The Big O of the Intelligence

Paul Rumkin

January 2020

1 Introduction

There are various options to specify an algorithm complexity or effectiveness behind it. It's very useful for representation of how algorithm scales or to compare two algorithms. There are several classic options to describe the scalability like n, $log(n)$, etc. But we have no value or constant to specify tasks which require intelligence and thus couldn't be automated now and couldn't be described or compared with another algorithm. What makes it harder to analyze or describe weakness or strength of technology. This paper proposes to introduce new constant to describe algorithms which couldn't be automated without having artificial intelligence or without decomposition into independent tasks and replacing them with non-intelligent analogs.

2 The I

Usage of a constant I to represent such complexity $O(I)$ could help to describe, understand and manipulate algorithms to simplify technological solutions, reduces the costs on development and ownership, and speed up the progress. The closer analogous for this is a large number notations (Stainhaus–Moser notation[2] or Knuth's up-arrow notation[1]). They are using to represent and manipulate numbers which are too big to write them out. It simplifies understanding and make possible to operate such numbers in mind.

2.1 Properties

The I is a probabilistic range of numbers which are very high, but we don't know now what's the boundaries are. We know that people couldn't multiply very large numbers in mind, but we know that growing person can speak and calculate. So we can apply fuzzy logic to it, for example the intelligence of an average child is lower than intelligence of an average graduated adult:

$$I_{Infant} < I_{Adult}$$

What means we can sort tasks according with required intelligence even without reveling the exact values. And make it possible to use fuzzy computations.

The more people involved in the task execution, the faster the complexity raises, due to communication and management costs. So it's not cumulative. Thus we can say that:

$$n * I < I(n)$$

.

And, what's important, small organized groups can solve more complex tasks than disorganized crowd, but less than big organized group. So it's correct to say that:

$$n * I < \sqrt{n} * I(\sqrt{n}) < I(n)$$

.

But after some value the effectiveness of the big group starts to fall. What means there is some theoretical limit for creation a network of independent artificial intelligence units.

Any decision which should be made increases complexity of the task, the more decisions we should make, the higher probability of mistake, so independent decisions has lower complexity, than a sequence of decisions. Thus it behaves like probability and independent decisions should be presented as a sum $I_1 + I_2$ and sequence as a multiplication $I_1 * I_2$.

2.2 Application Example

For example let's take a task to find some API documentation and get URL which serves for authorization. Documentation could be in vary formats HTML, PDF, or Markdown, has the latest and outdated versions, contains wrong and correct examples, etc. This task requires understanding of the context, what, therefore, requires an intelligence. So it's complexity now is $O(I^x)$, where x is the number of steps. To parallelize this task we need one more human ($O(n * I^x)$). It couldn't be automated in current view. To automate it we need to reduce the complexity converting documentation into machine readable format and to put it on a standard route. It can give us $O(1)$ complexity by hitting a single URL and $O(I(y))$ where y is the number of people involved into the standard creation.

3 Conclusion

$O(I)$ is useful to describe non-automated tasks and for understanding of how the intelligence can increase or affect the effectiveness. And that a huge number of very good but intelligence-capable units can make the whole system inefficient. Analysis based on intelligence capacity can help to build more simple and effective software and enhance methods of software and information systems architecture creation. What's important for building more effective social structures and societies.

References

[1] Wikipedia. *Knuth's Up Arrow Notation*. URL: http://web.archive.org/web/20191208012849/https://en.wikipedia.org/wiki/Knuth's_up-arrow_notation. (accessed: 2020-01-01).

[2] Wikipedia. *Steinhaus–Moser notation*. URL: http://web.archive.org/web/20190619205505/https://en.wikipedia.org/wiki/Steinhaus%E2%80%93Moser_notation. (accessed: 2020-01-01).