COMP2001/COMP2011 Artificial Intelligence Methods

Ender Özcan

Revision Lecture

The University of Nottingham
UNITED KINGDOM · CHINA · MALAYSIA

What did we cover?

- Introduction to heuristic search / optimisation / decision support, heuristic types, search paradigms, case studies: bin packing, TSP, graph colouring, classroom assignment, pseudo-random numbers.
- Components of heuristic search: representation, evaluation, neighbourhoods
- Basic hill climbing/local search methods: First Improvement, Best Improvement, Davis's bit hill climbing, Random mutation hill climbing

What did we cover? II

- Metaheuristics
  - Components: representation, evaluation, initialisation, move operators, guideline, stopping conditions, mechanism for escaping the local optima
  - Single point based search (Local Search Metaheuristics): Iterated Local Search, Tabu Search
  - Move acceptance methods: Taxonomy, Late Acceptance, Great Deluge, Simulated Annealing,…
  - Parameter setting: parameter tuning vs parameter control.
- Performance Comparison of Algorithms

What did we cover? III

- Evolutionary Algorithms: GAs, MAs, MMAs
- Hyper-heuristics, classification
  - Selection Hyper-heuristics: heuristic selection and move acceptance, methods controlling constructive heuristics, HyFlex
  - Generation Hyper-heuristics: genetic programming, genetic algorithm for generating online bin packing heuristics
  - Advanced topics
- Fuzzy systems and planning for robotics

Examinable Material

- Materials provided over Moodle and in lecture
  - What's written on the slides
  - What's in the papers provided for reading
  - What's in the videos
  - What's in the engagement activities
  - What's said in lectures
  - What's written on the whiteboard
  - Asynchronous/synchronous exercises solved (forums and quizzes)
  - Questions: Asked during a lecture
  - Answers: Said or written during a lecture

Bloom's Taxonomy (a subset)

- The form of knowledge needed to answer an exam question:
  - **K**nowledge – for what we often term 'bookwork' based upon recall of factual information'
  - **C**omprehension – or 'understanding', where students are asked to perform such actions as to describe, explain, classify ideas or concepts
  - **A**pplication – for the situations where a student is asked to undertake an 'unseen' task by using knowledge in a new way (or a variant upon one previously shown in class)

SAMPLE PAST EXAM QUESTIONS

Warning

- Wherever you provide your reasoning or explanation do not return lengthy answers.
  - Providing a lengthy answer may increase the chances that you make a mistake
  - Every erroneous comment in a solution/answer will get penalised
- Be concise and do not exceed the word limitation.
- There is no solution where your answer should exceed half a page.

What is/are …?
Define…
Name X (e.g., 2) …

- Heuristic
- Graph (colouring) heuristics
- Types of search methods
- Tardiness
- Hill climbing
- Metaheuristic
- Hyper-heuristic
- Combinatorial optimisation problem
- Single point based search
- Evolutionary algorithm
- Delta/incremental evaluation
- Fuzzy set, …

True/False

- Which one of the following statements are TRUE?

  (i) A seed value will correspond to a specific sequence of generated values for a given random number generator.

  (ii) A heuristic search algorithm might result with a poor solution.

  (iii) A search algorithm based on the nearest neighbor constructive heuristic/operator can always find a better solution than a search algorithm based on the pairwise exchange perturbative heuristic/operator for solving a given instance of the Travelling Salesman Problem.

  (i), (ii) only

Multiple Choice

- Please indicate which algorithm below is a local search metaheuristic:

  a. Ant colony optimisation

  b. Particle Swarm Optimisation

  c. Iterated Local Search

  d. Greedy Randomized Adaptive Search Procedure

What is/are …?
Define… II
Name X (e.g., 2) …

- Mechanisms for escaping local optima
- Population based metaheuristics
- Local search metaheuristics
- Components of metaheuristics

What is (are) the difference(s)/similarity (-ies) between…?

- Memetic algorithm and multimeme memetic algorithms
- Genetic algorithm metaheuristic and genetic algorithm hyper-heuristic

What is (are) the difference(s)/ similarity(-ies) between…?

- System effectiveness and system efficiency
- Optimisation and decision support
- Mutational and hill climbing heuristics
- Constructive and perturbative heuristics
- Simulated annealing and choice function
- Iterated local search and memetic algorithms
- Exploration and exploitation in search

Computation Question

- Consider 4 jobs having the processing times p_j, the due-dates d_j and the weights w_j of the jobs j=1, ..., 4, given in the table:

jobs	1	2	3	4
p_j	10	10	13	4
d_j	4	2	1	12
w_j	14	12	1	12

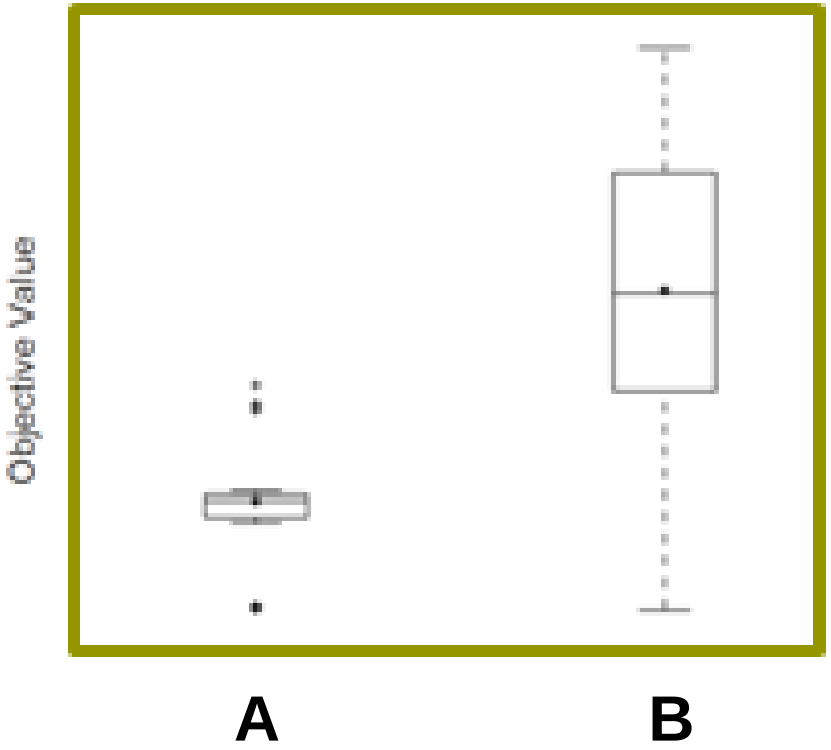
What is the weighted tardiness for the schedule:

<2, 1, 4, 3> ?

Performance Comparison I

- Given two algorithms A and B which are run 100 times on an instance of MAX-SAT and box plot of all objective values from all runs for each algorithm are provided on the right,
- Which algorithm performs better on the instance with a statistically significant performance difference?

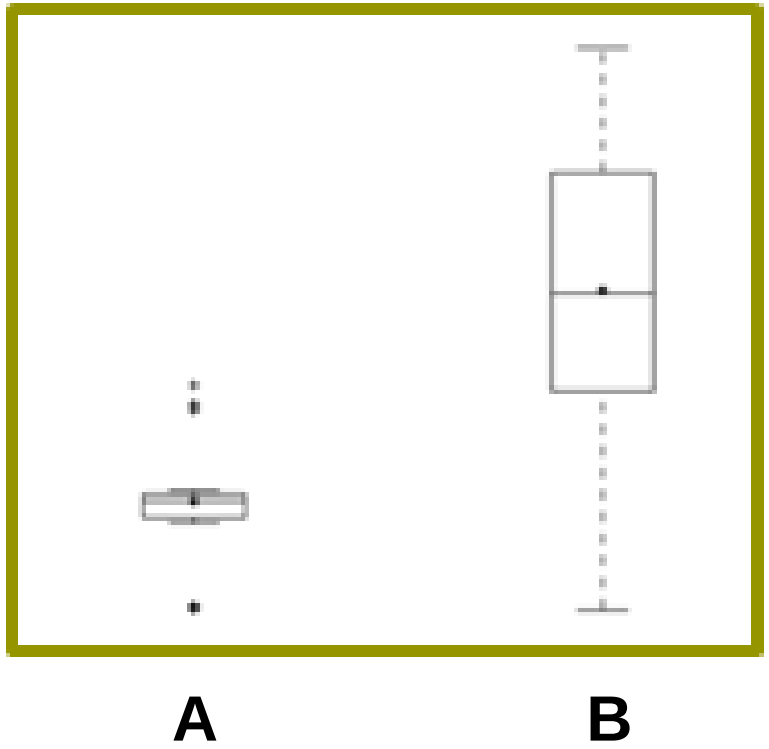
Assume a minimisation objective function



Performance Comparison II

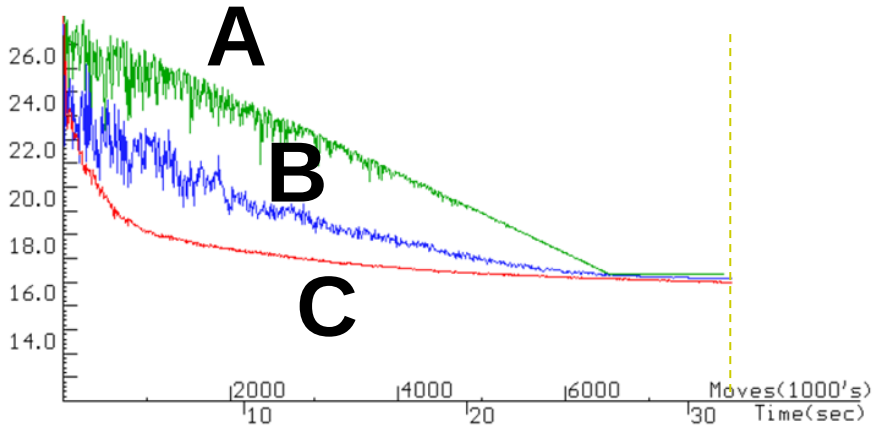
- Given two algorithms A and B which are run 100 times on an instance of MAX-SAT and box plot of all objective values from all runs for each algorithm are provided on the right,
- Which algorithm performs better on MAX-SAT with a statistically significant performance difference?

Assume a minimisation objective function



Performance Comparison III

- Given three algorithms A, B and C which are run 100 times on an instance of a MAX-SAT problem and the mean objective value progress plots in time are in the right,
- Which algorithm performs the best with significance in the given duration?



Computation Question – Applying an Algorithm

- A bus company has 50 busses, each with 52 seats.
- There will be a concert outside a city and groups of people will arrive at the city centre to attend it. The company will be responsible for the transportation of all groups to the concert site from the city centre. Each group of people has to be placed into the same bus. Different groups are allowed in the same bus. The capacity of a bus cannot be exceeded.
- The company aims to reduce the number of busses used for transportation.
- 33 groups arrive each having the following number of people: 8, 44, 10, 10, 10, 10, 10, 10, 25, 25, 25, 25, 25, 25, 25, 9, 9 , 12, 12, 12, 13, 13, 11, 11, 11, 11, 12, 12, 12, 44, 46, 37
- By using an appropriate packing heuristic, place all groups into busses. Please give details in each step of your heuristic.

Computation Question –Executing an Algorithm

```
1  long seed = 123456789;
2  Random generator = new Random(seed);
3  double num;
4  for (int trial=0; trial<4; trial++)
5      num = generator.nextDouble() ;
6  System.out.print(num); // print out num
```

Given the pseudo-code above and assuming that the seed 345678912 produces the following sequence of double values in the given order

<0.13, 0.14, 0.75, 0.39, 0.234, 0.005, 0.81, ...>

which value would be printed out after the for loop?

0.39

Computation Question – Representation Design

- **Maximum Satisfiability** is the problem of determining the maximum number of clauses, of a given Boolean formula in conjunctive normal form, that can be made true by an assignment of truth values to the variables of the formula
- Given a MAX-SAT problem with *N* variables. How many configurations can be encoded using a binary representation?
- How would you evaluate a given solution?

Algorithm Design Problem

- **Maximum Satisfiability** is the problem of determining the maximum number of clauses, of a given Boolean formula in conjunctive normal form, that can be made true by an assignment of truth values to the variables of the formula.

- Design an Iterated Local Search/Tabu Search/Simulated Annealing… algorithm for solving this problem, discussing each algorithmic component and parameter setting.

Computation Question – Applying an Algorithm

- **Maximum Satisfiability** is the problem of determining the maximum number of clauses, of a given Boolean formula in conjunctive normal form, that can be made true by an assignment of truth values to the variables of the formula.
- Given the problem instance: $(x1 \vee x2) \wedge (\neg x4 \vee x6) \wedge (\neg x1 \vee x3) \wedge (x2 \vee \neg x6) \wedge (\neg x2 \vee x3) \wedge (x3 \vee x5)$ and current solution of 110100, what is the solution that will be returned by best-improvement (steepest ascent)? Show all your steps.

Exercise Question: Illustration of a Run of Tabu Search on a Scheduling Problem

Example:

jobs	1	2	3	4
p_j	10	10	13	4
d_j	4	2	1	12
w_j	14	12	1	12

$1 \mid d_j \mid \sum w_j T_j$

Schedule four jobs on a machine

$T_j = \max(C_j - d_j, 0)$
tardiness of job j

Neighbourhood operator: go through all schedules that can be obtained through adjacent pairwise interchanges, choose the best.

Tabu-list: pairs of jobs (j, k) that were swapped within the last two moves

Run the algorithm for 2 while loop iterations, starting with the initial solution: S_0 = <2, 1, 4, 3>

Simulated Annealing with Geometric Cooling

INPUT: T_0 (> 0), α (cooling rate<1.0)

Generate an initial schedule S_0 using some heuristics

Set $S_k = S_{best} = S_0$, k=0;

REPEAT

Select $S_{new} \in \mathcal{N}(S_k)$ // Make a move from S_k to S_{new} based on $\mathcal{N}$

If $F(S_{new}) < F(S_k)$ **then** $S_{k+1} = S_{new}$ // an improving move is made

else // A worsening solution is obtained

generate a random uniform number in (0,1], U(0,1)

If U(0,1) < $e^{-\frac{F(S_{new})-F(S_k)}{T_k}}$ **then** $S_{k+1} = S_{new}$ // Accept worsening move

else $S_{k+1} = S_k$ // Reject worsening move

// Keep track of the best solution found so far

If $F(S_{new}) < F(S_{best})$ **then** $S_{best} = S_{new}$

$T_{k+1} = \alpha T_k$; // *geometric cooling:* multiply previous temp with α (value<1.0)

$k = k+1$;

UNTIL (*stopping condition* = true)

Exercise Question:

jobs	1	2	3	4
p_j	9	9	12	3
d_j	10	8	5	28
w_j	14	12	1	12

- Consider the scheduling problem 1 | d_j | $\sum w_j Tard_j$ *(F)*

  $Tard_j = \max(C_j - d_j, 0)$ **tardiness** of job *j*
- Apply the simulated annealing to the problem starting out with the <3, 1, 4, 2> as an initial sequence.
- **Neighbourhood operator**: perform a random adjacent pairwise interchange
- Choose α = 0.9 and T_0 = 0.9
- Use the following numbers in the given order as **random numbers where appropriate**
  - for choosing a random job i in [1..4]: <1, 3, 2, 4, …> to apply adjacent pairwise interchange of (i)th and (i+1)th entries in the permutation
  - for *U*(0,1) : <0.17, 0.009, ...>

Algorithm Design Problem II

- High school timetabling (HST) is the problem of assigning a time-slot to each lesson taught by a teacher and taken by students without any clashes, minimising the number constraints that are violated.

- Design a genetic algorithm using a direct solution representation for solving the HST assigning each lesson to a time-slot, e.g., 5 days x 8 time-slots (per day).
- Your solution must explain all your algorithmic choices, in particular chromosome length, (candidate solution) representation showing how a complete solution can be obtained with respect to the chromosome length, initialisation, genetic operators, replacement, termination and any other relevant parameter settings.

Q&A

Thank you.

Ender Özcan ender.ozcan@nottingham.ac.uk

University of Nottingham, School of Computer Science
Jubilee Campus, Wollaton Road, Nottingham
NG8 1BB, UK
http://www.cs.nott.ac.uk/~pszeo