Chapter 2

# Quantum walks with time-dependent Hamiltonians and their application to the search problem on graphs

The main goal of this thesis is to study quantum walks with time dependent Hamiltonians, focusing in particular on their application to the spatial search problem on graph. The general idea is trying to improve a time-independent implementation of quantum walks search using a time-dependent Hamiltonian analogous to the one used in the adiabatic evolution. In order to determine whether this new approach produces successfull results we study two selected graph topologies: the cycle graph, for which the time-independent approach is not able to solve the search problem, and the complete graph for which the search problem is solved for both the time-independent and adiabatic evolution approaches.
We compare the two methods for the *optimized-search*, *localization* - which represents a search without needs to optimize the time - and a measure of *robustness*.

## 2.1. Search with time-dependent Hamiltonian

In Section 1.4 we discussed the use of quantum walks for the spatial search problem [3] and the application to the complete graph where the search problem is solved. We then illustrated how the adiabatic theorem can be used to solve computational problems with an adiabatic evolution of a quantum system [4]. Efforts to combine the two approaches showed that the search problem can be solved only with structures stronger than the usual Grover's oracle [13], thus making an *Adiabatic-Quantum-Walk-Search* impossible. However this leaves space to a merely time-dependent quantum walk search, that takes advantage of a time-dependent implementation similar to the adiabatic evolution but that is not bounded to the strict adiabatic-theorem conditions and the limitation of the standard Grover's oracle.

### 2.1.1 Time-dependent quantum walks

Following from the adiabatic evolution discussed in the preliminaries, we consider a search Hamiltonian that interpolates between an initial Hamiltonian, i.e. the Laplacian of the graph $L$, and the final oracle Hamiltonian $H_w = \gamma|w\rangle\langle w|$

$$H(s) = (1-s)L - s\gamma|w\rangle\langle w| \tag{2.1}$$

where the interpolation schedule $s = s(t)$ goes from 0 to 1 as the time $t$ goes from 0 to the runtime $T$.

In order to find the evolved ground state of the beginning Hamiltonian we need to determine the evolution operator that for a time-dependent Hamiltonian is given by:

$$S(t,t_0) = \mathrm{T}\ \exp\Big\{-i\int_{t_0}^{t} dt' H(t')\Big\} \tag{2.2}$$

where T is the time-ordering operator. Since we are only interested in the evolved state, having the exact evolution operator is somewhat irrelevant. We therefore proceed by solving the differential Schroedinger equation:

$$i\frac{d}{dt}|\psi(t)\rangle = H|\psi(t)\rangle \tag{2.3}$$

Recalling that we are dealing with matrices and vectors in an N-dimensional Hilbert space, we solve N-differential equations of the form

$$\frac{d}{dt}|\psi_i(t)\rangle = \sum_j H_{ij}|\psi_i(t)\rangle \tag{2.4}$$

with the boundary conditions $|\psi(0)\rangle = |\psi_0\rangle$, where the latter is the ground state of the Laplacian.

### 2.1.2 Interpolating schedule s(t)

As pointed by Wong the interpolating schedule s(t) plays a crucial role in the evolution of the system and in the overall scaling of the algorithm. The original adiabatic evolution by Farhi and Gutmann [4] uses a linear interpolating schedule defined as $s_L(t) =\equiv \frac{t}{T}$. Roland and Cerf show that in order to obtain a quadratic speedup for the complete graph a non linear schedule is essential [11][8].

Thus, from a linear interpolating schedule we consider also quadratic and cubic schedules:

$$s_S(t) \equiv \sqrt{\frac{t}{T}} \qquad s_C(t) \equiv \sqrt[3]{\frac{t}{T}} \tag{2.5}$$

We then consider the interpolating schedule analitically derived by Roland and Cerf for the unstructured search discussed in Section 1.5.3. As we have already mentioned the shape of this interpolating schedule follows from gap $g(s)$ between the lowest two eigenvalues, changing faster when the gap is large, while it evolves slower when the gap is small. We therefore take these key aspects of $s(t)$ - derived for the unstructured search - and consider a similar non-linear interpolating schedule defined as follows

$$s_{NL}(t) = \frac{1}{2}\Big[\Big(2\frac{t}{T} - 1\Big)^3 + 1\Big] \tag{2.6}$$

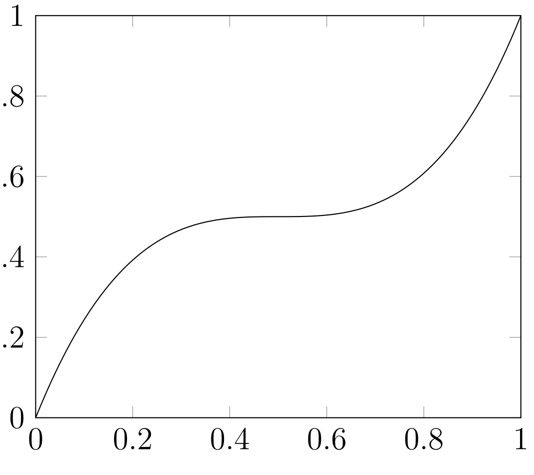


Figure 2.1: **Non-linear interpolating schedule $s_{NL}(t/T)$.** The figure shows the non-linear interpolating schedule $s_{NL}$ that we introduced to improve the standar linear one of Farhi and Gutmann [4]. It draws the key aspects from the interpolating schedule derived by Roland and Cerf [11], namely it evolves faster for large energy separation $g$ and slower for small separation.

### 2.1.3 Multiple runs for one search

To give a complete picture of the usefulness of the time-depedent approach, we consider the possibility of repeating the search multiple times. If the probability at which the solution is found is $p = 1$ the search is *perfect* and the problem is solved. However, if the proabability is less than one, i.e. *imperfect search*, the problem can be solved by searching multiple times . Since the results of the search are checked independently, a single successfull search is sufficient, and this kind of routine is efficient as long as the probability $\left|\langle\psi(T)|w\rangle\right|^2$ is greater than $1/poly(N)$ (where N is the dimension of the graph) [8] - which as we shall later see is verified for all the scenarios considered.

Repeating the search multiple times does however come at a cost. It is indeed necessary to take into account for a non-zero *initialization* time $t_{init}$ to prepare the system in the correct state as well a physical time for the measurement. Therefore, computing multiple searches with small $T$ becomes less efficient than less searches with larger $T$, where the quantity $t_{init}$ makes a lesser contribution to the overall $T$. Clearly this consideration is particularly relevant for increasing graph size.

## 2.2. Selected topologies: cycle and complete graph

Throughout our analysis we will focus on two selected graph topology, the cycle graph $Cy(N)$ and the complete graph $C(N)$.

As previously discussed in Section 1.4.1 and **??** the **complete graph** represents the best case scenario since it has been shown to solve the search problem both for the standard time-independent quantum walk approach [3] and the local adiabatic evolution [11], with a quadratic speed up. An adiabatic implementation of the quantum walk search does not work with the usual Grover's oracle, requiring a more elaborate structure [13], however as we shall later see a merely time-dependent approach might give interesting results in terms of robustness and localization.

The **cycle graph** on the other hand is not able to solve the search problem with the time-independent approach, and can give some interesting insights on the performance of the time-dependent quantum walk search.

The goal is therefore to study the time-dependent approach on a non-working topology - the cycle graph - and for completeness show its application to the already perfect search on the complete graph.

## 2.3. Characterization of the results: Search, Localization and Robustness

Firstly it is necessary to define what type of results we are looking for. We begin by showing the difference between **search** and **localization**. Then we introduce a measure for the **robustness** of the search algorithm.

### 2.3.1 Search vs Localization

In order to study the performance of the search algorithms we have to characterize two particular classes of results, the (optimized)**search** and the **localization**, that help us decide whether the time-dependent approach brings any advantages.

- the (optimized) **search** describes the usual search, namely the finding of the solution with high probability (possibly unitary) for the smallest time as possible. As previously mentioned we also take into account the possibility of repeating the search multiple times.

- we call **localization** the finding of the solution with high probablity without the need to optimize the time. This description becomes necessary if we take into account the adiabatic nature of the time-dependent approach, that guarantees unitary probability for large $T$.

### 2.3.2 Robustness

We have seen that the probability of a search problem depends on the time $T$ at which the quantum state is measured and the parameter $\gamma$. The optimal highest probability clearly is given by the optimal combination of $T$ and $\gamma$. These two parameters might be affected by noise or perturbation, leading to variation from the maximum probability. Therefore it is interesting to define a quantity that is able to quantify this phenomenon.

Following from [12] we define the **robustness**, a quantitative measure representing the variation on the probability due to some perturbation/noise on $\gamma$. We begin by finding the highest probability $p$, evaluated with the single or multiple run for one search approach. For the corresponding $(T, \gamma)$ combination we evaluate the robustness as follows:

$$R^{\pm} = p(T, \gamma) - p(T, \gamma \pm \delta) \tag{2.7}$$

where $\delta$ is some positive perturbation of the $\gamma$ parameter. As we shall later see this quantity is given in terms of some percentage of $\gamma$. To find a unique value for the robustness an average of $R^{\pm}$ is done:

$$R_{\gamma} = \left[\frac{R^{+} + R^{-}}{2}\right] \tag{2.8}$$

The quantity R should be positive[1], since the $(T, \gamma)$ combination corresponds to the hightest probability. Additionally we also notice that the perturbation on the parameter has equal probability of being positive or negative, thus the average ponderates between these two possibilities.

[1] Although not considered in our work, it is also possible to evaluate the robustness for non-maximal probability. In that particular scenario the value of the robustness could be negative.

As we mentioned, the variation in probability could also be due to some error on the time $T$ at which the state is measured. We can extend the measure of robustness to this particular scenario, where the value is similarly evaluated:

$$R = p(T, \gamma) - p(T \pm \tilde{\delta}, \gamma) \tag{2.9}$$

and $R$ follows directly from Equation (2.8). In this scenario the robustness in respect to $T$ is not necessarily positive, in particular for the time-dependent approach, since due to the consequences of the adiabatic theorem the probability increases with time. In order to differentiate between these two measures of robustness we call them **$\boldsymbol{\gamma}$-robustness** and **$\boldsymbol{T}$-robustness** respectively.

Although the value of $R$ does not have any absolute physical significance, it fits well for our specific scenario where the interest is focused on the comparison between two specific approaches. Therefore this quantity will be used to characterize a particular approach as **more** or **less robust**, where the first is characterized by a smaller value of $R$ compared to the latter.

## 2.4. Results for the cycle graph

We now address the results for the cycle graph. We begin by evaluating the probabilities for the time-independent and the time-dependent Hamiltonians. We compare the localization of the two approaches, followed by the search - which requires the introduction of a new quantity that takes into account the possibility of doing multiple runs for one search. Lastly we study the robustness of the time-dependent approach and determine whether the newly introduced Hamiltonian makes the search more robust than the time-independent one.

### 2.4.1 Time-independent benchmarks

The first step of the analysis is to compute time-independent benchmarks. We comput the probability over a $(T, \gamma)$ grid, with $T \in [0, N]$ and $\gamma \in [0, 2.5]$, where N is the dimension of the graph considered. An initial run shows that the probability does not increase with time (Figure 2.7), thus the need to evaluate for $T$ greater than $T = N$ proves to be unnecessary. In addition, Grover's algorithm for the unstructured search has a time scaling of $O(\sqrt{N})$ and the Farhi and Gutmann's global adiabatic evolution is $O(N)$, thus focusing on $T \leq N$ suffices.

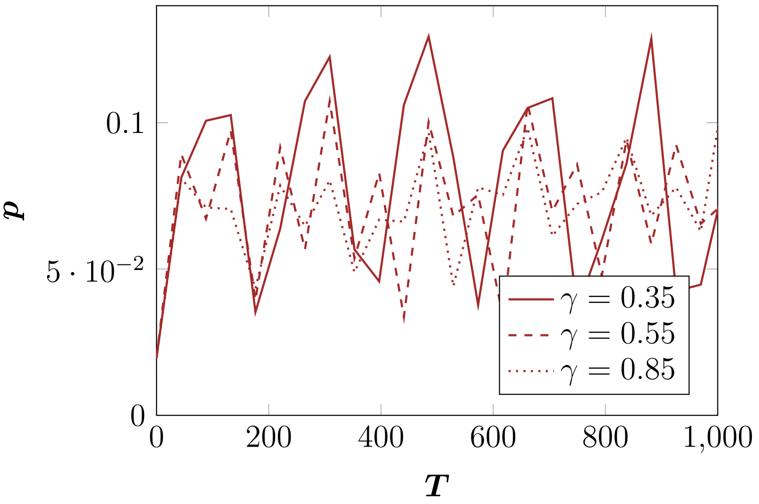


Figure 2.2: **Probability for a $Cy(51)$ with sampled $\gamma$**: The figure shows the probability for the cycle graph $Cy(51)$, evaluated with the time-independent Hamiltonian and a few sampled $\gamma$. We can see that the probability does not increase significantly with time, therefore we can focus our analysis on $T \leq N$.

Throughout the analysis we will consider graphs up to $N = 71$, with only odd dimensions which makes a easier oracle placement in the center vertex of the graph[2].

To display the results in an intuitive way we used a heatmap plot, which gives a good idea on the probability for varying combinations of $(T, \gamma)$:

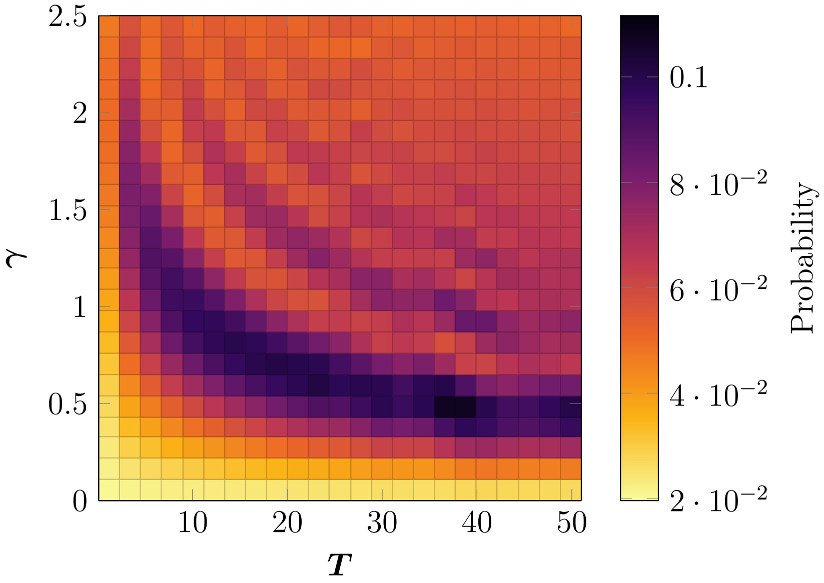


Figure 2.3: **Probability distribution for the time-independent Hamiltonian.** The figure shows the probability distribution for the cycle graph $Cy(51)$ evaluated using the time-independent Hamiltonian, providing the necessary benchmarks. Note that dark color regions do not represent probabilities close to one.

### Comments on the probability distribution[3]

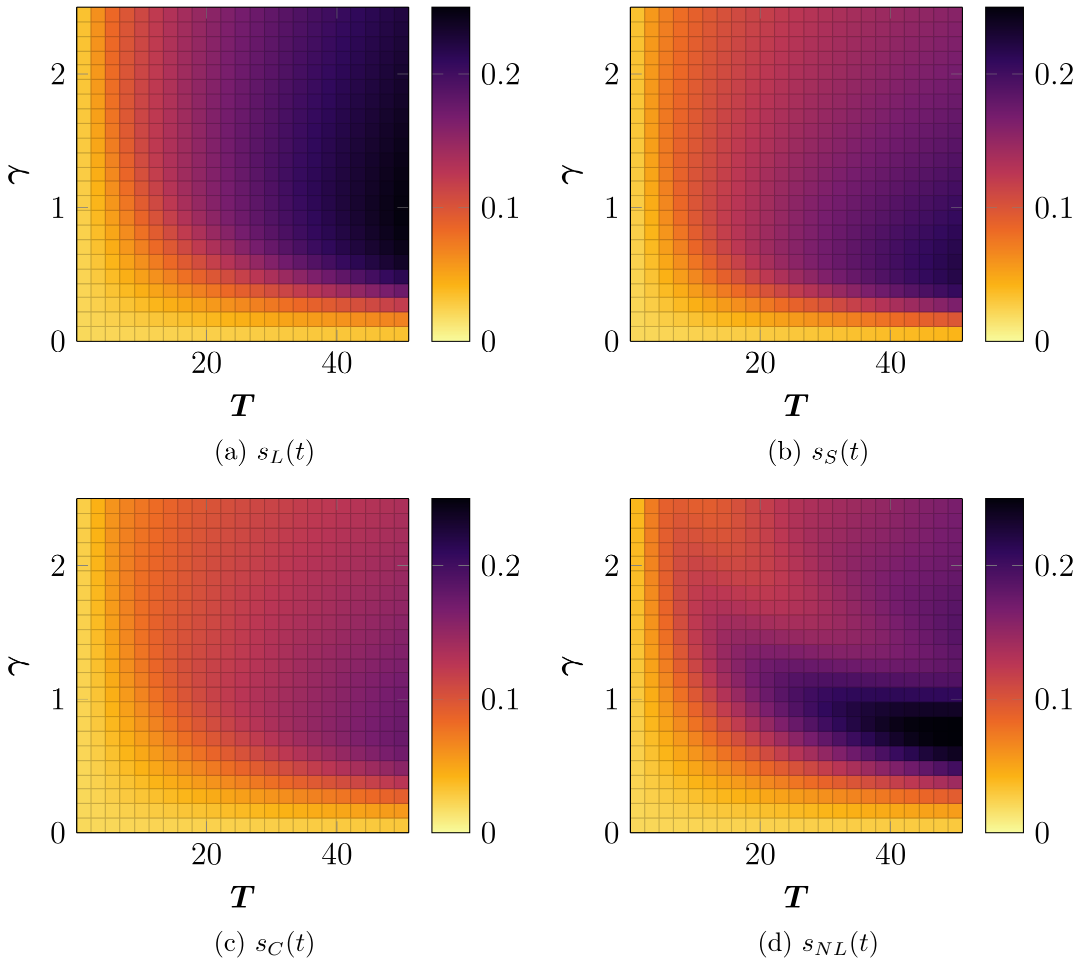
In Figure 2.3 we see that the probability does not increase smoothly with increasing time, but shows peaks (dark regions) and valleys (lighter regions). Additionally high probability regions are scattered throughout the grid and do not appear necessarily for large $T$. We will later see that this is a weakness of the time-independent approach, since a small variation of the parameter $\gamma$ leads to possibly great variation of the probability.

[2] The position of the oracle is irrelevant for the graph topologies considered, since every node is equivalent to all the $N-1$ others. Nevertheless odd dimensions allows to place the oracle in the central vertex of the graph, namely $|\frac{N+1}{2}\rangle$.

[3] The use of *probability distribution* is not to be intended for the meaning it has in probability theory. In our scenario it describes the distribution of the probability for the combinations of $(\gamma, T)$

### 2.4.2 Time-dependent results

Similarly we compute the probability with the time-dependent Hamiltonian using the interpolating schedules introduced in Section 2.1.2. To easily compare the two methods we consider the same time $T = N$ and $\gamma$ used for the time-independent benchmarks. Indeed, from an initial run we see that the $\gamma$ parameter affects the probability similarly to the time-independent approach, namely the probability tends to be higher for smaller values of $\gamma$.
The probability is then evaluated for all the interpolating schedules $s(t)$ , and the results are again presented with an intuitive heatmap plot.



(a) $s_L(t)$

(b) $s_S(t)$

(c) $s_C(t)$

(d) $s_{NL}(t)$

Figure 2.4: **Probability heatmap plot for the time-dependent Hamiltonian, for different shapes of s(t) and $Cy(51)$.** The figure shows the probability heatmap plots for a circular graph of $N = 51$ evaluated using the time-dependent Hamiltonian using the following interpolating schedules: (a) linear, (b) $s_S(t)$, (c) $s_C(t)$ and (d) non-linear $s_{NL}(t)$.

### Comments on the probability distribution

Compared to the time-independent Hamiltonian approach we can clearly see that the probability distribution is smoother - has no valley and peaks - for both constant $\gamma$ and $T$ (any orizontal and vertical sections, respectively). In particular we notice that the probability increases for increasing time, and as we shall later see, for large enough $T$ it reaches probability equal to one.

If we look at the different interpolating schedules it is immediately evident that $s_S(t)$ and $s_C(t)$ schedules perform poorly compared to the linear $s_L(t)$. On the other hand $s_{NL}(t)$ schedule performs similarly to $s_L(t)$, but it has a significantly different probability distribution that might affect its robustness.

### 2.4.3 Comparison: localization

We now compare the localization properties of the two algorithm. As we have already mentioned, from an intial look at the probability heatmap plots we discovered the following:

- **Time-Independent QW**: the time-independent based algorithm is not able to solve the search problem with a single iteration, making it necessary to run multiple searches. This implies that the approach does not show any localization properties, in fact the probability does not increase with time as seen in Figure 2.3.
- **Time-Dependent QW**: the time-dependent based search on the other hand solves the search problem with a single iteration, although that happens for large values of $T$, as can be seen in Figure 2.5. This is indeed a consequence of the *adiabatic inspired* implementation, for which at large $T$ the probability goes to one.

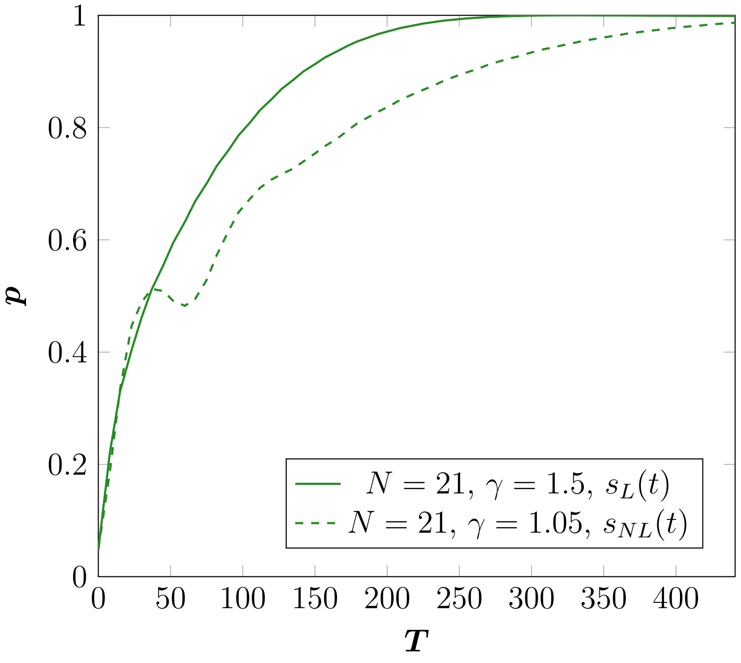


Figure 2.5: **Probability for a $Cy(21)$ with sampled $\gamma$ up to large $T$.**: The figure shows the probability distribution for the cycle graph $Cy(21)$, evaluated with the time-dependent Hamiltonian using linear $s_L$ (solid) and non-linear $s_{NL}$ (dashed) interpolating schedules. We can see that the probability increases with time, as expected.

Although the time-dependent approach is able to get to unitary probability for large $T$, it is able to produce large enough probabilities in much less time, as we can see from this plot. This is a consequence of the fact that the probability does not grow linearly with time, thus needing larger $T$ closer it gets to $p = 1$. If we look at the following plot - evaluated for a $Cy(51)$ and linear $s_L(t)$ - we can see that the algorithm solves the search problem with $p > 0.99$ in a time $T \approx 1200$, while it is able to solve it with $p > 0.8$ in $T/2 \approx 600$.

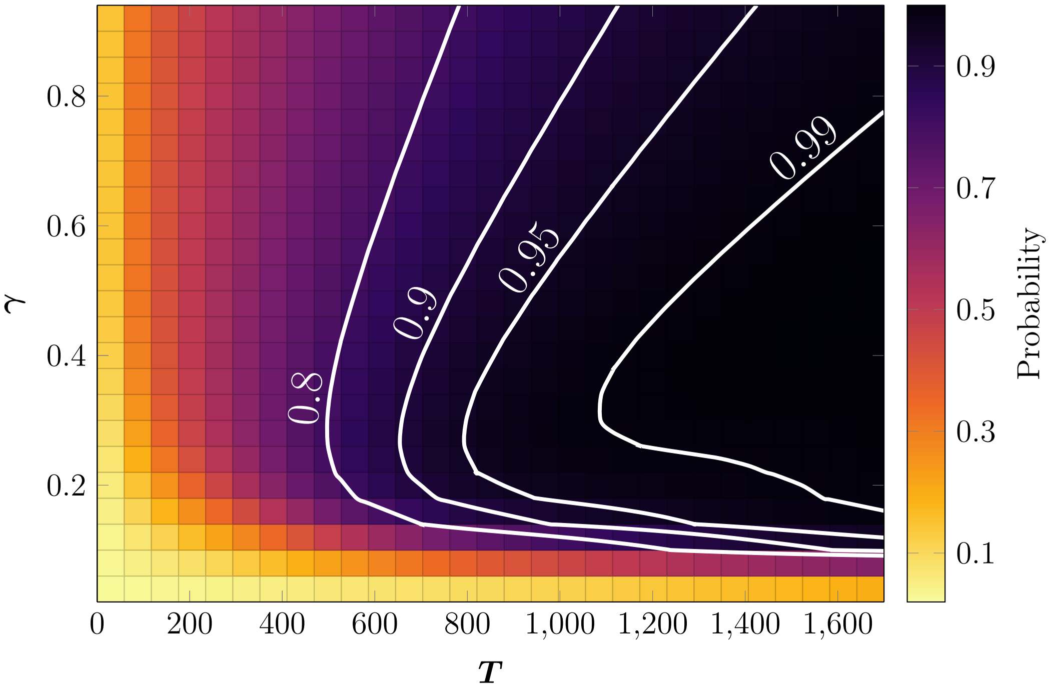


Figure 2.6: **Localization at large $T$ for the time-dependent Hamiltonian**. The figure shows the probability heatmap for the time-dependent Hamiltonian using the linear interpolating schedule $s_L(t)$ up to large values of $T$. Although, as we can see from the white contour lines, the algorithm gets to $p > 0.99$ for a large value of $T$, it is able to produce $p = 0.9$ in $T/2$ and $p = 0.8$ in $T/3$.

### 2.4.4 Comparison: search

In order to compare the two approaches for the search it is clear that we cannot simply consider the time at which the solution is found with unitary probability, since that particular $T$ does not exist for the time-independent approach and is not optimized for the time-dependent one as seen for the localization results. Therefore, as previously mentioned, we consider the possibility of doing multiple runs for one search. For this reason we introduce the following quantity

$$\tau = \min \left( \frac{T}{p} \right)_{T,\gamma} \tag{2.10}$$

where the minimization is done over $T$ and $\gamma$ [4]. The quantity $1/p$ represents the number of runs necessary to get to unitary probability (statistically), and combining it with $T$ gives the total time necessary get to $p = 1$. Minimizing over the combination of $T$ and $p$ gives the smallest time necessary to solve the search problem with unitary probability using the multiple runs approach.
The minimization thus consists in finding for fixed $T$ the maximum probability, evaluating then the quantity $T/p$ and finding the minimum.

Additionally, the number of runs performed - from now on referred to as **iterations** - given by

$$I = \min \left( p^{-1} \right)_{T,\gamma} \tag{2.11}$$

might give some useful insights on the performance of the approach, in particular if you take into account the initialization time $t_{init}$, since large number of $I$ are penalized by such $t_{init}$.

However, this approach poses a few problems. In fact if we look at how the quantity $T/p$ varies for varying $T$, we discover that the minimum of such quantity always corresponds to the smallest $T$ available, regardless of the type of Hamiltonian and interpolating schedule $s(t)$ considered. The following plot indeed shows, for a sampled $\gamma$, the shape of $T/p$ with increasing time.

[4] Remember that the probability depends directly on the combination of $T$ and $\gamma$.

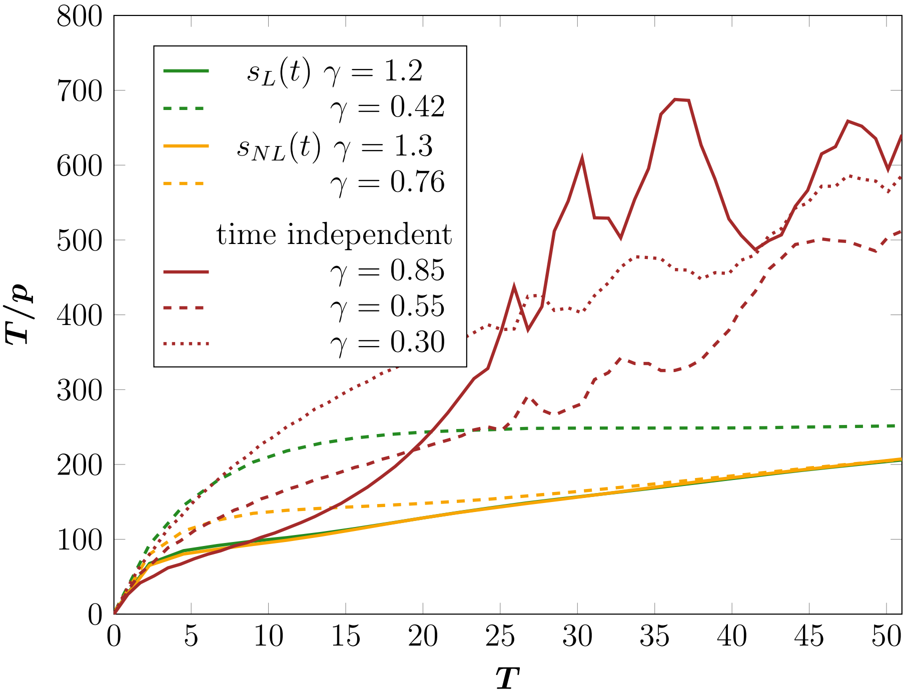


Figure 2.7: ***T*/*p* for sampled *γ*, showing that *τ* always corresponds to the smallest *T*.** The plot shows $T/p$ for some sampled values of the $\gamma$ parameter, using the linear $s_L$ (green) and non-linear $s_{NL}$ (orange) interpolating schedules for the time-dependent Hamiltonian, and the time-independent Hamiltonian (red). It is clear that $\tau = \min(T/p)$ will always correspond to the smallest $T$ available, regardless of the interpolating schedule and the type of Hamiltonian, requiring therefore a constrain on the time.

In addition, in Section 2.1.3 we mentioned that when dealing with multiple runs we need to take into account an initialization time $t_{init}$. It becomes therefore necessary to introduce a minimum value of time, since for small $T$ the contribution given by $t_{init}$ can have a significant impact. We call this particular $T$ **lower bound time**, reffered to as $T_{\min}$, and study its effect on $\tau$ and the overall performance of the time-independent and time-dependent algorithms.

### $\tau$ and $I$ for increasing lower bound time $T_{\min}$

We begin by studying how $\tau$ and $I$ vary with increasing lower bound time $T_{\min}$ . In the following plot we show the shape of $\tau$ with the time-independent approach and the time-dependent one; since we are interested in the general effect of $T_{\min}$ on $\tau$ and $I$ we only consider the linear interpolating schedule $s_L(t)$[5]. For

[5] We have seen in Section 2.4.2 the probability distribution for the different interpolating schedules is quite similar, therefore looking only at $s_L(t)$ is sufficient.

sake of semplicity we study the cycle graph $Cy(51)$ with time up to $T = N$

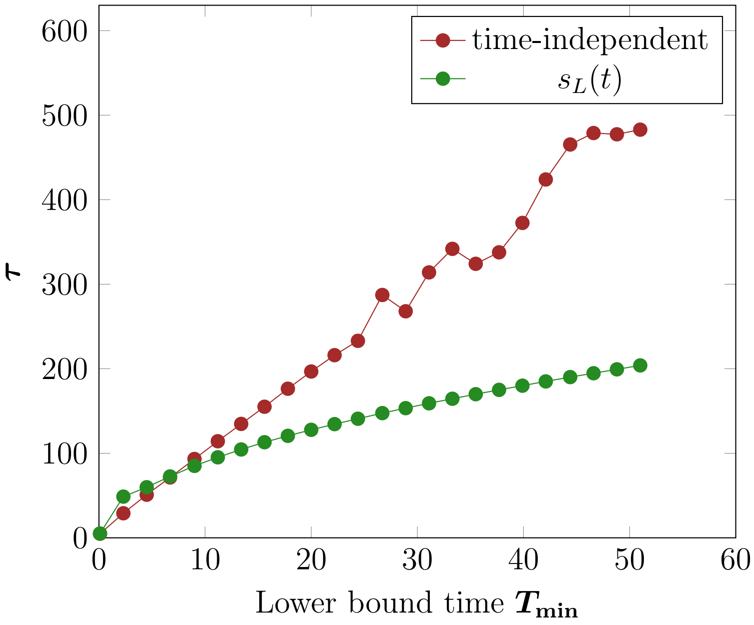


Figure 2.8: **$\tau$ for increasing lower bound time $T_{\text{min}}$, $Cy(51)$.** The plot shows $\tau$ for increasing values of lower bound time $T_{\text{min}}$, using the time-independent Hamiltonian (red) and time-dependent Hamiltonian (green) with $s_L(t)$ and evaluated for a $Cy(51)$. We see that for times smaller than a characteristic time $T^*$ the time-independent approach performs slightly better, while for large time the time-dependent one performs significantly better.

As we can see from the plot, the distributions can be divided into two sections marked by a particular $T^*$, representing the time at which the two distributions - time-dependendent and time-independent - intersect (for the time being the value of such time is not of our interest):

- for $T_{\text{min}} < T^*$ the time-dependent approach has a comparable performance with the time-independent one, although the latter has a slight advantage.
- for $T_{\text{min}} > T^*$ however the time-dependent approach performs significantly better, in particular with increasing $T_{\text{min}}$

The behaviour for large $T$ is to be expected, considering that the time-dependent approach shows localization properties and the probability increases with increasing time as we showed in Figure 2.6 and in Section 2.4.3, in contrast with the time-independent approach that does not show localization properties.
What this shows is that the choice of $T_{\text{min}}$ has a great impact on the outcome of

our time-dependent approach, making it a successfull or unsuccesfull alternative.

Let us now look at the distribution of the number of iterations $I$. The following plots shows $I$ for increasing $T_{\text{min}}$ up to $N$, using the time-independent Hamiltonian and time-dependent one with the linear interpolating schedule $s_L(t)$.

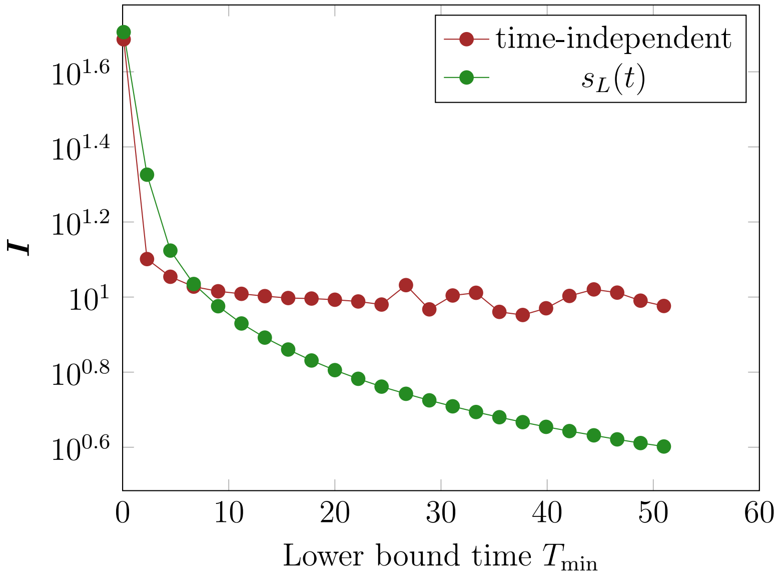


Figure 2.9: ***I* for increasing lower bound time $T_{\text{min}}$.** The plot shows $I$ for increasing values of lower bound time $T_{\text{min}}$, using the time-independent Hamiltonian (red) and time-dependent Hamiltonian (green) with linear $s_L(t)$ and evaluated for a Cy(51). This distribution reflects the probability distribution of the two approaches: for the time-independent Hamiltonian the probability does not increase with time, resulting in a (almost) constant $I$, while the time-dependent Hamiltonian showing localization properties requires less iterations to get to unitary probability. Note that the plot is given with logscale y axis, which helps to highlight the time-independent trend.

The iterations trend reflects the overall probability distribution of the time-dependent and time-independent Hamiltonian approaches. For small lower bound time the two approaches show a similar performance: the probability is very small, thus requiring a large number of iterations to get to unitary probability. As $T_{\text{min}}$ increases we see two very different trends:

- The time independent approach requires an almost constant number of iterations[6]. This reflects the non-localization properties of this particular approach, for which the probability does not increase with time. It also shows that the maximum probability found is (almost) equal for all $T$, provided that $T_{\text{min}}$ is large enough.

[6] The numerical value of $I$ is irrelevant since it reflects only the Cy(51).

- On the other hand the time-dependent approach requires less iterations to solve the search with unitary probability, as expected.

Taking into account the multiple runs and the initialization time $t_{init}$ previously discussed, it is clear that the time-depedent approch performs better than the time-independent counterpart in most of the scenarios, where imposing a lower bound time $T_{\min}$ becomes necessary.

### $\tau$ and run iterations with constrained lower bound time

In order to show that the lower bound time does indeed have such a great impact on the performance of the time-dependent approach relative to the time-independent one, we study the distribution of $\tau$ and $I$ with a constrain on $T_{\min}$ . It should be noted that in this particular scenario $\tau$ will no longer be a minimization over $T$ since the time is fixed, while the dependance on $\gamma$ remains.

The choice of constrain is arbitrary and somewhat biased since the larger the constrain the better the performance of the time-dependent approach, as We have just shown in Figure 2.8. Therefore, to make the choice fair, we consider the lower bound time to be $T_{\min} = \frac{\pi}{2}\sqrt{N}$ which is the same order of magnitude of the standard Grover's time scaling, the time-independent quantum walk search on the complete graph and the unstructured local adiabatic search. In the best-case scenario we might discover that the number of iterations necessary to get to unitary probability remains constant regardless of the dimension of the graph, making this approach have the same time scaling as the ones just mentioned; in the most probable scenario we discover that the number of iterations increases with the graph size, thus adding a scaling factor that depends on some power of $N$.

We begin by investigating the effects of the interpolating schedule $s(t)$ on the $\tau$ distribution. Figure 2.10 shows the results for cycle graphs $Cy(N)$ with $N$ up to 71, using the time-dependent Hamiltonian and the interpolating schedules defined in Section 2.1.2.

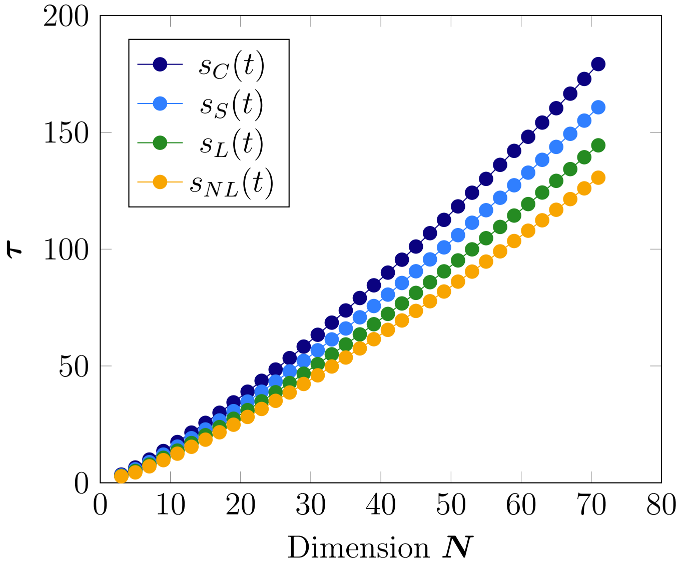


Figure 2.10: **$\tau$ with constrained time at $T = \pi/2\sqrt{N}$, for the different intepolating schedules.** The plot shows $\tau$ for cycle graphs up to $N = 71$, using the time-dependent Hamiltonian with different intepolating schedules: (orange) non-linear $s_{NL}(t)$, (green) linear $s_L(t)$, (light blue) $s_S(t)$ and (blue) $s_C(t)$. As expected $s_{NL}(t)$ is the best performing intepolating schedule, followed by the linear $s_L(t)$. The others do not show any advantage.

As we could have expected the non-linear interpolating schedule $s_{NL}(t)$ performs the best, compared to the linear one. The $s_S(t)$ and $s_C(t)$ show poor performance compared to all the others, therefore are ignored for the next analysis. This shows indeed that the shape of the interpolating function has a great impact on the performance of the algorithm, in particular when considering large graphs.

We now compare the time-independent approach to the time-dependent one using the linear and non-linear interpolating schedules. Again, the time is constrained as in the previous case. We begin by looking at the $\tau$.
In Figure 2.11 we can divide the plot into two sections. For small graph dimension $N < 20 - 25$ the time-dependent approach performs comparably to the time-independent one, although the latter has a slight advantage. On the other hand, for large graphs the time-dependent search distribution deviates considerably from the time-independent, for both the linear $s_L(t)$ and Roland-Cerf $s_{NL}(t)$ interpolating schedules. This is indeed a consequence of the lower bound time $T_{\text{min}}$ that we considered, begin dependent on the dimension of the graph and increasing with $N$. Nevertheless, for large $N$ the mamimum probability found for

the time-independent approach decreses, while for the time-dependent Hamiltonian it increases making it a succesfull alternative for larger graph dimensions.

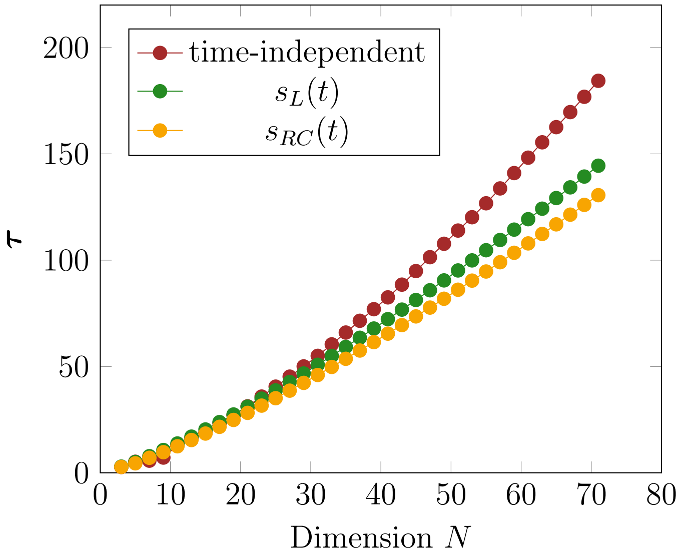


Figure 2.11: **$\tau$ with constrained time at $\pi/2\sqrt{N}$, for the time-dependent and time-independent approaches.** The plot shows $\tau$ for cycle graphs up to N=71, using the time-independent Hamiltonian (red) and time-dependent Hamiltonian with linear $s_L$ (green) and non-linear $s_{NL}$ (orange) interpolating schedules. We notice that for small graphs, up to $N \approx 20$, both approaches perform similarly. For large $N$ however the time-dependent algorithm performs significanly better.

We now turn our attention to the run iterations. Figure 2.12 shows the number of iterations $I$ for the two considered approaches. We are able to qualitatively describe the time scaling of the algorithms by superimposing the function $\frac{2}{\pi}\sqrt{N}$ (black line): if the plot of the iterations $I$ lies below the line the time-dependent algorithm performs better than the classical search. Remember in fact that we imposed a lower bound time $T_{\text{min}} = \pi/2\sqrt{N}$, and combining it with $2/\pi\sqrt{N}$ gives us the time scaling of the classical search algorithm $O(N)$.

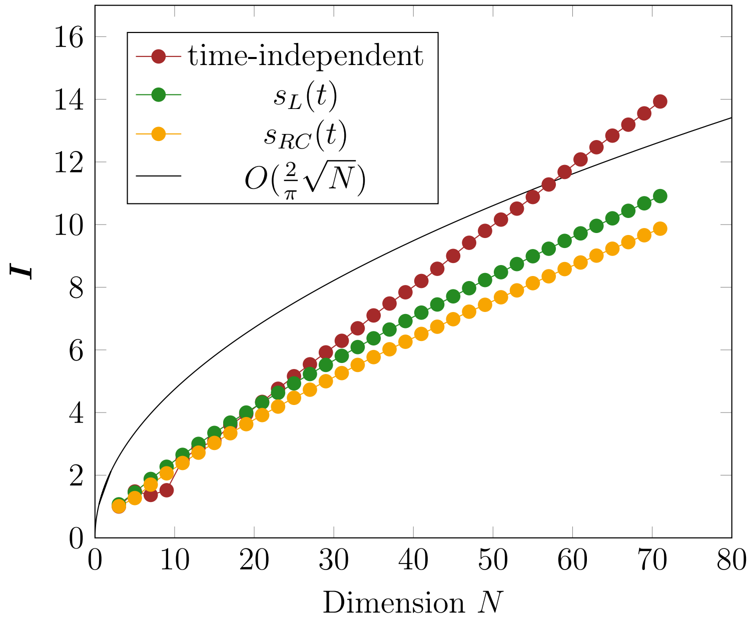


Figure 2.12: ***I* with constrained time at $\pi/2\sqrt{N}$, for the time-dependent and time-independent approaches.** The plot shows $I$ for cycle graphs up to $N = 71$, using the time-independent Hamiltonian (red) and time-dependent Hamiltonian with linear $s_L$ (green) and non-linear $s_{NL}$ (orange) interpolating schedules. The black line represents a time scaling of $\sqrt{N}$. Combining the value of $I$ with the constrained time of $\pi/2\sqrt{(N)}$ we can estimate the performance of the algorithms: if the distribution is below the black line the algorithm performs better than the classical search $O(N)$. From this plot we can clearly see that for $N$ up to 71 the performance is better than the classical search, although we are not able to make predictions for larger $N$.

Clearly this plot reflects what we have just seen for $\tau$ distribution, but it is able to give an immediate idea on the number of run iterations necessary to solve the search problem. For all the graph considered - with $N$ up to 59 - both the time-dependent and time-independent perform better than the classical search algorithm. However, for $N > 59$ and up to 71, the time-independent approach starts to perform worse than the classical counterpart, while the time-dependent still has a better performance. It it difficult however to predict with confidence the trend for larger graph. Nevertheless the time-dependent approach, in particular with the Roland-Cerf $s_{NL}(t)$ interpolating schedule shows promising results.

### 2.4.5 Comparison: robustness

We now address the robustness of the time-dependent and time-independent search. As we mentioned in Section 2.3.2 we are only interested in the comparison of the two approaches, and not an absolute measure of their robustness. Therefore we will use this measure as a comparison value.
We proceed by evaluating the $\gamma$-robustness considering small variations on the parameter $\gamma$. Recalling that we evaluated the probability on a grid, we consider the variations in the order of 2 *square*. We begin by finding the quantity $\tau$ with time constrains $T = \pi/2\sqrt{N}$. For the corresponding $(T, \gamma)$ combination we evaluate the robustness $R_\gamma$. For the time-dependent search with only consider the linear interpolating schedule and the Roland-Cerf $s_{NL}(t)$. As we showed in the previous section the Roland-Cerf Hamiltonian performs better than the linear conterpart, while from a qualitative point of view the linear Hamiltonian has a smoother probability distribution (see Figure 2.4(a)-(d)). The following plots show the $\gamma$-robustness for the time-independent and time-dependent search.

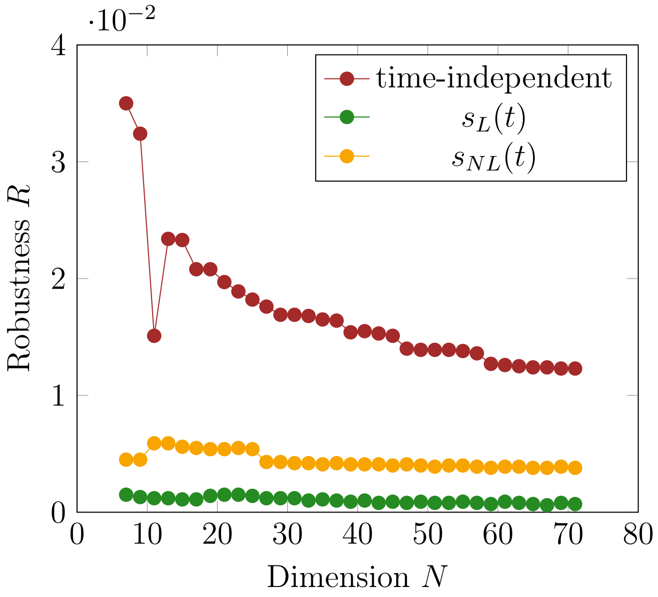


Figure 2.13: **$\gamma$-Robustness for the time-independent and time-dependent approaches.** The figure shows the $\gamma$-robustness for the time-independent approach (red), the time-dependent one with linear $s_L(t)$ (green) and non-linear $s_{NL}(t)$ (red) interpolating schedules. Recalling that the lower $R$ value the highest the robustness, this distribution reflects the probability seen in Section 2.4.2, where the probability distribution was smoother for $s_L(t)$ than $s_{NL}(t)$.

Indeed Figure 2.13 shows that the time-dependent approach is more robust than the time-independent one. In particular, the linear interpolating schedule

$s_L(t)$ leads to a greater robustness compared to the non-linear $s_{NL}(t)$. It is to be noted however that the difference in robustness between the time-independent and time-dependent approach is much greater than the difference between the interpolating schedules.

We now turn our attention the $T$-robustness, evaluated similarly to the $\gamma$-robustness. From Figure 2.14 we see that, in contrast with the $\gamma$ counterpart, the time-indipendent approach performs better compared to the time-dependent one. However, if we compare the $R$ values with the $R_\gamma$ values we notice that the difference between the two approaches is much smaller for the $T$-robustness. For this reason we can safely say that the search with time-dependent Hamiltonian is more robust than the time-independent one. Additionally, the improved performance acheived with the non-linear interpolating schedule $s_{NL}(t)$ comes at the cost of less robustness.

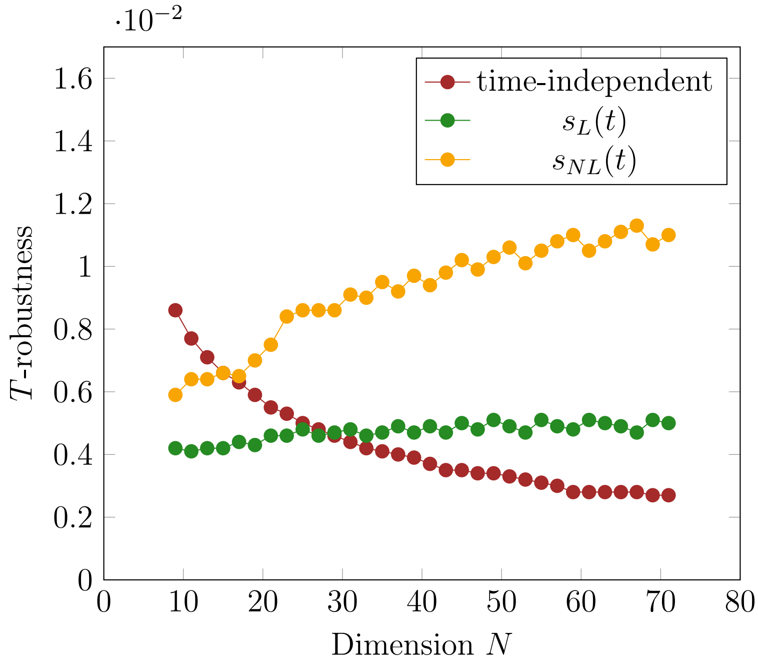


Figure 2.14: ***T*-Robustness for the time-independent and time-dependent approaches.** The figure shows the $T$-robustness for the time-independent approach (red), the time-dependent one with linear $s_L(t)$ (green) and non-linear $s_{NL}(t)$ (red) interpolating schedules. Surprisingly, the time-independent approach is more robust than the time-dependent one for large $N$. However it is to be noted that the difference in values is much smaller than the one obtained for the $\gamma$-robustness.

## 2.5. Results for the complete graph

We now turn our attention to the complete graph. As we discussed in the preliminaries in Section 1.4.1 and Section 1.5.3, with the complete graph we are able to solve the search problem using the standard time-independendent quantum walks Hamiltonian with time scaling of $O(\sqrt{N})$. Additionally for the unstructured search - which is equivalent to a search on the complete graph - we showed that with the local adiabatic evolution it is possible to get the same speedup of $O(\sqrt{N})$, while that was not the case for the global adiabatic evolution that had the same time scaling as the classical search. Although Wong proved that for the complete graph it is not possible to solve the search problem with an adiabatic quantum walk algorithm, for completeness we extend the time-dependent Hamiltonian implementation to the complete graph. Clearly we are not able to achieve any speedup nor necessarily any comparable time scaling, but we might get some interesting insights in terms of probability distribution and robustness.

### 2.5.1 Comments on the placement of $\gamma$

Firstly we recall that in Section 1.1.2 we introduced the search Hamiltonian as

$$H = \gamma L - |w\rangle\langle w| \tag{2.12}$$

We quickly notice that the $\gamma$ parameter is in front of the Laplacian, compared to the time-dependent Hamiltonian considered throughout our work where $\gamma$ was placed in front of the oracle $|w\rangle\langle w|$. We could now proceed by comparing the time-independent and time-dependent approaches with the $\gamma$ placement as in literature - i.e. in front of the Laplacian - or as we did in **??**. In order to be consistent with the standard quantum walks search on the complete graph discussed in Section 1.4.1 we proceed by considering the following Hamiltonians, with $\gamma$ in front of the Laplacian. This ensures that the time-dependent probability distribution is compatible with the time-independent one, and does not require to re-evaluate the optimal $\gamma$ for the time-independent approach. Nevertheless the two approaches can be considered equivalent. The Hamiltonians are given by

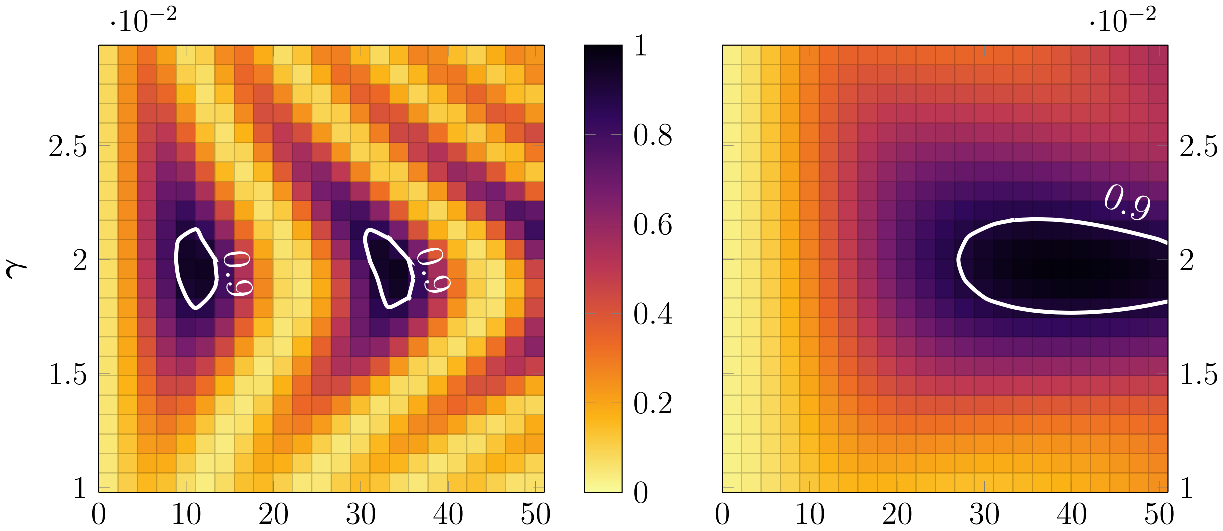
$$H = \gamma L - |w\rangle\langle w| \qquad \text{(independent)} \tag{2.13}$$

$$H(s) = (1-s)\gamma L - s|w\rangle\langle w| \qquad \text{(dependent)} \tag{2.14}$$

Having defined the Hamiltonians, we proceed as in the previous section, following however a more qualitative approach. In order to do so we compare the probability distribution using the heatmap plots introduced in ????.

### 2.5.2 Probability distribution and qualitative robustness

Recalling that for the time-independent quantum walks implementation the optimal $\gamma$ is given by $\gamma = 1/N$, we evaluate the probability for $\gamma$ in a neighbourhood of $1/N$ and up to $T = N$. This allows us to have a complete picture of the probability distribution, as can be seen in the following heatmap plots.



(a) time-independent

(b) $s_{NL}(t)$

Figure 2.15: **Probability distributions for the complete graph $C(51)$ with time-dependent and time-independent Hamiltonians.** The two figures show the probability distribution for the complete graph $C(51)$ with the time-independent Hamiltonian (a) and time-dependent Hamiltonian (b) using the non-linear $s_{NL}(t)$ interpolating schedule. The white contour lines indicate a region with probability $p > 0.9$.

As previously discussed we are able to solve the search problem with the time-independent algorithm in a time of the order of $T = \pi/2\sqrt{N}$. Indeed we see from the figure above that the probability is close to $p = 1$ for $T = 11$, as expected. On the other hand the time-dependent approach is able to solve the search with a single iteration for $T \to N$, showing no significant speedup compared to the classical search. This however leaves space for a multiple run search as introduced in Section 2.1.3. We therefore investigate this possibility by evaluating the maximum probability found for $T = \pi/2\sqrt{N}$ with the time-dependent algorithm and study the **iterations** distribution.

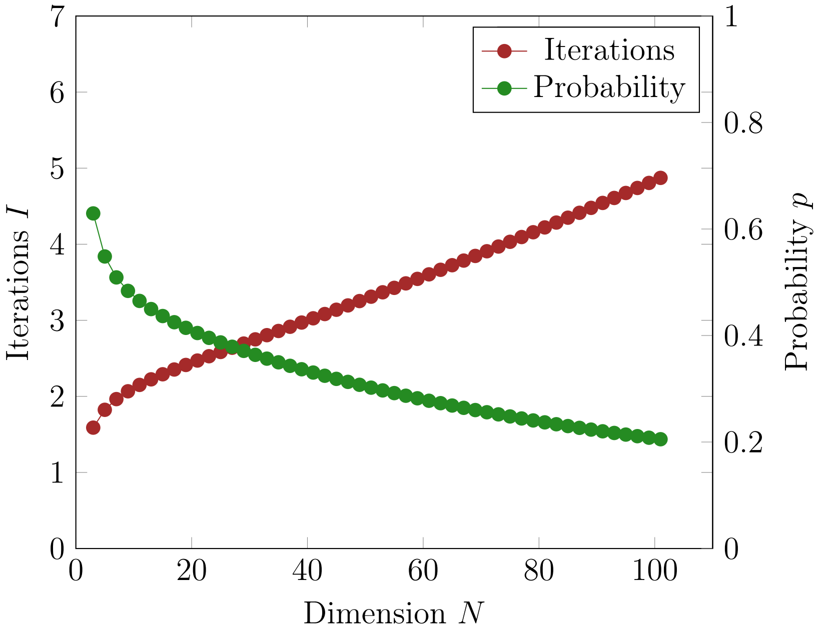
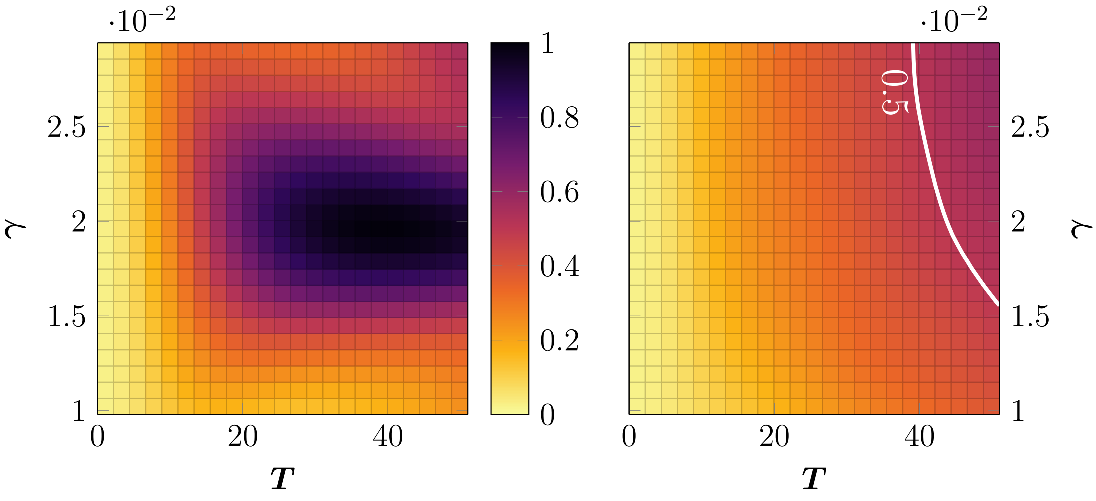


Figure 2.16: **Iterations $I$ and probability $p$ for the multiple run search on complete graph.** The plot shows the probability $p$ and the number of iterations $I$ for the multiple run search with the time-dependent Hamiltonian. The time is constrained to $\pi/2\sqrt{N}$ as in the solution of the standard quantum walks search. It is clear that the number of iterations increases linearly, although very slowly. Notice in fact that $N$ goes all the way up to $N = 101$. For the limit of large $N$ the time-dependent approach is therefore not a valid alternative, nor of comparable performance.

It is clear from the plot that the number of run iterations increases with the graph size linearly, although very slowly (notice that $N$ goes up to 101). In the limit for large $N$ the multiple runs search does not improve the time-dependent approach, therefore the standard quantum walks search remains the strongest option.

However, we can now address the **robustness** properties of the two approaches. Similarly to the cycle graph, the complete graph has a probability distribution with regions of high and low probability when considering the time-independent Hamiltonian. Compared to the smooth probability distribution of the time-dependent algorithm we can safely say that the latter is more robust than the first, both for noise on $T$ and $\gamma$. It is to be noted however that for the cycle graph the time-independent and time-dependent approaches are of comparable performance, therefore the robustness has a great impact to the overall performance. In this scenario the time-independent approach performs better in terms of probability, and having less robustness does not justify the *almost* linear time-scaling of the time-dependent one.

Although the time-dependent algorithm is not able to perform as well as the time-independent one, striking is the difference in probability distribution between the non-linear $s_{NL}(t)$ and the linear $s_L(t)$, showing the importance of the shape of the interpolating schedule. From the following plots we can see that the time-dependent Hamiltonian with the linear interpolating schedule $s_L(t)$ is not able to solve the search for $T = N$. The improvement on the performance is therefore achieved by findind the optimal interpolating schedule.



(a) $s_{NL}(t)$

(b) $s_L(t)$

Figure 2.17: **Probability distributions for the complete graph $C(51)$ with the time-dependent Hamiltonian.** The figure shows the probability distribution for a complete graph of $N = 51$ using the time-dependent Hamiltonian with the non-linear $s_{NL}(t)$ (a) and linear $s_L(t)$ (right) interpolating schedules. It illustrates the great impact of the interpolating schedule on the overall performance of the time-dependent algorithm, where the choice of $s(t)$ is critical.

# Conclusions

We conclude our work with a summary of what has been accomplished in this thesis, some thoughts on the results and future perspective.

In Chapter 1 we reviewed some basic notion of graph theory, quantum walks, and the main characteristics of the graph topologies considered. We then introduced the search problem as originally posed by Grover - with particular enphasis on the action of the *oracle* - followed by its quantum walks implementation of Farhi and Gutmann on the complete graph. We then discussed the adiabatic theorem and its application to the search problem, focusing on the difference between the global and local adiabatic evolution. To set the basis for our work we presented the work by Wong et. al [13], where they show that an adiabatic-quantum walk implementation of the search algorithm is not possible with the structure of the standard Grover's oracle.

In Chapter 2 we introduced the main topic of our work which is a quantum walks search algorithm with time-dependent Hamiltonians, *inspired* by the adiabatic implementation but free of the constrains of the adiabatic theorem. We introduced a few classes of interpolating schedules with the goal of improving the standard linear one of Farhi and Gutmann. We draw from the one derived by Roland and Cerf, and consider the following non-linear interpolating schedule:

$$s_{NL}(t) = \frac{1}{2}\Big[\big(2\frac{t}{T} - 1\big)^3 + 1\Big].$$

Additionally we also consider the possibility of repeating the search multiple times if the search is not perfect - that made us take into account an initialization and measure time. In order to compare the time-independent and time-dependent approach we introduced three classes of results: the search, the localization and the robustness.

We then turned our attention to the cycle graph, the main graph topology considered. We studied the localization, search and robustness.

In terms of localization we discovered that the probability evaluated with the time-independent Hamiltonian does not increase with time, therefore it does not show localization properties. On the other hand, the time-dependent approach - given that it is based on the adiabatic implementation - for large $T$ (far larger than the classical $O(N)$ search) it is able to achieve unitary probability. More interestingly, since the probability does not increase linearly, the solution of the search can be found with probability in the order of $p = 0.8 \div 0.9$ in much less time.

We then studied the search performance of the two approaches in terms of multiple runs search. Therefore we introduced a new quantity which represents the minimum time necessary to get to unitary probability:

$$\tau = \min\left(\frac{T}{p}\right)_{T,\gamma}.$$

We also consider the number of run iterations $I = \min(p^{-1})_{T,\gamma}$. Additionally we discover that $\tau$ requires to consider a minimum time $T_{\min}$ to be effective at comparing the two approaches. To be consistent with the standar Grover's and the quantum walks search we set $T_{\min} = \pi/2\sqrt{N}$.

After seeing that for the time-dependent approach the linear and non-linear interpolating schedules $s_L$ and $s_{NL}$ perform the best, we compare them to the time-independent approach.
We discovered that both approaches have similar performance up to $N \approx 25$, while for large $N$ it gets significanly different, with the $s_{NL}$ performing the best and the time-independent approach the worst. In terms of run interations $I$ we see that the time-dependent approach, regardless of the interpolating schedules, performs better than the classical search, but still much worse than the optimal time scaling of $O(\sqrt{N})$. Although promising, this result is limited to the dimension considered - up to $N = 71$ in our analysis - and therefore cannot be generalized for large $N$.

We then studied the robustness for both time and $\gamma$. The time-independent approach has a very discontinous probability, made of scattered regions of high and low probability, leading to being less $\gamma$-robust than the time-dependent approach. Indeed the latter has a smooth probability distribution regardless of the interpolating schedule considered. Nevertheless the linear $s_L$ leads to more robust results than $s_{NL}$. In terms of $T$-robustness the time-independent approach is surprisingly more robust than the others, although the difference in robustness is much smaller than the one encountered for the $\gamma$-robustness. Therefore we can safely say that the time-dependent Hamiltonian-based algorithm is more robust than the time-independent one.

For completeness we at last turned our attention to the complete graph, qualitatively comparing the probability distribution and the robustness. As expected the time-dependent approach is not able to achieve comparable performance with the standard time-independent algorithm. In terms of qualitative robustness we see that the time-dependent algorithm has a smoother probability distribution and therefore better robustness. However the improvement in robustness does not justify the much worse time scaling.

The complete graph however illustrates the importance of the interpolating schedule. In particular, with the time-dependent approach and linear $s_L$ the maximum probability reached is $p = 0.5$ in $T = N$ for a $C(51)$. With the improved non-linear interpolating schedule $s_{NL}$ we are able to achieve in $T = N$ a maximum probability of $p > 0.9$.

Although it is not able to achieve the same time scaling as the time-independent approach it suggests that the improvement on performance comes from the choice of the optimal interpolating schedule. Future investigation on the interpolating schedule can lead to improvements on performance, in particular for the cycle graph. Additionally one could also apply the time-dependent Hamiltonian to other graph topologies.

# Bibliography

[1] G. Brassard, P. Høyer, and A. Tapp. Quantum Algorithm for the Collision Problem. *Encyclopedia of Algorithms*, pages 1–2, 2015.

[2] A. M. Childs, E. Farhi, and S. Gutmann. An example of the difference between quantum and classical random walks. (2):1–4, 2001.

[3] A. M. Childs and J. Goldstone. Spatial search by quantum walk. *Physical Review A - Atomic, Molecular, and Optical Physics*, 70(2), 2004.

[4] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser. Quantum Computation by Adiabatic Evolution. 2000.

[5] E. Farhi and S. Gutmann. Analog analogue of a digital quantum computation. *Physical Review A - Atomic, Molecular, and Optical Physics*, 57(4):2403–2406, 1998.

[6] M. Fürer. Solving np-complete problems with quantum search. In E. S. Laber, C. Bornstein, L. T. Nogueira, and L. Faria, editors, *LATIN 2008: Theoretical Informatics*, pages 784–792, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[7] L. K. Grover. Quantum mechanics helps in searching for a needle in a haystack. *Physical Review Letters*, 79(2):325–328, 1997.

[8] J. G. Morley, N. Chancellor, S. Bose, and V. Kendon. Quantum search with hybrid adiabatic-quantum walk algorithms and realistic noise. pages 1–24, 2018.

[9] O. Mülken and A. Blumen. Continuous-time quantum walks: Models for coherent transport on complex networks. *Physics Reports*, 502(2-3):37–87, 2011.

[10] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambrigde University Press, 2000.

[11] J. Roland and N. J. Cerf. Quantum search by local adiabatic evolution. *Physical Review A - Atomic, Molecular, and Optical Physics*, 65(4):6, 2002.

[12] S. Z. SH. Hung, S. Hietala. Quantitative Robustness Analysis of Quantum Programs (Extended Version). *Proceedings of the ACM on Programming Languages*, 3(January), 2019.

[13] T. G. Wong and D. A. Meyer. Irreconcilable difference between quantum walks and adiabatic quantum computing. *Physical Review A*, 93(6):1–8, 2016.