UNIVERSITAT POLITÈCNICA DE CATALUNYA

RANDOMIZED ALGORITHMS

SECOND ASSIGNMENT

Balls and Bins
Balanced Allocation

Authors
Vidal Sulé GUILLERMO

November 17, 2025

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona

Contents

1 Introduction

The project aims to study a collection of allocation schemes for the balls and bins model. Although theoretical, this experiment is useful for the analysis of other well-known cases, such as a router processing incoming packages that arrive at different queues. It encompasses several strategies such as one, two, and D choice bin selection, batched arrivals, and partial information analysis. This is achieved through a thorough study of the gap and its variance/standard deviation.

1.1 The code

Part of the importance of this assignment lies within the code, so it is imperative to provide a succinct explanation about it. This is especially important considering the chosen programming language: Ada, since it is not commonly used. Accordingly, this section presents descriptions of selected Ada features to ensure that readers can fully comprehend the logic and purpose of the code.

The core procedure[1] of the program (`Bins_And_Balls`) simulates N balls being allocated to M bins a given number of times T. These program constants, and some others, are assigned a value of the sort `$X`, meaning that their value is assigned during preprocessing. It is also in charge of enforcing batching when it is enabled and registering the values of the gap for each simulation. A simplified version is shown in Algorithm 1.

The maximum gap is subsequently calculated with the given formula:

$$G_N = \underset{1 \leq i \leq n}{max} \{X_i(n) - \frac{N}{M}\}$$

Listing 1 shows how this is achieved in a single line of code. For each element of the array `Bins`, being the number of balls, it calculates de gap and then uses a maximum reduction to produce the final result.

```
Gaps (I) := [for E of Bins => E - N_Balls / M_Bins]'Reduce
   (Integer'Max, 0);
```

Listing 1: Maximum gap calculation

The calculations for the average gap, variance, and standard deviation are similarly programmed by making use of Ada array aggregates and reduce expressions. A simplified representation would be:

```
[array]'Reduce (operation, start_value)
```

[1] In Ada, a *procedure* does not return a value while the *function* does.

Algorithm 1 Bins and Balls Simulation (Simplified)

Input: N (balls), M (bins), D, β, K, T
Parameters: Strategy, Batched, Batch_Size
Output: $Gaps[1..T]$
for $i = 1$ to T **do**
    Initialize $Bins[1..M] \leftarrow 0$
    $Current\_Batch \leftarrow$ **if** Batched **then** Batch_Size **else** 1
    $Bin\_Allocation[1..M] \leftarrow 0$
    for $b = 1$ to N **do**
        $j \leftarrow$ Assign_Bin$(Strategy, M, D, \beta, K, Bins)$
        if Batched and Batch_Size > 1 **then**
            $Current\_Batch \leftarrow Current\_Batch - 1$
            $Bin\_Allocation[j] \leftarrow Bin\_Allocation[j] + 1$
            if $Current\_Batch = 0$ **then**
                $Bins \leftarrow Bins + Bin\_Allocation$
                Reset $Bin\_Allocation \leftarrow 0$, $Current\_Batch \leftarrow Batch\_Size$
            end if
        else
            $Bins[j] \leftarrow Bins[j] + 1$
        end if
    end for
    $Gaps[i] \leftarrow \max_{e \in Bins} |\, e - N/M \,|$
end for

```
Avg_Gap   : constant Float :=
  [for E of Gaps => Float (E)]'Reduce ("+", 0.0) / Float (T);
Variance  : constant Float :=
  [for E of Gaps => (Float (E) - Avg_Gap) ** 2]'Reduce ("+", 0.0) /
   Float (T);
Std_Dev   : constant Float := Sqrt (Variance);
```

Listing 2: Average gap, variance, and standard deviation calculation

Which roughly translates to: *for each element of the array, perform an operation with the starting value*. In the case of the variance also note that the `**` operator means *to the power of*.

As previously stated, the main procedure is also responsible for simulating batching. For each batch, every ball runs the allocation algorithm with the same initial view of the current distribution. The results are stored in a separate array and only updated at the end of the batch. This operation has been simplified by overloading the binary `+` operator of unbounded integer arrays. Hence, the line `Bins :=`

`@ + Bin_Allocation;`, which is a shorthand for `Bins := Bins + Bin_Allocation`, where both variables are arrays, equates to $Z_i = X_i + Y_i$.[2]

2 Simple choices

The first part of the analysis focuses on the most simple allocation schemes: one choice, two choice, d choice, and with probability β. For these strategies we will primarily examine the gap for which the formula is provided earlier in the document, but it will also be relevant to discuss the standard deviation for the d choice. Note that for all experiments values $T = 100$ and $M = 20$ have been used[3].

Figure 1 illustrates a steady increase in the average gap as the number of balls grows in the one-choice scheme. The recurrent peaks are due to the number of bins, which is twenty for all experiments. Since the slope shows no sign of plateauing, the gap appears to grow without bound. In contrast, the two-choice strategy exhibits a slope that stabilizes much more rapidly. In addition, $G(N = M) = G(20) = 1.1$, which is less than half of that observed under the one-choice scheme. To put it into perspective, the two-choice strategy never produces a gap that is remotely close to $G(20)$ of the one-choice scheme.

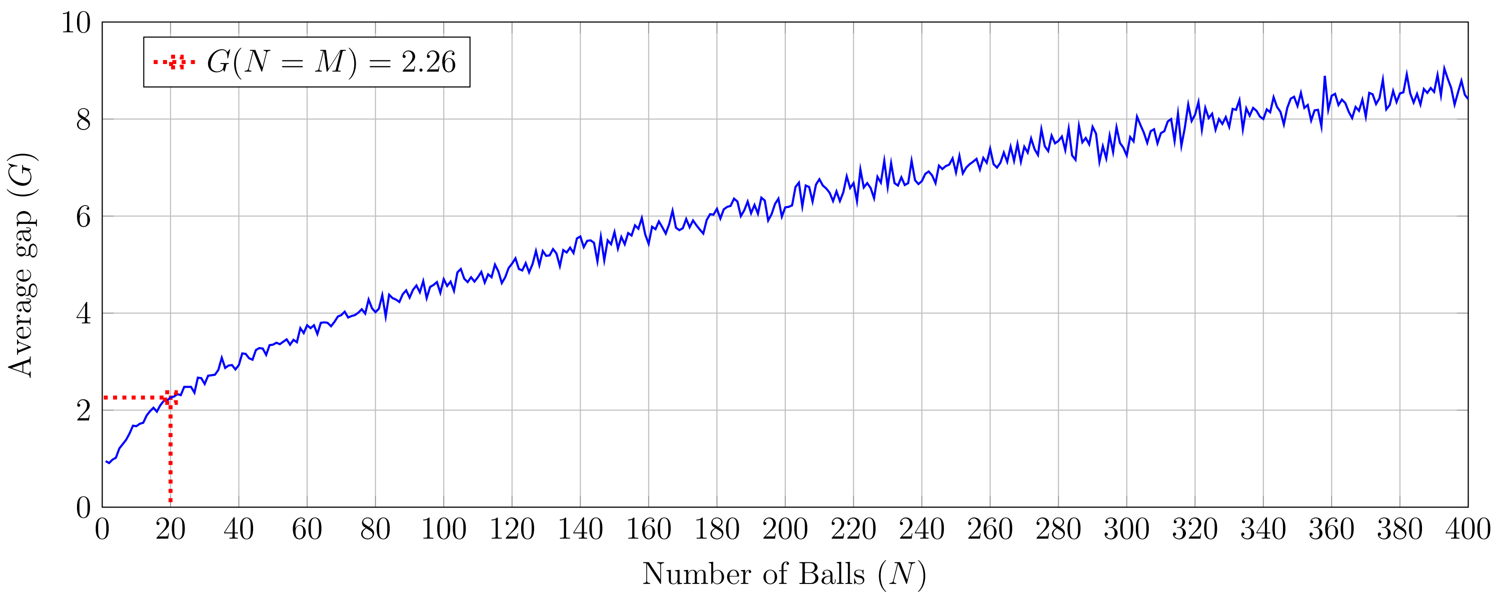

Figure 1: Gap progression with an increasing number of balls when using the **one choice** strategy.

[2]This has been programmed so that arrays of different sizes and bounds do not produce errors. The result is an array with minimum length with range `1 .. Min_Length`.

[3]This number was chosen arbitrarily.

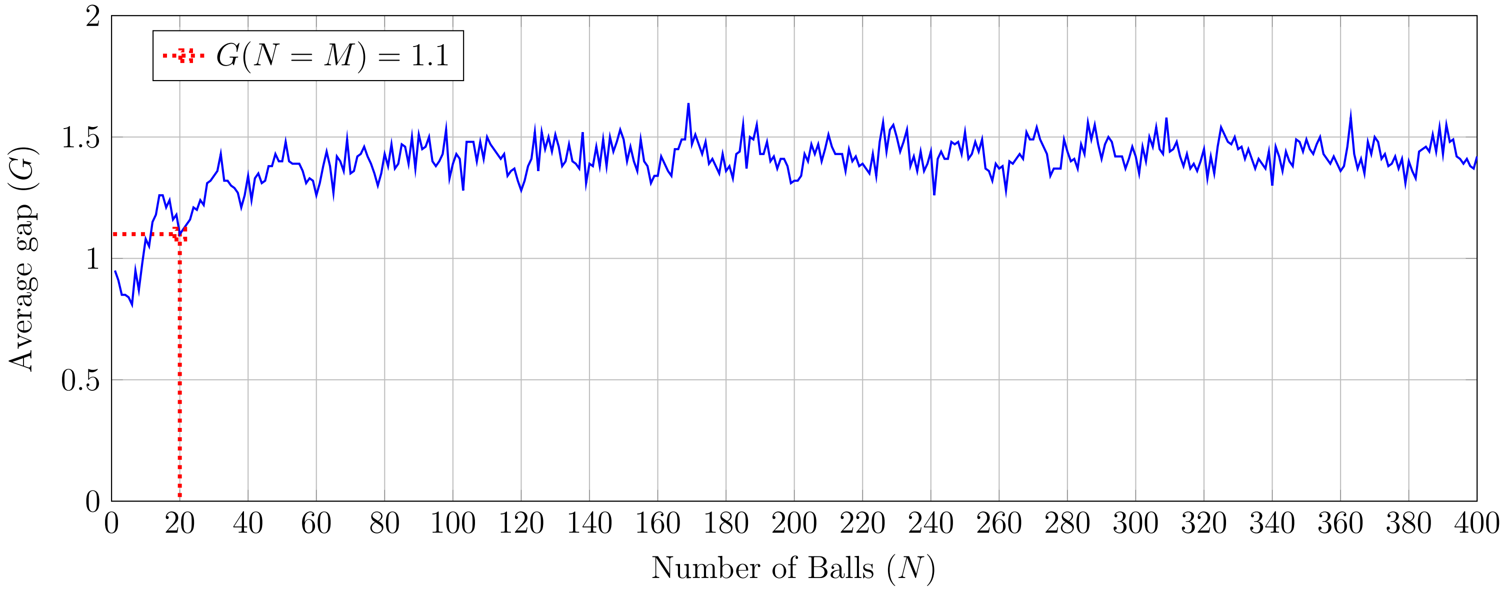

Figure 2: Gap progression with an increasing number of balls when using the **two choice** strategy.

The β-scheme, otherwise known as probabilistic strategy, selects either the two or the one choice strategy given a probability β and $\beta - 1$ respectively. Despite the fact that the statement formulates this differently, the probability of the one choice strategy and drawing one bin at random from the two bins also drawn at random is equal. The results in Figure 3 illustrate how the gap decreases as β grows, hence when the two-choice strategy is more likely. It is also worth noting how comparatively small the standard deviation is the closer beta gets to 1.

The last scheme left to analyse is the d-choice scheme. Figure 3 is a prelude to its results since it shows that the higher the β probability, meaning the model resembles more the two-choice scheme ($D = 2$), the less the average gap is. In addition, the standard deviation is certainly significantly smaller, implying that experiments with a higher D will produce more accurate results.

The experimental results included in Table 1 show a decreasing gap the progressively leans toward zero. Some elements with large D values show an increase from 0 to an average gap of 0.1 or 0.2 because of how small the probability of having one of the repetitions produce $G > 0$. Nonetheless, the probability is not zero, albeit it is very low, so in in certain cases one or two elements vary from the mean and therefore producing these surprising results.

Returning to the analysis of the standard deviation, we can see a distinct pattern from the average gap that was just discussed. Figure 4 illustrate the similar trend toward zero, but it also shows as small increase in σ just after the two choice strategy. It is also worth mentioning how the standard deviation is very similar for both the

$N = M$ and $N = M^2$ cases. This contradicts our previous hypothesis that the larger the D the lower the variance, even if this is true for most values.

As a final note, the deviation starts decreasing around $D = M = 20$, since at that point the odds of having the full picture (all balls for all bins) becomes quite large, hence choosing with high probability the absolute minimum.

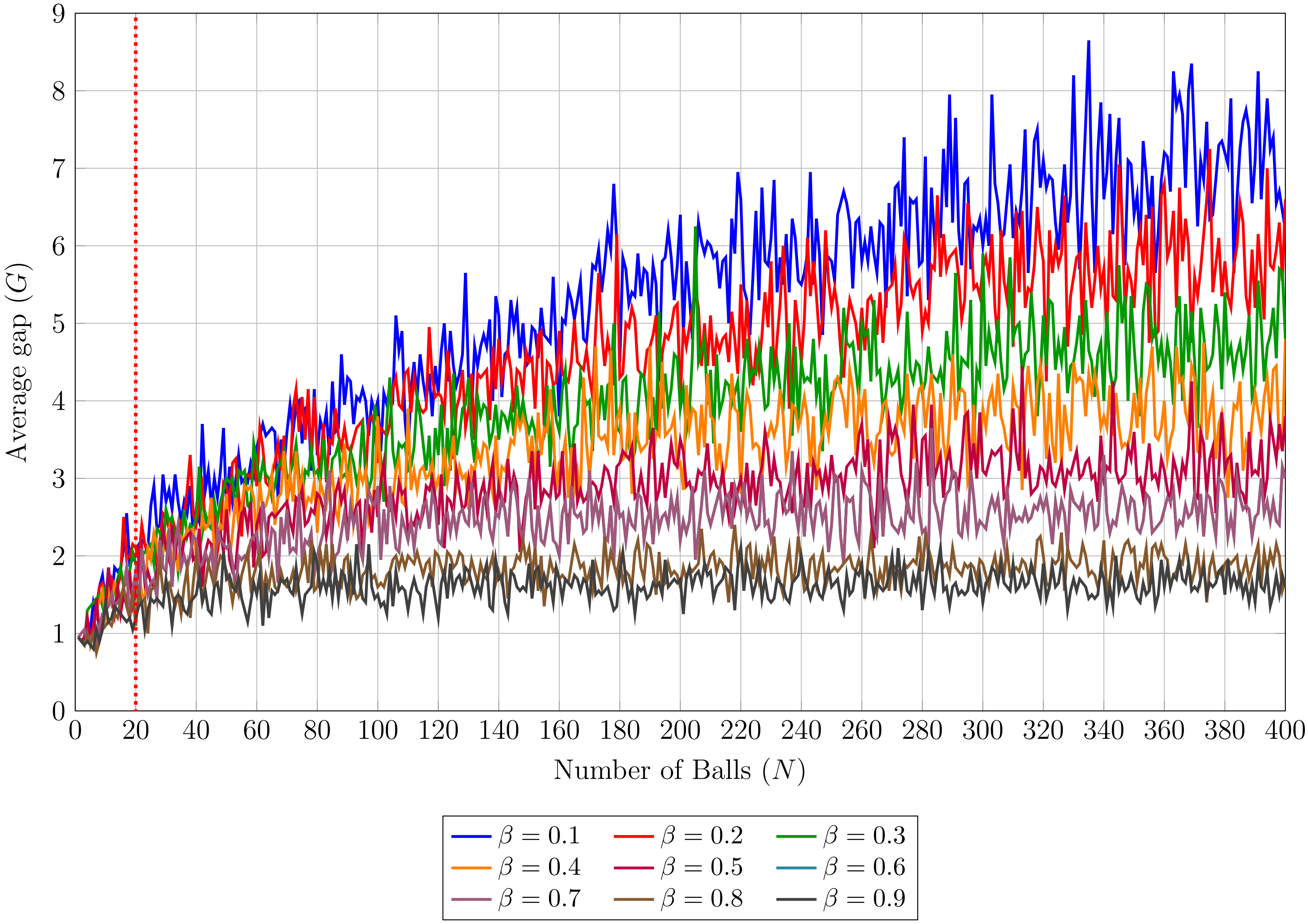

Figure 3: Gap progression with an increasing number of balls when using the β-**choice** strategy.

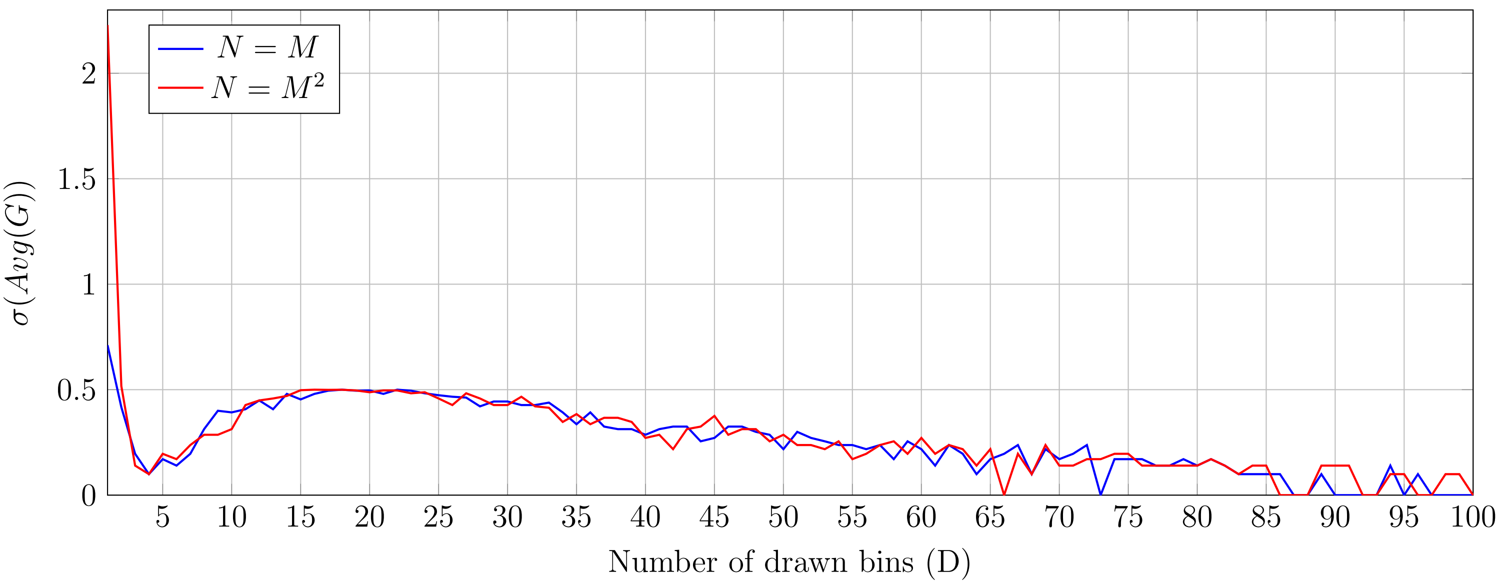

Figure 4: Standard deviation when increasing the number of drawn bins.

D	Avg. Gap	D	Avg. Gap	D	Avg. Gap	D	Avg. Gap
1.0	2.12	26.0	0.32	51.0	0.1	76.0	0.03
2.0	1.19	27.0	0.31	52.0	0.08	77.0	0.02
3.0	1.04	28.0	0.23	53.0	0.07	78.0	0.02
4.0	0.99	29.0	0.27	54.0	0.06	79.0	0.03
5.0	0.97	30.0	0.27	55.0	0.06	80.0	0.02
6.0	0.98	31.0	0.24	56.0	0.05	81.0	0.03
7.0	0.96	32.0	0.24	57.0	0.06	82.0	0.02
8.0	0.89	33.0	0.26	58.0	0.03	83.0	0.01
9.0	0.8	34.0	0.19	59.0	0.07	84.0	0.01
10.0	0.81	35.0	0.13	60.0	0.05	85.0	0.01
11.0	0.79	36.0	0.19	61.0	0.02	86.0	0.01
12.0	0.72	37.0	0.12	62.0	0.06	87.0	0.0
13.0	0.79	38.0	0.11	63.0	0.04	88.0	0.0
14.0	0.64	39.0	0.11	64.0	0.01	89.0	0.01
15.0	0.71	40.0	0.09	65.0	0.03	90.0	0.0
16.0	0.64	41.0	0.11	66.0	0.04	91.0	0.0
17.0	0.57	42.0	0.12	67.0	0.06	92.0	0.0
18.0	0.51	43.0	0.12	68.0	0.01	93.0	0.0
19.0	0.43	44.0	0.07	69.0	0.05	94.0	0.02
20.0	0.44	45.0	0.08	70.0	0.03	95.0	0.0
21.0	0.36	46.0	0.12	71.0	0.04	96.0	0.01
22.0	0.5	47.0	0.12	72.0	0.06	97.0	0.0
23.0	0.43	48.0	0.1	73.0	0.0	98.0	0.0
24.0	0.37	49.0	0.09	74.0	0.03	99.0	0.0
25.0	0.34	50.0	0.05	75.0	0.03	100.0	0.0

Table 1: Average gap for different D values.

3 Batching

An accurate way of modelling uncertainty is by allocating balls in batches so that when a given batch of balls is to be placed, all balls only see the initial state of the bins and therefore take decisions based on old information. Let us first take a look at how this influences the one, two and probabilistic choices schemes that were discussed in the previous section.

When $N = M$ and the batch size is bigger or equal to M no ball will have at any point information regarding previous balls. Consequently, each ball allocation is always a one-choice allocation, since the only record the ball has of the bins is when they were all empty, so it always ends up picking one at random. This is why the results are varied but at the same time show no progression, which is very similar to the results obtained from the one-choice scheme without batching.

On the other hand, when $N = M^2$, the increase in batch size progressively hides the ball distribution to more and more balls thereby increasing randomness and the average gap. Table 2 shows precisely that the higher the size of the batch the larger the average gap is. Notably, the two-choice scheme shows a sudden reduction around $b = 220$.

Finally, the average gap firstly increases and subsequently decreases, being the max usually concentrated around $b = \frac{m^2}{2}$. This phenomenon occurs due to the limited visibility when a large number of bins is drawn and all balls from a batch are placed in the same bin, being the one with lowest amount of balls. Later, once the batch size increases even further (toward m^2), the average gap decreases again because most balls only see the initial state (all bins are empty), so they all effectively choose a bin at random.

4 Partial information

Alternatively, another way of simulating uncertainty is by only providing balls with partial information of the ball distribution. As Figures 5 and 6 illustrate, the average gap constantly increases albeit very slowly. For $K = 2$ we obviously get a more precise distribution, since by asking two questions the algorithm gets a more accurate view of the data. Notably, in order to find the mean and certain elements the program uses random **Quickselect** since it is one of the most efficient ways.

Final note: everything is hosted in a public repository which includes under releaseas a tarball with all the data used in this report.

`https://github.com/guillermovidalsule/ra_assignment_2#`

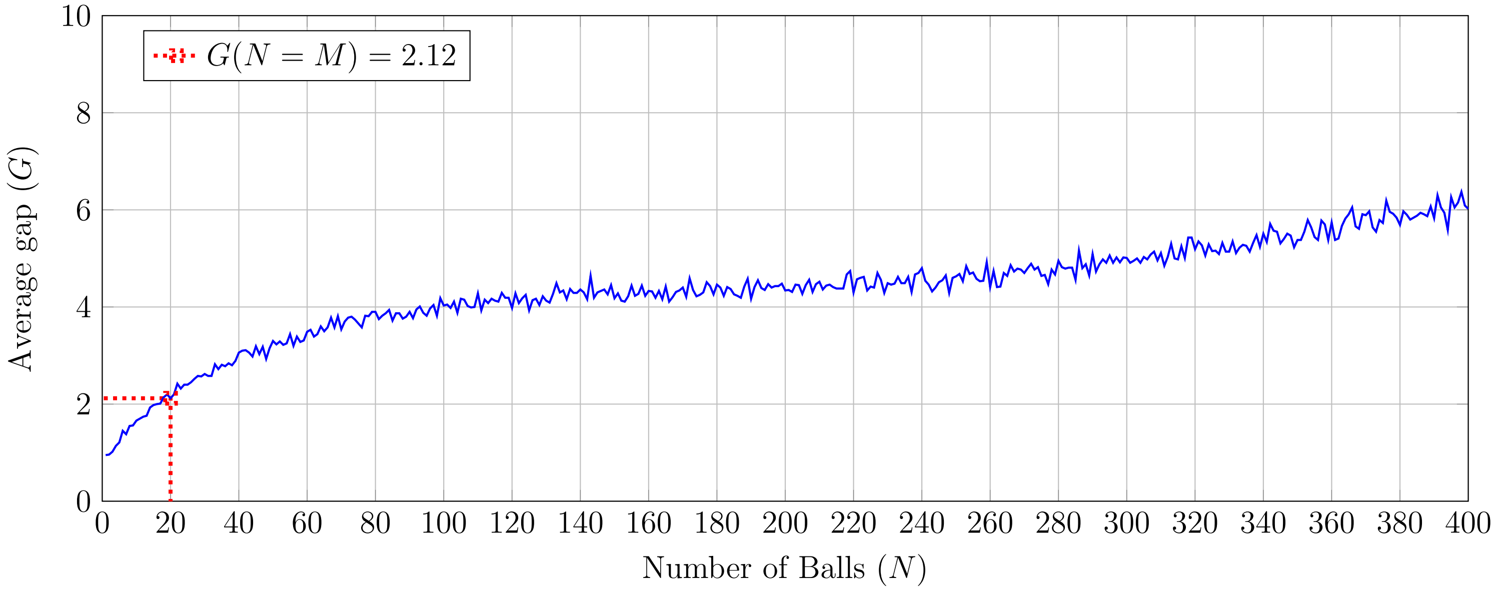

Figure 5: Gap progression with an increasing number of balls when $K = 1$.

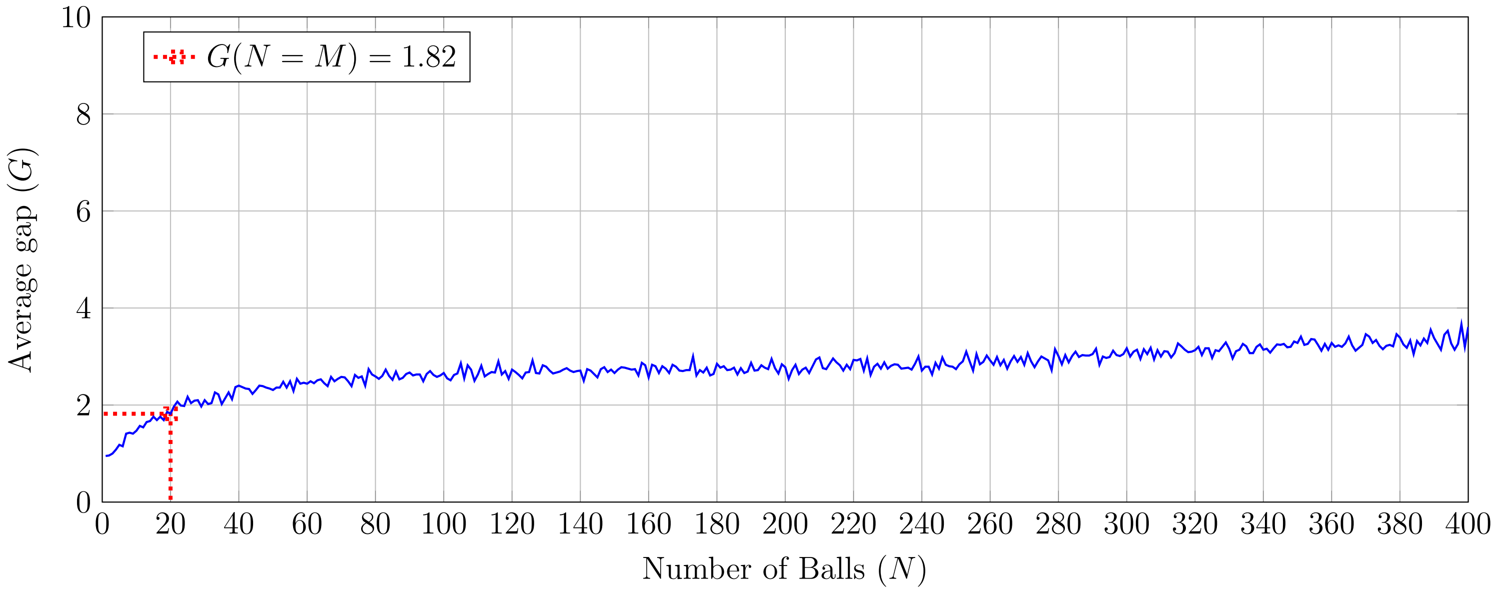

Figure 6: Gap progression with an increasing number of balls when $K = 2$.

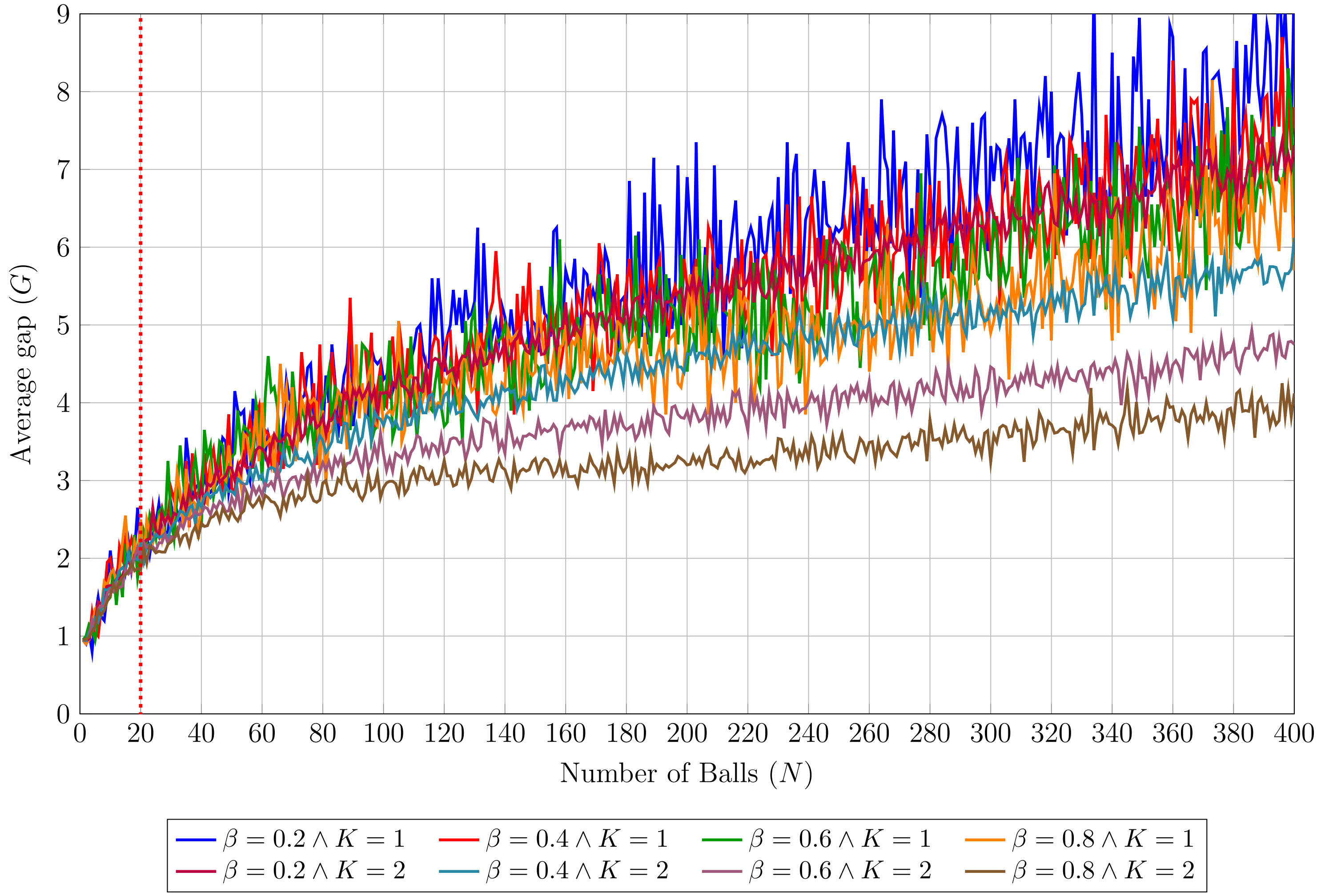

Figure 7: Gap progression with an increasing number of balls when using the β-**choice** strategy with $K = 1$ and $K = 2$.

A Batching tables

Batch Size	$\beta = \mathbf{0.00}$		$\beta = \mathbf{0.25}$		$\beta = \mathbf{0.5}$		$\beta = \mathbf{0.75}$		$\beta = \mathbf{1.00}$	
	$n = m$	$n = m^2$	$n = m$	$n = m^2$	$n = m$	$n = m^2$	$n = m$	$n = m^2$	$n = m$	$n = m^2$
20	2.2	8.9	2.2	4.6	2.3	3.0	2.2	2.3	2.6	2.4
40	1.8	8.8	2.3	5.1	2.2	4.4	2.1	2.7	2.2	3.2
60	2.2	9.1	2.3	6.0	2.1	5.0	2.5	3.6	2.7	3.3
80	2.3	9.7	2.0	5.6	2.3	4.4	2.2	4.3	1.9	4.0
100	2.5	7.4	2.2	6.2	1.9	3.9	2.4	4.6	2.0	4.6
120	2.4	8.7	2.5	5.9	2.4	4.2	2.5	4.2	2.5	3.9
140	2.5	8.8	2.1	7.1	2.4	5.6	2.5	6.7	2.0	5.2
160	2.4	9.6	2.1	6.7	2.0	5.7	2.3	5.5	2.3	5.6
180	2.3	9.0	2.2	5.9	2.2	6.2	2.0	5.7	2.2	7.4
200	2.1	9.5	2.3	6.9	1.9	4.9	1.6	6.7	2.2	10.0
220	2.3	8.5	2.2	7.4	2.0	5.8	2.4	7.6	2.0	8.8
240	2.0	8.9	2.3	6.3	2.0	6.3	2.4	6.5	2.5	6.4
260	1.8	8.0	2.4	6.5	2.8	6.5	2.5	5.2	2.1	6.2
280	2.4	7.6	1.8	7.3	1.9	6.3	2.0	4.7	2.0	6.1
300	2.2	10.2	2.1	8.0	2.1	5.6	2.4	4.3	2.4	4.6
320	2.0	8.9	1.9	7.7	2.4	6.4	2.3	6.6	2.2	4.6
340	1.7	8.9	2.2	9.3	2.1	6.9	2.6	6.3	1.8	6.3
360	2.2	8.7	2.3	8.0	1.9	8.4	2.1	7.5	2.4	6.3
380	2.4	11.0	2.3	9.0	2.5	8.4	1.9	6.4	2.7	6.6
400	2.2	9.3	2.0	10.0	2.5	8.0	2.0	7.8	2.4	9.0

Table 2: Results when batching for one, two, and β-choice; when $N = M$ and $N = M^2$.

Batch Size	**d = 3**		**d = 7**		**d = 11**		**d = 15**	
	$n = m$	$n = m^2$	$n = m$	$n = m^2$	$n = m$	$n = m^2$	$n = m$	$n = m^2$
20	2.3	2.1	2.2	3.7	2.5	4.7	1.8	5.4
40	2.6	3.8	2.7	6.9	2.0	8.4	2.2	11.4
60	2.3	4.0	2.1	9.3	2.0	12.9	2.1	15.1
80	2.1	5.4	2.2	14.3	2.4	20.8	2.0	28.1
100	2.1	7.4	2.0	16.7	1.7	26.9	2.4	29.3
120	2.3	6.8	2.6	20.0	2.6	32.0	2.0	41.6
140	2.5	8.5	2.0	26.4	2.1	42.0	2.5	50.0
160	2.0	8.7	2.2	31.0	2.6	49.6	2.5	61.6
180	2.5	11.6	2.4	37.5	2.4	55.8	2.3	71.4
200	2.5	15.9	2.1	41.0	2.1	64.2	2.4	85.7
220	1.9	12.4	2.0	41.9	2.3	62.8	2.2	81.6
240	1.9	12.4	1.9	31.2	2.4	48.4	2.3	65.1
260	2.0	9.4	2.3	32.5	2.1	45.8	2.2	61.0
280	2.3	8.5	2.4	26.0	2.5	37.6	2.1	49.4
300	2.5	6.1	2.2	17.6	2.0	31.1	1.9	39.0
320	2.3	4.7	2.2	12.7	2.7	22.0	2.4	33.5
340	2.5	5.2	2.2	9.9	1.8	13.9	2.0	20.3
360	2.0	5.4	2.1	5.4	2.1	9.3	2.3	9.4
380	2.1	5.6	2.2	7.0	1.7	7.4	2.3	8.4
400	1.9	8.5	2.3	8.9	2.0	8.8	2.1	8.2

Table 3: Results when batching for $d-choice$ when $N = M$ and $N = M^2$.