Transactions on Engineering and Computer Science

Review| Vol 4 Iss 1

Bounding of 3-Partitioning Algorithm for N_K Set

Osama Y. Abu Tammam*

Palestine Technical University-Kadoorie, Tulkarm, Palestine

***Corresponding author:** Abu Tammam OY, Department of Mathematical Modeling, Palestine Technical University-Kadoorie, Tulkarm, Palestine; E-mail: osamaabutammam[AT]hotmail.com

Received: June 04, 2023; **Accepted**: September 12, 2023; **Published:** September 30, 2023

Abstract

Bounding of 3 partition algorithms decrease the time period of applying the algorithm for N_K set depending on k value and required summation which generate the bounds of the algorithm. The flow chart of partitioning algorithm is constructed and simulated through Matlab. Matlab results are compared between bounded and un-bounded algorithm.

Keywords: ***Bounding; 3 Partition Algorithm; Matlab; N_K Set; un-bounded.***

1. Introduction

Number partitioning is one of the classical NP-hard problems of combinatorial optimization. It has applications in areas like public key encryption and task scheduling [1]. In computer science, Number partitioning problems (NPP) are classified as a basic NP-hard problem. A spin-glass problem is another formulation for NPP [3].

Rule Knowledge bases simplification of expert systems is the key to improve the computational performance, because the complexity of queries depends on the knowledge base length. While shorter representation of a given knowledge base increases the efficiency of computational performance and reduces the memory requirements of expert systems [2]. Phase transition can be observed in NPP. Phase transition is a characteristic that can't be proven, but observed. It is described as a phenomenon of abnormal behavior, where size increment of the input does not essentially make the problem harder [4].

The authors, in their paper, have illustrated how phase transition behavior can be used to study heuristics. By means of an annealed theory, they defined a parameter that measures the " constrained ness " of an ensemble of number

Citation: Abu Tammam OY. Bounding of 3-Partitioning Algorithm for N_K Set. Trans Eng Comput Sci. 2023;4(1):129.

partitioning problems. Contrary to a conjecture of Y. Fu (Fu, 1989) a phase transition occurs at a critical value of this parameter [5].

The authors, in their paper, have shown how a limited amount of enumeration allows to improve considerably the worst-case performance of approximate algorithms for the Partition problem [6].

The proposed algorithm in this paper is checking the elements within a boundaries instead of covering all of N_K set elements (represented below).

$$N_K = \{1,2,3,4,5,\ldots\ldots\ldots\ldots,k\} \quad (1)$$

2. Generating the Bounds for 3 Partition Algorithm

The boundaries of the three elements in 3 partition algorithms are noticed while working on it and using some mathematical logic. The boundaries depend on the required summation and maximum value in N_K set strongly. The researchers in [7] have concluded boundaries which is different from the bounds in this paper, while the basic idea of applying the bounds was previously introduced in many researches. The bounds are modified in this study and the final results are compared with the basic algorithm which has no bounds.

The Bounds of the 3 elements which have to be considered in our algorithm process is shown below for each element.

The exact maximum and minimum bounds for the first element which have to be checked in the three-partition algorithm can be written as shown in equation (1.1)

$$M1\,/2 \;<\; X1 \;\leq\; M1 \qquad (1.1)$$
$$M1 = \; sum\_req1 - (s1 + s2) \qquad (1.2)$$

Where

$M1$ is considered as maximum bound for the first element which have to be checked in the algorithm.

Sum_req1 is the required summation for the three elements.

$s1$ is the minimum value of element in N_K set and it equals to " 1".

$s2$ is the second minimum valued in N_K set and it equals to "2".

$X1$ is the elements which can considered as first element of our algorithm.

The maximum and minimum bounds for the second element (X2) which have to be checked in the algorithm can be written as shown in equation (2.1)

$$\frac{M2}{2} < \; X2 \;\leq\; M2 \qquad (2.1)$$
$$M2 = \; sum\_req2 - \; s1 \qquad (2.2)$$
$$Sum\_req2 \; = \; sum\_req1 - \; X1 \quad (2.3)$$

Where

$M2$ is considered as the maximum bound for the second element which have to be checked in the algorithm

Sum_req2 is the required summation for the two remaining elements:

$s1$ is the minimum value of element in N_K set and it equals to " 1"

$X2$ is the elements which can considered as second element of our algorithm with respect to $X1$

The exact bounds of the third element which have to be checked in the algorithm can be written as shown in equation (2.1)

$s1 \leq X3 \leq M3$ (3.1)

$M3 = X2 - s1$ (3.2)

Where

$M3$ is considered as maximum bound for third element which have to be checked in the algorithm.

$s1$ is the minimum value of element in N_K set and it equals to " 1".

$X2$ is the second element in our algorithm.

$X3$ is the elements which can considered as third element of our algorithm with respect to $X2$.

3. Bounded and Un-Bounde 3 Partition Algorithm

The general algorithm of un-bounded 3 partition depend on checking all of the elements in N_K set with each other's until it covers all of the probabilities without considering any factors which can reduce the looping steps.

The general algorithm of un-bounded three partition can be written as shown below.

Input k value (which determine the largest element of N set), S_req and i=1

Output Matrix B.

Step 1 For $c1 = 1,2, \dots \dots k$, do Steps 2 to 5.

Step 2 For $c2 = 1,2, \dots \dots k$, do Steps 3 to 5.

Step 3 For $c3 = 1,2, \dots \dots k$, do Steps 4 to 5.

Step 4 if ($c1 + c2 + c3 = S\_req$) then enter value of c_1, c_2 and c_3 in the i^{th} row of B Matrix.

Step 5 increment i by 1

Step 6 OUTPUT (Matrix B)

STOP

The general algorithm of bounded three partition depend on checking the elements in N_K set within exact bounds with each other's until it covers all of the probabilities. It achieves our goal to reduce the steps in " For loops" when it is possible.

The general algorithm of bounded three partition can be written as shown below.

Input k value (which determine the largest element of N set), S_req and i=1

Output Matrix B.

Step 1 Calculate $max(X1) = S\_req - (N(1) + N(2))$ and $min(X1) = fix(max(X1)/2)$ While max(X_1) is less than or equal k)

Step 2 For $c1 = min(X1), min(X1) + 1, min(X1) + 2, \dots \dots \dots max(X1)$, do Steps 3 to 7.

Step 3 Calculate $S\_req2 = S\_req - c1, max(X2) = S\_req2 - N(1)$ and $min(X2) = fix(max(X2)/2)$

Step 4 For $c2 = min(X2), min(X2) + 1, min(X2) + 2, \dots \dots \dots max(X2)$, do Steps 5 to 7. (While $c2 < c1$)

Step 5 Calculate $max(X3) = c2 - N(1)$ and $min(X3) = N(1)$

Step 6 For $c3 = min(X3), min(X3) + 1, min(X3) + 2, \dots \dots \dots\ max(X3)$, do Step 7. (While $c3 < c2$)

Step 7 if ($c1 + c2 + c3 = S\_req$) then enter value of c_1, c_2 and c_3 in the i^{th} row of B Matrix and increase i by 1.

Step 8 OUTPUT (Matrix B)

STOP

Fig (1) show the flow chart for bounded 3-Partition Algorithm. It is complicated due to its awareness of not recurrence the elements with each other and arrangement of the three elements from larger to smaller.

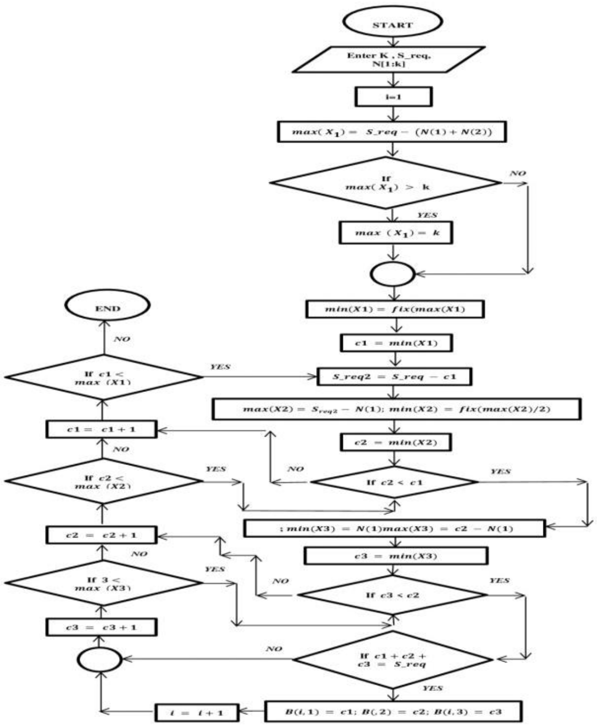

Fig 1. Flow chart for bounded 3-partition algorithm.

4. Matlab Simulation and Results Analysis

In order to conduct a simulation to analyze the algorithm, let's consider k= 20 and $S\_req = 25$.

So, Matlab code can be written as shown below.

```
Clc
Clear all
n = 20;
N = [1:n];
S_req = 25;
i = 1;
m1_max = S_req - ( N(1) + N(2) );
if ( m1_max > n )
  m1_max = n;
end
m1_min = fix ( m1_max/2 );
for c1 =  m1_min : m1_max
  S_req2 = S_req - c1;
  m2_max = S_req2 - N(1);
```

```
   m2_min = fix( m2_max/2 );
   for c2 = m2_min : m2_max
      if ( c2 < c1 )
    m3_max = c2 - N(1);
    m3_min = N(1);
    for c3 = m3_min : m3_max
       if (S_req == (c1+c2+c3) & c3<c2)
         B(i,1) = c1;
         B(i,2) = c2;
         B(i,3) = c3;
         i = i + 1;
       end
    end
      end
   end
end
```

And the result of running the Matlab code is shown in Table (1).
B =

Table 1: Simulation Results From Matlab.

10 8 7	14 10 1
10 9 6	15 6 4
11 8 6	15 7 3
11 9 5	15 8 2
11 10 4	15 9 1
12 7 6	16 5 4
12 8 5	16 6 3
12 9 4	16 7 2
12 10 3	16 8 1
12 11 2	17 5 3
13 7 5	17 6 2
13 8 4	17 7 1
13 9 3	18 4 3
13 10 2	18 5 2
13 11 1	18 6 1
14 6 5	19 4 2
14 7 4	19 5 1
14 8 3	20 3 2
14 9 2	20 4 1

Table (2) show the range values for the variables used in the bounded algorithm depending on the values k= 20 and $S\_req\ = 25$.

Table 2: Range of Variables in the Simulation.

	Maximum value	Minimum Value
S_req	25	25
S_req2	15	5
m1_max	20	20
m1_min	10	10
m2_max	14	4
m2_min	7	2
m3_max	10	1
m3_min	1	1

Analyzing the ranges in Table (2) show that:

- Range of first element " $X1$" lies between 20 and 10.
- Range of second element " $X2$ " lies between 14 and 2. Incase Upper bound of X2 is equal to 14 then lower bound is equal to 7 and if Upper bound of X2 is equal to 4 then lower bound is equal to 2. And there is another ranges in between which depends on the value of $X1$.
- Range of third element " $X3$ " lies between 10 and 1. Upper bound for third element is within $(10 - 1)$ range and value of lower bound is always equal to "1" always, while the upper bound vary according to $X1$ & $X2$. $X3$ is similar to $X2$ principle, however $X3$ follow the equations (3.1) & (3.2) which depend on respect $X2$ value.

By applying random values of K and required summation and calculating the timing arithmetic mean value of bounded 3 partition algorithm and unbounded algorithm. A linear relation is assumed as shown below.

$$T_{Avrg} = r \; T_{B\;Avrg} \qquad (4.1)$$

Where

T_{Avrg} is the arithmetic mean for timing of Un-bounded 3 partition algorithm.

$T_{B\;Avrg}$ is the arithmetic mean for timing of bounded 3 partition algorithm.

r is the linear coefficient that relate between T_{Avrg} & $T_{B\;Avrg}$.

After calculating r, r value can be approximated to 2.73 as noticed during my Matlab simulation. Another r values would be appear depending on many factors such as computer characteristics, used programming language…etc

So, we can write the mathematical relation between the timing of bounded and un-bounded 3 partition algorithm as shown below

$$T_B = \frac{T}{r} \qquad (4.2)$$

So,

$$T_B = c\,T \qquad (4.3)$$

While

T_B is the timing of bounded 3 partition algorithm.

T is the timing of un-bounded 3 partition algorithm.

c is a constant and equal to 0.366 if we rely on r value 2.73.

5. Discussion

The bounds are studied for partitioning N_K set. However, the set is not N_K set always. It can start from any integer number, even it can be a decimal number. Moreover, the set may need sorting from minimum to maximum. These cases are not included and need more practices to be realized and the bounds still possible to be noted. Key encryption application can be studied by specialists to check the efficiency of bounding similar algorithms.

6. Conclusion

In this article, an efficient bound are proposed to control and reduce the steps for 3-Partotion Algorithm. Such bounds are built depending on the required summation value. Many examples simulated through Matlab and measured the timing of the algorithm with bounds and without to compare the timing between. For sure, the bounds of the algorithms would improve the computational performance and efficiency for many cases.

REFERENCES

1. Martens S. The Easiest Hard Problem: Number Partitioning. 2003.
2. Hammer PL and Kogan A. Optimal compression of propositional horn knowledge bases: Complexity and approximation. Artificial Intelligence. 1993;64(1):131-145.
3. Lima AR and Argollo de Menezes M. Entropy-based analysis of the number partitioning problem. Phys Rev E. 2008;63(2):020106(R).
4. Ducha FA and Ricardo de Souza S. Algorithms Analysis for the Number Partition Problem. *CILAMCE XXXIV*. 2013.
5. Gent IP and Walsh T. Analysis of heuristics for number partitioning. Computat Intelligence. 1998;14(3):430-451.
6. Marchetti Spaccamela A and Pelaggi A. Worst case analysis of two heuristics for the set partitioning problem. 1987; 11-23p.
7. Panahi T, Heidari T, and Naeini VS. An efficient parallel algorithm for solving the 3-partition based on ADI. International J Recent Technol Eng. 2013;2(1):98.

Citation: Abu Tammam OY. Bounding of 3-partitioning algorithm for N_K Set. Trans Eng Comput Sci. 2023;4(1):129.