# A Reformulation of the Quantum Query Model with a Linear Condition for Exact Algorithms


Sebastian A. Grillo[1] and Franklin L. Marquezino[1,2]

[1]COPPE/PESC *and* [2]Xerem Campus, Universidade Federal do Rio de Janeiro, Brazil

*{sgrillo, franklin}@cos.ufrj.br*



19th Conference on Quantum Information Processing – Banff, Canada, 2016



**Abstract** We present a reformulation of the Quantum Query Model (QQM) without considering the measurement step. Our new formulation is described by a set of vectors satisfying some properties, which we call a *Block Set*. Given an algorithm in QQM we prove that there is an equivalent algorithm in the Block Set Formulation (BSF). If we keep the same measurement step, then we prove that both formulations have the same Gram Matrix of the output states, and the BSF allows us to describe it explicitly. Finally, we test the advantages of BSF by applying this approach to the construction of quantum exact algorithms.


## Motivation

The QQM is a framework that allows us to express most known quantum algorithms. The algorithms represented by this model consist on a set of unitary operators acting over a finite Hilbert space, and a final measurement step consisting on a set of projectors. A lack of frameworks for constructing quantum algorithms is specially noticeable for the exact case, which can be partially explained by the following reasons:

- Currently, some few exact quantum algorithms are known that could be combined for calculating more complex functions. Years ago, the only exact quantum algorithms known to produce a speed-up over classical algorithms for total functions were those that used Deutsch's algorithm as a subroutine.
- Numerical methods like Barnum *et al.* (2003) just give us approximate solutions, whose results can be very hard to translate into analytically defined algorithms for the exact case.

By using a formulation based on span programs several important results have been obtained for bounded-error algorithms. Ambainis (2013) proposes as a research problem the development of tools for designing exact quantum algorithms. **This work proposes another reformulation of the quantum query with the goal of making it easier to build exact quantum algorithms.**

## Block Set Formulation

In the quantum query model, the execution of an algorithm for input $x$ is given by a final state $\left|\Psi_x^f\right\rangle = U_t O_x U_{t-1} ... U_1 O_x U_0 \left|0,0\right\rangle$, where $U_i$ is a unitary operator and $O_x$ is the oracle operator. We define the oracle as $O_x \left|i\right\rangle \left|\Psi_i\right\rangle = (-1)^{x_i} \left|i\right\rangle \left|w\right\rangle$, such that $x_0 = 0$, $i \in \{0, .., n\}$ and $w$ are ancilla bits (or, working register).

We prove that **the application of unitary operators before the measurement step in the QQM is equivalent to decomposing an unit vector into a sum of vectors and then inverting some of their relative phases.** We call those vectors a *Block Set*. They must fulfill a list of properties.

**Definition 1.** *Let $n, t \geq 0$. We say that an indexed set $\left\{\left|\Psi(k)\right\rangle \in H_Q \otimes H_W : k \in \mathbb{Z}_{n+1}^{t+1}\right\}$ is a* Block Set *for the ordered pair of Hilbert spaces $(H_Q, H_W)$, if:*

*(i)* $\sum_{k_0=0}^{n} ... \sum_{k_t=0}^{n} \left\| \left|\Psi(k_0, .., k_t)\right\rangle \right\|^2 = 1.$

*(ii)* $\left\langle \Psi^i(b_0, .., b_{t-i}) \mid \Psi^i(c_0, .., c_{t-i}) \right\rangle = 0$ *if* $b_{t-i} \neq c_{t-i}$ *for* $0 \leq i \leq t$,
*where* $\left|\Psi^i(a_0, .., a_{t-i})\right\rangle = \sum_{k_1=0}^{n} ... \sum_{k_i=0}^{n} \left|\Psi(a_0, .., a_{t-i}, k_1, .., k_i)\right\rangle.$

*(iii)* $n = \dim(H_Q) - 1.$

*(iv)* $\dim(H(i,j)) \leq \dim(H_W)$ $\forall i, j$,
*where* $H(i,j) = span\left\{\left|\Psi^{t-i}(a_0, .., a_{i-1}, j)\right\rangle : a_k \in \mathbb{Z}_{n+1}\right\}.$

Let $S_x := \{i : x_i = 1\}$, where $x_i$ is the $i$-th bit in $x$. The Dirac measure, denoted as $\delta_z(A)$, equals 1 if $z \in A$ and 0 otherwise. The output state of the input $x$ under a Block Set $\left\{\left|\Psi(k)\right\rangle \in H_Q \otimes H_W : k \in \mathbb{Z}_{n+1}^{t+1}\right\}$ is defined as

$$\left|\Psi_x^f\right\rangle = \sum_{k_t=0}^{n} ... \sum_{k_0=0}^{n} (-1)^{\sum_{i=0}^{t} \delta_{k_i}(S_x)} \left|\Psi(k_0, ..., k_t)\right\rangle. \tag{1}$$

**For any QQM algorithm with $t$ queries there is a $t$-dimensional Block Set with the same Gram matrix of final states.** This formulation is useful for exact quantum algorithms. We applied it successfully for a generalization of Deutsch-Jozsa algorithm.

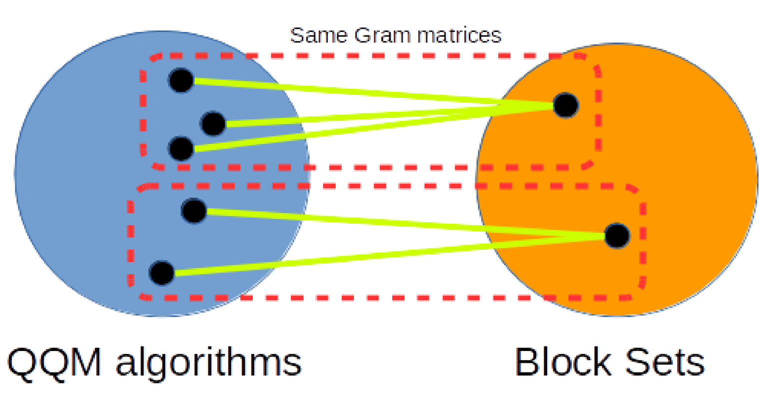


Fig. 1: The relation between both models is not bijective. The red box means that all BSF or QQM algorithm inside it have the same Gram matrix.


**Acknowledgements** The authors acknowledge financial support from CNPq and CAPES.


## Gram Matrices and an Application

We say that a Block Set is orthogonal if their elements are pairwise orthogonal. Let $\left\{\left|\Psi(k)\right\rangle \in H_Q \otimes H_W : k \in \mathbb{Z}_{n+1}^{t+1}\right\}$ be an orthogonal real Block Set for $(H_Q, H_W)$, then the Gram matrix of their output states $\left\{\left|\Psi_x^f\right\rangle\right\}$ is given by

$$G = 2\left(\sum_k \left(\bar{P}_k + \bar{Q}_k\right) \left\| \left|\Psi(k)\right\rangle \right\|^2\right) - J, \tag{2}$$

where $J$ is the matrix with all entries equal to 1. This explicit representation of the final Gram matrix gives us much control over the algorithm that we design.

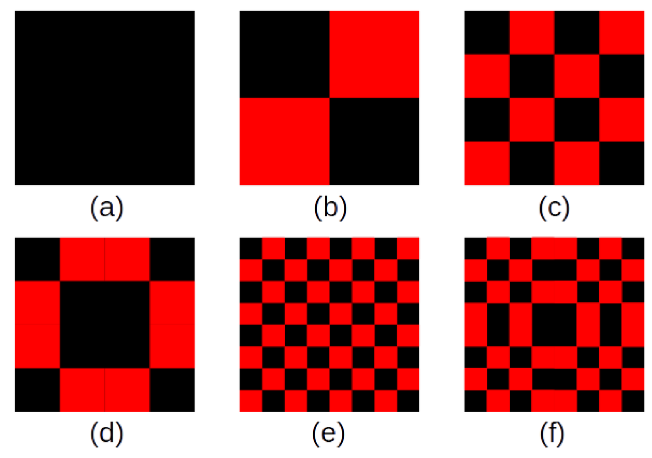


Fig. 2: In an orthogonal Block Set, each element $k$ controls a binary matrix $\bar{P}_k + \bar{Q}_k$ in Equation (2). Thus, each element has an independent influence over the Gram matrix. The figure represents such matrices for some elements in a two-dimensional orthogonal Block Set: (a) $(0, 0)$, (b) $(0, 1)$, (c) $(0, 2)$, (d) $(1, 2)$, (e) $(0, 3)$, (f) $(1, 3)$. Entries 1 are represented in black and entries 0 in red.

**By using orthogonal Block Sets, we can obtain a quantum exact algorithm that distinguishes two sets $X, Y \subset \{0,1\}^n$ in $t$ queries.** This is equivalent to the following problem. We denote by $\oplus$ the XOR operation between two binary inputs and by $XOR(X, Y)$ the set generated by the bitwise XOR operation amongst all elements in $X$ and $Y$.

**Definition 2** (XOR-Weighted Problem). *Let $\mathcal{J} \subset \mathbb{Z}_{n+1}^t$ and let $X, Y \subset \{0,1\}^n$. Take a set of boolean formulas $\left\{\bigoplus_{i \in j} x_i : x_0 = 0, j \in \mathcal{J}\right\}$, where each formula is associated to a value $w_j$. Find weights $w_j > 0$ satisfying both conditions: (i) $\sum_{j \in \mathcal{J}} w_j = 1$; and (ii) the sum of weights associated to formulas satisfied by $z \in XOR(X, Y)$ is equal to $1/2$.*

Solving the XOR-Weighted Problem in the BSF is equivalent to solving a linear system of equations using just positive variables which sum is 1. **Such reduction could be applied to any boolean function $f$ in order to find an exact quantum algorithm in BSF, even though an optimal solution is not guaranteed.**

Some QQM algorithms have a simple correspondence to BSF. For instance, the following BSF algorithm is equivalent to Deutsch-Jozsa algorithm. We define an orthogonal Block Set $\{\left|\Psi(i)\right\rangle : 0 < i \leq n\}$, such that $\left\| \left|\Psi(i)\right\rangle \right\|^2 = \frac{1}{n}$ for all $0 < i \leq n$. In terms of the XOR-Weighted Problem, this is equivalent to taking the set of formulas $\mathcal{J} = \{x_i : 0 < i \leq n\}$ and setting weight $w_i = \frac{1}{n}$ to each $x_i$ such that $0 < i \leq n$. In this case, if we take $X$ as the set of balanced inputs and take $Y$ as the set of constant inputs, then any element $z \in XOR(X, Y)$ satisfies exactly half the formulas from $\mathcal{J}$.

## Conclusions

Our theoretical result was the Block Set Formulation, which is a reformulation of the Quantum Query Model such that the unitary operators are replaced by phase inversions over a set of vectors. This contribution gives an alternative interpretation on how quantum query algorithms work. Our main algorithmic motivation is the application of the BSF to the challenging problem of constructing new exact quantum algorithms. With BSF, this problem may be reduced to a linear system and allow us to obtain non-trivial quantum exact algorithms for a wide range of functions. This kind of analysis should be much easier in BSF than in other linear formulations such as semi-definite programming.

## Selected Papers and Recommended Reading


- H. Barnum, M. Saks and M. Szegedy (2003), Quantum decision trees and semidefinite programming , In Proc. of the 18th IEEE Conf. on Computational Complexity.
- A. Ambainis (2013), Superlinear advantage for exact quantum algorithms , In Proc. of the 45th ACM STOC.
- A. Montanaro, R. Jozsa and G. Mitchison (2013), On Exact Quantum Query Complexity , Algorithmica.
- A. Ambainis, A. Iraids and J. Smotrovs (2013), Exact quantum query complexity of EXACT and THRESHOLD , arXiv:1302.1235.