

On the query complexity of finding a local maximum point

A. L. Rastsvetaev L. D. Beklemishev^{*†}

November 1, 2000

Abstract

We calculate the minimal number of queries sufficient to find a local maximum point of a function on a discrete interval, for a model with M parallel queries, $M \geq 1$. Matching upper and lower bounds are obtained. The bounds are formulated in terms of certain Fibonacci type sequences of numbers.

Keywords: computational complexity, decision trees, local maximum search, Fibonacci numbers

1 Introduction

Query complexity has been studied in the context of decision trees, sorting and searching problems (see e.g. [6]) as well as in recursion theory [4]. It is relevant in the situation, where one has costly access to (parts of) the input data, whereas the time and space costs of the computation can be ignored. If the input is given in the form of a finite function f and as a model of computation one considers, say, oracle Turing machines that may ask questions of the form $f(x) = ?$, the natural query complexity measure means the minimal number of oracle queries necessary to solve a given problem (depending on the size of the domain of f).

This paper arose out of the work of the second author on formal arithmetic. In this work, query complexity lower bounds were applied to separate some fragments of Peano arithmetic [2, 3]. In [3] a logarithmic lower bound on the query complexity of the problem of computing a local maximum point of a discrete function was used to show the independence of the schema of induction for decidable predicates $I\Delta_1$ from the set of all true arithmetical Π_2 -sentences.

As early as in the '50s the problem of finding a maximum point of a unimodal function (that is, the function with a single maximum on a given interval) attracted the attention of analysts in the continuous case. In [5] the so-called

^{*}Supported by Lise Meitner Research Fellowship (FWF), RFBR and INTAS.

[†]The corresponding author. Address: Department of Philosophy, Utrecht University, Heidelberglaan 8, 3584 CS Utrecht, The Netherlands. E-mail: lev@phil.uu.nl.

Fibonacci search techniques was proposed and shown to be optimal in the min-max sense. The same techniques applies in the discrete case for a model with dependent queries. It yields that $\log_\alpha(n)$ queries (up to an additive constant) are sufficient to find a local maximum point of a function on an interval of length n , where $\alpha = \frac{1+\sqrt{5}}{2}$ is the ‘golden section’ ratio. This number of queries can also be shown optimal in the worst case, we give an independent proof in Theorem 2.

In the present paper a more general situation is considered, when our computing device can ask M queries simultaneously (the model with parallel queries). We obtain matching upper and lower bounds on the number of such M -queries needed to find a local maximum point of a discrete function. This leads to a family of search procedures that generalize Fibonacci and binary search (respectively, for odd and even M).

Let $N = \{0, 1, \dots\}$ be the set of natural numbers. If $a, b \in N$ and $a \leq b$ let $I = [a, b]$ denote the *interval* $\{x \in N : a \leq x \leq b\}$ of *length* $\|I\| = b - a$. a and b will be called *boundary points* and $(a, b) = [a + 1, b - 1]$ is the set of *interior points* of I . $[a, b] \stackrel{\text{def}}{=} [a, b - 1]$, $(a, b) \stackrel{\text{def}}{=} [a + 1, b]$.

A function $f : I \rightarrow N$ has a *local maximum* at a point $x \in I$, if $(\forall y \in I)(|x - y| \leq 1 \Rightarrow f(y) \leq f(x))$.

An M -*query* to a function f consists of $M \geq 1$ queries of the form $f(x) = ?$ made and answered simultaneously. Thus, an M -query is determined by an M -tuple of arguments and receives an M -tuple of values as a result. $q_M(n)$ denotes the minimal number of M -queries that suffices for an oracle Turing machine \mathcal{M}^f to find a local maximum point of f on interval $[0, n]$. Alternatively, one could define $q(n)$ as the minimal depth of an appropriate decision tree solving the same problem.

2 Upper bound

Given an $M \geq 1$ we define a sequence of numbers η_k^M as follows. Let $\eta_0^M = \eta_1^M = 0$ and, for $k \geq 1$,

$$\eta_{k+1}^M = \begin{cases} m\tilde{\eta}_k^M - 1, & \text{where } \tilde{\eta}_k^M = \eta_k^M + \eta_{k-1}^M + 2 \quad (M = 2m - 1 \text{ odd}); \\ (m+1)\tilde{\eta}_k^M - 2, & \text{where } \tilde{\eta}_k^M = \eta_k^M + 2 \quad (M = 2m \text{ even}). \end{cases}$$

For brevity, in what follows we shall sometimes omit the superscript M . Notice that $\eta_{k-1}^1 + 1$ is the sequence of Fibonacci numbers, see e.g. [6].

Theorem 1 $q_M(\eta_{k+1}^M - 1) \leq k$, for all $M, k \geq 1$.

Proof. Let us fix $M \geq 1$. Obviously, $q_M(\eta_2^M - 1) = q_M(M - 1) = 1$, if $M > 1$, and $q_1(\eta_2^1 - 1) = q_1(0) = 0$.

Let $p_k^i = i\tilde{\eta}_k - 1$, for $1 \leq i \leq m = \lceil M/2 \rceil$.

We first consider the case of $M = 2m$ and reason by induction on k . For $k = 1$ the claim is obvious.

Induction step. Assume $q_M(\eta_k^M - 1) \leq k - 1$, where $k \geq 2$, and consider the interval $I = [0, \eta_{k+1}^M)$. Execute the M -query $X = \{x_1, \dots, x_M\}$ with $x_{2i} = p_k^i$ and $x_{2i-1} = p_k^i - 1$, for $1 \leq i \leq m$. We have the following three options, and in each case define the corresponding interval J :

1. $f(x_1) \geq f(x_2)$, $J = [0, x_1)$.
2. $f(x_{2i-1}) \leq f(x_{2i})$ and $f(x_{2i+1}) \geq f(x_{2i+2})$, for some $1 \leq i < m$; $J = (x_{2i}, x_{2i+1})$.
3. $f(x_{M-1}) \leq f(x_M)$, $J = (x_M, \eta_{k+1}^M)$.

J has length $\eta_k^M - 1$ and, by the induction hypothesis, one can find a local maximum point (lmp) z in J using $k - 1$ queries. If z is an interior point of J or a boundary point of I , then z is a lmp in I . Otherwise, its neighbour point $z' \notin J$ occurs, by construction, in X . If $f(z') \leq f(z)$, then z is a lmp in I , otherwise z' is a lmp in I . Thus, a lmp can be found using k queries.

In the case of $M = 2m - 1$ we introduce an auxiliary function $\tilde{q}_M(k)$. Suppose that a function f is defined on an interval $I = [a, a + \tilde{\eta}_{k+1}^M - 1)$, for $k \geq 1$, and the value $f(x_0)$ is already known, where x_0 is one of the two symmetric points $x_0 = a + \eta_k^M$ or $x_0 = a + \tilde{\eta}_{k+1}^M - \eta_k^M - 2$. (Thus, if y is the closest to x_0 boundary point of I , we have $|x_0 - y| = \eta_k^M$.) Then there is a lmp $z \in I$ such that $f(z) \geq f(x_0)$. We let $\tilde{q}_M(k)$ denote the minimal number of M -queries necessary to find such a point z . By the symmetry, \tilde{q}_M does not depend on a and x_0 , so we may also always assume $a = 0$.

Lemma. $\tilde{q}_M(k) \leq k$, for $M, k \geq 1$.

Proof by induction on k .

Basis: $\tilde{q}_M(1) = 1$, because $\tilde{\eta}_2^M - 2 = M$.

Induction step. Assume $\tilde{q}_M(k - 1) \leq k - 1$, where $k \geq 2$.

Consider the interval $I = [0, \tilde{\eta}_{k+1}^M - 1)$ and the set $X = \{x_1, \dots, x_{M+1}\} \subseteq I$, where $x_{2i} = p_k^i - 1$ and $x_{2i-1} = p_k^i - \eta_{k-1}^M - 1$, for $1 \leq i \leq m$. The value of f at x_1 or at x_{M+1} is already known. Hence, by executing the M -query $X' = \{x_2, \dots, x_{M+1}\}$ (resp., $X' = \{x_1, \dots, x_M\}$) we learn all the values of f on X .

Let $x_i \in X$ be such that $f(x) \leq f(x_i)$ for all $x \in X$. Then we have the following three options and define the corresponding interval J :

1. $i = 1$, $J = [0, x_2)$.
2. $1 < i \leq M$, $J = (x_{i-1}, x_{i+1})$.
3. $i = M + 1$, $J = (x_M, \tilde{\eta}_{k+1}^M - 1)$.

J has length $\tilde{\eta}_k^M - 2$, and f is defined at a suitable point $x_i \in J$ such that the condition of Lemma 1 is satisfied. (Indeed, it is easy to see that for the closest to x_i boundary point y of J , $|x_i - y| = \eta_{k-1}^M$.) Hence, by induction hypothesis,

one can find a lmp z in J within $k-1$ queries. If z is an interior point of J or a boundary point of I , then z is a lmp in I . Otherwise, its neighbour point $z' \notin J$ occurs, by construction, in X . By the definition of \tilde{q}_M , $f(z) \geq f(x_i)$, whereas by the choice of i , $f(x_i) \geq f(z')$. Therefore, $f(z) \geq f(z')$ and z is a lmp in I . This proves the Lemma.

Now we complete the proof of Theorem 1. For the initial interval $I = [0, \eta_{k+1}^M)$ and $k \geq 2$ we execute the M -query $X = \{x_1, \dots, x_M\}$ with $x_{2i} = p_k^i - 1$ and $x_{2i-1} = p_k^i - \eta_{k-1}^M - 1$, for $1 \leq i < m$, and $x_M = p_k^m - \eta_{k-1}^M - 1$. Let $x_i \in X$ be such that $f(x) \leq f(x_i)$ for all $x \in X$. Then we have three options and define the corresponding interval J :

1. $i = 1$, $J = [0, x_2)$.
2. $1 < i < M$, $J = (x_{i-1}, x_{i+1})$.
3. $i = M$, $J = (x_{M-1}, \eta_{k+1}^M)$.

$\|J\| = \tilde{\eta}_k^M - 2$ and f is defined at an appropriate point $x_i \in J$. Hence, by Lemma, one can find a lmp z in J using $k-1$ queries. By the reasoning similar to the one in Lemma we then find a lmp in I using k queries. **Q.e.d.**

3 Lower bound

Here we show that the algorithm presented in Theorem 1 is optimal.

Let us fix some $M, k \geq 1$. Let $p^i = i\tilde{\eta}_k^M - 1$, for $0 \leq i \leq m = \lceil M/2 \rceil$. Then $[0, \tilde{\eta}_{k+1}^M) = \cup_{i=1}^{m+1} I_i$, where $I_i = (p^{i-1}, p^i]$, for $1 \leq i \leq m$, and $I_{m+1} = (p^m, \tilde{\eta}_{k+1}^M)$. We have $I_i \cap I_j = \emptyset$, for $1 \leq i \neq j \leq m+1$; $\|I_i\| = p^i - p^{i-1} - 1 = \tilde{\eta}_k - 1$, for $1 \leq i \leq m$, and $\|I_{m+1}\| = \tilde{\eta}_{k+1}^M - p^m - 2 = \eta_k$ (M odd), or $\tilde{\eta}_k - 1$ (M even).

Proposition 1. For any interval I of length $\geq \eta_{k+1}^M$ and any set X satisfying $|X| \leq M$, there exists a (proper) subinterval $J \subset I$ of length $\tilde{\eta}_k^M - 1$ containing no more than one element of X .

Proof. For an interval $I = [0, \eta_{k+1}^M]$ assume, for a contradiction, that $|I_i \cap X| \geq 2$ for all $1 \leq i \leq m$. Then $M \geq |X| \geq \sum_{i=1}^m |I_i \cap X| \geq 2m = M + 1$, if M is odd.

If M is even, it follows that $|(p^m, \eta_{k+1}^M] \cap X| = 0$, hence $|(p^m, \eta_{k+1}^M] \cap X| \leq |(p^m, \eta_{k+1}^M] \cap X| + 1 \leq 1$. Yet, $\eta_{k+1}^M - p^m = \tilde{\eta}_k - 1$, a contradiction. **Q.e.d.**

Proposition 2. For any interval I of length $\tilde{\eta}_{k+1}^M - 1$ and any set X of size $\leq M + 1$, at least one of the following two statements holds:

- (1) There exists a subinterval $J \subset I$ of length η_k^M containing no elements of X .
- (2) There exists a subinterval $J \subset I$ of length $\tilde{\eta}_k^M - 1$ containing no more than one element of X .

Proof. For an interval $I = [0, \tilde{\eta}_{k+1}) = \cup_{i=1}^{m+1} I_i$ notice that, if M is even, then $\|I_i\| = \tilde{\eta}_k - 1$, for all $1 \leq i \leq m+1$.

If M is odd, let us assume, for a contradiction, that neither (1) nor (2) holds. Then $|I_i \cap X| \geq 2$ and $|U(I_i) \cap X| \geq 1$ for all $1 \leq i \leq m$, where $U([a, a + \tilde{\eta}_{k+1})) = [a, a + \eta_{k+1}]$. Therefore $M + 1 \geq |X| \geq \sum_{i=1}^m |I_i \cap X| \geq 2m = M + 1$. Hence $I_{m+1} \cap X = \emptyset$, and since $\|I_{m+1}\| = \eta_k$ we have (1), a contradiction. **Q.e.d.**

Theorem 2 $q_M(\eta_k^M) \geq k$, for all $M \geq 1$ and $k \geq 2$.

Proof. Let us fix $M \geq 1$. Given an interval I of length η_{k+1}^M , $k \geq 1$, we shall construct, for any oracle Turing machine \mathcal{M} that asks $\leq k$ queries, a function $f : I \rightarrow N$ such that \mathcal{M}^f fails to compute its lmp.

We simulate the computation of \mathcal{M} on input I while suitably defining the values of f , whenever we need an answer to an oracle query. We define f in stages. At stage i our data consists of a finite function f_i , an interval I^i , and a certain parameter *height* h_i . Initially $I^0 = [a_0, b_0] = I$, f_0 is an empty function, and $h_0 = 0$.

For $1 \leq i \leq k$, f_i will be the part of f defined immediately after the i -th query X^i is answered. We shall select the i -th interval I^i and the height h_i depending on the current query X^i and the preceeding state $\langle I^{i-1}, f_{i-1}, h_{i-1} \rangle$. Then f_{i-1} will be extended to a function f_i such that $X^i \subseteq \text{dom}(f_i)$, and we can proceed with the simulation of \mathcal{M} . Our aim is to satisfy the following conditions, for all $1 \leq i \leq k$:

1. $I^{i-1} \supseteq I^i$, and f_i is an extension of f_{i-1} .
2. f_i is defined at no more than one point of I^i . If f_i is defined at some point of I^i , then $\|I^i\| = \tilde{\eta}_{k-i+1} - 1$. Otherwise, $\|I^i\| = \eta_{k-i+1}$.
3. f_i is defined on the set $I \setminus I^i$ and strictly increases (resp., decreases) on the left (right) hand side of I^i .
4. $f_i(x) \leq h_i$ for all $x \in \text{dom}(f_i) \cap (I \setminus I^i)$, and $f_i(x) > h_i$ for all $x \in \text{dom}(f_i) \cap I^i$.

Now we describe the construction of $I^i = [a_i, b_i]$, f_i and h_i . Let $X^i = \{x_1^i, \dots, x_M^i\}$ be the i -th query. Consider the following cases:

Case I. f_{i-1} is defined at no point of I^{i-1} . By Proposition 1, there exists a subinterval $I^i \subset I^{i-1}$, of length $\tilde{\eta}_{k-i+1} - 1$, such that $|I^i \cap X^i| \leq 1$. We set $h_i = h_{i-1} + \|I^{i-1}\|$, $f_i(z^i) = h_i + \|I^i\|$ (if $z^i \in I^i \cap X^i$), and, for all $x \in I^{i-1}$:

(i) if $x < a_i$, set $f_i(x) = h_i + x - a_i$; if $x > b_i$, set $f_i(x) = h_i + b_i - x$.

Case II. f_{i-1} is defined at some point $z^{i-1} \in I^{i-1}$, hence $\|I^{i-1}\| = \tilde{\eta}_{k-i+2} - 1$. Let \tilde{X}^i be $X^i \cup \{z^{i-1}\}$.

Subcase 1 There exists a subinterval $I^i \subset I^{i-1}$ of length η_{k-i+1} such that $I^i \cap \tilde{X}^i = \emptyset$. We set $h_i = h_{i-1} + 2\|I^{i-1}\|$ and, for all $x \in I^{i-1}$:

- (ii) For $b_i < z^{i-1}$, if $x < a_i$, set $f_i(x) = h_i + x - a_i$;
if $x > b_i$, set $f_i(x) = f_{i-1}(z^{i-1}) + z^{i-1} - x$.
- (iii) For $z^{i-1} < a_i$, if $x < a_i$, set $f_i(x) = f_{i-1}(z^{i-1}) - z^{i-1} + x$;
if $x > b_i$, set $f_i(x) = h_i + b_i - x$.

Subcase 2 Subcase 1 does not hold. Then by Proposition 2 there exists a subinterval $I^i \subset I^{i-1}$ of length $\tilde{\eta}_{k-i+1} - 1$ such that $I^i \cap \tilde{X}^i = \{z^i\}$.

- (a) If $z^i = z^{i-1}$, we set $h_i = h_{i-1} + \|I^{i-1}\| - \|I^i\|$ and, for all $x \in I^{i-1}$, define f_i as in (i) above.
- (b) If $z^i \neq z^{i-1}$, we set $f_i(z^i) = h_i + \|I^i\|$, $h_i = h_{i-1} + 2\|I^{i-1}\|$ and, for all $x \in I^{i-1}$, define f_i as in (ii, iii) above.

The construction, in particular, uniquely defines the values f_{i-1} on X^i , therefore we can proceed with the computation of \mathcal{M} . The invariant conditions 1–4 are now easy to check by induction on i , for all $1 \leq i \leq k$.

Since we assume that \mathcal{M} asks at most k queries, it terminates within k steps of the construction and reaches some state $\langle I^k, f_k, h_k \rangle$ where, by Conditions 1 and 2, either (I) I^k is of length $\eta_1 = 0$ and f_k is not defined on I^k , or (II) I^k is of length $\tilde{\eta}_1 - 1 = 1$ and f_k is defined at no more than one point on I^k .

Let y be the output of \mathcal{M}^{f_k} . Clearly, it will have the same output y for any function f extending f_k . In both cases (I) and (II) above there is at least one point in I^k where f_k is not yet defined. Therefore, taking into account Conditions 3 and 4, f can always be defined at this point in such a way that y is not a lmp. For such an f machine \mathcal{M}^f does not faithfully compute a lmp on I . **Q.e.d.**

4 Conclusion

As a corollary of the previous theorems we obtain

Theorem 3 For $m \geq 1$,

- (a) $q_{2m}(n) = \log_{m+1}(n) + O(1)$,
- (b) $q_{2m-1}(n) = \log_{\alpha_m}(n) + O(1)$, where $\alpha_m = \frac{m + \sqrt{m^2 + 4m}}{2}$.

Note that α_1 is the ‘golden section’ ratio.

Proof. By Theorems 1 and 2 and monotonicity of $q_M(n)$ we have, for $M \geq 1$ and $k \geq 2$, $q_M(n) = k \Leftrightarrow \eta_k^M \leq n < \eta_{k+1}^M$.

(a) It is easy to see that, for $M = 2m$, $\eta_k^M = 2(m+1)^{k-1} - 2$.

(b) Fix an odd $M = 2m - 1$. For brevity we omit below the superscript M and the subscript m . We know that η_k satisfies the equation

$$\eta_{k+1} = m(\eta_k + \eta_{k-1}) + 2m - 1,$$

hence $\eta'_k = \eta_k + 1$ satisfies the homogeneous equation

$$\eta'_{k+1} = m(\eta'_k + \eta'_{k-1}). \quad (1)$$

The latter has two (linearly) independent solutions $\eta'_{k+1} = \alpha^k$ and $\eta'_{k+1} = \beta^k$, where $\beta = \beta_m = \frac{m - \sqrt{m^2 + 4m}}{2}$ and $\alpha = \alpha_m = \frac{m + \sqrt{m^2 + 4m}}{2}$ are the roots of the quadratic equation $x^2 - mx - m = 0$. Hence, (1) has a general solution of the form $\eta'_{k+1} = c_1 \alpha^k + c_2 \beta^k$.

From the initial conditions

$$\begin{cases} \eta'_1 = 1 = c_1 + c_2, \\ \eta'_2 = 2m = c_1 \alpha + c_2 \beta, \end{cases}$$

we find $c_1 = \frac{2m - \beta}{\alpha - \beta}$ and $c_2 = \frac{\alpha - 2m}{\alpha - \beta}$. Thus, we obtain for η_{k+1} an explicit formula $\eta_{k+1} = c_1 \alpha^k + c_2 \beta^k - 1$. Since $|\beta_m| < 1$, for every m from some k on there holds $|c_2 \beta_m^k| < 1/2$. Hence, η_{k+1} is the nearest integer to $c_1 \alpha^k - 1$. **Q.e.d.**

Variants and generalizations of the problem of finding a local maximum point also make sense in other contexts. In the subsequent paper we present results on complexity of finding a local maximum of an integer-valued function on some classes of connected graphs, including multidimensional grids $[0, n]^d$.

Acknowledgements. The authors wish to thank Oleg Verbitsky for helpful discussions. Our thanks also go to the Institute for Algebra and Computational Mathematics, University of Technology, Vienna, for providing an excellent research environment.

References

- [1] M. Avriel and D. Wilde. Optimality proof for the symmetric Fibonacci search technique. *The Fibonacci Quarterly*, 4(1966), 265–269.
- [2] L. D. Beklemishev. A proof-theoretic analysis of collection. *Archive for mathematical logic*, 37 (1998), 275–296.
- [3] L. D. Beklemishev. On the induction schema for decidable predicates. *Logic Group Preprint Series*, Department of Philosophy, Utrecht University, 2000.
- [4] W. Gasarch. Bounded Queries in Recursion Theory. Birkhäuser, 1999.
- [5] J. Kiefer. Sequential minimax search for a maximum. *Proc. Amer. Math. Soc.*, 4 (1953), 502–506.
- [6] D. E. Knuth. The art of computer programming. Volumes I, II. CalTech, 1968.