Let's suppose we have the following data for player 2:

|  |  |
| --- | --- |
| **round\_number** | **strategy\_selected** |
| 1 | 2 |
| 2 | 2 |
| 3 | 1 |
| 4 | 1 |
| 5 | 2 |
| 6 | 2 |
| 7 | 2 |
| 8 | 2 |
| 9 | 2 |
| 10 | 1 |

I want now to generate the arrays and in

So, I generate the first arrays based in the cumulative frequencies of selected strategies by , and get the following result:

|  |  |  |
| --- | --- | --- |
| **t** |  |  |
| 1 | 1 | 0 |
| 2 | 1 | 0 |
| 3 | 0.66666667 | 0.33333333 |
| 4 | 0.5 | 0.5 |
| 5 | 0.6 | 0.4 |
| 6 | 0.66666667 | 0.33333333 |
| 7 | 0.71428571 | 0.28571429 |
| 8 | 0.75 | 0.25 |
| 9 | 0.77777778 | 0.22222222 |
| 10 | 0.7 | 0.3 |

Now for , the formula is

**QUESTION 1:** if and it is a constant, so it means that , that is the sum takes place from previous to current round, correct?

Which makes sense if only the latest strategy is selected. And then second question, If this indicator function always return 1, then the result of this sum will always be 0.5, since for the selected strategy would be 1, and for the other it would be zero.

**QUESTION 2:** Why we divide 1/W in this case?

If I put these second arrays together how I think this works, I would have

|  |  |  |
| --- | --- | --- |
| **t** |  |  |
| 1 | 0 | 0 |
| 2 | 0 | 0 |
| 3 | 0 | 0 |
| 4 | 0 | 0 |
| 5 | 0 | 0 |
| 6 | 0 | 0 |
| 7 | 0 | 0 |
| 8 | 0 | 0 |
| 9 | 0 | 0 |
| 10 | 0.5 | 0 |

So I would calculate as:

Then I have

**QUESTION 3:** is it the same reasoning you applied?