Parallel Branch-and-Cut for Optimization in Production Planning

PDPTA’97

Dieter Homeister
Hartmut Schwab
Interdisziplinäres Zentrum für Wissenschaftliches Rechnen (IWR)
Universität Heidelberg

July 3rd, 1997
Cost minimal electricity generation

Decisions:
on/off times of the power plants
construction of additional links
new power plants
type of the generators (cole, gas, etc.)

Restrictions:
expected demand
maintenance intervals
environmental costs

Results:
submarine link decision: 6 000 000 ECU saved
cole/gas plant decision: 10 000 000 ECU saved

Parallel Branch-and-Cut for Optimization in Production Planning

D. Homeister
PDPTA’97
Heidelberg Univ., IWR
Parallel Branch-and-Cut for Optimization in Production Planning

Cost minimal telecommunication (BASF USA)

Decisions:
- leased / switched lines
- partial use of VPN
- carrier

Restrictions:
- expected demand
- tariff structure
- internal, external and 800 calls

Results:
10% savings in total service costs

(60 sites in USA)
Branch-and-Bound

max \, cx
\text{s.t.} \, Ax < b \quad \text{(some/all } x \text{ integer/binary)}
\quad \begin{align*}
l &< x < u
\end{align*}
Parallel Branch-and-Cut for Optimization in Production Planning

Branch-and-Bound

\[ \text{max } cx \]
\[ \text{s.t. } Ax < b \quad (\text{some/all } x \text{ integer/binary}) \]
\[ l < x < u \]

Branch 1
Branch 2
Example: 5 binary variables => 32 possibilities
Branch & Bound algorithm: only 11 nodes

First solution sets a limit for many other branches
optimal solution

Node computations are independent => parallel execution

Next step: Branch & Cut

Advantage of Branch & Cut: less nodes, faster solution
Disadvantages: longer CPU time per node,
global cut database necessary (hard to parallelize)
The Branch&Cut algorithm uses additional artificial constraints ("cuts") to restrict the interesting area and to get the solution faster.
Strategies

B&C (Branch and Cut)  B&C with skipfactor=3

C&B (Cut and Branch)  B&C with fast_startup +skipf.=3

B&C with maxlevel=3  B&C+fast_st.+skipf.=3+load balanc.
Modular software structure

- 40 modules, 500000 lines of code
- no global variables between modules
- alternative implementations of some modules exist; same purpose, same interface, different algorithm examples: PVM/PARIX, malloc/debug_malloc
- exchanging a module never requires changes in other modules
- dummy parallel modules allow a sequential version; good for debugging
- portability: only 3 machine-specific modules
- most modules include a stand-alone selftest
- strictest compiler checks and make dependency checks
Parallel Branch-and-Cut for Optimization in Production Planning

Module overview

- **Parallel frame**
  - **cut database**
  - **solve node**
  - **tree search**
  - **combinatorial cuts (knapsack)**
  - **lift and project**
  - **path ineqal. cuts**
  - **flow cover cuts**
  - **CPLEX-gomory cuts**

- **LP interface layer**
  - **CPLEX callable subroutine library**
  - **Xpress-MP OSL**
  - **interior point, ADM**

- **Utility modules:**
  - **memory alloc. + utils**
  - **debug vers.**
  - **machine specific**
  - **specific/Sun**
  - **Parsytec specif.**
  - **communic. PVM**
  - **comm./PARIX**
  - **PARIX**

**= alternative modules**

D. Homeister
PDPTA'97
Heidelberg Univ., IWR
Parallel Branch-and-Cut for Optimization in Production Planning

Current + future work:
- better speedup in case of finer granularity (here: 816 tasks / 3 min)
- modified tree search algorithm
- reduced double work (here: 30%)
- improved communication (here: 9%com, 11%wait)

BASF telecom model: speedup on MIMD parallel machine Parsytec GC

![Graph showing speedup S(p) vs number of processors p]

D. Homeister
PDPTA’97
Heidelberg Univ., IWR
Parallel Branch-and-Cut for Optimization in Production Planning

BASF telecom model: speedup on MIMD parallel machine Parsytec GC

speedup $S(p)$

number of processors $p$ (total time $p=1$: 51min, $p=15$: 4.1min)
Parallel Branch-and-Cut for Optimization in Production Planning

GESÄ cole/gas model: speedup on MIMD parallel machine Parsytec GC

- Speedup $S(p)$
- Number of processors $p$
- Total time $p=1$: 2.8h, $p=15$: 12min
Parallel Branch-and-Cut for Optimization in Production Planning

BASF telecom model: speedup on PVM workstation cluster

- Speedup $S(p)$
- Number of workstations $p$ (total time $p=1$: 6.0h, $p=6$: 0.8h)

D. Homeister  PDPTA’97  Heidelberg Univ., IWR
Parallelization of Branch & Cut

Parallel Hardware: Parsytec GC + PVM workstation clusters

Master-slave topology:

- **Master:**
  - holds nodelist
  - holds cut data base
  - makes branching decision
- **Slaves:**
  - solve single Branch&Cut nodes
    (linear algebra + cut generation)

- Load balancing: processor farm
- Coarse granularity (>0.1 sec per job)
- Speedup up to 33(!) on 31 slave processors
  (superlinear behavior of parallel tree execution)
- Efficiency/processor usage >90%, but some double work
- Robust code, modular, portable
Efficiency of the parallel Branch & Cut

Fast startup with multiple variable branching
Also useful against search anomalies

D. Homeister
PDPTA’97
Heidelberg Univ., IWR