Today

- Problem
- Brute-Force/Exhaustive
- Greedy
- Estimators
- LP/ILP Provision
- ILP Schedule and Provision

Previously

- General formulation for scheduled operator sharing
  - VLIW
- Fast algorithms for scheduling onto fixed resource set
  - List Scheduling

Today: Provisioning

- Given
  - An area budget
  - A graph to schedule
  - A Library of operators
- Determine:
  - Best (delay minimizing) set of operators
  - i.e. select the operator set

Exhaustive

1. Identify all area-feasible operator sets
   - E.g. preclass exercise
2. Schedule for each
3. Select best
   • \rightarrow\text{optimal}
   • Drawbacks?

Exhaustive

- How large is space of feasible operator sets?
  - As function of
    • operator types \(- N\)
      - Types: add, multiply, divide, ....
    • Maximum number of operators of type \(M\)
Size of Feasible Space

• Consider 10 operators
  – For simplicity all of unit area
• Total area of 100
  • How many cases?

Implication

• Feasible operator space can be too large to explore exhaustively

Greedy Incremental

• Start with one of each operator
• While (there is area to hold an operator)
  – Which single operator
    • Can be added without exceeding area limit?
    • And Provides largest benefit?
  – Add one operator of that type
• How long does this run?
• Weakness?

Example

Find best 5 operator solution.

Estimators

• Scheduling expensive
  – $O(|E|)$ or $O(|E| \times \log(|V|))$ using list-schedule
• Results not analytic
  – Cannot write an equation around them
• Saw earlier bounds sometimes useful
  – No precedence → is resource bound
  – Often one bound dominates

Review if this captures
Estimations

• Step 1: estimate with resource bound
  – $O(|E|)$ vs. $O(N)$ evaluation

• Step 2: use estimate in equations
  – $T = \max(N_1/R_1, N_2/R_2, \ldots)$

LP Formulation

• Linear Programming
• Formulate set of linear equation constraints (inequalities)
  - $Ax_1 + Bx_2 + Cx_2 \leq D$
  - $x_0 + x_1 = 1$
  - $A, B, C, D$ – constants
  - $x_i$ – variables to satisfy
• Solve in polynomial time
  – Software packages exist
• Solutions are real (not integers)

LP Constraints

• Let $A_i$ be area of operator type $i$
• Let $x_i$ be number of operators of type $i$
$$\sum A_i \times x_i \leq Area$$

Achieve Time Target

• Want to achieve a schedule in $T$ cycles
• Each resource bound must be less than $T$ cycles:
  - $N/x_i < T$
• But do we know $T$?
• Do binary search for minimum $T$
  – How does that impact solution time?

LP returns reals

• Solution to LP will be reals
  – $x_0 = 1.76$
• Not constrained to integers
• Try to round results
  – Sometimes works well enough
    • For some problems, can prove optimal

ILP

• Integer Linear Programming
• Can constrain variables to integers
• No longer polynomial time guarantee
  – But often practical
  – Solvers exist
• Option: ILP formulation on estimates
ILP Provision and Schedule

• Possible to formulate whole operator selection and scheduling as ILP problem

Formulation

• Integer variables $M_i$  
  – number of operators of type $i$
• 0-1 (binary) variables $x_{i,j}$  
  – 1 if node $i$ is scheduled into timestep $j$
  – 0 otherwise

• Variable assignment completely specifies schedule
• This formulation also for achieving a target time $T$  
  – $j$ ranges 0 to $T-1$

Constraints

1. Total area constraints
2. Not assign too many things to a timestep
3. Assign every node to some timestep
4. Maintain precedence

(1) Total Area

• Same as before  
  \[
  \sum A_i \times M_i \leq Area
  \]

(2) Not overload timestep

• For each timestep $j$  
  – For each operator type $k$
  \[
  \sum_{o_l \in FU_k} x_{i,j} \leq M_k
  \]

(3) Node is scheduled

• For each node in graph  
  \[
  \sum_j x_{i,j} = 1
  \]

Can narrow to sum over slack window.
(4) Precedence Holds

- For each edge from node i to node k

\[ \sum_j j \times x_{i,j} - \sum_j j \times x_{k,j} \leq -1 \]

Can narrow to sum over slack windows.

Round up Algorithms and Runtimes

- Exhaustive Schedule
- Exhaustive Resource Bound Estimate
- Greedy Schedule
- LP on estimates
  - Particular time bound
  - Minimize time
- ILP on estimates and exact
  - Particular time bound
  - Minimize time

Admin

- Assignment 2 out
  - Programming assignment
  - Now in two pieces
- Reading on web

Big Ideas:

- Estimators
- Dominating Effects
- Reformulate as a problem we already have a solution for
  - LP, ILP
- Technique: Greedy
- Technique: ILP