Efficient Online Evaluation

Eugene Kharitonov

(based on joint work with Aleksandr Vorobev, Craig Macdonald, Pavel Serdyukov, Iadh Ounis)

Yandex

a bit about me

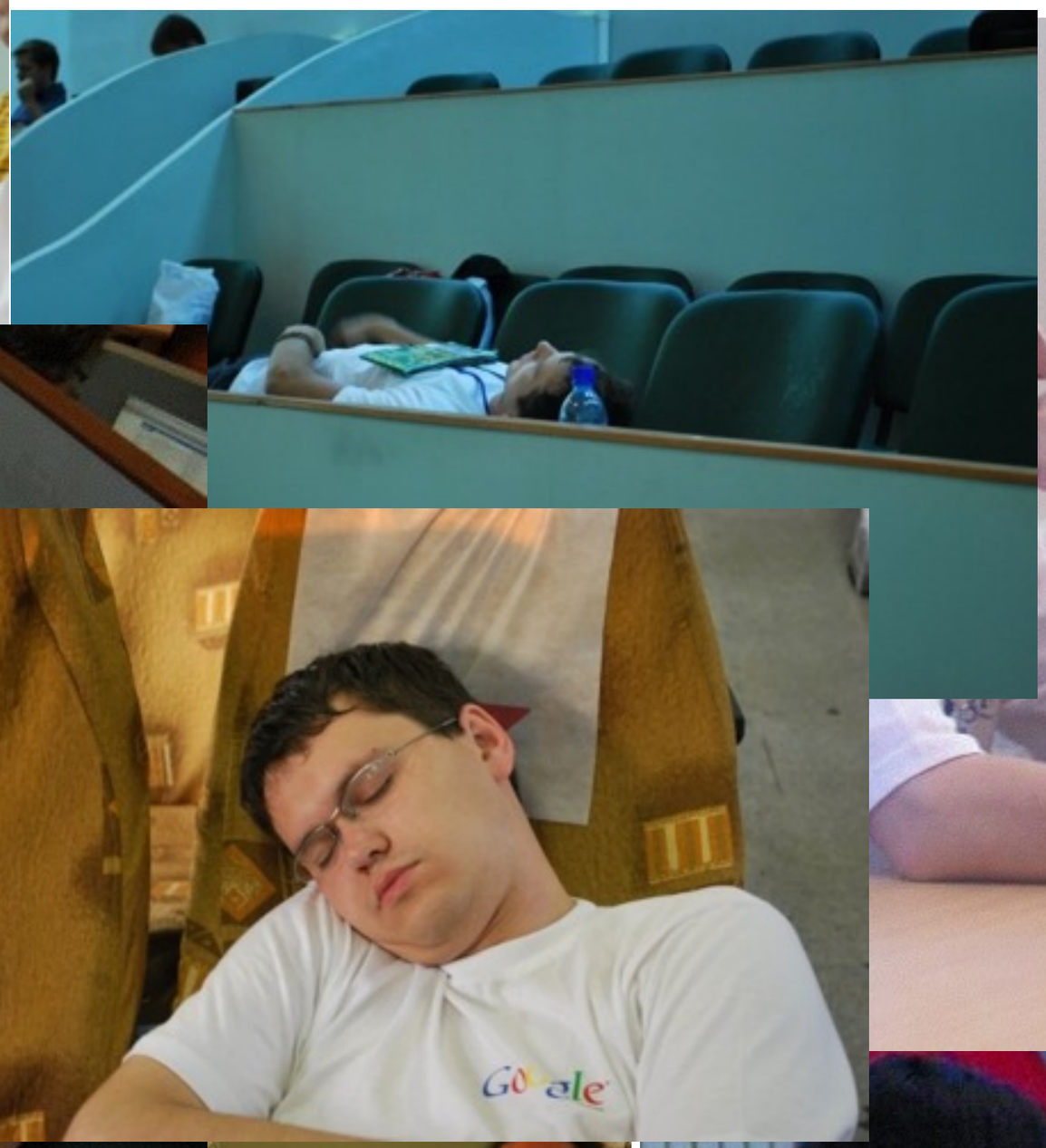
Currently:

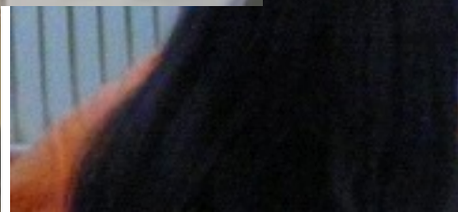
- Applied Researcher, Yandex
- 3rd year PhD student, University of Glasgow

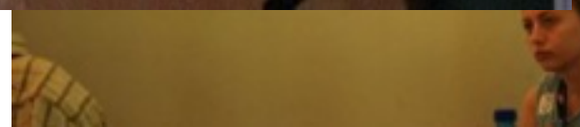
Earlier:

- Participated in Russir in 2008 and 2010!

russir 2008







russir 2010

I ♥ russir

- Exciting scientific part:
    - Ended up being an IR researcher myself!
- Insane social part:
    - Made a lot of friends
    - Married to a participant of Russir 2008

outline

Introduction

- online evaluation 101
- why efficiency is so Important?

Increasing the Online Evaluation Efficiency

- Generalized Team Draft
- sequential testing

Conclusions

outline

Introduction

- **online evaluation 101**
- why efficiency is so Important?

Increasing the Online Evaluation Efficiency

- Generalized Team Draft
- sequential testing

Conclusions

online evaluation 101

Suppose, we've implemented a change in a search engine:

- new learning to rank method
- new ranking feature
- change in the user interface
- …

Evaluation problem: will it improve the users' experience? Is it worth deploying at all?

online evaluation 101

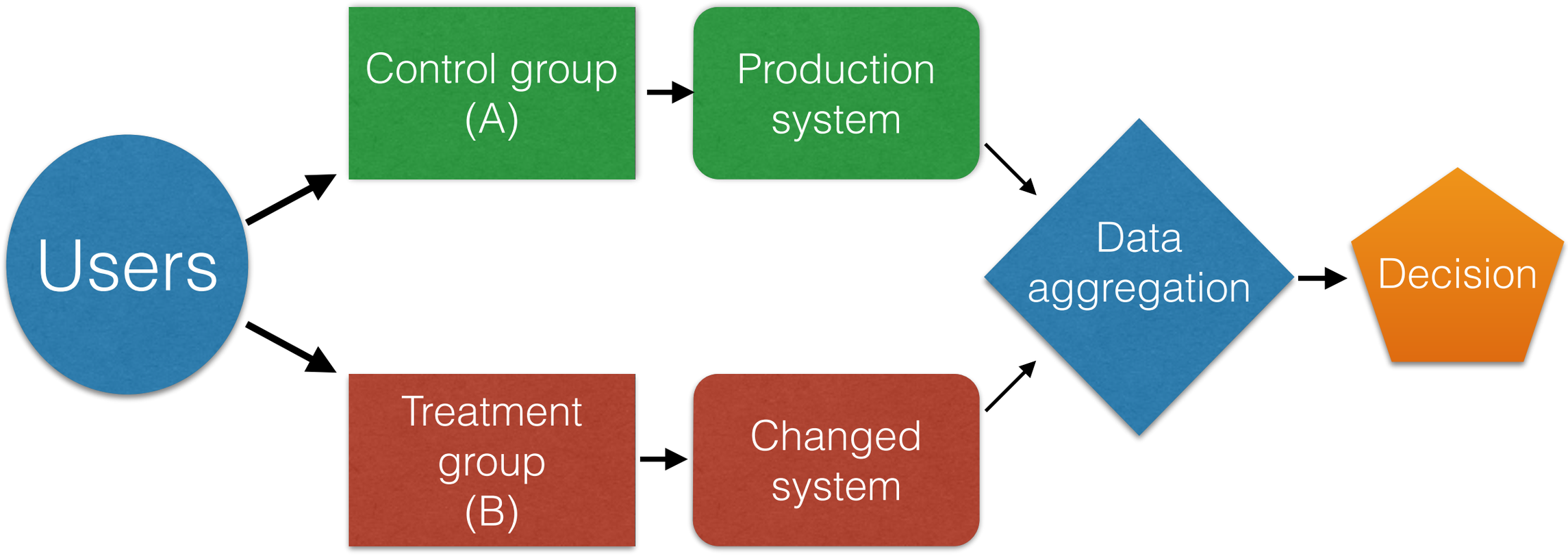
Offline

- Build a model of the user behavior
- Predict if they will be more satisfied with the changed version of the search engine than with the current version

Online

- Treat some of the users by a version of the search that is changed in some specific way
- Based on their behavior, infer if they are more likely to prefer the changed system

A/B testing

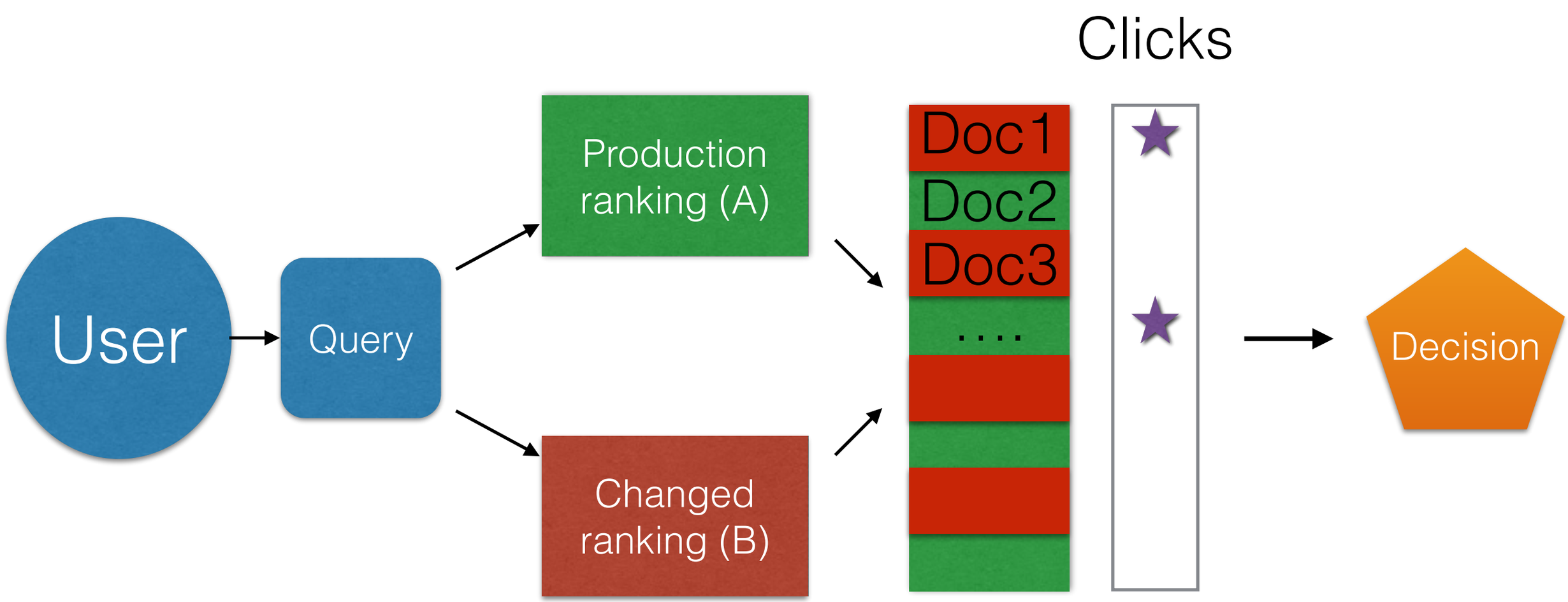

Users
Control group (A)
Production system
Treatment group (B)
Changed system
Data aggregation
Decision

A/B testing

Online metrics used:

- Abandonment rate
- Sessions per User
- Probability of Switching to another search engine
- User Engagement
- …

interleaving



interleaving

Metrics used:

- Relative difference in number of clicks received by the results from A and B

- Ratio of the sessions with the results from B getting more clicks

- …

statistical testing

But what if the change in the metrics is only due to a random chance?

- We use a statistical test to check if the observed change is statistically significant

statistical testing

We formulate two hypotheses (informally):

H_0 (null hypothesis):

- there is **no difference** between the tested systems

H_1 (alternative hypothesis):

- there is a **difference**

statistical testing

p-value = the probability of observing a difference in the metric at least as extreme as observed if H_0 holds (i.e. there is no difference between systems)

- pre-select acceptable significance level α (e.g. 10^{-3})
- start an experiment
- if the observed p-value is less than α then we reject H_0

A/B vs Interleaving

	A/B tests	Interleaving
Idea	Treat different users with different modifications of the search engine	Treat the same user with a combination of the results from both alternatives
Applicability	Very general (UI, ranking, new products, verticals, …)	Ranking only
Metrics used	Click-based, session-based, user-based, etc	Click-based only (somewhat restrictive)

So why do we need interleaving?

online evaluation efficiency

It turns out that:

- *interleaving is more sensitive* = evaluating the same change using interleaving requires 10x-100x times less data than the corresponding A/B test

- it requires less data = allows us to use the resource of user sessions more efficiently

online evaluation efficiency

Informal explanation:

- In A/B tests, different users are treated with different systems

- In interleaving, the same user compares the systems

    - the noise due to user variance is removed

A/B vs Interleaving

	A/B tests	Interleaving
Idea	Treat different users with different modifications of the search engine	Treat the same user with a combination of the results from both alternatives
Applicability	Very general (UI, ranking, new products, verticals, …)	Ranking only
Metrics used	Click-based, session-based, user-based, etc	Click-based only (somewhat restrictive)
Efficiency	Not too efficient	**Very efficient**

outline

Introduction

- online evaluation 101
- **why efficiency is so Important?**

Increasing the Online Evaluation Efficiency

- Generalized Team Draft
- sequential testing

Conclusions

why efficiency is Important

«*At Microsoft's Bing, the use of controlled experiments has grown exponentially over time, with over 200 concurrent experiments now running on any given day*» Kohavi et al., Online Controlled Experiments at Large Scale, KDD 2013

why efficiency is Important

Running 200 experiments:

- 10% of the query traffic per experiment for two weeks = 5 experiments per week = 40 weeks*
- 5% of the query traffic per experiment for two weeks = 10 experiments per week = 20 weeks*

* Only a motivational example: sometimes the same user might participate in several experiments at the same time + the number of the experiments reported by Bing might span several markets.

why efficiency is Important

- Number of experiments grows
- Each experiment consumes some resources (user sessions)
- The duration of the experiments limits the evolution of the search engine
    - faster a change is evaluated, faster it can be deployed

why efficiency is Important

- More than a half of the tested changes are either useless or harmful
- On average, the users who participate in an A/B or an interleaving experiment where the tested change B is worse than the production system A, have somewhat degraded experience

why efficiency is Important

Reducing the duration of the online experiments = increasing the online evaluation efficiency is **important** since:

- we do not want to harm our users
- we want to evolve faster

questions?

outline

Introduction

- online evaluation 101
- why efficiency is so Important?

Increasing the Online Evaluation Efficiency

- Generalized Team Draft
- sequential testing

Conclusions

approaches to increase efficiency

Two complimentary approaches:

- Reducing the variance of the observed metrics
- Improving the way statistical testing is performed

reducing variance

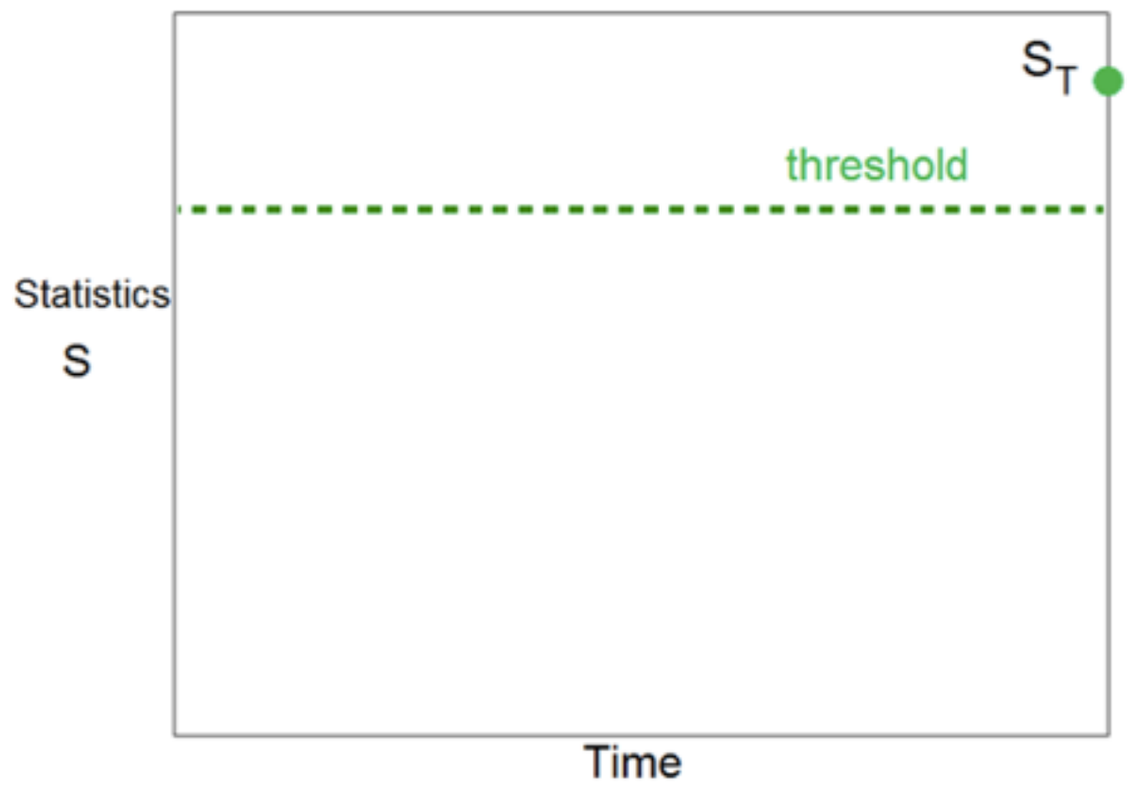
Intuition:

- a higher noise and spread in the observed metric implies that we need more data to average this noise out

- we have only 100% of the search traffic, thus we can only get more data by increasing the experiment's duration

So it's better to have a metric with a lower variance

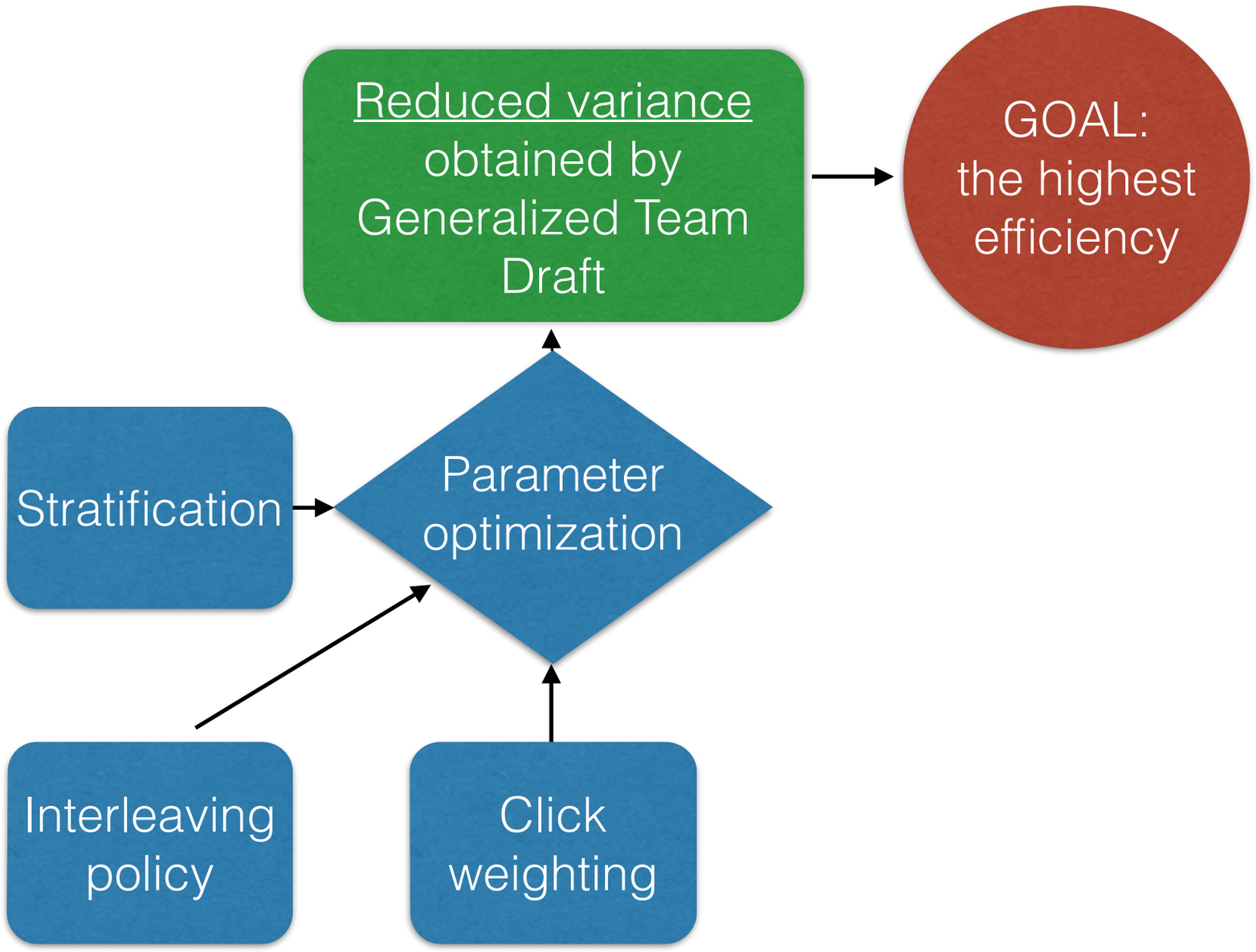
reducing variance



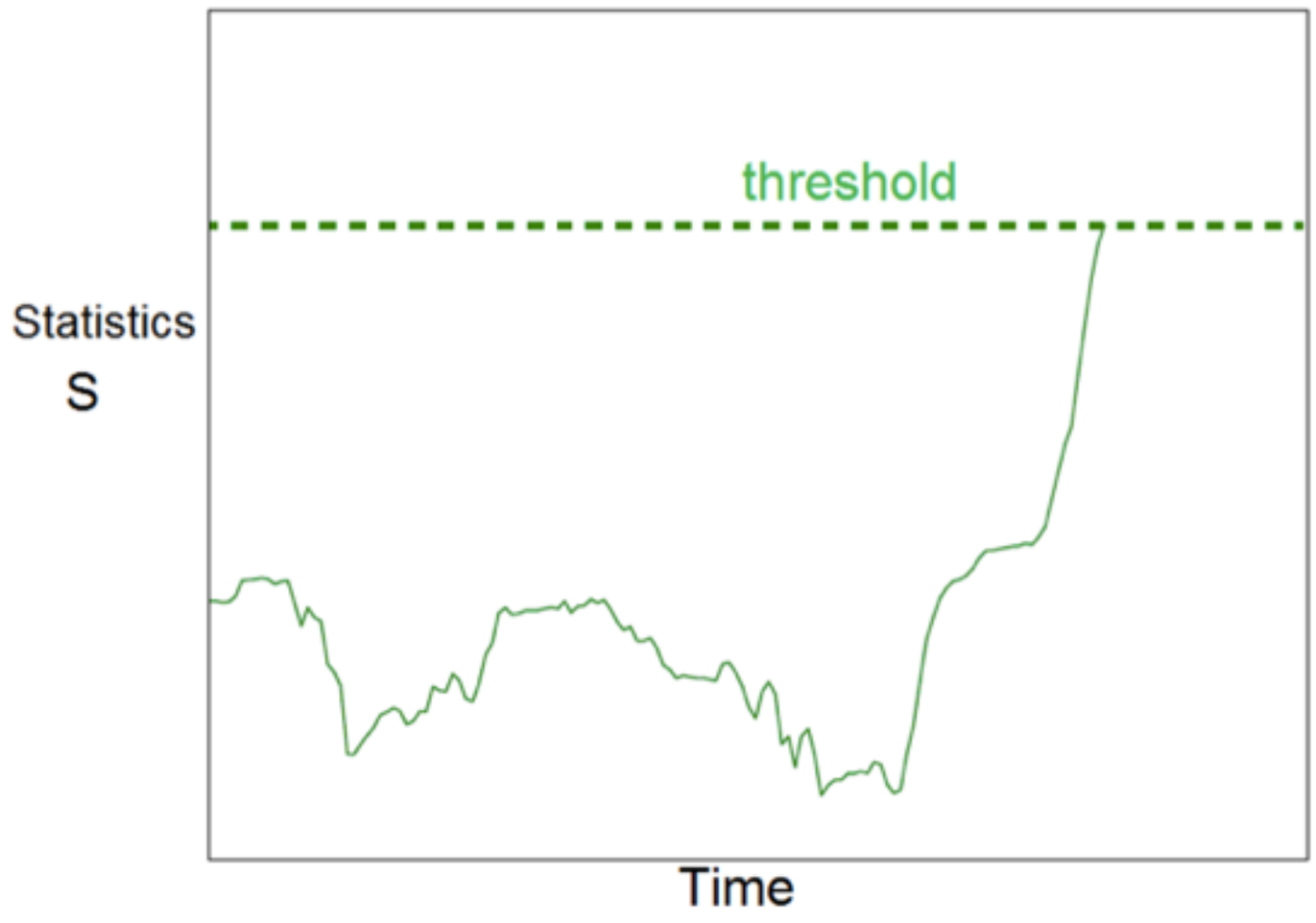
If we fix the size of the difference between the systems we want to detect then

$$time \propto sessions\ required \propto \frac{variance^2}{difference^2}$$

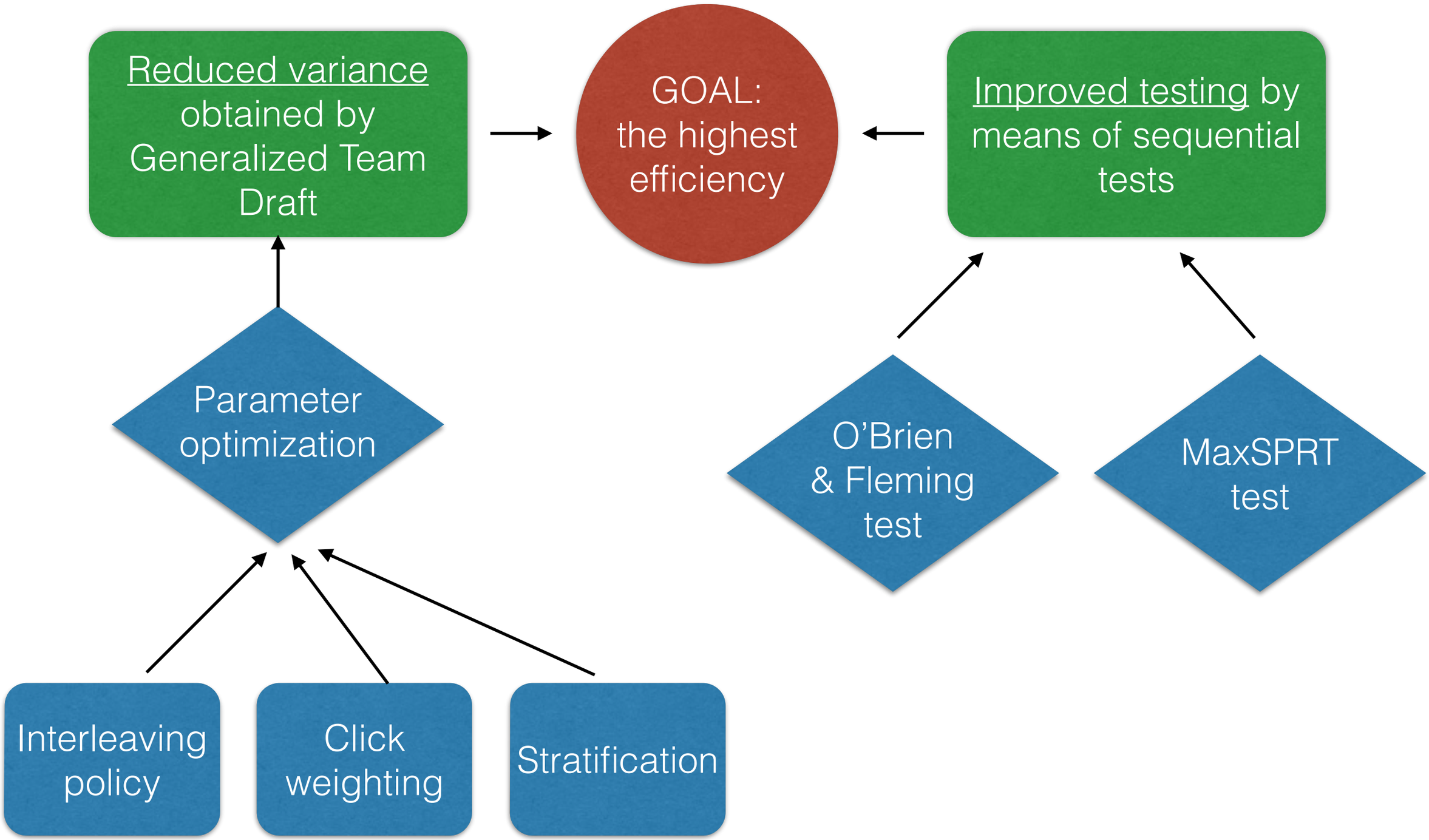
map of the talk



improving statistical decision criteria



map of the talk



outline

Introduction

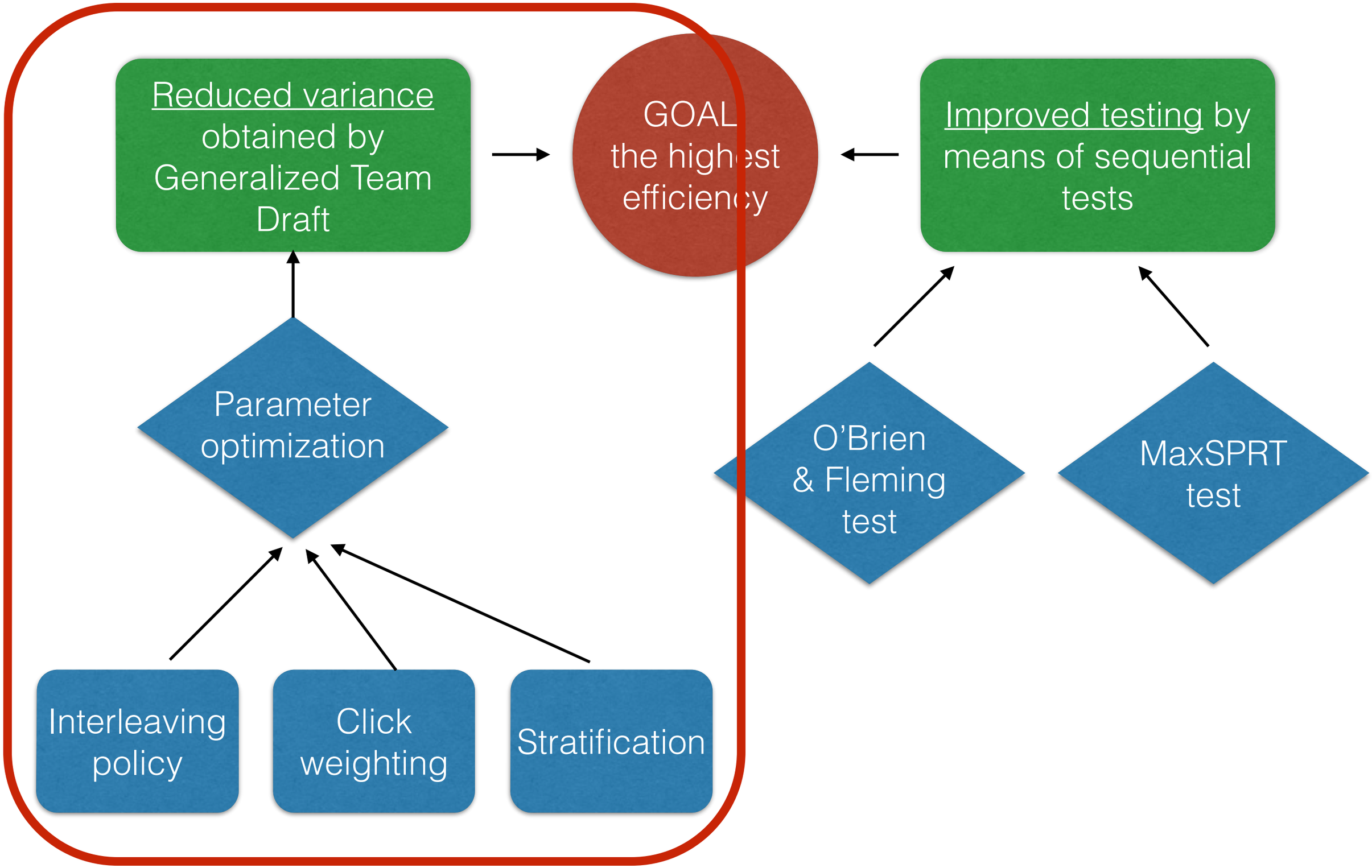
- online evaluation 101
- why efficiency is so Important?

Increasing the Online Evaluation Efficiency

- **Generalized Team Draft**
- sequential testing

Conclusions

map of the talk



intuition

Two cases:

(a)

1. A user clicked on a result
2. Stopped her search session after reading it

(b)

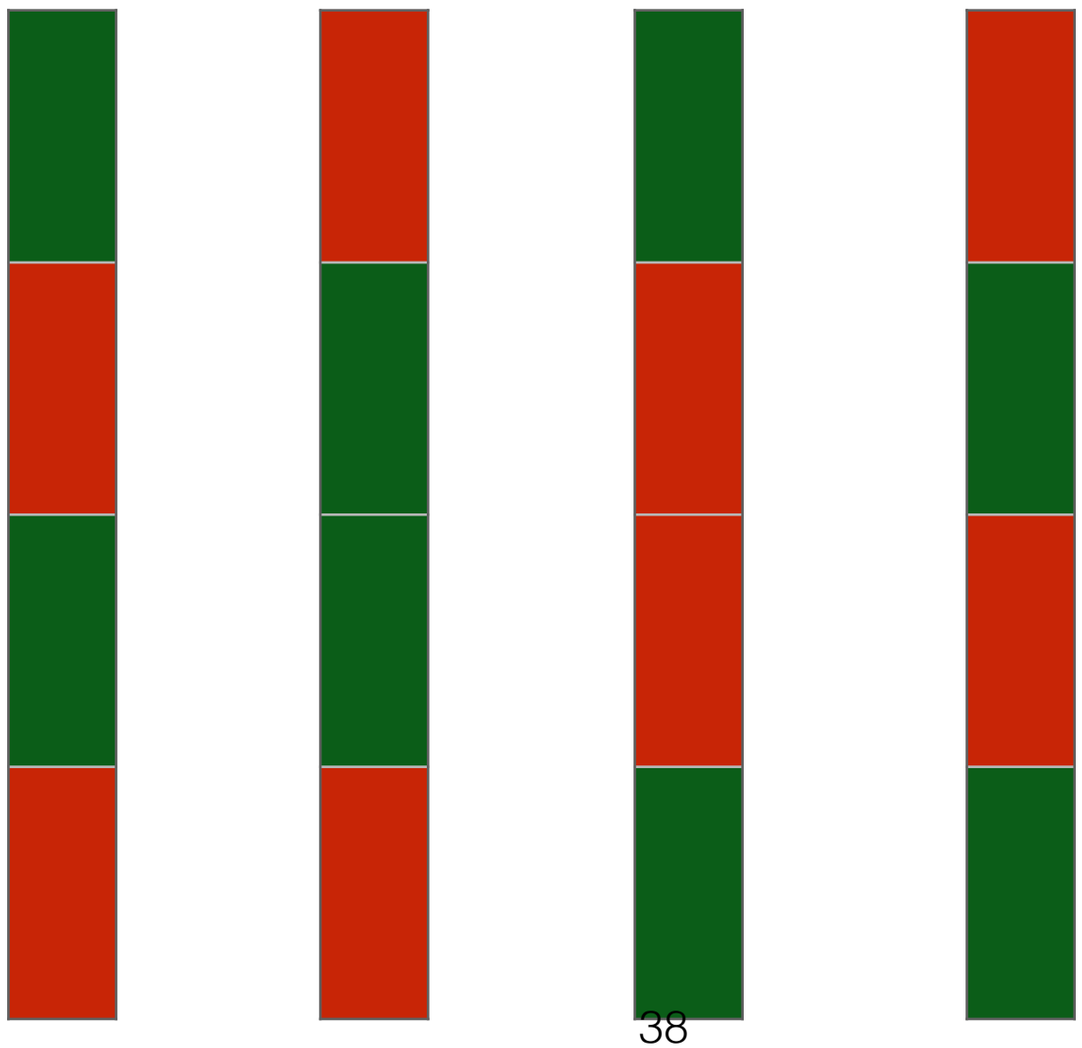
1. A user clicked on a result
2. 10 seconds later, clicked on another result

Which case is more likely to be informative?

Idea: represent a click with a feature vector describing it, learn how to weight it to form a click score

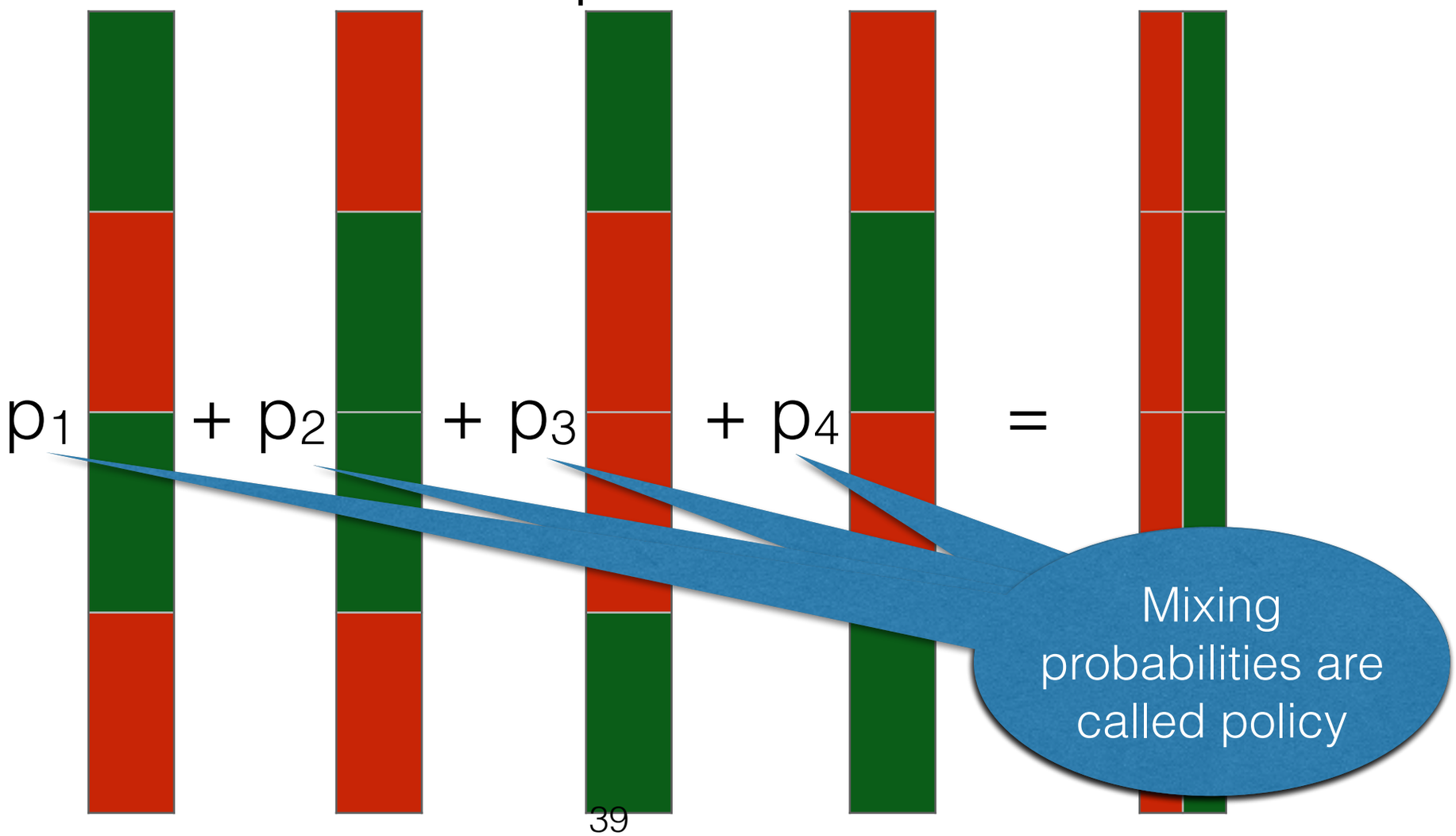
intuition

In interleaving, for each query we have several mixed result pages:



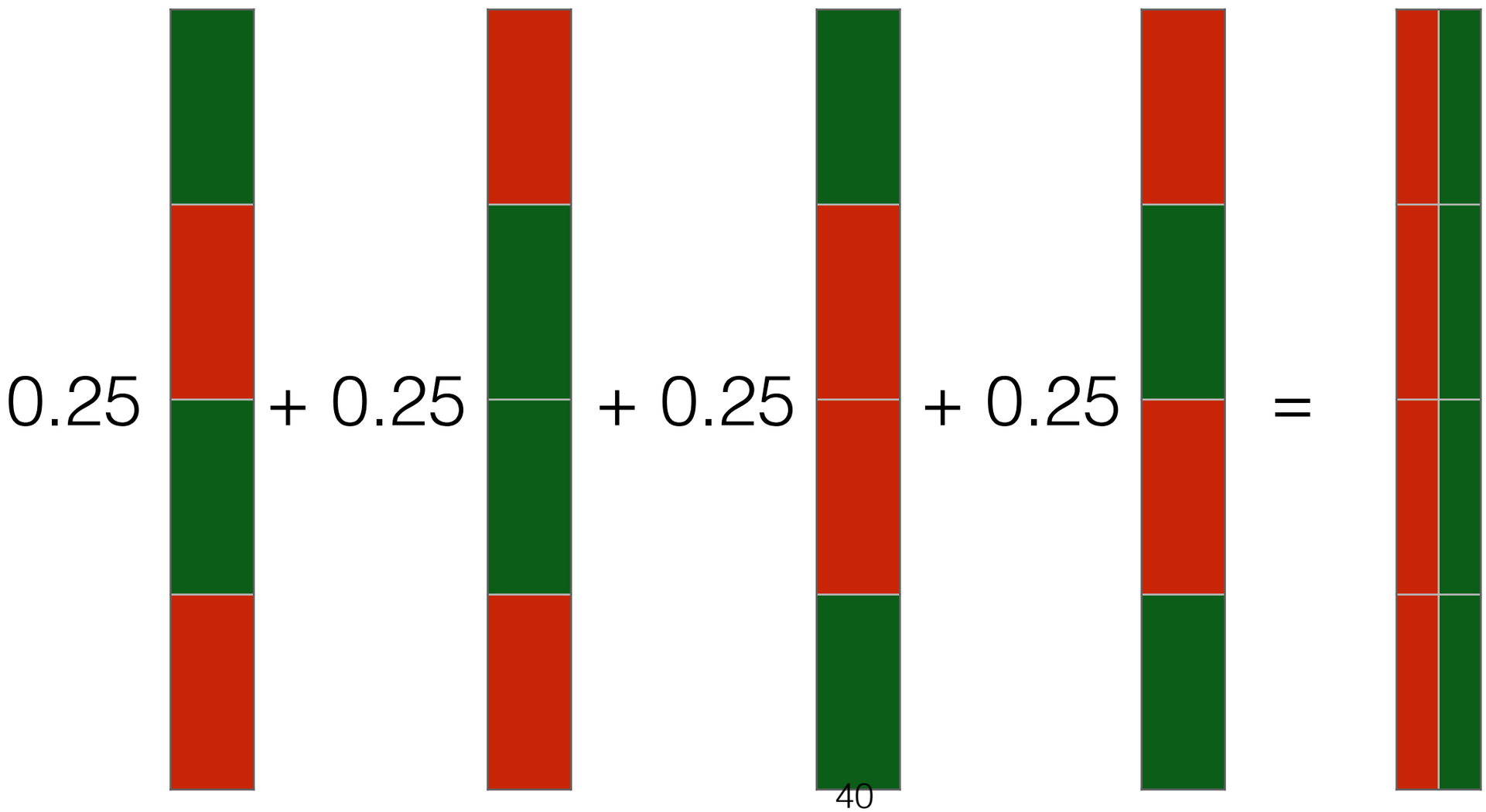
intuition

These pages are randomly demonstrated to the users, so that for each position the chances to get a result from A and B are equal:



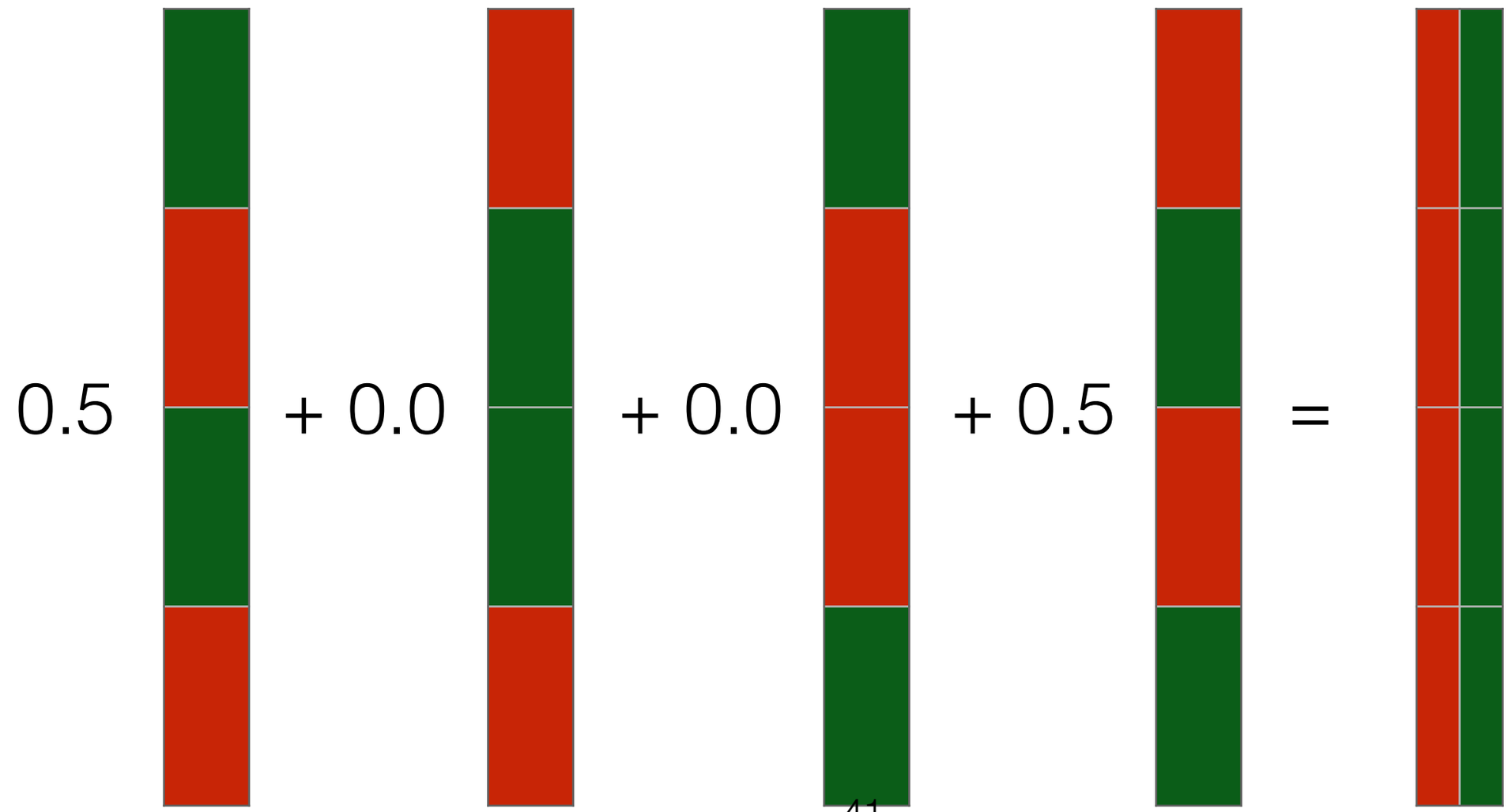
intuition

Team Draft uses the uniform policy $\{p_i\}$:



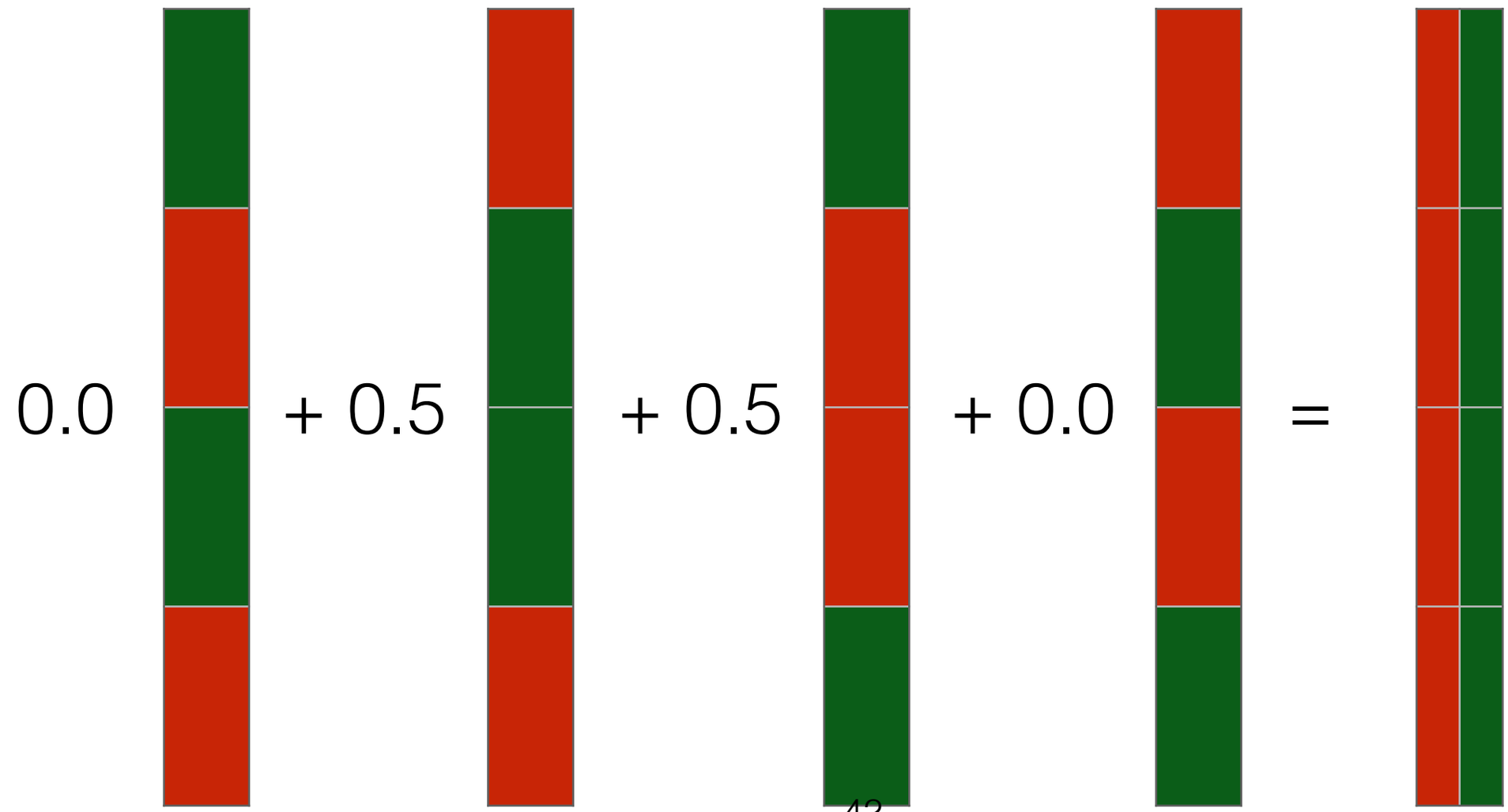
intuition

But other policies are also possible:



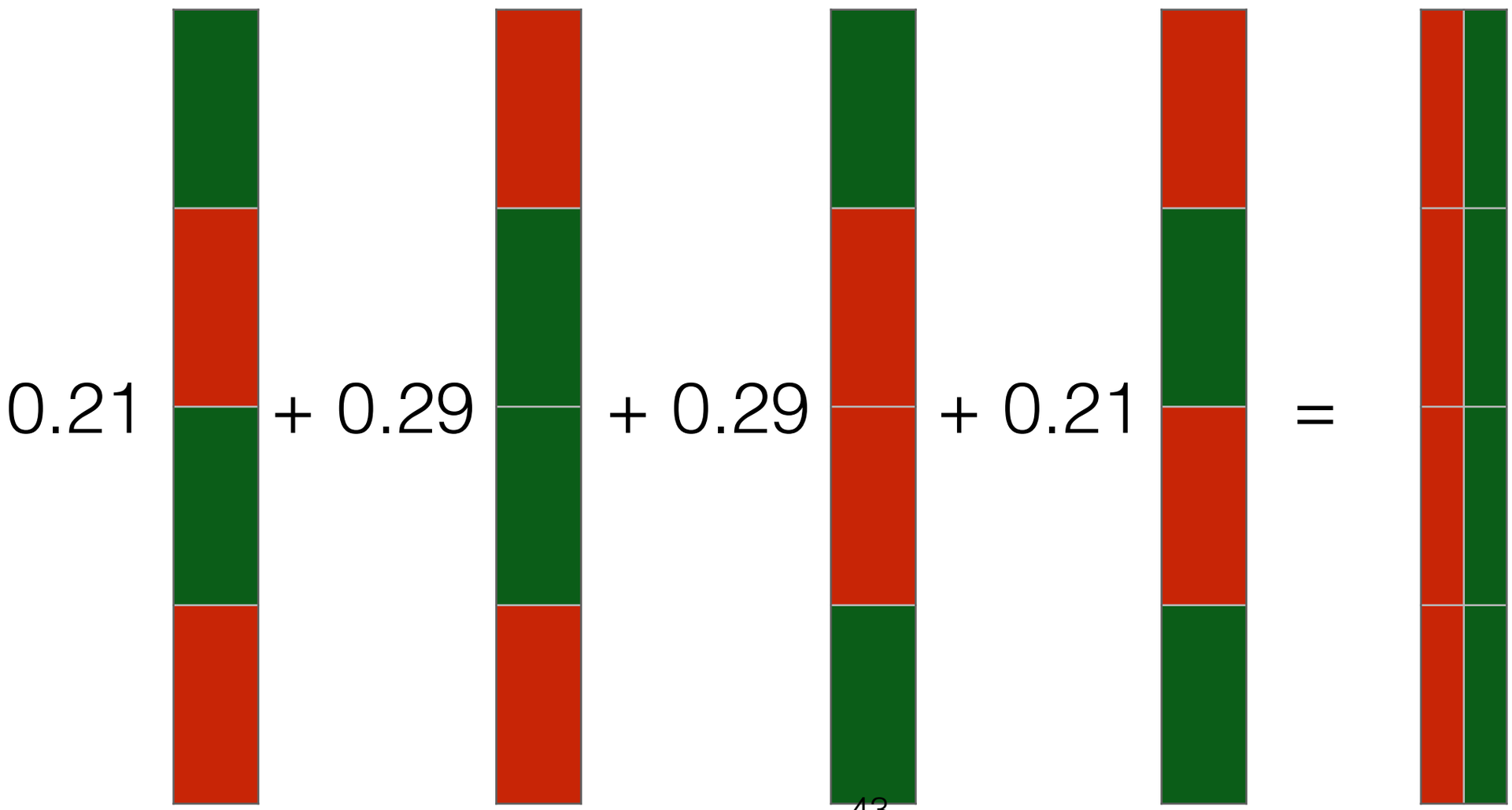
intuition

But other policies are also possible:



intuition

But other policies are also possible:



research question

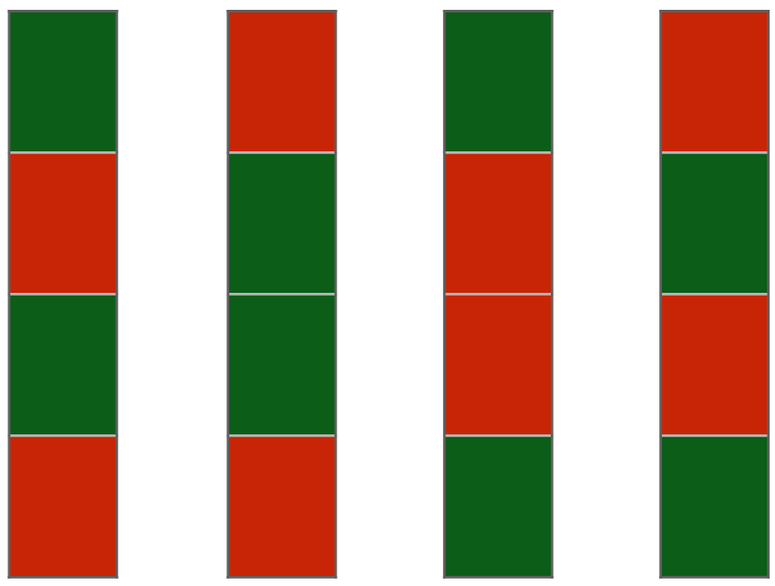
Is it possible to combine

- a freedom to select the interleaving policy
- a freedom to weight clicks differently

to build a more efficient interleaving algorithm?

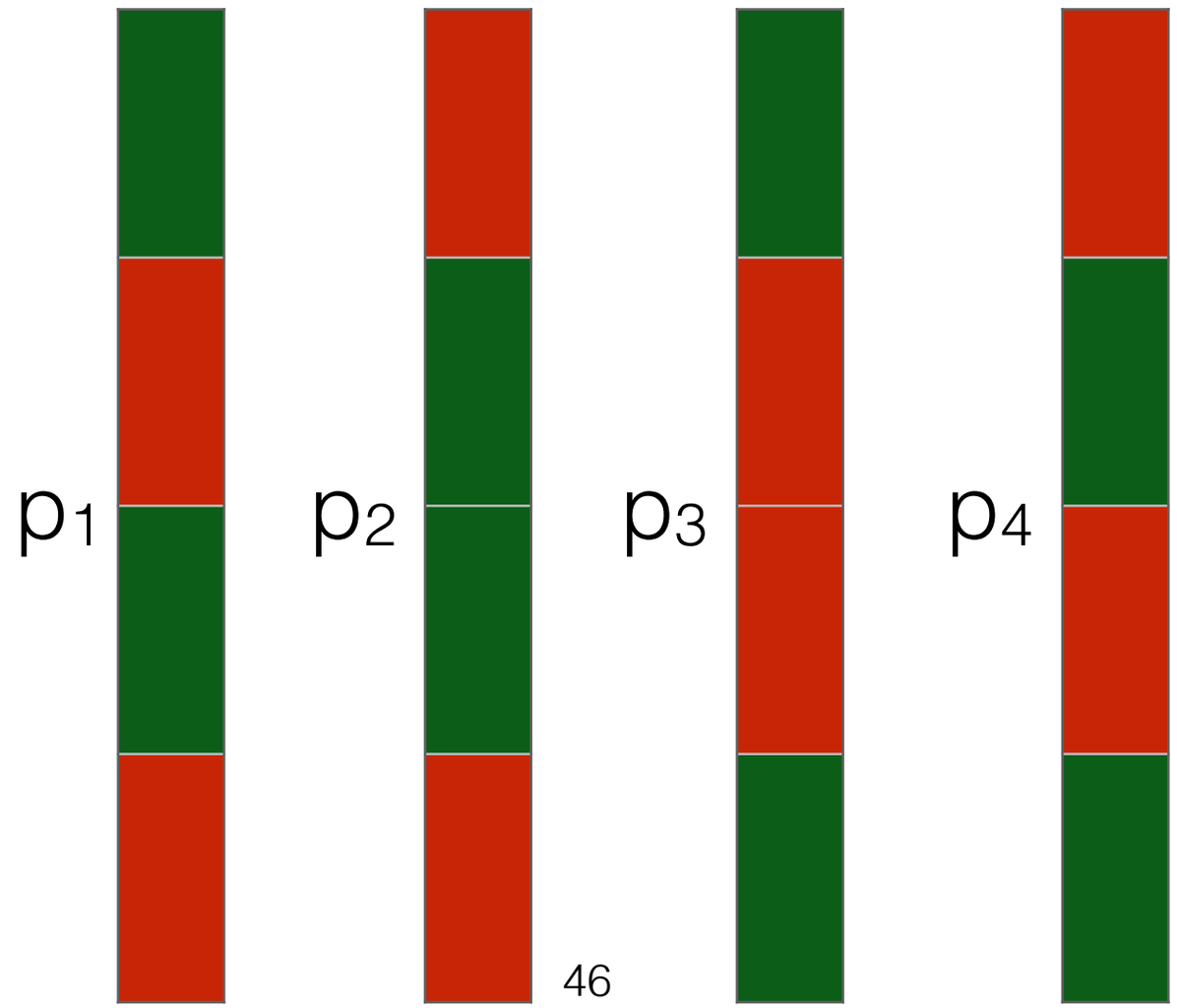
proposed framework

The set of the result pages associated with the distribution of the «teams» (**B** or **A**) on the corresponding page *{(L_i, T_i)}*



proposed framework

The interleaving policy $\pi = \{p_1, p_2, p_3, p_4, \ldots\}$ is a distribution that defines the probability of showing a particular combination (L_i, T_i) to a user



proposed framework

A function ϕ that maps a user click *c* on an interleaved result page to its feature representation is denoted $\phi(c)$.

T(c) is an auxiliary indicator that equates to 1 if the clicked result came from B, or -1 otherwise.

proposed framework

A scoring rule that maps a sequence of clicks in the interaction q to the differen of the alternatives B and A. The meter:

$$S = S(q; w) = \sum_{c \in q} T(c) \cdot w^T \phi(c)$$

Indicator of the clicked alternative (1 if B, -1 if A)

Score in an interaction *q*

Sum over clicks in *q*

Weights

The feature representation of *c*

proposed framework

After running an experiment *e* we calculate the experiment outcome as the mean score over all interactions:

$$\Delta(e) = \frac{1}{|Q|} \sum_{q \in Q} S(q; w)$$

All interactions in an online experiment

The score in a particular interaction *q*

unbiasedness

But

- what if we show only interleaved result pages where the first result always comes from A?

- what if we have a click feature «the result comes from A» and it has its weight higher than the feature «the result comes from B»?

Some policies and weight vectors can result in biases:

- if we are not careful, we might systematically favor one of the alternatives (B or A) due to the design of the interleaving algorithm not due to its better performance

- incorrect evaluation results

unbiasedness

A game-theoretic analogy:

- a malicious user wants to spoil our experiment
- he selects a randomized click sequence without knowing the result pages we are going to show
    - «click on the 3rd result, wait 30s, click …»
- we select a way to randomize our result pages (i.e. select the interleaving policy) without knowing the user's sequence
    - our goal is to randomize in such a way that no preference is inferred (neither A > B nor B > A) from the user's behavior

unbiasedness

Lemma 1:

For a feature representation function ϕ and a policy π to satisfy the unbiasedness requirement, it is sufficient that

- ϕ is independent from L_i
- For any position of the result list, the probabilities of observing a result from A and B should be equal

proposed framework

If we use the Team Draft-based result pages with *2m* results on a page:

- the number of independent constraints grows as *m + 1*
- the dimensionality of the policy space grows as 2^m
- 32 - 5 - 1= 27 «degrees of freedom» for the standard ten results per page
- can be used to find the most efficient policy

proposed framework

We can find the policy π and the feature vector w that maximize «sensitivity» = confidence in the outcomes of the experiments E that were performed earlier:

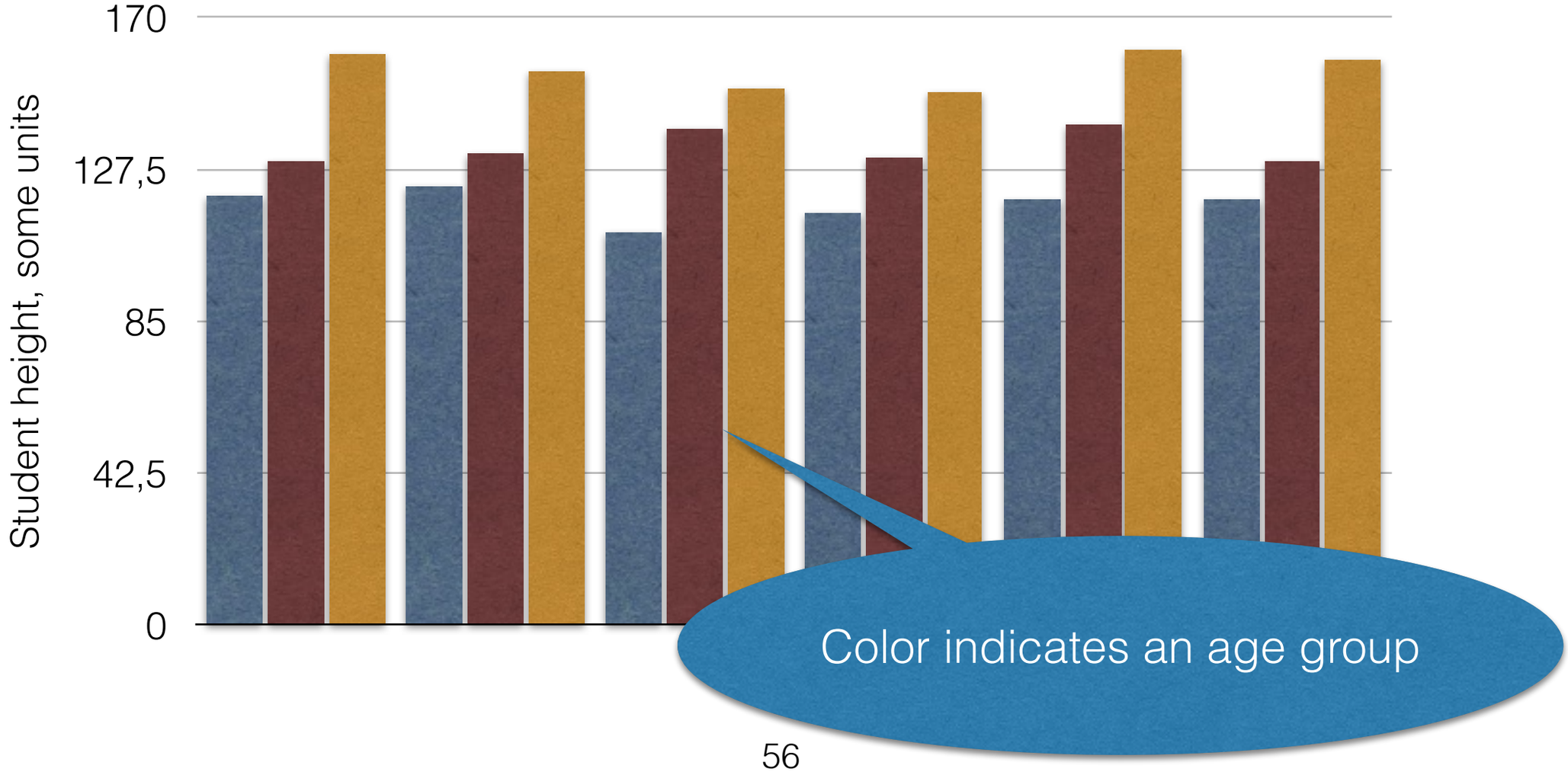
$$\hat{\pi}, \hat{w} = \arg\max_{\pi, w} D(E, \pi, w)$$

Confidence: score divided by sqrt(variance)

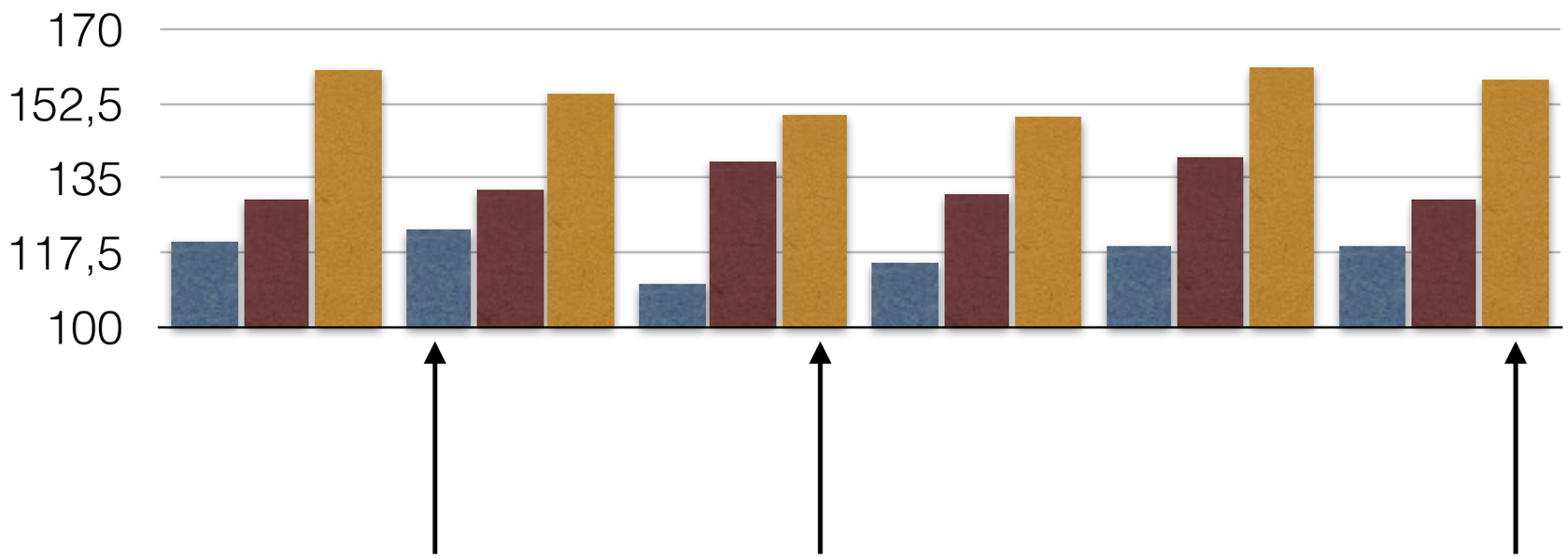

WE NEED TO GO
DEEPER
memegenerator.net

stratified sampling

Assume we want to find the mean height of students in a school, *but we can measure height of <u>three students only</u>*



simple sampling

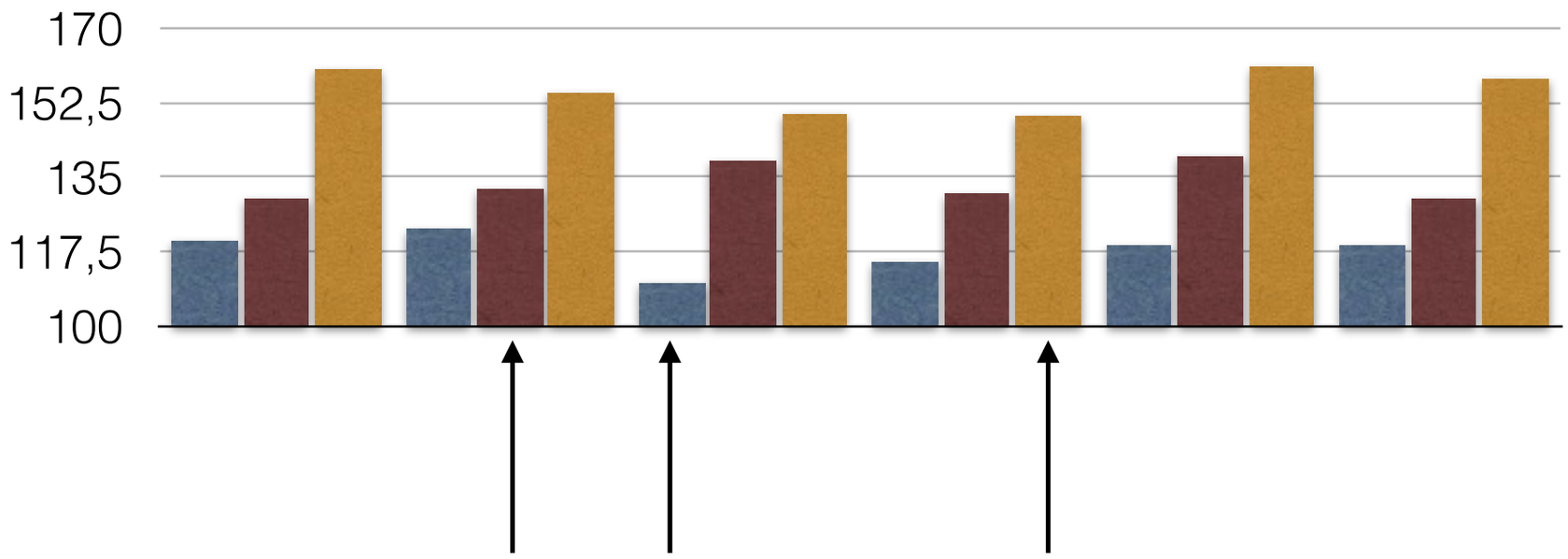


Simple sample mean: 146.7

True mean: 135.6

(note: no students from the red age group)

stratified sampling



Stratified sample mean: 130.3

True mean: 135.6

(we randomly select one student from each age group and average their heights)

«strata» = «cluster» = age group

stratified scoring

The probability of showing of a particular result page

$$\Delta_s(e) = \sum_i \pi_i \cdot \frac{1}{|Q_i|} \sum_{q \in Q_i} S(q; w)$$

Sum over all possible result pages (age groups)

The mean score for the i-th result page (mean height for a particular group)

stratified scoring

- The stratified estimate reduces (or at least does not increase) variance of the outcome:

$$var[\Delta(e)] \geq var[\Delta_s(e)]$$

- (reminder: experiment's duration is proportional to variance, so we want to minimize variance)

- it also greatly simplifies the optimization problem

stratified scoring

Variance of the simple outcome

Intra-strata variance (non-negative)

$$var\left[\Delta(e)\right] = \frac{\sum_i \pi_i \cdot var_i[S] + \sum_i \pi_i (\mathbb{E}_i[S] - \sum_i \pi_i \cdot \mathbb{E}_i[S])^2}{N}$$

$$\geq \frac{1}{N} \sum_i \pi_i \cdot var_i[S] = var\left[\Delta_s(e)\right]$$

Inner-strata variance

Variance of the stratified outcome

putting everything together

- We optimize parameters π and w to maximize our confidence in the stratified outcome of earlier performed experiments
- We use the optimal parameters $\hat{\pi}, \hat{w}$ for future experiments

$$\hat{\pi}, \hat{w} = \arg\max_{\pi, w} \frac{\pi^T X w}{\sqrt{w^T \left(\sum_i \pi_i \cdot Z_i \right) w}}$$

Per-strata means of the features

Policy

Per-strata covariance of the features

Feature weights

features

Feature family	id	Description
Rank-based		Transformations of the click's rank, normalized by the number of clicks
	1-10	position indicators, $f_i = \mathbb{I}\{rank = i\}$
	11	rank
	12	$\sqrt{rank}$
	13	$log(rank)$
	14	$\mathbb{I}\{rank > 4\}$
	15	$\mathbb{I}\{rank > d\}$, where d is the number of identical results in the tops of A and B
Dwell time-based		Indicators of the dwell time (seconds), normalized by the number of clicks
	16	$\mathbb{I}\{dwell \leq 30\}$
	17	$\mathbb{I}\{dwell \in (30, 60]\}$
	18	$\mathbb{I}\{dwell \in (60, 90]\}$
	19	$\mathbb{I}\{dwell \in (90, 120]\}$
	20	$\mathbb{I}\{dwell > 120\}$
Order-based		Indicators of the click's position in the interaction
	21	is the click first
	22	is the click last
Linear score-based		after applying the scoring rule F4, these features represent the (normalized) number of clicks the results from B received
	23	$f_{23} = 1$
	24	$f_{24} = 1/n$, where n is the total number of clicks

dataset

# exp	B > A	mean sessions	median sessions	mean days	median days
67	30	840K	620K	9.8	8.0

baselines

Simple classic Team Draft:

- difference in number of clicks (Linear)
- relative difference in the number of clicks on B and A (NLinear)
- ratio of the sessions with B winning over A (Binary)
- same, but the clicks on the top results that are identical between A and B are ignored (Deduped)

Machine-learned baselines:

- optimize *w* under the fixed uniform policy to maximize linear difference of scores between B and A, no stratification
- optimize *w* under the fixed uniform policy to maximize the confidence in the outcome, no stratification

metrics

- z-score (confidence) in the outcome of the experiments in the hold-out set (which are not used for training):

$$Z = \frac{\Delta_s(e)}{\sqrt{var[\Delta_s(e)]}} = \frac{\Delta_s(e)}{\sqrt{\sum_i \pi_i \cdot var_i[S]}} \sqrt{N}$$

- Normalize by the z-score of the Linear baseline

- Relative z-score value of *X* implies that our approach requires X^2 times less data to achieve the same level of confidence than Linear

methodology

Ten-fold cross-validation:

- 10% of the dataset is used for evaluation
- 90% of the dataset is used for training
- The process is repeated 10 times
- Same splits for all approaches

results

Non-stratified

	Linear	NLinear	Binary	Deduped	L_m	L_z
Mean	1.00	1.03	1.10	1.88	1.34	2.14
Median	1.00	0.93	0.98	1.59	1.20	1.80

Stratified

	Linear	NLinear	Binary	Deduped	L_m^s	L_z^s	F_m	F_z
Mean	1.06	1.16	1.22	1.88	1.39	2.28	1.38	**2.45**$^\diamond$
Median	1.04	1.03	1.10	1.60	1.24	1.96	1.23	**2.05**$^\diamond$

results

Non-stratified

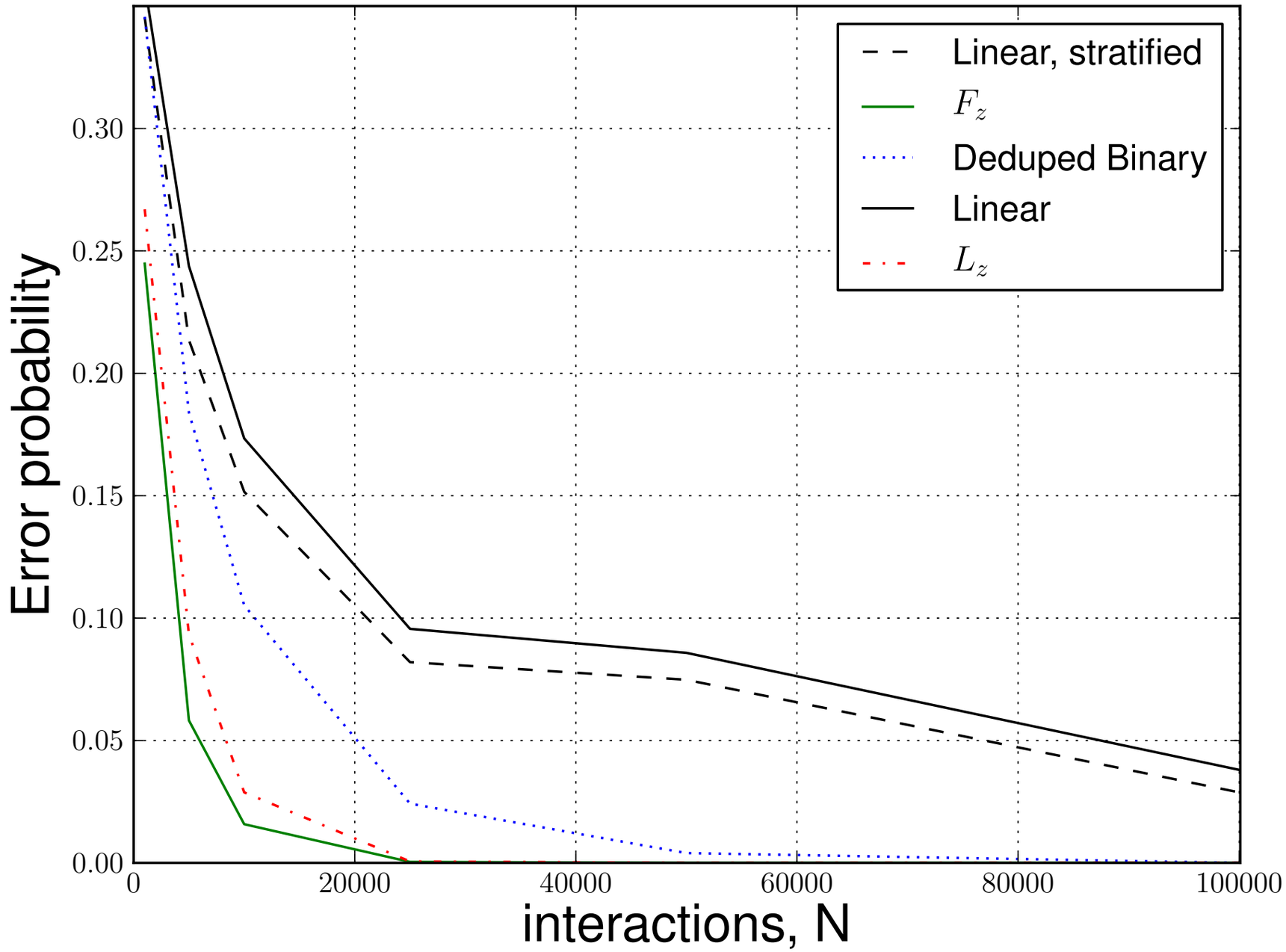
	Linear	NLinear	Binary	Deduped	L_m	L_z
Mean	1.00	1.03	1.10	1.88	1.34	2.14
Median	1.00	0.93	0.98	1.59	1.20	1.80

Stratified

	Linear	NLinear	Binary	Deduped	L_m^s	L_z^s	F_m	F_z
Mean	1.06	1.16	1.22	1.88	1.39	2.28	1.38	**2.45**$^\diamond$
Median	1.04	1.03	1.10	1.60	1.24	1.96	1.23	**2.05**$^\diamond$

2.05^2 = **4.20** times less data than Linear, and **1.30** times less data than the best baseline.

results


Linear, stratified
F_z
Deduped Binary
Linear
L_z
Error probability
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
20000
40000
60000
80000
100000
interactions, N

Generalized Team Draft

- Stratification helps us both to improve efficiency and to simplify the optimization problem

- We can considerably improve efficiency by optimizing the interleaving parameters: policy and click weights

- Check our CIKM 2015 paper:

    - Generalized Team Draft can be applied for image search

    - More exciting technical details & tables

questions?

outline

Introduction

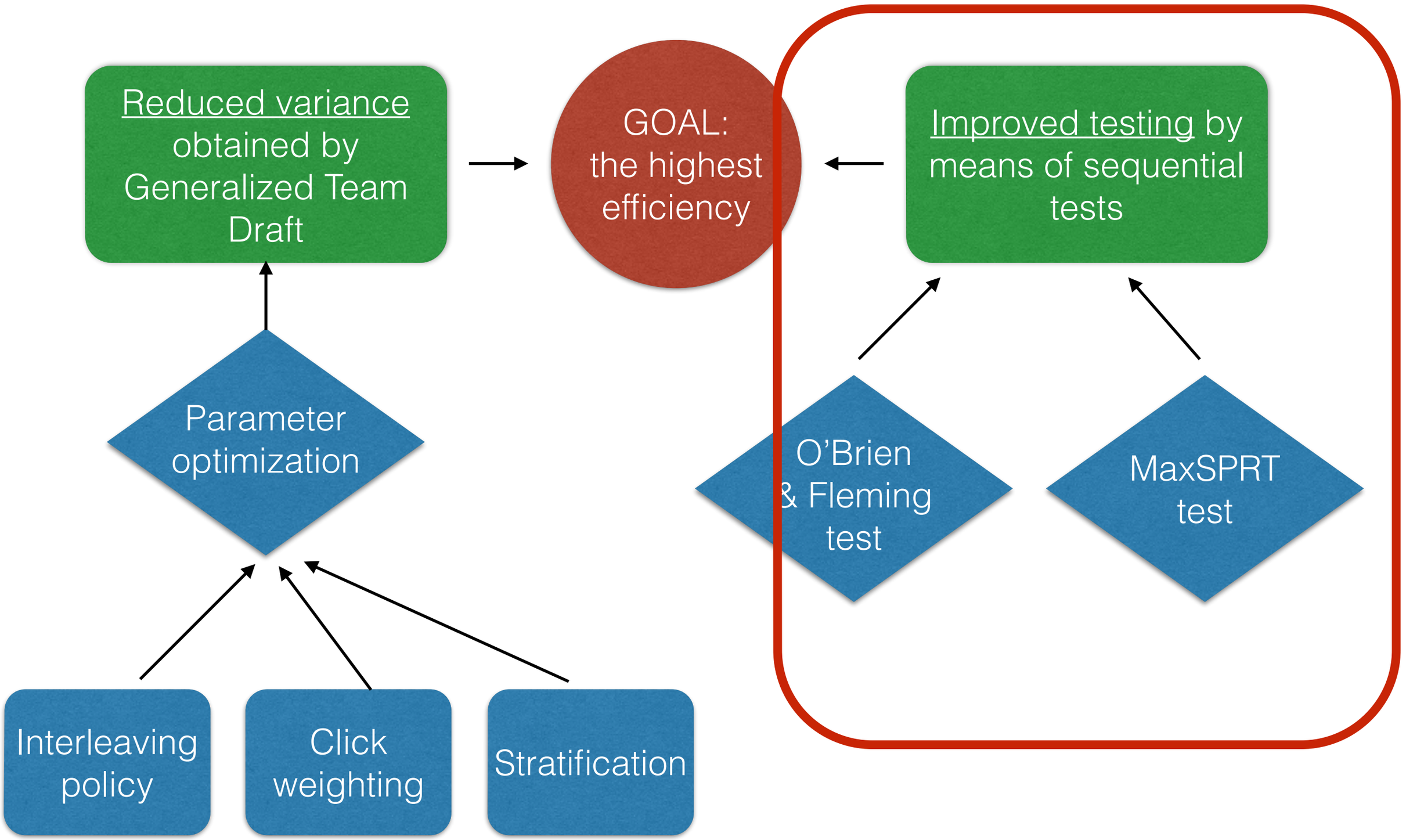
- online evaluation 101
- why efficiency is so Important?

Increasing the Online Evaluation Efficiency

- Generalized Team Draft
- **sequential testing**

Conclusions

map of the talk



statistical testing

We formulate two hypotheses (informally):

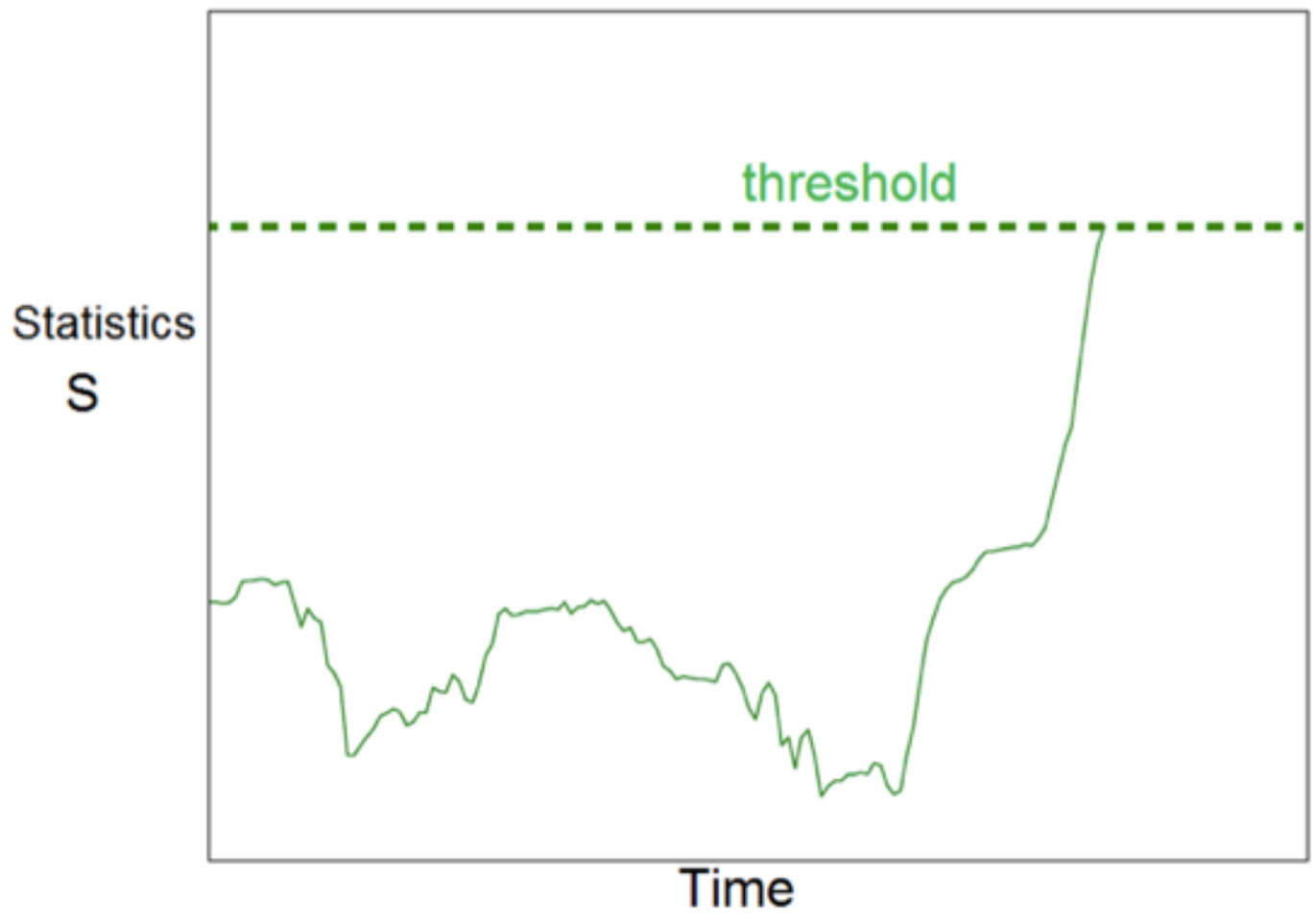
H_0 (null hypothesis):

- there is **no difference** between the tested systems

H_1 (alternative hypothesis):

- there is a **difference**

improving statistical decision criteria



sequential testing framework

We split the experiment in *N* equal time periods

After each period *i*:

- if the test statistic S_i exceeds threshold *b*:

  stop the experiment, reject H_0, accept H_1

- if N periods have finished:

  accept H_0, reject H_1

- continue

sequential testing framework

Problem: How can we specify S_i and b given the significance level α?

Monte Carlo

If H_0 completely specifies the distribution of the observed metric, then we can use the Monte Carlo approach:

- Repeat:
    - Simulate data by generating observations from H_0
    - Calculate the test statistics
- Select *b* to be *(1 - α)* percentile of the calculated statistics (i.e. in *α* of the simulations H_0 is rejected)

A/A experiments

We can use artificial «experiments» where both alternatives are identical (A/A experiments) as a source of the data generated from H_0

- Repeat many times:
    - Use data from A/A experiments as observations from H_0
    - Calculate the test statistics
- Select *b* to be *(1 - α)* percentile of the calculated statistics (i.e. in *α* of the observations H_0 is rejected)

interleaving

We assume that the possible outputs x of an observation (session) are:

- A won, i.e. got more clicks, $x = -1$
- B won, i.e. got more clicks, $x = 1$
- tie, i.e. A and B got equal number of clicks, $x = 0$

$$H_0 : \mathbb{E}x = 0$$

O’Brien & Fleming test

O'Brien & Fleming test

- Repeated test based on chi-sq

Number of sessions with B getting more clicks

$$S_i = i \cdot \frac{(wins_i^A - wins_i^B)^2}{T_i \cdot \widehat{D}[x]} \sim i \cdot \chi^2$$

Number of the observations (sessions)

Estimate of the variance

O’Brien & Fleming test

- Repeated test based on chi-sq

Number of sessions with B getting more clicks

$$S_i = i \cdot \frac{(wins_i^A - wins_i^B)^2}{T_i \cdot \widehat{D}[x]} \sim i \cdot \chi^2$$

Number of the observations (sessions)

Estimate of the variance

- How to find the corresponding threshold *b*?

O’Brien & Fleming test

Given a required significance level α and the number of periods N, we use Monte Carlo approach to learn the thresholds:

- Repeat:
  - draw U_1, U_2, … independently from the standard normal distribution
  - $U_m^2 = \max\{U_1^2, (U_1 + U_2)^2, \ldots, (U_1 + \ldots + U_N)^2\}$
- b = (1 - α) percentile of the distribution of U_m^2

MaxSPRT

MaxSPRT

Assume ties are broken randomly

- Denote the probability of B winning a comparison as $p = P(x = 1) + 0.5\,P(x = 0)$

$$S_i = \log \frac{P(Data_i \mid H_1)}{P(Data_i \mid H_0)}$$

Data observed before i-th stop

MaxSPRT

Assume ties are broken randomly

- Denote the probability of B wi… on as $p = P(x = 1) + 0.5\,P(x = 0)$

The probability of B winning under H_1

$$S_i = \log \frac{P(Data_i \mid H_1)}{P(Data_i \mid H_0)} = \log \frac{\widehat{p}_{H_1,i}^{wins_i^B + 0.5ties}(1 - \widehat{p}_{H_1,i})^{wins_i^A + 0.5ties}}{0.5^{wins_i^B + 0.5ties}(1 - 0.5)^{wins_i^A + 0.5ties}}$$

Data observed before i-th stop

The probability of B winning under H_0 : $p = 0.5$

MaxSPRT

But the probability of B winning a single comparison $\widehat{p}_{H_1,i}$ is not known!

- if we knew the real value, we wouldn't need to run an experiment in the first place
- we use the max-likelihood estimate

$$\widehat{p}_{H_1,i} = \frac{wins_i^B + 0.5ties_i}{T_i}$$

- in some sense, we compare H_0 with the most probable alternative

MaxSPRT

Learning the threshold *b*:

- Monte Carlo approach, by generating binomial random variables
  - not perfect, as ties are not emulated = variance higher than in practice
- From A/A logs

errors in statistical testing

Reality		Our decision: **H_0 is True**	Our decision: **H_0 is False**
	H_0 is True	Correct ☺	Type I error
	H_0 is False	Type II error	Correct ☺

Goal

Our goal is to find a sequential testing procedure that:

- has Type I error not higher than the significance level α
- has Type II error not very different from Type II error of the baseline single-step procedure
- has the smallest mean experiment deployment time = highest efficiency

dataset

- 2 A/A experiments deployed over 300 days:
    - to learn the thresholds
    - to estimate Type I errors
- 115 real-life experiments with known ($p < 0.001$) outcomes:
    - to estimate Type II errors
    - B > A in 56 experiments

results

Checking the test statistic every day

Duration of the experiment

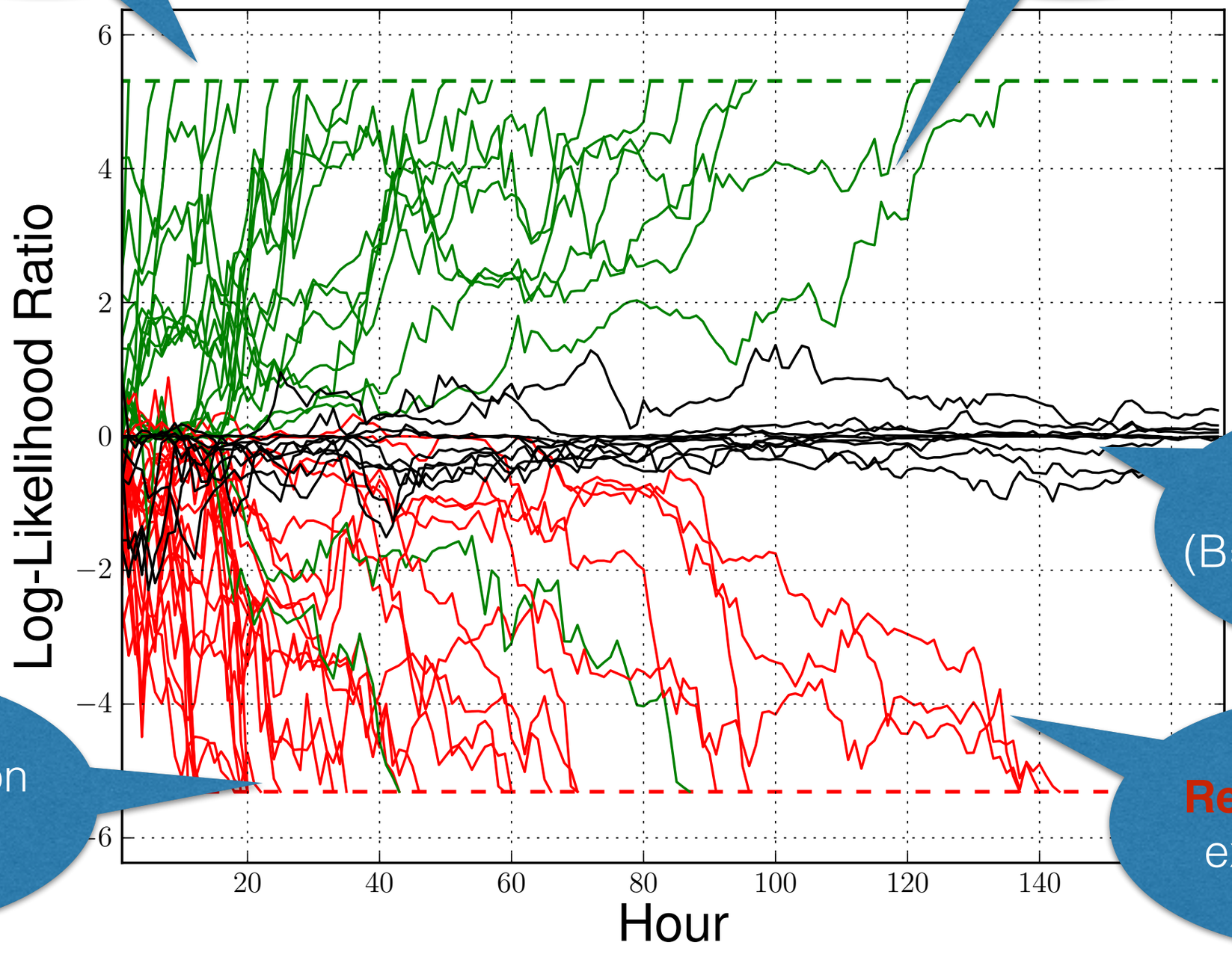
Test	# stops	Type I	Type II	$Acc_{B \succ A}$	$Acc_{A \succ B}$	$\mathbb{E}(T)$, days	$\mathbb{E}(T \mid B \succ A)$	$\mathbb{E}(T \mid A \succ B)$	$\mathbb{E}(\frac{N}{N_0})$
Binomial	1	**0.00**	0.10	**0.75**	0.90	7.00	7.00	7.00	1.00
OBF-I*	7	0.01	0.10	0.73	0.92	3.17	3.17	3.04	0.44
OBF-I	7	0.01	**0.09**$^{\triangle}$	0.73	**0.95**$^{\triangle}$	3.00	3.04	2.92	0.42
MaxSPRT-I-MC	7	**0.00**$^{\triangle}$	0.23	0.64	0.76	3.96	4.00	3.92	0.53
MaxSPRT-I-AA	7	**0.00**$^{\triangle}$	0.13	0.71	0.87	3.10	3.20	3.30	0.44
OBF-I*	$7 \cdot 24$	0.01	0.11	**0.75**$^{\triangle}$	0.88	3.58	3.54	3.67	0.45
OBF-I	$7 \cdot 24$	0.01	**0.09**$^{\triangle}$	**0.75**$^{\triangle}$	0.93	3.33	3.38	3.29	0.44
MaxSPRT-I-MC	$7 \cdot 24$	**0.00**$^{\triangle}$	0.19	0.71	0.76	3.38	3.38	3.42	0.43
MaxSPRT-I-AA	$7 \cdot 24$	**0.00**$^{\triangle}$	0.12	0.73	0.89	**2.61**$^{\triangle}$	**2.63**$^{\triangle}$	**2.58**$^{\triangle}$	**0.35**$^{\triangle}$

Checking the test statistic every hour

results

B > A decision boundary

Green are B > A experiments

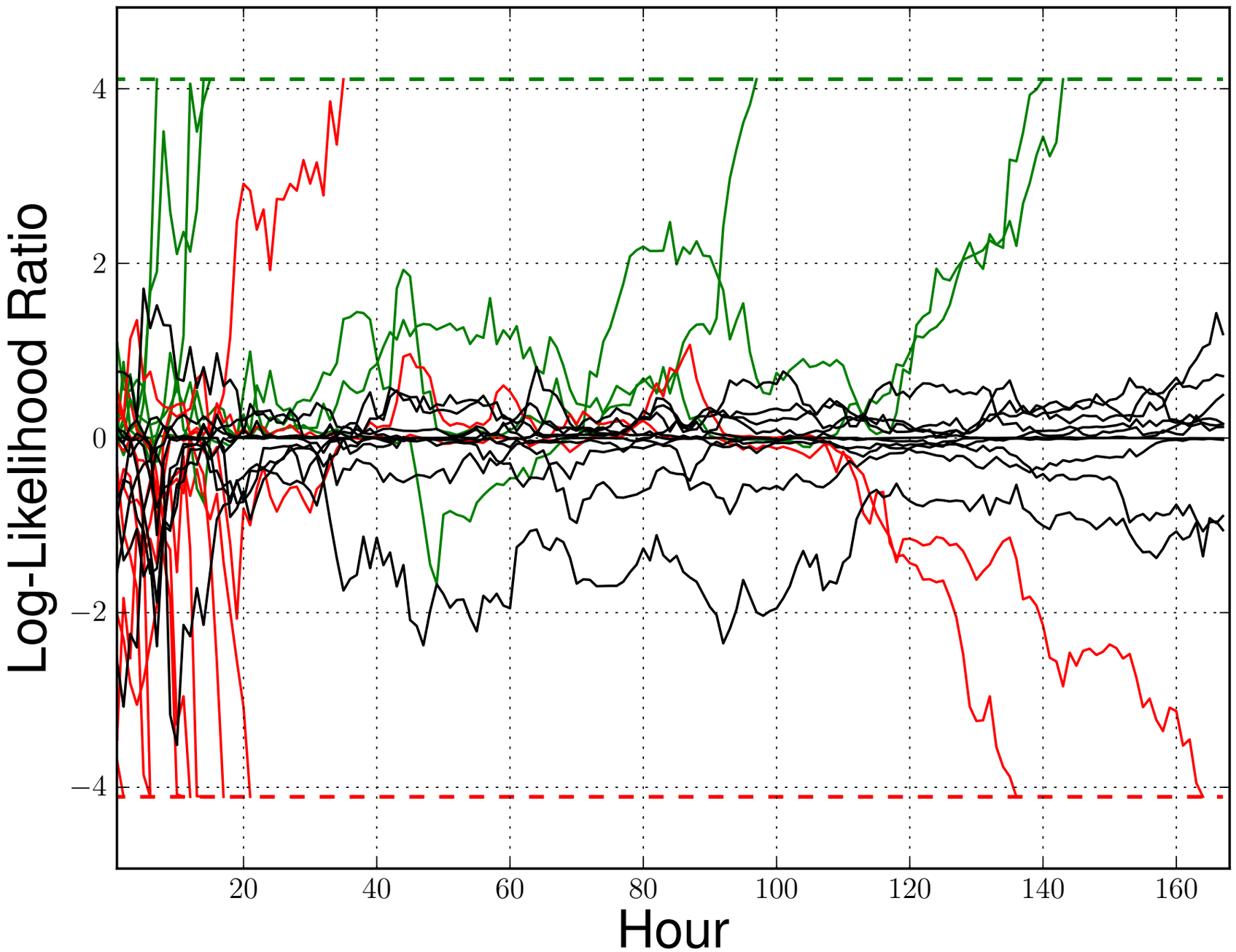


Black are A/A (B=A) experiments

A > B decision boundary

Red are A > B experiments

results for A/B tests



sequential testing

- By using sequential testing approaches we can markedly improve efficiency of the online evaluation

- Check our SIGIR 2015 paper:
  - A/B testing
  - Combing with a metric variance reduction method
  - More exciting technical details & tables & an additional figure

questions?

outline

Introduction

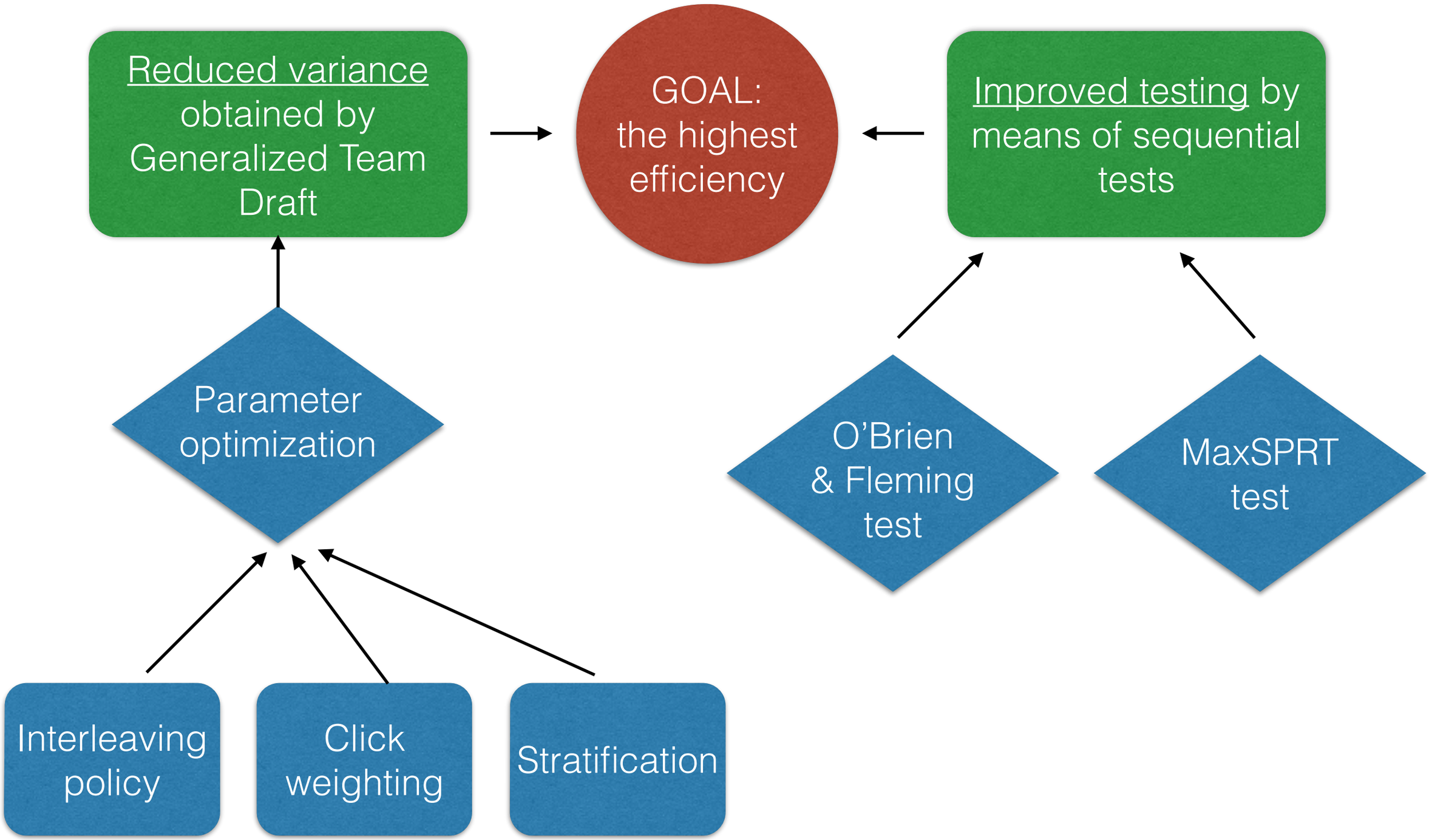
- online evaluation 101
- why efficiency is so Important?

Increasing the Online Evaluation Efficiency

- Generalized Team Draft
- sequential testing

Conclusions

today we discussed



today we discussed

- online evaluation
- an important challenge of increasing the efficiency of online evaluation
- how to address this problem from two perspectives
    - variance reduction
    - sequential testing
- some promising results

We’ve got more challenges than hands!

shameless advertisement

If you are a bit experienced in computer science & maths & programming, feel free to apply for a position:

- In our applied research group:

  https://yandex.ru/jobs/vacancies/dev/res_dmir/

- At Yandex:

  https://yandex.ru/jobs/vacancies/

Secret slides

future directions

- Our unbiasedness criterion is
    - necessary
    - sufficient (**formal proof?**)
- How can we learn non-linear scoring rules that remain unbiased?

future directions

- A MaxSPRT-like procedure for more sophisticated metrics, such as «sessions per users» and «user engagement»
- Optimization of the test statistic $\{S_i\}$ over i