Open Journal of Modelling and Simulation, 2017, 5, 47-58
http://www.scirp.org/journal/ojmsi
ISSN Online: 2327-4026
ISSN Print: 2327-4018

Analysis of the Multi-Pivot Quicksort Process

Mahmoud Ragab[1], Beih El-Sayed El-Desouky[2], Nora Nader[2]

[1]Department of Mathematics, Faculty of Science, Al Azhar University, Cairo, Egypt
[2]Department of Mathematics, Faculty of Science, Mansoura University, Mansoura, Egypt
Email: meragab23@yahoo.mail, b_desouky@mans.edu.eg, sci_nora2012@yahoo.com

How to cite this paper: Ragab, M., El-Desouky, B.E.-S. and Nader, N. (2017) Analysis of the Multi-Pivot Quicksort Process. *Open Journal of Modelling and Simulation*, **5**, 47-58.
http://dx.doi.org/10.4236/ojmsi.2017.51004

Received: September 26, 2016
Accepted: January 9, 2017
Published: January 12, 2017

Abstract

In this paper, we study a new version from Dual-pivot Quicksort algorithm when we have some other number k of pivots. Hence, we discuss the idea of picking k pivots $i_1, i_2, \cdots, i_k$ by random way and splitting the list simultaneously according to these. The modified version generalizes these results for multi process. We show that the average number of swaps done by Multi-pivot Quicksort process and we present a special case. Moreover, we obtain a relationship between the average number of swaps of Multi-pivot Quicksort and Stirling numbers of the first kind.

Keywords

Quicksort, Convergence, Multi-Pivot Quicksort Process, Stirling Number of the First Kind

1. Introduction

Quicksort studied in many books such as [1] [2] and [3]. It is an exhaustively anatomize sorting algorithm and following the idea of divide-and-conquer on an input consisting of n items [4]. Quicksort used a pivot item to divide its input items into two partitions; the items in one sublist seem diminutive or identically tantamount to the pivot; the items in the other sublist seem more sizably voluminous than or equipollent to the pivot, after then it uses recursion to order these sublists. It is prominent that the input consists of n items with different keys in arbitrary order and the pivot is picked by just picking an item, and then on average Quicksort utilizes $2n\ln n + O(n)$ comparisons between items from the input. The Partial Quicksort algorithm analyzed by Ragab [5] [6] and [7] depends on the idea of the standard Quicksort. It uses a smart strategy to find the l smallest elements out of n distinct elements and sort them. Yaroslavskiy declared in 2009 that he had made some improvements for the Quicksort algorithm, the demand being drawn by experiments.

Yaroslavskiy's algorithm replaced the new standard Quicksort algorithm in Oracle's Java 7 runtime library. This algorithm uses two items as pivots to divide the items. If two pivots p_1 and p_2 such that $p_1 < p_2$ are used, the splitting step sublists the remaining $n-2$ items into three sublists, items more minute than or equipollent to p_1, items between p_1 and p_2, and items more sizably voluminous than or equipollent to p_2. Recursion is then applied to the three sublists. It came as a surprise that two pivots should avail, since in his thesis [8] Sedgewick had introduced and explained a Dual-pivot technique inferior to classical Quicksort. Hence, Hennequin in his thesis studied the general technique of using $k-1$ pivot items [2].

We analyze the limiting distribution of the number of swaps needed by the duality process is proposed. It is known to be the unique fixed point of a certain distributional transformation T with zero mean and finite variance. Depending on the results of [1] and [9], we analyze the Multi-pivot Quicksort when we selected k pivots and we study the relationship with Striling numbers of the first kind.

2. Multi -Pivot Quicksort

Later, many researchers has received the interest of the visualization of multi-pivot Quicksort in accordance with Yaroslavskiy proposed the duality pivot process which outperforms standard Quicksort by Java JVM. After that, this algorithm has been explained in terms of comparisons and swaps by Wild and Nebel [10].

A normal expansion of duality process would be to have some other number k of pivots. Hence, we cogitation the approximation of pick k pivots $i_1, i_2, \cdots, i_k$ by random way and splitting the list simultaneously according to these. let a random permutation of the list $\{1, 2, \cdots, n\}$ be given to be ordered using this variant, with all the $n!$ substitution. The k rightmost item are picked as pivots are compared to each other and interchange, if they are out of order.

There are $n-k$ items are swaps to the pivots and the list is splitted to $k+1$ sublists. The partition step can be worked as follows. We compare the leftmost item to pivot which chosen by random way; if this pivot is bigger than it, it is compared with another pivot which was smaller than the first pivot. Otherwise it is swaped with a bigger pivot (to the right) and after a series of number of swaps are inserted to its place between any two pivots, or to the left of the smallest pivot or to the right of the biggest pivot. We continue with the same technique, until all items are examined.

Each item of the $n-k$ items swaps with the pivots by binary tree, first each item is swaps with the median of the sorted list of the pivots. If it is compared with the first element, otherwise is compared with the third element and after a collection of swaps is inserted to its placement.

For the pervious process, there are k sublists. If we let that the input is a random permutation $\{U_{n_1}, \cdots, U_{n_k}\}$ of $\{1, 2, 3, \cdots, n\}$.

We assume that $k \geq 1$ be an integer. The method "k-pivot quicksort" performs as follows:

As long as $n \leq k$ then sort the input directly. When $n > k$, order the first k items such that $U_{n_1} < U_{n_2} < \cdots < U_{n_k}$ and set $p_1 = U_{n_1}, \cdots,$ $p_k = U_{n_k}$. In the splitting step, the remaining $n-k$ items are divided to $k+1$ sets $L_0, L_1, \cdots, L_k$ where an item x belongs to set L_h as long as $p_h < x < p_{h+1}$. The sets $L_0, L_1, \cdots, L_k$ are then ordered recursively. Assume that $k \geq 1$ be fixed. As for duality Quicksort process, if we assume that $S_{n,k}$ and P_n give the random variables that count the swaps required to sort n items when we select k pivot items, uniformly selected from the list and partitioning respectively. The total number of swaps needed by Multi-pivot Quicksort sorting inputs given by

$$S_{n,k} \overset{d}{=} P_n + S_{a_{0,k}} + S^*_{a_{1,k}} + \cdots + \overline{S}_{a_{k,k}}, \tag{1}$$

where $S_{a_0,k}$ random variable that count the number of swaps made for sort the items smaller than first pivot $S^*_{a_{1,k}}$ denote the number of swaps need to order the items between first pivot and the second pivot. $\overline{S}_{a_k,k}$ denote the number of swaps need to order the items between $k-1$ pivot and the k pivot. The random variables $S_{a_0,k}$, $S^*_{a_1,k}, \tilde{S}_{a_2,k}$ and $\overline{S}_{a_{k,k}}$ have the same distribution and independent of $U_{n_1}, U_{n_2}, \cdots, U_{n_k}$ and $\underset{=}{d}$ means the equality in distribution.

The average number of swaps done by the multi algorithm applied to an list of n items by k-pivot Quicksort given by the following recurrence

$$E\left(S_{n,k}\right) \overset{d}{=} E\left(P_{n,k}\right) + \frac{1}{\binom{n}{k}} \sum_{a_0+\cdots+a_k=n-k} E\left(S_{a_0,k}\right) + E\left(S^*_{a_1,k}\right) + \cdots + E\left(\overline{S}_{a_{k,k}}\right). \tag{2}$$

$$E\left(S_{n,k}\right) \overset{d}{=} T(n,k) + \frac{1}{\binom{n}{k}} \sum_{U_{n_1}} \sum_{U_{n_1}} \cdots \sum_{U_{n_k}} E\left(S_{U_{n_1}-1,k}\right) + E\left(S^*_{U_{n_2}-U_{n1}-1,k}\right) + \cdots + E\left(\overline{S}_{n-U_{n_k},k}\right),$$

where $U_{n_1} < U_{n_2} < \cdots < U_{n_k}$ refers to the pivots in increasing order, see [11] and [12].

Let $T(n,k) = \overline{c}(k)n + \overline{d}(k)$ be the expected value of a "toll function" during the first recursive call, where $\overline{c}(k)$ and $\overline{d}(k)$ are constants and $E\left(S_{U_{n_1}-1,k}\right)$ denotes the average number of swaps for ordering the sublist of $U_{n_1}-1$ items less than U_{n_1} by the Multi-Pivot Quicksort on k pivots to simplify the relation by noting that the pivots are selected by the random way and the sums are equal,

$$\begin{aligned} E\left(S_{n,k}\right) &= \overline{c}(k)n + \overline{d}(k) \\ &\quad + \frac{1}{\binom{n}{k}} \sum_{U_{n_1}=1}^{n-k+1} \sum_{U_{n_1}=U_{n_1}+1}^{n-k+2} \cdots \sum_{U_{n_k}=U_{n_k}+1}^{n} E\left(S_{U_{n_1}-1,k}\right) + E\left(S^*_{U_{n_2}-U_{n_1}-1,k}\right) \ldots + E\left(\overline{S}_{n-U_{n_k},k}\right) \\ &= \overline{c}(k)n + \overline{d}(k) + \frac{1}{\binom{n}{k}} \sum_{\ell=1}^{n-k+1} (k+1) \binom{n-\ell}{k-1} E\left(S_{\ell-1,k}\right). \end{aligned}$$

By collecting terms with a common factor $E\left(S_{\ell,k}\right)$, when $1 \le \ell \le n-k+1$. Fix $\ell \in \{1, 2, \cdots, n-k+1\}$. There are $k+1$ methods of picking $j \in \{1, 2, \cdots, n-k+1\}$ with $a_j = \ell$.

$$E\left(S_{n,k}\right) = \overline{c}(k)n + \overline{d}(k) + \frac{1}{\binom{n}{k}} \sum_{\ell=1}^{n-k+1} (k+1) \binom{n-\ell}{k-1} E\left(S_{\ell-1,k}\right). \tag{3}$$

Multiplying both sides by $\binom{n}{k}$, the recurrence relation becomes

$$\binom{n}{k}E\left(S_{n,k}\right)=\binom{n}{k}\left[\bar{c}(k)n+\bar{d}(k)\right]+(k+1)\sum_{\ell=1}^{n-k+1}\binom{n-\ell}{k-1}E\left(S_{\ell-1,k}\right),$$

multiplying by z^n and summing over n., hence we get the generating function for the average number of swaps [10]. Let $a_n=E\left(S_{n,k}\right)$ and consider the generating function $g(z)=\sum_{n=0}^{\infty}a_n z^n$

$$\sum_{n=0}^{\infty}\binom{n}{k}a_n z^n=\sum_{n=0}^{\infty}\binom{n}{k}\left[\bar{c}(k)n+\bar{d}(k)\right]z^n+(k+1)\sum_{n=0}^{\infty}\left(\sum_{\ell=1}^{n}\binom{n-\ell}{k-1}a_{\ell-1}\right)z^n. \tag{4}$$

We find that

$$\begin{aligned}\sum_{n=0}^{\infty}\binom{n}{k}a_n z^n&=\frac{1}{k!}\sum_{n=0}^{\infty}n(n-1)\cdots(n-k)a_n z^n\\&=\frac{z^k g^{(k)}(z)}{k!}.\end{aligned}$$

Such that $g^{(k)}(z)$ gives the *k*-*th* order derivative of $g(z)$. In the right-hand side of Equation (4), the first sum becomes

$$\begin{aligned}\sum_{n=0}^{\infty}\binom{n}{k}\left[\bar{c}(k)n+\bar{d}(k)\right]z^n&=\frac{z^k}{k!}\left(\sum_{n=0}^{\infty}\left[\bar{c}(k)n+\bar{d}(k)\right]z^n\right).\\&=\frac{z^k\left(\bar{c}(k)(z+k)+\bar{d}(k)(1-z)\right)}{(1-z)^{k+2}}\end{aligned}$$

The recurrence becomes as follows

$$\begin{aligned}\frac{g^{(k)}(z)z^k}{k!}&=\sum_{n=0}^{\infty}\binom{n}{k}\left[\bar{c}(k)n+\bar{d}(k)\right]z^n+g(z)(k+1)\sum_{n=0}^{\infty}\binom{n-1}{k-1}z^n\\&=\sum_{n=0}^{\infty}\binom{n}{k}\left[\bar{c}(k)n+\bar{d}(k)\right]z^n\binom{n}{k}+(k+1)g(z)\left(\frac{z}{1-z}\right)^k\\&=\frac{z^k\left(\bar{c}(k)(z+k)+\bar{d}(k)(1-z)\right)}{(1-z)^{k+2}}+(k+1)g(z)\left(\frac{z}{1-z}\right)^k,\end{aligned} \tag{5}$$

In the right -hand side of Equation (5). The first sum becomes in this form because it may be easily explained by mathematical induction that the *k*-*th* order derivative of

$$\sum_{n=0}^{\infty}\left(\bar{c}(k)n+\bar{d}(k)\right)z^n=\frac{\bar{c}(k)z+\bar{d}(k)(1-z)}{(1-z)^2},$$

is

$$\frac{\left(\bar{c}(k)(z+k)+\bar{d}(k)(1-z)\right)}{(1-z)^{k+2}}.$$

The recurrence is converted to the following differential equation [13]:

$$\frac{g^{(k)}(z)z^k}{k!}=\frac{z^k\left(\bar{c}(k)(z+k)+\bar{d}(k)(1-z)\right)}{(1-z)^{k+2}}+(k+1)g(z)\left(\frac{z}{1-z}\right)^k. \tag{6}$$

Multiplying by $\left[\frac{z}{1-z}\right]^{-k}$, the previous Equation (6) is transformed to

$$\frac{g^{(k)}(z)(1-z)^k}{k!}=\frac{\left(\bar{c}(k)(z+k)+\bar{d}(k)(1-z)\right)}{(1-z)^k}+(k+1)g(x).$$

This differential equation is a Cauchy-Euler equation [14]. We change variables $x=1-z$, it is $h(x)=g(1-x)$

$$(-1)^k x^k h^k(x)=k!\left(\frac{\bar{c}(k)(1-x+k)+\bar{d}(k)x}{x^2}+(k+1)h(x)\right). \qquad (7)$$

By using the differential operator Θ to solve the previous differential Equation (7) which is defined by

$$\Theta\left(h((x))\right):=xh'(x),$$

and using the mathematical induction we find that at $n=1$

$$L.H.S=\Theta\left(h(x)\right),$$

and

$$R.H.S=x\dot{h}(x).$$

We find

$$L.H.S=R.H.S,$$

the relation holds at $n=1$. We assume the relation holds at $n=k$

$$\binom{\Theta}{k}h(x)=\frac{x^k h^k(x)}{k!},$$

at $n=k+1,$ we find

$$\begin{aligned} L.H.S&=\binom{\Theta}{k+1}h(x)\\ &=\frac{\Theta!}{(k+1)!(\Theta-k-1)!}h(x)\\ &=\frac{x^k h^k(x)}{k!}. \end{aligned}$$

So, it is easy to find the relation is satisfied for all values of k.

$$\binom{\Theta}{k}h(x)=\frac{x^k h^k(x)}{k!}.$$

When we apply the operator Θ, our relation seems in the form

$$\left((-1)^k\Theta(\Theta-1)\cdots(\Theta-k+1)-(k+1)!\right)h(x)=\frac{k!\left(\bar{c}(k)(1-x+k)+\bar{d}(k)x\right)}{x^2},$$

$$P_k(\Theta)h(x)=\frac{k!\left(\bar{c}(k)(1-x+k)+\bar{d}(k)x\right)}{x^2}, \qquad (8)$$

where $P_k(\Theta)$ is called as the initial polynomial and is given as follows, see [15],

$$P_k(\Theta)=(-1)^k(\theta)_k-(k+1)!,$$

where $(\Theta)_k=\Theta(\Theta-1)\cdots(\Theta-k+1)$ with $k\geq 0$, denotes the falling factorial. If we use the fundamental theorem of algebra which proposed that a polynomial of degree n

has n complex roots with multiplicities. Notice that −2 is constantly a simple root because,

$$\begin{aligned} P_k(-2) &= (-1)^k(-2)(-3)\cdots(-2+k+1)-(k+1)! \\ &= (-1)^k(-2)(-3)\cdots(k-1)-(k+1)! \\ &= (-1)^{2k}(k+1)!-(k+1)! = 0. \end{aligned}$$

And we get

$$\dot{P}_k(-2) = (k+1)!\sum_{j=0}^{k-1}\frac{1}{-2-j} = -(k+1)!(H_{k+1}-1) < 0.$$

Setting $r_k = -2$ and the residual roots be $r_1, r_2, \cdots, r_{k-1}$, see [16]. Our polynomial be in the form

$$(\Theta-r_1)(\Theta-r_2)\cdots(\Theta-r_{k-1})(\Theta+2)h(x) = \frac{k!\left(\bar{c}(k)(1-x+k)+\bar{d}(k)x\right)}{x^2}.$$

This differential equation can be written as

$$S_{k-1}(\theta)(\theta+2)h(z) = \frac{k!\left(\bar{c}((k)(1-z+k)+\bar{d}(k)z\right)}{z^2}, \quad (9)$$

where $S_{k-1}(\Theta) = (\Theta-r_1)(\Theta-r_2)\cdots(\Theta-r_{k-1})$ to solve our differential equation, we assume that there are two functions $h_1(z)$ and $h_2(z)$ where

$$h_1(z)+h_2(z) = h(z).$$

Then

$$\begin{aligned} &(\Theta-r_1)(\Theta-r_2)\cdots(\Theta-r_k)(\Theta+2)\left(h_1(z)+h_2(z)\right) \\ &= \frac{k!\left(\bar{c}(k)(1-z+k)+\bar{d}(k)z\right)}{z^2}. \\ &= \frac{\bar{c}(k)(k+1)!}{z^2}+\frac{\left(\bar{d}(k)-\bar{c}(k)\right)k!}{z}, \end{aligned}$$

By using the property of linearity of differential operator

$$(\Theta-r_1)(\Theta-r_2)\cdots(\Theta-r_{k-1})(\Theta+2)h_1(z) = \frac{\bar{c}(k)(k+1)!}{z^2}.$$

$$(\Theta-r_1)(\Theta-r_2)\cdots(\Theta-r_{k-1})(\Theta+2)h_2(z) = \frac{\left(\bar{d}(k)-\bar{c}(k)\right)k!}{z},$$

if we apply k times the solution, we get

$$h_1(z) = \frac{\bar{c}(k)(k+1)!}{(-2-r_1)(-2-r_2)\cdots(-2-r_{k-1})}\frac{\log z}{z^2}+_{i=1}^{k} d_i z^{r_i}. \quad (10)$$

$$h_2(z) = \frac{k!}{(1-r_1)(1-r_2)\cdots 1}\frac{\left(\bar{d}(k)-\bar{c}(k)\right)}{z}+\sum_{i=1}^{k} w_i z^{r_i}, \quad (11)$$

where d_i and w_i are constants of integration. Note that

$$P_K(\Theta) = (\Theta+2)S_{k-1}(\Theta).$$

Therefore, to evaluate $S_{k-1}(-2)$, we find

$$
\begin{aligned}
S_{k-1}(-2) &= \dot{P}_K(-2), \\
&= -\binom{-2}{k}\sum_{j=0}^{k-1}\frac{1}{-2-j}-1. && (12) \\
&= -(k+1)!\left(H_{k+1}-1\right). && (13)
\end{aligned}
$$

Moreover

$$P_k(-1) = -kk!.$$

Combining both solutions,

$$h(z) = \frac{-\bar{c}(k)}{\left(H_{k+1}-1\right)}\frac{\log(z)}{z^2} + \frac{1}{k}\frac{\left(\bar{c}(k)-\bar{d}(k)\right)}{z} +_{i=1}^{k} e_i z^{r_i}, \tag{14}$$

such that $e_i = w_i + d_i$. To solve this system of of equations, we should calculate the constants of integration

$$h(1) = \dot{h}(1) = \cdots = h^{(k-1)}(1) = 0.$$

In terms of series;

$$g(x) = \frac{\bar{c}(k)}{\left(H_{k+1}-1\right)}\sum_{n=0}^{\infty}\left((n+1)H_n - n\right)x^n + \sum_{n=0}^{\infty}\sum_{i=1}^{k} e_i(-1)^n\binom{r_i}{n}x^n + \frac{\bar{c}(k)-\bar{d}(k)}{k}\sum_{n=0}^{\infty}x^n. \tag{15}$$

The third sum of Equation (15) collects to the solution a stationary contribution. Furthermore, the root $k+1$ when k is even, participates a constant and the root $r_k = -2$, collects $s_k(n+1)$, with $s_k \in R$. Eliciting the coefficients, the average number of swaps for Multi-pivot Quicksort is

$$a_n = \frac{\bar{c}(k)}{H_{k+1}-1}\left((n+1)H_n - n\right) + O(n) \tag{16}$$

The number of methods to permute a list of n items into k cycles counted by the Stirling numbers of the first kind $s(n,k)$, see [17].

We show the relation between the number of swaps done by the multi Quicksort process and Sirling number of the first kind. Form Equation (17) we assume that $a_n = E\left(S_{n,k}\right)$ and consider the generating function $g(z) = \sum_{n=0}^{\infty} a_n z^n$

$$\sum_{n=0}^{\infty}\binom{n}{k}a_n z^n = \sum_{n=0}^{\infty}\binom{n}{k}T(n,k)z^n + (k+1)\sum_{n=0}^{\infty}\left(\sum_{\ell=1}^{n}\binom{n-\ell}{k-1}a_{\ell-1}\right)z^n. \tag{17}$$

The relation is converted to a *k-th* order differential equation

$$\frac{g^{(k)}(z)z^k}{k!} = \sum_{n=0}^{\infty}\left[\binom{n}{k}\left(\bar{c}(k)n + \bar{d}(k)\right) + (k+1)\binom{n-1}{k-1}g(z)\right]z^n.$$

This differential equation is a Cauchy-Euler equation. We use the deferential operator Θ for the solution of the differential equation. It is defined as follows

$$\Theta\left(g(z)\right) := z\dot{g}(z),$$

also, by induction

$$\binom{\Theta}{k}g(z)=\frac{z^k g^k(z)}{k!}$$

applying the operator Θ ,our equation becomes

$$\binom{\Theta}{k}g(z)=\sum_{n=0}^{\infty}\left[\binom{n}{k}\left[\bar{c}(k)n+\bar{d}(k)\right]+(k+1)\binom{n-1}{k-1}g(z)\right]z^n$$

$$\Theta(\Theta-1)(\Theta-2)\cdots(\Theta-k+1)g(z)$$

$$=\sum_{n=0}^{\infty}\left[\binom{n}{k}\left[\bar{c}(k)n+\bar{d}(k)\right]+(k+1)\binom{n-1}{k-1}g(z)\right]z^n$$

$$(\Theta)_k\, g(z)=k!\sum_{n=0}^{\infty}\left[\binom{n}{k}\left[\bar{c}(k)n+\bar{d}(k)\right]+(k+1)\binom{n-1}{k-1}g(z)\right]z^n,$$

where $(\Theta)_k=\Theta(\Theta-1)(\Theta-2)\cdots(\Theta-k+1)$ is falling factorial, see [18].

$$(\Theta)_k\, g(x)=\sum_{n=0}^{\infty}\left[k!\frac{n!}{k!(n-k)!}\left(\bar{c}(k)n+\bar{d}(k)\right)+k!(k+1)\frac{(n-1)!}{(k-1)!(n-1-k+1)!}g(x)\right]z^n$$

$$(\Theta)_k\, g(z)=\sum_{n=0}^{\infty}(n)_k\left[\bar{c}(k)n+\bar{d}(k)\right]z^n+\sum_{n=0}^{\infty}k(k+1)(n-1)(n-2)\cdots(n-k+1)g(z)z^n$$

$$=\sum_{n=0}^{\infty}(n)_k\left[\bar{c}(k)n+\bar{d}(k)\right]z^n+\sum_{n=0}^{\infty}k(k+1)s(n-1,k-1)g(z)z^n.$$

where $s(n-1,k-1)=(n-1)(n-2)\cdots(n-k+1)$ is Stirling numbers of the first kind, see [19]. we use the relation

$$(x)_n=\sum_{k=0}^{n}s(n,k)x^k, \text{where } s(0,0)=\text{land } s(0,k)=0 \text{ for } k>0,$$

hence

$$\sum_{r=0}^{k}s(k,r)\Theta^r\cdot g(z)=\sum_{n=0}^{\infty}(n)_k\left[\bar{c}(k)n+\bar{d}(k)\right]z^n$$
$$+\sum_{n=0}^{\infty}k(k+1)s(n-1,k-1)g(z)z^n.$$

$$\sum_{n=0}^{\infty}a_n\sum_{r=0}^{n}s(n,r)\Theta^r\cdot z^n=\sum_{n=0}^{\infty}\left[\bar{c}(k)n+\bar{d}(k)\right]\left[\sum_{l=0}^{k}s(k,l)n^l\right]z^n$$
$$+\sum_{n=0}^{\infty}k(k+1)s(n-1,k-1)\left(\sum_{n=0}^{\infty}a_nz^n\right)z^n$$

$$\sum_{n=0}^{\infty}a_n\sum_{r=0}^{n}s(n,r)\Theta^r\cdot z^n=\sum_{n=0}^{\infty}\left[\bar{c}(k)n+\bar{d}(k)\right]\left[\sum_{l=0}^{k}s(k,l)n^l\right]z^n$$
$$+{}_{n=0}^{\infty}\sum_{n=0}^{\infty}\sum_{r=0}^{n}\left[k(k+1)s(r-1,k-1)a_{n-r}\right]z^n,$$

by equality the coefficients we obtain

$$a_n\sum_{r=0}^{n}s(n,r)\Theta^r=\sum_{n=0}^{\infty}\left[\bar{c}(k)n+\bar{d}(k)\right]\left[\sum_{l=0}^{k}s(k,l)n^l\right]+\sum_{r=0}^{n}\left[k(k+1)s(r-1,k-1)a_{n-r}\right]$$

$$a_n=\frac{1}{\sum_{r=0}^{n}s(n,r)\Theta^r}\sum_{n=0}^{\infty}\left[\bar{c}(k)n+\bar{d}(k)\right]\left[\sum_{l=0}^{k}s(k,l)n^l\right]+\sum_{r=0}^{n}\left[k(k+1)s(r-1,k-1)a_{n-r}\right]$$

3. Quicksort

In this section we show the average number of swaps needed by the Quicksort is a particular case form the public case of the multi-pivot Quicksort [20]. For $k=1$

$$S_n \overset{d}{=} P_n + S_{U_{n_1}-1} + S^*_{n-U_{n_1}}, \tag{18}$$

where P_n give the random variables which counting the number of swaps needed for splitting the list of n items, such that the classical algorithm is applied to an list of n different items [5]. We find that $S_0 = S_1 = 0$ such that if $n \geq 2$, the following recurrence holds. We find the average number of swaps done by the Quicksort from Equation (2) we find at $k=1$ the equation becomes

$$E(S_n) \overset{d}{=} E(P_n) + E\left(S_{U_{n_1}-1}\right) + E\left(S^*_{n-U_{n_1}}\right), \tag{19}$$

Assume that if we need to sort list of n of different items, where their positions in the list are counted from left to right by $1,2,\cdots,n$ [6]. First, the item at position 1 compared with the pivot. The number of items which are bigger than pivot and were animated during split operation is

$$\frac{n-k}{n-1}\cdot(k-1).$$

Subsequently, we consider as well that pivots are uniformly picked and noticing that we have to number the final swap with the pivot at the end of split operation [21], we get

$$\sum \frac{(n-k)(k-1)}{n(n-1)} + 1 = \frac{n}{6} + \frac{2}{3}.$$

So, we find the toll function given by

$$T(n) = \overline{c}n + \overline{d} = \frac{n}{6} + \frac{2}{3}.$$

We find $\overline{c} = \frac{1}{6}$, $\overline{d} = \frac{2}{3}$. So the recurrence becomes

$$E(S_n) \overset{d}{=} \frac{n}{6} + \frac{2}{3} + \frac{1}{n}\sum_{U_{n_1}=1}^{n} E\left(S_{U_{n_1}-1}\right) + E\left(S^*_{n-U_{n_1}}\right), \tag{20}$$

$$E(S_n) \overset{d}{=} \frac{n}{6} + \frac{2}{3} + \frac{2}{n}\sum_{U_{n_1}=1}^{n} E\left(S_{U_{n_1}-1}\right). \tag{21}$$

We solve this recurrence relation by transforming into a differential equation. First multiply both sides by n

$$nE(S_n) = n\left(\frac{n+4}{6}\right) + 2\sum_{U_{n_1}=1}^{n} E\left(S_{U_{n_1}-1}\right).$$

Let $a_n = E(S_n)$

$$na_n = n\left(\frac{n}{6} + \frac{2}{3}\right) + 2\sum_{U_{n_1}=1}^{n} a_{U_{n_1}-1},$$

multiplying by z^n and summing over n, so as to get the generating function for the average number of swaps consider the generating function $g(z)=\sum_{n=0}^{\infty} a_n z^n$

$$\sum_{n=0}^{\infty} n a_n z^n = \sum_{n=0}^{\infty} n\left(\frac{2n+3}{6}\right) z^n + 2\sum_{n=0}^{\infty}\sum_{U_{n_1}=1}^{n} a_{U_{n_1}-1} z^n$$

$$z\dot{g}(z) = \sum_{n=0}^{\infty} n\left(\frac{n}{6}+\frac{2}{3}\right) z^n + 2g(z)\left(\frac{z}{1-z}\right)$$

$$= \frac{z\left[\frac{1}{6}(z+1)+\frac{2}{3}(1-z)\right]}{(1-z)^3} + 2g(z)\left(\frac{z}{1-z}\right) \tag{22}$$

$$= \frac{z\left(\frac{1}{6}z-\frac{2}{3}z+\frac{1}{6}-\frac{2}{3}\right)}{(1-z)^3} + 2g(z)\left(\frac{z}{1-z}\right).$$

Multiplying by $\left[\frac{z}{1-z}\right]^{-1}$, the differential equation is simplified to

$$(1-z)\dot{g}(z) = \frac{\frac{1}{6}(z+1)+\frac{2}{3}(1-z)}{(1-z)^2} + 2g(z).$$

We can solve this differential equation using basic principles

$$(1-z)\dot{g}(z) - 2g(z) = \frac{\frac{5}{6}-\frac{1}{2}z}{(1-z)^2} \tag{23}$$

This differential equation is a Cauchy-Euler equation [22]. We change variables $x=1-z$, it is $h(x)=g(1-x)$

$$(-1)x\dot{h}(x) = \left(\frac{\frac{1}{6}(1-x+1)+\frac{2}{3}x}{x^2} + 2h(x)\right),$$

we use the differential operator Θ to solve the differential equation which defined as follows

$$\Theta(h(x)) := x\dot{h}(x)$$

applying the operator Θ, our equation becomes

$$((-1)\Theta-2)h(x) = \frac{\frac{1}{6}(1-x+1)+\frac{2}{3}x}{x^2}$$

$$P_1(\Theta)h(x) = \frac{-\frac{1}{6}x+\frac{2}{3}x}{x^2}$$

$$P_1(\Theta)h(x) = \frac{\frac{1}{2}x}{x^2}$$

$$P_1(\Theta)h(x) = \frac{1}{2x}$$

and applying the pervious technique we find the solution of the differential equation given by

$$h(z)=\frac{\frac{1}{6}}{\left(\frac{2}{3}-1\right)}\frac{\log(z)}{z^{2}}+\frac{\left(\frac{1}{6}-\frac{2}{3}\right)}{z}+e_{1}z, \tag{24}$$

where e_1 is constant of integration. In terms of series

$$g(x)=\frac{\frac{1}{6}}{\left(H_{2}-1\right)}\sum_{n=0}^{\infty}\left(\left(n+1\right)H_{n}-n\right)x^{n}+\sum_{n=0}^{\infty}e_{1}\left(-1\right)^{n}\binom{r}{n}x^{n}+\frac{-1}{2}\sum_{n=0}^{\infty}x^{n}. \tag{25}$$

Extracting the coefficients, the expected number of swaps for Multi-pivot Quicksort is

$$g(x)=\frac{1}{3}(n+1)H_{n}+o(n) \tag{26}$$

4. Conclusion

We study a new version from Dual-pivot Quicksort algorithm when we have some other number k of pivots. Hence, we discuss the idea of picking k pivots $i_1, i_2, \cdots, i_k$ by random way and splitting the list simultaneously according to these. Moreover, we derive a generalization of this result for multi process. We show that the average number of swaps done by Multi-pivot Quicksort process and we present a special case. Furthermore, we present the relationship between the average number of swaps of Multi-pivot Quicksort and Stirling numbers of the first kind.

Acknowledgements

We thank the Editor and the referee for their comments.

References

[1] Ragab, M., El-Desouky, B.S. and Nader, N. (2016) On the Convergence of the Dual-Pivot Quicksort Process. *Open Journal of Modelling and Simulation*, **4**, 1-15. https://doi.org/10.4236/ojmsi.2016.41001

[2] Hennequin, P. (1991) Analyse en moyenne d'algorithmes: Tri rapide et arbres de recherche. Ph.D. Thesis, Ecole Politechnique, Palaiseau.

[3] Sedgewick, R. and Flajolet, P. (1996) An Introduction to the Analysis of Algorithms. Addison-Wesley, Longman, 1-492.

[4] Hoare, C.A.R. (1962) Quicksort. *Computer Journal*, **5**, 10-15. https://doi.org/10.1093/comjnl/5.1.10

[5] Ragab, M. (2011) Partial Quicksort and Weighted Branching Processes. Ph.D. Thesis, 28-35.

[6] Ragab, M. and Rosler, U. (2014) The Quicksort Process. *Stochastic Processes and Their Applications*, **2**, 1036-1054. https://doi.org/10.1016/j.spa.2013.09.014

[7] Ragab, M. (2015) On the Quicksort Algorithm and Its Related Process. *Journal of Mathematical Modeling and Operations Research*, 13-30.

[8] Sedgewic, R.K. (1975) Quicksort. Ph.D. Thesis, Garland Publishing.

[9] Iliopoulos, V. and Penman, D.P. (2012) Dual Pivot Quicksort. *Discrete Mathematics, Algorithms and Applications*, **4**, Article ID: 1250041. https://doi.org/10.1142/S1793830912500413

[10] Wild, S. and Nebel, M.E. (2012) Average Case Analysis of Java 7's Dual Pivot Quicksort. *Proceedings of the* 20*th European Symposium on Algorithms* (*ESA*'12), 825-836. https://doi.org/10.1007/978-3-642-33090-2_71

[11] Iliopoulos, V. (2013) Quicksorting on Multiple Pivots and a Vandermonde Matrix. Seminar in the Department of Mathematical Sciences, University of Essex, Colchester.

[12] Iliopoulos, V. (2013) The Quicksort Algorithm and Related Topics. PhD Thesis, Department of Mathematical Sciences, University of Essex, Colchester. http://repository.essex.ac.uk/13266

[13] Regnier, M. (1989) A Limiting Distribution for Quicksort. *RAIRO—Theoretical Informatics and Application*, **23**, 335-343.

[14] Roesler, U. (1992) A Fixed Point Theorem for Distributions. *Stochastic Processes and Their Applications*, **42**, 195-214. https://doi.org/10.1016/0304-4149(92)90035-O

[15] Rosler, U. (2001) On the Analysis of Stochastic Divide and Conquer Algorithms. *Algorithmica*, **29**, 238-261. https://doi.org/10.1007/BF02679621

[16] Fill, J.A. and Janson, S. (2001) Approximating the Limiting Quicksort Distribution. *Random Structures Algorithms*, **19**, 376-406. https://doi.org/10.1002/rsa.10007

[17] El-Desouky, B.S. and Cakic, N.P. (2011) Generalized Higher Order Stirling Numbers. *Mathematical and Computer Modelling*, **54**, 2848-2857. https://doi.org/10.1016/j.mcm.2011.07.005

[18] El-Desouky, B.S., Cakic, N.P. and Mansour, T. (2010) Modified Approach to Generalized Stirling Numbers via Differential Operators. *Applied Mathematics Letters*, **23**, 115-120. https://doi.org/10.1016/j.aml.2009.08.018

[19] El-Desouky, B.S, El-Bedwehy, N., Mustafa, A. and Menem, F. (2014) A Family of Generalized Stirling Numbers of the First Kind. *Applied Mathematics*, **5**, 1573-1585. https://doi.org/10.4236/am.2014.510150

[20] Fill, J.A. and Janson, S. (2004) The Number of Bit Comparisons Used by Quicksort: An Average-Case Analysis. *ACM-SIAM Symposium on Discrete Algorithms*, New Orleans, 11-13 January 2004, 300-307.

[21] Knuth, D.E. (1973) The Art of Computer Programming, Volume III: Sorting and Searching. Addison-Wesley, Upper Saddle River.

[22] El-Desouky, B.S. (2011) Generalized String Numbers of the First Kind: Modified Approach. *Journal of Pure and Mathematics Advances and Applications*, **5**, 43-59.