Experimental Design for Product Launches with Collaborative User Networks

Chiraag Kala
monu.kala@airbnb.com
Airbnb
New York, NY, USA

Jeffrey Wong
jeffrey.wong@airbnb.com
Airbnb
San Francisco, CA, USA

Abstract

Airbnb's hosting platform supports co-hosting and collaborative host teams, where multiple hosts can jointly manage listings. This collaborative network structure poses unique challenges for A/B testing. Conventional non-clustered experiment designs can violate the "stable unit treatment value assumption" (SUTVA) when collaborators influence each other, introducing interference that biases estimated treatment effects. To address these challenges, we propose a novel experimental design tailored for product launches with collaborative user networks. Our design combines network-level randomization with a launch model that allows control users to opt into treatment, enabling accurate measurement while respecting collaborative team structures and user autonomy. In this paper, we begin by examining the bias introduced by ignoring network effects within teams and analyze the bias-variance trade-off with a network-level assignment. To address non-compliance due to voluntary opt-in, we use a two-stage least squares (2SLS) model with network-level assignment as an instrumental variable. Our framework offers a scalable and adaptable solution for any platform with user networks where traditional experimentation methods risk introducing bias or degrading the user experience. By aligning robust causal inference methodologies with a user-centric deployment strategy, our framework demonstrates that organizations can achieve statistical precision without compromising on the product experience.

CCS Concepts

• **Information systems** → **Online marketplaces**; • **Computing methodologies** → **Causal inference**; • **Mathematics of computing** → *Graph algorithms*; • **Theory of computation** → *Experimental design.*

Keywords

Online Controlled Experiment, A/B Testing, Encouragement Design, Graph Networks, User Networks, Instrumental Variables, Two-Stage Least Squares (2SLS), Clustered Experimental Design, Clustered Standard Errors, SUTVA Violations

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
Conference acronym 'XX, Woodstock, NY
© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

ACM Reference Format:
Chiraag Kala and Jeffrey Wong. 2018. Experimental Design for Product Launches with Collaborative User Networks. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 6 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Airbnb's hosting model is designed to support collaboration. It enables hosts to add co-hosts who help manage listings—sharing responsibilities such as guest communication, calendar coordination, and onsite support through a unified set of tools accessible to the entire team. This functionality allows hosts to build teams, delegate tasks, and sustain high-quality guest experiences as their operations grow. Hosting on Airbnb, therefore, is not always a solo operation—it can also scale into a coordinated team effort when hosts choose to collaborate.

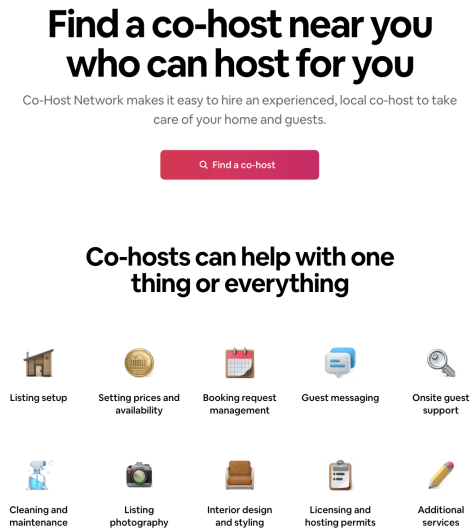


Figure 1: Airbnb's Co-Host Network landing page that highlights the range of services co-hosts can provide

Co-host relationships create interconnected user networks where the actions of one member can directly impact others within the same team. These networks create unique challenges for product experimentation that traditional A/B testing frameworks fail to address. First, from a user experience perspective, inconsistent access

to new features within a host network can lead to fragmented experiences. For instance, during a launch experiment if a host gains access to a new messaging tool while their co-host does not, it can result in confusion and operational misalignment, undermining the effectiveness of the launch and eroding user trust. Second, these inconsistencies can violate SUTVA, as hosts and their co-hosts may be assigned to different treatment arms, allowing one host's treatment status to influence the outcomes of connected co-hosts. This interference can lead to biased estimates of the treatment effect, compromising the validity of the experiment.

The impact of these challenges is amplified during high-visibility product launches at Airbnb. Major releases are often publicly announced and prominently surfaced within the product, generating strong user expectations around access. Under these conditions, traditional control group restrictions become problematic. When features are widely promoted but withheld from control users, it can lead to confusion and dissatisfaction—especially if peers or collaborators receive access. To address this, we adapt our experimentation design to account for both network interdependencies and concerns around equitable access—ensuring that control users have the ability to opt into new features during product launch experiments when feasible.

In this paper, we propose a novel experimentation design that combines network-level randomization with a launch model that allows control users to opt into treatment. Rather than assigning treatment at the individual level, we construct host networks—clusters of users connected through co-hosting—and randomize treatment at the network level. Each network is assigned either to a treatment group, in which users receive active prompts encouraging adoption of the new product experience, or to a control group, in which no prompt is shown but users have the ability to opt in voluntarily.

By assigning treatment consistently across all users within a network, we reduce interference bias and avoid inconsistencies that arise when collaborators receive conflicting experiences. To address noncompliance introduced by the opt-in functionality, we adopt a randomized encouragement design, using network-level assignment as an instrumental variable to obtain unbiased estimates of the treatment effect even when some users do not comply with the assigned condition.

This paper presents a practical framework for running product launch experiments in collaborative, networked user environments. We begin by examining the bias that arises when network effects are ignored in experimental designs. We then analyze the bias–variance trade-off introduced by network-level randomization and discuss its implications for statistical power. Finally, we detail our methodology for constructing host networks and estimating treatment effects under noncompliance using a two-stage least squares (2SLS) model within a randomized encouragement design. Our approach enables practitioners to run valid experiments that respect user relationships while preserving the integrity of both measurement and user experience.

2 Related Work

This work draws on three key areas of literature: (1) interference bias in online marketplace experiments, (2) network-aware experimental designs, and (3) encouragement-based experiments that address noncompliance. While interference has been widely studied in social network and marketplace settings, less attention has been given to collaboration networks, where users can collaborate—such as Airbnb hosts and co-hosts.

2.1 Interference Bias in Online Marketplaces

Interference bias can severely distort experimental results in marketplace platforms. Early work by [2] and [9] showed that naive A/B tests overestimate treatment effects by ignoring buyer-seller interactions. To address this, methods such as two-sided randomization [12], switchback designs [4], and interference-aware frameworks [11] have been proposed. However, these approaches are typically focused on bipartite or two-sided markets, not on collaboration among users on the same side of the platform.

2.2 Network Interference and Clustered Designs

Graph Cluster Randomization (GCR), introduced by [16], assigns treatment to clusters of connected users to reduce contamination. Subsequent work [8, 15] demonstrated its effectiveness across social platforms. We aim to build on these methods by adapting them to the context of product launches, where users may not comply with the treatment condition given.

2.3 Encouragement Designs and Noncompliance

Encouragement designs offer a principled approach for experiments with voluntary uptake. [10] formalized the use of instrumental variables in such settings to estimate causal effects despite noncompliance. These methods are well-suited to modern product launches, where users are exposed to global announcements and can opt in at will. Building on this, we provide a statistical framework for identifying and estimating causal effects under network-level encouragement, accounting for noncompliance and the complexities introduced by network randomization.

3 Understanding the Bias from Spillover Effects in Collaboration Networks

We begin by exploring the bias that can arise when treatment is assigned inconsistently across a user network. In collaborative settings like Airbnb, individual host randomization can lead to interference, where a host's outcome is influenced by their co-hosts' treatment status, violating SUTVA. This can result in control users being indirectly exposed to the treatment and treated users experiencing diluted effects if their collaborators are in the control group.

To illustrate this, we compare two designs: individual host randomization, which ignores co-hosting ties and allows for within-network contamination, and host network randomization, which assigns treatment at the network level.

3.0.1 Formalizing the Bias with Individual Host Randomization. Let $Y_h(1)$ and $Y_h(0)$ denote the potential outcomes for host h under treatment and control, respectively. Each host is assigned to treatment $T_h = 1$ or control $T_h = 0$. In *individual host randomization*, assignment occurs independently, without considering the network structure.

When randomization occurs at the individual level, network interference can arise due to variation in the proportion of treated co-hosts. We define the degree of exposure to treatment for host h as:

$$E_h = \frac{1}{|N_h|} \sum_{u \in N_h} T_u$$

where N_h represents the set of co-hosts connected to host h, and T_u is the treatment assignment of co-host u.

Exposure to differently treated co-hosts (i.e., $E_h > 0$ for control units or $E_h < 1$ for treated units) can influence observed outcomes, leading to biased estimation of the treatment effect. Specifically, control units may be affected by treated co-hosts, while treated units may be influenced by untreated co-hosts. We define this bias formally as:

$$\begin{aligned} B = {} & \mathbb{E}[Y_h(0) \mid T_h = 0,\ E_h > 0] - \mathbb{E}[Y_h(0) \mid T_h = 0,\ E_h = 0] \\ & + \mathbb{E}[Y_h(1) \mid T_h = 1,\ E_h < 1] - \mathbb{E}[Y_h(1) \mid T_h = 1,\ E_h = 1] \end{aligned}$$

where:

- The first term quantifies the shift in control group outcomes due to exposure to treated co-hosts [5].
- The second term quantifies the shift in treatment group outcomes due to exposure to untreated co-hosts.

These spillover effects introduce a systematic bias in the estimation of the treatment effect under individual-level randomization. In general, when the direction of the spillover aligns with the direction of the treatment effect (e.g., positive effects spill positively into control or negative effects spill negatively), it follows that:

$$B > 0$$

This implies a downward bias in the estimated treatment effect under individual-level randomization:

$$\hat{\tau}_{\text{indiv}} = \mathbb{E}[Y_h \mid T_h = 1] - \mathbb{E}[Y_h \mid T_h = 0] - B$$

This bias leads to an underestimation of the true treatment effect, making individual randomization unsuitable in cases where network spillovers are significant [1].

3.0.2 Eliminating Bias with Host Network Randomization. A natural solution to mitigate this bias is to randomize treatment at the host network level, ensuring that all hosts within a network C_i are assigned the same treatment condition. Under this approach, exposure variation is eliminated within host networks, such that for any host $h \in C_i$, the exposure level is deterministic:

$$E_h = \begin{cases} 1, & \text{if } T_{C_i} = 1 \\ 0, & \text{if } T_{C_i} = 0 \end{cases}$$

Substituting into the bias formula, where T_{C_i} represents the treatment assignment for the host network C_i that host h belongs to:

$$\begin{aligned} B = {} & \mathbb{E}[Y_h(0) \mid T_{C_i} = 0,\ E_h = 1] - \mathbb{E}[Y_h(0) \mid T_{C_i} = 0,\ E_h = 0] \\ & + \mathbb{E}[Y_h(1) \mid T_{C_i} = 1,\ E_h = 0] - \mathbb{E}[Y_h(1) \mid T_{C_i} = 1,\ E_h = 1] \\ = {} & 0 + 0 = 0 \end{aligned}$$

Consequently, the bias term vanishes ($B = 0$), yielding an unbiased estimate of the treatment effect:

$$\hat{\tau}_{\text{network}} = \mathbb{E}[Y_h \mid T_{C_i} = 1] - \mathbb{E}[Y_h \mid T_{C_i} = 0]$$

This approach aligns with prior work on network-aware experimental designs, demonstrating that randomization at the level of connected user networks can mitigate the bias introduced by within-network interference [8, 16].

4 Bias-Variance Trade-Off with Network Randomization

While network randomization reduces spillover bias, it can also increase variance and reduce statistical power. Assigning treatment at the cluster or network level leads to correlated outcomes within groups—such as hosts within the same host network—which can result in higher standard errors and lower statistical power [13].

In practice, however, this tradeoff can be favorable when spillover effects are substantial. Bias reduction in our case not only improves causal validity but can also increase the measured effect size, helping offset some of the loss in power. At Airbnb, co-hosting relationships introduce significant interference under individual randomization, leading to diluted treatment effects. This observation is consistent with the theoretical framework in Section 3, where we show that inconsistent assignment within networks can bias estimates downward. In our empirical analysis, we estimate this bias to be approximately 25% of the true effect, with the gain in effect size helping to counterbalance the increase in variance introduced by clustering.

To quantify the increase in variance caused by network assignment, we use the *design effect* (DE)—a standard metric that compares the variance under clustered assignment to that under individual user assignment. Formally, the design effect is defined as:

$$\text{DE} = \frac{\text{SE}^2_{\text{cluster}}}{\text{SE}^2_{\text{indiv}}} \tag{1}$$

In our experiments, the design effect ranged from 1.0 to 1.5.

To assess when cluster randomization improves statistical power despite higher variance, we derive in Appendix A that the key condition is:

$$B \geq \tau_{\text{true}} \left(1 - \frac{1}{\sqrt{\text{DE}}}\right), \tag{2}$$

where B denotes the bias under individual randomization and τ_{true} is the true average treatment effect (ATE).

This leads to a simple rule of thumb: cluster randomization yields higher or equivalent power when the expected spillover bias exceeds a threshold fraction of the true treatment effect, determined by the design effect. The following table summarizes the minimum bias threshold needed:

Design Effect (DE)	Minimum Bias Threshold (as % of ATE)
1.0	0%
1.5	18.3%
2.0	29.3%
3.0	42.3%
4.0	50.0%

In our case, the increase in measured effect size under network randomization (approximately 25%) was sufficient to offset the rise in variance (design effects between 1.0 and 1.5), resulting in a neutral or even favorable impact on statistical power.

Thus, our findings demonstrate that when spillover bias is substantial, network experiments can both improve internal validity and preserve—or even increase—statistical power. A full derivation of this tradeoff is provided in Appendix A.

5 Constructing Host Networks for Experimentation

To enable experimentation, we model Airbnb's host ecosystem as a user network based on co-hosting relationships. We identify networks of hosts through graph construction and connected component detection. In this section we discuss network modeling and other implementation details to ensure experimental validity.

We model the Airbnb host network as an undirected graph $G = (V, E)$, where V is the set of hosts and E represents co-hosting relationships. This graph-based representation allows us to identify clusters of interacting users by computing connected components—subgraphs in which each node is reachable from any other node within the component [14, 17].

Formally, a connected component is a maximal set of nodes such that each pair is connected by a path, and no additional node can be added without breaking connectivity. We detect these components using standard graph traversal algorithms such as Depth-First Search (DFS) or Breadth-First Search (BFS) [6]. Each traversal explores one component by visiting all reachable nodes from a starting point.

To perform this computation efficiently, we leverage the `NetworkX` library in Python, which offers built-in functions for graph construction and for identifying connected components. `NetworkX` supports seamless creation of graphs from edge lists and includes traversal algorithms based on both depth-first and breadth-first search.

Each component C_i is treated as an independent host network. This segmentation supports further analysis, such as computing network sizes, centrality of key hosts, and identifying highly interconnected groups. While edges could be weighted to reflect co-hosting strength, our implementation uses unweighted edges.

5.1 Practical Considerations for Experimental Use of Networks

5.1.1 Preventing Pre-Assignment Bias from Large Host Networks. While most host networks on Airbnb are small, a minority of outlier networks are large and more densely connected. These large networks typically form when a host is added as a co-host across listings managed by multiple other hosts, creating a highly connected structure with many members. These large networks pose a risk to experimental validity because their inclusion in either treatment or control can introduce *pre-assignment bias*. If a large network, particularly one responsible for a disproportionate share of bookings, ends up fully on one side of the experiment, it can skew outcome metrics independently of the treatment effect. This undermines the assumption that treatment and control groups are comparable at baseline.

To mitigate this risk, we apply the Louvain clustering algorithm to break large connected components into smaller, more tightly-knit subgroups. Louvain is a modularity-based community detection method that partitions graphs by maximizing the density of within-group connections relative to between-group links [3]. This approach ensures that hosts who collaborate frequently are grouped together, while also reducing the likelihood that any one network dominates the experimental outcome.

After using Louvain clustering to break apart very large host networks into smaller clusters, we tested for pre-assignment bias across key business metrics we track, including bookings, nights booked, and other supply-side metrics. Our post-assignment diagnostics confirmed that by segmenting large connected components, we maintained balance across these metrics, with no dominant cluster skewing the results. In addition to improving baseline comparability, this approach increased the effective sample size and increased the statistical power of the experiment.

For the connected components that were split into smaller clusters, there is a possibility of spillover bias if users from the same original network are assigned to different experimental arms. However, this is a minor concern in practice. Only a very small subset of hosts were part of large connected components that required splitting, and even within those, the vast majority of first-degree co-hosting relationships remained within the same resulting cluster. As a result, the risk of meaningful spillover due to fragmentation was limited, and the integrity of most host collaborations is preserved.

5.1.2 Network Stability Assumptions. Another key design consideration is ensuring that network structures remain stable throughout the experiment. Fortunately, host networks on Airbnb are generally stable over time. To preserve internal validity, we exclude users who switch networks during the experiment from the final analysis. Additionally, we restrict this methodology to treatments that are unlikely to affect a host's propensity to add or remove co-hosts, thereby reducing the risk that treatment exposure alters the network structure. This constraint helps maintain consistency in the underlying network and mitigates bias from endogenous structural changes.

6 Estimating Treatment Effects with Network-Based Encouragement Design

At Airbnb, we recognize and respect collaborative host teams, and during product launches, we aim to support user choice by enabling voluntary access to new features whenever feasible. Even when an experiment assigns a user or a network to the control group, we want to maintain fairness by still making the new experience available through voluntary opt-in. To balance this commitment to fairness and user autonomy with the need for rigorous experimentation, we use adopt a network-based encouragement design. Under this approach, entire host networks are randomly assigned either to a treatment group—actively encouraged to try the new features — or to a control group, which doesn't receive active prompts but individual users have the option to voluntarily opt-in to receive the new experience. Because this opt-in mechanism introduces noncompliance between assigned condition and actual treatment exposure, naïvely regressing outcomes on observed treatment may yield biased estimates. To address this, we apply a 2SLS model,

using the randomized network-level assignment as an instrument for actual treatment received. This approach enables us to estimate the causal effect of treatment on outcomes, even in the presence of partial compliance.

We construct the dataset by defining the following variables:

- Y_h: the outcome variable of interest (e.g., number of bookings);
- T_h: the actual treatment received by host h;
- $X_h(t_0)$: a vector of covariates measured prior to the start of the experiment;
- Z_h: the instrument or assignment variable indicating whether host h was in a treated network.

The index h denotes an individual host, and C_i refers to the network (cluster) to which host h belongs. All hosts within a given network C_i are assigned the same value of the instrument Z_h, ensuring that treatment assignment is determined at the network level.

To estimate the effect of the treatment on the individual host, we use the following two-stage least squares (2SLS) model:

$$Y_h = \alpha + T_h\beta_1 + X_h(t_0)\beta_2 + \varepsilon_h, \tag{3}$$

$$T_h = \nu + Z_h\pi_1 + X_h(t_0)\pi_2 + \gamma_h. \tag{4}$$

The second stage models the host-level outcome as a function of the actual treatment received, while controlling for relevant pre-treatment covariates. The first stage models the treatment uptake as a function of the randomized instrument and the same covariates. The use of an instrumental variable here addresses the fact that actual treatment receipt (T_h) may not perfectly align with the assigned condition (Z_h) due to noncompliance.

To address the correlation in outcomes among co-hosts within the same network, the error structure is clustered by C_i in order to model the within-network correlation. Clustering the data avoids understating the standard error on the treatment effect.

The coefficients in the first stage have a physical interpretation. When T is the treatment that was served to the host, and Z is the intent-to-treat, then the predicted values of the first stage represent the probability that a host, who we intend to treat, will be treated, also known as the compliance rate.

The key parameter of interest, β_1, captures the local average treatment effect (LATE) — the average effect of treatment on hosts whose treatment status was influenced by their assignment. This estimate is valid under standard instrumental variables assumptions: (1) relevance, meaning that the instrument Z_h must be correlated with the endogenous treatment T_h; (2) exclusion, requiring that the instrument affects the outcome Y_h only through its effect on T_h; and (3) monotonicity, which assumes that assigning a host or a network to the treatment group does not reduce its likelihood of receiving the treatment. These assumptions are generally plausible in network-based product rollouts where assignment is randomized at the network level, and there is no direct effect of assignment apart from inducing uptake of the feature.

Under full compliance, the model simplifies to ordinary least squares with clustered errors. In this setting, β_1 estimates the average treatment effect (ATE) for the entire host population rather than just the compliers.

7 Conclusion

This paper presents an experimentation framework tailored for product launches in environments with collaborative user networks. Traditional experimentation methods often require strict control over user exposure, which can conflict with platform goals around autonomy and user experience. Our approach addresses two central challenges: ensuring consistent treatment within interconnected user groups, and allowing voluntary access to new features through an opt-in mechanism. We also show that this methodology is statistically effective. While cluster-based designs typically increase variance and may reduce statistical power, we demonstrate that the corresponding increase in effect size—driven by lower spillover bias—can more than offset the loss in power. As a result, cluster-level experimentation proves robust, particularly in settings with high potential for interference or among platforms with inherently interconnected user bases.

Though developed in the context of Airbnb, this methodology is broadly applicable. Platforms such as social media networks, collaborative tools, and online marketplaces can adopt similar frameworks to balance rigorous measurement with a user-centric product experience. By aligning experimental design with the way users interact on the platform, organizations can enhance user trust, foster inclusion, and build more adaptive and respectful experimentation platforms at scale. As digital platforms become more interconnected, experimentation strategies must evolve accordingly. Our work serves as a starting point for future experimentation on platforms with user collaboration and interdependence, helping companies make smarter decisions without compromising on the the user experience.

Generative AI Usage Disclosure

In accordance with ACM authorship policies, the authors disclose that generative AI tools were used only for limited editorial purposes, such as language editing and grammar correction. No AI tools were used to conduct the research, design experiments, develop models, or generate scientific results.

Acknowledgments

We would like to thank Mae Qiu, Antoine Creux, Jordan Young, Nikos Diamantopoulos, Hector Lagos, and Zack Loebel-Begelman for developing the engineering solutions that enabled the implementation of this methodology, and for being exceptional engineering partners throughout the process. We would also like to thank Maggie Jarley, Navin Sivanandam, and Erik Iverson for their feedback and for their guidance in implementing this methodology at Airbnb.

References

[1] Peter M. Aronow and Cyrus Samii. 2017. Estimating Average Causal Effects Under General Interference, with Application to a Social Network Experiment. *The Annals of Applied Statistics* 11, 4 (2017), 1912–1947. doi:10.1214/16-AOAS1005

[2] Thomas Blake and Dominic Coey. 2014. Statistical Analysis of A/B Testing with Interference. In *Proceedings of the 15th ACM Conference on Economics and Computation*. ACM, New York, NY, USA, 459–459. doi:10.1145/2600057.2602880

[3] Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast Unfolding of Communities in Large Networks. *Journal of Statistical Mechanics: Theory and Experiment* 2008, 10 (2008), P10008. doi:10.1088/1742-5468/2008/10/P10008

[4] Iavor Bojinov, Chris Ramirez, and Edoardo M. Airoldi. 2022. Design and Analysis of Experiments in the Presence of Network Interference. *Biometrika* 109, 1 (2022),

1–18. doi:10.1093/biomet/asab017
[5] Jake Bowers, Mark Fredrickson, and Costas Panagopoulos. 2013. Reasoning about Interference Between Units: A General Framework. *Political Analysis* 21, 1 (2013), 97–124. doi:10.1093/pan/mps038
[6] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2009. *Introduction to Algorithms* (3 ed.). MIT Press, Cambridge, MA, USA.
[7] Allan Donner and Neil Klar. 2000. *Design and Analysis of Cluster Randomization Trials in Health Research.* John Wiley & Sons, Inc., New York, NY, USA.
[8] Dean Eckles, Brian Karrer, and Johan Ugander. 2017. Design and Analysis of Experiments in Networks: Reducing Bias from Interference. *Journal of Causal Inference* 5, 1 (2017), 1–29. doi:10.1515/jci-2016-0020
[9] Andrey Fradkin. 2015. Search Frictions and the Design of Online Marketplaces. (2015). https://ssrn.com/abstract=2261465 Available at SSRN: https://ssrn.com/abstract=2261465.
[10] Keisuke Hirano, Guido W. Imbens, Donald B. Rubin, and Xiao-Hua Zhou. 2000. Assessing the Effect of an Influenza Vaccine in an Encouragement Design. *Biostatistics* 1, 1 (2000), 69–88. doi:10.1093/biostatistics/1.1.69
[11] David Holtz, Andrew Yates, and Sinan Aral. 2021. Reducing Interference in Online Experimentation by Removing Existing Social Ties. *Nature* 595, 7867 (2021), 482–487. doi:10.1038/s41586-021-03647-0
[12] Ramesh Johari, Kevin Joseph, and Alex Liang. 2022. Experimental Design for Two-Sided Platforms. *Econometrica* 90, 2 (2022), 705–741. doi:10.3982/ECTA18542
[13] David M. Murray. 1998. *Design and Analysis of Group-Randomized Trials.* Oxford University Press, New York.
[14] M. E. J. Newman. 2018. *Networks.* Oxford University Press, Oxford, UK.
[15] Martin Saveski, Eugene Chou, Daniel M. Roy, Iyad Rahwan, and Himabindu Lakkaraju. 2017. Detecting and Reducing Bias in A/B Testing. In *Proceedings of the 26th International Conference on World Wide Web.* International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 679–688. doi:10.1145/3038912.3052662
[16] Johan Ugander, Lars Backstrom, Jon Kleinberg, and Jure Leskovec. 2013. Graph Cluster Randomization: Network Exposure to Multiple Universes. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* ACM, New York, NY, USA, 329–337. doi:10.1145/2487575.2487695
[17] Douglas B. West. 2001. *Introduction to Graph Theory.* Prentice Hall, Upper Saddle River, NJ, USA.

A Appendix: Power Tradeoff Derivation

The following derivation builds on standard formulations of statistical power under cluster randomization [7, 13], adapting them to derive a rule of thumb for when increase in effect size can offset reduction in power with variance inflation.

A.1 Notation

We define the following terms used:

- τ_{true}: The true average treatment effect (ATE).
- B: The spillover bias under individual user randomization.
- SE_{indiv}: Standard error under individual user randomization.
- $\text{SE}_{\text{cluster}}$: Cluster-robust standard error under cluster randomization.

A.2 Noncentrality Parameter (NCP) and Design Effect

Statistical power is governed by the noncentrality parameter:

$$\text{NCP} = \frac{\text{Effect Size}}{\text{Standard Error}}. \tag{5}$$

For individual randomization and cluster randomization, the respective noncentrality parameters are:

$$\text{NCP}_{\text{indiv}} = \frac{\tau_{\text{true}} - B}{\text{SE}_{\text{indiv}}}, \quad \text{NCP}_{\text{cluster}} = \frac{\tau_{\text{true}}}{\text{SE}_{\text{cluster}}}. \tag{6}$$

The design effect (DE) captures the increase in variance due to clustering and is defined as [7, 13]:

$$\text{DE} = \frac{\text{SE}^2_{\text{cluster}}}{\text{SE}^2_{\text{indiv}}}. \tag{7}$$

A.3 Threshold Condition for Power Preservation

To determine when cluster randomization yields higher or equal statistical power compared to individual randomization, we start by comparing their noncentrality parameters (NCP):

$$\frac{\tau_{\text{true}}}{\text{SE}_{\text{cluster}}} \geq \frac{\tau_{\text{true}} - B}{\text{SE}_{\text{indiv}}} \tag{8}$$

Substituting $\text{SE}_{\text{cluster}} = \text{SE}_{\text{indiv}} \cdot \sqrt{\text{DE}}$, where DE is the design effect, we get:

$$\frac{\tau_{\text{true}}}{\text{SE}_{\text{indiv}} \cdot \sqrt{\text{DE}}} \geq \frac{\tau_{\text{true}} - B}{\text{SE}_{\text{indiv}}} \tag{9}$$

Multiplying both sides by SE_{indiv} to eliminate the denominator:

$$\frac{\tau_{\text{true}}}{\sqrt{\text{DE}}} \geq \tau_{\text{true}} - B \tag{10}$$

Rearranging terms to isolate B:

$$B \geq \tau_{\text{true}} \left(1 - \frac{1}{\sqrt{\text{DE}}}\right) \tag{11}$$

This final inequality provides the threshold condition: cluster randomization offers higher or equal power when the spillover bias B under individual randomization is at least $\tau_{\text{true}} \left(1 - \frac{1}{\sqrt{\text{DE}}}\right)$.

Received XX May 2025; revised XX XX 2025; accepted XX XX 2025