D.4 Search problems

D.4.a Overview

Many problems can be formulated as search problems over a solution space $\mathcal{S}$.[17] That is, find the $x \in \mathcal{S}$ such that some predicate $P(x)$ is true. For example, hard problems such as the Hamiltonian path problem and Boolean satisfiability can be formulated this way. An *unstructured search problem* is a problem that makes no assumptions about the *structure* of the search space, or for which there is no known way to make use of it (also called a *needle in a haystack problem*). That is, information about a particular value $P(x_0)$ does not give us usable information about another value $P(x_1)$. In contrast, a *structured search problem* is a problem in which the structure of the solution space can be used to guide the search, for example, searching an alphabetized array. In general, unstructured search takes $\mathcal{O}(M)$ evaluations, where $M = |\mathcal{S}|$ is the size of the solution space (which is often exponential in the size of the problem). On the average it will be $M/2$ (think of searching an unordered array) to find a solution with 50% probability.

We will see that Grover's algorithm can do unstructured search on a quantum computer with bounded probability in $\mathcal{O}(\sqrt{M})$ time, that is, quadratic speedup. This is provably more efficient than any algorithm on a classical computer, which is good (but not great). Unfortunately, it has been proved that Grover's algorithm is optimal for unstructured search. Therefore, to do better requires exploiting the structure of the solution space. *Quantum computers do not exempt us from understanding the problems we are trying to solve!* Shor's algorithm is an excellent example of exploiting the structure of a problem domain. Later we will take a look at *heuristic quantum search algorithms* that do make use of problem structure.

D.4.b Grover's Algorithm

algorithm Grover:

Input: Let M be the size of the solution space and pick n such that $2^n \geq M$.

[17] This section is based primarily on Rieffel & Polak (2000).

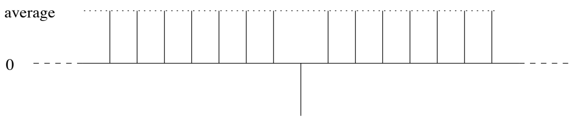

Figure III.28: Depiction of the result of phase rotation (changing the sign) of solutions in Grover's algorithm. [source: Rieffel & Polak (2000)]

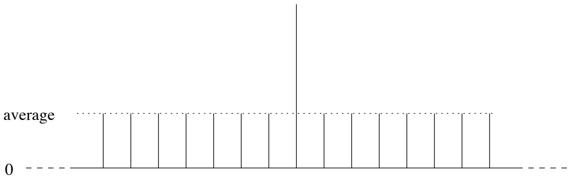

Figure III.29: Depiction of result of inversion about the mean in Grover's algorithm. [source: Rieffel & Polak (2000)]

Let $N \stackrel{\text{def}}{=} 2^n$ and let $\mathbf{N} \stackrel{\text{def}}{=} \mathbf{2}^n = \{0, 1, \ldots, N-1\}$, the set of n-bit strings. We are given a computable predicate $P : \mathbf{N} \to \mathbf{2}$. Suppose we have a quantum gate array U_P (an *oracle*) that computes the predicate:

$$U_P|x, y\rangle = |x, y \oplus P(x)\rangle.$$

Application: Consider what happens if, as usual, we apply the function to a superposition of all possible inputs $|\psi_0\rangle$:

$$U_P|\psi_0\rangle|0\rangle = U_P\left[\frac{1}{\sqrt{N}}\sum_{x\in\mathbf{N}}|x, 0\rangle\right] = \frac{1}{\sqrt{N}}\sum_{x\in\mathbf{N}}|x, P(x)\rangle.$$

Notice that the components we want, $|x, 1\rangle$, and the components we don't want, $|x, 0\rangle$, all have the same amplitude, $\frac{1}{\sqrt{N}}$. So if we measure the state, the chances of getting a hit are very small, $\mathcal{O}(2^{-n})$. The trick, therefore, is to amplify the components that we want at the expense of the ones we don't want; this is what Grover's algorithm accomplishes.

Sign-change: To do this, first we change the sign of every solution (a phase rotation of π). That is, if the state is $\sum_j a_j|x_j, P(x_j)\rangle$, then we want to change a_j to $-a_j$ whenever $P(x_j) = 1$. See Fig. III.28. I'll get to the technique in a moment.

Inversion about mean: Next, we invert all the components around their mean amplitude (which is a little less than the amplitudes of the non-solutions); the result is shown in Fig. III.29. As a result of this operation, amplitudes of non-solutions go from a little above the mean to a little below it, but amplitudes of solutions go from far below the mean to far above it. This amplifies the solutions.

Iteration: This *Grover iteration* (the sign change and inversion about the mean) is repeated $\frac{\pi\sqrt{N}}{4}$ times. Thus the algorithm is $\mathcal{O}(\sqrt{N})$.

Measurement: Measurement yields an x_0 for which $P(x_0) = 1$ with high probability. Specifically, if there is exactly one solution $x_0 \in \mathcal{S}$, then $\frac{\pi\sqrt{N}}{8}$ iterations will yield it with probability $1/2$. With $\frac{\pi\sqrt{N}}{4}$ iterations, the probability of failure drops to $1/N = 2^{-n}$. Unlike with most classical algorithms, additional iterations will give a *worse* result! This is because Grover iterations are unitary rotations, and so excessive rotations can rotate past the solution. Therefore it is critical to know when to stop. Fortunately there is a quantum technique (Brassard et al. 1998) for determining the optimal stopping point. Grover's iteration can be used for a wide variety of problems as a part of other quantum algorithms.
□

In the following geometric analysis, I will suppose that there is just one answer α such that $P(\alpha) = 1$; then $|\alpha\rangle$ is the desired answer vector. Let $|\omega\rangle$ be a uniform superposition of all the other (non-answer) states, and observe that $|\alpha\rangle$ and $|\omega\rangle$ are orthonormal. Therefore, initially the state is $|\psi_0\rangle = \frac{1}{\sqrt{N}}|\alpha\rangle + \sqrt{\frac{N-1}{N}}|\omega\rangle$. In general, after k iterations the state is $|\psi_k\rangle =$

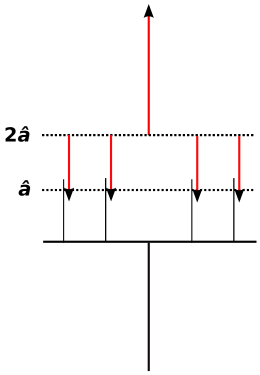

Figure III.30: Process of inversion about the mean in Grover's algorithm. The black lines represent the original amplitudes a_j. The red lines represent $2\bar{a} - a_j$, with the arrow heads indicating the new amplitudes a'_j.

$a|\alpha\rangle + w|\omega\rangle$, for some a, w with $|a|^2 + |w|^2 = 1$.

The sign change operation transforms the state as follows:

$$|\psi_k\rangle = a|\alpha\rangle + w|\omega\rangle \quad \mapsto \quad -a|\alpha\rangle + w|\omega\rangle = |\psi'_k\rangle,$$

where I've called the result $|\psi'_k\rangle$. This is a reflection across the $|\omega\rangle$ vector, which means that it will be useful to look at reflections more generally.

Suppose that $|\phi\rangle$ and $|\phi^\perp\rangle$ are orthonormal vectors and that $|\psi\rangle = a|\phi^\perp\rangle + b|\phi\rangle$ is an arbitrary vector in the space they span. The reflection of $|\psi\rangle$ across $|\phi\rangle$ is $|\psi'\rangle = -a|\phi^\perp\rangle + b|\phi\rangle$. Since $a = \langle\phi^\perp \mid \psi\rangle$ and $b = \langle\phi \mid \psi\rangle$, we know $|\psi\rangle = |\phi\rangle\langle\phi \mid \psi\rangle + |\phi^\perp\rangle\langle\phi^\perp \mid \psi\rangle$, you can see that $|\psi'\rangle = |\phi\rangle\langle\phi \mid \psi\rangle - |\phi^\perp\rangle\langle\phi^\perp \mid \psi\rangle$. Hence the operator to reflect across $|\phi\rangle$ is $R_\phi \overset{\text{def}}{=} |\phi\rangle\langle\phi| - |\phi^\perp\rangle\langle\phi^\perp|$. Alternate forms of this operator are $2|\phi\rangle\langle\phi| - I$ and $I - 2|\phi^\perp\rangle\langle\phi^\perp|$, that is, subtract twice the perpendicular component.

The sign change can be expressed as a reflection:

$$R_\omega = |\omega\rangle\langle\omega| - |\alpha\rangle\langle\alpha| = I - 2|\alpha\rangle\langle\alpha|,$$

which expresses the sign-change of the answer vector clearly. Of course we don't know $|\alpha\rangle$, which is why we will have to use a different process to accomplish this reflection (see p. 150). We also will see that the inversion about the mean is equivalent to reflecting that state vector across $|\psi_0\rangle$.

But first, taking this for granted, let's see the effect of the Grover iteration. Let θ be the angle between $|\psi_0\rangle$ and $|\omega\rangle$. It's given by the inner product

$\cos\theta = \langle\psi_0 \mid \omega\rangle = \sqrt{\frac{N-1}{N}}$. Therefore the sign change reflects $|\psi_0\rangle$ from θ above $|\omega\rangle$ into $|\psi_0'\rangle$, which is θ below it. Inversion about the mean reflects $|\psi_0'\rangle$ from 2θ below $|\psi_0\rangle$ into a state we call $|\psi_1\rangle$, which is 2θ above it. Therefore, in going from $|\psi_0\rangle$ to $|\psi_1\rangle$ the state vector has rotated 2θ closer to $|\alpha\rangle$.

You can see that after k iterations, the state vector $|\psi_k\rangle$ will be $(2k+1)\theta$ above $|\omega\rangle$. We can solve $(2k+1)\theta = \pi/2$ to get the required number of iterations to bring $|\psi_k\rangle$ to $|\alpha\rangle$. Note that for small θ, $\theta \approx \sin\theta = \frac{1}{\sqrt{N}}$ (which is certainly small). Hence, we want $(2k+1)/\sqrt{N} \approx \pi/2$, or $2k+1 \approx \pi\sqrt{N}/2$. That is, $k \approx \pi\sqrt{N}/4$ is the required number of iterations. Note that after $\pi\sqrt{N}/8$ iterations, we are about halfway there (i.e., $\pi/4$), so the probability of success is 50%. In general, the probability of success is about $\sin^2\frac{2k+1}{\sqrt{N}}$.

Now for the techniques for changing the sign and inversion about the mean. Let $|\psi_k\rangle$ be the state after k iterations ($k \geq 0$). To change the sign, simply apply U_P to $|\psi_k\rangle|-\rangle$. To see the result, let $X_0 = \{x \mid P(x) = 0\}$ and $X_1 = \{x \mid P(x) = 1\}$, the solution set. Then:

$$\begin{aligned}
& U_P|\psi_k\rangle|-\rangle \\
&= U_P\left[\sum_{x\in\mathbf{N}} a_x|x,-\rangle\right] \\
&= U_P\left[\frac{1}{\sqrt{2}}\sum_{x\in\mathbf{N}} a_x|x,0\rangle - a_x|x,1\rangle\right] \\
&= \frac{1}{\sqrt{2}}\,U_P\left[\sum_{x\in X_0} a_x|x,0\rangle + \sum_{x\in X_1} a_x|x,0\rangle - \sum_{x\in X_0} a_x|x,1\rangle - \sum_{x\in X_1} a_x|x,1\rangle\right] \\
&= \frac{1}{\sqrt{2}}\left[\sum_{x\in X_0} a_x U_P|x,0\rangle + \sum_{x\in X_1} a_x U_P|x,0\rangle\right. \\
&\qquad \left. - \sum_{x\in X_0} a_x U_P|x,1\rangle - \sum_{x\in X_1} a_x U_P|x,1\rangle\right] \\
&= \frac{1}{\sqrt{2}}\left[\sum_{x\in X_0} a_x|x,0\rangle + \sum_{x\in X_1} a_x|x,1\rangle\right. \\
&\qquad \left. - \sum_{x\in X_0} a_x|x,1\oplus 0\rangle - \sum_{x\in X_1} a_x|x,1\oplus 1\rangle\right]
\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{\sqrt{2}}\left[\sum_{x\in X_0} a_x|x\rangle|0\rangle + \sum_{x\in X_1} a_x|x\rangle|1\rangle - \sum_{x\in X_0} a_x|x\rangle|1\rangle - \sum_{x\in X_1} a_x|x\rangle|0\rangle\right] \\
&= \frac{1}{\sqrt{2}}\left(\sum_{x\in X_0} a_x|x\rangle - \sum_{x\in X_1} a_x|x\rangle\right)(|0\rangle - |1\rangle) \\
&= \left(\sum_{x\in X_0} a_x|x\rangle - \sum_{x\in X_1} a_x|x\rangle\right)|-\rangle.
\end{aligned}
$$

Therefore the signs of the solutions have been reversed (they have been rotated by π). Notice how $|-\rangle$ in the target register can be used to separate the 0 and 1 results by rotation. This is a useful idea!

It remains to show the connection between inversion about the mean and reflection across $|\psi_0\rangle$. This reflection is given by $R_{\psi_0} = 2|\psi_0\rangle\langle\psi_0| - I$. Note that:

$$
|\psi_0\rangle\langle\psi_0| = \left(\frac{1}{\sqrt{N}}\sum_{\mathbf{x}\in\mathbf{N}}|\mathbf{x}\rangle\right)\left(\frac{1}{\sqrt{N}}\sum_{\mathbf{y}\in\mathbf{N}}\langle\mathbf{y}|\right) = \frac{1}{N}\sum_{\mathbf{x}\in\mathbf{N}}\sum_{\mathbf{y}\in\mathbf{N}}|\mathbf{x}\rangle\langle\mathbf{y}|.
$$

This is the *diffusion matrix*:

$$
\begin{pmatrix}
\frac{1}{N} & \frac{1}{N} & \cdots & \frac{1}{N} \\
\frac{1}{N} & \frac{1}{N} & \cdots & \frac{1}{N} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{1}{N} & \frac{1}{N} & \cdots & \frac{1}{N}
\end{pmatrix},
$$

which, as we will see, does the averaging.

To perform inversion about the mean, let $\bar{a}$ be the average of the a_j (see Fig. III.30). Inversion about the mean is accomplished by the transformation:

$$
\sum_{j\in\mathbf{N}} a_j|x_j\rangle \mapsto \sum_{j\in\mathbf{N}}(2\bar{a} - a_j)|x_j\rangle.
$$

To see this, write $a_j = \bar{a} \pm \delta_j$, that is, as a difference from the mean. Then $2\bar{a} - a_j = 2\bar{a} - (\bar{a} \pm \delta_j) = \bar{a} \mp \delta_j$. Therefore an amplitude δ_j below the mean will be transformed to δ_j above, and vice verse. But an amplitude that is negative, and thus very far below the mean, will be transformed to an amplitude much above the mean. This is exactly what we want to amplify the negative components, which correspond to solutions.

Inversion about the mean is accomplished by a "Grover diffusion transformation" D. To derive the matrix D, consider the new amplitude a'_j as a function of all the others:

$$a'_j \stackrel{\text{def}}{=} 2\bar{a} - a_j = 2\left(\frac{1}{N}\sum_{k=0}^{N-1} a_k\right) - a_j = \sum_{k\neq j}\frac{2}{N}a_k + \left(\frac{2}{N}-1\right)a_j.$$

This matrix has $\frac{2}{N}-1$ on the main diagonal and $\frac{2}{N}$ in the off-diagonal elements:

$$D = \begin{pmatrix} \frac{2}{N}-1 & \frac{2}{N} & \cdots & \frac{2}{N} \\ \frac{2}{N} & \frac{2}{N}-1 & \cdots & \frac{2}{N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{2}{N} & \frac{2}{N} & \cdots & \frac{2}{N}-1 \end{pmatrix}.$$

Note that $D = 2|\psi_0\rangle\langle\psi_0| - I = R_{\psi_0}$. It is easy to confirm that $DD^\dagger = I$ (Exer. III.45), so the matrix is unitary and therefore a possible quantum operation, but it remains to be seen if it can be implemented efficiently.

We claim $D = WRW$, where $W = H^{\otimes n}$ is the n-qubit Walsh-Hadamard transform and R is the *phase rotation matrix*:

$$R \stackrel{\text{def}}{=} \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -1 \end{pmatrix}.$$

To see this, let

$$R' \stackrel{\text{def}}{=} R + I = \begin{pmatrix} 2 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix}.$$

Then $WRW = W(R' - I)W = WR'W - WW = WR'W - I$. It is easy to show (Exer. III.46) that:

$$WR'W = \begin{pmatrix} \frac{2}{N} & \frac{2}{N} & \cdots & \frac{2}{N} \\ \frac{2}{N} & \frac{2}{N} & \cdots & \frac{2}{N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{2}{N} & \frac{2}{N} & \cdots & \frac{2}{N} \end{pmatrix}.$$

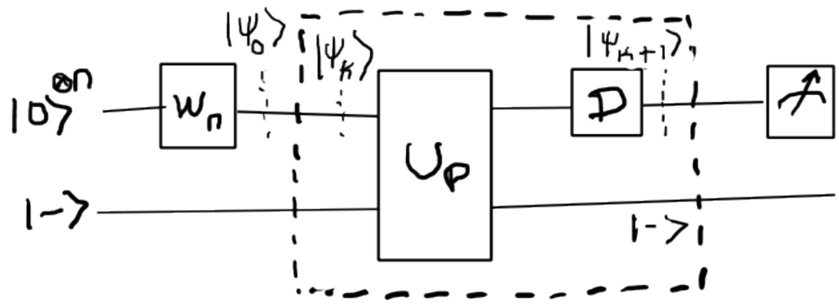

Figure III.31: Circuit for Grover's algorithm. The Grover iteration in the dashed box is repeated $\frac{\pi\sqrt{N}}{4}$ times.

It therefore follows that $D = WR'W - I = WRW$. See Fig. III.31 for a diagram of Grover's algorithm.

It remains to consider the possibility that there may be several solutions to the problem. If there are s solutions, then run the Grover iteration $\frac{\pi\sqrt{N/s}}{4}$ times, which is optimal. It can be done in $\sqrt{N/s}$ iterations even if s is unknown.

D.4.c Hogg's heuristic search algorithms

Many important problems can be formulated as *constraint satisfaction problems*, in which we try to find a set of assignments to variables that satisfy specified *constraints*. More specifically let $V \overset{\text{def}}{=} \{v_1, \ldots, v_n\}$ be a set of variables, and let $X \overset{\text{def}}{=} \{x_1, \ldots, x_n\}$ be a set of values that can be assigned to the variables, and let $C_1, \ldots, C_l$ be the constraints. The set of all possible assignments of values to variables is $V \times X$. Subsets of this set correspond to full or partial assignments, including inconsistent assignments. The set of all such assignments is $\mathcal{P}(V \times X)$.

The sets of assignments form a *lattice* under the $\subseteq$ partial order (Fig. III.32). By assigning bits to the elements of $V \times X$, elements of $\mathcal{P}(V \times X)$ can be represented by mn-element bit strings (i.e., integers in the set $\mathbf{MN} = \{0, \ldots, 2^{mn} - 1\}$); see Fig. III.33. Hogg's algorithms are based on the observation that if an assignment violates the constraints, then so do all those assignments above it in the lattice.

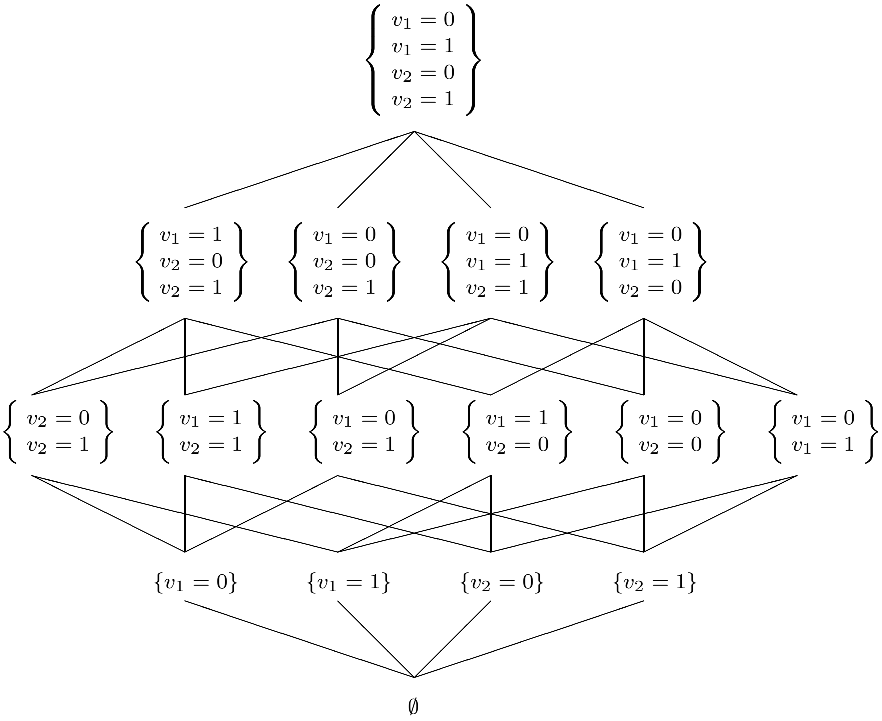

Figure III.32: Lattice of variable assignments. [source: Rieffel & Polak (2000)]

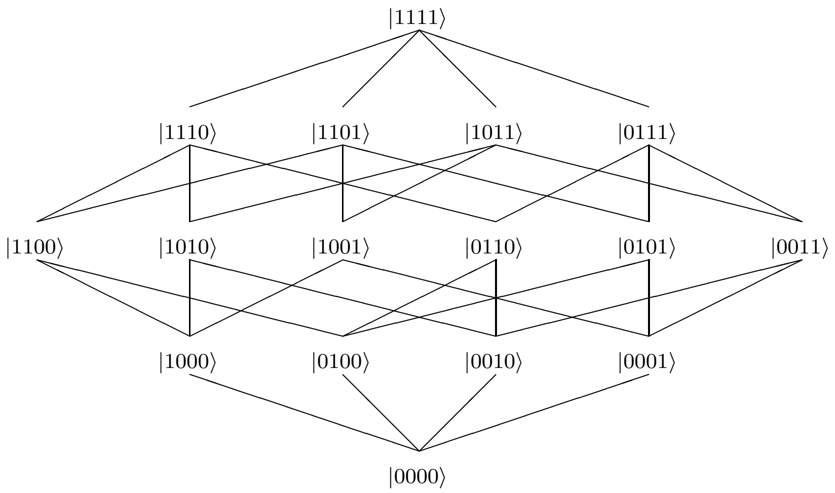

Figure III.33: Lattice of binary strings corresponding to all subsets of a 4-element set. [source: Rieffel & Polak (2000)]

algorithm Hogg:

Initialization: The algorithm begins with all the amplitude concentrated in the bottom of the lattice, $|0 \cdots 0\rangle$ (i.e., the empty set of assignments).

Movement: The algorithm proceeds by moving amplitude up the lattice, while avoiding assignments that violate the constraints; that is, we want to move amplitude from a set to its supersets. For example, we want to redistribute the amplitude from $|1010\rangle$ to $|1110\rangle$ and $|1011\rangle$. Hogg has developed several methods. One method is based on the assumption that the transformation has the form WDW, where $W = H^{\otimes mn}$, the mn-dimensional Walsh-Hadamard transformation, and D is diagonal. The elements of D depend on the size of the sets. Recall (D.1.b, p. 129) that

$$W|x\rangle = \frac{1}{\sqrt{2^{mn}}} \sum_{z \in \mathbf{MN}} (-)^{x \cdot z} |z\rangle.$$

As shown in Sec. A.2.c (p. 70), we can derive a matrix representation for W:

$$\begin{aligned} W_{jk} &= \langle j \mid W \mid k \rangle \\ &= \langle j | \frac{1}{\sqrt{2^{mn}}} \sum_{z \in \mathbf{MN}} (-)^{k \cdot z} | z \rangle \\ &= \frac{1}{\sqrt{2^{mn}}} \sum_{z \in \mathbf{MN}} (-)^{k \cdot z} \langle j \mid z \rangle \\ &= \frac{1}{\sqrt{2^{mn}}} (-1)^{k \cdot j}. \end{aligned}$$

Note that $k \cdot j = |k \cap j|$, where on the right-hand side we interpret the bit strings as sets.
□

The general approach is to try to steer amplitude away from sets that violate the constraints, but the best technique depends on the particular problem. One technique is to invert the phase on bad subsets so that they tend to cancel the contribution of good subsets to supersets. This could be done by a process like Grover's algorithm using a predicate that tests for violation of constraints. Another approach is to assign random phases to bad sets.

It is difficult to analyze the probability that an iteration of a heuristic algorithm will produce a solution, and so its efficiency is usually evaluated empirically, but empirical tests will be difficult to apply to quantum heuristic search until larger quantum computers are available, since classical computers require exponential time to simulate quantum systems. Small simulations, however, indicate that Hogg's algorithms may provide polynomial speedup over Grover's algorithm.