Metaheuristics on Quantum Computers: Inspiration, Simulation and Real Execution

Zakaria Abdelmoiz Dahi ‡§ and Enrique Alba †

‡ *Dep. of Lenguajes y Ciencias de la Computación, Fac. ETSI Informática, University of Malaga, Spain*

§ *Dep. of Fundamental Computer Science and its Applications, Fac. NTIC, Constantine II University, Algeria.*

† *ITIS Software, Edificio Ada Byron, University of Malaga, Spain,*

Appendix A. Parameters Sensitivity Analysis

Considering that the *insp*-QCGA-*inv* is the starting point for answering this work's research questions (see Section 1 of the paper), it is important to understand its efficiency and behaviour in order to link them with those of the *sim*-QCGA and *real*-QCGA. Thus, to understand the main factors that influence the *insp*-QCGA-*inv*, this appendix presents a systematic study to provide further insight of the results got in Section 4.3.1 of the paper. It is also worth stating that such fine analysis could not be done for *sim*-QCGA and *real*-QCGA because the IBMQ quantum machines used in the present work are extremely constrained in terms of number of qubits, number of iterations allowed, etc.

Since that the *insp*-QCGA-*inv* is at the crossroads between a cellular EA (cEA) and a Quantum-Inspired EA (QIEA), this study investigates the main parameters that rule the efficiency of these types of algorithms: the *rotation angle* in QIEAs, the *grid shape* and *replacement strategy* in cEAs. With regard to the above explanations, a first analysis of the "*quantum*" facet of the *insp*-QCGA-*inv* is done. The latter is run using different values of rotation angle θ in the rotation gate: 1°, 5°, 10°, 15°, 30°, 45° and 60°. Then, the "*cellular*" side of the *insp*-QCGA-*inv* is investigated. This is done by setting the rotation angle θ to 1° and executing the devised approach using three grid shapes of different ratios: square (20×20), rectangular (10×40) and narrow (4×100). Finally, by setting θ to 1° and using a square 20×20 grid, the *insp*-QCGA-*inv* is run using two different replacement strategies: synchronous and asynchronous.

Table A.1 displays the *average* (i.e. `Mean`) and *STandard Deviation* (`STD`) of the `best`, `worst` and `mean` fitnesses achieved all along 30 executions and this when applying each setting. It is worth stating that this appendix includes, on purpose, these three comparison metrics and no statistical tests considering that the principal goal is not to establish the superiority/inferiority of a given configuration over others, but aims at establishing the type of influence the grid's shape, rotation angle and the replacement strategy could have on the proposal. In addition, Figures A.1(a-d) show the influence of each parameter setting on the fitness evolution (a and b: rotation angle, c: grid shape and d: replacement strategy). Similarly to Figures A.1(a-d), Figures A.2(a-d) display the impact of the parameters' changes on the convergence of the population. The latter is expressed in terms of variance of each qubit in the quantum register, which is represented by a *thin* (i.e. one-line) plot in Figures A.2(a-d). It is important also to note that all results displayed in both Figures are those attained when tackling network 1 of size 8×8 cells. The choice was made for this size of networks because it has been noticed that the results are enough representative (*but not identical*) to the remaining sizes.

Considering the results in Table A.1, it can be seen that the value of the rotation angle θ and the replacement strategy are having a greater impact on the *insp*-QCGA-*inv* efficiency than the grid shape does. Besides, according to Figures A.1(a-d), it can be noted that, unlike the grid shape that results in almost-identical plots, the rotation angle θ and the replacement strategy are the parameters that are creating various and more interesting *insp*-QCGA-*inv* fitness evolution plots. Actually, when looking at Figures A.1(a and b), it can be noticed that, depending on the value of θ, the fitness plot contains at some points, a "*stagnation*" (straight horizontal line) or an "*evolution*" phase (stairs-like line). Although, the order in which these phases appear might change (e.g. stagnation-evolution-stagnation or evolution-stagnation). Moving to the evolution phase, when setting $\theta = 1°$, the evolution starts around 91 iterations and ends around 151 iterations. As the value of θ increases, the bounds of this evolution phase seem to shrink (i.e. evolution takes a shorter time) and move towards the beginning of iterations (i.e. happens a few iterations after the *insp*-QCGA-*inv* starts running). So, these pieces of evidence confirm that the θ value has a direct influence on the *insp*-QCGA-*inv*'s efficiency. It can be said that the increase of the θ value seems to have some sort of "*geometrical translation*" on the *evolution step* that is supposed to occur during the search process. So, for example, such knowledge can be used to provoke and make such evolution to happen as needed (e.g. design necessities). Although, it is important to bear in mind that even if the influence of the θ value on the *insp*-QCGA-*inv*'s efficiency is clear, no firm conclusions can be made on the real influence of the replacement strategy.

Moving now to the variance evolution displayed in Figures A.2(a-d), a clear observation is that the change of grid shape has no effect on the variance evolution (i.e. convergence of the population). Indeed, for different configurations of this parameter, the variance evolution remains the same in terms of curves, maximum and minimum values (see Figure A.2(c)). On the other hand, like in Figures A.1(a, b and d), in Figures A.2(a, b and d) also, the change of θ value and replacement strategy are

Table A.1: Parameters' sensitivity study: θ value, grid shape and replacement strategy

Network		Best		Worst		Mean	
		Mean	STD	Mean	STD	Mean	STD
4×4 (1)	**θ Value**	98535.000	0.000	98535.000	0.000	98535.000	0.000
4×4 (2)		97156.000	0.000	97404.286	316.903	97184.486	50.011
4×4 (3)		95038.000	0.000	95463.429	1125.578	95151.448	300.154
6×6 (1)		181589.143	5455.231	197631.000	8503.906	191513.524	7694.022
6×6 (2)		191019.571	4207.060	204421.429	8467.134	199200.710	7599.819
6×6 (3)		181474.857	4178.164	193490.143	8038.871	187699.243	5829.663
8×8 (1)		333771.857	6198.923	348166.000	9177.440	342139.424	7668.402
8×8 (2)		323531.714	8050.588	336644.714	7990.820	330533.348	7973.613
8×8 (3)		293434.143	8161.916	308843.857	8770.026	302636.910	8340.433
10×10 (1)		433236.714	6271.069	445712.857	5617.818	440367.900	5425.007
10×10 (2)		394446.000	4591.840	405513.857	4282.762	400566.229	4803.600
10×10 (3)		403683.429	3967.614	413336.000	4007.358	409955.348	3508.929
4×4 (1)	**Grid Shape**	98535.000	0.000	98535.000	0.000	98535.000	0.000
4×4 (2)		97156.000	0.000	97156.000	0.000	97156.000	0.000
4×4 (3)		95038.000	0.000	95038.000	0.000	95038.000	0.000
6×6 (1)		173701.000	0.000	183303.667	2544.019	177410.167	541.476
6×6 (2)		182510.333	310.614	187829.667	1154.923	184541.511	762.980
6×6 (3)		175014.333	436.367	180114.667	1578.980	177406.989	420.588
8×8 (1)		321354.667	2106.002	331566.667	314.249	327345.422	745.243
8×8 (2)		307509.333	1715.420	320472.000	1120.387	315589.800	392.138
8×8 (3)		279757.667	1858.555	292335.667	684.453	286995.744	399.402
10×10 (1)		421961.000	1523.477	436263.000	1027.070	430147.578	846.973
10×10 (2)		385797.333	1638.048	396345.667	1007.355	392111.967	309.959
10×10 (3)		398704.667	785.255	405926.000	284.944	403130.311	221.547
4×4 (1)	**Replacement Strategy**	98535.000	0.000	98771.500	334.462	98558.883	33.776
4×4 (2)		97156.000	0.000	97207.000	72.125	97159.400	4.808
4×4 (3)		95038.000	0.000	96527.000	2105.764	95137.267	140.384
6×6 (1)		180042.000	8967.528	189481.000	10859.746	184342.967	10125.816
6×6 (2)		187300.500	7027.934	196130.500	11617.057	191840.633	10989.712
6×6 (3)		178486.000	5610.185	184795.500	9074.301	182082.717	7294.726
8×8 (1)		326632.500	4488.007	341861.500	15030.969	336648.383	12651.908
8×8 (2)		316230.500	12556.095	328959.000	10845.604	323596.133	11915.598
8×8 (3)		285485.500	8287.999	302184.500	14012.735	296155.683	13270.485
10×10 (1)		432916.000	17961.926	450629.000	21989.607	442839.033	19250.652
10×10 (2)		395433.500	13297.143	408471.500	15547.157	402089.167	14573.188
10×10 (3)		403247.500	7703.928	415267.000	13572.208	411376.633	11501.940

those resulting in more diverse and thought-provoking plots. In addition, like it is the case for the fitness evolution displayed in Figures A.1(a, b and d), the variance evolution plots in Figures A.2 (a, b and d) are composed of two typical phases; stagnation and evolution, where the order in which they might come differs from one parameter setting to another. The stagnation zones can be defined as a straight horizontal line in the plot, while the evolution phases can be described as the moment the plot starts going up in concordance with the variance value increasing until it reaches a peak (i.e. maximum), then starts going down again. The moment when the plot goes back to a monotone straight-line evolution marks the end of the evolution step.

Regarding now the Figures A.2(a and b), it can be seen that as the value of the rotation angle θ increases, the evolution zones appear to move closer to the origin of the plot (i.e. the first algorithm's iterations). They also seem to shrink in terms of how long the evolution lasts. This remark is identical to the one made for the fitness value evolution in Figures A.1(a and b). In fact, the fitness evolution phases in Figures A.1(a and b) start and end at the same moment (*approximately*) as the variance evolution in Figures A.2 (a and b). It might be thought that the results in Table 3 of the paper are explained by the fitness behaviour in Figures A.1 (a and b). The latter is resulting from the variance evolution in Figures A.2 (a and b). Indeed, it can be seen from results in Table A.1, that some configurations of θ value seems to make the *insp*-QCGA-*inv* perform better than when using other θ values. This can potentially be explained by the fact that for each instance of the problem, the *insp*-QCGA-*inv* needs to start evolving the solution at a different moment. This what seems to provide the fact of using different θ values of the rotation gate. In fact, each setting makes the population evolve or converge at a different moment.

In conclusion, it can be supposed that the *insp*-QCGA-*inv* efficiency is driven by the rotation angle θ, which affects the convergence of the quantum registers in the population. The value of θ controls when the convergence starts and ends and, therefore, when the *insp*-QCGA-*inv* search starts and ends. This makes the *insp*-QCGA-*inv* an easy-to-control-and-predict solver that is controlled by only one parameter. An obvious way to make use of such knowledge is to conceive an intelligent way to decide "*when*" to change the value of the rotation angle.

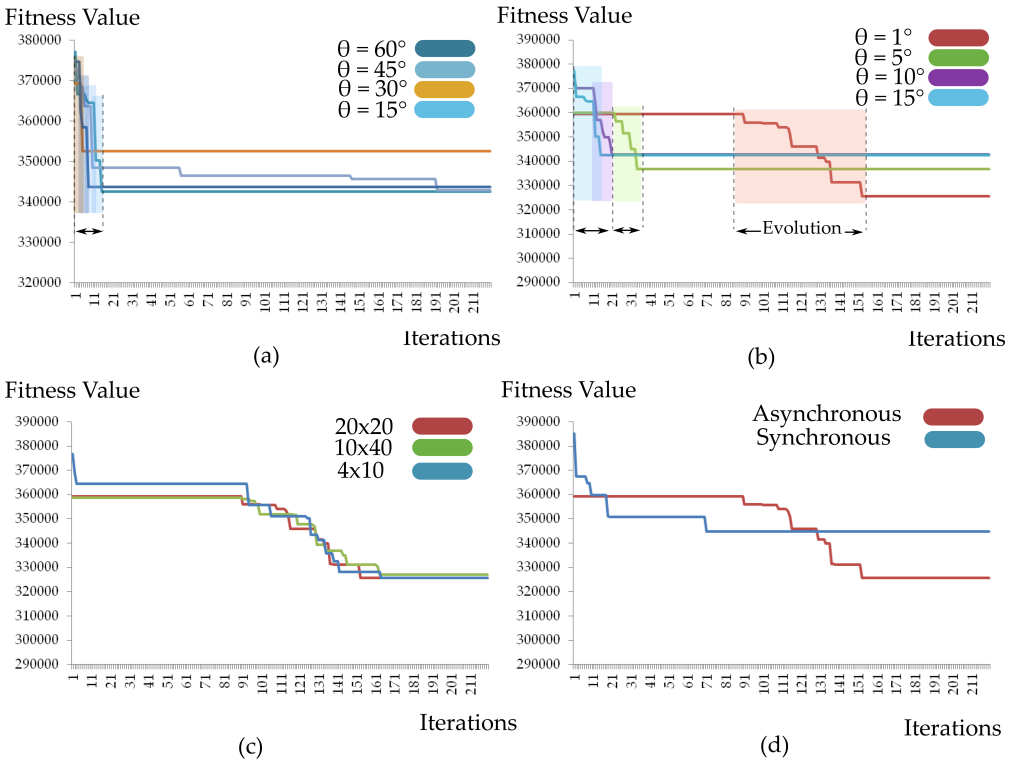

Figure A.1: Fitness evolution: (a and b) θ value, (c) grid shape and (d) replacement strategy

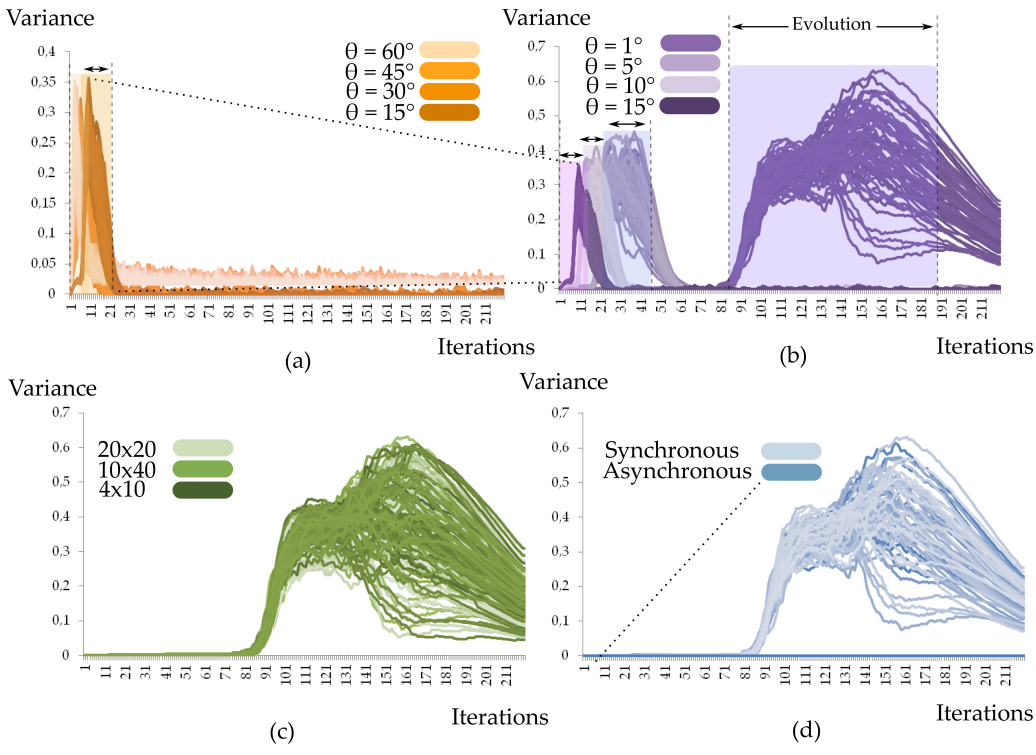

Figure A.2: Variance evolution: (a and b) θ value, (c) grid shape and (d) replacement strategy