Opportunities in Theoretical Computer Science

Nitin Saurabh

IIT Hyderabad

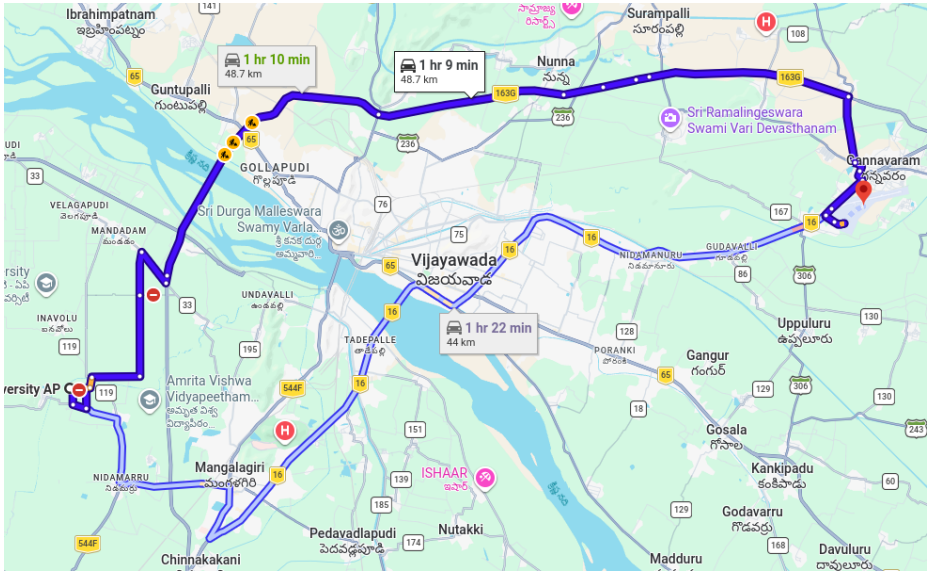
Research Opportunities in Computer Science

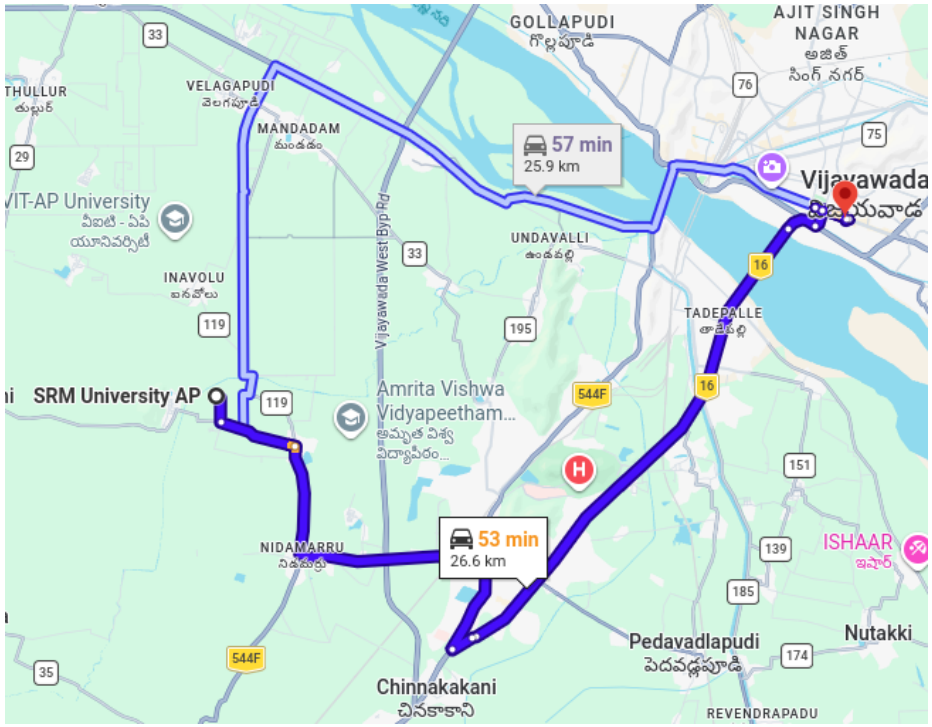
ACM India, SRM University AP
Feb 21, 2026

Theoretical Computer Science

What and Why?

Shortest Way Home





Shortest Path

- **Input:** A directed graph $G = (V, E)$ with weights on edges $w_e > 0$, a starting point s and a destination t

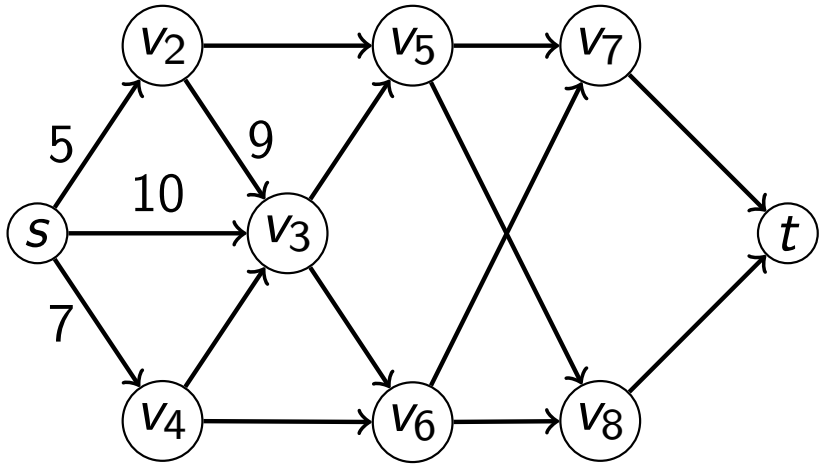


Figure: A weighted directed graph $G = (V, E)$

- **Output:** A shortest path from s to t, e.g., $s \to v_3 \to v_5 \to v_8 \to t$.

The Shortest Path Algorithm

$d(s) = 0$
$p(s) = null$

For all vertices $v \neq s$
$d(v) = \infty$
$p(v) = null$
EndFor

While there is an edge $u \rightarrow v$ such that $d(u) + w(u \rightarrow v) < d(v)$
$d(v) = d(u) + w(u \rightarrow v)$
$p(v) = u$
EndWhile

- Correctness? Efficiency?
- Runtime: $O(|E||V|)$

The Shortest Path Algorithm

$d(s) = 0$
$p(s) = null$

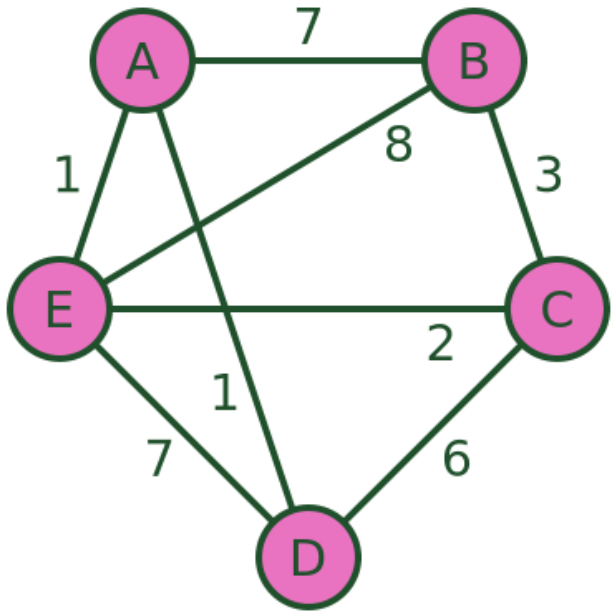
For all vertices $v \neq s$
$d(v) = \infty$
$p(v) = null$
EndFor

While there is an edge $u \rightarrow v$ such that $d(u) + w(u \rightarrow v) < d(v)$
$d(v) = d(u) + w(u \rightarrow v)$
$p(v) = u$
EndWhile

- Correctness? Efficiency?
- Runtime: $O(|E||V|)$

The Travelling Salesman Problem

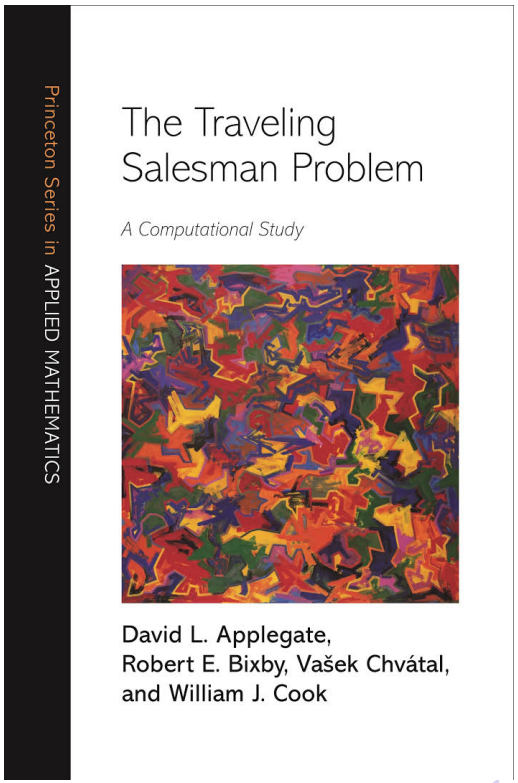
Problem Statement: Consider a delivery boy who must visit n locations in a city labeled $v_1, v_2, \ldots, v_n$. The delivery boy starts at the depot v_1 to pick up parcels for delivery, and wants to find a *tour* – an order in which to visit all the other locations and return to depot. His goal is to find a tour that causes him to travel as little total distance as possible.



The Travelling Salesman Problem

- Applications

  Logistics/delivery
  VLSI chip design
  serving I/O requests on a disk
  execution of software modules to minimize context switching time
  data network routing
  ...

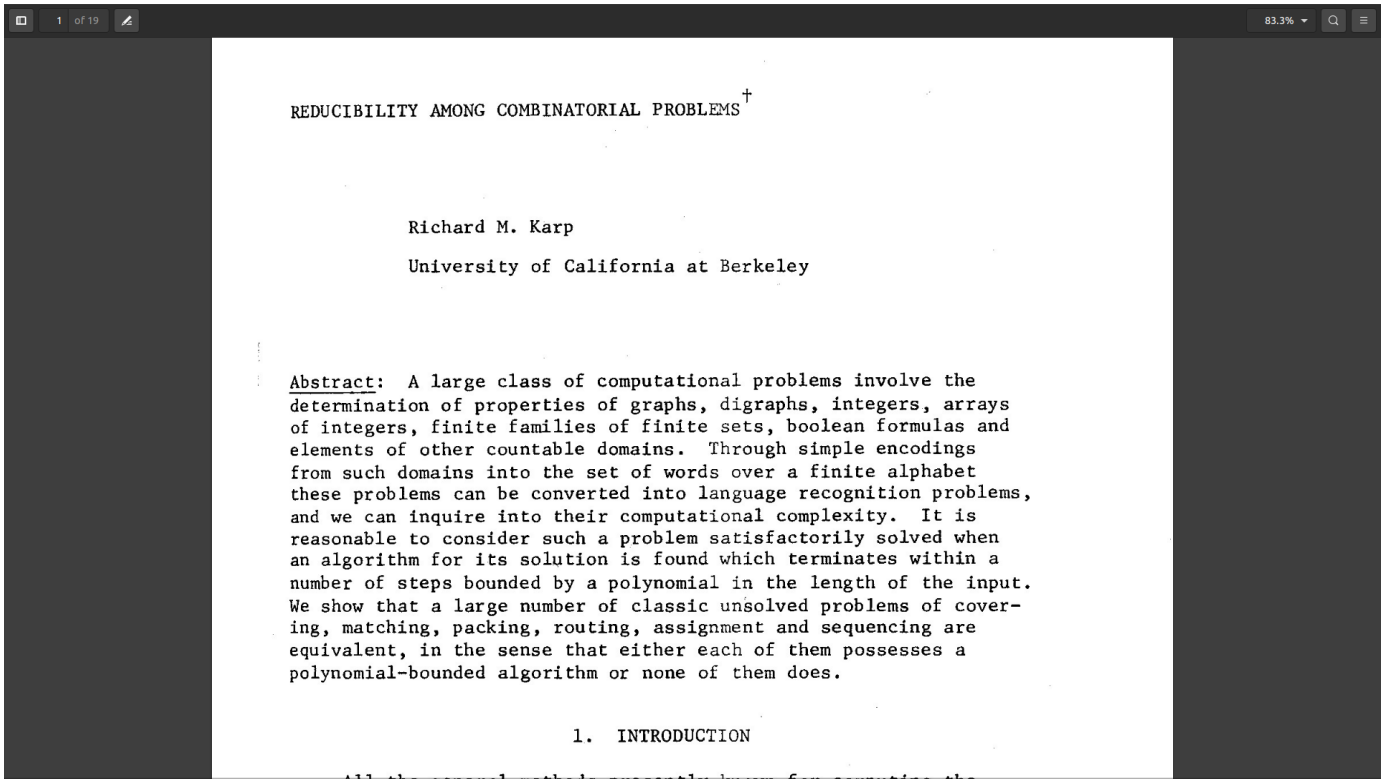


The Travelling Salesman Problem

- unlikely to have a polynomial time algorithm

  theory of NP-hardness

  a web of reductions among problems implying if one has an efficient algorithm then all of them!



REDUCIBILITY AMONG COMBINATORIAL PROBLEMS†

Richard M. Karp

University of California at Berkeley

Abstract: A large class of computational problems involve the determination of properties of graphs, digraphs, integers, arrays of integers, finite families of finite sets, boolean formulas and elements of other countable domains. Through simple encodings from such domains into the set of words over a finite alphabet these problems can be converted into language recognition problems, and we can inquire into their computational complexity. It is reasonable to consider such a problem satisfactorily solved when an algorithm for its solution is found which terminates within a number of steps bounded by a polynomial in the length of the input. We show that a large number of classic unsolved problems of covering, matching, packing, routing, assignment and sequencing are equivalent, in the sense that either each of them possesses a polynomial-bounded algorithm or none of them does.

1. INTRODUCTION

Coping with Hardness

- may be one doesn't need exact solutions
  Approximation
- may be one can work with errors
  Randomization
- may be one is dealing with special instances
  Fixed-Parameter Tractable, etc.
- using hardness to one's advantage
  Cryptography, Pseudorandomness, etc.

Different Paradigms

- different models of computation
- Sublinear algorithms: Algorithms for big data; time/space typically much less than input size
- Streaming algorithms: input comes as a stream; limited processing time & space
- Online algorithms: typically action has to be taken as soon as imput arrives
- Parallel algorithms: mutliple processors are available to carry out tasks in parallel
- Dynamic algorithms: needs to maintain a solution while input may be changing
- Distributed algorithms: input is distributed among parties
- Coding Theory: error correction; needs to recover data even when the data is corrupted

Different Paradigms

- Computational Complexity Theory: study of computational resources required
- Computational Algebra: using algebra for (efficient) computation
- Computational Learning Theory: deals with learning data from fewer probes
- Logic: formal verification of softwares; studying computational problems using formal language theory
- Quantum Computing: deals with computation on devices using principles of quantum mechanics

"aims to understand limits of computation with mathematical rigour"

Theoretical Computer Science

Where?

Theoretical CS at IIT Hyderabad

M V Panduranga Rao

Theoretical CS

Manaswi Paraashar

Quantum Computing
Computational Complexity

Maria Francis

Computational Algebra
Symbolic Computation
Lattice Cryptography

Theoretical CS at IIT Hyderabad

N R Aravind

Graph Theory
Combinatorics
Algorithms

Rakesh Venkat

Approximation Algorithm
Complexity Theory

Rameshwar Pratap

Algorithms in ML
Theoretical CS

Theoretical CS at IIT Hyderabad

Rogers Mathew

Graph Theory
Combinatorics
Graph Algorithms

Subrahmanyam Kalyanasundaram

Probabilistic Techniques
Complexity Theory
Randomized Algorithms

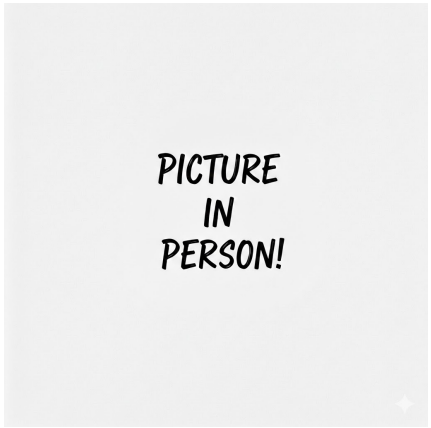
Sobhan Babu

Big Data Analytics
Graph Theory
Applied Algorithms

Theoretical CS at IIT Hyderabad

Sathya Peri

Theory of Databases
Distributed Algorithms
Networking Algorithms



Nitin Saurabh

Complexity Theory
Circuit Complexity
Quantum Complexity

Theoretical CS in India

- all IITs
- The Institute of Mathematical Sciences (IMSc), Chennai
- Tata Institute of Fundamental Research (TIFR), Mumbai
- Indian Institute of Science (IISc), Bangalore
- Indian Statistical Institute (ISI), Kolkata
- Chennai Mathematical Institute (CMI), Chennai
- National Institute of Science Education and Research (NISER), Bhubaneswar
- IIIT Hyderabad, Delhi and Bangalore
- IISERs and NITs
- Microsoft Research India
- IBM Research India
- Google India

How to approach them?

- write to them with your broad interest in their research areas
- seek research internships, pre-doc fellowships, summer internships, etc.
- MS by research
- PhD programmes

"Computer Science is no more about computers than astronomy is about telescopes"
– Don E. Knuth

"The infinite we shall do right away. The finite may take a little longer."
– Satnislaw Ulam

Thank You! Questions?