Using NVIDIA cuOpt for prescriptive optimization

Nathan Stephens | Arizona State University | RTO Ignite: AI | July 1, 2025

Enterprises Face Challenges Allocating Resources Efficiently

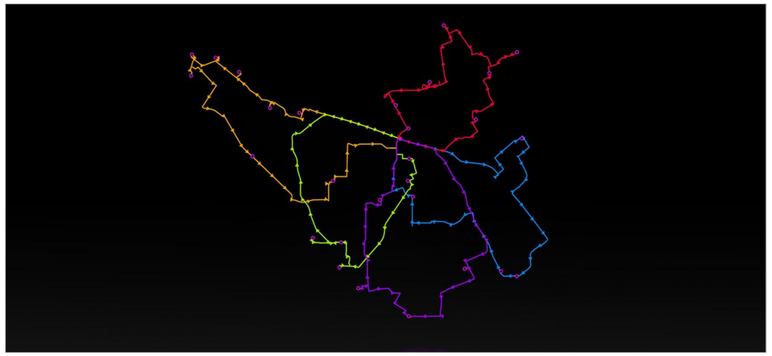
Many decision problems arise as linear/integer/mixed integer optimization problems

Supply Chain Management
Resource Allocation, Inventory Planning, Cost Optimization

Field Dispatch
Multi-variants scheduling

Fleet Management
Multi-constraints Optimization

Warehouse and Factory Robotics
Digital Twins Integration, Intralogistics Routing, Facility Locations



Last Mile Delivery
Dynamic Route Planning

Pick Up and Drop Off
Dispatch Optimization

NVIDIA cuOpt

Open Source GPU-accelerated LP / MIP / VRP Solver Library

What is Available?

- Linear Programming PDLP Solver on GPU with concurrent mode running dual-simplex solver on CPU
- Mixed Integer Programming (MIP) Heuristics on GPU and Branch & Bound on CPU
- Vehicle Routing Problem (VRP) Heuristic Solver on GPU
- Containers on Docker Hub and NGC
- Google Colab and Brev Launchable notebooks

Why Open Source?

- NVIDIA develops optimized CUDA-level code for LP/MIP/VRP solvers
  - accelerated optimization is an emerging field
  - exposing source code allows ease of adoption and further development by our partners
  - leading to enhancing the decision optimization field with new techniques involving GPU parallelization

Visit cuOpt Git Repository for access...

Our Partners

Optimization ecosystem

UNIVERSITÀ DI PAVIA

Deloitte.

cuOpt Benefits

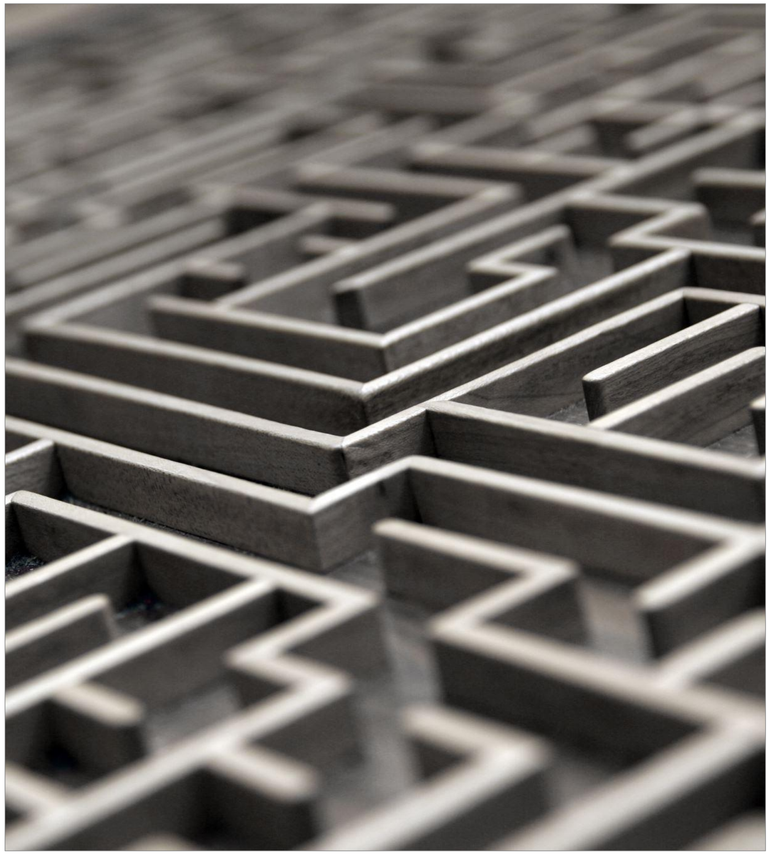
Faster Optimization, Smarter Decisions

Accuracy

World Record Solutions on Leading Li&Lim and Gehring Homberger Vehicle Routing Benchmarks

Speed

Up to 100x faster solution time unlocks new opportunities for dynamic rerouting



Scale

Scale out to 10s of thousands of locations and address use cases not otherwise possible today

AI predicts, cuOpt prescribes

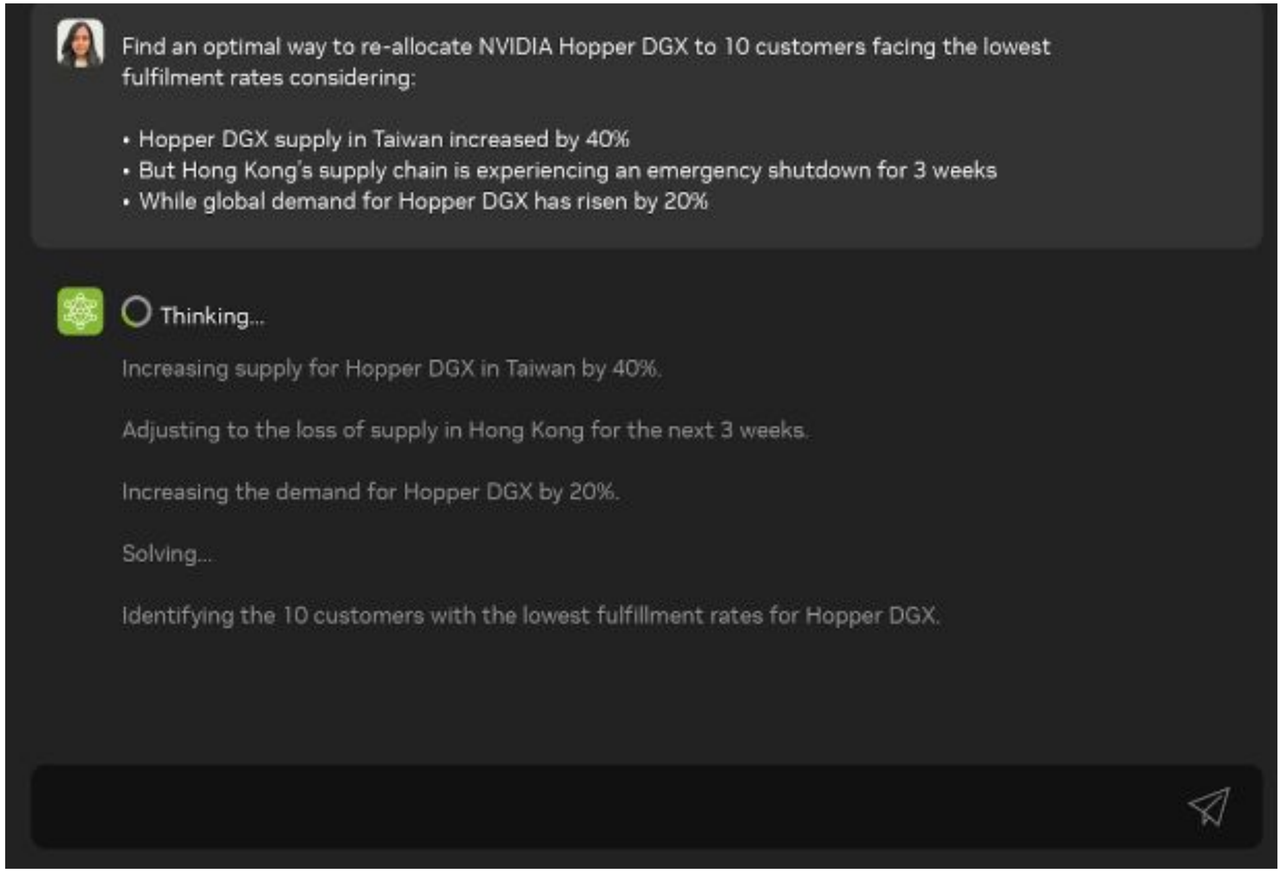
Together they drive decisions

AI is predictive

It forecasts demand, identifies anomalies, estimates delays, or predicts customer behavior — answering “What is likely to happen?”

cuOpt is prescriptive

It uses those AI-generated insights to make real-time, optimized decisions — answering “What should we do about it?”



Examples of AI and cuOpt for decision making

AI predicts, cuOpt prescribes

Problem	AI	cuOpt
VRP – Route Optimization	Predicts delivery delays due to traffic/weather	Reroutes vehicles in real time to minimize impact
VRP – Route Optimization	Forecasts demand surges for online grocery orders	Replans delivery routes and fleet assignments
LP	Predicts inventory shortfalls at regional warehouses	Optimizes product allocation and restocking logistics
LP	Forecasts electricity consumption patterns across the grid	Adjusts power generation and distribution plans
MIP or Dispatch Optimization	Detects likely shift gaps based on historical trends	Re-optimizes workforce schedules and shift assignments
MIP – Job Shop	Predicts machine downtime using sensor data	Reschedules production tasks to maintain throughput
LP/MIP – Resource allocation	Estimates sales spikes for specific SKUs next quarter	Optimizes raw material sourcing and production planning
MIP	Forecasts high risk of order cancellations	Reprioritizes fulfillment and adjusts inventory buffers

How cuOpt Helps

Speed, scale and Accuracy

GPU acceleration helps:

- Massively parallelize solver algorithms to achieve high efficiency and accuracy
- While utilizing NVIDIA GPU libraries, CUDA features and primitives
- On cutting-edge NVIDIA GPUs (eg. NVIDIA H100 and Blackwell)
- Driving down TCO by orders of magnitude

Optimization ISVs and Business Planners can:

- Integrate cuOpt into their solver platform as an accelerator for CPU solver
- Or implement a thin orchestration layer between CPU solver and cuOpt
- Use cuOpt to quickly generate feasible solutions for warm start or for what-if analysis
- Include cuOpt as part of their agentic workflows that talk to LLMs (eg. "talk to your supply chain data")

Massive Parallelization

Finding optimal solutions exponentially faster for specific instances

Traditional CPU-Based Solvers

- Rely on Simplex and Interior Method (Barrier) methods
- That are not easily parallelizable

cuOpt complements CPU-based solvers by taking advantage of

- Most recently emerging First Order Methods (FOM)
- That are **gradient-descent** based and **GPU parallelization**-friendly
- Which also helps with massively speeding up Mixed Integer Programming solvers with additional heuristics
- Leading to ability to take on*:
  - **LP** with up to **74M rows/constraints**, **74M columns/variables** and **1.5B non-zeros** in the constraint matrix
  - **MIP** with up to **3M rows/constraints**, **1.5M columns/variables** and **27M non-zeros** in the constraint matrix

* there is no inherit limit on the size of problems cuOpt can solve except limited by GPU memory and sparsity/density of the underlying matrix. The values presented here are based on H100 GPU.

Cost Reduction

Optimization Is So Hard — cuOpt Does It Better

How cuOpt drives compute cost reduction:

- Up to **5600x** speed up on NVIDIA GPUs
- Reduced unit cost per compute time per solve

Example: cuOpt provides

- With a benchmarked **136x** speed up on H100 GPUs using PDLP Solver
  - On a LP problem* with 8.6M variables, 6.4M constraints and 27.8M nonzeroes
  - CPU** compute time with commercial solver: **2,726,040 seconds** @ $0.035/hr** = $26.50 total cost
  - GPU*** compute time with **cuOpt**: **20,099 seconds** @ $2.49/hr*** = $13.90 total cost **47.5% cost reduction**

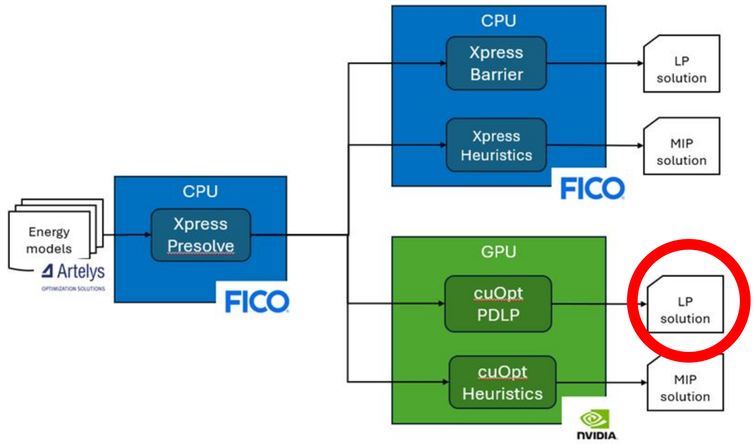
* Mittelmann Benchmark problem thk_48
** AMD EPYC Milan as listed by Coreweave
*** 1x H100 PCIe as listed by Lambda

Supercharging Optimisation at FICO

How Artelys, FICO and NVIDIA cuOpt Join Efforts to Scale Up Energy System Optimisation

The unit commitment problem (UCP) is an optimization problem in power systems that determines which generating units should be turned on or off (committed) and at what power output levels to meet electricity demand at the lowest possible cost, while respecting various operational constraints.

- Models were reduced on CPU
- Optimization with cuOpt was carried on on the GPU
- LP solution for BE 3 instance (the largest and most complex model) - reduced time from 6 hours to 18 minutes (20x speedup)



Instance	CPU Presolve [sec]	CPU Solve Time (Xpress Barrier) [sec]	GPU Solve Time (cuOpt PDLP) [sec]	GPU Solve Time SpeedUp
SE 1	216	94	22	**4x**
ME 2	36	323	229	**1.4x**
BE 3	116	22,450 (~6hrs)	1,124 (~18min)	**20x**

Table 1. Solve time comparisons across large presolved UCP instances.

Note: Test environments:

– FICO Xpress: AWS Graviton 3 CPU, 64 cores, 2.6 GHz, 128GB RAM

– NVIDIA cuOpt: NVIDIA B200 180GB HBM3e GPU, CUDA 12.8

The models used in this study were independently developed by Artelys as part of its internal research, based on publicly available datasets and planning scenarios.

Benchmarks and Performance Results

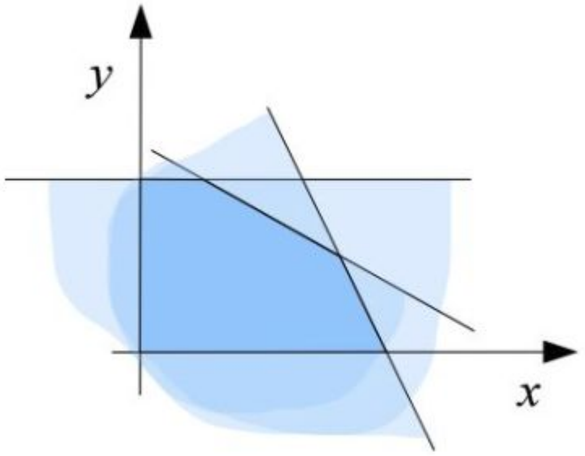
Linear Programming (LP)

A mathematical representation of a real-world problem

"Tell me **how much** of Product X and Product Y I should produce, given my production **constraints** on labor, material, storage space, while **maximizing** profit margin

and tell me **fast**!"

$$\begin{aligned} \max z &= 4x + 5y \\ \text{s.c.} \quad & 2x + y \leq 8 \\ & x + 2y \leq 7 \\ & y \leq 3 \\ & x, y \geq 0 \end{aligned}$$



Well-studied and easy-to-solve, but problem size matters!

May have millions of variables and constraints, impacting solution time

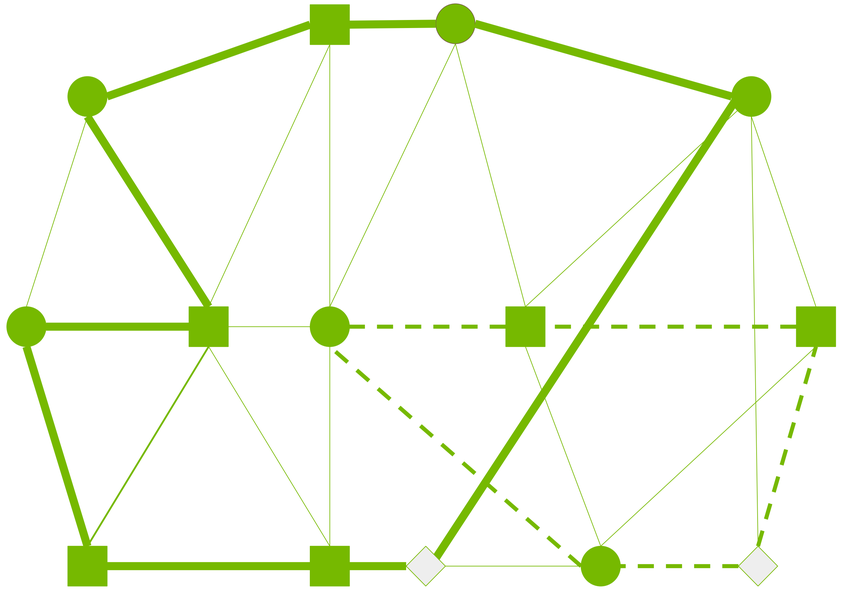
Need efficient, **parallelizable** methods to solve that can also achieve optimality

Also forms the backbone of Mixed Integer Programming solvers

Mixed Integer Programming (MIP)

Yet another type of a mathematical model for real-world problems

"Tell me **which jobs** to be **scheduled** on **which machine** on the shop floor in **what order**, considering **arrival time** and completion **deadline** of each job, **capacity** and **setup time** of the machines as well as the **job types** they can accept, while **maximizing** jobs completed on time and tell me **fast**!"



More complicated with **integer** and **continuous** variables **mixed**

May still have millions of variables and constraints, and can take **VERY** long to solve

Can highly benefit from **GPU Acceleration** while achieving optimal or near-optimal solution quality

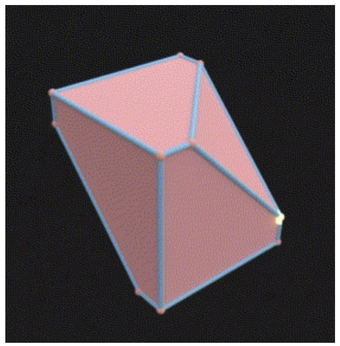
Primal-Dual Hybrid Gradient for LP (PDLP)

Accelerate Large Linear Programming Problems with NVIDIA cuOpt

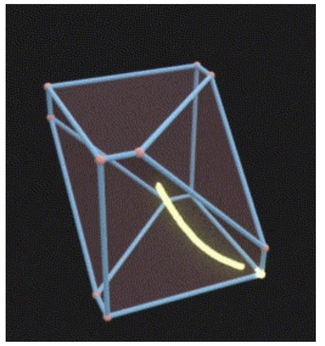
Techniques that face limitations in massive parallelization

- Simplex (1947)
  - Works by following the edges of the feasible region to find the optimum
- IPM - Interior point method (1967)
  - Moves through the interior of the polytope towards the optimum
  - Considered state-of-the-art for solving large-scale LPs on CPUs

Simplex



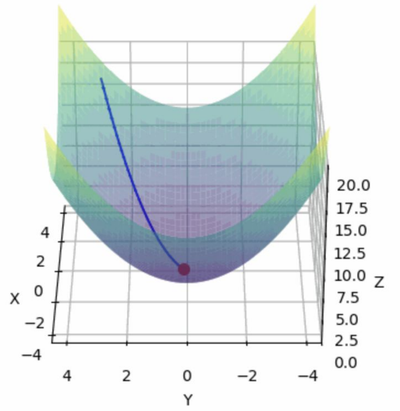
Interior Point Method



Technique that is well-suited for GPU implementation

- PDPL - Primal-Dual Hybrid Gradient (2021)
  - Introduced by the Google Research team
  - Uses the derivative of the problem to iteratively optimize the objective and minimize constraint violation.
  - Employs two highly parallelizable computational patterns:
    - Map operations
    - Sparse matrix-vector multiplications (SpMV)
  - Orders of magnitude faster than CPU implementations

Primal-Dual Hybrid Gradient (PDLP



Primal-Dual Hybrid Gradient for LP (PDLP)

Based on Google's PDLP paper

cuOpt adds:

- Libraries to handle massively sparse matrices on GPU
- cuSparse, Thrust, and RMM
- Highly optimized sorting for calculating duality gaps
- Presolve functionality
- Most importantly: GPU parallelization
- Ability to warm start with feasible solutions
- Outperformed Google's PDLP solver 72x on average, 8x-5600x faster than commercial solver with MCF problems.
- Technical blog: Accelerate Large Linear Programming Problems with NVIDIA cuOpt

Benchmark Results

Mittelmann's Benchmark for PDLP

GPUs are now added to Mittelmann benchmarks for the first time.

The Mittelmann Benchmark measures how fast LP Solvers find an optimal value while respecting the constraints.

The problems represent various scenarios and contain between hundreds of thousands to tens of millions of values.

The following codes were tested:

COPT-7.2.0	COPT
MOSEK-11.0.5	www.mosek.com
HiGHS-1.11.0	HiGHS
KNITRO-14.2.0	www.artelys.com/knitro/
ORTOOLS-9.10	PDLP
XOPT-0.0.8	XOPT
Optverse-1.0.0	huawei.com
cuPDLP-C	cuPDLP-C
cuOpt-25.05	cuOpt

MATLAB has HiGHS as default dual-simplex solver starting with 2024a.

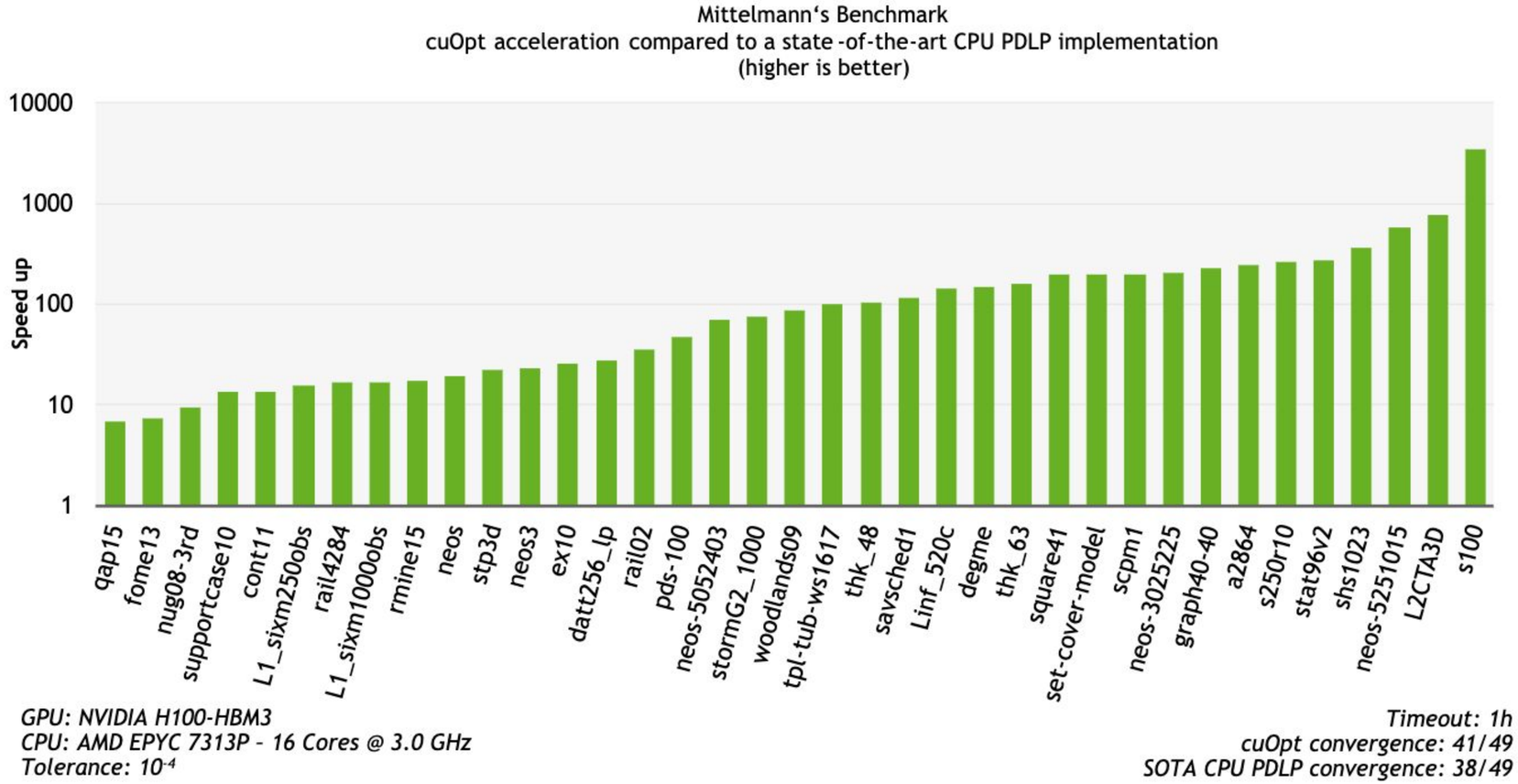
Scaled shifted (by 10 sec) geometric mean of runtimes

unscaled	25.3	83.3	556	546	422	146	33.9	61.4	59.5	45.4	39.0
scaled	1	3.29	22.0	21.6	16.7	5.77	1.54	2.43	2.36	1.80	1.54
solved	65	59	49	49	50	59	65	56	56	57	58

65 probs	COPT	MOSEK	HiGHS	KNITRO	PDLP%	XOPT	OPTV	CUPDL&CUOPT	CUPDL$CUOPT
L1_sixm	2	3	257	t	88	8	3	8	10	5	4
Linf_520c	3	6	t	4858	233	15	4	8	11	6	7
a2864	1	1	262	121	10	1	1	4	1	3	1
bdry2	6	19	t	488	t	146	6	t	t	t	t
cont1	1	1	41	1	256	6	1	228	8	66	7
cont11	1	1	278	2	1988	23	1	144	101	208	59
datt256	1	2	74	6	2	5	1	1	1	1	1
dlr1	27	118	700	2293	t	279	80	2331	t	1808	1123
ex10	1	1	9	7	1	1	1	1	1	1	1
fhnw-bin1	19	47	462	378	3827	86	36	143	210	54	65
fome13	1	1	20	32	10	1	1	1	1	1	1
graph40-40	1	1	6	6	2	1	1	1	1	1	1
irish-e	10	3	22	78	t	12	4	8	6	7	4
neos	5	8	63	252	257	16	5	45	76	59	17
neos3	1	1	16	31	3	1	1	1	1	1	1
neos3025225	8	6	73	33	94	45	8	7	3	4	1
neos5052403	3	4	20	15	13	9	3	3	3	2	1
neos5251015	2	5	102	31	3	14	1	1	1	1	1
ns1687037	3	f	972	14	t	53	7	f	175	t	109
ns1688926	2	1	t	5841	t	47	8	f	105	t	98
nug08-3rd	1	1	170	4406	1	1	2	1	1	1	1
pds-100	13	19	58	631	37	34	16	3	2	1	1
psched3-3	5	17	17	77	5165	15	10	f	t	t	t

PDLP Benchmark Results

Results



37 out of 37 problem classes with 8x-5607x speedup compared to CPU PDLP solver.

cuOpt for Last-Mile Delivery

What is Vehicle Routing Problem?

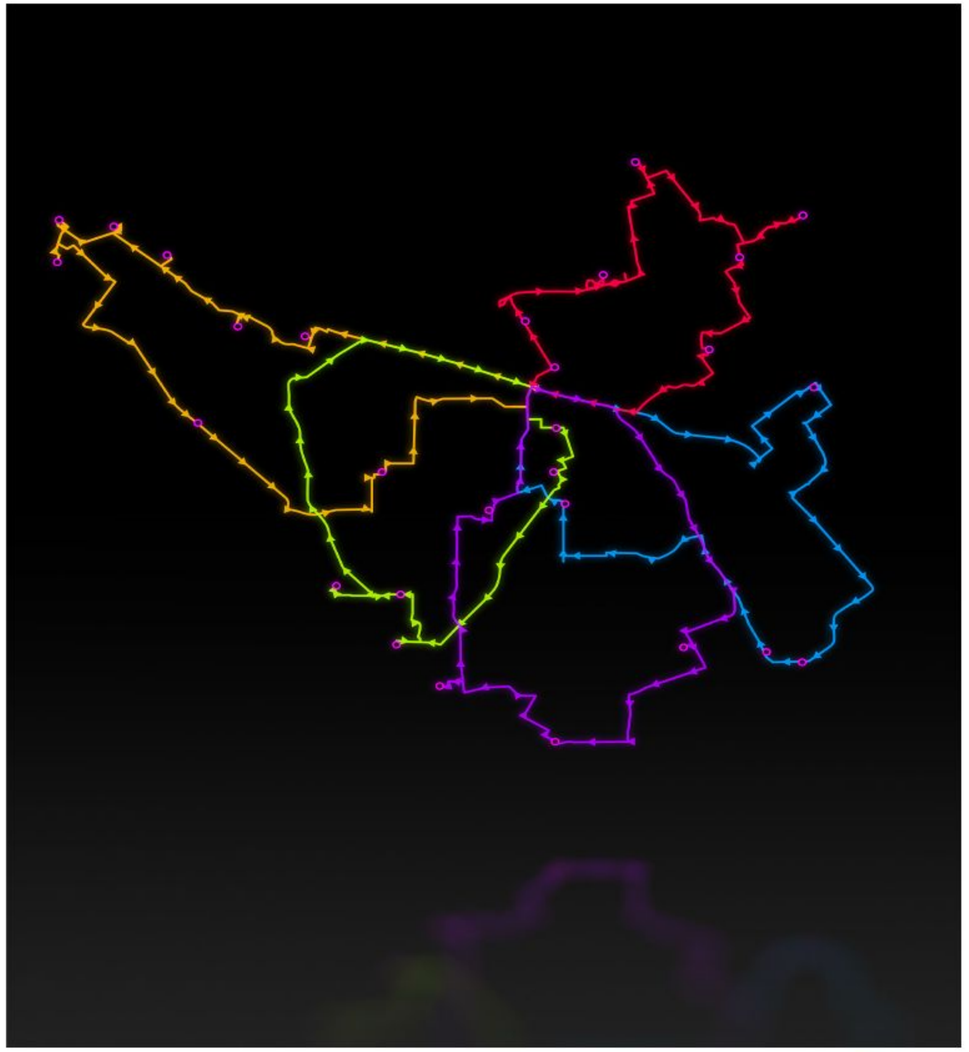
GPU Accelerated solver that uses heuristics to calculate complex vehicle routing problem variants

- "What is the optimal set of routes for a fleet of vehicles to traverse in order to deliver to a given set of customers?“
- With 10 destinations, there can be more than 3,000,000 roundtrip permutations and combinations. With 15 destinations, the number of possible routes could exceed a trillion.

Leverage heuristics on GPU with parallel compute

Accelerated speed and accuracy to deliver dynamic re-optimization

Scale to 1000s of locations



World's Best Accuracy

23 Validated Vehicle Routing and Pickup and Delivery Records

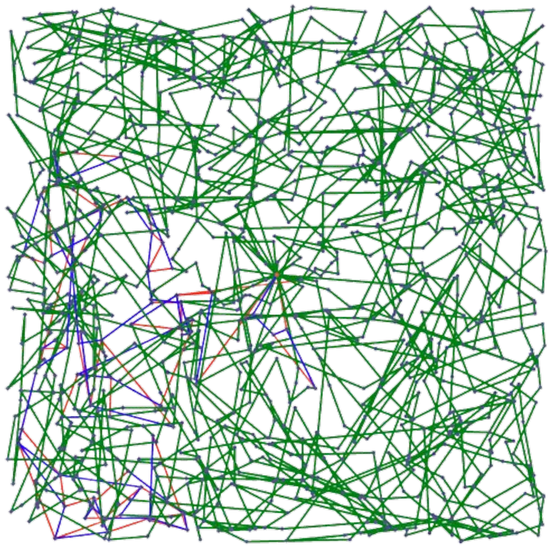
cuOpt Holds **Every** Record Established in the Last 3 Years on the Two Leading

8 Pickup and Delivery Records (Li & Lim); 15 Vehicle Routing Records (Gehring & Homberg)

cuOpt Records Exhibit Meaningfully Novel vs. Improvements in Prior Solution Records

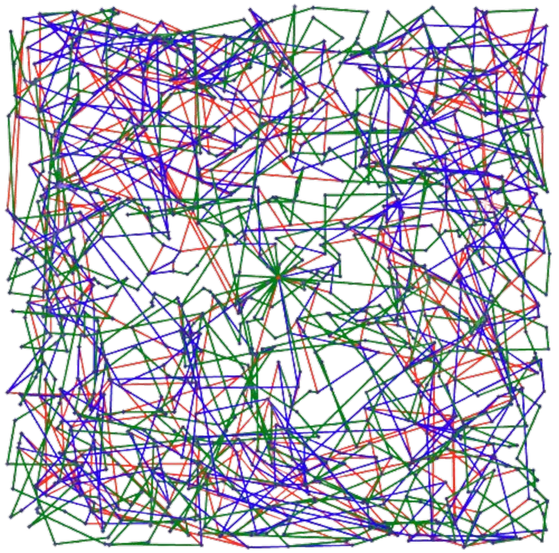
Route differences from prior record solution
(Li & Lim LR2_10_9)

2020 record
(+0.03% improvement over prior record)



- Shared by both solutions
- Only in previous solution
- Only in new solution

2023 cuOpt record
(+0.80% improvement over 2020 record)



Best solutions and datasets available on

Viz credit:

Getting Started Resources

Getting Started with cuOpt

Resources

Product Page | Product Documentation | Quick Start Blog

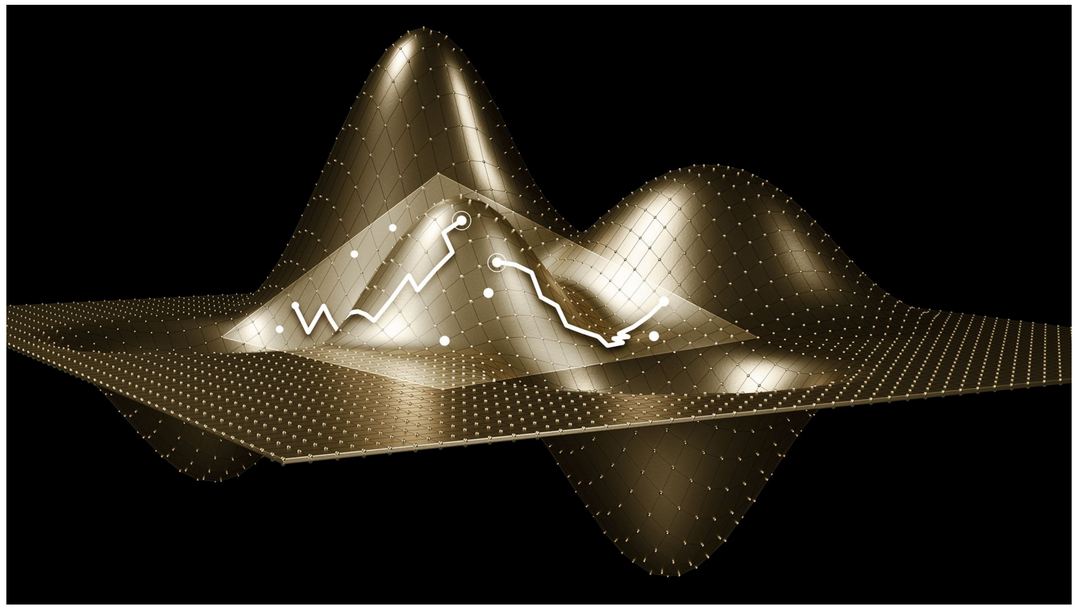
Technical Guidance

cuOpt User Guide

GitHub Resources

cuOpt Open Source Repo

cuOpt Examples Repo



Videos

cuOpt Cloud Service Demo

Talk to Your Supply Chain Demo

YouTube Shorts

Getting Started: NVIDIA cuOpt on Microsoft Azure Marketplace

Getting Started: NVIDIA cuOpt on AWS Marketplace

Blogs

Corp blog: NVIDIA Open-Sources cuOpt, Ushering in New Era of Decision Optimization

Corp blog: Staying in Sync: NVIDIA Combines Digital Twins With Real-Time AI for Industrial Automation

Corp blog: All Aboard: NVIDIA Scores 23 World Records for Route Optimization

Tech blog: Deep Dive: Unveiling the Technical Breakthroughs Behind NVIDIA cuOpt, 23 world record solver

Tech blog: Getting Started with cuOpt

Case Study with Kawasaki Heavy Industries