ANALYSIS OF HEURISTICS

EMPIRICAL ANALYSIS

- Heuristic is programmed and compared with existing heuristics and against lower bounds on optimal solutions for representative set of test problems.

WORST CASE ANALYSIS

- Determine bounds on the largest possible difference between the value of the optimal solution and the value of the solution generated by the heuristic.

ASYMPTOTIC PROBABILISTIC ANALYSIS

- This approach considers algorithms that are guaranteed to provide optimal or near-optimal solutions on almost all large problem instances.
- A probability distribution is specified over the set of problem instances of each size (e.g., number of nodes, n).

- The analysis leads to results of the form:

  - $$\text{Prob}\left\{\frac{\text{heuristic solution}}{\text{optimal solution}} \leq 1 + \varepsilon\right\} \geq 1 - \delta$$
    where $\delta \rightarrow 0$ as $n \rightarrow \infty$ for any prespecified $\varepsilon > 0$.

  - As the problem size n becomes large, the algorithm almost always ($\delta \rightarrow 0$) gives arbitrarily good solutions (within an amount ε).

STATISTICAL ANALYSIS

- For a given hard optimization problem, if a systematic procedure for generating independent heuristic solutions exists, then we can apply standard statistical inference techniques to estimate the optimal solution.

- Researchers have been successful in computing accurate point and interval estimates for a variety of problems.

WORST CASE ANALYSIS

- Properties
  - Most popular (after empirical analysis)
  - Seems like magic
- Structure of results
  - For minimization problems

$$\frac{\text{Value of heuristic solution}}{\text{Optimal solution value}} \leq R_H(n)$$

where $R_H(n)$ is a worst case ratio for the heuristic algorithm H which may be a function of n, a measure of problem size.

THE BIN PACKING PROBLEM

PROBLEM STATEMENT

Given a set of items $\{I_1, I_2, \ldots, I_n\}$ with weights $\{W_1, W_2, \ldots, W_n\}$. The object of the bin packing problem is to pack these items into as few bins, each of capacity W, as possible.

➨ This problem arises in many practical situations

EXAMPLE — the routing and scheduling of vehicles

- ♦ Suppose a firm needs to assign round-trips from a depot to a fleet of vehicles.
- ♦ Each round-trip has a duration of from 1 to 8 hours.
- ♦ If each vehicle can spend no more than 10 hours away from the depot each day, then $W = 10$.
- ♦ The smallest feasible fleet size is desired.

➨ Despite its seeming simplicity, this problem is extremely difficult to solve optimally.

HEURISTIC 1

- First-fit is a straightforward heuristic approach.
  - ♦ Consider the ordered list $L = (W_1, W_2, \ldots, W_n)$ of weights.
    - Pack the weights of L in the order in which they occur.
    - Fill each bin as much as possible before moving on to the next bin.
    - When it is W_k's turn to be packed, we find the first (lowest index) bin that W_k can fit into.
- Analysis
  - ♦ For a list L of weights, if we let FF(L) denote the number of bins required by the first-fit heuristic and OPT(L) be the number of bins required in the optimal solution, then as OPT(L) gets large, $\frac{FF(L)}{OPT(L)} \leq \frac{17}{10} + \varepsilon$.
  - ♦ This result indicates that, for large problems, in the worst case the first-fit heuristic will be no more than 70% away from optimality.

HEURISTIC 2

- A modification of the first-fit heuristic is the *first-fit decreasing heuristic*.
  - Initially, the weights are rearranged into decreasing order.
  - Packing then proceeds as in the first-fit case.
  - By handling the larger weights in the beginning, the first-fit decreasing heuristic avoids some of the difficulties that befall the first-fit heuristic. In this case, as OPT(L) gets large,

$$\frac{\text{FFD(L)}}{\text{OPT(L)}} \leq \frac{11}{9} + \varepsilon .$$

  - FFD(L) is the number of bins required by the first-fit decreasing heuristic. Thus, the first-fit decreasing heuristic will never be more than approximately 22% away from optimality.

KNAPSACK PROBLEM VARIANT

PROBLEM STATEMENT

$$\text{Min} \sum_{j=1}^{n} c_j x_j \qquad \text{(KP)}$$

$$\text{s.t.} \quad \sum_{j=1}^{n} a_j x_j \geq b$$

$x_j \geq 0$ and integer, $j = 1,\ldots,n$. Assume that $\frac{c_1}{a_1} \leq \frac{c_2}{a_2} \leq \ldots \leq \frac{c_n}{a_n}$, $a_j \leq b \;\forall\, j$.

HEURISTIC

- Problem KP is a hard problem to solve optimally.
- An obvious heuristic for KP would be to solve the continuous version of KP and then round the resulting solution up.
- If $x_1^*, \ldots, x_n^*$ is the optimal continuous solution, then $\lceil x_1^* \rceil, \ldots, \lceil x_n^* \rceil$ is the heuristic solution.

ANALYSIS

- The optimal continuous solution to KP is

$$x_1{}^* = \frac{b}{a_1} \text{ and } x_j{}^* = 0 \text{ for } j \geq 2.$$

- The heuristic solution is

$$x_1 = \left\lfloor \frac{b}{a_1} \right\rfloor \text{ and } x_j = 0 \text{ for } j \geq 2.$$

- Let OPT denote the value of the optimal solution to KP, HEU the heuristic solution value, and CON the continuous solution value. Then,

$$\frac{\text{HEU}}{\text{OPT}} = \frac{c_1 \left\lfloor \frac{b}{a_1} \right\rfloor}{\text{OPT}} \leq \frac{c_1 \left\lfloor \frac{b}{a_1} \right\rfloor}{\text{CON}} \text{ as OPT} \geq \text{CON},$$

$$= \frac{c_1 \left\lfloor \frac{b}{a_1} \right\rfloor}{c_1 b / a_1} = \frac{a_1}{b} \left\lfloor \frac{b}{a_1} \right\rfloor \leq 2.$$

FURTHER ANALYSIS

- We have now identified an upper bound on $\frac{\text{HEU}}{\text{OPT}}$, but is this bound tight?
- To determine the tightness of the bound, construct an instance for KP for which $\frac{\text{HEU}}{\text{OPT}} = 2$ or $\frac{\text{HEU}}{\text{OPT}} = 2 - \varepsilon$ for any $\varepsilon > 0$.
- Let δ be any number less than 1/2. Consider the following problem:

$$
\begin{aligned}
&\text{Min } x_1 + 2\delta x_2 \\
&\text{s.t. } x_1 + \delta x_2 \geq 1 + \delta \\
&\qquad x_1, x_2 \geq 0 \text{ and integer.}
\end{aligned}
$$

- HEU $= 2$ and OPT $= 1 + 2\delta$ for this instance of KP.
- $\frac{\text{HEU}}{\text{OPT}} \leq \frac{2}{1+2\delta}$.
- Now the quantity $\frac{2}{1+2\delta}$ can be made arbitrarily close to 2 by making δ small enough.

♦ Thus, the bound $\frac{\text{HEU}}{\text{OPT}} = 2$ is the best possible (tight) and the rounding heuristic will never be more than 100% away from optimality.

WORST CASE ANALYSIS — SUMMARY

- There are two phases involved in performing a worst case analysis.

  1. Identify R(n).

  2. Show that R(n), the bound found, is tight.

- Note that R(n) need not always be a constant, it may be a function of n or some other parameter of the problem.

- Identifying R(N) in general is a demanding exercise. As a consequence, only the simplest heuristics are amenable to this form of analysis.

COMPUTATIONAL IMPLEMENTATION OF ALGORITHMS

NETWORK ALGORITHM PERFORMANCE INDICATORS

- Central processing time: The amount of *central processing time* needed for program execution which encompasses input/output, preprocessing, and starting techniques.

- Accuracy:
  - Heuristic procedures: Performance measures previously discussed.
  - Exact algorithms: Numerical stability of the algorithm is key.

- Number of iterations required by the algorithm. This statistic is independent of the computer used.

PERFORMANCE INDICATORS (CONTINUED)

- Robustness: Domain of problems on which the algorithm performs effectively.
- User friendliness:
  - Versatility
  - Portablility
  - Ease of use
- Storage requirements associated with the algorithm.
- The basic operations count: Represents the number of times an operation such as an addition is required during the execution of an algorithm.
- Parallel computing raises a host of additional issues.