it is a matching and uses at most q_k edges from a bucket that costs at most δ_k per edge.

We also compute the best feasible single edge

$$e^\star(b) \in \arg\max\{w_{ij}(b) : d_{ij} \le B\},$$

with deterministic tie-breaking. This "single-edge" candidate is the standard knapsack safeguard: if the optimal solution relies on one very expensive but very valuable insertion, a bucket that enforces uniform per-edge costs can miss it.

Final allocation rule. Let the candidate set be

$$\mathcal{C}(b) = \{x^{(1)}(b), \ldots, x^{(L)}(b), x^{(\star)}(b)\},$$

where $x^{(\star)}(b)$ allocates only $e^\star(b)$. We output

$$x(b) \in \arg\max_{x \in \mathcal{C}(b)} \sum_{i,j} w_{ij}(b)\, x_{ij},$$

with deterministic tie-breaking over candidates (e.g., prefer $x^{(\star)}$ last, and otherwise prefer smaller k).

The salient design feature is that the only bid dependence is through a *single global ordering by* $w_{ij}(b)$ *inside each bucket*, plus a deterministic comparison of a constant number of candidate matchings. There are no guessed sets and no Lagrange-price searches whose regime changes can be hard to control.

5.3 Why the rule is monotone

We now argue that the induced click allocation

$$y_i(b) = \sum_j p_{ij} x_{ij}(b)$$

is nondecreasing in b_i for each i holding b_{-i} fixed.

Step 1: monotonicity within a bucket. Fix a bucket k and hold b_{-i} fixed. Increasing b_i scales all incident weights $\{w_{ij}(b)\}_j$ by the same factor while leaving all other weights unchanged. Therefore, in the sorted order used by GreedyMatch(E_k, q_k), every edge incident to i can only move (weakly) *earlier* relative to edges not incident to i, while the relative order among i's own edges is unchanged (since it is determined by p_{ij} and fixed tie-breaking).

Because the greedy routine accepts the first incident edge it sees whose position endpoint is free (and then never revisits advertiser i), moving i's incident edges earlier can only weakly *expand* the set of available positions at the moment i is matched. Consequently, if bidder i is matched under $x^{(k)}(b)$ to some position j, then under a higher bid $b_i' > b_i$ she is still matched in bucket k, and the resulting matched position j' satisfies

$$p_{ij'} \;\geq\; p_{ij}.$$

In particular, the bucket-level click allocation

$$y_i^{(k)}(b) = \sum_j p_{ij} x_{ij}^{(k)}(b) \in \{0\} \cup \{p_{ij} : (i,j) \in E_k\}$$

is nondecreasing in b_i.

A closely related (and operationally important) stability property also holds: if i is *not* matched by the greedy routine in bucket k, then changing b_i cannot affect which edges among other advertisers are accepted. The reason is simply that edges that are *rejected* do not change the matching state; if none of i's edges are ever accepted, the evolution of the matching among other vertices is identical. This "loser-independence" is exactly what fails in many PTAS templates, where even a losing bidder can affect which global case is selected.

Step 2: monotonicity of selecting the best candidate. Consider the overall rule that picks the best candidate in $\mathcal{C}(b)$. Suppose that at bids (b_i, b_{-i}) the winning candidate allocation $x(b)$ gives advertiser i positive click probability, i.e., $y_i(b) > 0$. Increase the bid to $b_i' > b_i$.

For any candidate allocation that does *not* allocate to i at the higher bid (i.e., yields $y_i = 0$), loser-independence implies that its total weight is unchanged when we vary b_i (since i remains unmatched throughout that subroutine, the selected edges and thus the realized weights of other bidders are identical). In contrast, for the candidate that did allocate to i at b_i, Step 1 implies that when we rerun it at b_i' it still allocates to i and with weakly larger click probability, so its total weight weakly increases.

Therefore, a candidate that excludes i cannot overtake the previously winning, i-including candidate when we raise b_i. It follows that i cannot lose allocation by bidding more:

$$b_i' > b_i \quad \Longrightarrow \quad y_i(b_i', b_{-i}) \geq y_i(b_i, b_{-i}).$$

This is precisely the single-parameter monotonicity condition required for DSIC with envelope payments.

Two remarks clarify what is doing the work. First, deterministic tie-breaking is essential: without it, equal-weight perturbations can lead to bid-dependent selection among ties, which is indistinguishable from a hidden case distinction. Second, we are not claiming that *every* greedy heuristic is monotone; rather, bucketing collapses the budget to a size constraint, and the particular maximal-matching greedy has the key loser-independence property that makes the outer "take the best" step safe.

5.4 Approximation guarantee

We finally show that the above monotone rule achieves a constant-factor approximation to the optimal welfare subject to the hard trust budget.

Within-bucket approximation. Fix a bucket k and consider the best feasible matching that uses only edges in E_k and at most q_k edges; denote its value by $\mathrm{OPT}_k(b)$. The greedy maximal matching in nonincreasing weight

order is a standard 1/2-approximation for maximum weight matching under a cardinality cap:[4]

$$\sum_{i,j} w_{ij}(b)\, x_{ij}^{(k)}(b) \;\geq\; \frac{1}{2}\,\mathrm{OPT}_k(b).$$

From buckets to a global bound. Let $\mathrm{OPT}(b;B)$ be the true optimum over $X(B)$. Partition the edges used by an optimal solution by their buckets. If Case A holds (true discrete costs), then the optimal solution's value is exactly the sum of its bucket contributions. By the pigeonhole principle, there exists a bucket $\bar{k}$ such that the value contributed by edges in bucket $\bar{k}$ is at least $\mathrm{OPT}(b;B)/L$. Feasibility of the optimal solution implies it uses at most $q_{\bar{k}}$ edges from that bucket, so this bucket-contribution is at most $\mathrm{OPT}_{\bar{k}}(b)$. Hence

$$\max_k \mathrm{OPT}_k(b) \;\geq\; \frac{1}{L}\,\mathrm{OPT}(b;B).$$

Combining with the within-bucket 1/2 bound and the fact that our final rule takes the best bucket output yields

$$\max_k \sum_{i,j} w_{ij}(b)\, x_{ij}^{(k)}(b) \;\geq\; \frac{1}{2L}\,\mathrm{OPT}(b;B).$$

The role of the best single edge. The bucket argument can be pessimistic when the optimum is dominated by a single expensive edge (a common knapsack corner case). Including $x^{(\star)}(b)$ ensures we do not lose more than a constant factor in that regime. In particular, if an optimal solution derives at least half its value from its highest-value edge, then $x^{(\star)}(b)$ alone achieves at least $\frac{1}{2}\,\mathrm{OPT}(b;B)$. Taking the best of the bucket matchings and $x^{(\star)}(b)$ therefore yields an overall approximation factor

$$\alpha = O(L) \quad \text{in Case A,}$$

and under bounded spread,

$$\alpha = O(\log c) \quad \text{in Case B,}$$

which is a constant when c is a fixed structural parameter of the application.

[4]The argument is the usual charging proof: each edge in an optimal matching conflicts with at most two edges in the greedy matching (one per endpoint), and the greedy edge encountered first has weight at least that of the conflicting optimal edge. Summing over conflicts yields $W(\mathrm{Greedy}) \geq \frac{1}{2}W(\mathrm{OPT})$.

Interpretation and limitation. The economic content of the approximation is straightforward: when disruption costs live on only a few scales, the platform can restrict attention to "homogeneous-cost" insertion plans, where the trust budget behaves like an insertion count constraint. Within each scale, a simple greedy matching already captures a constant fraction of the best feasible plan, and the entire procedure can be made monotone. The limitation is equally clear: if costs span many orders of magnitude (large L or large c), then the knapsack aspect is genuinely multi-scale and we should not expect a constant-factor monotone rule without sacrificing either welfare or generality. The next section shows how to convert this monotone allocation rule into a DSIC mechanism via envelope payments, and how the same logic extends to revenue objectives via virtual values under standard regularity conditions.

6 Mechanism Design: Payments for Welfare and Revenue Variants

Having constructed a deterministic allocation rule A that is feasible and monotone in each reported bid, we can now complete the mechanism by specifying transfers. In our setting each advertiser i has a single private parameter v_i (value per click), and the outcome relevant for incentives is the *total click probability* assigned to i,

$$y_i(b) \;=\; \pi_i(x(b)) \;=\; \sum_{j=1}^{m} p_{ij}\, x_{ij}(b) \in [0,1],$$

where $x(b) = A(b) \in X(B)$ is the (budget-feasible) matching computed from bids b. Because A is monotone, $y_i(b_i, b_{-i})$ is nondecreasing in b_i for each fixed b_{-i}. This is the only substantive requirement for dominant-strategy truthfulness in single-parameter environments; the budget constraint and matching structure affect feasibility, but do not alter the incentive characterization.

6.1 Envelope payments for welfare maximization

We consider the direct-revelation implementation in which bidders report b_i (truthfully $b_i = v_i$ under DSIC) and the platform runs $x(b) = A(b)$. The standard envelope theorem implies that *any* DSIC payment rule must satisfy, up to a constant chosen by normalization,

$$t_i(b) \;=\; t_i(0, b_{-i}) \;+\; b_i\, y_i(b) \;-\; \int_0^{b_i} y_i(z, b_{-i})\, dz. \tag{5}$$

Imposing individual-rationality normalization $t_i(0, b_{-i}) = 0$ yields ex post IR (since $u_i(v) = v_i y_i(v) - t_i(v) = \int_0^{v_i} y_i(z, v_{-i}) dz \geq 0$) and pins down payments

uniquely for our deterministic rule. We emphasize an interpretation that is operationally useful: $t_i(b)$ is an *expected* transfer per impression. If the platform prefers charging per click, it can equivalently levy a per-click price

$$\mathrm{cpc}_i(b) \;=\; \frac{t_i(b)}{y_i(b)} \quad \text{when } y_i(b) > 0,$$

so that the expected payment equals $\mathrm{cpc}_i(b) \cdot y_i(b)$. (When $y_i(b) = 0$, we set $t_i(b) = 0$.)

Two points are worth making explicit. First, the payment depends on the entire allocation curve $z \mapsto y_i(z, b_{-i})$, not only on the realized $y_i(b)$. This matters because our allocation may assign bidder i to different positions at different bids, so y_i can increase in steps that correspond to jumps among distinct p_{ij} values. Second, the budget constraint does not enter (5) directly; it affects transfers only through its effect on the allocation curve. In particular, "paying for disruption" is not required for truthfulness: we enforce trust via the feasibility constraint $x(b) \in X(B)$, not via bidder-facing prices.[5]

Given monotonicity, DSIC follows in the usual way: for fixed b_{-i}, advertiser i faces a one-dimensional choice. The envelope formula ensures that truthful reporting maximizes $v_i y_i(\hat{b}_i, b_{-i}) - t_i(\hat{b}_i, b_{-i})$ over $\hat{b}_i$, and the monotonicity of y_i ensures that the induced utility is the integral of the allocation curve up to v_i. Thus our approximation guarantee from the allocation rule transfers verbatim to a truthful welfare mechanism: when all advertisers report truthfully, the realized welfare is within the same constant factor α of the optimal welfare subject to the trust budget.

6.2 Computing payments: exact thresholds versus numerical integration

While (5) is conceptually clean, implementing it requires computing the integral of a monotone, piecewise-constant function. In principle, for a deterministic allocation algorithm A, $y_i(z, b_{-i})$ changes only when varying z changes the relative order of some i-incident edge (i, j) (whose weight is $z\, p_{ij}$) against an edge not incident to i (whose weight is fixed at $b_{i'} p_{i'j'}$), or when it changes which candidate among a constant set (our buckets plus the best single edge) is selected. Therefore, for fixed b_{-i}, there exists a finite set of *critical bids* at which y_i can jump.

One can compute $t_i(b)$ exactly by enumerating these breakpoints. Concretely, fix advertiser i and consider a given bucket k. In that bucket, the greedy routine sorts edges by weights $w_{ij}(b) = b_i p_{ij}$. Holding b_{-i} fixed, the only comparisons that depend on b_i are of the form

$$b_i\, p_{ij} \;\geq\; b_{i'}\, p_{i'j'} \qquad (i' \neq i).$$

[5] In applications where d_{ij} also represents an advertiser-specific compliance burden, one can add explicit terms to utilities; our analysis isolates the case where d_{ij} is a platform/user-side constraint.

Thus, for each $(i,j) \in E_k$ and each competitor edge $(i',j') \in E_k$ with $p_{ij} > 0$, there is a threshold

$$\tau \;=\; \frac{b_{i'} p_{i'j'}}{p_{ij}}$$

at which the order between those two edges flips. Between successive thresholds, the relative order of all edges is fixed, and hence the greedy scan (with deterministic tie-breaking) produces a fixed matching, implying a constant $y_i^{(k)}(b_i, b_{-i})$. The outer step that selects the best candidate among $L+1 = O(1)$ matchings can only introduce additional (but still finite) breakpoints coming from comparisons of total weights of candidate matchings as functions of b_i. Since those weights are affine in b_i within each region (they equal a fixed constant plus b_i times the realized click probability of i in that candidate), we can also locate candidate-switch thresholds exactly.

This "exact critical-bid" approach is polynomial-time but can be heavy in the worst case, because the number of raw pairwise thresholds τ scales with the number of edges. In practice, we can take a simpler view: we only need to evaluate the integral in (5), not to explicitly describe all discontinuities. A standard black-box approach is to approximate the integral numerically by querying the allocation rule at a grid of bids. Let $\Delta > 0$ be a step size and let $z_\ell = \ell\Delta$ for $\ell = 0, 1, \ldots, \lfloor b_i/\Delta \rfloor$. Define the Riemann-sum approximation

$$\widehat{t}_i(b) \;=\; b_i y_i(b) \;-\; \Delta \sum_{\ell=0}^{\lfloor b_i/\Delta \rfloor - 1} y_i(z_\ell, b_{-i}).$$

Because $0 \le y_i \le 1$, the absolute integration error is at most Δ, and hence the induced deviation from exact DSIC is small: bidder i can gain at most $O(\Delta)$ in utility by misreporting.[6] This yields an *ε-IC* mechanism (dominant strategies up to additive ε) that is often sufficient when values are large relative to the discretization unit and when regulators accept auditable numerical procedures.

A closely related implementation is to discretize the bid space itself (e.g., bids in cents) and run the mechanism on the discrete grid. On a finite grid, one can compute exact discrete envelope payments by summation, obtaining exact DSIC *on the grid* and ε-IC relative to the underlying continuum. We flag this explicitly because in policy-constrained environments the payment computation is part of what must be explainable and verifiable: discretization makes both the allocation (via deterministic tie-breaking) and the payment rule mechanically auditable.

[6] More precisely, if we hold b_{-i} fixed and use the approximate payment rule $\widehat{t}_i$, then the difference between truthful utility and best-response utility is bounded by the maximal integration error of the Riemann sum, which is at most Δ since $y_i \in [0,1]$. This yields an ε-IC guarantee with $\varepsilon = \Delta$.

6.3 Revenue objective: virtual values under regularity

The same mechanism-design logic extends to revenue once we invoke Myerson's transformation. Suppose each v_i is drawn independently from a known distribution F_i with density f_i, and define the virtual value

$$\phi_i(v) \;=\; v - \frac{1 - F_i(v)}{f_i(v)}.$$

Myerson's lemma states that for any DSIC and IR mechanism, expected revenue equals expected *virtual surplus*,

$$\mathbb{E}\Big[\sum_i t_i(v)\Big] \;=\; \mathbb{E}\Big[\sum_i \phi_i(v_i)\, y_i(v)\Big],$$

up to standard boundary conditions satisfied by our normalization. Therefore, a natural revenue-oriented variant of our mechanism is: replace each bid b_i used in weights by the corresponding virtual value $\phi_i(b_i)$, run the same monotone allocation template on weights

$$w^{\phi}_{ij}(b) \;=\; \phi_i(b_i)\, p_{ij},$$

and then compute payments via the envelope formula with respect to the allocation curve $y^{\phi}_i(b)$ induced by this virtual-weighted rule.

This construction relies on a regularity condition. If F_i is *regular* so that $\phi_i(\cdot)$ is nondecreasing, then monotonicity of the allocation in the "virtual bid" implies monotonicity in the true bid: increasing b_i weakly increases $\phi_i(b_i)$, which (by the same order-based argument as before) cannot decrease y^{ϕ}_i. Under regularity, we thus obtain a DSIC and IR mechanism whose expected revenue is within the same approximation factor α of the optimal *constrained* revenue benchmark (the optimum virtual surplus subject to $X(B)$).

Two practical refinements are standard and carry over directly. First, since negative virtual values reduce virtual surplus, one typically imposes a reserve by truncating ϕ_i at zero (or, equivalently, refusing to allocate to advertisers whose virtual values are negative), which preserves IR and improves revenue. Second, if distributions are not regular, one can iron ϕ_i to a monotone virtual value $\bar{\phi}_i$; the allocation remains monotone in b_i and the revenue guarantee applies with respect to the ironed benchmark.

6.4 What the mechanism does (and does not) claim

It is tempting to read the above as saying that trust constraints are "just another feasibility constraint." This is true for incentives in the narrow single-parameter sense: DSIC depends on monotonicity of y_i and the envelope payments, and the budget only shapes which outcomes are feasible. However, two limitations are worth keeping in view.

First, our revenue result is an approximation to an *information-theoretic* optimum that itself respects the trust budget. If B is tight, the revenue-maximizing truthful mechanism may rationally leave high-paying ads unserved to preserve trust; this is a feature, not a bug, but it means revenue comparisons must always condition on the compliance regime. Second, implementing virtual values presumes either known distributions or a defensible empirical estimation procedure. In many ad markets, distribution shift across contexts is material; here the transparent option is to treat the welfare mechanism as the robust baseline and to interpret revenue optimization as a secondary layer whose assumptions (regularity, stationarity, sample size) can be audited.

These remarks set up the next step in the analysis. Having specified a truthful mechanism, we can now ask how welfare changes as the trust budget B varies, how to interpret the dual shadow price μ as a "trust tax," and how to derive auditable welfare-loss bounds from tightening compliance constraints.

7 Price of Trust: Comparative Statics in B and Dual-Based Welfare Bounds

We now treat the trust/compliance budget B not as a fixed engineering parameter, but as a policy-relevant lever. The central object is the *constrained welfare value function*

$$W^*(B) \;=\; \max_{x \in X(B)} \sum_{i=1}^{n} \sum_{j=1}^{m} v_i p_{ij} x_{ij},$$

and its algorithmic counterpart (from our monotone approximation rule) which we denote by $W^A(B)$ for the welfare achieved by $x = A(v)$ at budget B. The economic question is: how much welfare is "purchased" by relaxing B, and how can that marginal tradeoff be stated in an auditable way?

7.1 Monotonicity, saturation, and the role of K

The most robust comparative static is immediate: the feasible set expands in B, so

$$B' \geq B \quad \Rightarrow \quad X(B) \subseteq X(B') \quad \Rightarrow \quad W^*(B) \leq W^*(B').$$

However, the shape of $W^*(B)$ is not smooth in general because the underlying problem is integer. Indeed, $W^*(B)$ is typically a *step function*: as B increases, additional matchings become feasible only when B crosses disruption totals $\sum_{i,j} d_{ij} x_{ij}$ of candidate allocations.

A second operational point is *saturation* induced by the cardinality cap K. Let x^{∞} denote an optimal allocation in the unconstrained problem with budget constraint removed (but still respecting matching and K). If B is large enough that $x^{\infty} \in X(B)$, then the trust constraint ceases to bind and further increases in B have no effect:

$$\exists B_0 : \; x^{\infty} \in X(B_0) \quad \Rightarrow \quad W^*(B) = W^*(B_0) \;\; \forall B \geq B_0.$$

In particular, when K is small, saturation can occur at relatively modest B because at most K edges can be chosen. This matters for governance: a regulator may be tempted to infer that "more budget always buys more value," but if K is binding then marginal returns are literally zero beyond the saturation point.

7.2 LP relaxation, concavity, and the shadow price μ

Because the integer objective is discontinuous in B, it is useful to introduce the LP relaxation value function

$$W^{LP}(B) \;=\; \max_{x \in X^{LP}(B)} \sum_{i,j} v_i p_{ij} x_{ij},$$

where $X^{LP}(B)$ relaxes integrality to $x_{ij} \in [0,1]$ while keeping the same matching, cap, and budget constraints. This relaxation has two advantages that are directly interpretable:

1. $W^{LP}(B)$ is concave and nondecreasing in B, so it supports well-defined marginal values.

2. The dual variable on the budget constraint yields a *shadow price of trust* that can be used both as an economic summary and as an audit certificate.

Formally, the Lagrangian of the relaxed problem separates the budget constraint via a multiplier $\mu \geq 0$:

$$\mathcal{L}(x, \mu) \;=\; \sum_{i,j} v_i p_{ij} x_{ij} \;+\; \mu\Big(B - \sum_{i,j} d_{ij} x_{ij}\Big).$$

For fixed μ, maximizing $\mathcal{L}(x, \mu)$ over matchings (and the cap K) is equivalent to maximizing a *penalized weight* on each edge:

$$(v_i p_{ij}) - \mu d_{ij}.$$

Thus, at the level of first principles, the shadow price converts the hard trust budget into a per-unit "tax" on disruption, and the platform behaves as if each insertion pays a penalty proportional to its predicted disruption. In

this sense, μ plays the role of a *trust tax rate*: a higher μ makes disruptive edges less attractive even if their click value is high.

Let $\mu^*(B)$ be an optimal dual multiplier for the LP at budget B. Standard LP sensitivity results imply the subgradient inequality: for any $B' \geq 0$,

$$W^{LP}(B') \;\leq\; W^{LP}(B) + \mu^*(B)\,(B' - B). \tag{6}$$

Concavity means that $\mu^*(B)$ is a *marginal welfare* proxy: it upper bounds the welfare gain from relaxing the budget by one unit (locally, and exactly as a subgradient).

Two limiting cases are economically informative:

- If the budget is slack at the LP optimum, then complementary slackness yields $\mu^*(B) = 0$. Interpreting (6), additional trust budget has no value at the margin (consistent with saturation).
- If the budget is tight, then $\mu^*(B) > 0$ and the platform is willing to "pay" (in forgone click value) up to $\mu^*(B)$ per unit disruption to stay within compliance.

7.3 Dual-based welfare loss bounds from tightening B

A regulator often cares about the welfare cost of tightening compliance rules, i.e., moving from B to a smaller $B^- < B$. The concavity bound (6) yields an immediate and auditable upper bound on the LP welfare loss:

$$W^{LP}(B) - W^{LP}(B^-) \;\leq\; \mu^*(B)\,(B - B^-). \tag{7}$$

The right-hand side has a transparent interpretation: "the marginal price of trust at the current regime" times "the tightening magnitude." This is precisely the sense in which μ operationalizes a *price of trust*.

While (7) is stated for the LP relaxation, it is still practically valuable for the original integer problem for two reasons. First, $W^{LP}(B)$ is an *upper bound* on $W^*(B)$, so comparing two LP values gives a conservative (over-)estimate of what is achievable, hence a conservative (over-)estimate of marginal returns. Second, the dual multiplier can be computed alongside the relaxed solution and logged as part of a compliance record: an auditor can verify that the reported $\mu^*(B)$ indeed corresponds to a feasible dual solution and hence certifies the inequality.

We can also write (7) in a form that emphasizes counterfactual explainability. Suppose a policy proposal reduces the budget by $\Delta B > 0$. Then

$$\text{LP welfare loss} \;\leq\; \mu^*(B)\,\Delta B.$$

If, for example, the platform reports that $\mu^*(B) = 0.02$ welfare-units per disruption-unit, then a tightening of $\Delta B = 10$ implies an LP loss bound of at most 0.2 welfare-units per impression. This style of statement is coarse, but it is *auditable*: it depends on a single scalar and a verifiable dual feasibility condition, not on proprietary details of the full allocation.

7.4 From shadow prices to implementable "trust taxes"

Although our mechanism enforces trust through a hard constraint, it is useful to observe that the shadow price also induces a *soft-constraint* proxy problem:

$$\max_{x \in X} \sum_{i,j} \Big(v_i p_{ij} - \mu d_{ij} \Big) x_{ij}.$$

For fixed μ, this is a standard maximum-weight matching with a cap K (and no knapsack constraint). Hence, a platform can compute a family of candidate allocations by sweeping μ and then select the one that best respects the desired budget. This leads to a practical calibration procedure: use bisection on μ to find a penalization level at which the induced allocation consumes disruption close to B.

Economically, this is exactly the construction of a Pigouvian tax: if a policymaker (or an internal trust team) has an external cost of disruption equal to λ welfare-units per disruption-unit, then setting $\mu = \lambda$ makes the platform internalize that cost in its allocation logic. Put differently, the hard budget B and the shadow price μ are two ways of describing the same underlying tradeoff: B specifies a *quantity* constraint, while μ summarizes the *marginal value* (or marginal social cost) at the optimum.

This perspective also clarifies what we mean by *compliance rent*. If the platform were allowed to marginally increase B, it could gain welfare at rate approximately $\mu^*(B)$. Therefore, access to additional compliance capacity—better disclosure UI, safer rendering, improved user controls that reduce d_{ij}—is economically valuable, and that value is priced by $\mu^*(B)$. In organizations, this often manifests as an internal transfer: teams that reduce disruption effectively create budget capacity, whose marginal benefit is measured by the current shadow price.

7.5 Scaling costs and invariances

The shadow-price interpretation also makes comparative statics under cost scaling nearly mechanical. If disruption scores are uniformly scaled by $\lambda > 0$, i.e., $d'_{ij} = \lambda d_{ij}$, then the feasibility condition $\sum d'_{ij} x_{ij} \le B$ is equivalent to $\sum d_{ij} x_{ij} \le B/\lambda$. Thus,

$$W^*_{\text{scaled}}(B) \;=\; W^*\Big(\frac{B}{\lambda}\Big),$$

and in the LP relaxation the shadow price rescales inversely:

$$\mu^*_{\text{scaled}}(B) \;=\; \frac{1}{\lambda}\,\mu^*\Big(\frac{B}{\lambda}\Big),$$

consistent with the idea that μ is denominated in "welfare per disruption unit." This is practically relevant because measurement pipelines for d_{ij}

often change (e.g., new auditors, new definitions of disclosure footprint). The above invariance tells us how to translate shadow prices and budget policies across revisions, at least under uniform rescaling.

7.6 Relating the approximation algorithm to the price of trust

Our monotone allocation rule A is designed to be truthful and approximately welfare-maximizing under structural assumptions on d_{ij}. It is not an LP solver, so it does not directly produce an exact dual multiplier. Nevertheless, the dual perspective remains useful in three ways.

First, we can use $W^{LP}(B)$ as a benchmark to quantify the welfare gap:

$$\frac{W^A(B)}{W^{LP}(B)} \;\le\; \frac{W^A(B)}{W^*(B)} \;\le\; 1,$$

so a reported LP upper bound immediately yields a conservative performance certificate for the implemented allocation.

Second, the Lagrangian weights $(v_i p_{ij} - \mu d_{ij})$ motivate a simple, auditable explanation of what the algorithm is doing when B is tight: it prioritizes edges that are "high value per unit disruption," even though (by design) it enforces the constraint hard rather than via prices. When we bucket costs (few magnitudes or bounded spread), we are in effect discretizing the disruption axis, which makes the knapsack structure closer to a cardinality constraint within each bucket; this is precisely the regime in which a single scalar μ is a good summary of the marginal tradeoff.

Third, dual bounds yield a principled way to communicate the consequences of policy changes even when the implemented rule is approximate. If tightening B by ΔB produces a change in realized welfare ΔW^A, then comparing ΔW^A to $\mu^*(B)\Delta B$ distinguishes two cases: either the change is within the worst-case LP slope bound (suggesting the observed loss is consistent with marginal scarcity of trust budget), or it exceeds that bound (suggesting that the loss is driven by integrality, approximation, or mismeasurement of d_{ij}). This diagnostic is helpful precisely because it is not tied to any particular allocation heuristic.

7.7 A policy reading: choosing B by equating marginal values

Finally, the "price of trust" language supports a clean normative guideline. Suppose an external stakeholder assigns a social cost s to disruption units (in the same welfare units as $v_i p_{ij}$). Then an economically coherent target is to choose B such that the shadow price satisfies

$$\mu^*(B) \approx s,$$

i.e., the marginal welfare gain from relaxing compliance equals the marginal social cost of doing so. When $\mu^*(B) \gg s$, compliance is too tight relative to its cost; when $\mu^*(B) \ll s$, compliance is too loose. We do not claim this pins down the unique "right" budget—the mapping from disruption metrics d_{ij} to social harm is itself contestable—but it does give a disciplined way to express disagreements. Competing stakeholders may argue about the appropriate s or about the measurement of d_{ij}, yet conditional on those primitives the shadow price provides a single, interpretable statistic that summarizes the tradeoff and yields testable comparative statics.

This completes the bridge from mechanism design to policy analysis: once the allocation and payments are incentive-compatible, the remaining question is how the feasible set should be chosen. The dual variable μ lets us describe that choice as a transparent "trust tax" and bound the welfare cost of tightening compliance in a way that is compact enough to be audited and communicated.

8 Extensions and Discussion

We have deliberately analyzed a *base-click* environment in which click probabilities are additive across positions and independent of what else is shown. That abstraction makes the welfare problem a budgeted matching and lets us cleanly separate (i) computational structure (matching plus a single knapsack) from (ii) incentive structure (single-parameter monotonicity). In practice, however, insertions interact through attention, substitution, and disclosure frictions; compliance itself is multi-faceted; and governance depends on what an auditor can actually verify. We sketch three extensions that preserve the spirit of the model—a transparent welfare–trust tradeoff under DSIC constraints—while clarifying where new technical issues arise.

8.1 Beyond additive clicks: cascade and MNL substitution

A first extension relaxes the assumption that $\pi_i(x) = \sum_j p_{ij} x_{ij}$, allowing the realized click probability to depend on the *set and order* of insertions. Two canonical families are cascade models and multinomial logit (MNL) substitution models.

Cascade/position externalities. Suppose positions are ordered $j = 1, \ldots, m$ and users scan from top to bottom, stopping with some probability after each insertion. A simple cascade specification is

$$\pi_i(x) \;=\; \sum_{j=1}^{m} x_{ij}\, p_{ij} \prod_{\ell<j} \Big(1 - \sum_{k=1}^{n} x_{k\ell}\, s_{k\ell}\Big),$$

where $s_{k\ell} \in [0,1]$ is the probability that insertion (k,ℓ) ends the session (or absorbs attention) conditional on being encountered. In such a model, adding an ad earlier can *reduce* the click probability of all later insertions. The trust budget B then interacts with welfare in a qualitatively different way: tightening B does not merely remove some edges, it can also reallocate attention by shifting ads later (or removing early ads) and thereby *increase* the effectiveness of remaining ads.

From the optimization perspective, the welfare objective $\sum_i v_i \pi_i(x)$ is no longer linear in x. Even if we keep the matching and cap constraints, the problem becomes a non-linear combinatorial optimization over ordered matchings. Depending on the cascade form, the induced set function can be approximately submodular, suggesting greedy-style approximation under matroid and knapsack constraints; but the approximation algorithms that are most natural (e.g., adaptive greedy, local search) typically do not come with monotonicity guarantees in bids. This is exactly where incentive constraints bite: even when we can compute a near-optimal allocation under cascade effects, small increases in b_i can change the chosen set in non-monotone ways because the algorithm trades off an advertiser's value against its *attention externality* on others.

One pragmatic approach is to separate modeling layers: treat p_{ij} as already incorporating expected displacement from an *exogenous* attention model (estimated under the platform's typical policy), and reserve explicit externality modeling for counterfactual evaluation rather than allocation. The limitation is clear: when the platform materially changes B or K, the externality environment changes, and the "standalone" p_{ij} cease to be stable primitives. This motivates the learning-robust direction flagged in our open problems, but it also motivates governance: if the platform uses a base-click allocation rule, it should be explicit that p_{ij} are policy-dependent predictions.

MNL substitution. A second workhorse model is MNL, in which the user chooses among displayed options (including an outside option) with probabilities proportional to latent attractiveness parameters. One stylized version for an insertion set S (where S indexes chosen advertiser-position pairs) is

$$\Pr[\text{click } (i,j) \mid S] \;=\; \frac{\alpha_{ij}}{1+\sum_{(k,\ell)\in S}\alpha_{k\ell}}, \qquad (i,j)\in S,$$

so welfare equals

$$W(S;v) \;=\; \sum_{(i,j)\in S} v_i \cdot \frac{\alpha_{ij}}{1+\sum_{(k,\ell)\in S}\alpha_{k\ell}}.$$

Here, adding any insertion mechanically reduces the click probability of all others via the shared denominator. This turns the trust budget into an

even more economically meaningful lever: a larger B can permit more insertions, but those additional insertions may cannibalize attention and lower the marginal value of further expansion. Thus, even though the optimal value as a function of B is still weakly increasing (the feasible set expands), the *composition* effects are sharper: optimal policy may prefer a smaller, less disruptive set even when B is large because attention becomes the binding resource rather than trust.

Algorithmically, MNL welfare is neither linear nor obviously submodular in general when weights v_i vary, and the matching structure (at most one per advertiser and per position) couples decisions in a way that blocks simple reductions. A common tactic is to optimize a surrogate such as $\sum_{(i,j)\in S} v_i \alpha_{ij}$ subject to an additional constraint controlling total attractiveness $\sum_{(i,j)\in S} \alpha_{ij}$, which resembles a second budget. This directly foreshadows the *multiple budgets* extension below: in substitution models, it is natural to interpret attention as another scarce resource, and then trust is one constraint among several.

Incentives with externalities. With either cascade or MNL, the single-parameter DSIC characterization (monotone allocation plus envelope payments) still applies *formally* if we define $y_i(b)$ as advertiser i's total click probability under the chosen allocation. The difficulty is constructive: we must exhibit an allocation rule that is both (approximately) welfare-maximizing under externalities and monotone in each b_i. In our view, this is not merely a technicality: it is the economic tension between *correcting externalities* (which often requires non-myopic tradeoffs across agents) and *incentive compatibility* (which restricts how sharply allocations can react to bids). A policy implication is that, when externalities are large, one should expect either weaker welfare guarantees under DSIC or the need for richer type spaces (e.g., allowing advertisers to report nuisance parameters that affect externalities), which in turn raises verification issues.

8.2 Multiple budgets: disclosure, brand safety, and other compliance dimensions

A second extension replaces the single scalar disruption budget with multiple hard constraints. For example, regulators and platforms often separate (i) *disclosure footprint* (how intrusive or frequent sponsored labels are) from (ii) *brand-safety or suitability risk* (the probability of adjacency to sensitive content), and sometimes from (iii) *user-experience* metrics (expected dissatisfaction, bounce risk). A minimal formalization introduces R resource dimensions with costs $d_{ij}^{(r)} \geq 0$ and budgets $B^{(r)}$:

$$\sum_{i,j} d_{ij}^{(r)} x_{ij} \;\leq\; B^{(r)}, \qquad r = 1, \ldots, R.$$

The relaxed Lagrangian now uses a vector of multipliers $\mu \in \mathbb{R}_+^R$ and induces penalized edge weights

$$v_i p_{ij} \; - \; \sum_{r=1}^{R} \mu_r d_{ij}^{(r)}.$$

Economically, this yields a *menu of shadow prices*: a "price of disclosure," a "price of safety," etc. The governance appeal is immediate: different internal stakeholders can negotiate in terms of marginal rates (shadow prices) rather than opaque global rules, and an auditor can ask for a certificate that each constraint is satisfied together with the implied marginal values.

Computationally, however, multiple budgets move us from a single knapsack to a *multi-knapsack* constraint intersected with matching. Even in the absence of incentive constraints, approximation factors typically deteriorate with R (and the dependence can be exponential without further structure). Our structured-cost approach suggests one path forward: if each $d_{ij}^{(r)}$ takes only $O(1)$ rounded magnitudes (or bounded spread) *in each dimension*, then we can bucket edges by a *cost vector class*. Within any fixed class, each constraint behaves like a cardinality cap, so greedy matching remains a natural primitive. The caveat is combinatorial explosion: the number of classes can scale as L^R, so even constant L can become large when R grows.

This tradeoff has a practical interpretation. Platforms often do not treat all constraints symmetrically: one constraint (say, legal disclosure) is enforced as a hard budget, while others (say, brand safety) are implemented as conservative filtering or as a penalty in the objective. In our language, that corresponds to keeping one $B^{(r)}$ hard and moving others into the Lagrangian with fixed μ_r chosen by policy. This hybrid is not "fully optimal" in a multi-constraint sense, but it is transparent, tunable, and more likely to admit monotone allocation rules (because it reduces the number of hard combinatorial couplings that can cause non-monotone threshold effects).

A final nuance is that multiple budgets complicate *auditable counterfactuals*. With one budget, a regulator can ask "what is the welfare loss of tightening B by ΔB?" With R budgets, the relevant counterfactual is a vector perturbation, and the platform must clarify which constraint is binding and which is slack. This pushes reporting toward a multi-dimensional "trust dashboard" consisting of $(B^{(r)}, \mu_r)$ pairs and realized consumption, rather than a single scalar.

8.3 Auditing and governance: what can be verified, and what can be gamed

A third extension is not a change to preferences or feasibility, but to the institutional environment: who observes and verifies (p_{ij}, d_{ij}, B), and what commitments are credible.

Auditability of decisions. Our mechanism is only as interpretable as the objects an auditor can reconstruct. In the base-click model, a useful compliance record for each context c can be remarkably compact: the declared budgets, the realized allocation x, the realized disruption $\sum_{i,j} d_{ij}x_{ij}$, and (when computed) a dual-feasible certificate for the LP relaxation. When the allocation algorithm is greedy-with-tie-breaking (as in our structured-cost regime), auditability further requires logging the deterministic tie-breaking rule (or, if randomized, the random seed) so that the platform cannot ex post rationalize a different allocation under the same bids.

Measurement governance for d_{ij}. The deepest governance question is not whether the platform satisfied $\sum d_{ij}x_{ij} \leq B$ given the reported d_{ij}, but whether the d_{ij} themselves are measured in a stable and non-manipulable way. Two failure modes are especially salient. First, *policy drift*: if the pipeline producing d_{ij} changes over time (new classifiers, new definitions of "harm"), then budgets become incomparable across periods unless the platform publishes a translation (e.g., a rescaling or re-bucketing) and revalidates historical compliance. Second, *strategic content*: advertisers may alter creatives or landing pages to reduce predicted disruption scores without reducing true harm. This is not a standard bid manipulation and is not addressed by DSIC. It is closer to adversarial robustness, and it suggests that compliance metrics should be (i) hard to spoof, (ii) periodically audited with human review, and (iii) accompanied by penalties for misrepresentation that operate outside the auction (e.g., account sanctions).

Commitment and credible constraints. A hard budget B is meaningful only if it is credibly binding. Internally, that requires organizational separation: the team setting B (or certifying d_{ij}) should not be the same team optimizing revenue. Externally, it requires that a regulator can observe either (a) the realized consumption of disruption units or (b) enough aggregated statistics to test whether the platform is systematically exceeding the budget. In many deployments, the platform will be reluctant to reveal edge-level (p_{ij}, d_{ij}) because of proprietary models. This is where dual certificates and aggregation become valuable: a regulator may not need the full matrix to verify that a reported allocation is within budget, provided the platform commits to a measurement standard and exposes sufficient logs for sampling-based audits.

Fairness and disparate impact. A subtle governance implication of a trust budget is that it can create disparate impacts across advertisers or user groups. If d_{ij} is higher in certain contexts (e.g., sensitive topics) or for certain creatives (e.g., political content requiring stronger disclosures), then the budget couples these segments through a global constraint. Even

a welfare-maximizing mechanism may systematically exclude high-d_{ij} segments when B is tight. From a policy standpoint, this may be desirable (it is exactly what a safety budget is meant to do), but it should be explicit: a single global B implicitly defines a *rationing rule* across categories. One governance response is to introduce category-specific budgets (a special case of multiple budgets) or minimum-serve constraints; another is to publish category-level consumption and shadow prices so that stakeholders can see where rationing occurs.

Limits of the mechanism-design lens. Finally, we should be clear about what our mechanism does *not* solve. DSIC aligns bids with values given the allocation rule, but it does not ensure that the platform's estimates p_{ij} and d_{ij} are correct, stable, or welfare-relevant in the social sense. Nor does it resolve normative disagreements about what counts as "disruption." What the model contributes is a disciplined interface: once primitives are fixed and auditable, we can articulate how welfare, trust budgets, and incentives trade off, and we can identify which empirical objects (prediction quality, cost measurement, binding constraints) are driving outcomes. In our view, that interface is exactly what makes the framework useful for governance: it narrows debates from vague arguments about "too many ads" to concrete, testable claims about budgets, costs, and marginal values.

9 Conclusion and Open Problems

We have studied a simple but, we believe, operationally meaningful interface between sponsored insertions and governance constraints: advertisers have single-parameter values per click, the platform predicts a click gain p_{ij} and a disruption cost d_{ij} for each potential insertion, and a hard budget B limits total disruption. The base-click assumption makes the welfare objective linear, so the constrained allocation problem becomes a matching with a single knapsack constraint. The core economic point is that, once we can implement a monotone allocation rule under this feasibility set, DSIC and IR follow from standard envelope payments, and the budget constraint admits a transparent "price of trust" interpretation via a Lagrange multiplier.

The main limitations of our positive results are also the natural boundary of the framework: they rely on (i) a stylized click model, (ii) a single-period environment with fixed and correctly measured (p_{ij}, d_{ij}), and (iii) structured disruption costs (few magnitudes or bounded spread) to obtain monotone approximation in polynomial time. We close by describing three research directions that would make the framework both more general and more realistic, while highlighting where new technical obstacles appear.

9.1 Removing structured-cost assumptions: monotonicity versus approximation

Our monotone constant-factor allocation relies on a cost structure that makes the knapsack aspect "almost" a cardinality constraint within buckets. Without such structure, the welfare problem is computationally harder even before we impose incentives, and the incentive requirement interacts with hardness in a particularly sharp way.

The algorithmic bottleneck. In the general case, d_{ij} can vary widely, and the feasible set $X(B)$ is the intersection of a matching polytope with an arbitrary knapsack. Even ignoring integrality, the LP relaxation may have a nontrivial integrality gap; with integrality, the problem inherits knapsack-style combinatorial hardness. Standard approximation schemes for budgeted matching (e.g., guessing heavy items, local search, or randomized rounding) typically break monotonicity: a small increase in b_i can change which "heavy" edges are guessed, or which local move is accepted, causing advertiser i's allocation probability to decrease.

This suggests an open problem that is more structural than it may first appear: *characterize when knapsack–matching admits monotone approximation.* In single-parameter domains, monotonicity is not merely a design preference; it is the implementability constraint for DSIC. Thus, even if we can approximate the welfare optimum within $(1 - \varepsilon)$, that guarantee may be irrelevant if the induced allocation rule cannot be turned into a truthful mechanism.

Possible ways forward. We see at least three plausible paths, each with its own costs.

First, one can relax determinism and seek *monotone-in-expectation* randomized mechanisms. Randomization enlarges the design space substantially: rather than insisting that $x_{ij} \in \{0, 1\}$ be a deterministic matching, one can output a distribution over feasible matchings and ensure that $y_i(b)$ is nondecreasing in expectation. However, randomness complicates auditability and governance unless the platform logs the random seed and provides an ex post verifiable description of the distribution. Moreover, in knapsack-like domains, even monotonicity in expectation can be delicate: a distributional change that maintains feasibility and approximation may still be non-monotone in b_i.

Second, one can accept weaker welfare guarantees but insist on monotonicity by design, aiming for *simple posted-price or threshold mechanisms* driven by the shadow price μ. For example, fix a penalty $\mu \geq 0$ and maximize $\sum_{i,j}(b_i p_{ij} - \mu d_{ij})x_{ij}$ subject only to matching and cap constraints, then tune μ until the realized disruption is near B. This Lagrangian approach is computationally attractive and monotone for a fixed μ (because it

is a maximum-weight matching with weights linear in b_i), but it does not, in general, enforce the budget exactly, and the mapping from bids to the tuned μ can itself break monotonicity. Understanding when "dual tuning" can be done in a bid-independent way (e.g., using historical calibration or a regulator-set μ) is therefore crucial if we want a truthful mechanism with a hard budget and general costs.

Third, one can broaden the design objective to include *bicriteria* guarantees, such as approximating welfare while allowing a small budget violation, or satisfying the budget while allowing a bounded welfare loss relative to a slightly larger budget. In governance settings, such bicriteria tradeoffs may be acceptable if the violation is auditable and rare, but they require the institution (regulator or internal policy) to specify what constitutes an acceptable violation probability. Formally, one would seek mechanisms such that

$$\Pr\left[\sum_{i,j} d_{ij}x_{ij} > B\right] \leq \delta \qquad \text{and} \qquad \mathbb{E}[W(x;v)] \geq \frac{1}{\alpha}\, W^*(B),$$

under DSIC constraints. Whether such mechanisms exist with good (α, δ) for general costs remains open.

Open questions. We would distill the "remove structure" agenda into the following concrete questions:

1. For general d_{ij}, what is the best achievable approximation ratio among deterministic DSIC mechanisms running in polynomial time? Are there hardness-of-truthfulness results that separate DSIC-approximability from plain approximability?

2. Can Lagrangian-based mechanisms be made DSIC with a hard budget by choosing μ bid-independently (or via a truthful auxiliary market for disruption units), and what welfare loss is unavoidable in doing so?

3. Under what distributional assumptions on (p_{ij}, d_{ij}) (e.g., smoothed analysis, bounded density) do monotone approximations exist generically, even if worst-case instances are hard?

Progress on any of these would materially expand the domain in which a "trust budget" can be enforced without resorting to ad hoc heuristics.

9.2 Robust and learning-based primitives: when (p_{ij}, d_{ij}) are estimated

Our model treats p_{ij} and d_{ij} as known primitives. In reality, they are predictions produced by machine learning systems, and both are subject to

uncertainty, drift, and strategic adaptation. This raises two intertwined issues: robustness (how outcomes change when predictions are wrong) and learning (how to update predictions while keeping incentives and governance coherent).

Robustness to misspecification. A first question is ex post robustness: if the true click probability is $\tilde{p}_{ij}$ and the true disruption is $\tilde{d}_{ij}$, but the mechanism uses (p_{ij}, d_{ij}), can we bound welfare loss and constraint violations in terms of prediction error? A minimal goal is a Lipschitz-style guarantee such as

$$W(x(p,d);v) \geq W^*(B;\tilde{p}) - \mathrm{Err}(p,\tilde{p}), \qquad \sum_{i,j} \tilde{d}_{ij} x_{ij}(p,d) \leq B + \mathrm{Err}(d,\tilde{d}),$$

where Err depends on norms of deviations. Even such a bound is nontrivial because the allocation $x(p,d)$ can change discontinuously when edge weights cross. Our structured-cost design already suggests a partial remedy: deterministic tie-breaking and bucketing dampen sensitivity to small perturbations by reducing the number of "knife-edge" comparisons. More generally, one can explicitly regularize the allocation rule (e.g., via smoothing of weights) to improve stability, at the possible cost of welfare.

Learning with incentive constraints. A second question is dynamic learning. Clicks provide feedback about p_{ij}, and user responses provide feedback about d_{ij} (at least indirectly, through dissatisfaction metrics). But learning requires exploration, and exploration is inherently incentive-relevant: showing an advertiser more often to learn their performance is a valuable allocation. In standard sponsored search, this tension is managed via multi-armed bandits and truthful mechanisms with learning; here, the presence of a hard disruption budget makes the exploration problem a *constrained* bandit with an additional resource consumption signal.

One promising direction is to model each edge (i,j) as having unknown mean reward (clicks) and unknown mean cost (disruption), and to design an online mechanism that satisfies, with high probability,

$$\sum_{t=1}^{T} \sum_{i,j} d_{ij}^{(t)} x_{ij}^{(t)} \leq B_T,$$

while achieving sublinear regret relative to the best fixed feasible policy in hindsight. The open mechanism-design question is: can we do this under DSIC when advertisers strategically report b_i each round, and when the learning algorithm's exploration choices depend on past clicks that are affected by the allocation? The single-parameter structure helps, but only if we can maintain monotonicity round by round (or in expectation) while updating beliefs about p_{ij} and d_{ij}.

Strategic manipulation of the measurement layer. A deeper difficulty is that d_{ij} is not a natural outcome like a click; it is a policy metric. If advertisers can manipulate d_{ij} (by changing creatives, landing pages, or metadata to look "safer"), then the problem becomes one of mechanism design with endogenous features, not just endogenous bids. DSIC does not protect us here: truthfulness concerns the reported b_i, not the strategic choice of content that changes (p_{ij}, d_{ij}).

This suggests that robust governance must be modeled explicitly. One could treat d_{ij} as produced by an audit process with noise and penalties: advertisers choose an action a_i that affects both value and measured cost, and misreporting or manipulation is deterred by expected sanctions. Embedding such an enforcement layer into the mechanism is conceptually straightforward but technically open: we would need equilibrium notions that combine bidding incentives with compliance incentives, and we would need to clarify what an auditor can observe.

9.3 Dynamic trust budgets across turns: state, replenishment, and online feasibility

Finally, in many deployments (e.g., conversational assistants or multi-step content generation), the platform makes a *sequence* of insertion decisions. Trust is then naturally modeled as a state variable: users may become fatigued or more skeptical after repeated disclosures, and regulators may require compliance over a horizon rather than per-response.

From per-instance to intertemporal constraints. A minimal dynamic variant replaces the per-instance budget with a horizon budget B_T:

$$\sum_{t=1}^{T} \sum_{i,j} d_{ij}^{(t)} x_{ij}^{(t)} \leq B_T,$$

or a rolling-window constraint. This turns the allocation into an online knapsack–matching problem with adversarial arrivals (contexts) and strategic bids each round. The platform must decide whether to "spend" disruption units now or save them for future contexts with higher value. In such settings, the dual variable μ becomes an intertemporal shadow price: it represents the opportunity cost of spending trust today rather than tomorrow.

State-dependent disruption. A richer model lets disruption depend on past allocations, e.g., $d_{ij}^{(t)} = d_{ij}(c_t, s_t)$ where s_t is a trust state that evolves as

$$s_{t+1} = f(s_t, x^{(t)}, \text{user reactions}),$$

and the constraint becomes $s_t \geq \underline{s}$ (do not drop below a minimum trust level) rather than $\sum d_{ij} x_{ij} \leq B$. This captures the idea that repeated insertions

can have compounding effects even if each individual insertion is "within budget." It also aligns more closely with user-experience realities, where the harm of additional insertions depends on how saturated the session already is.

Designing DSIC mechanisms in such stateful environments is largely open. Even defining $y_i(b)$ requires specifying how click probabilities depend on state, and monotonicity can fail because increasing b_i today may worsen state tomorrow, indirectly reducing i's future allocation. This creates a dynamic externality that is internal to the mechanism's state, not just across advertisers.

Open questions for dynamic budgets. We see several concrete questions at the boundary of online algorithms, learning, and incentives:

1. Can we design truthful online mechanisms that achieve constant-factor (or no-regret) welfare relative to the best offline policy while satisfying a hard cumulative disruption budget?

2. Under what conditions can the optimal policy be implemented by a bid-independent shadow price process $\{\mu_t\}$ (a "trust exchange rate") so that each round reduces to a monotone matching with weights $b_i p_{ij} - \mu_t d_{ij}$?

3. How should a regulator specify dynamic constraints so that they are both behaviorally meaningful (capturing fatigue and erosion of trust) and operationally auditable (verifiable from logs without revealing proprietary prediction models)?

Closing perspective. We view these open problems as complementary rather than competing. Removing structured-cost assumptions expands the computational frontier of truthful allocation under hard budgets; learning-based primitives address the empirical reality that (p_{ij}, d_{ij}) are estimated and strategically pressure-tested; and dynamic budgets capture the temporal nature of trust in real user interactions. Across all three, the unifying theme is that governance constraints are not merely "filters" on ad delivery: they are scarce resources that can be priced, audited, and optimized—but only if the mechanism's response to bids and predictions is stable enough to be both incentive compatible and institutionally credible.