Alistair: Efficient On-device Budgeting for Differentially-Private Ad-Measurement Systems

Pierre Tholoniat*
Columbia University

Kelly Kostopoulou*
Columbia University

Peter McNeely
Columbia University

Prabhpreet Singh Sodhi
Columbia University

Anirudh Varanasi
Columbia University

Benjamin Case
Meta Platforms, Inc.

Asaf Cidon
Columbia University

Roxana Geambasu
Columbia University

Mathias Lécuyer
University of British Columbia

Abstract

With the impending removal of third-party cookies from major browsers and the introduction of new privacy-preserving advertising APIs, the research community has a timely opportunity to assist industry in qualitatively improving the Web's privacy. This paper discusses our efforts, within a W3C community group, to enhance existing privacy-preserving advertising measurement APIs. We analyze designs from Google, Apple, Meta and Mozilla, and augment them with a more rigorous and efficient differential privacy (DP) budgeting component. Our approach, called *Alistair*, enforces well-defined DP guarantees and enables advertisers to conduct more private measurement queries accurately. By framing the privacy guarantee in terms of an individual form of DP, we can make DP budgeting more efficient than in current systems that use a traditional DP definition. We incorporate Alistair into Chrome and evaluate it on microbenchmarks and advertising datasets. Across all workloads, Alistair significantly outperforms baselines in enabling more advertising measurements under comparable DP protection.

1 Introduction

Major changes are occurring in Web advertising, offering significant potential to enhance online privacy. For years, various parties, known and unknown to users, have exploited vulnerabilities in Web protocols, such as third-party cookies and remote fingerprinting, to track user activity across the Web. They have used this data to target individuals with ads and measure ad campaign effectiveness. This situation is evolving in two significant ways. First, major browsers are making it harder to track people's activity across sites. Apple's Safari and Mozilla's Firefox have disabled third-party cookies since 2019 [20] and 2021 [32], respectively, with Google Chrome set to follow suit by the end of 2024 [34]. Browsers are also enhancing defenses against IP tracking [18] and remote fingerprinting [31, 2, 41].

Second, recognizing that online advertising is a critical component of the Web economy – and that perfect tracking protection is impossible – browsers are opening new, explicit APIs that enable measuring ad effectiveness and improving ad delivery across populations of users while establishing privacy protections for individual-level data. Initial designs, like Apple's PCM [35] and Google's FLoC [8], focused on intuitive but not rigorous privacy methods, leading to limited adoption either due to poor utility [3] or poor privacy [16]. Recently, browsers have shifted to more theoretically-sound privacy technologies: differential privacy (DP), secure multi-party computation (MPC), and trusted execution environments (TEEs). The hope is that theoretically-sound technologies can deliver more actionable privacy–utility tradeoffs than intuitive methods.

Despite progress, significant challenges remain in implementing these privacy technologies at Web scale. We believe the research community has a timely opportunity – nay, responsibility – to assist industry in enhancing these technologies so they can both deliver strong privacy protections and meet Web advertising needs. Only in this way can we hope to achieve widespread adoption of privacy-preserving APIs, remove incentives for individual tracking, and qualitatively improve the Web's privacy. This paper describes our efforts to understand and improve current *ad-measurement API* proposals, which enable advertisers to measure and optimize the effectiveness of their ad campaigns based on how often people who view or click certain ads go on to purchase the advertised product. Separate ad-targeting APIs are being developed [9], but we focus on ad-measurement APIs here.

The Private Advertising Technology Community Group (PATCG) [37] in the W3C is pursuing an interoperable standard for private ad-measurement APIs. The major proposals under consideration include Google's Attribution Reporting API (ARA) [4], Meta and Mozilla's Interoperable Private Attribution (IPA) [21], Apple's Private Ad Measurement (PAM) [36], and a hybrid proposal (Hybrid) [17] that builds on these three previous proposals. Our work focuses on these proposals, systematizing them into abstract models that we then analyze and compare for the purpose of identifying opportunities to improve their privacy-utility tradeoffs. This systematization, which can serve future researchers engaging in this space, constitutes our first contribution (§2).

*These authors contributed equally to this work.

Focusing on the differential privacy (DP) component, which is present in all four systems, we propose an improved design for DP budgeting that achieves higher utility at the same privacy level compared to existing systems. Ad-measurement systems use DP to prevent advertisers from learning too much about any individual user through the outputs of their measurement queries. They define a *privacy loss budget* (or *budget*) and execute each query with DP, accounting for the privacy loss from each query against the budget. Once the budget is exhausted, queries must stop to prevent advertisers from learning "too much" about any one person. This process is called *DP budgeting*. We observe that IPA operates in a traditional centralized-DP setting, where both query execution and DP budgeting are done centrally, while the other three systems operate under a non-standard DP setting, with queries executed centrally but DP budgeting done separately on each device. We find that this *on-device budgeting* cannot be formalized under a standard DP definition and instead requires a variant of DP, known as *individual DP* (IDP) or personalized DP [13], to properly formalize. This is a previously unknown aspect, and its establishment, along with the formal modeling of on-device budgeting systems and their analysis under IDP, constitutes our second contribution (§4, §5).

Through our IDP formalization, we uncover a set of powerful optimizations of privacy loss accounting that can enhance the utility of on-device budgeting systems by letting advertisers run more queries accurately under a given DP budget. IDP permits devices to maintain separate DP guarantees and to account for privacy loss based on their data, allowing a device to deduct zero privacy loss when it lacks relevant data for a query. Interestingly, one of the optimizations, which we prove through IDP machinery, is already inadvertently employed in existing on-device ad-measurement systems, such as ARA, without a clear justification. Our third contribution is to provide solid proof for this optimization and for a wider range of optimizations readily-available to ad-measurement systems doing on-device budgeting (§6).

Our fourth and last contribution is prototyping our optimized DP budgeting component in ARA within the Chrome browser, in a system called *Alistair* (§3 and §7), and evaluating it on microbenchmarks and advertising datasets (§8). Alistair goes beyond optimizing DP budgeting, representing the first DP budgeting design in existing ad-measurement systems to enforce a well-defined, fixed, user-time DP guarantee [25]. This guarantee is generally considered more meaningful for users compared to the event-level DP implemented by ARA or the unbounded, continuously-refreshed, user-level guarantee previously used by IPA before switching, following our recommendation, to user-time DP. Our experiments with datasets from PATCG and Criteo show that Alistair's efficient on-device budgeting is capable of completing all advertiser queries in our workloads and obtains a $\times$1.16–2.88 improvement on their median accuracy compared to the user-time version of ARA, which also completes all queries. IPA only completes 3.75%–16.59% of the workload before it runs out of budget. These results mark our initial step toward promoting the widespread adoption of ad-measurement systems that provide strong privacy guarantees. We will open-source our prototype.

2 Systematization of Ad-Measurement APIs

We systematize the designs of the privacy-preserving ad-measurement systems considered for a potential interoperable standard at PATCG: Meta and Mozilla's IPA, Google's ARA, Apple's PAM, and Meta and Mozilla's Hybrid. ARA and IPA are implemented; PAM and Hybrid exist only as design docs. We abstract their functionality for comparison and articulate the improvement opportunity addressed in this paper.

2.1 Example Scenario

We use a fictitious scenario to convey the general motivation and requirements behind ad-measurement systems from the perspective of multiple players: Ann, a web user; Nike, an advertiser seeking to measure the effectiveness of its ad campaigns; and Meta, an ad-tech that serves as an intermediary to optimize placement of ads from multiple advertisers onto multiple users.

User perspective. Ann browses various *publisher* sites that provide content she is interested in, such as nytimes.com and facebook.com. Ann does not mind seeing relevant advertising, understanding that it funds the free content she enjoys. At times, Ann also finds them useful for discovering new products. For instance, she recently clicked on a nytimes.com ad for Nike running shoes that absorb knee shocks, eventually buying a pair to alleviate her knee problem. However, she also values her privacy and expects that no site, including publishers, advertisers, or ad-tech intermediaries, track her across sites (*no cross-site tracking*). She also expects *limited within-site linkability*, meaning sites cannot link her activities even within the same site across cookie-clearing browsing sessions (*e.g.,* across incognito sessions). Ann knows that effective advertising requires some privacy loss but expects it to be *explicitly bounded* and *transparently reported* by her browser.

Fig. 1 shows a screenshot of the privacy loss dashboard we built for Alistair in Chrome. Ann can use it to monitor the privacy loss resulting from various sites and intermediaries querying her ad interactions, including *impressions* (*e.g.,* ad views and clicks) and *conversions* (*e.g.,* purchases, cart additions). While Ann may not grasp the concept of differential privacy that underpins the reported privacy loss, she trusts her browser to always enforce protective bounds on it.

Advertiser perspective. Nike conducts multiple ad campaigns for their running shoes, some focused on the technical details of shock-absorbing technology, others focused on aesthetics. Nike aims to understand which campaigns resonate best with different demographics and contexts (*e.g.,* publisher

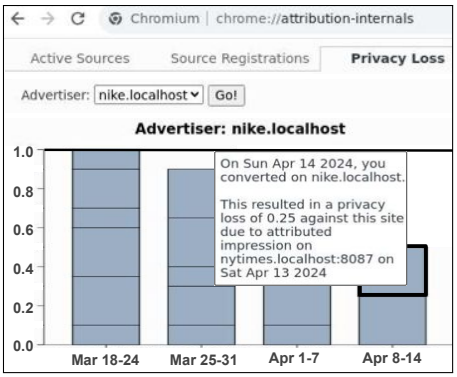


Fig. 1. Privacy loss dashboard. Screenshot from our Chrome implementation of Alistair (minimally edited for visibility).

sites, content types). Previously, Nike used third-party cookies and remote device fingerprinting[1] to match individuals who viewed campaigns with those who made purchases, attributing purchase value to specific impressions based on an *attribution function*, such as last-touch (last impression gets all value) or equal credit (value split evenly across recent impressions). Using such *attribution reports* from many users, Nike measured the total purchase value attributed to different campaigns. Nike then used these results to target different demographics with the most effective campaigns.

With third-party cookies disabled and remote fingerprinting becoming more challenging, Nike is transitioning to ad-measurement APIs. It expects to conduct similar attribution measurements as before with similar accuracy. It is important to note that ad measurement has always been prone to imprecision, such as inaccurate matches, cookie clearings, and fraud. Therefore, Nike's expectation of accuracy from the APIs is not overly stringent. Nike plans to perform numerous conversion attribution measurements over time to adapt to changing user preferences and product offerings. These measurements are multi-publisher, single-advertiser, summation queries and represent a first crucial query type that ad-measurement systems strive to support.

Ad-tech perspective. In addition to advertising on nytimes.com, Nike also advertises on Meta, a content provider (*a.k.a.* publisher or ad-tech) that runs its own, in-house advertising platform. Ann uses Meta's facebook.com site to read posts related to running and other interests. To show her the most relevant ads, the site requires her to log into her account and then tracks her activity within the site to build a profile of her interests. Ann accepts that Meta learns about her interests as she interacts with content on the site while logged into her account; however, Ann expects Meta not to be tracking her across other sites on the Web, and also to not be linking her interactions as part of different accounts. For example, while Meta may learn that Ann is passionate about running, and hence may show her the Nike running-shoe ad, Meta should not be able to tell whether Ann later buys the shoes, as that conversion occurs on nike.com. Still, to maximize the effectiveness of ads (and return on Nike's ad spend), Meta needs to be able to train a machine learning (ML) model that can predict, for each user profile, in each context, which ad coming from which advertiser would be most effective to show, in terms of maximizing the likelihood of an eventual conversion. This model-training procedure can be thought of as bringing together many conversion attributions reports corresponding to impressions that occur on one or more publishers (facebook.com here, but also potentially instagram.com) and conversions that occur on the many advertiser sites buying ads through Meta. This type of multi-advertiser, optimization query is a second class of queries that ad-measurement APIs aim to support without exposing cross-site information and while limiting within-site linkability (to meet expectations when the user switches accounts).

[1]The example is fictitious, and claims about the companies are also fictitious.

2.2 Ad-Measurement Systems

IPA, ARA, PAM, and Hybrid aim to meet the dual requirements of privacy and transparency for people and utility for advertisers and other Web-advertising parties, referred to as *queriers*. We define utility as the number of accurate measurement queries a querier can execute under a privacy constraint. Despite differences in terminology, privacy properties, and mechanisms, these systems share commonalities. One commonality is the use of DP techniques, with ARA emphasizing event-level DP and IPA, PAM, and Hybrid focusing on user-time DP. This paper adopts a view of all systems through the lens of user-time DP, enforced separately for each querier site. We define this semantic in §5.3.

Common architecture. The high-level architecture of all four systems is very similar (depicted in Fig. 2(a)). All four act as intermediaries between user devices and sites. Previously, these parties collected impression and conversion events directly, matched them based on the originating device through third-party cookies, performed attribution functions, and aggregated reports. To break these privacy-infringing direct data flows, ad-measurement systems interpose themselves between devices and sites, offering a differentially-private querying interface over impression and conversion data. All ad-measurement systems consist of three main components: (1) the *attribution function*, which matches conversions to relevant impressions on the same device and assigns conversion value based on selected attribution logic; (2) the *DP query execution*, which aggregates attribution reports and adds calibrated noise to ensure DP; and (3) the *DP budgeting component*, which uses DP composition to account for privacy loss from each query against a maximum privacy loss bound called a DP budget.

A critical difference between the ad-measurement systems lies in the execution location of these functions. In IPA, all components operate off the device within an MPC protocol. In the other three systems (ARA, PAM, and Hybrid), the attribution function and DP budgeting occur on the device, with

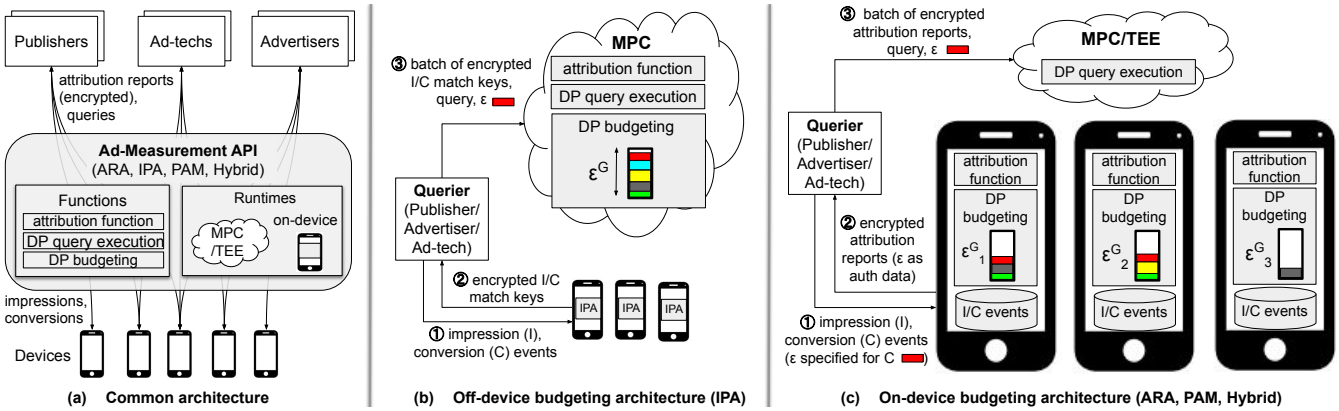


Fig. 2. Architectures of ad-measurement systems. Plenty of commonality, with a core distinction being where attribution and DP budgeting occur: off device (IPA) vs. on device (ARA, PAM, Hybrid).

DP query execution taking place in either an MPC protocol (PAM, Hybrid) or a TEE (ARA). In all systems, the MPC and TEE engines are trusted to not leak inputs or intermediary states of computation, and the devices are trusted not to leak their own data. The distinction in where attribution and DP budgeting are executed has significant implications.

Off-device budgeting (IPA). Fig. 2(b) depicts the IPA architecture, operating in a standard centralized-DP setting. The MPC handles all DP querying functions, while the device side of IPA is minimal, focusing on generating a *match key* for matching impressions with conversions on the same device. When nytimes.com sends an ad for Nike shoes to Ann's device ①, the device responds with a match key for that device, encrypted and secret shared toward the MPC parties ②. When Ann later purchases the shoes on nike.com, her device sends Nike the same match key, also encrypted and secret shared toward the MPC parties. Periodically, NYtimes sends batches of encrypted impression match keys to Nike, who cannot directly match these with its conversion match keys due to the encryption and secret sharing. Instead, Nike collects its conversion match keys and NYtimes' impression match keys into batches and submits them to the MPC, specifying the privacy budget ϵ to spend on the query ③. The MPC verifies Nike's budget, joins impressions and conversions based on the match keys, runs the attribution function between each conversion and its matching impressions, enforces a match key level L^1 cap on attributed values (for sensitivity control), aggregates, and finally ads DP noise to enforce ϵ-DP. The MPC parties deduct ϵ from Nike's remaining budget, which is maintained by them. When that budget runs out, IPA refuses to run queries for Nike, until the per-site budget is "refreshed," which happens periodically (e.g., daily).

On-device budgeting (ARA, PAM, Hybrid). Fig. 2(c) shows the on-device architecture, which operates in a rather non-standard DP setting. While DP query execution occurs centrally on the MPC or TEE, attribution and DP budgeting are done separately on each device. Every device maintains a local database of its impression and conversion events. For instance, when Ann receives a shoe ad impression on nytimes.com, her device logs it locally ①. Later, when Ann buys the shoes on nike.com, Nike requests an attribution report for that conversion. Ann's device checks its database for relevant impressions, runs the attribution function with an L^1 cap on attributed values (to control sensitivity), and returns an encrypted attribution report ②. Nike collects batches of encrypted reports and sends them to the MPC or TEE for DP aggregation, where the reports are summed and noise is added based on the L^1 cap and Nike's ϵ parameter ③.

One crucial aspect missing from this description is DP budgeting. Unlike centralized DP, privacy loss accounting in on-device systems occurs at the time a conversion report is requested by the advertiser, well before query execution. When Nike requests a conversion report, it specifies the ϵ parameter for the future query on a batch of reports. The device checks Nike's budget, generates the report, encrypts it, attaches ϵ as authenticated data, and sends it to Nike, deducting ϵ from Nike's budget. Because the DP budget is spent at the device, each report can only be aggregated once, or a finite and prearranged number of times, for sensitivity control.

2.3 Improvement Opportunity

On-device budgeting systems have some advantages over off-device budgeting systems, but also raise a challenge, which we undertake as our opportunity for improvement. First, on-device budgeting systems can support user transparency by controlling per-site budgets and tracking privacy losses tied to specific attributions, as demonstrated in the Alistair privacy loss dashboard (Fig. 1). In IPA, the device can track only the encrypted match keys it returns to sites and not specific privacy losses incurred by each user through subsequent matching and aggregation in the MPC.

Second, on-device systems do budgeting at finer granularity, which can be more efficient. In off-device budgeting, there's a single system-wide budget, ϵ^G, enforced for each site. In on-device budgeting, each device maintains a separate privacy budget ϵ_d^G and only consumes from it for queries to which the device provides reports. This fine granularity

means that even if Nike runs out of privacy budget on Ann's device, it can still run measurement queries for other product campaigns where other users may have budget left. However, justifying this behavior requires formalization under the less standard (but equally protective) privacy definition known as IDP [13], which enforces privacy guarantees for each device.

The challenge lies in formalizing the data, query, and system model that captures the behavior of on-device ad-measurement systems, and to prove its IDP properties. This formalization opens opportunities for optimizing DP budgeting in on-device systems by deducting privacy loss based on the device's data. However, it also requires keeping the remaining privacy budgets on each device private, as revealing the remaining budgets leaks data. Thus, we focus on developing a formally-justified, practical and efficient DP budgeting module, called *Alistair*, that is suitable for on-device systems like ARA, PAM, and Hybrid and provides queriers with higher utility at the same level of DP protection.

3 Alistair Overview

In designing Alistair, we are guided by three principles. First, it should enforce well-defined DP guarantees at an industry-acceptable level of granularity. We focus on a fixed "user-time" DP guarantee enforced separately for each querier, a granularity supported by IPA, PAM, and Hybrid, and acknowledged by Apple, Meta, and Mozilla as the minimum acceptable. Second, Alistair should accommodate similar use cases and queries as existing systems. Third, given browsers' heightened efforts against tracking, Alistair should not introduce new vectors for illicit tracking.

Fig. 3 shows Alistair's architecture as well as an example execution shown as a red overlay atop the architectural components. We discuss each in turn.

3.1 Architecture

Alistair adopts on-device budgeting similar to ARA, PAM, and Hybrid. DP query execution occurs within an MPC or TEE, which we trust to not leak its inputs or intermediary state of its computation. Alistair does not modify this component, so it is omitted from Fig. 3. Alistair focuses on altering the on-device component, notably ARA in our prototype. While the externally-facing APIs remain unchanged, we modify: (1) the on-device events database to support a "user-time" guarantee and (2) the internal workings of the attribution function and DP budgeting to implement this guarantee efficiently.

Alistair enforces what we call *individual device-epoch* ϵ_d^G-*DP* for each querier site, formally defined in §5.3. The device-epoch granularity is the same as the traditional "user-time" from DP literature [25, 27, 28], but we rename it to reflect that a "user" is not directly observable to a device or browser, the scope in which Alistair operates. To support this guarantee, we partition the on-device events database by time into *epochs*, such as a week or a month. Within each epoch e, device d collects impression and conversion events into a *device-epoch database*, denoted D_d^e. Queriers submit multiple queries over time, accessing data from one or more epochs, overlapping with each other. For each epoch e, we guarantee to device d that no single querier using Alistair will learn more about d's data in epoch e (D_d^e) than permitted by a traditional ϵ_d^G-DP guarantee.

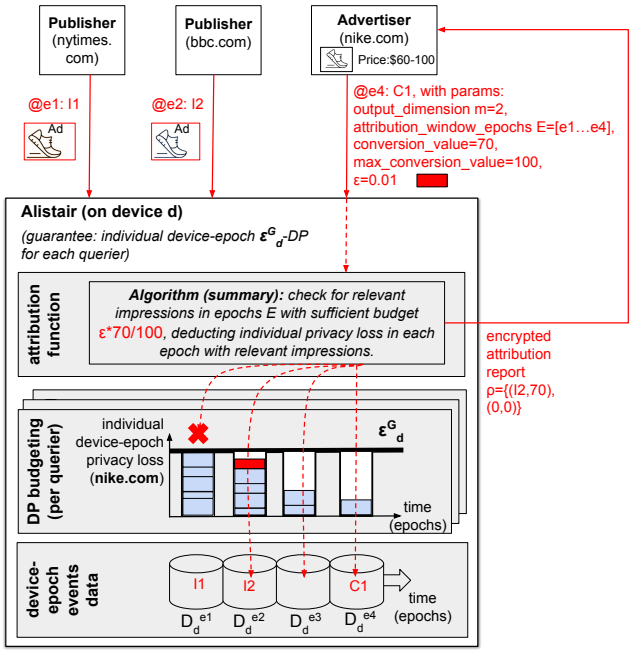


Fig. 3. Alistair architecture and example execution (red overlay). §3.1 describes the architecture and §3.2 the example execution. Some notation: $@e_1 : I_1$ symbolizes that Ann's device gets an impression I_1 of a Nike shoe ad from nytimes.com in time epoch e_1. Red dotted arrows symbolize the attribution function's search for relevant impressions in epochs $e_1 - e_4$.

The formal method we use to implement *DP budgeting* in Alistair is a privacy filter [39], an abstraction that ensures that the privacy loss from a composed sequence of queries exhibiting various forms of adaptivity does not exceed a pre-specified privacy budget. In Alistair, the DP budgeting component maintains, for each querier, multiple privacy filters – one for each device-epoch database. Fig. 3 shows the set of filters for nike.com. Each filter is initialized with pre-specified privacy budget ϵ_d^G and its purpose is to bound the cumulative privacy loss as a result of nike.com running queries that incorporate data from that device-epoch.

Recall that in on-device budgeting systems, privacy loss accounting happens not upon the execution of the query, but when the attribution report is generated. The component responsible for generating this report is the *attribution function*. Upon a conversion, the attribution function checks for relevant impressions in the device-epoch databases in a specified window of epochs. The privacy filters prevent the use of impression data from epochs lacking sufficient budget. If access is denied in an epoch, the attribution function continues its search for relevant impressions in other epochs.

For epochs with sufficient budget, the filter permits use of the device-epoch data, but must deduct some privacy loss.

Viewed through a standard centralized-DP lens, the privacy loss to deduct would be ϵ, the privacy parameter of the DP query that will be later enforced by the MPC or TEE. A key insight coming from our formal modeling of on-device budgeting systems (§4) is that they must be viewed under a (less standard) individual-DP lens (§5), but once we do so, multiple opportunities emerge for optimizing privacy loss accounting, i.e., deducting "less than ϵ." We specify these optimizations in §6 and only provide an example here.

3.2 Execution Example

The red overlay in Fig. 3 shows how the attribution function operates for an example from the scenario in §2.1. Suppose Ann gets two impressions of Nike shoe ads, from different campaigns: first impression in epoch e_1 and the second in epoch e_2. She gets no impression of a Nike ad in epoch e_3. Later, in epoch e_4, Ann purchases the shoe. At that time, nike.com registers a conversion $C1$ for this product with Ann's device and requests an attribution report for it, with the following parameters: the set of epochs E in which to look for relevant impressions; the maximum number of impressions, m, to attribute value to; the conversion's value, which is the price she paid for the shoes, \$70; and ϵ, the privacy parameter enforced by the MPC or TEE when running the query.

Let us assume that the advertised shoes range in price depending on color. Ann purchased a pair valued at \$70, but other people who have converted may have selected versions at different prices, up to a maximum of \$100. Thus, while Ann's specific conversion value is \$70, Nike's query will likely involve attribution reports from devices with conversion values up to \$100. This means that in the MPC or TEE, assuming a summation query and the Laplace mechanism, the standard deviation of the noise added to the aggregate will depend on $100/\epsilon$, where 100 is the *global sensitivity* of the summation query, i.e., the biggest change *any* device-epoch entry can make on the output of the summation. Yet, due to her lower purchase value, Ann can only contribute up to \$70, from all her device-epochs, to the summation query.

Herein lies the intuition of IDP: it lets us account for privacy loss on the basis of *individual sensitivity*, defined as the maximum change that *a given device-epoch* can make on the output of a query. This means that Ann's device should be able to deduct privacy loss of $\epsilon' = \$70/\$100 * \epsilon$ from the privacy filters for the epochs in the attribution window E. This is one optimization implied by our IDP formalization, but there are others. For example, if an epoch in the attribution window E has no relevant impressions (such as epoch e_3 in our Fig. 3 example), must we deduct ϵ' from it? The answer is no, as that device-epoch's individual sensitivity for an additive query is 0, and hence its individual privacy loss is 0, too. §6 documents our general optimization results.

Here's how the optimizations apply for the example in Fig. 3. Alistair's attribution function checks for relevant impressions in epochs $e_1 - e_4$. In epoch e_1, its access to the data $D_d^{e_1}$ is denied because the epoch's filter ran out of budget for nike.com. In epoch e_2, there is budget and the attribution function finds a relevant impression, I_2, in that epoch's data, so the filter deducts ϵ' (shown as a red square in the e_2 filter). The attribution function checks for relevant impressions in epoch 3 since its filter still has budget left. However, since it finds no relevant impression there, it deducts nothing from that filter. Finally, in epoch e_4, where the conversion happened but no relevant impression happened, then, through a formalization of publicly available information that we support (§4), we can justify that no privacy loss occurs in e_4. In the end, the returned attribution report assigns the entire 70 value to the single impression found, I_2, but for privacy, it also includes a null value since Nike asked for two attributions. If the attribution function had found no relevant impressions, or Nike had run out of budget in all epochs, then the attribution function would still need to return a two-dimensional report, with both entries containing null values. This prevents leaking information regarding the presence or absence of a relevant ad on the device.

3.3 IDP Implications

The preceding examples illustrate the kinds of budget savings we can expect from Alistair. §8 confirms experimentally that these savings result in significant increases in the number of queries advertisers can execute accurately for the same privacy loss. However, despite the benefits, IDP also has the potential to introduce bias into query results. Because privacy loss and remaining budgets must be kept secret from advertisers, when a device runs out of budget, it must continue to participate in queries, providing "null" information and biasing the results. In the preceding example, the report should have returned two impressions, not one, but because Nike ran out of budget epoch e_1, I_1 is not returned. This changes what the report looks like, in a way that Nike cannot know. As another example, suppose Nike wants to do first-touch attribution, i.e., assign the entire \$70 value to the first-seen impression. Since e_1 is out of budget, I_2 gets returned instead of I_1, changing the semantic of first-touch. Once again, because remaining budgets in each filter must be kept private, the device cannot reveal to Nike that it had changed this report.

We have begun to investigate this problem and have encouraging theoretical results on a method to enable advertisers to *measure* an upper bound on the level of bias in their query results. The insight is to allow the advertiser to run, together with an ad-measurement query, a side query that DP-counts the number of reports in a batch that may have been changed due to some devices running out of budget in one or more epochs. The advertiser spends a bit more budget for each report (in epochs that have impressions) for that bias-measurement query, but in return gets a rigorous high-probability bound on the bias they incurred in their results. We include our results in Appendix §D, but because they are presently only theoretical, we do not claim a contribution in

this direction and leave implementation and evaluation of bias detection and mitigation techniques for future work.

Our experimental evaluation addresses the question of how hiding privacy budgets affects query accuracy (question Q2 in §8). It shows that for our workloads, Alistair's particularly efficient budget optimizations enable it to still execute more queries with higher accuracy compared to the baselines, ARA and IPA. Over the next three sections, we thus focus on the theoretical contributions that make this efficient budgeting possible. §4 presents a formal model of Alistair's functionality. §5 defines the privacy properties under individual DP, and §6 presents our optimization results.

4 Formal System Model

To enable rigorous analysis of DP properties and optimization opportunities, we must first establish a formal model that captures Alistair's behavior. Such a model is missing from today's ad-measurement systems, making it impossible to analyze them, or to formally justify optimizations. While our model is tuned for Alistair, it should serve as a good basis for other systems' analyses.

We begin by specifying the types of data and queries that Alistair operates with. We take the perspective of a *fixed querier*, such as an advertiser, publisher, or ad-tech. Appendix §A specifies the end-to-end algorithm that formally captures, using the data and query models we define in this section, and the Alistair behavior that we informally described in §3. Since we only use this algorithm in proofs of the DP guarantees we claim in §5, we omit it here.

4.1 Data Model

Our data model is based on conversion and impression events collected by user devices and grouped by the time epoch in which they occurred. We view the data available to queriers as a database of such device-epoch groups of events, coming from many devices and defined formally as follows.

Conversion and impression events (F). Consider a domain of impression events $\mathcal{I}$ and a domain of conversion events $\mathcal{C}$. A set of impression and conversion events F is a subset of $\mathcal{I} \cup \mathcal{C}$. The powerset of events is $\mathcal{P}(\mathcal{I} \cup \mathcal{C}) := \{F : F \subset \mathcal{I} \cup \mathcal{C}\}$.

Device-epoch record (x). Consider a set of epochs $\mathcal{E}$ and a set of devices $\mathcal{D}$. We define the domain for device-epoch records $\mathcal{X} := \mathcal{D} \times \mathcal{E} \times \mathcal{P}(\mathcal{I} \cup \mathcal{C})$. That is, a *device-epoch record* $x = (d, e, F)$ contains a device identifier d, an epoch identifier e, and a set of impression and conversion events F.

Database (D). A *database* is a set of device-epoch records, $D \subset \mathcal{X}$, where a device-epoch appears at most once. That is, $\forall d, e \in \mathcal{D} \times \mathcal{E}, |\{F \subset \mathcal{I} \cup \mathcal{C} : (d, e, F) \in D\}| \leq 1$. We denote the set of all possible databases by $\mathbb{D}$. This will be the domain of queries in Alistair. Given a database $D \in \mathbb{D}$ and $x \in \mathcal{X}$, $D + x$ denotes that device-epoch record x is added to database D that initially did not include it.

Device-epoch events data ($\mathbf{D^e_d}$, $\mathbf{D^E_d}$). Given a database $D \in \mathbb{D}$, we define $D^e_d \subset \mathcal{I} \cup \mathcal{C}$ as $D^e_d = F$ if there exist (a unique) F such that $(d, e, F) \in D$, and $D^e_d = \emptyset$ otherwise. Think of this as the event data of device d at epoch e. We also define $D^E_d := (D^e_d)_{e \in E} \in \mathcal{P}(\mathcal{I} \cup \mathcal{C})^{|E|}$ the events of device d over a set of epochs E (typically a contiguous window of epochs).

Public events (P). A key innovation in Alistair's data model is to support incorporation of side information that can be reliably assumed as available to the querier. For example, an advertiser such as Nike can reliably know when someone places a product into a cart (i.e, a conversion occurred), though depending on whether the user is logged in or not, Nike may or may not know who did that conversion.

We model such side information as a domain of *public events* for a querier, denoted $P \subseteq \mathcal{I} \cup \mathcal{C}$. P is a subset of all possible events, that will be disclosed to the querier if they occur in the system. We do *not* assume that the querier knows the devices on which events in P occur, and different queriers can have knowledge about different subsets of events. Such side information is typically not modeled explicitly in DP systems, as DP is robust to side information. Alistair also offers such robustness to generic side information. However, we find that additionally modeling the "public" events known to the querier has two key benefits. First, it opens DP optimizations that leverage this known information to consume less privacy budget. Second, it lets us formally define within-site linkability and adapt our design to provide a DP guarantee against such linkability.

4.2 Query Model

In on-device systems, queries follow a specific format: first the attribution function runs locally to generate an attribution report, on a set of devices with certain conversions; then, the MPC sums the reports together and returns the result with DP noise. Formally, we define three concepts: attribution function, attribution report, and query.

Attribution function, *a.k.a.* attribution (A). Fix a set of events relevant to the query $F_A \in \mathcal{P}(\mathcal{I} \cup \mathcal{C})$, and $k, m \in \mathbb{N}^*$ where k is a number of epochs. An *attribution function* is a function $A : \mathcal{P}(\mathcal{I} \cup \mathcal{C})^k \to \mathbb{R}^m$ that takes k event sets $F_1, \ldots, F_k$ from k epochs and outputs an m-dimensional vector $A(F_1, \ldots, F_k)$, such that only *relevant events* contribute to A. That is, for all $(F_1, \ldots, F_k) \in \mathcal{P}(\mathcal{I} \cup \mathcal{C})^k$, we have:

$$A(F_1, \ldots, F_k) = A(F_1 \cap F_A, \ldots, F_k \cap F_A).$$

Attribution report, *a.k.a.* report (ρ). This is where the non-standard behavior of on-device budgeting systems, which deduct budget only for devices with specific conversions, becomes apparent. Intuitively, we might consider attribution reports as the "outputs" of an attribution function. However, in the formal privacy analysis, we must account for the fact that only certain devices self-select to run the attribution function (and thus deduct budget). We model this in two steps. First, we introduce a conceptual *report identifier*, r, a unique random number that the device producing this report generates and shares with the querier at report time.

Second, we define an *attribution report* as a function over the whole database D, that returns the result of an attribution function A for a set of epochs E *only for one specific device* d *as uniquely identified by a report identifier* r. Formally, $\rho_r : D \in \mathbb{D} \mapsto A(D_d^E)$. At query time, the querier selects the report identifiers it wants to include in the query (such as those associated with a type of conversion the querier wants to measure), and devices *self-select* whether to deduct budget based on whether they recognize themselves as the generator of any selected report identifiers. Defining attribution reports on D lets us account for this self-selection in the analysis.

Query (Q). Consider a set of report identifiers $R \subset \mathbb{Z}$, and a set of attribution reports $(\rho_r)_{r\in R}$ each with output in $\mathbb{R}^m$. The *query* for $(\rho_r)_{r\in R}$ is the function $Q : \mathbb{D} \to \mathbb{R}^m$ is defined as $Q(D) \coloneqq \sum_{r\in R} \rho_r(D)$ for $D \in \mathbb{D}$.

4.3 Instantiation in Example Scenario

To make our data and query models concrete, we instantiate the scenarios from §2.1.

User Ann's data, together with that of other users, populates dataset D. Each device Ann owns has an identifier d, and events logged from epoch e go into observation $x = (d, e, F)$. $F = I \cup C$ is the set of all events logged on that device during that epoch, including impressions (I) shown to Ann by various publishers, and conversions (C) with various advertisers. Other devices of Ann, other epochs, and other users' device-epochs, constitute other records in the database.

The advertiser, Nike, can observe some of Ann's behavior on its site. As a result, any such behavior logged in C on nike.com constitutes public information for querier Nike. This might include purchases, putting an item in the basket, as well as associated user demographics (*e.g.,* when Ann is logged-in). However, Nike cannot observe impression or conversion events on other websites. As a result, for this querier $P = C_{\text{Nike}}$, which denotes all possible events that can be logged on nike.com. Each actual event in this set (e.g., $F \cap C_{\text{Nike}}$, including Ann's purchase) is associated with an identifier r in Alistair. Using these identifiers, Nike can analyze the relative effectiveness of two ad campaigns a_1 and a_2 on a given demographics for a product p, such as the shoes Ann bought. First, Nike defines the set of relevant events for the shoe-buying conversion; these are any impressions of a_1 and a_2. Nike uses these relevant events in an attribution function $A : \mathcal{P}(\mathcal{I} \cup \mathcal{C})^{|E|} \to \mathbb{R}^2$ that looks at epochs in E and returns, for example, the count (or value) of impression events corresponding to ads a_1 and a_2. Third, using the set of report identifiers r from purchases of p from users in the target demographic, Nike constructs a query Q that will let it directly compare the proportion of purchases associated with ad campaign a_1 with campaign a_2.

An ad-tech, such as Meta, is interested in learning ML models to better target ads to its users, using conversions as a metric to optimize. To this end, Meta can learn a logistic regression mapping public (to Meta) features from its users and attributes of ads (together denoted X_d for device d), to conversion labels. This is possible under Alistair's queries by defining an attribution function A that returns X_d if there is a conversion, zero otherwise, and using algorithms to fit logistic regressions under known features but private labels [44].

Other perspectives we can readily support. We have kept the display and conversion events well separated, but many companies would likely be in both camps (e.g., Meta advertising for other services of the company). In this case, $P = \mathcal{I}_{\text{Meta}} \cup \mathcal{C}_{\text{Meta}}$, which is supported the same way by Alistair. In this scenario, Meta can run ML models "in the clear" for events happening exclusively on its platform, but would use the Alistair API to train models for displays from other companies (with private conversions).

5 IDP Formulation and Guarantees

Having defined the data and query models for Alistair, we next define and prove its privacy guarantees, which we model through the lens of individual DP. After defining our neighboring relation §5.1, we define traditional DP in §5.2, primarily for reference. Then we define individual DP in §5.3. In §5.4, we then state the IDP guarantees we have proven for Alistair, which protect against both cross-site tracking and within-site linkability (proofs in Appendix §B).

5.1 Neighboring Databases

A DP guarantee establishes the neighboring database relation, determining the unit of protection. In our case, this unit is the device-epoch record. To account for the existence of public event data (§4.1), we constrain neighboring databases to differ by one device-epoch record *while preserving public information*. This ensures that a database containing an arbitrary device-epoch record is indistinguishable from a database containing a device-epoch record with the same public information but no additional data.

Neighboring databases under public information ($D \sim_x^P D'$). Given $D, D' \in \mathbb{D}$, $x = (e, d, F) \in \mathcal{X}$ and $P \subset \mathcal{I} \cup \mathcal{C}$, we write $D \sim_x^P D'$ if there exists $D_0 \in \mathbb{D}$ such that $\{D, D'\} = \{D_0 + (e, d, F), D_0 + (e, d, F \cap P)\}$. This definition corresponds to a replace-with-default definition [14] combined with Label DP [15]. Although public data is baked into our neighboring relation, which makes it specific to each individual querier, we have proven that composition across queriers is still possible, which is important to reason about collusion (Appendix §B.3).

5.2 DP Formulation (for Reference)

The main implication of DP's neighboring definition is that noise needs to be applied on query results based on the query *sensitivity*, the worst-case change in query result between two neighboring databases. Traditional DP mechanisms use the global sensitivity.

Global sensitivity. Fix a query $q : \mathbb{D} \to \mathbb{R}^m$ for some m (so q could be either a query or an individual report in our formulation). We define the *global L_1 sensitivity of q* as follows:

$$\Delta(q) := \max_{D,D' \in \mathbb{D}: \exists x \in \mathcal{X}, D' = D + x} \|q(D) - q(D')\|_1. \quad (1)$$

Device-epoch DP. When scaling DP noise to the global sensitivity under our neighboring definition, we can provide device-epoch DP. Fix $\epsilon > 0$ and $P \subset \mathcal{I} \cup \mathcal{C}$. A randomized computation $\mathcal{M} : \mathbb{D} \to \mathbb{R}^m$ satisfies *device-epoch ϵ-DP* if for all databases $D, D' \in \mathbb{D}$ such that $D \sim_x^P D'$ for some $x \in \mathcal{X}$, for any set of outputs $S \subseteq \mathbb{R}^m$ we have $\Pr[\mathcal{M}(D) \in S] \leq e^\epsilon \Pr[\mathcal{M}(D') \in S]$. This is the traditional DP definition, instantiated for our neighboring relation.

5.3 IDP Formulation

Since queries are aggregated from reports computed on-device with known data, we would prefer to scale the DP noise to the individual sensitivity, which is the worst case change in a query result triggered by the specific data for which we are computing a report.

Individual sensitivity. Fix a function $q : \mathbb{D} \to \mathbb{R}^m$ for some m (so q could be either a query or an individual report in our formulation) and $P \subset \mathcal{I} \cup \mathcal{C}$. Fix $x \in \mathcal{X}$. We define the *individual L^1 sensitivity of q for x* as follows:

$$\Delta_x(q) := \max_{D,D' \in \mathbb{D}: D' = D + x} \|q(D) - q(D')\|_1. \quad (2)$$

While we cannot directly scale the noise to individual sensitivity, we can scale the on-device budget consumption using this notion of sensitivity. That is, for a fixed and known amount of noise that will be added to the query, a lower individual sensitivity means that less budget is consumed from a device-epoch. This approach provides a guarantee of individual DP [13, 14] for a device-epoch, defined as follows.

Individual device-epoch DP. Fix $\epsilon > 0$, $P \subset \mathcal{I} \cup \mathcal{C}$, and $x \in \mathcal{X}$. A randomized computation $\mathcal{M} : \mathbb{D} \to \mathbb{R}^m$ satisfies *individual device-epoch ϵ-DP for x* if for all databases $D, D' \in \mathbb{D}$ such that $D \sim_x^P D'$, for any set of outputs $S \subseteq \mathbb{R}^m$ we have $\Pr[\mathcal{M}(D) \in S] \leq e^\epsilon \Pr[\mathcal{M}(D') \in S]$.

Intuitively, IDP ensures that, from the point of view of a fixed device-epoch x, the associated data F is as hard to recover from query results as it would be under DP.

5.4 IDP Guarantees

Through IDP, we prove two main properties of Alistair: (1) **Individual DP guarantee**, which bounds *cross-site leakage*, showing Alistair guarantees individual DP under public information; and (2) **Unlinkability guarantee**, which bounds *within-site linkability*, demonstrating that even a first-party adversary cannot distinguish (in a DP sense) whether a set of events is all on one device, or spread across two devices. Proofs are in Appendix §B.

For the IDP guarantee, we give two versions. First, a stronger version under a mild constraint on the class of allowed queries, specifically that $\forall i, \forall F,\ A(F_1, ..., F_{i-1}, F_i \cap P, F_i, ..., F_k) = A(F_1, ..., F_{i-1}, \emptyset, F_i, ..., F_k)$. A sufficient condition for this is to ensure that queries leverage public events only through their report identifier, *i.e.,* $F_A \cap P = \emptyset$. The queries from the scenarios we consider (§2.1 and Appendix **??**) satisfy this property. Second, a slightly weaker version of the DP guarantee with increased privacy loss, but with no constraints on the query class, which is useful when considering colluding queriers.

Theorem 1 (**Individual DP guarantee).** *Fix a set of public events $P \subset \mathcal{I} \cup \mathcal{C}$, and budget capacities $(\epsilon_d^G)_{d \in \mathcal{D}}$.* **Case 1:** *If all the queries use attribution functions A satisfying $\forall i, \forall F,\ A(F_1, ..., F_{i-1}, F_i \cap P, F_i, ..., F_k) = A(F_1, ..., F_{i-1}, \emptyset, F_i, ..., F_k)$, then for $x \in \mathcal{X}$ on device d, Alistair satisfies individual device-epoch ϵ_d^G-DP for x under public information P.* **Case 2:** *For general attribution functions, Alistair satisfies individual device-epoch $2\epsilon_d^G$-DP for x under public information P.*

Intuitively, the information gained on cross-site (private to the querier) events in device-epoch x under the querier's queries is bounded by ϵ_x^G (or $2\epsilon_x^G$ without query constraints).

Theorem 2 (**Unlinkability guarantee).** *Fix budget capacities $(\epsilon_d^G)_{d \in \mathcal{D}}$. Take any $d_0, d_1 \in \mathcal{D}$, $e \in \mathcal{E}$, and $F_1 \subset F_0$. Denote $x_0 := (d_0, e, F_0), x_1 := (d_1, e, F_1), x_2 := (d_0, e, F_0 \setminus F_1) \in \mathcal{X}$. Take any $D \in \mathbb{D}$ such that $(d_0, e) \notin D$ and $(d_1, e) \notin D$, and any instantiation $\mathcal{M}$ of Alistair. For all $S \subset Range(\mathcal{A})$ we have:* $\Pr[\mathcal{M}(D + x_0) \in S] = e^{2\epsilon_{d_0}^G + \epsilon_{d_1}^G} \Pr[\mathcal{M}(D + x_1 + x_2) \in S]$.

Intuitively, linking a set of events across two devices—compared to detecting these events on one device—is only made easier by the amount of budget on the second device; Alistair does not introduce additional privacy loss for linkability, above what is revealed through DP queries.

6 IDP Optimizations

IDP offers the opportunity to discount DP budget based on individual sensitivity, which is no greater, and often smaller, than global sensitivity. The clearest way to size this opportunity is to compare the global sensitivity of reports and queries with their individual sensitivity. Recall that Alistair programmatically enforces a bound on the reports by capping the value of each coordinate in the output of the attribution function to a querier-provided maximum value. Given this enforced cap, we prove the following formulas for the two types of sensitivities (proofs in Appendix §C):

Theorem 3 (**Global sensitivity of reports and queries).** *Fix a report identifier r, a device d_r, a set of epochs E_r, an attribution function A and the corresponding report $\rho : D \mapsto A(D_{d_r}^{E_r})$. We have:*

$$\Delta(\rho) = \max_{i \in [k], F_1, ..., F_k \in \mathcal{P}(\mathcal{I} \cup \mathcal{C})} \|A(F_1, ..., F_k) - A(F_1, ..., F_{i-1}, \emptyset, F_{i+1}, ..., F_k)\|_1$$

Next, fix a query Q with reports $(\rho_r)_{r \in R}$ such that each device-epoch participates in at most one report. We have $\Delta(Q) = \max_{r \in R} \Delta(\rho_r)$.

Theorem 4 (Individual sensitivity of reports and queries). *Fix a device-epoch record $x = (d, e, F) \in \mathcal{X}$. Fix a report identifier r, a device d_r, a set of epochs $E_r = \{e_1, \ldots, e_k\}$, an attribution function A with relevant events F_A, and the corresponding report $\rho : D \mapsto A(D_{d_r}^{E_r})$.*

We have: $\Delta_x(\rho) = \max_{F_1,\ldots,F_{i-1},F_{i+1},\ldots,F_k \in \mathcal{P}(\mathcal{I} \cup \mathcal{C})} \|A(F_1, ..., F_{i-1}, F, F_{i+1}, ..., F_k) - A(F_1, ..., F_{i-1}, \emptyset, F_{i+1}, ..., F_k)\|_1$ *if* $d = d_r$ *and* $e = e_i \in E_r$, *and* $\Delta_x(\rho) = 0$ *otherwise.*

In particular,

$$\Delta_x(\rho) \leq \begin{cases} \|A(F) - A(\emptyset)\|_1 & \text{if } d = d_r \text{ and } E_r = \{e\} \\ \Delta(\rho) & \text{if } d = d_r, e \in E_r \text{ and } F \cap F_A \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$$

Next, fix a query Q with reports $(\rho_r)_{r \in R}$ such that x participates in at most one report ρ_r. We have $\Delta_x(Q) = \Delta_x(\rho_r)$.

This theorem justifies both the inherent optimization inadvertently applied by all on-device systems, and new optimizations that we implement in Alistair.

Inherent on-device optimization. The condition $d = d_r$ in Thm. 4 justifies under IDP the behavior of on-device systems of deducting privacy loss for a query only on devices that participate in the query. This is much more efficient compared to off-device systems such as IPA, which operate under traditional DP and therefore use global sensitivity. Indeed, Thm. 3 shows that under DP, these systems must deduct budget based on $\Delta(Q)$ from *all devices*, regardless of query participation.

Examples of new optimizations. First, a device that participates in a query but has no data relevant to the query (*i.e.,* $F \cap F_A = \emptyset$ or $A(F) = A(\emptyset)$ in Thm. 4) need not pay budget. This is why in the example from § 3.2, we do not deduct from epoch e_3 that lacks impressions of the Nike ad. Second, a device's individual sensitivity only depends on the reports it participates in ($\Delta_x(Q) = \Delta_x(\rho_r)$ in Thm. 4), while the global sensitivity depends on other reports in a query ($\Delta(Q) = \max_{r \in R} \Delta(\rho_r)$ in Thm. 3). Since the report ρ typically depends on the public information $F \cap P$ of a record (d, e, F), this optimization lets us use a cap of \$70 instead of \$100 in the Nike example. Third, if an attribution takes only one epoch (or is decomposed into a sum of single-epoch reports), then the individual sensitivity can be further lowered depending on the private information F of a record. Suppose Nike wants to measure the average impression-to-conversion delay, where the delay is between 0 and 30 days. If a record x contains one impression that occurred only 1 day before the conversion, the resulting individual budget will be 1/30th of the budget obtained with global sensitivity.

7 Chrome Prototype

We integrated Alistair into Google Chrome and based the implementation on the existing support for ARA. We disabled ARA's own impression-level budgeting, added support for epochs and extended Chrome's SQL database with a new table to store a privacy filter for every unique pair of epoch and querier. ARA supports only the last-touch attribution policy allowing it to fetch only the latest impression from the database. We fetch all impressions related to the conversion instead and group them by epoch to identify epochs that are "empty" and therefore are not enforced to consume budget. We implemented Alistair's IDP based optimizations, and added support for visualizing privacy loss (see Fig. 1).

8 Evaluation

Our evaluation answers the following questions:

- **Q1:** How do individual-sensitivity optimizations impact privacy budget consumption?
- **Q2:** Does hiding privacy budgets affect query accuracy?
- **Q3:** How does Alistair perform across workloads?
- **Q4:** What is Alistair's performance overhead?

8.1 Methodology

Microbenchmark dataset. We develop a synthetic dataset to methodically assess Alistair's effectiveness under different scenarios. The dataset has two knobs: (a) the fraction of users who converted per query, and (b) their ad exposure rate. It contains 40,000 conversions tied to a single advertiser and evenly dispersed across 10 products, across 120 days.

PATCG dataset. This dataset was developed by PATCG [33]. It consists of 24M conversions from a single advertiser across 30 days. Only 1% of conversions are attributed to an impression. On average, each user is exposed to an ad 3.2 times, while users who convert take part in 1.5 conversions.

Criteo dataset. This dataset has 12M records sourced from a subset of 90-day logs recording live traffic data from Criteo [42]. It contains 1.3M conversions, 10M distinct users and 292 unique advertisers. Criteo's dataset has two significant limitations. First, all conversions are linked to impressions, which is not realistic, although it does not directly impact our evaluation. Second, the dataset is heavily subsampled, which means many impressions for each user are likely missing. This favors Alistair, as it amplifies the effect of its optimizations. Thus, we also evaluate on a version of Criteo's dataset augmented with extra synthetic impressions.

Evaluation Scenario. We construct the evaluation based on the scenario described in §2.1, but with an exclusive focus on per-advertiser queries; we plan to extend our evaluation to cover ad-tech queries as well in the future. We assume an advertiser, call it Nike, runs an ad campaign for 10 of its products over 4 months. Nike measures the efficacy of the campaign by measuring how many purchases of each product were attributed to it. Every time a customer purchases a quantity C of a product Nike requests a report for this conversion. Nike's measurement by default runs over the last 30 days and determines which ad was last seen by the user before the conversion ("last-touch"). Upon conversion, the user's browser generates a report using the impression with the most recent timestamp. If no relevant impressions exist then the report will be 0 otherwise its value is C. Nike

generates two queries for each of the 10 products (20 batches of reports). We use the following default parameters across all workloads: initial budget is 1; epoch size is 7 days.

Accuracy. Each batch is forwarded to the aggregation service, and the aggregated results, with added noise using the Laplace mechanism, are sent back to the advertiser. Nike selects the noise-scale of the noise distribution so each of the 20 queries independently will have a target accuracy of 95% with a 99% probability. This configuration ignores the effect of null reports that occur when filters run out of budget during the experiment. The advertiser computes the privacy budget for each query that achieves this target accuracy by controlling the size of each batch sent for aggregation, and by estimating the expected attribution rate. For Alistair, Nike sends this requested budget to the individual devices.

Baselines. We compare Alistair to two baselines. The first is IPA-like which employs the centralized budgeting and query execution components of IPA described in §2. The second is ARA-like, which employs a similar on-device budgeting to ARA but provides device-epoch instead of impression level IDP guarantees and does not incorporate any IDP-based optimizations.

8.2 Microbenchmark evaluation (Q1)

Varying the user population (knob1). We use the microbenchmark to evaluate Alistair across different configurations. We first vary the user participation rate per query (knob1). With a default batch size of 2,000 reports and 10 products, each queried twice, we create 40,000 conversions. This knob controls how we assign those conversions to users, so it implicitly controls the number of users in our workload, or population size. A large population size favors on-device budgeting systems like Alistair because it increases the number of privacy filters that are available to an advertiser. For a knob1 value of 1, each user participates in all 20 batches once, so the minimum required population size is 2,000, while a value of 0.001 creates 2M users. For reference, in the PATCG dataset users who convert do so with a 0.05 daily rate, corresponding to a default value of 0.1 for knob1.

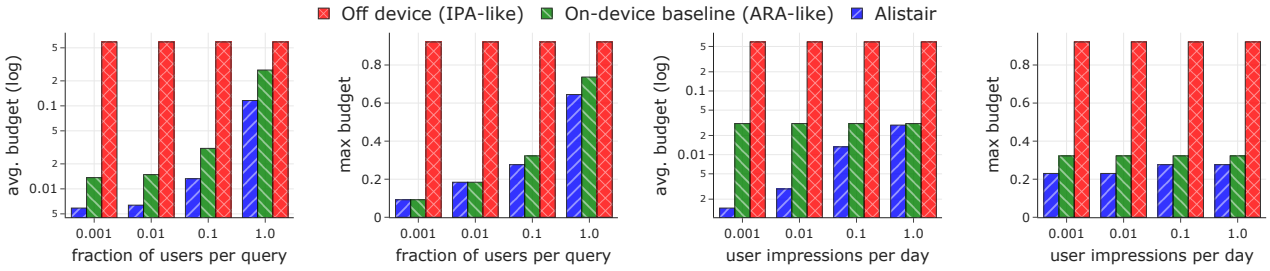
Fig. 4a and 4b show the average and maximum budget Nike consumed across all device-epochs they ever requested, respectively. Qualitatively, the average budget consumption is a much more useful metric to assess the efficiency of the three systems, but we include the maximum because it reduces IDP guarantees to standard DP guarantees, thereby providing a more mathematically-rigorous comparison between on-device and off-device budgeting. Recall that IPA-like does not distribute budget consumption across the various participating devices but has a centralized privacy filter from which it deducts budget upon executing each query. Thus, increasing the user population size does not impact its budget consumption, which is always higher than the other methods. Alistair systematically consumes the least amount of budget due to its optimizations. The optimizations provide a relatively larger improvement when the population size is higher (lower knob1), because there are more epochs that do not contain impressions relevant to conversions. Even under the conservative max budget consumption metric, the on-device budgeting systems outperform IPA-like, and Alistair remains the most effective.

Varying the number of impressions per user (knob2). We now fix knob1 to its default value and vary the number of impressions of a user per day (knob2). For reference, by default in PATCG, knob2 is 0.1. Fig. 4c and 4d present the results. As before, Alistair's optimizations provide the most benefit with a small number of impressions per user.

8.3 PATCG evaluation (Q1, Q2, Q3)

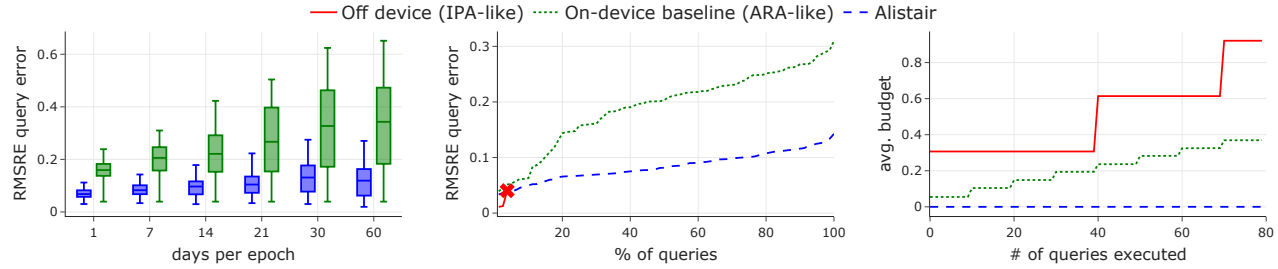
Next, we analyze how Alistair's optimizations affect budget consumption and query accuracy on the PATCG dataset. Each impression and conversion in this dataset is linked to a set of attributes, with values randomly sampled from various distributions. We focus on a conversion-side attribute named 'conv-attribute-2', whose values are uniformly sampled from 0 to 9. These values could represent 10 distinct products potentially sold by Nike. Nike queries each product 8 times, resulting in a total of 80 queries with batch sizes ranging from 280,000 to 303,009 reports. We opt for large batch sizes, considering most reports within a batch will likely return 0 due to the low attribution rate (1%). In Fig. 5a, we run the workload with varying epoch lengths and plot the distribution of query errors, using the RMSRE metric, defined by $\sqrt{\mathbb{E}[(\mathcal{M}(D) - Q(D))^2/Q(D)^2]}$ for an estimate $\mathcal{M}(D)$ of the query output $Q(D)$. In the graph, a horizontal line represents the mean, the filled rectangle is standard deviation, and the dots are the minimum and maximum. Unlike the on-device budgeting systems that need to conceal their budgets, IPA-like doesn't run the query when the requested privacy resources are depleted. Instead, it produces an "out-of-budget" error message to notify the advertiser. IPA-like is only able to execute at most 3.75% of the queries across all runs, hence we omit it from the figure. IPA-like's accuracy degrades as a function of epoch length, because larger epochs lead to fewer available privacy filters. The two on-device budgeting methods always provide a query output to the advertiser albeit by silently inducing bias. Alistair conserves budget more efficiently due to the optimizations resulting in fewer null reports and less compromised accuracy compared to ARA-like. For these two systems accuracy also worsens as a function of epoch length for the same reason.

In Fig. 5b and 5c we run the workload with the default epoch of 7 days. The RMSRE distribution shows that Alistair has systematically lower error than ARA-like, since it conserves budget and thus drops fewer records, resulting in lower bias. The average budget consumption over time shows that Alistair consumes budget much more conservatively. The "staircase" behavior of IPA-like is attributed to the shifting of



(a) Avg., knob1=0.1 **(b)** Max, knob1=0.1 **(c)** Avg., knob2=0.1 **(d)** Max, knob2=0.1

Fig. 4. Budget consumption on the microbenchmark: (a) and (b) show average and maximum budget consumption across all device-epochs as a function of the fraction of users that participate per query, respectively; c) and (d) show the same metrics as a function of user impressions per day.



(a) RMSRE as a function of epoch length. **(b)** CDF of RMSRE. **(c)** Avg. budget consumed across all device-epochs.

Fig. 5. Query accuracy and budget consumption on the PATCG dataset: (a) distribution of RMSREs with varying epoch length; (b) CDF of RMSREs with epoch of 7 days; (c) average budget consumption across all device-epochs as a function of the number of queries executed.

the attribution window across time. This results in the system requesting new epochs whose privacy filters have not yet been depleted.

8.4 Criteo Evaluation (Q1, Q2, Q3)

We now run on the Criteo dataset, which has 1.3M conversions, unevenly distributed across 274 advertisers. A challenge with this dataset is to generate queries with sufficiently-large batch sizes that cover as many advertisers as possible. We set the minimum batch size to 350 reports and the maximum size for aggregation to 400 reports. Any reports received beyond this maximum form a new batch, potentially leading to query re-execution. Focusing on the conversion-side attribute "product-category-3", we create a pool of 898 queries spanning 109 advertisers. The remaining advertisers cannot generate queries with batches reaching the minimum size.

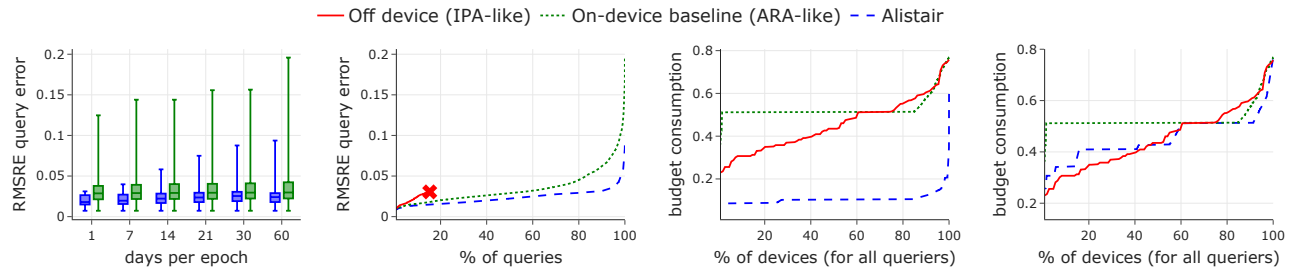
Fig. 6a shows RMSRE when varying epoch length. Once again we observe that as the epoch length increases so does the bias induced by the on-device budgeting systems. IPA-like performs poorly yet again, and runs only 16.59% of the queries, hence we omit it for readability. Fig. 6b plots the CDF of the query error with an epoch of 7 days. While Alistair outperforms ARA-like their gap is smaller than that observed in the PATCG dataset. In Criteo the vast majority of users convert only once and so there is no "contention" within each device's privacy filters.

In Fig. 6c we plot the CDF of the average budget consumed for all the device-epochs. The figure shows ARA-like has utilized a significant portion of its filters' privacy budget, indicating improved utilization compared to IPA-like which rejected most of the queries. Meanwhile, Alistair achieves significantly lower budget consumption compared to ARA-like while achieving higher accuracy. As anticipated, Criteo's extensive subsampling favors Alistair.

Next, we augment Criteo by injecting 9 additional synthetic impressions for every conversion within the conversion's attribution window. The resulting CDF of budget consumption is shown in Fig. 6d. We observe that Alistair performs significantly worse. This is because with epochs of 7 days and an attribution window of 30 days, covering 5 epochs on average, it attempts to consume budget from most of the requested epochs, since most will contain relevant impressions. As expected, the performance of the other systems is invariant to the change we made. Despite the increase in the budget consumed by Alistair we observe that the error CDF has no visible difference compared to Fig. 6b. Thereby, we omit showing the figure again. Alistair achieves high accuracy due to the optimization that allows epochs depleted of budget to abstain from creating the report rather than being forced to nullify it. Therefore, as long as the impression with the latest timestamp resides in an epoch that is not yet depleted, which is commonly the case for the most recent epochs, then Alistair does not induce bias.

8.5 Performance Overhead (Q4)

Next, we measure the performance overhead of Alistair compared to Google's ARA in our Chrome prototype. As mentioned in §7, Alistair needs to iterate over all the impressions relevant to the conversion, so that it can infer which of the querier's privacy filters will consume budget. In contrast,



(a) RMSRE as function of epoch length. **(b)** CDF of RMSRE. **(c)** CDF of budget on Criteo. **(d)** CDF of budget on Criteo++.

Fig. 6. Query accuracy and budget consumption on Criteo: (a) distribution of RMSREs with varying epoch length; (b) CDF of RMSREs for epoch of 7 days; (c) and (d) are the CDF of average budget consumption across all devices for the original and augmented Criteo datasets, respectively.

ARA tracks only the impression with the latest timestamp. We compare two versions of Chrome that run Alistair and ARA, respectively. We use Selenium [19] to make them interact with a publisher and generate impressions, which are distributed randomly across 20 epochs. We vary the number of impressions injected into the browsers from 10 to 100 and we measure the time it takes to create a report upon triggering a conversion. As expected, ARA reports always a constant time of 5.4ms while Alistair's reporting time increases linearly from 9.1ms to 57.3ms as a function of the number of impressions it iterates over. Note this is an obvious side channel that would need to be made constant time to avoid revealing whether impressions relevant to a conversion were found on the device.

9 Related Work

DP systems. Most DP systems operate in the central model, where a trusted curator runs queries, typically using global sensitivity [12]. Some use fine-grained accounting through parallel composition [30, 27, 28, 26], a coarse form of IDP that does not provide optimizations akin to Alistair. Others operate in the local DP model, where each device randomizes its data locally [24]. Such systems use on-device budgeting natively, but have a higher utility cost. There are distributed systems emulating the central model due to cryptographic constructions. [40, 29], as IPA, maintain a single privacy filter, not leveraging IDP to conserve budget. [5] uses the shuffle model [7] to combine local randomization with a minimal trusted party. Alistair operates in the central model with on device budgeting, enabling new optimizations.

Private ads measurement. Several private ad measurement system proposals exist. Apple's PCM [22] relies on entropy limits for privacy. Meta and Mozilla's IPA [6] uses centralized budgeting while Google's ARA [4] and Apple's PAM [36] use on device budgeting. ARA has been mostly studied to optimize in-query budget and utility. [10] optimizes a single vector-valued hierarchical query while [1] assumes a simplified version of ARA with off-device impression-level DP guarantees, and attempts to efficiently bound each impression's contribution for known-upfront queries (not online queries). [11] provides a framework for attribution logic and DP neighborhood relations, and proposes clipping strategies that yield bounds on global sensitivity. Meanwhile, we optimize on-device budgeting across queries and use tighter individual sensitivity bounds. Our work is agnostic to how the sensitivity bounds are enforced, and could therefore benefit from the clipping algorithms from [10, 1, 11].

IDP. IDP was introduced in the central setting [13, 23, 14], where a trusted curator maintains individual budgets and decides which data points to query. Individual sensitivity was used to optimize SQL-like queries and gradient descent. All literature notes that individual budgets must be kept private, and [45] studies the release of DP aggregates computed over the budgets. [13] notes that out-of-budget records must be dropped silently and leaves bias analysis for future work.

References

[1] Hidayet Aksu et al. *Summary Reports Optimization in the Privacy Sandbox Attribution Reporting API*. Nov. 22, 2023. arXiv: 2311.13586 [cs].

[2] Inc. Apple. *Apple announces powerful new privacy and security features*. https://www.apple.com/newsroom/2023/06/apple-announces-powerful-new-privacy-and-security-features/. 2023.

[3] *Apple announces powerful new privacy and security features*. https://blog.mozilla.org/en/mozilla/understanding-apples-private-click-measurement/. 2022.

[4] *Attribution Reporting API (ARA))*. https://github.com/WICG/attribution-reporting-api/blob/main/AGGREGATE.md. 2022.

[5] Andrea Bittau et al. "Prochlo: Strong Privacy for Analytics in the Crowd". In: *Proceedings of the 26th Symposium on Operating Systems Principles*. SOSP '17. Shanghai, China: Association for Computing Machinery, 2017, pp. 441–459. ISBN: 9781450350853. DOI: 10.1145/3132747.3132769. URL: https://doi.org/10.1145/3132747.3132769.

[6] Benjamin Case et al. *Interoperable Private Attribution: A Distributed Attribution and Aggregation Protocol*. Cryptology ePrint Archive, Paper 2023/437. https://eprint.iacr.org/2023/437. 2023. URL: https://eprint.iacr.org/2023/437.

[7] Albert Cheu et al. "Distributed Differential Privacy via Shuffling". In: *Advances in Cryptology – EUROCRYPT 2019*. Ed. by Yuval Ishai and Vincent Rijmen. Cham: Springer International Publishing, 2019, pp. 375–403. ISBN: 978-3-030-17653-2.

[8] Google Chrome. *Federated Learning of Cohorts (FLoC)*. https://privacysandbox.com/proposals/floc/.

[9] Google Chrome. *Protected Audience API overview*. https://developers.google.com/privacy-sandbox/relevance/protected-audience.

[10] Matthew Dawson et al. *Optimizing Hierarchical Queries for the Attribution Reporting API*. Comment: Appeared at AdKDD 2023 workshop; Final proceedings version. Nov. 27, 2023. arXiv: 2308.13510 [cs].

[11] John Delaney et al. *Differentially Private Ad Conversion Measurement*. 2024. arXiv: 2403.15224 [cs.CR].

[12] Cynthia Dwork and Aaron Roth. "The Algorithmic Foundations of Differential Privacy". In: *Foundations and Trends® in Theoretical Computer Science* 9.3-4 (2013), pp. 211–407. ISSN: 1551-305X, 1551-3068. DOI: 10.1561/0400000042.

[13] Hamid Ebadi, David Sands, and Gerardo Schneider. "Differential Privacy: Now It's Getting Personal". In: *Proceedings of the 42nd Annual ACM SIGPLAN - SIGACT Symposium on Principles of Programming Languages*. POPL '15: The 42nd Annual ACM SIGPLAN SIGACT Symposium on Principles of Programming Languages. Mumbai India: ACM, Jan. 14, 2015, pp. 69–81. ISBN: 978-1-4503-3300-9. DOI: 10.1145/2676726.2677005.

[14] Vitaly Feldman and Tijana Zrnic. "Individual Privacy Accounting via a Rényi Filter". In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato et al. Vol. 34. Curran Associates, Inc., 2021, pp. 28080–28091.

[15] Badih Ghazi et al. "Deep Learning with Label Differential Privacy". In: *Advances in Neural Information Processing Systems*. Vol. 34. Curran Associates, Inc., 2021, pp. 27131–27145.

[16] *Google's FLoC Is a Terrible Idea*. https://www.eff.org/deeplinks/2021/03/googles-floc-terrible-idea. 2021.

[17] *Hybrid Proposal*. https://github.com/patcg-individual-drafts/hybrid-proposal. 2024.

[18] *iCloud Private Relay Overview*. https://www.apple.com/icloud/docs/iCloud_Private_Relay_Overview_Dec2021.pdf. 2021.

[19] *iCloud Private Relay Overview*. https://www.selenium.dev/. 2024.

[20] *Intelligent Tracking Prevention 2.3*. https://webkit.org/blog/9521/intelligent-tracking-prevention-2-3/. 2019.

[21] *Interoperable Private Attribution (IPA))*. https://github.com/patcg-individual-drafts/ipa. 2022.

[22] *Introducing Private Click Measurement, PCM*. https://webkit.org/blog/11529/introducing-private-click-measurement-pcm/. 2021.

[23] Zach Jorgensen, Ting Yu, and Graham Cormode. "Conservative or liberal? Personalized differential privacy". In: *2015 IEEE 31st International Conference on Data Engineering*. 2015, pp. 1023–1034. DOI: 10.1109/ICDE.2015.7113353.

[24] Shiva Prasad Kasiviswanathan et al. "What Can We Learn Privately?" In: *SIAM Journal on Computing* 40.3 (2011), pp. 793–826. DOI: 10.1137/090756090. URL: https://doi.org/10.1137/090756090.

[25] Daniel Kifer et al. *Guidelines for Implementing and Auditing Differentially Private Systems*. Tech. rep. 2020.

[26] Nicolas Küchler et al. "Cohere: Privacy Management in Large Scale Systems". In: *CoRR* abs/2301.08517 (2023). DOI: 10.48550/ARXIV.2301.08517. arXiv: 2301.08517. URL: https://doi.org/10.48550/arXiv.2301.08517.

[27] Mathias Lecuyer et al. "Privacy Accounting and Quality Control in the Sage Differentially Private Machine Learning Platform". In: *Proceedings of the ACM Symposium on Operating Systems Principles (SOSP)*. 2019.

[28] Tao Luo et al. "Privacy Budget Scheduling". In: *15th USENIX Symposium on Operating Systems Design*

and Implementation (OSDI 21). USENIX Association, July 2021, pp. 55–74. ISBN: 978-1-939133-22-9. URL: https://www.usenix.org/conference/osdi21/presentation/luo.

[29] Elizabeth Margolin et al. "Arboretum: A Planner for Large-Scale Federated Analytics with Differential Privacy". In: *Proceedings of the 29th Symposium on Operating Systems Principles*. SOSP '23. , Koblenz, Germany, Association for Computing Machinery, 2023, pp. 451–465. ISBN: 9798400702297. DOI: 10.1145/3600006.3624566. URL: https://doi.org/10.1145/3600006.3624566.

[30] Frank D. McSherry. "Privacy Integrated Queries: An Extensible Platform for Privacy-Preserving Data Analysis". In: *Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data*. SIGMOD '09. New York, NY, USA: Association for Computing Machinery, June 29, 2009, pp. 19–30. ISBN: 978-1-60558-551-2. DOI: 10.1145/1559845.1559850.

[31] Mozilla. *Over a decade of anti-tracking work at Mozilla*. https://blog.mozilla.org/en/privacy-security/mozilla-anti-tracking-milestones-timeline/. 2022.

[32] *Over a decade of anti-tracking work at Mozilla*. https://blog.mozilla.org/en/privacy-security/mozilla-anti-tracking-milestones-timeline/. 2022.

[33] *PATCG Attribution Synthetic Data*. https://docs.google.com/document/d/1Vxq4LrMe3A2WIlu-7IYP1Hycr_nz3_qTpPAICX9fLcw. 2024.

[34] *Preparing for the end of third-party cookies*. https://developers.google.com/privacy-sandbox/blog/cookie-countdown-2023oct. 2023.

[35] *Privacy Preserving Ad Click Attribution For the Web*. https://webkit.org/blog/8943/privacy-preserving-ad-click-attribution-for-the-web/. 2019.

[36] *Private Ad Measurement (PAM)*. https://github.com/patcg-individual-drafts/private-ad-measurement. 2023.

[37] *Private Advertising Technology Community Group*. https://www.w3.org/community/patcg. 2024.

[38] Ryan Rogers et al. "Privacy odometers and filters: pay-as-you-go composition". In: *Proceedings of the 30th International Conference on Neural Information Processing Systems*. NIPS'16. Barcelona, Spain: Curran Associates Inc., 2016, pp. 1929–1937. ISBN: 9781510838819.

[39] Ryan M Rogers et al. "Privacy Odometers and Filters: Pay-as-you-go Composition". In: *Advances in Neural Information Processing Systems*. Ed. by D. Lee et al. Vol. 29. Curran Associates, Inc., 2016.

[40] Edo Roth et al. "Orchard: differentially private analytics at scale". In: *Proceedings of the 14th USENIX Conference on Operating Systems Design and Implementation*. OSDI'20. USA: USENIX Association, 2020. ISBN: 978-1-939133-19-9.

[41] Google Privacy Sandbox. *Privacy Sandbox for the Web*. https://privacysandbox.com/intl/en_us/open-web. 2023.

[42] Marcelo Tallis and Pranjul Yadav. "Reacting to Variations in Product Demand: An Application for Conversion Rate (CR) Prediction in Sponsored Search". In: *arXiv preprint arXiv:1806.08211* (2018).

[43] Salil Vadhan and Wanrong Zhang. "Concurrent Composition Theorems for Differential Privacy". In: *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*. STOC 2023. Orlando, FL, USA: Association for Computing Machinery, 2023, pp. 507–519. ISBN: 9781450399135. DOI: 10.1145/3564246.3585241. URL: https://doi.org/10.1145/3564246.3585241.

[44] *Weighted Aggregate Logistic Regression*. https://github.com/patcg-individual-drafts/ipa/blob/main/logistic_regression.md. 2024.

[45] Da Yu et al. "Individual Privacy Accounting for Differentially Private Stochastic Gradient Descent". In: *Transactions on Machine Learning Research* (Apr. 27, 2023). ISSN: 2835-8856.

A Alistair Algorithm

Algorithm 1 Alistair Algorithm

Config
Public events $P \subset \mathcal{I} \cup \mathcal{C}$
Parametrized noise distribution $\mathcal{L}$
Device-epoch budget capacity $(\epsilon_x^G)_{x \in \mathcal{X}}$
Input
Database D
Stream of interactively chosen queries $Q_1, \ldots, Q_k$
function Main($D, Q_1, \ldots, Q_k$)
$S = \emptyset$
for $(d, e, F) \in D$ **do**
for $f \in F \cap P$ **do**
Generate report identifier $r \xleftarrow{\$} U(\mathbb{Z})$
Save mapping from r to the device that generated it: $d_r \leftarrow d$
$S \leftarrow S \cup \{(r, f)\}$
output S *// report identifiers and public events* $D \cap P$
for $i \in [k]$ **do**
output AnswerQuery(Q_i)
// Collect, aggregate and noise reports to answer Q_i
function AnswerQuery(report identifiers R, target epochs $(E_r)_{r \in R}$, attribution functions $(A_r)_{r \in R}$ and noise parameter σ)
for $r \in R$ **do**
$\rho_r \leftarrow$ GenerateReport(d_r, E_r, A_r)
Sample $X \sim \mathcal{L}(\sigma)$
return $\sum_{r \in R} \rho_r + X$
// Generate report and update on-device budget
function GenerateReport(d, E, A)
for $e \in E$ **do**
$x \leftarrow (d, e, D_d^e)$
if $\mathcal{F}_x$ is not defined **then**
Initialize filter $\mathcal{F}_x$ with capacity ϵ_x^G
$\epsilon_x \leftarrow$ ComputeIndividualBudget($x, d, E, A, \mathcal{L}, \sigma$)
if $\mathcal{F}_x$.tryConsume(ϵ_x) = *Halt* **then**
$F_e \leftarrow \emptyset$
else
$F_e \leftarrow D_d^e$
$\rho \leftarrow A((F_e)_{e \in E})$ *// Clipped attribution report*
return ρ

Alg. 1 describes the formal view of Alistair, whose privacy guarantees we establish in §5.4. Alistair answers a stream of the querier's queries by generating reports based on a device's data in the queried epochs and an attribution function A passed in the query. It does so while the querier still has available budget. The function GenerateReport in Alg. 1 models this logic of privacy budget checks and consumption, followed by report creation if enough budget is available. The attribution function A has bounded sensitivity (defined in §5.3), enforced through clipping. Function AnswerQuery then sums reports together to compute the final query value. DP noise is added to the result before returning it to the querier (see the output of Alg. 1).

The algorithm captures the fact that reports that do not contribute to a query are not actually generated (the summation is over $r \in R$). This is how all on-device systems inherently work (not only Alistair), and it's an important optimization that preserves privacy budget, as reports that are not generated do not consume budget. Yet, as previously mentioned, it is very non-standard behavior for DP, so its privacy justification, which we do in the next section, requires both the formalization of reports with unique identifiers r and an individual DP framework.

We instantiate the filter methods and the ComputeIndividualBudget function for the Laplace distribution in the next section (§B).

B Proofs of Privacy Guarantees (§5.4)

Filter and budget semantics for Laplace. In this section, we focus on the Laplace noise distribution: $\mathcal{L}(\sigma) = \text{Lap}(\sigma/\sqrt{2})$. We use pure differential privacy accounting, hence the budgets are real numbers $\epsilon > 0$. To track the budget of adaptively chosen queries, we use a Pure DP filter [38]. For a budget capacity ϵ^G, this filter simply adds up the budget consumed by the first k queries, and outputs Halt for the next query with budget ϵ_{k+1} if:

$$\epsilon_1 + \cdots + \epsilon_k + \epsilon_{k+1} > \epsilon^G \tag{3}$$

Finally, for a datapoint x, a report $\rho = (d, E, A)$, the Laplace distribution $\mathcal{L}$ and a standard deviation σ, we have:

$$\text{ComputeIndividualBudget}(x, d, E, A, \mathcal{L}, \sigma) = \frac{\Delta\sqrt{2}}{\sigma} \tag{4}$$

where Δ is an upper bound on the individual sensitivity of the report $\Delta_x(\rho)$. We provide such upper bounds in §6.

Finally, we use a slightly more general way of initializing budget capacities, by setting one capacity for each possible record $(\epsilon_x^G)_{x \in \mathcal{X}}$. In the body of the paper we set the same capacity for all the records belonging to the same device d: $(\epsilon_x^G)_{d \in \mathcal{D}}$. For practical purposes it is enough to set capacities at the device level, but using per-record capacities simplifies certain proofs, such as Thm. 7.

B.1 Individual DP Guarantees (Thm. 1)

To prove Thm. 1 from §5.4, we need to define an intermediary "inner" privacy game Alg. 2, which we analyze in Thm. 5. Next, we define another "outer" privacy game Alg. 3, that is a generalized version of Alg. 1 and internally calls Alg. 2. Finally, Thm. 6 and Thm. 7 imply Thm. 1.

Theorem 5 (IDP of Alg. 2 when removing x)**.** *Fix a device-epoch budget capacity* $(\epsilon_x^G)_{x \in \mathcal{X}}$ *for every possible record*

Algorithm 2 Inner Privacy Game

Config
Parametrized noise distribution $\mathcal{L}$
Device-epoch budget capacity $(\epsilon_x^G)_{x\in\mathcal{X}}$
Upper bound on number of epochs $e_{\max}$
Upper bound on number of queries per epoch $k_{\max}$

Input
Challenge bit $b \in \{0,1\}$
Opt-out device $x_0 = (d_0, e_0, F_0) \in \mathcal{X}$
Adversary $\mathcal{A}$

Output
View $V^b = (v^b_{1,1}, \dots, v^b_{1,k_{\max}}, v^b_{2,1}, \dots)$ of $\mathcal{A}$

$D \leftarrow \emptyset$
for $e \in [e_{\max}]$ **do**
  // Generate data for the epoch e
  Receive a database G for epoch e from $\mathcal{A}$
  if $e = e_0$ and $(d_0, e_0) \notin G$ **then**
    $G^0 \leftarrow G + (d_0, e_0, \emptyset), G^1 \leftarrow G + (d_0, e_0, F_0)$
  else
    $G^b \leftarrow G$
  $D \leftarrow D + G^b$

  // Answer queries after epoch e
  for $k \in [k_{\max}]$ **do**
    Receive query Q_k from $\mathcal{A}$ with corresponding indices R, devices $(d_r)_{r\in R}$, target epochs $(E_r)_{r\in R}$, attribution functions $(A_r)_{r\in R}$ and noise std-dev σ.
    for $r \in R$ **do**
      // Compute report for r
      for $e \in E_r$ **do**
        $x \leftarrow (d_r, e, D^e_{d_r})$
        if $\mathcal{F}_x$ is not defined **then**
          Initialize filter $\mathcal{F}_x$ with capacity ϵ_x^G
        $\epsilon_x \leftarrow \text{ComputeIndividualBudget}(x, d, E, A, \mathcal{L}, \sigma)$
        if $\mathcal{F}_x$.tryConsume$(\epsilon_x) = Halt$ **then**
          $F_e \leftarrow \emptyset$
        else
          $F_e \leftarrow D^e_d$
      $\rho_r \leftarrow A((F_e)_{e\in E})$
    // Aggregate and noise reports to answer Q_k
    Sample $X \sim \mathcal{L}(\sigma)$
    Send $v^b_{e,k} = \sum_{r\in R} \rho_r + X$ to $\mathcal{A}$

$x \in \mathcal{X}$. For any opt-out record $x \in \mathcal{X}$, for any adversary $\mathcal{A}$, and V^0, V^1 defined by Alg. 2, for all $v \in \text{Supp}(V)$ we have:

$$\left|\ln\left(\frac{\Pr[V^0 = v]}{\Pr[V^1 = v]}\right)\right| \le \epsilon_x^G \tag{5}$$

Proof. Fix an upper bound on the number of epochs and queries per epoch $e_{\max}, k_{\max}$. Fix an opt-out record $x = (d_0, e_0, F_0) \in \mathcal{X}$ and an adversary $\mathcal{A}$. Take V^0, V^1 the view of $\mathcal{A}$ in Alg. 2.

Consider a view $v \in \text{Supp}(V^1)$. We have:

$$\ln\left(\frac{\Pr[V^0 = v]}{\Pr[V^1 = v]}\right) = \ln\left(\prod_{e=1}^{e_{\max}}\prod_{k=1}^{k_{\max}} \frac{\Pr[V^0_{e,k} = v_{e,k}|v_{<e,k}]}{\Pr[V^1_{e,k} = v_{e,k}|v_{<e,k}]}\right) \tag{6}$$

where, for $e \in [e_{\max}], k \in [k_{\max}]$, $b \in \{0,1\}$ and $v_{e,k}$ we have:

$$\begin{aligned}&\Pr[V^b_{e,k} = v_{e,k}|v_{<e,k}]\\ &\quad= \Pr[V^b_{e,k} = v_{e,k}|V^b_{1,1} = v_{1,1}, \dots, V^b_{e,k-1} = v_{e,k-1}]\end{aligned}$$

Even though data and query parameters are adaptively chosen, they only depend on the adversary $\mathcal{A}$ (fixed) and its previous views, which are fixed once we condition on $v_{<e,k}$. Take the database ${}^bD^{\le e}$ and the query parameters R, $(\rho_r, d_r, E_r, A_r)_{r\in R}, \sigma$ corresponding to $\mathcal{A}$ conditioned on $v_{<e,k}$. Note ϵ_{x_0} the state (accumulated privacy loss) of $\mathcal{F}_{x_0}$ in the world with $b = 1$ before answering query e, k.

On one hand, if $(d_0, e_0) \notin \{(d_r, e), r \in R, e \in E_r\}$, we observe that for all $r \in R$, ${}^0D^{e_r}_{d_r} = {}^1D^{e_r}_{d_r}$, because ${}^0D^{\le e}$ and ${}^1D^{\le e}$ differ at most on $x_0 = (d_0, e_0, F_0)$. In this case, $\forall r \in R, \rho_r({}^0D^{\le e}) = \rho_r({}^1D^{\le e})$, and hence $\Pr[V^0_{e,k} = v_{e,k}|v_{<e,k}] = \Pr[V^1_{e,k} = v_{e,k}|v_{<e,k}]$.

On the other hand, suppose that we have $r_1, \dots, r_\ell$ (processed in this order) such that for all $i \in [\ell]$ we have $d_{r_i} = d_0, e_0 \in E_{r_i}$.

We pose $\hat{R} \subset R$ the set of reports that do not pass the filter in the world with $b = 1$. (In the world with $b = 0$, the filter for $(d_0, e_0, \emptyset)$ has no effect on $\rho_r({}^0D^{\le e})$ because whether it halts or not we have $F_{e_0} = \emptyset$). For $r \notin \hat{R}$, we have $\rho_r({}^0D^{\le e}) = \rho_r({}^1D^{\le e})$ because both worlds use $F_{e_0} = \emptyset$.

Hence, we have:

$$\begin{aligned}\|\sum_{r\in R}\rho_r({}^0D^{\le e}) - \rho_r({}^1D^{\le e})\|_1 &= \|\sum_{r\in\hat{R}}\rho_r({}^0D^{\le e}) - \rho_r({}^1D^{\le e})\|_1\\ &\le \sum_{r\in\hat{R}}\Delta_x\rho_r\end{aligned} \tag{7}$$

since ${}^0D^{\le e}$ and ${}^1D^{\le e}$ differ at most on $x = (d_0, e_0, F_0)$.
Take $X^0 \sim X^1 \sim \text{Lap}(b)$ with $b = \sigma/\sqrt{2}$. We have:

$$\frac{\Pr[V^0_{e,k} = v_{e,k}|v_{<e,k}]}{\Pr[V^1_{e,k} = v_{e,k}|v_{<e,k}]} = \frac{\Pr[\sum_{r\in R}\rho_r({}^0D^{\le e}) + X^0 = v_{e,k}]}{\Pr[\sum_{r\in R}\rho_r({}^1D^{\le e}) + X^1 = v_{e,k}]} \tag{8}$$

By property of the Laplace distribution, combining Eq. 7 and Eq. 8 gives:

$$\left|\frac{\Pr[V^0_{e,k} = v_{e,k}|v_{<e,k}]}{\Pr[V^1_{e,k} = v_{e,k}|v_{<e,k}]}\right| \le \sum_{r\in\hat{R}}\Delta_x\rho_r/b \tag{9}$$

By definition of ComputeIndividualBudget, we have $\epsilon_r = \Gamma_{x,r}/b$ where $\Delta_x\rho_r \le \Gamma_{x,r}$. Thus, we get $\sum_{r\in\hat{R}}\Delta_x\rho_r/b \le \sum_{r\in\hat{R}}\epsilon_r$.

Taking the sum over all queries, we get:

$$|\ln\left(\frac{\Pr[V^0=v]}{\Pr[V^1=v]}\right)| \leq \sum_{e=1}^{e_{\max}}\sum_{k=1}^{k_{\max}}\sum_{r\in\hat{R}_{e,k}}\epsilon_r \tag{10}$$

$$\leq \epsilon_x^G \tag{11}$$

where Eq. 11 is by definition of a Pure DP filter. □

Theorem 6 (IDP of Alg. 3 when replacing x_0 by x_1 for fixed public information)**.** *Fix a device-epoch budget capacity $(\epsilon_x^G)_{x\in\mathcal{X}}$ for every possible record $x \in \mathcal{X}$. Fix a set of public events $P \subset \mathcal{I} \cup \mathcal{C}$.*

For any pair of records $x_0 = (d_0, e_0, F_0), x_1 = (d_1, e_1, F_1) \in \mathcal{X}$ such that $e_0 = e_1$ and $F_0 \cap P = F_1 \cap P$, for any adversary $\mathcal{B}$, and W^0, W^1 defined by Alg. 3, for all $w \in \mathrm{Supp}(W^1)$ we have:

$$\left|\ln\left(\frac{\Pr[W^0=v]}{\Pr[W^1=v]}\right)\right| \leq \epsilon_{x_0}^G + \epsilon_{x_1}^G \tag{12}$$

Proof. Fix an upper bound on the number of epochs and queries per epoch $e_{\max}, k_{\max}$. Take a record pair $x_0, x_1 \in \mathcal{X}$, an adversary $\mathcal{B}$, W^0, W^1 defined by Alg. 3 and $w \in \mathrm{Supp}(W^1)$. We define $v := (w_{1,1}, \ldots, w_{1,k_{\max}}, w_{2,1}, \ldots, w_{e_{\max},k_{\max}}$ the truncated version of the view w without nonce information (steps with $k = 0$).

We have:

$$\ln\left(\frac{\Pr[W^0=w]}{\Pr[W^1=w]}\right) = \ln\left(\prod_{e=1}^{e_{\max}}\prod_{k=1}^{k_{\max}}\frac{\Pr[W^0_{e,k}=w_{e,k}|w_{<e,k}]}{\Pr[W^1_{e,k}=w_{e,k}|w_{<e,k}]}\right) + \ln\left(\prod_{e=1}^{e_{\max}}\frac{\Pr[W^0_{e,0}=w_{e,0}|v_{<e,0}]}{\Pr[W^1_{e,0}=w_{e,0}|w_{<e,0}]}\right) \tag{13}$$

Take $e \in [e_{\max}], k \in [k_{\max}], c \in \{0,1\}$. Take the database ${}^cD^{\leq e}$ corresponding to $\mathcal{B}$ conditioned on $w_{<e,k}$. $\mathcal{B}$ receives two types of results:

- If $k = 0$, $W^c_{e,k}$ is about nonces and public events. We denote by Z the random variable that returns $\{(U_f, f), f \in F\}$ with i.i.d. $U_f \sim \mathcal{U}(\mathbb{Z})$. Since $F_0 \cap P = F_1 \cap P$, we have:

$$\Pr[W^0_{e,k}=w_{e,k}|w_{<e,k}] = \Pr[Z=w_{e,k}] = \Pr[W^1_{e,k}=w_{e,k}|w_{<e,k}] \tag{14}$$

- For $k > 0$, $W^c_{e,k}$ is the noisy answer to a query. In Alg. 3, we instantiate $\mathcal{A}$ as a valid adversary for Alg. 2 with opt-out record x_c and challenge bit $b = 1$ (*i.e.,* x_c is included in the database). We denote by $(V_{x_c})^1_{e,k}$ the view of this adversary $\mathcal{A}$, and by definition of the truncated view v, we have:

$$\Pr[W^c_{e,k}=w_{e,k}|w_{<e,k}] = \Pr[(V_{x_c})^1_{e,k}=v_{e,k}|v_{<e,k}] \tag{15}$$

Thanks to Eq. 14 and Eq. 15, Eq. 13 becomes:

Algorithm 3 Outer Privacy Game

Config

Parametrized noise distribution $\mathcal{L}$
Device-epoch budget capacity $(\epsilon_x^G)_{x\in\mathcal{X}}$
Upper bound on number of epochs $e_{\max}$
Upper bound on number of queries per epoch $k_{\max}$
Public events $P \subset \mathcal{I} \cup \mathcal{C}$

Input

Pair of records $x_0 = (d_0, e_0, F_0), x_1 = (d_1, e_1, F_1) \in \mathcal{X}$ such that $e_0 = e_1$ and $F_0 \cap P = F_1 \cap P$
Challenge bit c
Adversary $\mathcal{B}$

Output

View $W^c = (w^c_{1,0}, w^c_{1,1}, \ldots, w^c_{1,k_{\max}}, w^c_{2,0}, w^c_{2,1}, \ldots)$ of $\mathcal{B}$

Initialize Alg. 2 with same configuration, challenge bit $b = 1$, opt-out device x^c and adversary $\mathcal{A}$ (whose behavior is defined next)
for $e \in [e_{\max}]$ **do**
  // Generate data for the epoch e
  Receive a database G for epoch e from $\mathcal{B}$
  Ask $\mathcal{A}$ to submit G
  if $e = e_0$ and $(d_0, e_0) \notin G$ and $(d_1, e_1) \notin G$ **then**
    // At this point, $\mathcal{A}$ also adds x_c in his own game
    $G^c \leftarrow G + x_c$
  else
    $G^c \leftarrow G$
  // Release public information
  $S = \emptyset$
  for $(d, e, F) \in G^c$ **do**
    for $f \in F \cap P$ **do**
      Generate report nonce $r \xleftarrow{\$} U(\mathbb{Z})$
      Save device corresponding to nonce $d_r \leftarrow d$
      $S \leftarrow S \cup \{(r, f)\}$
  Send $w^c_{e,0} = S$ to $\mathcal{B}$
  // Answer queries after epoch e
  for $k \in [k_{\max}]$ **do**
    Receive query Q_k from $\mathcal{B}$ with corresponding nonces R, target epochs $(E_r)_{r\in R}$, attribution functions $(A_r)_{r\in R}$ and noise std-dev σ.
    Ask $\mathcal{A}$ to send Q_k with devices $(d_r)_{r\in R}$, receive $(v_{x_c})^1_{e,k}$
    Send $w^c_{e,k} = (v_{x_c})^1_{e,k}$ to $\mathcal{B}$

$$\ln\left(\frac{\Pr[W^0 = w]}{\Pr[W^1 = w]}\right) = \ln\left(\frac{\Pr[V^1_{x_0} = v]}{\Pr[V^1_{x_1} = v]}\right) = \ln\left(\frac{\Pr[V^1_{x_0} = v]}{\Pr[V^0_{x_1} = v]}\right) + \ln\left(\frac{\Pr[V^0_{x_1} = v]}{\Pr[V^1_{x_1} = v]}\right) \tag{16}$$

We now show that $\Pr[V^0_{x_1} = v] = \Pr[V^0_{x_0} = v]$. Take $e \in [e_{\max}], k \in [k_{\max}]$, and condition on a prefix $v_{<e,k}$. Then, the only difference between $(V_{x_0})^0_{e,k}$ and $(V_{x_1})^0_{e,k}$ is the underlying database in Alg. 2, that we denote respectively D and D'. There exists a database G such that ${}^0D^{\leq e} = G + \mathbb{1}[e \leq e_0](d_0, e_0, \emptyset)$ and ${}^0D'^{\leq e} = G + \mathbb{1}[e \leq e_1](d_1, e_1, \emptyset)$. Either way, for a report ρ_r and a database $\mathbb{D}$, adding device-epoch records with empty events does not change the value of $\rho_r(D)$. Indeed, by definition D^e_d already returns $\emptyset$ if $(d, e) \notin D$. Hence, $\sum_{r\in R} \rho_r({}^0D^{\leq e}) = \sum_{r\in R} \rho_r({}^0D'^{\leq e}) = \sum_{r\in R} \rho_r(G)$.

Thus,

$$\ln\left(\frac{\Pr[V^1_{x_0} = v]}{\Pr[V^0_{x_1} = v]}\right) = \ln\left(\frac{\Pr[V^1_{x_0} = v]}{\Pr[V^0_{x_0} = v]}\right) \tag{17}$$

Finally, by Thm. 5, Eq. 16 becomes:

$$\left|\ln\left(\frac{\Pr[W^0 = v]}{\Pr[W^1 = v]}\right)\right| \leq \epsilon^G_{x_0} + \epsilon^G_{x_1} \tag{18}$$

□

Theorem 7 (Tighter Thm. 6 with constraint on queries)**.** *Fix a set of public events $P \subset \mathcal{I} \cup \mathcal{C}$, and budget capacities $(\epsilon^G_x)_{x\in\mathcal{X}}$.*

Take any $x = (d, e, F) \in \mathcal{X}$, and define $x_P := (d, e, F \cap P)$. Suppose that all the attribution functions A verify $\forall i, \forall F$, $A(F_1, ..., F_{i-1}, F_i \cap P, F_i, ..., F_k) = A(F_1, ..., F_{i-1}, \emptyset, F_i, ..., F_k)$.

Then, for the record pair (x, x_P), for any adversary $\mathcal{B}$, for W^0, W^1 defined by Alg. 3 and for all $w \in \text{Supp}(W^1)$ we have:

$$\left|\ln\left(\frac{\Pr[W^0 = v]}{\Pr[W^1 = v]}\right)\right| \leq \epsilon^G_x \tag{19}$$

Proof. First, we show that under such queries with $F_A \cap P = \emptyset$, for any $x \in \mathcal{X}$, Alg. 3 produces the same output on $\epsilon^G_{x_P} = 0$ and $\epsilon^G_{x_P} > 0$.

Take any $x = (d_0, e_0, F) \in \mathcal{X}$, and define $x_P := (d_0, e_0, F \cap P)$. Take a report ρ with an attribution function A that is executed on d_0 and E such that $e_0 \in E$. If $\epsilon^G_{x_P} = 0$, Alg. 3 sets $F_{e_0} = \emptyset$ and returns $\rho = A((F_e)_{e\in E\setminus\{e_0\}}||\emptyset)$. If $\epsilon^G_{x_P} > 0$ and $\mathcal{F}_{x_P}$ has enough budget, Alg. 3 sets $F_{e_0} = F \cap P$ and returns $\rho = A((F_e)_{e\in E\setminus\{e_0\}}||F \cap P)$. Thanks to the constraint on A, we have $A((F_e)_{e\in E\setminus\{e_0\}}||\emptyset) = A((F_e)_{e\in E\setminus\{e_0\}}||F \cap P)$.

Finally, we conclude with Thm. 6. □

B.2 Unlinkability Guarantees (Thm. 2)

Definition 1 (Unlinkability privacy game)**.** *We define a variant of Alg. 3 by applying the following modifications:*

- *We do not require $F_0 \cap P = F_1 \cap P$ anymore, and we define $x_2 := (d_0, e_0, F_0 \setminus F_1)$*
- *If $c = 1$, after receiving G from $\mathcal{B}$, if $e = e_0$ and $x_2 \notin G$, we perform $G \leftarrow G + x_2$.*

In this variant, $\mathcal{B}$ tries to distinguish between World 0 in which the database is $G + x_0 = G + (d_0, e_0, F_0)$, and World 1 in which the database is $G + x_1 + x_2 = G + (d_1, e_1, F_1) + (d_0, e_0, F_0 \setminus F_1)$. In World 0, all the events in F_0 are located on the same device, while in World 1 there are some events on device d_0 and some events on device d_1.

Theorem 8 (Unlinkability guarantees)**.** *Fix a set of public events $P \subset \mathcal{I} \cup \mathcal{C}$, and budget capacities $(\epsilon^G_x)_{x\in\mathcal{X}}$.*

Take any $d_0, d_1 \in \mathcal{D}$, $e \in \mathcal{E}$, $F_0 \subset \mathcal{I} \cup \mathcal{C}$ and $F_1 \subset F_0$, and pose $x_0 := (d_0, e, F_0), x_1 := (d_1, e, F_1), x_2 := (d_0, e, F_0 \setminus F_1) \in \mathcal{X}$. Take any adversary $\mathcal{B}$ for the game from Def. 1 with record triple (x_0, x_1, x_2), and note U^0, U^1 the views of $\mathcal{B}$.

Then, for all $u \in \text{Supp}(U^1)$ we have:

$$\left|\ln\left(\frac{\Pr[U^0 = u]}{\Pr[U^1 = u]}\right)\right| \leq \epsilon^G_{x_0} + \epsilon^G_{x_1} + \epsilon^G_{x_2} \tag{20}$$

This bounds the ability of $\mathcal{B}$ to tell whether all the events F_0 (both public and private) belong to a single device or not.

Proof. Take $u \in \text{Supp}(U^1)$. Similar to Thm. 6, the nonce and public information follow the same distribution in U^0 and U^1, and the rest of the view corresponds to an execution of Alg. 2 with challenge bit $b = 1$. Hence we have:

$$\ln\left(\frac{\Pr[U^0 = u]}{\Pr[U^1 = u]}\right) = \ln\left(\frac{\Pr[V^1_{x_0} = v]}{\Pr[V^1_{x_1,x_2} = v]}\right) \tag{21}$$

where $u, V^1_{x_0}, V^1_{x_1,x_2}$ are defined as follows:

- v is the truncated version of u obtained by removing the nonces and public information.
- $V^b_{x_0}$ is the view of the adversary $\mathcal{A}$ defined in Alg. 3 with $b \in \{0, 1\}$, that if $b = 1$ inserts the opt-out record x_0 in the database submitted by $\mathcal{B}$.
- $V^b_{x_1,x_2}$ is the view of the adversary $\mathcal{A}'$ defined in Def. 1 with $b \in \{0, 1\}$, that if $b = 1$ inserts the opt-out record x_1 in the database submitted by $\mathcal{B}$ extended with x_2.
- $V^b_{x_2}$ the view of the adversary $\mathcal{A}$" defined in Alg. 3 with $b \in \{0, 1\}$, that if $b = 1$ inserts the opt-out record x_2 in the database submitted by $\mathcal{B}$.

With the same reasoning as in Thm. 6 (Eq. 17), we have $V^0_{x_0} \sim V^0_{x_2}$. We also have $V^0_{x_1,x_2} = V^1_{x_2}$. Thus, Eq. 21 becomes:

$$\ln\left(\frac{\Pr[U^0 = u]}{\Pr[U^1 = u]}\right) = \ln\left(\frac{\Pr[V^1_{x_0} = v]}{\Pr[V^1_{x_1,x_2} = v]} \frac{\Pr[V^0_{x_1,x_2} = v]}{\Pr[V^1_{x_2} = v]} \frac{\Pr[V^0_{x_2} = v]}{\Pr[V^0_{x_0} = v]}\right)$$

We conclude with Thm. 5. □

Theorem 9 (Simplified Expression for Thm. 8)**.** *Fix a set of public events $P \subset \mathcal{I} \cup \mathcal{C}$, and budget capacities $(\epsilon_x^G)_{x\in\mathcal{X}}$. Take any $d_0, d_1 \in \mathcal{D}$, $e \in \mathcal{E}$, $F_1 \subset F_0 \subset P$ (*i.e., *all the events we consider here are public events), and pose $x_0 := (d_0, e, F_0), x_1 := (d_1, e, F_1), x_2 := (d_0, e, F_0 \setminus F_1) \in \mathcal{X}$. Take any adversary $\mathcal{B}$ for the game from Def. 1 with record triple (x_0, x_1, x_2), and note U^0, U^1 the views of $\mathcal{B}$. Suppose that all the attribution functions A submitted by $\mathcal{B}$ have relevant events sets $I \cup C$ that verify $F_A \cap P = \emptyset$*

Then, for all $u \in \mathrm{Supp}(U^1)$ we have:

$$\left|\ln\left(\frac{\Pr[U^0 = u]}{\Pr[U^1 = u]}\right)\right| = 0 \tag{22}$$

Proof. First, we observe that $F_0 \cap F_A = F_1 \cap F_A = (F_0 \setminus F_1) \cap F_A = \emptyset$. Then, by applying the same reasoning as Thm. 7, we can suppose without loss of generality that $\epsilon^G_{x_0} = \epsilon^G_{x_1} = \epsilon^G_{x_2} = 0$. We conclude with Thm. 8. □

B.3 Privacy Guarantees Under Colluding Queriers

We show that, as in DP, colluding parties can be analyzed using DP composition. This property is not immediate, because queriers in Alistair possess side information that they use to define queries with good IDP properties. Informally, for a record x on device d, the collusion of n parties with budget $\epsilon_d^{G_1}, \ldots, \epsilon_d^{G_n}$ is $2\epsilon_d^{G_1} + \cdots + 2\epsilon_d^{G_n}$-DP for x under the joint public information. We can remove the factor 2 when queries never look at the public data from *any* colluding querier.

Theorem 10 (Colluding Queriers)**.** *Fix $n > 1$ a number of colluding queriers (*i.e., *adversaries from Alg. 3). For simplicity, we suppose that the data is not adaptively chosen, allowing us to see each querier as an interactive mechanism with view $\mathcal{M}_i^{\leftrightarrow}(D)$ when executed on a database $D \in \mathbb{D}$. Fix a set of public events $P_i \subset \mathcal{I} \cup \mathcal{C}$ for each querier $i \in [n]$, and budget capacities $(\epsilon_x^{G_i})_{x\in\mathcal{X}}$. Define $P := P_1 \cup \cdots \cup P_n$.*

For any pair of records $x_0 = (d_0, e_0, F_0), x_1 = (d_1, e_1, F_1) \in \mathcal{X}$ such that $e_0 = e_1$ and $F_0 \cap P = F_1 \cap P$, for any database $D \in \mathbb{D}$ with $(d_0, e_0) \notin D, (d_1, e_1) \notin D$, for any adversary $\mathcal{M}$ that concurrently executes $\mathcal{M}_1^{\leftrightarrow}, \ldots, \mathcal{M}_n^{\leftrightarrow}$ on the same data (potentially interleaving and adaptively choosing queries), for all $S \in \mathrm{Range}(\mathcal{M})$ we have:

$$\Pr[\mathcal{M}(D + x_0) \in S] \leq \exp\left(\sum_{i=1}^{n} \epsilon_{x_0}^{G_i} + \epsilon_{x_1}^{G_i}\right) \Pr[\mathcal{M}(D + x_1) \in S] \tag{23}$$

When the attribution functions used by any querier satisfy $\forall i, \forall F,\ A(F_1, ..., F_{i-1}, F_i \cap P, F_i, ..., F_k) = A(F_1, ..., F_{i-1}, \emptyset, F_i, ..., F_k)$, and when $x_1 = (d_0, e_0, F_0 \cap P)$, then we can remove the $\epsilon_{x_1}^{G_i}$ term.

In such a case of colluding queriers, the constraint that $\forall F,\ A(F \cap P) = A(\emptyset)$ is more restrictive than merely asking $\forall F, A^i(F \cap P_i) = \emptyset$ for a single querier as in Thm. 7. For instance, the queries we describe in §4.3 will not verify this constraint if an advertiser and a publisher collude. However, the guarantee under general queries of $2\sum_{i=1}^n \epsilon_d^{G_i}$-DP still applies.

Proof. The key observation is that Thm. 6 shows that Alg. 3 is in particular DP under a more restrictive Change One neighborhood relation over the union of the public information across queriers. We can then compose n mechanisms under this restrictive neighborhood relation.

More formally, fix $Q \subset \mathcal{I} \cup \mathcal{C}$ and $x = (d, e, F), x' = (d', e', F') \in \mathcal{X}$. We define the following neighborhood relation over databases. For $D, D' \in \mathbb{D}$, we say $D \underset{x,x'}{\overset{Q}{\sim}} D$ if $e = e'$, $F \cap Q = F' \cap Q$, and there exists $D_0 \in \mathbb{D}$ such that $D = D_0 + x$ and $D' = D_0 + x'$ or vice versa. Consider $x_0 = (d_0, e_0, F_0), x_1 = (d_1, e_1, F_1) \in \mathcal{X}$ such that $e_0 = e_1$. For all $i \in [n]$, we have $F_0 \cap P = F_1 \cap P \implies F_0 \cap P_i = F_1 \cap P_i$, and thus:

$$\forall D, D' \in \mathbb{D}, D \underset{x_0,x_1}{\overset{P}{\sim}} D \implies D \underset{x_0,x_1}{\overset{P_i}{\sim}} D \tag{24}$$

Thm. 6 shows the interactive mechanism $\mathcal{M}_i^{\leftrightarrow}$ is $\epsilon_{x_0}^{G_i} + \epsilon_{x_1}^{G_i}$-DP under the $\underset{x_0,x_1}{\overset{P_i}{\sim}}$ relation. Thanks to Eq. 24, $\mathcal{M}_i^{\leftrightarrow}$ is also $\epsilon_{x_0}^{G_i} + \epsilon_{x_1}^{G_i}$-DP under the $\underset{x_0,x_1}{\overset{P}{\sim}}$ relation. Note that this conclusion would not be true if we had proved Thm. 6 under the replace-with-default definition $D \sim_x^Q D'$ introduced in §4.1.

Next, the adversary that concurrently executes the n queriers is operating a concurrent composition of interactive mechanisms $\mathcal{M}_1^{\leftrightarrow}, \ldots, \mathcal{M}_n^{\leftrightarrow}$. Thanks to the concurrent composition theorem [43], the resulting mechanism $\mathcal{M}$ is $\sum_{i=1}^n \epsilon_{x_0}^{G_i} + \epsilon_{x_1}^{G_i}$-DP under the $\underset{x_0,x_1}{\overset{P}{\sim}}$ relation.

□

C Proofs for IDP Optimizations (§6)

Theorem 11 (Global sensitivity of reports)**.** *Fix a report identifier r, a device d_r, a set of epochs E_r, an attribution function A and the corresponding report $\rho : D \mapsto A(D_{d_r}^{E_r})$. We have:*

$$\Delta(\rho) = \max_{i\in[k], F_1,\ldots,F_k \in \mathcal{P}(\mathcal{I}\cup\mathcal{C})} \|A(F_1, ..., F_k) - A(F_1, ..., F_{i-1}, \emptyset, F_{i+1}, ..., F_k)\|_1$$

If A has m-dimensional output and verifies $\forall \mathbf{F} \in \mathcal{P}(\mathcal{I} \cup \mathcal{C})^k, \forall i \in [m], A(\mathbf{F})_i \in [0, A^{\max}]$, then we have $\Delta(\rho) \leq mA^{\max}$.

Proof. Take such a report ρ. We enumerate the requested epochs from 1 to $k = |E_r|$: $E_r = \{e_1, \ldots, e_k\}$.

First, by definition of the global sensitivity, we have:

$$\Delta(\rho) = \max_{D,D' \in \mathbb{D}: \exists x \in \mathcal{X}, D' = D + x} \|\rho(D) - \rho(D')\|_1 \quad (25)$$

$$= \max_{D,D' \in \mathbb{D}: \exists x \in \mathcal{X}, D' = D + x} \|A(D^{E_r}_{d_r}) - A((D')^{E_r}_{d_r})\|_1 \quad (26)$$

$$= \max_{D,D' \in \mathbb{D}: \exists x = (d_r, e, F) \in \mathcal{X}: e \in E_r, D' = D + x} \|A(D^{E_r}_{d_r}) - A((D')^{E_r}_{d_r})\|_1 \quad (27)$$

since for $x = (d, e, F)$ with $d \neq d_r$ or $e_r \notin E_r$ we have $A(D^{E_r}_{d_r}) = A((D')^{E_r}_{d_r})$.

Next, we show that the two following sets are equal:

- $\{(D^{E_r}_{d_r}), (D')^{E_r}_{d_r}) | D, D' \in \mathbb{D} : \exists x = (d_r, e, F) \in \mathcal{X} : e \in E_r, D' = D + x\}$
- $\{((F_1, ..., F_{i-1}, \emptyset, F_{i+1}, ..., F_k), (F_1, ..., F_k)) | i \in [k], F_1, \ldots, F_k \in \mathcal{P}(\mathcal{I} \cup \mathcal{C})\}$

On one hand, take $D, D' \in \mathbb{D}$ such that there exists $x = (d_r, e, F) \in \mathcal{X}$ verifying $e_r \in E_r$ and $D' = D + x$. We pose $F_j := (D')^{e_j}_{d_r}$ for $e_j \in E_r$. If x has epoch $e = e_i \in E_r$ for some i, then we have $F_i = F$. Hence, since D must not contain (d_r, e), we have: $D^{E_r}_{d_r} = (F_1, ..., F_{i-1}, \emptyset, F_{i+1}, ..., F_k)$ and $(D')^{E_r}_{d_r} = (F_1, ..., F_k)$.

Reciprocally, take $F_1, ..., F_k \in \mathcal{P}(\mathcal{I} \cup \mathcal{C})$ and $i \in [k]$. We define $D' := \{(d_r, e_1, F_1), \ldots, (d_r, e_k, F_k)\}$ and $D' := D \setminus (d_r, e_i, F_i)$. We have $D, D' \in \mathbb{D}$ and there is $x = (d_r, e_i, F_i) \in \mathcal{X}$ such that $D' = D + x$.

Thus both sets are equal, and the maximum becomes:

$$\Delta(\rho) = \max_{i \in [k], F_1, ..., F_k \in \mathcal{P}(\mathcal{I} \cup \mathcal{C})} \|A(F_1, ..., F_k) - A(F_1, ..., F_{i-1}, \emptyset, F_{i+1}, ..., F_k)\|_1 \quad (28)$$

Finally, suppose that A has output in $\mathbb{R}^m$. Take $\mathbf{F}, \mathbf{F}'$. We have $\|A(\mathbf{F}) - A(\mathbf{F}')\|_1 = \sum_{i=1}^m |A(\mathbf{F})_i - A(\mathbf{F}')_i|$. For $i \in [m]$ we have $A(\mathbf{F})_i \in [0, A^{\max}]$ so $A(\mathbf{F})_i - A(\mathbf{F}')_i \in [-A^{\max}, A^{\max}]$. Hence $\|A(\mathbf{F}) - A(\mathbf{F}')\|_1 \leq mA^{\max}$.

This upper bound on $\Delta(\rho)$ can be refined if A has certain properties, such as being a histogram query. □

Theorem 12 (Global sensitivity of queries)**.** *Fix a query Q with corresponding report identifiers R and reports, devices and epoch windows $(\rho_r, d_r, E_r)_{r \in R}$.*

$$\Delta(Q) \leq \max_{d,e} \sum_{r \in R: d = d_r, e \in E_r} \Delta(\rho_r) \quad (29)$$

In particular, if each device-epoch participates in at most one report, then $\Delta(Q) = \max_{r \in R} \Delta(\rho_r)$.

Proof. Take such a query Q. We observe that

$$\Delta(Q) = \max_{D,D' \in \mathbb{D}: \exists x \in \mathcal{X}, D' = D + x} \|Q(D) - Q(D')\|_1 \quad (30)$$

$$= \max_{x \in \mathcal{X}} \max_{D,D' \in \mathbb{D}: D' = D + x} \|Q(D) - Q(D')\|_1 \quad (31)$$

Take $x = (d, e, F) \in \mathcal{X}$. For $r \in R$ such that $d \neq d_r$ or $e \notin E_r$, we have $\rho_r(D) = \rho_r(D')$. Thus:

$$\|Q(D) - Q(D')\|_1 = \|\sum_{r \in R} \rho_r(D) - \rho_r(D')\|_1 \quad (32)$$

$$= \|\sum_{r \in R: d = d_r, e \in E_r} \rho_r(D) - \rho_r(D')\|_1 \quad (33)$$

Using the triangle inequality and the definition of $\Delta(\rho)$ we get:

$$\|Q(D) - Q(D')\|_1 \leq \sum_{r \in R: d = d_r, e \in E_r} \|\rho_r(D) - \rho_r(D')\|_1 \quad (34)$$

$$\leq \sum_{r \in R: d = d_r, e \in E_r} \Delta(\rho_r) \quad (35)$$

This bound is independent on D, D' so:

$$\max_{D,D' \in \mathbb{D}: D' = D + x} \|Q(D) - Q(D')\|_1 \leq \sum_{r \in R: d = d_r, e \in E_r} \Delta(\rho_r) \quad (36)$$

Finally, this does not involve F so we can replace the max over $x = (d, e, F)$ by a max over (d, e):

$$\max_{x \in \mathcal{X}} \max_{D,D' \in \mathbb{D}: D' = D + x} \|Q(D) - Q(D')\|_1 \leq \max_{d,e} \sum_{r \in R: d = d_r, e \in E_r} \Delta(\rho_r) \quad (37)$$

If each device-epoch participates in at most one report, then this becomes $\Delta(Q) \leq \max_r \Delta(\rho_r)$. For each r there exists a pair D, D' such that $\|\rho_r(D) - \rho_r(D')\|_1 = \Delta(\rho_r)$. Taking the max across reports shows that the upper bound on $\Delta(Q)$ is tight in this case.

□

Theorem 13 (Individual sensitivity of reports)**.** *Fix a report identifier r, a device d_r, a set of epochs E_r, an attribution function A with relevant events F_A, and the corresponding report $\rho : D \mapsto A(D^{E_r}_{d_r})$. Fix a device-epoch record $x = (d, e, F) \in \mathcal{X}$.*

If the report requests a single epoch $E_r = \{e_r\}$, we have:

$$\Delta_x(\rho) = \begin{cases} \|A(F) - A(\emptyset)\|_1 & \text{if } d = d_r \text{ and } e = e_r \\ 0 & \text{otherwise} \end{cases} \quad (38)$$

Otherwise, we have:

$$\Delta_x(\rho) \leq \begin{cases} \Delta(\rho) & \text{if } d = d_r \text{ and } e \in E_r \text{ and } F \cap F_A \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \quad (39)$$

Proof. Fix such a report ρ and $x \in (d, e, F) \in \mathcal{X}$. Consider any $D, D' \in \mathbb{D}$ such that $D' = D + x$. We have $\rho(D) = A(D^{e_r}_{d_r})$ and $\rho(D') = A((D')^{e_r}_{d_r})$

- First, suppose that the report requests a single epoch e_r.
  - If $d = d_r$ and $e = e_r$, then since $D + x \in \mathbb{D}$ we must have $(d_r, e_r) \notin D$, and thus $D^{e_r}_{d_r} = \emptyset$. On the other

hand, we have $(D')^{e_r}_{d_r} = F$. Thus, $\|\rho(D) - \rho(D')\|_1 = \|A(F) - A(\emptyset)\|_1$

– If $d \neq d_r$ or $e \neq e_r$, then $(D')^{e_r}_{d_r} = D^{e_r}_{d_r}$. Hence $\|\rho(D) - \rho(D')\|_1 = 0$.

These equalities are independent on D, D', so taking the max gives $\Delta_x(\rho) = \|A(F) - A(\emptyset)\|_1$ if $d = d_r$ and $e = e_r$, and 0 otherwise.

- Second, suppose that the report requests an arbitrary range of epochs E_r.
  – If $d \neq d_r$ or $e \neq E_r$, then $(D')^{E_r}_{d_r} = D^{E_r}_{d_r}$. Hence $\|\rho(D) - \rho(D')\|_1 = 0$.
  – If $d = d_r$ and $e = e_i \in E_r$ and $F \cap F_A = \emptyset$, we have $(D')^{E_r}_{d_r} = (D^{e_1}_{d_r}, ..., D^{e_{i-1}}_{d_r}, F, D^{e_{i+1}}_{d_r}, ..., D^{e_k}_{d_r})$. By definition of $I_A \cup C_A$, we have $A((D')^{E_r}_{d_r}) = A(D^{e_1}_{d_r} \cap F_A, ..., D^{e_{i-1}}_{d_r} \cap F_A, F \cap F_A, D^{e_{i+1}}_{d_r} \cap F_A, ..., D^{e_k}_{d_r} \cap F_A)$.
  We also have $D^{E_r}_{d_r} = (D^{e_1}_{d_r}, ..., D^{e_{i-1}}_{d_r}, \emptyset, D^{e_{i+1}}_{d_r}, ..., D^{e_k}_{d_r})$. Since $F \cap F_A = \emptyset = \emptyset \cap F_A$, we get $A((D')^{E_r}_{d_r}) = A(D^{E_r}_{d_r})$ *i.e.,* $\|\rho(D) - \rho(D')\|_1 = 0$.
  – Otherwise, we must have $d = d_r$ and $e \in E_r$ and $F \cap F_A \neq \emptyset$. In that case, $\|\rho(D) - \rho(D')\|_1 = \|A(D^{e_1}_{d_r}, ..., D^{e_{i-1}}_{d_r}, F, D^{e_{i+1}}_{d_r}, ..., D^{e_k}_{d_r}) - A((D^{e_1}_{d_r}, ..., D^{e_{i-1}}_{d_r}, \emptyset, D^{e_{i+1}}_{d_r}, ..., D^{e_k}_{d_r}))\|_1$

The first two identities are independent on D, D', so taking the max gives $\Delta_x(\rho) = 0$. Unfortunately, the third identity depends on D, D'. Taking the max gives:

$$\begin{aligned}\Delta_x(\rho) &= \max_{F_1,...,F_{i-1},\emptyset,F_{i+1},...,F_k \in \mathcal{P}(\mathcal{I} \cup C)} \|A(F_1, ..., F_{i-1}, F, \quad F_{i+1}, ..., F_k) \\ &\quad - A(F_1, ..., F_{i-1}, \emptyset, F_{i+1}, ..., F_k)\|_1 \\ &\leq \max_{i \in [k], F_1,...,F_k \in \mathcal{P}(\mathcal{I} \cup C)} \|A(F_1, ..., F_{i-1}, F, F_{i+1}, ..., F_k) \\ &\quad - A(F_1, ..., F_{i-1}, \emptyset, F_{i+1}, ..., F_k)\|_1 \\ &= \Delta(\rho)\end{aligned}$$

thanks to Thm. 11. Although we can technically keep the first equality to get a tighter expression for $\Delta_x(\rho)$, for common attribution functions $\Delta(\rho)$ is just as tight (*e.g.,* if the attribution cap is attained because of another possible epoch F_j, $j \neq i$).

□

Theorem 14 (Individual sensitivity of queries)**.** *Fix a query Q with corresponding report identifiers R and reports $(\rho_r)_{r \in R}$. Fix a device-epoch record $x = (d, e, F) \in \mathcal{X}$. We have:*

$$\Delta_x(Q) \leq \sum_{r \in R} \Delta_x(\rho_r) \tag{40}$$

In particular, if x participates in at most one report ρ_r, then $\Delta_x(Q) = \Delta_x(\rho_r)$.

Proof. The inequality is immediate by triangle inequality and definition of individual sensitivity. When x participates in at most one report ρ_r, we get $\Delta_x(\rho_{\hat{r}}) = 0$ for $\hat{r} \neq r$, and thus $\Delta_x(Q) \leq \Delta_x(\rho_r)$. The inequality is tight in that case. □

D IDP-Induced Bias Detection

Since individual privacy budgets depend on the data, they must be kept private. That is why Alistair silently replaces out-of-budget device-epoch data by $\emptyset$ instead of raising an exception like IPA. This missing data induces a bias in the query answers and increases the overall error.

IDP-induced bias. Consider a query Q with report identifiers R, target epochs $(E_r)_{r \in R}$, attribution functions $(A_r)_{r \in R}$ and noise parameter σ. For a database D, the true result is $Q(D) = \sum_{r \in R} A_r(D^{E_r}_{d_r})$. When a device-epoch (d_r, e) is out of budget, Alistair drops it.

More formally, Alg. 1 in Appendix A defines $F_e = \emptyset$ instead of $F_e = D^e_{d_r}$. We pose:

$$\tilde{Q}(D) := \sum_{r \in R} A_r((F_e)_{e \in E_r}) \tag{41}$$

We denote by $\mathcal{M}(D)$ the value returned by AnswerQuery: $\mathcal{M}(D) := \tilde{Q}(D) + X$ where $X \sim \mathcal{L}(\sigma)$ has mean zero and variance σ^2. Hence, Alg. 1 returns an estimate for $Q(D)$ with the following bias:

$$\mathbb{E}[\mathcal{M}(D) - Q(D)] = \tilde{Q}(D) - Q(D) \tag{42}$$

Detecting bias with global sensitivity. When no device-epoch is out of budget, Alg. 1 returns an unbiased estimate. We can guarantee that no device-epoch is out of budget by keeping track of a budget consumption bound as follows. Assume we know (1) a lower bound ϵ^G on the individual budget capacity: $\forall x \in D, \epsilon^G_x \geq \epsilon^G$, and (2) an upper bound on the individual budget for each report r in each query k: $\epsilon^{k,r}_x \leq \epsilon^{k,r}$. Then, for all $x \in D$, $\sum_{k,r} \epsilon^{k,r} \leq \epsilon^G \implies \sum_{k,r} \epsilon^{k,r}_x \leq \epsilon^G_x$.

In practice, the individual budget can be bounded by using the fact that the individual sensitivity is upper bounded by the (data-independent) global sensitivity. Hence, a querier can run its own off-device budgeting scheme to detect the earliest potentially biased query. This approach does not consume any budget since it only relies on public query information. However, once $\sum_{k,r} \epsilon^{k,r} > \epsilon^G$ this approach doesn't guarantee that queries are biased (or unbiased).

Estimating bias with DP counting. To get a more granular estimate of the bias, we can run a special query counting the number of out-of-budget device-epochs, as follows. Given a query Q with output in $\mathbb{R}^m$, we atomically execute (Q_0, Q) as a single query with output in $\mathbb{R}^{m+1}$, where $Q_0(D) := \sum_{r \in R} \sum_{e \in E_r} \mathbb{1}[D^e_{d_r} = \emptyset]$. Prepending a counting query to Q gives a high probability bound on the bias, formally stated in Thm. 15.

Theorem 15. *Take a query Q with report identifiers R, parameters $(d_r, E_r, A_r, \rho_r)_{r \in R}$, and output in $\mathbb{R}^m$. Fix $\kappa > 0$, a parameter to control the precision of the bound. For $r \in R$, we define*

$\hat{A}_r : \mathcal{P}(\mathcal{I} \cup \mathcal{C}) \to \mathbb{R}^{m+1}$ *by:* $\hat{A}_r(F_1, \ldots, F_k)_0 = \kappa \cdot \sum_{i=1}^{k} \mathbb{1}[F_i = \emptyset]$ *and* $\forall i \in [m+1], \hat{A}_r(F_1, \ldots, F_k)_i = A_r(F_1, \ldots, F_k)_i$. *We pose* $Q_0(D) := \sum_{r \in R} \kappa \cdot \sum_{e \in E_r} \mathbb{1}[D^e_{d_r} = \emptyset]$, *and denote by* $(\mathcal{M}_0(D), \mathcal{M}(D))$ *the output of Alg. 1 on* (Q_0, Q).

For $\beta \in (0, 1)$, *with probability* $1 - \beta$ *we have:*

$$\|\mathbb{E}[\mathcal{M}(D) - Q(D)]\|_1 \leq \frac{\mathcal{M}_0(D) + \sigma \ln(1/\beta)/\sqrt{2}}{\kappa} \max_{r \in R} \Delta(\rho_r)$$

Intuitively, for a fixed noise standard deviation σ (query results quality), a querier can consume extra privacy budget compared to running Q alone (because Q_0 adds κ to the sensitivity of reports) to be able to detect bias above a threshold with high probability.

Proof of Thm. 15. First, $\mathcal{M}_0(D)/\kappa$ provides an *unbiased* estimate of an upper bound $\tilde{Q}_0(D)/\kappa$ on the number of out-of-budget device-epochs in Q, where $\tilde{Q}_0(D)$ is the query result obtained after dropping out-of-budget device-epochs, defined in Eq. 41. Indeed, when d_r, e runs out of budget, $\hat{A}_r$ receives $F_{r,e} = \emptyset$ in Alg. 1. Hence:

$$\tilde{Q}_0(D)/\kappa = |\{(r, e) : r \in R, e \in E_r, F_{r,e} = \emptyset\}| \tag{43}$$

$$\geq |\{(r, e) : r \in R, e_r \in E_r, \mathcal{F} \text{ returned Halt for } (e, r)\}| \tag{44}$$

If the original database D verifies $D^e_{d_r} \neq \emptyset$ for all $r, e \in E_r$, then Eq. 44 becomes an equality. We can programatically enforce $D^e_{d_r} \neq \emptyset$ by adding a special heartbeat event $f_0 \in F$ in every device-epoch.

Second, we can use combine this estimate with the attribution cap to bound the bias:

$$\|\mathbb{E}[\mathcal{M}(D) - Q(D)]\|_1 = \|\tilde{Q}(D) - Q(D)\|_1 \tag{45}$$

$$= \| \sum_{r \in R} A(F_{r,e_1}, \ldots, F_{r,e_k}) - A(D^{e_1}_{d_r}, \ldots, D^{e_k}_{d_r}\|_1 \tag{46}$$

$$\leq \sum_{r \in R} \|A(F_{r,e_1}, \ldots, F_{r,e_k}) - A(\emptyset, D^{e_2}_{d_r}, \ldots, D^{e_k}_{d_r}) \tag{47}$$

$$+ A(\emptyset, D^{e_2}_{d_r}, \ldots, D^{e_k}_{d_r}) - A(D^{e_1}_{d_r}, D^{e_2}_{d_r}, \ldots, D^{e_k}_{d_r})\|_1$$

$$\leq \ldots$$

$$\leq \sum_{r \in R} |\{e \in E_r : F_{r,e} = \emptyset\}| \Delta(\rho_r) \tag{48}$$

$$\leq |\{(r, e) : r \in R, e \in E_r, F_{r,e} = \emptyset\}| \max_{r \in R} \Delta(\rho_r) \tag{49}$$

$$= (\tilde{Q}_0(D)/\kappa) \max_{r \in R} \Delta(\rho_r) \tag{50}$$

Finally, when $\mathcal{M}_0(D)$ is a noisy version of $\tilde{Q}_0(D)$, we can use a tail bound to get a high probability bound on the expected bias. The knob κ controls the precision of the out-of-budget count: higher κ gives a more precise estimate but consumes more budget. More precisely, when $\mathcal{L} = \text{Lap}$, for an absolute error τ in the number of out-of-budget device-epochs and a failure probability target $\beta \in (0, 1)$, setting $\kappa = \frac{\sigma \ln(1/\beta)}{\tau\sqrt{2}}$ gives:

$$\Pr[|\mathcal{M}_0(D)/\kappa - \tilde{Q}_0(D)/\kappa| > \tau] = \beta \tag{51}$$

□

Theorem 16. *Consider Thm. 15, and replace* Q_0 *by the following counting query:* $Q_0(D) := \sum_{r \in R} \kappa \cdot \mathbb{1}[\exists e \in E_r : D^e_{d_r} = \emptyset]$. *This gives an estimate of the number of reports that contain an out-of-budget epoch, which is smaller than the number of out-of-budget epochs across all reports.*

For a report ρ *with attribution function* A *over* k *epochs, we also define:*

$$\Delta^{\max}(\rho_r) := \max_{\mathbf{F}, \mathbf{F}' \in \mathcal{P}(\mathcal{I} \cup \mathcal{C})^k : \forall i \in [k], \mathbf{F}'_i = \mathbf{F}_i \text{ or } \mathbf{F}'_i = \emptyset} \|A(\mathbf{F}) - A(\mathbf{F}')\|_1 \tag{52}$$

That is, $\Delta^{\max}(\rho_r)$ *is the maximum L1 change that happens when we remove* any *number of device-epochs from the database. By comparison, the global sensitivity* $\Delta(\rho_r)$ *is the maximum change that happens when we remove* a single *device-epoch from the database. For certain attribution functions, such as last touch attribution,* $\Delta^{\max}(\rho_r) = \Delta(\rho_r)$, *as detailed in Lemma18.*

Then, we have:

$$\|\mathbb{E}[\mathcal{M}(D) - Q(D)]\|_1 \leq \frac{\mathcal{M}_0(D) + \sigma \ln(1/\beta)/\sqrt{2}}{\kappa} \max_{r \in R} \Delta^{\max}(\rho_r)$$

Proof. There are two differences compared to the proof of Thm. 15.

First, $\mathcal{M}_0(D)$ is an unbiased estimate of $\tilde{Q}_0(D)$, which is an upper bound on the number of reports containing at least one out-of-budget epoch.

Second, we can use $\tilde{Q}_0(D)$ to bound the bias as follows:

$$\|\mathbb{E}[\mathcal{M}(D) - Q(D)]\|_1 = \|\tilde{Q}(D) - Q(D)\|_1 \tag{53}$$

$$= \|\sum_{r\in R} A(F_{r,e_1}, \ldots, F_{r,e_k}) - A(D^{e_1}_{d_r}, \ldots, D^{e_k}_{d_r})\|_1 \tag{54}$$

$$\leq \sum_{r\in R} \|A(F_{r,e_1}, \ldots, F_{r,e_k}) - A(D^{e_1}_{d_r}, \ldots, D^{e_k}_{d_r})\| \tag{55}$$

$$\leq \sum_{r\in R:\forall i\in[k], F_{r,e_i}=D^{e_i}_{d_r}} \|A(F_{r,e_1}, \ldots, F_{r,e_k}) - A(D^{e_1}_{d_r}, \ldots, D^{e_k}_{d_r})\|_1 \tag{56}$$

$$+ \sum_{r\in R:\exists i\in[k]:F_{r,e_i}\neq D^{e_i}_{d_r}} \|A(F_{r,e_1}, \ldots, F_{r,e_k}) - A(D^{e_1}_{d_r}, \ldots, D^{e_k}_{d_r})\|_1 \tag{57}$$

$$\leq \sum_{r\in R:\exists i\in[k]:F_{r,e_i}=\emptyset} \|A(F_{r,e_1}, \ldots, F_{r,e_k}) - A(D^{e_1}_{d_r}, \ldots, D^{e_k}_{d_r})\|_1 \tag{58}$$

$$\leq \sum_{r\in R:\exists i\in[k]:F_{r,e_i}=\emptyset} \Delta^{\max}(\rho_r) \tag{59}$$

$$\leq (\tilde{Q}_0(D)/\kappa) \max_{r\in R} \Delta^{\max}(\rho_r) \tag{60}$$

We conclude with a Laplace tail bound as in Thm. 15.

□

Theorem 17 (Sensitivity of counting queries)**.** *Consider a query $(d_r, E_r, A_r, \rho_r)_{r\in R}$ augmented by a counting query as in Thm. 15 or Thm. 16. Take a report $\hat{\rho}_r : D \mapsto (\rho^0_r(D), \rho_r(D)) \in \mathbb{R}^m$ where $\rho^0_r(D) = \kappa \cdot \sum_{e\in E_r} \mathbb{1}[D^e_{d_r} = \emptyset]$ (Thm. 15) or $\rho^0_r(D) = \kappa \cdot \mathbb{1}[\exists e \in E : D^e_{d_r} = \emptyset]$ (Thm. 16).*

Take $x = (d, e, F) \in \mathcal{X}$. We have:

$$\Delta_x(\hat{\rho}_r) \leq \kappa \cdot \mathbb{1}[d = d_r, e \in E_r \text{ and } F \neq \emptyset] + \Delta_x(\rho_r) \tag{61}$$

This means that every requested device-epoch that has budget left and contains data should pay additional budget for the DP count.

Proof. First, we have:

$$\Delta_x(\hat{\rho}_r) \leq \Delta_x(\rho^0_r) + \Delta_x(\rho_r) \tag{62}$$

because for all D, D' such that $D' = D+x$ we have $\|\hat{\rho}_r(D') - \hat{\rho}_r(D)\|_1 \leq \|\rho^0_r(D') - \rho^0_r(D)\|_1 + \|\rho_r(D') - \rho_r(D)\|_1 \leq \Delta_x(\rho^0_r) + \Delta_x(\rho_r)$.

Second, we have:

$$\Delta_x(\rho^0_r) = \begin{cases} \kappa & \text{if } d = d_r, e \in E_r \text{ and } F \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \tag{63}$$

Indeed, consider D, D' such that $D' = D + x$.

- If $F = \emptyset$, $d \neq d_r$, or $e \notin E_r$ we have $\rho^0_r(D) = \rho^0_r(D')$ for all such D, D' so $\Delta_x(\rho^0_r) = 0$.
- If $F \neq \emptyset$, $d = d_r$ and $e \in E_r$ we have:
  - For Thm. 15, $\|\rho^0_r(D') - \rho^0_r(D)\|_1 = \|\kappa \cdot \sum_{\hat{e}\in E_r} \mathbb{1}[(D')^{\hat{e}}_{d_r} = \emptyset] - \mathbb{1}[D^{\hat{e}}_{d_r} = \emptyset]\| = \kappa \cdot \mathbb{1}[F = \emptyset] - [\emptyset = \emptyset] = \kappa$. This is true for all such D, D', so $\Delta_x(\rho^0_r) = \kappa$.
  - For Thm. 16, $\|\rho^0_r(D') - \rho^0_r(D)\|_1 = \|\kappa \cdot \mathbb{1}[\exists \hat{e} \in E_r : (D')^{\hat{e}}_{d_r} = \emptyset] - \mathbb{1}[\exists \hat{e} \in E_r : D^{\hat{e}}_{d_r} = \emptyset]\| \leq \kappa$. Moreover, this max is attained for $D = \{(d_r, \hat{e}, F), \hat{e} \in E_r \setminus \{e\}\}, D' = \{(d_r, \hat{e}, F), \hat{e} \in E_r\}$. Hence $\Delta_x(\rho^0_r) = \kappa$.

□

Theorem 18 (Sensitivity for certain histogram attribution functions)**.** *Consider an attribution function A of the following form. First, A attributes a positive value $a_{\mathbf{F}}(f)$ to each relevant event $f \in F_1 \cap F_A \cup \cdots \cup F_k \cap F_A$. Next, each event is mapped to a one-hot vector $H(f) \in \mathbb{R}^m$ (*i.e., *$H(f) \in \{0,1\}^m$ and $\|H(f)\|_1 = 1$). Finally, the attribution is the weighted sum:*

$$A(F_1, \ldots, F_k) = \sum_{i=1}^{k} \sum_{f\in F_i\cap F_A} a_{\mathbf{F}}(f) \cdot H(f) \tag{64}$$

We define:

$$A^{\max} := \max_{\mathbf{F}\in\mathcal{P}(\mathcal{I}\cup C)^k} \sum_{i=1}^{k} \sum_{f\in F_i\cap F_A} a_{\mathbf{F}}(f) \tag{65}$$

Consider any attribution report ρ_r with attribution function A with output in $\mathbb{R}^m$.

- *If $m = 1$ or $k = 1$, we have*

$$\Delta(\rho_r) \leq \Delta^{\max}(\rho_r) \leq A^{\max} \tag{66}$$

Moreover, if there exists $\mathbf{F}^{\max} = (\emptyset, \ldots, \emptyset, \{f_0\}, \emptyset, \ldots, \emptyset)$ containing a single relevant event $f_0 \in F_A$ such that $A^{\max}$ is attained, i.e., *$a_{\mathbf{F}^{\max}}(f_0) = A^{\max}$, then*

$$\Delta(\rho_r) = \Delta^{\max}(\rho_r) = A^{\max} \tag{67}$$

- *If $m \geq 2$ and $k \geq 2$, we have:*

$$\Delta(\rho_r) \leq \Delta^{\max}(\rho_r) \leq 2A^{\max} \tag{68}$$

Moreover, if there exists $\mathbf{F}^{\max} = (\emptyset, \ldots, \emptyset, \{f_0\}, \emptyset, \ldots, \{f_1\}, \emptyset)$ and $\mathbf{F'}^{\max} = (\emptyset, \ldots, \emptyset, \{f_0\}, \emptyset, \ldots, \emptyset)$ such that $a_{\mathbf{F}^{\max}}(f_0) = A^{\max}$, $a_{\mathbf{F'}^{\max}}(f_1) = A^{\max}$ and $H(f_0) \neq H(f_1)$, then:

$$\Delta(\rho_r) = \Delta^{\max}(\rho_r) = 2A^{\max} \tag{69}$$

Proof. Consider a report ρ_r with such an attribution function A. First, we observe that $A(\emptyset) = 0 \in \mathbb{R}^m$, because of Eq. 64.

We start by upper bounding $\Delta^{\max}(\rho_r)$. Take $\mathbf{F}, \mathbf{F'} \in \mathcal{P}(\mathcal{I} \cup C)^k : \forall i \in [k], \mathbf{F}'_i = \mathbf{F}_i$ or $\mathbf{F}'_i = \emptyset$.

- If $m = 1$, for any event f we have $H(f) = 1$. Since $a_{\mathbf{F}}(f) \geq 0$, we have: $\sum_{i=1}^{k} \sum_{f\in F_i\cap F_A} a_{\mathbf{F}}(f) \cdot H(f) - \sum_{f\in F'_i\cap F_A} a_{\mathbf{F'}}(f) \cdot H(f) \leq \sum_{i=1}^{k} \sum_{f\in F_i\cap F_A} a_{\mathbf{F}}(f) \cdot 1 \leq A^{\max}$ and $\sum_{i=1}^{k} \sum_{f\in F_i\cap F_A} a_{\mathbf{F}}(f) \cdot H(f) - \sum_{f\in F'_i\cap F_A} a_{\mathbf{F'}}(f) \cdot$

$H(f) \geq -\sum_{f \in F_i' \cap F_A} a_{\mathbf{F}'}(f) \cdot 1 \geq -A^{\max}$. Hence, $\|A(\mathbf{F}) - A(\mathbf{F}')\|_1 \leq A^{\max}$, and thus $\Delta^{\max} \leq A^{\max}$.

- If $k = 1$, we have $\mathbf{F}' = F_1$ or $\emptyset$. In the first case, $\|A(\mathbf{F}) - A(\mathbf{F}')\|_1 = 0 \leq A^{\max}$. In the second case,

$$\|A(\mathbf{F}) - A(\mathbf{F}')\|_1 = \|A(\mathbf{F})\|_1 \tag{70}$$
$$\leq \sum_{f \in F_1 \cap F_A} a_{\mathbf{F}}(f) \|H(f)\|_1 \tag{71}$$
$$\leq A^{\max} \tag{72}$$

  Hence $\Delta^{\max} \leq A^{\max}$.
- If $m \geq 2$, we have:

$$\|A(\mathbf{F}) - A(\mathbf{F}')\|_1 = \| \sum_{i=1}^{k} \sum_{f \in F_i \cap F_A} a_{\mathbf{F}}(f) \cdot H(f) \tag{73}$$
$$- \sum_{f \in F_i' \cap F_A} a_{\mathbf{F}'}(f) \cdot H(f) \|_1 \tag{74}$$
$$\leq \sum_{i=1}^{k} \sum_{f \in F_i \cap F_A} a_{\mathbf{F}}(f) \|H(f)\|_1 \tag{75}$$
$$+ \sum_{i=1}^{k} \sum_{f \in F_i' \cap F_A} a_{\mathbf{F}'}(f) \|H(f)\|_1 \tag{76}$$
$$\leq 2A^{\max} \tag{77}$$

  This is true for any such $\mathbf{F}, \mathbf{F}'$, so $\Delta^{\max} \leq 2A^{\max}$.

Next, we lower bound $\Delta^{\max}$.

- If $m = 1$ or $k = 1$, and if there exists $\mathbf{F}^{\max} = (\emptyset, \ldots, \emptyset, \{f_0\}, \emptyset, \ldots, \emptyset)$ such that $a_{\mathbf{F}^{\max}}(f_0) = A^{\max}$, we have

$$\Delta^{\max}(\rho_r) = \max_{\mathbf{F},\mathbf{F}' \in \mathcal{P}(\mathcal{I} \cup C)^k : \forall i \in [k], \mathbf{F}_i' = \mathbf{F}_i \text{ or } \mathbf{F}_i' = \emptyset} \|A(\mathbf{F}) - A(\mathbf{F}')\|_1 \tag{78}$$
$$\geq \|A(\mathbf{F}^{\max}) - A(\emptyset)\|_1 \tag{79}$$
$$= \|A^{\max} \cdot H(f_0) - 0\|_1 \tag{80}$$
$$= A^{\max} \tag{81}$$

  (in fact this is true even when $m \neq 1$ and $k \neq 1$).
- If $m \geq 2$ and $k \geq 2$, and there exists f_0, f_1 such that removing f_1 shifts the attribution to f_0, and $H(f_0) \neq H(f_1)$, then:

$$\Delta^{\max}(\rho_r) \geq \|A(\mathbf{F}^{\max}) - A(\mathbf{F}'^{\max})\|_1 \tag{82}$$
$$= \|A^{\max} \cdot H(f_0) - A^{\max} \cdot H(f_1)\|_1 \tag{83}$$
$$= 2A^{\max} \tag{84}$$

We now focus on $\Delta(\rho_r)$. First, we have $\Delta(\rho_r) \leq \Delta^{\max}(\rho_r)$, because if we note $N := \{\mathbf{F}, \mathbf{F}' \in \mathcal{P}(\mathcal{I} \cup C)^k : \exists i \in [k] : \mathbf{F}_i' = \emptyset \wedge \forall j \neq i, \mathbf{F}_i' = \mathbf{F}_i\}$ and $N^{\max} := \{\mathbf{F}, \mathbf{F}' \in \mathcal{P}(\mathcal{I} \cup C)^k : \forall i \in [k], \mathbf{F}_i' = \mathbf{F}_i \text{ or } \mathbf{F}_i' = \emptyset\}$ we have $N \subset N^{\max}$.

Second, the pairs of databases $\mathbf{F}^{\max}, \mathbf{F}'^{\max}$ exhibited in Eq. 78 and Eq. 82 happen to belong to both N and $N^{\max}$, so the upper bounds hold. □

Instantiation. In particular, the upper bounds from the previous lemma apply when the attribution function A distributes a predetermined conversion value across impressions (*e.g.,* last-touch, first-touch, uniform, etc.), maps each impression to a bin (*e.g.,* $H(f)$ is a one-hot encoding of one of m campaign identifiers), and then sums up the value in each bin. The resulting report $\rho_r(D) \in \mathbb{R}^m$ contains a histogram of the total attributed conversion value per bin.

The first tightness result (Eq. 67) applies if there exists an impression that can be fully attributed. The second tightness result (Eq. 69) applies if there exists two impressions f_0, f_1 with different one-hot encodings, such that removing f_1 shifts the maximum attribution value $A^{\max}$ to f_0 (*e.g.,* in last-touch attribution).

Note that we allow $A^{\max}$ to have any value, and we don't require every database to be fully attributed. This is a slight generalization of [10], which defines an *attribution rule* that requires $\sum_{i=1}^{k} \sum_{f \in F_i \cap F_A} = 1$.