Continuous Local Search - Corrigendum

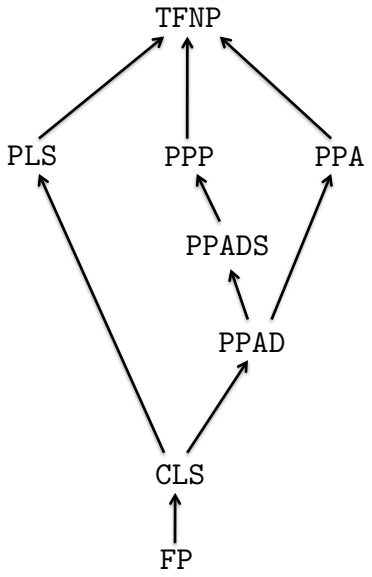
Constantinos Daskalakis
EECS and CSAIL, MIT

Christos H. Papadimitriou
CS, Columbia University

October 2020

This is a corrigendum to our paper "Continuous Local Search," published in the Proceedings of the 2011 ACM-SIAM Symposium on Discrete Algorithms (SODA) with ISBN 978-0-89871-993-2. We would like to fix a couple of issues that have come to our attention:

- First, Figures 1 and 2 in the proceedings version are incorrect in that the complexity class `PPADS` is shown to be a subset of the complexity class `PPA`, when no such containment is known. Thus Figure 2 should be updated to the following (and Figure 1 should look similar to the figure shown below, albeit with `CLS` removed from the figure).



- A second, more subtle, issue arises from our definition of circuits of real-valued operations. Various problems defined in our paper take as input an arithmetic circuit with binary gates from the set $\{+, -, *, \max, \min, >\}$ plus 0-ary gates supplying rational constants in their output.

  A subtle issue arises from the fact that arithmetic circuits of this type can, in fact, compute numbers at their final output and/or intermediate gates, whose binary description size scales exponentially in the size of the circuit. See, e.g., Figure 1 below, where n multiplication gates suffice to compute the 2^n-th power of some number. This property of arithmetic circuits can be exploited to turn our problems more powerful than intended, when our goal in this paper is to capture complexity classes that are subsets of `FNP`. Indeed, as defined in the proceedings, our problems are not known to even lie in `FNP`.

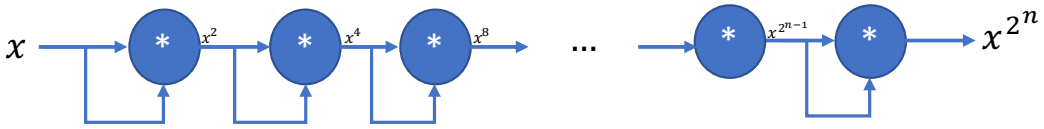


Figure 1: A sequence of n multiplication gates compute the 2^n-th power of their input.

To correct this issue we must restrict the arithmetic circuits provided as input to our computational problems so that the circuits may only compute numbers at their output and intermediate gates whose description complexity is polynomial in the size of their own input (the input to the circuit) as well as in the size of the overall input to our computational problems (i.e. the size of the description of the circuit itself and the size of any other input to our problems). There are a few natural ways to achieve this. Here is one:

– Besides all their other inputs, including an arithmetic circuit C_f as above, our computational problems take an additional input K, written in unary representation. The additional input K is interpreted as specifying a purported property satisfied by circuit C_f. The claim is that, for every input x to the circuit, each gate of the circuit satisfies that, if its output is written in the form a/b, where a and $b > 0$ are integers with $\gcd(|a|, b) = 1$, then both $|a|$ and b are $(\text{size}(x) \cdot K)$-bit natural numbers. Again, this is a property that is purportedly satisfied by C_f. Among all other acceptable outputs, our computational problems also accept the following type of output: providing some x for which the circuit violates this property at some gate.

Acknowledgements. We thank Kousha Etessami for useful discussions about the subtle issue above.