Towards a quantum algorithm for evaluating WCETs

Gabriella Bettonte[1], Stéphane Louise[1], Renaud Sirdey[1]

Université Paris-Saclay, CEA List, France

{gabriella.bettonte, stephane.louise, renaud.sirdey}@cea.fr

Abstract : *In this paper we propose a quantum-based solution to the problem of counting the cache hits, an important issue when analyzing real-time embedded applications. This field has already seen the development of "quantum-inspired" classical algorithms which are competitive with the state of the art. We designed a polynomial-time dynamic programming algorithm for computing the lowest number of cache hits realized by a deterministic sequence of memory accesses, in the presence of preemptions. Our contribution consists in porting that algorithm to the quantum framework, improving the complexity of the algorithm from $O(N^3)$ to $O(N^2+N)$.*

Keywords : *WCETs, preemption, quantum computing, dynamic programming.*

Introduction

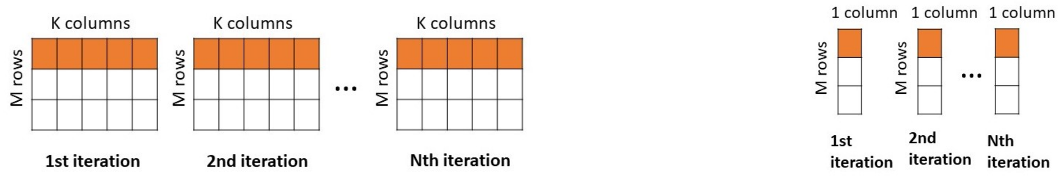
Despite a definitive advantage in terms of efficiency on a restricted set of problems, a lot of questions remain open about the real world applications and benefits of quantum computing. This is especially true for more arbitrary problems and fields without any a priori "quantum-friendlyness". In this paper, we consider the field of worst-case evaluation time evaluation by static analysis of programs, highly relevant to the design of safety-critical real-time systems. The field has recently seen the developments of "quantum-inspired" classical algorithms which are competitive, in terms of precision and efficiency, with the state-of-the-art [1]. Furthermore, the problems arising in that topic cover a wide-range of complexity classes, from undecidability in the general case down to NP-hardness and polynomial-time solvability in some restricted cases [2]. As such, it appears to provide a relevant playground to put the quantum computing promise to the test. As a first step, we tackle only a restricted setup with a simple program model : the evaluation of the worst-case number of cache misses of a programs performing a deterministic sequence of memory accesses, in case of arbitrary preemptions (the act of interrupting one task to allow the execution of another task on a machine). There is indeed a strong connection between the number of cache misses done by a program and its execution time, as uncached memory accesses are highly time-consuming on modern processors. To solve this problem, we designed a dynamic programming polynomial-time classical algorithm which computes the minimal number of cache hits induced by the sequence of memory accesses. Although this model would be considered straightforward by the WCET community [2], we use it, as well as the classical polynomial-time algorithm solving it, as the basis to derive a lower complexity hybrid quantum-classical algorithm. In doing so, we demonstrate a first benefit of explicitly using the quantum computing paradigm to the field of WCETs calculations.

Overview of main results

We consider programs producing deterministic sequences of memory accesses of length N. Let us suppose we know if each of these accesses is a cache miss or hit in an execution without preemptions (in this case this is done in linear time as a preprocessing step). We denote P the set of possible memory addresses. The number of preemptions interrupting the program is K. Preemptions can happen at any time and, when they occur, we consider that all the content of the cache is flushed.

The main structure of the classical dynamic programming algorithm we designed is made of three nested loops : (a) on the sequence of memory accesses (N iterations), (b) on all the possible elements in P ($|P| = M$ iterations) and (c) on all the possible number of preemptions (K iterations). In order to compute the final solution, at each iteration of the first loop, we create a table of dimensions $M \times K$, containing the number of the cache hits, for each item in P and each k', with $0 < k' \leq K$. Each iteration of the second loop is independent from the others, meaning that we can compute any row of the next table independently from the others rows. Then, the complexity of "updating" a row (most inner loop) is $O(K)$. Therefore the dynamic programming classical algorithm has a complexity of $O(M \times N \times K)$ (see Fig 1a). After performing this classical algorithm we obtain the smallest number of cache hits (our chosen measure for WCETs) for each object in P, when up to K preemptions occur.

In the context of porting classical applications to quantum computing, we designed a "hybrid quantum-classical version" of the previous algorithm, whose output is the number of cache hits corresponding to K preemption. We consider a row in one table as a superposition of states representing all the numbers of preemptions k' and all the possible values of the number of cache hits. In this way, the complexity of the act of "updating" a row becomes $O(1)$ (see Fig. 1b). Therefore the complexity of the resulting quantum algorithm becomes $O(M \times N)$. As a drawback, we need to post-process the result we obtained at the previous step : now we have to deal with superpositions of dimension $K \times U$, where $U < N$ is the maximum number of cache hits. However, we only need the value of the number of cache hits when K preemptions occur (i.e. only one value per row in the last set of rows). Therefore, for each row, we can use Grover algorithm to extract the value of interest in the associated superposition.This means that this post-processing has complexity $\sqrt{K \times U} < \sqrt{K \times N} < \sqrt{N^2} = N$. Hence, the complexity of the whole quantum algorithm is decreased to $O(N^2 + N)$ compared to $O(N^3)$ for the classical algorithm.



(a) Classic algorithm

(b) Quantum algorithm

FIG. 1 – In orange : the number of cache hits, for one object in P and all k'

Conclusions

In this paper, as a first step to deal with WCET-related problems by means of quantum-classic hybrid algorithms, we worked on a simplified program model since we just considered linear sequences of instructions, yet in the presence of arbitrary preemptions. A natural perspective would then be to generalize this approach to more complex program models, allowing for some non-determinism in the control-flow. With this work, we have ported a polynomial-time dynamic programming algorithm to the quantum framework. In essence, most such algorithms follow the same regular pattern of several nested loops updating an array data structure with the final result obtained in only one of the entries of the last array. That being said, the approach in this paper, consists in turning the inner loop of our algorithm in a "quantum parallel for" associated to a Grover-style amplification on the *single* result of interest. As a consequence, the approach in this paper may be generalized to other dynamic programming algorithms in order to derive quantum speedups.

References

[1] S. Louise, "A First Step Toward Using Quantum Computing for Low-Level WCETs Estimations." 2019.

[2] R. Wilhelm, et al. " The worst-case execution-time problem : overview of methods and survey of tools." 2008.