Implementing Pure Adaptive Search with Grover's Quantum Algorithm

*D. Bulger†, W. P. Baritompa‡, G. R. Wood§

September 21, 2000

Abstract

Pure Adaptive Search (PAS) is an idealised stochastic algorithm for unconstrained global optimisation. The number of PAS iterations required to solve a problem increases only linearly in the domain dimension. However, each iteration requires the generation of a random domain point uniformly distributed in the current improving region. If no regularity conditions are known to hold for the objective function, then this task requires a number of 'classical' function evaluations varying inversely with the proportion of the domain constituted by the improving region, entirely counteracting PAS's apparent speed-up. Grover's quantum computational search algorithm provides a way to generate the PAS iterates. We show that GAS realizes Pure Adaptive Search for functions satisfying certain conditions, and (when quantum computers are available) will be a practical algorithm.

Key words: Discrete optimization, Global optimization, Grover iterations, Markov chains, Quantum computers, Random search
Abbreviated Title: Grover Adaptive Search
Report Number: UCDMS2000/10 June 2000

*The authors would like to thank the Marsden Fund of the Royal Society of New Zealand for support of this research.

†College of Sciences, Massey University, Wellington, New Zealand. d.bulger@massey.ac.nz.

‡Department of Mathematics and Statistics, University of Canterbury, Christchurch, New Zealand.

§Institute of Information Sciences and Technology, Massey University, Palmerston North, New Zealand.

1 Introduction

Pure Adaptive Search (PAS) is a theoretical stochastic global optimisation algorithm. If PAS could be implemented efficiently, then under certain conditions the number of PAS iterations required to solve a problem would increase only linearly in the dimension of the domain [10, 11]. The algorithm specifies, though, that each iteration requires the generation of a random domain point uniformly distributed in the current *improving region*, that is, the subset of the domain on which the objective function takes better values than any yet seen. If no regularity conditions are known to hold for the objective function, then the generation of this random point in the improving region requires a number of 'classical' function evaluations varying inversely with the proportion of the domain constituted by the improving region, annulling completely PAS's apparent exponential efficiency gain. However, Grover's quantum computational search algorithm provides a way to generate the PAS iterates. We show that GAS realizes Pure Adaptive Search for functions satisfying certain conditions, and (when quantum computers become available) will be a practical algorithm.

This article merges known results from the theories of global optimisation and quantum computation. Section 2 describes Pure Adaptive Search, from the literature of stochastic global optimisation theory. Section 3 briefly introduces quantum computation in general, and Section 4 describes Grover's search method, an algorithm for execution on quantum computers. What is known about the durations required for these two algorithms is then combined in Section 5 to explore the future feasibility of implementing PAS on quantum computers using the Grover method. We term the combined algorithm *Grover Adaptive Search*.

Terminology and Notation

We consider the following finite global optimization problem:

$$\begin{aligned} \text{minimize} \quad & f(x) \\ \text{subject to} \quad & x \in S \end{aligned}$$

where $f(x)$ is a real valued function on a finite set S.

The following algorithms for finite optimisation are considered in this paper: *Pure Random Search* (PRS) [2] samples the domain at each iteration according to a fixed distribution. *Pure Adaptive Search* (called *Strong-PAS* in [11]) samples from that part of the domain giving a strictly improving objective function value at each iteration. *Hesitant Adaptive Search* (HAS) [3] behaves like PAS, except that each iteration is only *successful* with a certain

probability; otherwise the iteration passes with no improvement found. The success probability in HAS is a function of the best objective function value yet at hand.

Grover Adaptive Search (GAS), introduced here, is a stochastic algorithm using at each step a Grover Quantum Search [4] to find a point with a strictly improving objective function value. This is a direct application of Grover Quantum Search, where the 'marked' points are those with function values better than the best value seen up to that iteration.

Throughout this paper we associate with the objective function f the following definitions. Let N denote the size of the domain S. Let $\ell_1 < \ldots < \ell_K$ be the distinct objective function values, that is, the distinct levels appearing on the graph of f. Notice that there may be more than K points in S; in fact the most interesting cases are when $K \ll N$. Given the uniform probability measure μ on S, we define a probability measure $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)$ on the range of f as follows. Let π_j be the probability that any iteration of pure random search attains a value of ℓ_j. That is, $\pi_j = \mu(f^{-1}(\ell_j))$ for $j = 1, 2, \ldots, K$, so more common range values are more likely under $\boldsymbol{\pi}$. Let p_j denote $\sum_{i=1}^{j} \pi_i$, the probability that PRS attains a value of ℓ_j or less. In particular, $p_K = 1$.

2 Pure Adaptive Search

The Pure Adaptive Search algorithm, introduced in [7], is a stochastic algorithm for unconstrained global optimisation which has a very favourable convergence rate, but which has been thought impossible to implement efficiently. The algorithm states simply that each sample point should be distributed uniformly in the current *improving region*, that is, in the set of points yielding objective function values better than any yet seen. We will consider only PAS in the case of a finite domain and a uniform initial sampling distribution. Formally,

Pure Adaptive Search

- ⋆ Generate a PRS iterate $x_1 \in S$, and set $y_1 = f(x_1)$.
- ⋆ For $i = 1, 2, \ldots$, as long as $y_i \neq \ell_1$, do:
  - ⋆ Define $S_i = \{x \in S : f(x) < y_i\}$.
  - ⋆ Randomly generate x' uniformly from S_i.
  - ⋆ Set $y' = f(x')$.
  - ⋆ Set $x_{i+1} = x'$ and $y_{i+1} = y'$

The Hesitant Adaptive Search algorithm [3] is a simple generalisation of

PAS. Define a success (or "bettering") probability function $b : \mathsf{R} \to (0, 1]$. The algorithm is as follows:

Hesitant Adaptive Search

- ⋆ Generate a PRS iterate $x_1 \in S$, and set $y_1 = f(x_1)$.
- ⋆ For $i = 1, 2, \ldots$, as long as $y_i \neq \ell_1$, do:
  - ⋆ Define $S_i = \{x \in S : f(x) < y_i\}$.
  - ⋆ With probability $b(y_i)$,
    - ⋆ Randomly generate x' uniformly from S_i.
    - ⋆ Set $y' = f(x')$.
    - ⋆ Set $x_{i+1} = x'$ and $y_{i+1} = y'$.
  - ⋆ otherwise,
    - ⋆ Set $x_{i+1} = x_i$ and $y_{i+1} = y_i$.

Pure Adaptive Search is just HAS with b equal to the constant function 1. In [11], the expected number of iterations before termination has been shown to equal $\sum_{j=2}^{K} \pi_j / p_j$, and thus to be bounded above by $\ln(1/p_1)$. This convergence rate of PAS is very appealing. It equates to a linear dependence on domain dimension (see for instance [10] and [11]), whereas typically the effort required to solve global optimisation problems using known practical methods increases exponentially with domain dimension.

The flaw preventing practical use of PAS can be seen in the step requiring the generation of the random point x'. If the objective function is unknown (as with a problem of the "black-box" type), or unless it yields readily to analytical study, the improving region may be unknown at each iterate. In addition, even the task of generating uniformly distributed points in *known* regions can present computational difficulties. Attempts to implement PAS, either exactly or approximately, have concentrated on methods for producing points distributed approximately uniformly in the unknown improving region: for instance, the Improving Hit-and-Run algorithm in [12] uses the first sample from a Markov chain whose limiting distribution is known to be the required uniform distribution on the improving region.

Recently, however, physicists studying quantum computing have provided a method that opens up the possibility of an efficient implementation of PAS in the future. A broad overview of this method will be provided in Sections 3 and 4. Some readers may wish to skip directly to the summary at the end of Section 4, where this overview closes and the pertinent quantum result is given.

3 Quantum Computation

The Grover mechanism referred to in this article is one of the major advances to date in the fledgling field of quantum computation, the study of the workings of quantum computers. Quantum computers are predicted technology; research into the hardware implementation of these ideas is still in the very early experimental stages [6]. But the continuing miniaturisation of electronics, combined with our growing awareness of the possible benefits of quantum computation over conventional computation, suggest that these devices are likely to appear within a few decades.

The characteristic feature of a quantum computer is that, in place of a conventional computer's *bits*, it uses *quantum bits*, or *qubits*. A qubit is a quantum system with two eigenstates, denoted $|0\rangle$ and $|1\rangle$. (Dirac notation is used in this and the following section to describe quantum states, as is standard in quantum mechanics. If the possible states of a quantum system are situated in a Hilbert space H, then the general element of H is denoted by $|u\rangle$.) Whereas the state of an ordinary bit can be either 0 or 1, the state of a qubit can be any combination

$$\alpha|0\rangle + \beta|1\rangle,$$

where α and β are complex numbers with $|\alpha|^2 + |\beta|^2 = 1$. When the value stored in a qubit in this state is measured, an output of either 0 or 1 is observed: 0 is observed with probability $|\alpha|^2$, and 1 is observed with probability $|\beta|^2$.

That a qubit can be in a simultaneous superposition of "off" and "on" eigenstates presents the possibility of what has been called *quantum parallelism*, where a single quantum circuit can simultaneously perform a calculation on a superimposed input, corresponding to very many conventional inputs.

The study of quantum computation received a marked increase in attention in 1994 with the publication of Shor's factoring algorithm [8]. This and other developments of the theory have indicated that quantum computers may one day enjoy a huge speed advantage over conventional computers for many types of problems.

4 Grover Search

The quantum procedure germane to our purposes is Grover Search. This section introduces it, and defines the Grover probability distribution (1).

Consider the following general search problem. Let n be a positive integer, and let $S = \{0,1\}^n$, so that the domain size $N = 2^n$. Let $g : S \to \{0,1\}$, and assume that we have a black-box quantum circuit implementing g, called the *oracle*. We wish to find a point $u \in S$ such that $g(u) = 1$. We call such points *marked*, we denote the set of marked points by $M \subseteq S$, and we denote the number of marked states by $m = |M|$. We may or may not be aware of the value of m.

Grover introduced in [4] a means of implementing a certain phase-space rotation of the state of a quantum system encoding points in the domain S. This rotation can be used to move toward the unknown marked states. The description here draws heavily from [1].

For any real numbers k and l such that $mk^2 + (N-m)l^2 = 1$, define

$$|\Psi(k,l)\rangle = \sum_{u \in M} k|u\rangle + \sum_{u \in S \setminus M} l|u\rangle.$$

The pivotal step of the Grover algorithm is an efficient method for transforming the quantum state $|\Psi(k,l)\rangle$ into the state

$$\left|\Psi\left(\frac{N-2m}{N}k + \frac{2N-2m}{N}l, \frac{-2m}{N}k + \frac{N-2m}{N}l\right)\right\rangle,$$

using a single (superimposed) oracle query.

As explained in [4, 1] this is a rotation in the plane spanned by the vectors

$$\frac{1}{\sqrt{m}} \sum_{u \in M} |u\rangle \text{ and } \frac{1}{\sqrt{N-m}} \sum_{u \in S \setminus M} |u\rangle,$$

by an angle determined by m and N. Note that the equal amplitude state

$$\frac{1}{\sqrt{N}} \sum_{u \in S} |u\rangle,$$

which is relatively simple to prepare within a quantum computer, lies in this plane. When the equal amplitude state is transformed by the Grover rotation operator r times in succession, we call it a *Grover run of r rotations*, and the resulting quantum state is

$$\frac{\sin((2r+1)\theta)}{\sqrt{N}\sin\theta} \sum_{u \in M} |u\rangle + \frac{\cos((2r+1)\theta)}{\sqrt{N}\cos\theta} \sum_{u \in S \setminus M} |u\rangle,$$

where $\theta \in [0, \frac{\pi}{2}]$ is such that $\sin^2\theta = \frac{m}{N}$. If this state is then observed, it will collapse to each eigenstate (element of S) with probability equal to the squared modulus of that eigenstate's amplitude in the quantum state.

Grover Run and Distribution

Thus if S is a finite set with a marked subset $M \subset S$ with $|M| = m$, and if we can construct a quantum circuit (the *oracle*) to test individual domain points for membership of M, and if $\theta \in [0, \frac{\pi}{2}]$ is such that $\sin^2\theta = \frac{m}{N}$, and $r \in \mathsf{N}$, then we can execute a *Grover run of r rotations*, for a cost in proportion to that of $r+1$ oracle queries, in order to generate domain points according to the probability distribution γ on S with

$$\gamma(\{x\}) = \begin{cases} \frac{\sin^2(2r+1)\theta}{N\sin^2\theta}, & x \in M, \\ \frac{\cos^2(2r+1)\theta}{N\cos^2\theta}, & x \in S \backslash M. \end{cases} \tag{1}$$

We define $\mathcal{G}(S, M, r)$, the *Grover distribution on S with parameters (M, r)*, to equal this distribution.

If $X \sim \mathcal{G}(S, M, r)$, then

$$\mathsf{P}[X \in M] = \sum_{u \in M} \mathsf{P}[X = m] = \frac{m\sin^2(2r+1)\theta}{N\sin^2\theta} = \sin^2(2r+1)\theta.$$

We define a new function g_r and **summarise** this section:

Axiom *There is a search procedure on a quantum computer called a Grover run of r rotations that will find a marked state with probability*

$$g_r(p) = \sin^2\left[(2r+1)\arcsin\sqrt{p}\right]$$

when the proportion of marked states is p.

5 Grover Adaptive Search

The Grover mechanism described in the previous section is the key opening up the possiblity of efficient general implementation of PAS (that is, implementations for use in the absence of known regularity conditions). In this section, we define an algorithm using the Grover distribution, and provide the main result of this paper.

Let r be a fixed positive integer, called the *rotation count*. Then *Grover Adaptive Search with r rotations per step*, or GAS(r), is as follows:

Grover Adaptive Search

- ⋆ Generate a PRS iterate $x_1 \in S$, and set $y_1 = f(x_1)$.
- ⋆ For $i = 1, 2, \ldots$, as long as $y_i \neq \ell_1$, do:
  - ⋆ Define $S_i = \{x \in S : f(x) < y_i\}$.
  - ⋆ Set x to the observed output of a Grover run of r rotations, starting from the equal amplitude vector, with the states in S_i considered "marked".
  - ⋆ Do a classical evaluation to determine $y = f(x)$.
  - ⋆ If $y < y_i$, then
    - ⋆ Set $y_{i+1} = y$.
  - ⋆ otherwise,
    - ⋆ Set $y_{i+1} = y_i$.

In the algorithm as presented, the number of Grover rotations used in each step is constant, r. Variations of this algorithm are possible, in which the number of rotations per step varies, either according to a fixed schedule, or adaptively.

Markov Chain Description

The stochastic process $\{Y_k : k = 1, 2, \ldots\}$ for GAS can be modeled as a Markov chain with states $\ell_1, \ldots, \ell_K$, where state ℓ_1 represents the global optimum. The initial probability distribution Y_1 is given by $\boldsymbol{\pi}$ (i.e., we begin with a PRS step). In standard Markov chain terminology [5, 11], ℓ_1 is the absorbing state of this chain and all other states are transient. GAS converges when the chain reaches the absorbing state.

A GAS iteration will always return a value less than or equal to the best so far. If the best so far is ℓ_{m+1}, the Grover run of r rotations succeeds with probablility $g_r(p_m)$. In that event, all domain points in the improving region are equally likely, so the K by K transition matrix P, in standard form, having the i, jth element $\mathsf{P}[Y_k = \ell_j | Y_{k-1} = \ell_i]$, for GAS is as follows. Note that the $(2,1)$-entry is equal to $g_r(p_1)$, but is written as $g_r(p_1)\frac{\pi_1}{p_1}$ to clarify the structure of the matrix.

$$\begin{bmatrix} 1 & 0 & 0 & \ldots & 0 & 0 \\ g_r(p_1)\frac{\pi_1}{p_1} & 1 - g_r(p_1) & 0 & \ldots & 0 & 0 \\ g_r(p_2)\frac{\pi_1}{p_2} & g_r(p_2)\frac{\pi_2}{p_2} & 1 - g_r(p_2) & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ g_r(p_{K-1})\frac{\pi_1}{p_{K-1}} & g_r(p_{K-1})\frac{\pi_2}{p_{K-1}} & g_r(p_{K-1})\frac{\pi_3}{p_{K-1}} & \ldots & g_r(p_{K-1})\frac{\pi_{K-1}}{p_{K-1}} & 1 - g_r(p_{K-1}) \end{bmatrix}$$

It follows that $\mathsf{P}[Y_k = \ell_i]$, the probability of objective function value ℓ_i for GAS on the kth iteration, is the ith entry of $\boldsymbol{\pi} P^{k-1}$.

Note that for the variation of GAS with a schedule of different rotation counts, the corresponding stochastic process would be modelled by a nonhomogeneous Markov chain.

Realisation of HAS

Hitherto HAS and PAS have been viewed primarily as theoretical tools, but quantum computing casts these algorithms in a new light.

Theorem *GAS is an implementation of HAS. The number of GAS(r) iterations before termination has expectation*

$$\sum_{j=2}^{K} \frac{\pi_j}{g_r(p_j)p_j}.$$

and variance

$$\sum_{j=2}^{K} \left(\frac{2}{g_r(p_j)} - \frac{\pi_j}{g_r(p_j)p_j} - 1 \right) \frac{\pi_j}{g_r(p_j)p_j}.$$

Proof: Using any improvement probability function b satisfying $b(\ell_i) = g_r(p_i)$ for each $i \in \{1, \ldots, K\}$ reveals GAS as an implementation of HAS. The moment results follow from Corollary 3.2 in [9]. ■

Note that all higher moments are also available [9].

Realisation of PAS

The exact expressions given in the theorem are valid for an arbitrary distribution reflected by π. However, if the distribution is particularly well-behaved, GAS becomes PAS.

Corollary *If a function f has associated p_j satisfying $g_r(p_j) = 1$, then GAS(r) realizes PAS for this f.*

Proof: The matrix P reduces to the transition matrix for strong-PAS in [11] since $g_r(p_j) = 1$ for all $j = 1, \ldots, K$. ■

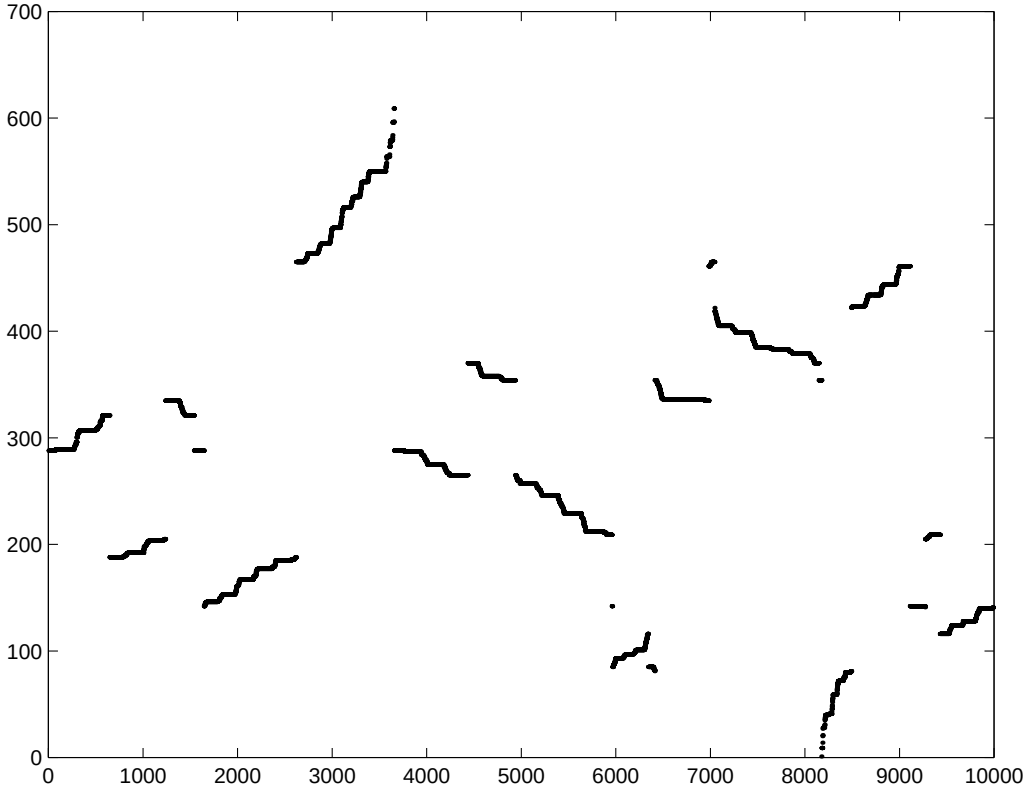

Figure 1: A computer-generated example objective function

Example

As an example, consider the univariate function depicted in Figure 1. This is a function on a domain of size 10000, whose image consists of 624 distinct values. The relative frequency distribution of these values is shown in Figure 2.

When applied to this function, the number of iterations required by GAS(65) before termination has expectation 8.9 and standard deviation 3.0. For the same function, the number of PRS iterations before termination has expectation 6736 and standard deviation 8495.

To compare these results fairly, recall that each iteration performed by GAS(65) uses 65 Grover rotations, each requiring a single oracle query, followed by a classical evaluation, that is, 66 function evaluations. Thus GAS(65) has an expected *effort* comparable to 587.1 PRS iterations, with a standard deviation comparable to 198.5 PRS iterations.

Note that the behaviour of GAS on this function depends only on the distribution given in Figure 2. We generated the example by first building

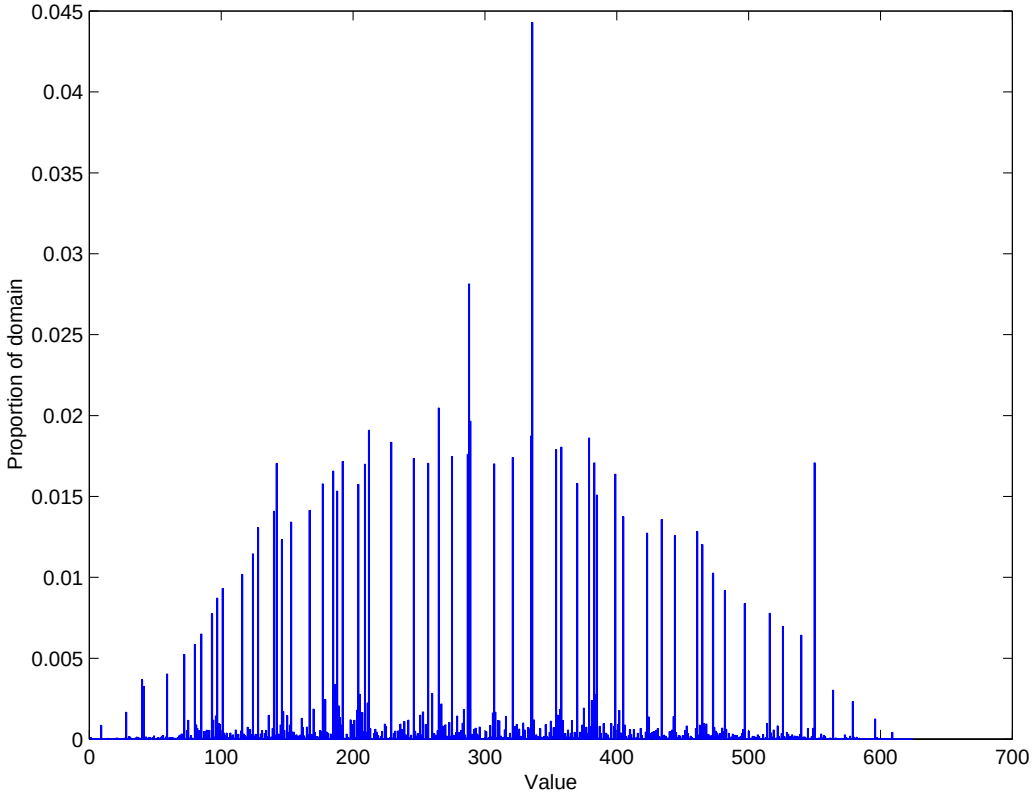

Figure 2: Relative frequency distribution of values taken by the objective function in Figure 1

a monotone function with the distribution shown in Figure 2 and then shuffling to produce the function shown in Figure 1. This example was made using some very preliminary results aimed at identifying characteristics of functions for which GAS(r) will be particularly suited. A more thorough study is planned for future work. Briefly, though, we chose $r = 65$ because $2r + 1$ is prime, and constructed a distribution for which g_r gave values of at least 0.8 at all points. As expected, this example required very few iterations of GAS(65), and it is encouraging to see that this advantage remains pronounced when total effort is considered.

6 Discussion

Remarks

Throughout this article it has been assumed that the domain's cardinality is a power of two. Where this is not the case, note that the domain can be artifically couched within a larger domain whose cardinality *is* a power of two. The artificial domain need not be larger than twice the size of the true domain. An objective function can be extended to have maximal value on the artifically added points, and similarly a marking oracle can merely ignore (not mark) the artificial points. Thus this restriction is vacuous.

The study of stochastic algorithms has often been seen as less worthwhile than that of deterministic ones. In the future, with quantum machines intrinsically based on physics showing stochastic behaviour, it would appear that algorithms such as PAS and HAS will come to the fore.

Further Work

As mentioned earlier, it might be expected that efficiency gains could be made by allowing the rotation count to vary from one GAS step to another, either according to a fixed schedule, or dynamically, depending on which Grover runs yielded improvements.

A future paper will investigate convergence time bounds for GAS which apply under certain favourable conditions.

7 Summary

Physicists studying quantum computing have provided a method that opens up the future possibility of an efficient global optimisation algorithm. Grover Adaptive Search (GAS) *realises* Hesitant Adaptive Search and on occasions Pure Adaptive Search, which up to now have been thought of primarily as theoretical tools.

References

[1] M. Boyer, G. Brassard, P. Høyer and A. Tapp, Tight bounds on quantum searching, *Fortschr.Phys.* 46 (1998) 493–506.

[2] S.H. Brooks, A discussion of random methods for seeking maxima, *Oper.Res. 6* (1958) 244-251.

[3] D.W. Bulger and G.R. Wood, Hesitant adaptive search for global optimisation, *Math.Program.* 81 (1998) 89–102.

[4] L.K. Grover, A fast quantum mechanical algorithm for database search, *Proceedings of the 28th Annual ACM Symposium on Theory of Computing* (1996).

[5] J.G. Kemeny and J.L. Snell, *Finite Markov Chains* (Springer-Verlag, New York, 1960).

[6] R. Laflamme, Los Alamos scientists make seven bit quantum leap, `http://www.lanl.gov/worldview/news/releases/archive/00-041.html` (2000).

[7] N.R. Patel, R.L. Smith and Z.B. Zabinsky, Pure adaptive search in Monte Carlo optimization, *Math.Program.* 43 (1988) 317–328.

[8] P.W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, *SIAM J.Comput.* 26 (1997) 1484-1509.

[9] G.R. Wood, Z.B. Zabinsky and B.P. Kristinsdottir, Hesitant adaptive search: the distribution of the number of iterations to convergence, *Department of Mathematics and Computing Technical Report* 97-002, Central Queensland University, Rockhampton, Australia, 19pp. (1997).

[10] Z.B. Zabinsky and R.L. Smith, Pure adaptive search in global optimization, *Math.Program.* 53 (1992) 323–338.

[11] Z.B. Zabinsky, G.R. Wood, M.A. Steel and W.P. Baritompa, Pure adaptive search for finite global optimisation, *Math.Program.* 69 (1995) 443–448.

[12] Z.B. Zabinsky, R.L. Smith, J.F. McDonald, H.E. Romeijn and D.E. Kaufman, Improving hit-and-run for global optimization, *J.Glob.Optim.* 3 (1993) 171–192.