



**An Empirical Evaluation of Algorithms
for solving Subset Sum Problem in terms of Total Bit Length**

Dissertation

Submitted To:

Central Department of Computer Science & Information Technology

Tribhuvan University

Kirtipur, Kathmandu

Nepal

In partial Fulfillment of the requirements for the Degree of Master of Science in
Computer Science & Information Technology

Submitted by:

Rajendra Shrestha

August, 2015

Supervisor

Mr. Arjun Singh Saud



Tribhuvan University

Institute of Science and Technology

Central Department of Computer Science and Information Technology

Student's Declaration

I hereby declare that I am the only author of this work and that no sources other than the listed here have been used in this work.

.....
Rajendra Shrestha

Date: 12 August, 2015



Tribhuvan University
Institute of Science and Technology
Central Department of Computer Science and Information Technology

Supervisor's Recommendation

I hereby recommend that the dissertation prepared under my supervision by **Mr. Rajendra Shrestha** entitled “**An Empirical Evaluation of Algorithms for solving Subset Sum Problem in terms of Total Bit Length**” be accepted as in fulfilling partial requirement for the completion of Masters Degree of Science in Computer Science & Information Technology.

Mr. Arjun Singh Saud

Lecturer,
Central Department of Computer Science and Information Technology,
Institute of Science and Technology,
Kirtipur, Kathmandu, Nepal

Date: 12 August, 2015



Tribhuvan University
Institute of Science and Technology
Central Department of Computer Science and Information Technology

LETTER OF APPROVAL

We certify that we have read this dissertation work and in our opinion it is appreciable for the scope and quality as a dissertation in the partial fulfillment of the requirements of Masters Degree of Science in Computer Science & Information Technology.

Evaluation Committee

Asst. Prof. Nawaraj Paudel
Head of Department
Central Department of Computer Science
& Information Technology
Tribhuvan University
Kirtipur

Mr. Arjun Singh Saud
Lecturer
Central Department of Computer Science
and Information Technology
(Supervisor)

(External Examiner)

(Internal Examiner)

Date:

Acknowledgement

I would like to express my gratitude to all the people who supported and accompanied me during the preparation of this dissertation “**An Empirical Evaluation of Algorithms for solving Subset Sum Problem in terms of Total Bit Length**”. This research work has been performed under Central Department of Computer Science and Information Technology (*Tribhuvan University*), Kirtipur. I am very grateful to my department for giving me an enthusiastic support.

First, I would like to express my gratitude to my supervisor **Mr. Arjun Singh Saud**, who gave me an enthusiastic support from the beginning to the end of the preparation of this dissertation. He is the one who listened to all my problems I faced during this thesis and showed me the way to overcome them.

Most importantly I would like to thank to respected Head of Department of Central Department of Computer Science and Information Technology, Asst. Prof. Nawaraj Paudel for his kind support, help and constructive suggestions. I am very much grateful and thankful to all the respected teachers Prof. Dr. Shashidharam Joshi, Prof. Sudarsan Karanjit, Prof. Dr. Subarna Sakya, Mr. Min Bahadur Khatri, Mr. Bishnu Gautam, Mr. Jagdish Bhatta, Mr. Dheeraj Kedar Pandey, Mr. Sarbin Sayami, Mrs. Lalita Sthapit, Mr. Yog Raj Joshi and Mr. Bikash Balami of CDCSIT, TU, for providing me such a broad knowledge and inspirations.

Special thanks to my family and members of educational organizations that I have been working, for their endless motivation, constant mental support and love which have been influential in whatever I have achieved so far. All my class fellows are worthy of my gratefulness for their direct or indirect support in completion of my dissertation.

I have done my best to complete this research work. Suggestions from the readers are always welcomed, which will improve this work.

Abstract

Subset Sum Problem is an important decision problem in complexity theory and cryptography. Subset sum problem can simply be described as: given a set of positive integers S and a target sum t , is there a subset of S whose sum is t ? The complexity of subset sum can be viewed as depending on two parameters: n , the number of values, and m , the precision of the problem (number of bits required to state the problem). Backtracking algorithm for Subset Sum Problem can be modeled as a binary tree where each node represents a single activation of the recursive code. The worst-case time complexity is $O(2^n)$ when n is used as the complexity parameter [7]. Dynamic Programming breaks a problem down into smaller problems and solves them recursively as divide-and-conquer technique. It solves the problem in $O(m.n^2)$ time. Dynamic Dynamic Programming is the extension of the Dynamic Programming with a dynamically allocated list of target sums. It has the time complexity of $2^{O(x)}$ when the total bit length x of the input set is used as the complexity parameter.

The empirical analysis shows that time complexity of DP and DDP increase sub-exponentially when bit length, m , is increased by 1. At the same time BT is not sensitive to m and its time complexity increases exponentially when number of inputs, n , is increased by 1.

Keywords: Subset Sum Problem, Backtracking, Dynamic Programming, Dynamic Dynamic Programming, Total Bit Length

Table of Contents

CHAPTER 1

Background & Problem Formulation

Pages

1.1. Background	1-6
1.1.1. Subset Sum Problem (SSP)	1
1.1.2. Complexity classes	1-4
1.1.3. Approaches for solving SSPs	4-6
1.1.3.1.1 Backtracking	4-5
1.1.3.1.2 Dynamic Programming	5-6
1.1.3.1.3 Dynamic Dynamic Programming	6
1.2. Introduction	6-7
1.3. Problem Definition	8
1.4. Objective	8
1.5. Motivation	8
1.6. Report Organization	8-9

CHAPTER 2

Literature Review & Methodology

2.1 Literature Review	10-13
2.2 Research Methodology	13

CHAPTER 3

Design & Implementation

3.1. Tools & Language	13-14
-----------------------	-------

3.2. Algorithms	14-25
3.2.1 Dynamic Dynamic Programming	14-15
3.2.2 DDP Example Trace	15-19
3.2.3 Dynamic Programming	19-20
3.2.4 DP Example Trace	20-21
3.2.5 Backtracking	21-22
3.2.6 BT Example Trace	23-25

CHAPTER 4

Data Collection & Analysis

1 Test Case Design	25
2 Data Collection	26
1 For the Dataset with $n = 15$ and $m = 10$	26-28
3 Analysis	29-41

CHAPTER 5

Conclusion & Future Work

1 Conclusion	42
2 Limitations and Future Work	42

References	43-45
-------------------	-------

Bibliography	50
---------------------	----

Appendices

List of Figures

Fig. No.	Caption	Pages
Figure 1	- Euler diagram for P, NP, NP-complete, and NP-hard set of problems	3
Figure 2	- Backtracking	5
Figure 3	- Tracing Backtracking Algorithm	24
Figure 4	- Step-counts for $n = 15, m = 10..20$	30
Figure 5	- Step-counts for $n = 16, m = 10..20$	31
Figure 6	- Step-counts for $n = 17, m = 10..20$	32
Figure 7	- Step-counts for $n = 18, m = 10..20$	33
Figure 8	- Step-counts for $n = 19, m = 10..20$	34
Figure 9	- Step-counts for $n = 20, m = 10..20$	35
Figure 10	- Step-counts for $n = 21, m = 10..20$	36
Figure 11	- Step-counts for $n = 22, m = 10..20$	37
Figure 12	- Step-counts for $n = 23, m = 10..20$	38
Figure 13	- Step-counts for $n = 24, m = 10..20$	39
Figure 14	- Step-counts for $n = 25, m = 10..20$	40
Figure 15	- BT analysis in terms of n	42

List of Tables

Table No.	Caption	Pages
Table 1	- Dynamic Programming	5
Table 2	- Tabular work for Dynamic Programming	20
Table 3	- Average step-count for $n = 15, m = 10..11$	28
Table 4	- Step-counts for $n = 15, m = 10..20$	29
Table 5	- Step-counts for $n = 16, m = 10..20$	30
Table 6	- Step-counts for $n = 17, m = 10..20$	31
Table 7	- Step-counts for $n = 18, m = 10..20$	32
Table 8	- Step-counts for $n = 19, m = 10..20$	33
Table 9	- Step-counts for $n = 20, m = 10..20$	34
Table 10	- Step-counts for $n = 21, m = 10..20$	35
Table 11	- Step-counts for $n = 22, m = 10..20$	36
Table 12	- Step-counts for $n = 23, m = 10..20$	37
Table 13	- Step-counts for $n = 24, m = 10..20$	38
Table 14	- Step-counts for $n = 25, m = 10..20$	39
Table 15	- DDP and DP growth rate when complexity parameter is m	40
Table 16	- BT analysis in terms of n	41

List of Abbreviations

DDP	-	Dynamic Dynamic Programming
DP	-	Dynamic Programming
SSP	-	Subset Sum Problem
BT	-	Backtracking
..	-	ranges (eg 1..5 ranges from 1 to 5)