**Balancer Simulations Math Challenge - Basic Exercises**

This is a series of exercises to gain intuition for the core algorithm in Balancer Pools: the Value Function, and invariant V.

**Exercise:** Please work on the questions in this notebook individually or in teams, we'll discuss solutions in our working session on June 16th, facilitated by Octopus and Angela.  
**Challenges:** Step 2 will be more advanced challenges to be solved until July 28th.

**Exercise:**

Let's set up a pool.

We have 100 Token A and 100 Token B, with equal weights.  
The price definition in our pool is constraint by the Invariant V in Balancer Pools.

a\_bal = balance of token A  
b\_bal = balance of token B  
a\_weight = weight of token A  
b\_weight = weight of token B

The weights in this pool are equal for both tokens. For now, we don't have a swap fee.

**Q1.1:**

What's the initial spot price of token A in token B?

Initial spot price of one Token A is 1 Token B since nothing yet has been swapped in a perfectly balanced pool. This is not the Effective Price however.

**Q1.2:**

Now let's assume a series of 99 swaps. With every swap, **1.0 token A is bought from the pool, against token B**.

**a) Create a table "buy\_A"** with

the token A balances (swap by swap)

the token B balances (swap by swap) - that are constraint by the value function.

**b) What do you notice in general?** Write down your findings (in words).

In the AMM model established with 100 of each tokens and equal weights the curve that I was able to find using the change in the spot price in each swap was quite gradual and then dramatic as it moved up the curve. This is inherently the mechanism upon which the value is maintained as the price should not move much from one swap but as more and more Token A are swapped for Token B the compensating effect in price of Token B must overcome the diminishing amount of Token A or seen another way the price must compensate for the increase in Token B over the initial pool creation.

**c) How much would Alice have to pay in token B when buying the first 1.0 token A?** Write down your findings (in words). Compare with the initial Spotprice

The spot price is the perfect price on the curve that has not yet taken into account any changes to the balances. Thus the spot price was lower, and increasingly so from the effective price. In the first swap Alice would have to effectively pay 1.0101 of Token B for one Token A whereas the Spot Price was .9801. This is a difference of 0.03, which is the automated spread (Effective – Spot Price) on the first swap.

**Q1.3:**

Now let's assume a series of 99 swaps in the opposite direction. We start again with the original state: We have 100 Token A and 100 Token B.  
With every swap, **1.0 token B is bought from the pool, against token A**.

Create a table **'buy\_B'** with

the token A balances (swap by swap)

the token B balances (swap by swap) - that are constraint by the value function.

**Q1.4:**

What is the new price of token A in token B after 90 swaps token A for B?

The effective price for Token A after 90 swaps (meaning the 91st) is 90.9091

**Q1.5:**

Now create a graph (use plotly or similar), and draw the full curve for this series of both kinds of swaps - the AMM curve.

**Q1.6:**

Take this plot, and mark

* the initial price in Q1.1 (starting price)
* the new price in Q1.4 (after 90 swaps)

**Q1.7:**

Formulate a "rule of a thumb", how do swaps effect the price?

Swaps impact the effective price commensurate to the number of tokens out of the pool, adjusted to the weights, are being swapped.

**Now, let's consider weights!**

We continue with the value function V = a^w\_a\*b^w\_b  
where  
a = balance of token asset A  
b = balancer of token asset B  
w\_a = weight of token asset A  
w\_b = weight of token asset B

**Q2.1:**

Write down the value function for the pool in Q1.1!

V = a^w\_a\*b^w\_b for Q1.1 is 100 = (100^.5)\*(100^.5)

**Q2.2:**

Let's got back to your initial balances in Step 1 in the pool:  
100 tokens A  
100 tokens B

How do you need to change the weights in order to land at a **price of  
4 tokens A : 1 token B**

Provide the new value function!

To arrive at a ratio of 4 to 1 the Weights would be Token A 0.8 and Token B 0.2

The Value Function would be 100=(100^.8)\*(100^.2)

**Q2.3:**

Create a graph showing the new AMM Curve in Q2.2  
Compare to the graph in Q1.4 - how does a change in weights change the graph?

The 80/20 pool is exponentially steeper in the change in the effective price.

**Q2.4:**

Compare token prices in this pool.  
How much would Alice have to pay in case there are only 2 tokens left in the pool  
**a) buy 1.0 token A for token B** 501,543,209  
**b) buy 1.0 token B for token A** 84,394,290