BELADY'S ANOMALY IS UNBOUNDED

by Antal Miklós Iványi and Péter Fornai
29 November 2001
(Institute of Informatics of Eötvös Loránd University,
1117 Budapest, Pázmány Péter sétány 1/c.)
(mailto: tony@inf.elte.hu)
(*Paper is dedicated to seventies birthday of Mátyás Arató and László Varga*)

Abstract—Virtual memory of computers is usually implemented by demand paging. For some page replacement algorithms the number of page faults may increase as the number of page frames increases. Belady, Nelson and Shedler constructed reference strings for which page replacement algorithm FIFO produces near twice more page faults in a larger memory than a smaller one. They formulated the conjecture that 2 is a general bound. We prove that this ratio can be arbitrarily large.
Keywords—Algorithm performance, Memory management, FIFO, Anomaly.

1 INTRODUCTION

Let us consider a computer with two-level virtual memory [1]. Physical and secondary memory are divided into equal-sized blocks called frames. The processes reside in secondary memory and are broken into fixed-sized blocks called pages (the sizes of the frames and pages are the same). The run of the processes is described by reference strings.

According to the pure demand paging the execution of a process starts with empty physical memory (in the following simply *memory*) and a page is brought into the memory only when it is necessary. If a new page is demanded when the memory is full, page replacement algorithm chooses a page being in memory to be replaced by the demanded one.

We use the formal model proposed by P. Denning [1]. Let m, M, n and p be positive integers ($1 \leq m \leq M \leq n < \infty$), k a nonnegative integer, $A = \{a_1, a_2, \ldots, a_n\}$ a finite alphabet, A^k the set of k-length words and A^* be the set of finite words over A, where n is the number of pages, m is the number of frames in the small and M is the number of frames in the large memory, A is the set of pages.

Page replacement algorithms are handled as automatons having a memory of size m (or M), set of input signals $X = A$, set of output signals $Y = A \cup \{\epsilon\}$ and processing the sequence of input symbols $R = (r_1, r_2, \ldots, r_p)$ or $R = (r_1, r_2, \ \ldots\ )$. Memory state S_t $(t = 1,\ 2,\ \ldots,)$ is defined as a set of symbols stored in memory at the given moment t (after the processing of reference r_t). Any page replacement algorithm starts with an empty memory that is $S_0 = \{\}$. A concrete page replacement algorithm P is defined as a triple $P = (Q_P, q_0, g_P)$, where Q_P is the set of control states, $q_0 \in Q_P$ is the initial control state and g_P is the transition function determining the new memory state S', new control state q' and output symbol y using the old memory state S, old control state q and input symbol x.

We consider three page replacement algorithms: FIFO (First In First Out), LRU (Least Recently Used) and MIN (Minimal) [2].

FIFO is defined by $q_0 = ()$ and

$$g_{\text{FIFO}}(S, q, x) = \begin{cases} (S, q, \epsilon), & \text{if } x \in S, \\ (S \cup \{x\}, q', \epsilon), & \text{if } x \notin S \text{ and } |S| < m, \\ (S \cap \{y_1\} \cup \{x\}, q", y_1), & \text{if } x \notin S \text{ and } |S| = m, \end{cases}$$

where $q = (y_1, y_2, \ldots, y_k),\ q' = (y_1, y_2, \ldots, y_k, x)$ and $q'' = (y_2, y_3, \ldots, y_m, x)$.

LRU replaces the least recently used page of the memory, MIN replaces the page having maximal distance up to its next occurence in the reference string.

The number of page faults (number of changes of the memory states) is denoted by $f_P(R, m)$. If $M > m$ and $f_P(R, M) > f_P(R, m)$, then we have **an anomaly** and the ratio $f_P(R, M)/f_P(R, m)$ is called **anomaly ratio**. The first anomaly was observed in the practice [3, 4] and is called Belady's anomaly [2]. Later other examples of unexpected events were described [5,6,7].

Stack algorithms [2] do not suffer from anomaly.

Mihnovskiy and Shor [8] and Gecsei et al. [9] proved independently that MIN guarantees the minimal number of page faults at a fixed size of the memory.

Arató [10] and Benczúr et al. [11] proved that LRU is optimal in statistical sense among the algorithms having no concrete information on the continuation of the reference string.

A possible measure of the efficiency $E_P(R, m)$ of P is called **paging rate** and is defined by $f_P(R, m)/p$ for finite $R = (r_1, r_2, \ldots, r_p)$ and by

$$E_P(R, m) = \liminf_{k \to \infty} \frac{f_P(R_k, m)}{k}$$

for an infinite reference string R, where $R_k = (r_1, r_2, \ldots, r_k)$.

Let $1 \leq m < n$ and $C = (1, 2, \ldots, n)^*$ be an infinite cyclical reference string. Then the paging rates of FIFO and MIN are given in [12] $E_{\mathrm{FIFO}}(C, m) = 1$ and $E_{\mathrm{MIN}}(C, m) = (n - m)/(n - 1)$.

2 CLASSICAL EXAMPLE AND RESULTS

If we execute the reference string R = (1,2,3,4,1,2,5,1,2,3,4,5) [4] having 3 frames in the memory (for the simplicity natural numbers are used to denote the pages), then FIFO results the control state sequence $q_0 = ()$, $q_1 = (1)$, $q_2 = (1, 2)$, $q_3 = (1, 2, 3)$, $q_4 = (2, 3, 4)$, $q_5 = (3, 4, 1)$, $q_6 = (4, 1, 2)$, $q_7 = (1, 2, 5)$, $q_8 = (1, 2, 5)$, $q_9 = (1, 2, 5)$, $q_{10} = (2, 5, 3)$, $q_{11} = (5, 3, 4)$, $q_{12} = (5, 3, 4)$ and 9 page faults occured.

The execution of R using 4 frames will end in the control state (2,3,4,5) and 10 page faults occured, so $f_{\mathrm{FIFO}}(R, M)/f_{\mathrm{FIFO}}(R, m) = 10/9$.

Belady, Nelson and Shedler [4] gave a necessary and sufficient condition for the existence of anomaly and constructed reference strings resulting anomaly ratio which is close to 2.

Theorem 1 [4] *Let m and M be positive integers* $(1 \leq m < M)$ *and let A be a finite alphabet of cardinality at least $M + 1$. Then there exists a reference string $R \in A^*$ which produces an anomaly if and only if $M < 2m - 1$.*

Theorem 2 [4] *If m and M are positive integers satisfying the relations $m < M < 2m - 1$, then for sufficiently large p there exists a reference string $R = (r_1, r_2, \ldots, r_p)$ resulting anomaly ratio $f_{\mathrm{FIFO}}(R, M)/f_{\mathrm{FIFO}}(R, m)$ arbitrarily close to 2.*

Belady, Nelson and Shedler [4] formulated the following conjecture.

Conjecture *If m and M are positive integers and R is a reference string, then $f_{\mathrm{FIFO}}(R, M)/f_{\mathrm{FIFO}}(R, m) \leq 2$.*

3 DISPROVE OF THE CONJECTURE

Let $m = 5,\ M = 6,\ n = 7,\ k \geq 1$ and $R = UV^k$, where U = (1,2,3,4,5,6,7,1,2, 4,5,6,7,3,1,2,4,5,7,3,6,2,1,4,7,3,6,2,5,7,3,6,2,5) and $V = (1, 2, 3, 4, 5, 6, 7)^3$.

Now execution of U using m frames results the control state (7,3,6,2,5) and 29 page faults. After this each execution of V results 7 new faults and the same control state. Execution of U using M frames results the control state (2,3,4,5,6,7) and 14 page faults. After this each execution of V results 21 new page fault and the same control state.

Choosing $k = 7$ we get an anomaly ratio $(14 + 7 \times 21)/(29 + 7 \times 7) = 161/78 > 2$. If the number of repetitions of V grows, then the anomaly ratio tends to 3.

4 ANOMALY IS UNBOUNDED

We start with a well-known assertion of the number theory. Let $[a]_n$ be the equivalence class modulo n containing the number a and let $\mathbf{Z}_n$ be the set of all equivalence classes modulo n:

$[a]_n = \{a + kn : \ k \in \mathbf{Z}\}$ and $\mathbf{Z}_n = \{[a]_n : 0 \leq a \leq n-1\}$.

LEMMA 1 [12] *If the set* $c_1, c_2, \ldots, c_n$ *is a complete residue system* $\mathbf{Z}_n$ $(\bmod\ n)$, *the greatest common divisor of* a *and* n *equals to* 1, *then the set* $\{ac_1 + d, ac_2 + d, \ldots, ac_n + d\}$ *is also a complete residue system* $(\bmod\ n)$ *for arbitrary integer* d.

Let $a \pmod{b}$ denote the smallest positive representative of the residue class of numbers congruent with $a \pmod{b}$.

LEMMA 2 *Let* n *be an odd positive integer. Then the elements of the sequence* $W = (w_1, w_2, \ldots, w_n) = (1 \pmod{n}, 1 + (n-1)/2 \pmod{n}, 1 + 2(n-1)/2 \pmod{n}, \ \ldots, \ (n-1)(n-1)/2 \pmod{n})$ *have the following properties.*

a) They form a complete system of residues $(\bmod\ n)$.

b) If $W^k = s_1, s_2, \ldots s_{k \times n}$ *is the concenation of* k *copies of* W, *then any* n *neighbouring elements of* W^k *form a complete residue system* $(\bmod\ n)$.

c) If $j \geq 1$, *then* $s_{j+n+1} \equiv s_j + 1 \pmod{n}$.

PROOF. It is sufficient to use Lemma 1 and some concrete calculations. □

LEMMA 3 *Let* n *be an odd number* $(n \geq 5)$, $M = n - 1$ *and* $m = n - 2$. *If the small memory starts with control state* $(w_3, w_4, \ldots, w_n)$ *and the large memory starts with control state* $(2, 3, \ldots, n)$, *then the reference string* $R = (1, 2, \ \ldots, \ n)^{n-1)/2}$ *results* n *page faults in the small memory,* $n(n-1)/2$ *page faults in the large memory and the initial control states in both memories.*

Proof. This is a consequence of Lemma 3. □

Lemma 4 *Let* $1 \leq m < M < n$ and $b_1, b_2, \ldots, b_m$ *be arbitrary different elements of* $A = \{1, 2, \ldots, M+1\}$. *Then algorithm* ANOMALY *constructs a reference string resulting* $q = (b_1, b_2, \ldots, b_m)$ *in the small memory of size* m *while the pages are loaded into the large memory of size* M *in cyclical order* $1, 2, \ldots, n$.

The next algorithm is written in PASCAL-like pseudocode [13].

Construction algorithm ANOMALY(m, M, q, U).

Input: m size of the small memory; M size of the large memory; $q = (b_1, b_2, \ldots, b_m)$ the required control state of the small memory.
Output: $U = (u_1, u_2, \ldots)$ reference string.
Working variables: S actual state of the small memory; t index of the next reference; $A = \{1, 2, \ldots, M, M+1\}$ the set of pages.

1. **for** $k \leftarrow 1$ **to** M
2.   **do** $u_k \leftarrow j$
3. $t \leftarrow M+1$
4. $i \leftarrow 0$
5. **while** $i < m$ **do**
6.   **if** $b_{i+1} \in S$
7.   **then** **while** $b_{i+1} \in S$ **do**
8. $u_t \leftarrow \min\{k | k \in \Sigma \text{ and } k \notin S\}$
9. $t \leftarrow t+1$
10.   **while** $b_1 \in S(m)$ **do**
11. $u_k \leftarrow \min\{k | k \in \Sigma \text{ and } k \notin S \text{ and } k \neq b_{i+1}\}$
12. $t \leftarrow t+1$
13. $j \leftarrow 1$
14.   **while** $j < i+1$ **do**
15. $u_t \leftarrow b_j$
16. $t \leftarrow t+1$
17. $j \leftarrow j+1$
18. $u_t \leftarrow b_{i+1}$
19. $t \leftarrow t+1$
20. $i \leftarrow i+1$

Instead of a long proof of correctness of ANOMALY we explain how this algorithm generates U of the example used to disprove the conjecture. The input of the algorithm is $m = 5$, $M = 6$ and $q = (7,3,6,2,5)$. Lines 1-2 result the references 1, 2, 3, 4, 5, 6 and control state q = (2,3,4,5,6) in the small memory. Now line 3 results $t = 7$, line 4 gives $i = 0$ and we start the while cycle in line 5. Since $b_1 = 7$ is missing from the small memory, we add $u_7 = 7$ to U (lines 18-20) changing the control state to $q = (3, 4, 5, 6, 7)$, increment t and i and return to line 5 where we observe that $b_2 = 3$ is in the memory. Therefore continue with the while cycle in line 7 and add the minimal missing page to the reference string resulting control state (4,5,6,7,1). Since page 1 replaced b_2, we go to the next cycle in lines 10-12 and add the minimal missing pages differing from $b_2 = 3$ (that is 2, 4, 5 and 6) to U changing the control state to (1,2,4,5,6). Now this while cycle ends and using lines 13-17 we add 7 then using lines 18-20 add 3 to U ending in control state (4,5,6,7,3).

Now we return to line 5 and using lines 7-9 add 1 to U, then using lines 10-12 add 2, 4 and 5 to U changing the control state to (3,1,2,4,5). Now using lines 14-18 we add 7 and 3, then using lines 19-21 add 6 to U changing the control state to (4,5,7,3,6). Now lines 5-6 send us to lines 14-18 resulting reference $u_{22} = 2$ and control state (5,7,3,6,2).

We observe in lines 5-6 that b_5 is in the memory therefore we add 1 to U and so remove 2 from the memory. Now we continue U with 4 resulting the control state (3,6,2,1,4). Finally lines 14-18 add 7, 3, 6 and 2 and lines 19-21 add 5 to U implying the required control state.

Lemma 5 *Let n be an odd positive integer* ($n \geq 5$) and let $M = n-1$, $m = n-2$. *Then there exists a reference string R resulting anomaly ratio arbitrarily close to* $(n-1)/2$.

Proof. This assertion is a consequence of Lemma 3 and Lemma 4. $\square$

Main Theorem *For any large number L there exist parameters m, M, and R such that the anomaly ratio* $f_{\mathrm{FIFO}}(R, M)/f_{\mathrm{FIFO}}(R, m) > L$.

Proof. Let n be an odd integer satisfying $n > 2L+1$ (and $n \geq 5$). Then the parameters m, M and R in Lemma 5 result an anomaly ratio greater than L. $\square$

5 SUMMARY

If the memory parameters m and M are fixed, then the anomaly is bounded.

We suppose that the maximal anomaly occurs at cyclical reference strings, similar memory sizes, FIFO-like replacement in the large memory and MIN-like replacement in the small memory. If so then for fixed odd n our construction results the maximal possible anomaly ratio. We remark that there is a similar construction for even n resulting anomaly ratio arbitrarily close to $(n-2)/2$.

ACKNOWLEDGEMENT. The authors are indebted to Mihály Molnár, student of Eötvös University who using genetic algorithms discovered the first reference string resulting an anomaly ratio greater than 2.

REFERENCES

[1] P. Denning, Virtual memory. *Computing Surveys.* **2** (3) 153-189, (1980).

[2] A. Silberschatz, P. Galvin and G. Gagne, *Applied Operating System Concepts.* John Wiley and Sons, New York, (2000).

[3] L.A. Belady, A study of replacement algorithms for a virtual storage computer. *IBM Syst. J.* **5** (2) 78-101, (1965).

[4] L.A. Belady, R.A. Nelson and G.S. Shedler, An anomaly in space-time characteristics of certain programs running in paging machine. *Comm. ACM* **12** (1) 349-353, (1969).

[5] A.M. Iványi, On dumpling-eating giants. In: *Finite and Infinite Sets*, pp. 379-390 (Ed. by A. Hajnal, L. Lovász and V. T. Sós), North-Holland, Amsterdam, (1984).

[6] A.M. Iványi and R.L. Smelyanskiy, *Elements of Theoretical Programming* (in Russian). Moscow State University, Moscow, (1980).

[7] S. Roosta, *Parallel Procesing and Parallel Algorithms.* Springer-Verlag, New York, (1999).

[8] S.D. Mihnovskiy and N.Z Shor, Estimation of the number of page faults in paged virtual memory (in Russian). *Kibernetika* **1** (5) 18-20, (1965).

[9] J. Gecsei et al., Evaluation techniques for storage hierarchies. *IBM Syst. J.* **9** (2) 78-117, (1970).

[10] M. Arató, A note on optimal performance of page storage. *Acta Cybern.* (Szeged) **3** (1) 25-30, (1976/77).

[11] A. Benczúr, A. Krámli and J. Pergel, On the Bayesian approach to optimal performance of page hierarchies. *Acta Cybern.* (Szeged) **3** (2) 79-89, (1976/77).

[12] L.N. Korolev and A.M. Iványi, Classification of paging algorithms. In: *Problems of Interpretation of Experiments*, pp. 105-125 (Ed. by A.N.

Tihonov, P.N. Zaikin and V.Y. Galkin), Moscow State University, Moscow, (1975).

[13] T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, *Introduction to Algorithms* (Second edition). The MIT Press/McGraw Hill Book Company, Cambridge/Boston, (2001).