



What's New in
Gurobi 13.0

- 한국어 세션 -

Where Speed Meets Innovation

한충균
기술담당, Gurobi Optimization

목차



성능 향상



대규모 선형계획법(LPs), PDHG & GPU



비선형 최적화(Nonlinear Optimization)



플랫폼 기능 개선

Gurobi 13: 성능

© 2025 Gurobi Optimization, LLC. Confidential, All Rights Reserved



GUROBI
OPTIMIZATION

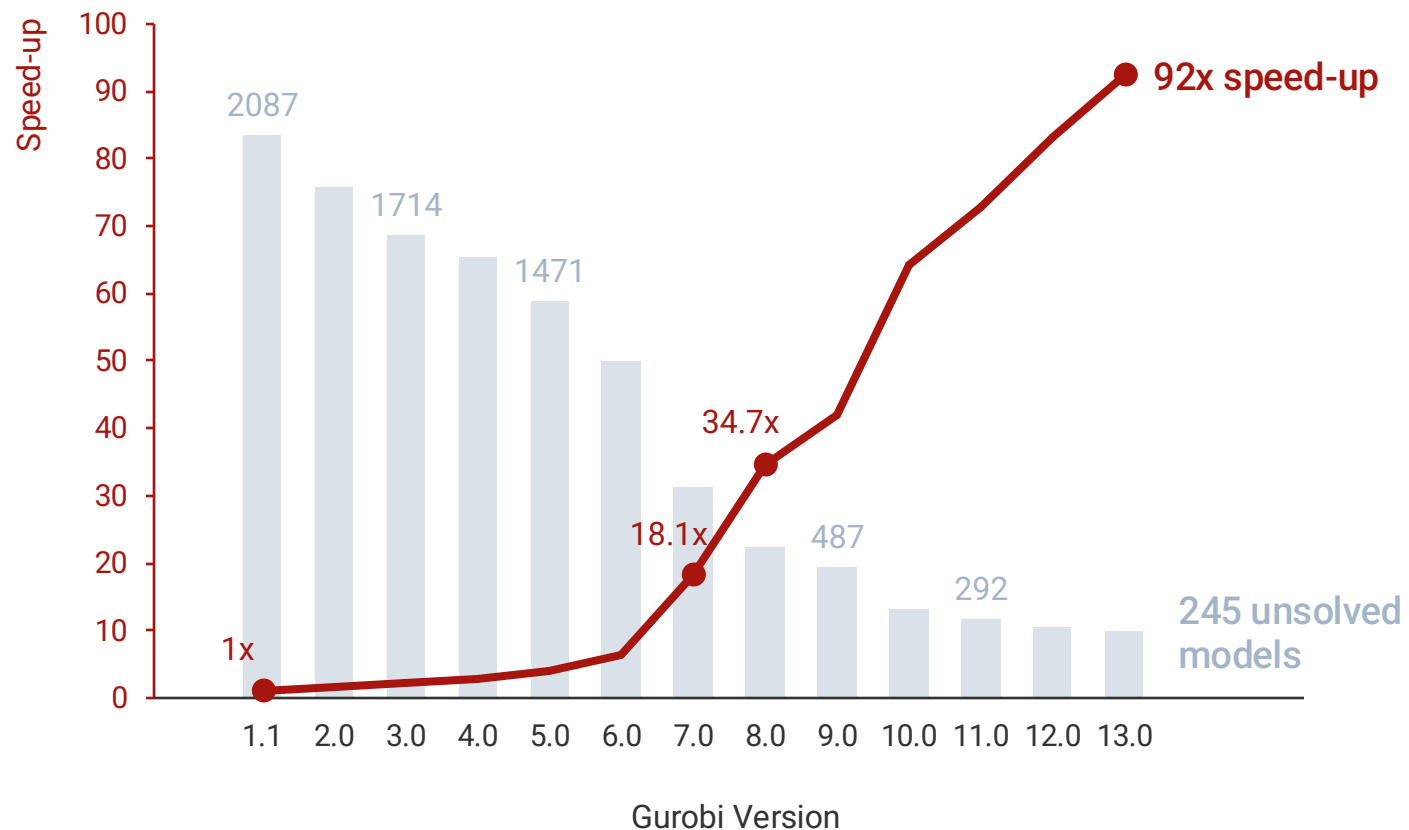
Gurobi 12 vs. 13 비교

문제유형	속도 향상 전체 모델 (>1 초)	속도 향상 어려운 모델 (>100 초)
LP	0.6%	3.9%
MIP	8.2%	15.9%
MIQP	5.1%	7.3%
MIQCP	13.2%	25.8%
Nonconvex MIQCP	54.7%	2.68x
MINLP	2.52x	6.34x*
IIS	67.8%	2.01x*

* 난이도 높은(hard) 모델의 수가 충분하지 않음

Gurobi Version Comparison: Speed and Solvability (PAR-10)

Gurobi MILP Benchmark Suite



MILP 성능 발전

Gurobi의 MILP 벤치마크 세트에서 최신 버전은 다음과 같은 결과를 보여줍니다:

- **92배** 속도 향상 버전 1.1 대비, PAR-10 기준 기하평균(geometric mean) 실행 시간
- 최신 버전에서 245개 모델만이 10,000초 이후에도 미해결 상태로 남아 있음 (최소 하나의 Gurobi 버전에서 풀 수 있는 모든 모델들로 구성되어 있습니다.)

Time limit: 10000 sec.
Intel Xeon CPU E3-1240 v5 @ 3.50GHz
4 cores, 8 hyper-threads
32 GB RAM

Test set has 9423 models:
- 960 discarded due to inconsistent answers
- 2641 discarded that none of the versions can solve
- speed-up measured on >100s bracket: 3517 models

MIP 향상 사항

• Presolve

- Aggregator
 - 수치 안정성 개선
 - LU 분해 (LU factorization) 기반의 새로운 접근 방식
 - 공격적인 등식 집계 (Aggressive equality aggregation)
- 암시적 자유변수 (Implied free variable) 기반 영(0) 계수 제거

• Cutting planes

- Master knapsack cuts 추가
- 보다 공격적인 Gomory cuts
- 향상된 MIR cuts

• Disconnected component 개선

- 메모리 사용량을 줄이기 위한 코드 재구성
- 부분 해 또는 무해(no solution) 상태에서도 동작 가능
- 컴포넌트 단위 풀이에서 presolve 활성화

• Propagation 개선

- non-binary 영역에서의 과도한 propagation 방지

• Improvement heuristics 에서 비실행가능 해(infeasible vector)를 시작점으로 사용

• NoRel 휴리스틱 개선

• On-Off 제약 감지

- 새로운 휴리스틱
- 경계 개선을 위한 OBBT 적용

• 비활성 제약 감지

• Simplex

- Primal ratio 테스트 버그 수정
- Dual ratio 테스트 개선
- Presolve 단계에서 더 나은 Bound Propagation

• 일반 MIP 프레임워크 개선

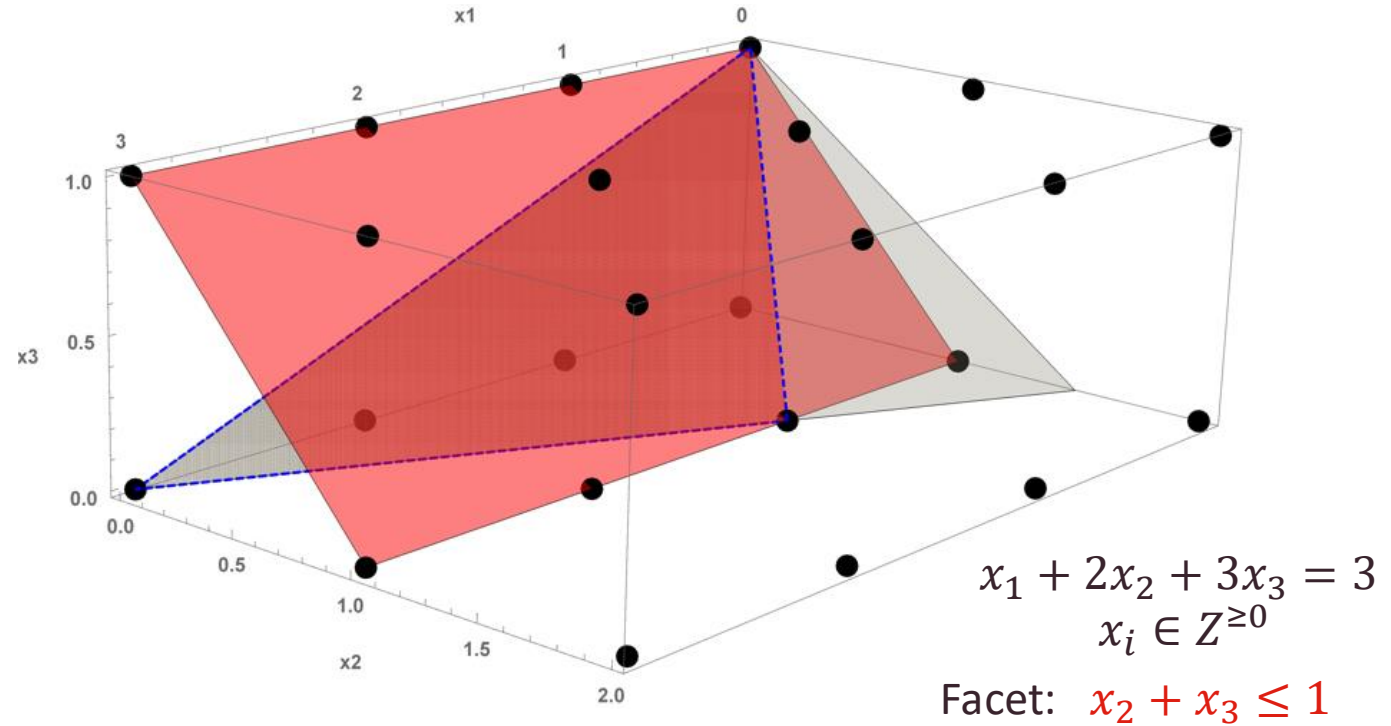
- Degeneracy 처리 코드 개선
- 병렬화(parallelization) 개선

Master Knapsack Cuts

- Master Knapsack 구조에서 유도되는 강력한 cut
- 아래와 같은 단일 제약 이완 (single-row relaxation) 으로부터 생성

$$\sum_{i=1}^n ix_i = n, x_i \in Z, x_i \geq 0$$

- 단순한 형태의 컷이며, 계수가 단순한 유리수 형태여서 수치적으로 안정적
- 관련 모델에서 평균 2.5% 성능 향상 (100초 이상 걸리는 모델에서는 약 10% 향상)



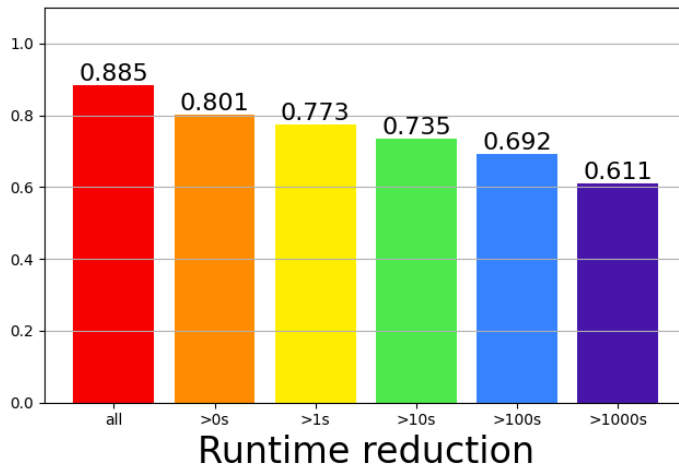
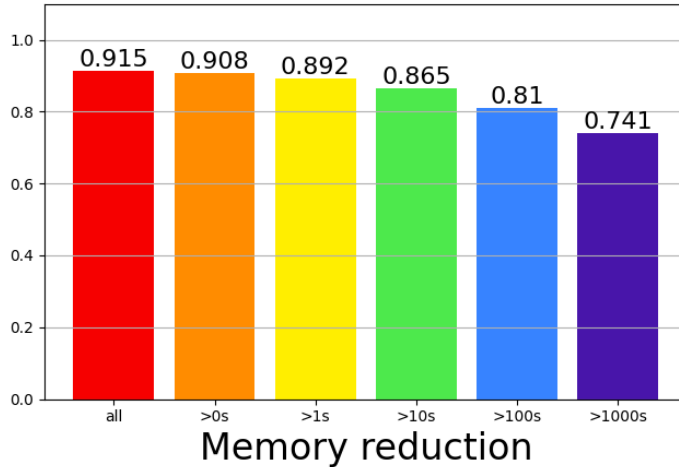
Example in higher dimension

$$x_1 + 2x_2 + 3x_3 + 4x_4 + 5x_5 + 6x_6 + 7x_7 + 8x_8 = 8, \quad x_i \in Z^{\geq 0}$$

$$\text{Facet: } \frac{1}{4}x_2 + \frac{1}{4}x_3 + \frac{1}{2}x_4 + \frac{3}{4}x_5 + \frac{3}{4}x_6 + x_7 + x_8 \leq 1$$

733 409 357 260 182 80

Instance count



Disconnected Components

Disconnected component 처리 방식 새롭게 재설계

- 메모리 사용량(memory footprint)을 줄임
- 수치적으로 더 견고(numerically robust)하도록 개선
- 전체 MIP 벤치마크 세트 중 단 2.7%만 영향을 받음

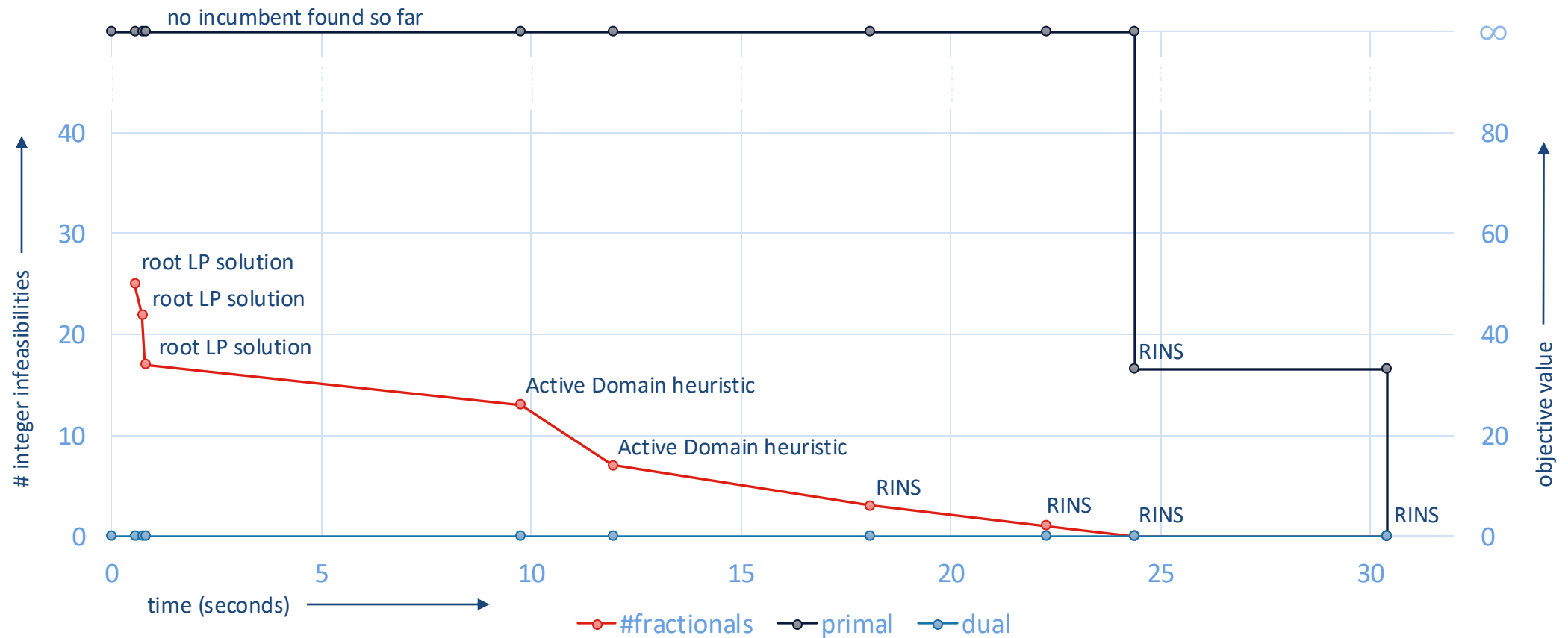
특정 고객 사례에서의 성능 개선:

- 메모리 사용량: 19% 감소
- 실행 시간: 30% 감소

비실행가능 해의 활용 in improvement heuristics

- RINS와 같은 improvement heuristics 은 보통 실행가능 해(feasible solution) 또는 현재 최선해(incumbent solution)가 필요함
- 아이디어: 현재 최선해가 없을 경우,
비실행가능 정수해(infeasible integer solution)를 대신 사용
 - root cut loop LP 해를 반올림한 값
 - 수치적 문제로 인해 거부된 휴리스틱 해
 - dual presolve에서 제거된 원래 모델의 해 (MIP start 등)
 - 휴리스틱 내부 sub-MIP 풀이에서 얻어진 비실행가능 해
 - 사용자가 제공한 infeasible MIP start
- 비실행가능 해는 다음과 같은 휴리스틱에서 활용됨:
 - RINS와 같은 improvement heuristics
 - NoRel heuristic
 - Partition heuristic
- 성능(100초 이상 걸리는 모델 기준):
 - 최적성 증명 시간(time to proven optimality) 5% 단축
 - 첫 실행가능 해 발견 시간(time to first feasible solution) 11% 단축

neos-5074751-dziwna



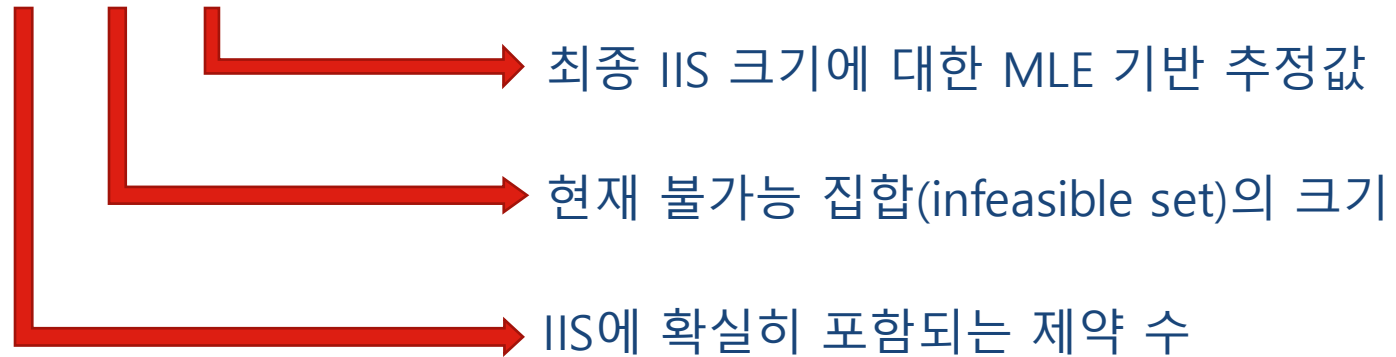
* IIS: Irreducible Inconsistent Subset

IIS 성능

- Gurobi 9.1부터 IIS 크기를 추정하기 시작
 - 초기에는 단순한 근사 사용
- Gurobi 10.0부터 최대우도추정 (MLE, Maximum Likelihood Estimate) 기반 추정 도입
- Gurobi 13의 새로운 기능: MLE를 사용하여 어떤 IIS 기본 절차를 적용할지 판단
 - subset delete
 - IIS에서 제거 가능한 제약이 많을 때 빠르게 동작
 - parallel delete
 - IIS에서 제거 가능한 제약이 거의 없을 때 빠르게 동작
 - single delete
 - 느리지만 항상 성공하는 방식
- 성능 v12 vs. v13: +67.8%

Computing Irreducible Inconsistent Subsystem (IIS)...

Constraints		Bounds		Runtime		
Min	Max	Guess	Min	Max	Guess	
0	37698	-	0	0	0	0s
0	18626	-	0	0	0	23s
0	18606	-	0	0	0	25s
0	15405	40	0	0	0	30s
0	12458	49	0	0	0	35s
0	8056	41	0	0	0	40s
0	4014	46	0	0	0	45s
2	168	44	0	0	0	50s
5	145	45	0	0	0	55s
14	67	43	0	0	0	60s
30	43	43	0	0	0	65s
43	43	43	0	0	0	67s



대규모 선형계획법(LPs), PDHG & GPU

© 2025 Gurobi Optimization, LLC. Confidential, All Rights Reserved



GUROBI
OPTIMIZATION

LP 알고리즘의 새로운 경쟁자

- 기존 LP 알고리즘
 - Primal Simplex, dual Simplex
 - 여러 확장을 포함한 Barrier Method
- 1차 방법(First-order methods)
 - 최근 몇 년간 점점 주목을 받아 옴
 - GPU 덕분에
가시성과 매력도가 크게 증가함



PDHG - 1차 최적화 방법

간단한 소개

Primal

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

Dual

$$\begin{aligned} \max \quad & b^T y \\ \text{s.t.} \quad & A^T y \leq c \end{aligned}$$

이 문제들은 다음과 같은 '안장점 문제(Saddle-point problem)'를 풀어서 해결할 수 있음

$$\min_x \max_y L(x, y) = c^T x - y^T Ax + b^T y$$

- Chambolle, A. and Pock, T., 2011. A first-order primal-dual algorithm for convex problems with applications to imaging. Journal of mathematical imaging and vision, 40, pp.120-145.

PDHG - 1차 최적화 방법

간단한 소개

Saddle-point 문제

$$\min_x \max_y L(x, y) = c^T x - y^T A x + b^T y$$

이 문제는 **Primal-Dual Hybrid Gradient (PDHG)** 방법을 사용해 x_k 와 y_k 반복적으로 계산함으로써 해결할 수 있음 (단, $\tau\sigma < 1/\|A\|_2^2$ 조건을 만족해야 함)

$$\nabla_x L(x, y) = c - A^T y$$

$$\nabla_y L(x, y) = b - A x$$

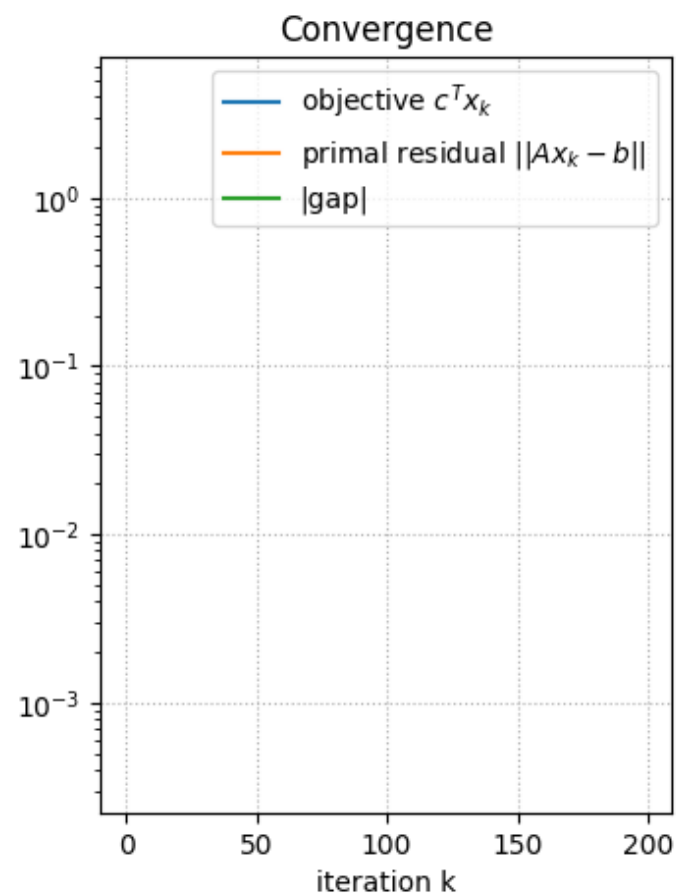
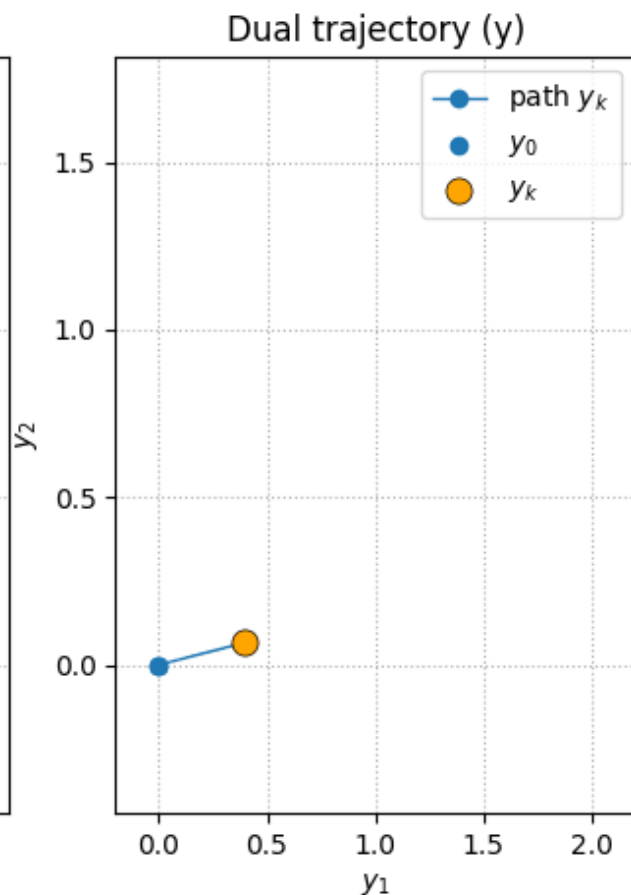
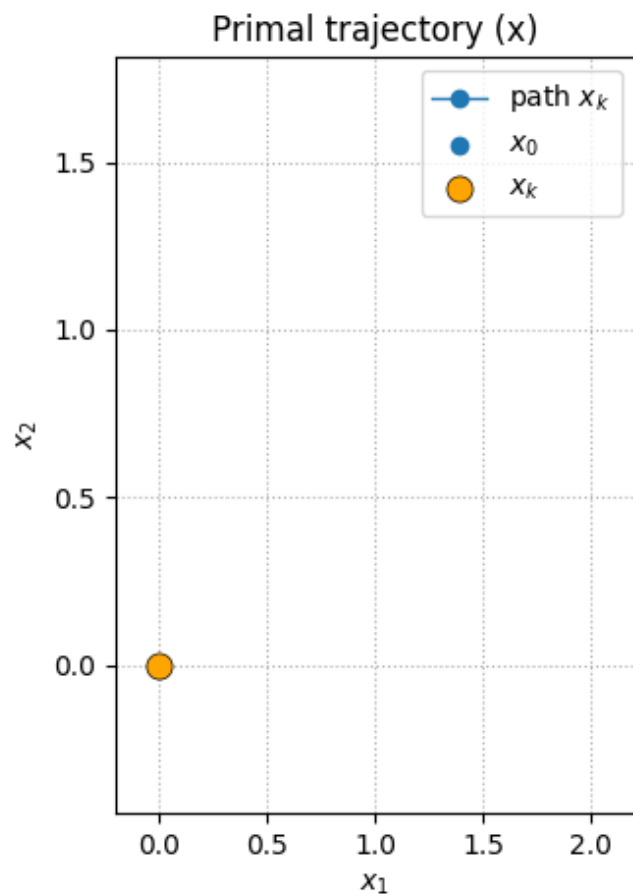
$$x_{k+1} = \text{proj}_{R_+^n}(x_k - \tau(c - A^T y_k))$$

$$y_{k+1} = y_k + \sigma(b - A(2x_{k+1} - x_k))$$

- 여기서 가장 계산 비용이 큰 연산은 두 개의 **행렬-벡터 곱**, 즉 $A^T y_k$ 와 $A x$ 임
- 행렬-벡터 곱은 메모리 접근 속도 의존적 연산이지만,
GPU에서는 이러한 연산이 매우 잘 스케일링됨
- 따라서 PDHG는 GPU 환경과 매우 잘 맞는 알고리즘 임

PDHG 의 동작 패턴

$$\begin{aligned} \min \quad & x_1 + 0.5x_2 \\ \text{s.t.} \quad & x_1 + 2x_2 = 3 \\ & x_1 - x_2 = 0.5 \end{aligned}$$



대규모 LP 를 풀기위한 PDHG의 개선

- PDHG 알고리즘이 실제 LP(PDLP)를 풀 수 있도록 해준 다양한 알고리즘적 구성 요소들
 - 적응형 스텝 크기 (Adaptive step size)
 - 재시작(Restarts, 다양한 기준 기반)
 - Primal 가중치 업데이트

- Applegate, D., Díaz, M., Hinder, O., Lu, H., Lubin, M., O'Donoghue, B. and Schudy, W., 2021. Practical large-scale linear programming using primal-dual hybrid gradient. *Advances in Neural Information Processing Systems*, 34, pp.20243-20257.
- Lu, H., Peng, Z. and Yang, J., 2025. [cuPDLpx: A further enhanced GPU-based first-order solver for linear programming](#). arXiv preprint arXiv:2507.14051.

PDHG – 종료 기준(Termination Criteria)

상대적 종료(Relative Termination)

- Primal residual (새로운 파라미터 PDHGRelTol 로 제어)

$$\|b - Ax\|_2 \leq \epsilon_{rel}(1 + \|b\|_2)$$

- Dual residual (새로운 파라미터 PDHGRelTol 로 제어)

$$\|c - A^T y - \lambda\|_2 \leq \epsilon_{rel}(1 + \|c\|_2)$$

- Objective gap (새로운 파라미터 PDHGConvTol 로 제어)

$$|b^T y - c^T x| \leq \epsilon_{conv}(1 + |b^T y| + |c^T x|)$$

- 문제 데이터에 **상대적인** 기준임! → 큰 위반이 발생할 수 있음 ⚠
 - 대부분의 PDHG 기반 솔버는 상대적 종료 기준만 사용함
 - PDLP 솔버 벤치마킹 시 해의 품질을 주의해야 함 ⚠

절대적 종료(Absolute Termination)

- Gurobi는 다른 알고리즘과 마찬가지로 절대 허용오차(absolute tolerance)도 적용할 수 있음
- 파라미터 PDHGAbsTol로 제어 가능
- 절대 종료 기준은 보통 crossover 과정에서 충족되며, 이때 기초해(basic solution)도 생성됨

모델 예시: rwth-timetable

Defaults on GH200



Gurobi Optimizer version 13.0.0 build v13.0.0beta1 (armlinux64gpu)
Copyright (c) 2025, Gurobi Optimization, LLC

CPU model: ARM64

Thread count: 72 physical cores, 72 logical processors, using up to 32 threads

Optimize a model with 440134 rows, 923564 columns and 4510786 nonzeros (Min)

Model fingerprint: 0xf4eab31e

Model has 834949 linear objective coefficients

Coefficient statistics:

Matrix range [1e+00, 2e+00]
Objective range [1e+00, 4e+05]
Bounds range [1e+00, 1e+01]
RHS range [1e+00, 3e+00]

Presolve removed 206440 rows and 549304 columns

Presolve time: 3.14s

Presolved: 233694 rows, 374260 columns, 3675892 nonzeros

Concurrent LP optimizer: primal simplex, dual simplex, and barrier

Showing barrier log only...

...

Ordering time: 69.98s

Barrier statistics:

Dense cols : 385

AA' NZ : 2.565e+07

Factor NZ : 1.273e+09 (roughly 10.0 GB of memory)

Factor Ops : 1.885e+13 (roughly 35 seconds per iteration)

Threads : 30

...

...

63 7.15260973e+05 7.15260972e+05 4.57e-09 7.28e-11 1.59e-09 855s

Barrier solved model in 63 iterations and 855.23 seconds

Optimal objective 7.15260973e+05

Crossover log...

50685 DPushes remaining with DInf 9.9961285e-02 857s

11303 DPushes remaining with DInf 4.4423273e-02 860s

0 DPushes remaining with DInf 0.0000000e+00 864s

32095 PPushes remaining with PInf 1.5185350e-06 864s

4650 PPushes remaining with PInf 0.0000000e+00 865s

0 PPushes remaining with PInf 1.5402888e+00 866s

Push phase complete: Pinf 1.5402888e+00, Dinf 1.9161667e+01 866s

Iteration	Objective	Primal Inf.	Dual Inf.	Time
-----------	-----------	-------------	-----------	------

74367	7.1526097e+05	0.000000e+00	1.916167e+01	866s
-------	---------------	--------------	--------------	------

Crossover time: 11.23 seconds (18.50 work units)

Solved with barrier

Extra simplex iterations after uncrush: 19

74434	7.1526097e+05	0.000000e+00	0.000000e+00	867s
-------	---------------	--------------	--------------	------

Solved in 74434 iterations and 866.95 seconds (3307.94 work units)

Optimal objective 7.152609726e+05

모델 예시: rwth-timetable

PDHG on CPU



Non-default parameters:

Method 6

...

Start PDHG using 32 threads

Iter	Objective		Residual		Compl	Time
	Primal	Dual	Primal	Dual		
0	2.43298400e+07	-1.18242230e+07	2.00e+00	0.00e+00	6.45e+01	3s
2603	7.18700010e+05	7.16453294e+05	2.06e-02	6.16e+01	1.06e-02	5s
10303	7.15956714e+05	7.15075108e+05	6.23e-03	6.31e-02	1.06e-03	10s
17903	7.15652265e+05	7.15076646e+05	2.07e-03	1.26e-01	7.39e-04	15s
25603	7.15111796e+05	7.15081045e+05	2.00e-03	1.60e-01	1.02e-04	20s
33303	7.15052264e+05	7.15090353e+05	2.00e-03	4.06e-01	2.19e-05	25s
41003	7.15052801e+05	7.15102002e+05	2.00e-03	4.06e-01	1.37e-05	30s
48703	7.15094111e+05	7.15201781e+05	2.00e-03	3.23e+00	2.84e-05	35s
56303	7.15261014e+05	7.15248962e+05	2.62e-07	1.19e+01	9.63e-04	40s
64003	7.15261011e+05	7.15258972e+05	3.41e-08	2.81e+00	2.14e-04	45s
70403	7.15261010e+05	7.15261265e+05	1.33e-07	5.20e-01	6.28e-05	49s

PDHG solved model in 70403 iterations and 49.21 seconds (60.59 work units)

Optimal objective 7.15261010e+05

...

Crossover log...

```
60385 DPushes remaining with DInf 7.8870150e-02 49s
3665 DPushes remaining with DInf 3.9477411e+01 50s
0 DPushes remaining with DInf 1.2871262e+02 55s
```

```
26153 PPushes remaining with PInf 1.5976922e-02 55s
19432 PPushes remaining with PInf 1.6360125e-02 55s
0 PPushes remaining with PInf 0.0000000e+00 57s
```

Push phase complete: Pinf 0.0000000e+00, Dinf 1.9323679e+03 57s

Iteration	Objective	Primal Inf.	Dual Inf.	Time
77774	7.1526099e+05	0.0000000e+00	1.932368e+03	57s
78514	7.1526097e+05	0.0000000e+00	6.154455e+01	60s

Crossover time: 12.80 seconds (17.84 work units)

Extra simplex iterations after uncrush: 17

```
79005 7.1526097e+05 0.0000000e+00 0.0000000e+00 62s
```

Solved in 79005 iterations and 62.46 seconds (78.92 work units)

Optimal objective 7.152609726e+05

모델 예시: rwth-timetable

PDHG on GPU



Non-default parameters:

Method 6

PDHGGPU 1

Optimize a model with 440134 rows, 923564 columns and 4510786 nonzeros (Min)

Model fingerprint: 0xf4eab31e

Model has 834949 linear objective coefficients

Coefficient statistics:

Matrix range [1e+00, 2e+00]

Objective range [1e+00, 4e+05]

Bounds range [1e+00, 1e+01]

RHS range [1e+00, 3e+00]

Presolve removed 206440 rows and 549304 columns

Presolve time: 3.06s

Presolved: 233694 rows, 374260 columns, 3675892 nonzeros

Start PDHG on GPU

Iter	Objective		Residual		Compl	Time
	Primal	Dual	Primal	Dual		
0	2.43298400e+07	-1.18242230e+07	2.00e+00	0.00e+00	6.45e+01	4s
13104	7.15852435e+05	7.15074867e+05	3.44e-03	3.59e-02	9.59e-04	5s
63304	7.15261009e+05	7.15260533e+05	9.93e-08	5.12e-01	9.59e-05	10s
65504	7.15261011e+05	7.15261180e+05	8.11e-08	3.10e-01	6.43e-05	10s

PDHG solved model in 65504 iterations and 10.22 seconds (246.44 work units)

Optimal objective 7.15261011e+05

...

Crossover log...

60163 DPushes remaining with DInf 8.9044933e-02 10s
0 DPushes remaining with DInf 2.1340349e+01 15s

26143 PPushes remaining with PInf 8.0291180e-03 15s
0 PPushes remaining with PInf 0.0000000e+00 17s

Push phase complete: Pinf 0.0000000e+00, Dinf 9.1626111e+02 17s

Iteration	Objective	Primal Inf.	Dual Inf.	Time
77513	7.1526100e+05	0.000000e+00	9.162611e+02	17s
78213	7.1526098e+05	0.000000e+00	7.924577e+01	20s

Crossover time: 12.68 seconds (18.17 work units)

Extra simplex iterations after uncrush: 17

78771 7.1526097e+05 0.000000e+00 0.000000e+00 23s

Solved in 78771 iterations and 23.36 seconds (265.10 work units)

Optimal objective 7.152609726e+05

속도 향상: 867초 -> 62.5초 (**14X** with Method=6) -> 23.4초 (**37X** with Method=6 PDHGGPU=1) 🤖

테스트 세트 및 결과

- 현 벤치마크 세트는 2000개 이상의 모델로 구성, 대부분은 실제 고객 모델임
- 해당 세트는 *현재 Simplex나 Barrier로 해결 가능한 모델들에 편향되어 있음*
 - 이 세트에서는 PDHG가 기본 LP 알고리즘에 비해 불리하게 보일 수 있음
- 테스트 세트를 다음과 같이 필터링함:
Simplex나 Barrier로 10,000초 안에 풀 수 없는 모델들만 선택
→ 모델 475개로 테스트 세트 구성
- 93개는 PDHG/GPU(GH200)로 풀 때
MacBook Pro M4에서 어떤 LP 알고리즘보다도 더 빠르게 해결됨
- PDHG는 기존 기법으로는 풀기 어려운 LP들을 해결 가능
→ PDHG는 기존 알고리즘 포트폴리오를 보완하는 역할

v13 의 업데이트 사항

- CPU 기반 PDHG 구현 제공, Method=6으로 설정하여 사용
- GPU 기반 PDHG는 별도 다운로드 제공
 - 추가로 PDHGGPU=1 설정 필요
- 기초 최적해(basic optimal solution)를 얻기 위해 crossover 와 통합됨
- LP warm start 도 지원

- PDHG/GPU는 13.0.0에서 "beta" 기능
- 13.0.1에서는 "preview" 기능으로 전환될 예정

* Beta: 아직 실사용(production) 용도는 아니며, 제한된 지원만 제공

** Preview: 정식 지원되지만, 향후 버전에서 큰 변경(API·동작·패키징 등) 가능

적합한 사용자 유형

- 모델 크기가 큰* 가요?
- Root relaxation 에 어려움을 겪는 LP/MIP인가요?
- Barrier 로 풀기 어려운 모델인가요?
- Crossover=0 옵션으로 Barrier 를 자주 사용하나요?
- (성능이 좋은) GPU 머신이 있나요?



*큰 모델 = presolve 이후 비제로(nonzeros) 항이 10^5 개 이상

비선형 최적화(Nonlinear Optimization)

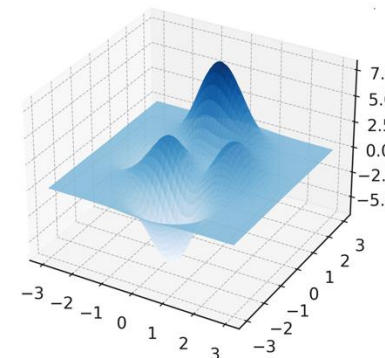
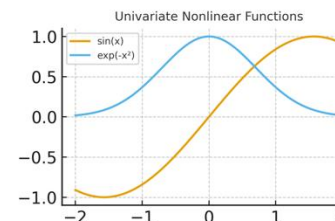
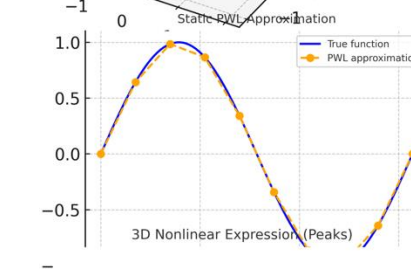
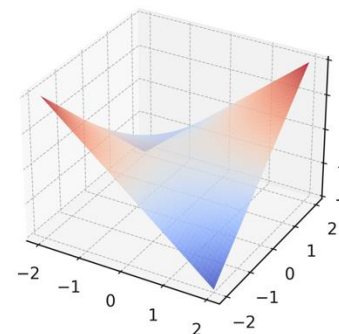
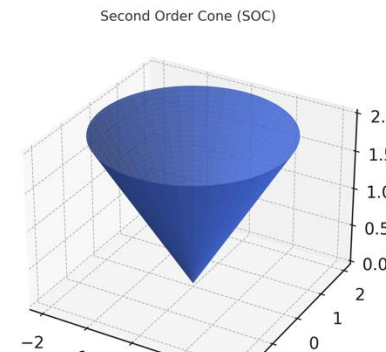
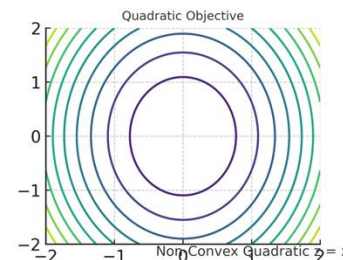
© 2025 Gurobi Optimization, LLC. Confidential, All Rights Reserved



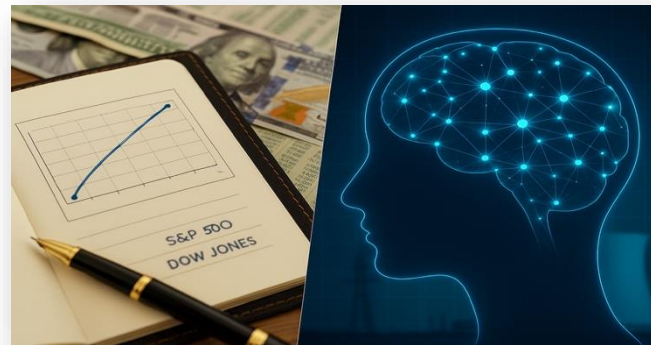
GUROBI
OPTIMIZATION

Gurobi의 비선형(nonlinear) 지원 발전사

버전	비선형 구성요소	모델 종류	알고리즘
4.0	Quadratic Objective	QP, MIQP	Simplex, Barrier, B&B
5.0	Convex Quadratic Constraints (SOC)	QCP, SOCP, MIQCP	Barrier, B&B, OA
9.0	Non-Convex Quadratic	NCQCP, MINCQCP	Spatial B&B
9.0	Static 1-D PWL	MILP	LP based B&B
11.0	Univariate Functions	MINLP	Spatial B&B
12.0	Nonlinear Expressions	MINLP	Spatial B&B



왜 비선형 모델이 필요한가?



분야	응용	현상	비선형성
금융	포트폴리오 최적화	위험/변동성	Convex Quadratic
물리	Truss Topology Optimization	물리적 힘	Convex Quadratic
석유·화학	풀링(Pooling)	제품 혼합	Non-Convex Quadratic
에너지	ACOPF	교류(AC) 전류	Cos/Sin or products
가스/수도망	네트워크 최적화	파이프 내 흐름	Signpower: $\text{sign}(x) \cdot x ^a$
Machine Learning	--	활성화 함수 (Activation)	Logistic, tanh



비선형 제약 API (Gurobi ≥ 12)

다음 형태의 비선형 제약을 표현함

$$y = f(\mathbf{x}), \mathbf{x} \in \mathbb{R}^n$$

f 는 아래 연산들로 구성될 수 있음

- $+, *, -, /$
- $x^a, a^x, \exp x$
- $\log(x), \log_2(x), \log_{10}(x)$
- $\sin(x), \cos(x), \tan(x)$
- $\text{Logistic}(x)$
- $\text{tanh}(x)$
- $\text{signpow}(x, a) = \text{sign}(x) \cdot |x|^a, a \in \mathbb{R}_{\geq 1}$

Added in Gurobi 13

$$y = \sin x_1 \cdot x_2 + \frac{\sin x_1}{\exp x_2}$$

```
from gurobipy.nfunc import sin,
exp

model.addConstr(
    y == sin(x1) * x2 +
    sin(x1)/exp(x2)
)
```

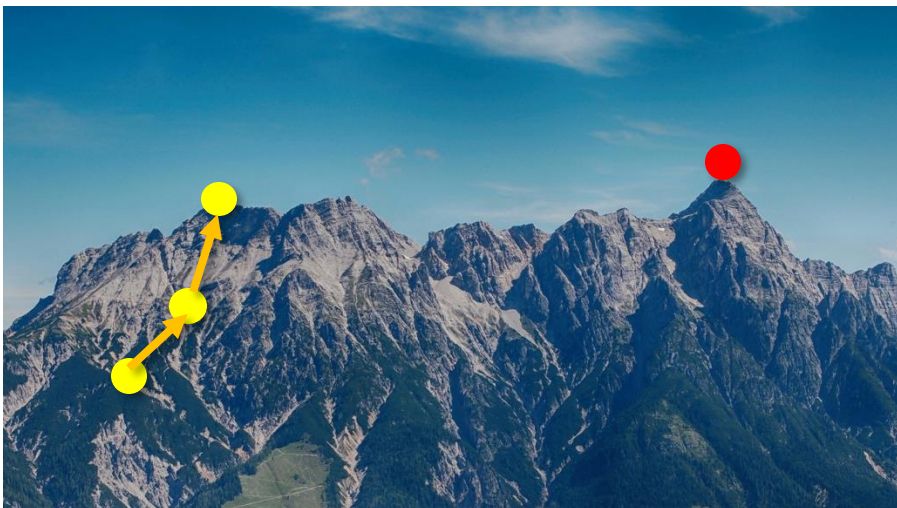


비선형 배리어 알고리즘 (Nonlinear Barrier Algorithm)



비볼록 최적화(Non-Convex Optimization)

비볼록 영역(Non-Convex Region)



- 높은 지점을 찾기
 - 임의의 지점에서 시작
 - 실행가능 영역(feasible region) 안에서 개선되는 방향으로 이동
 - 국소 최적해(locally optimal solution)로 수렴
- 전역 최적해를 찾으려면 분할-정복(divide-and-conquer) 기반 알고리즘이 필요
 - 하지만 조합폭발(combinatorial explosion)이 문제임

국소해(Local) vs 전역해(Global)

- 국소 최적해를 찾는 것은 전역 최적성을 증명하는 것보다 "쉬운 편"
- 하지만, 실제로 "쉽다"는 의미는 아님:
 - 이론적으로 NP-hard
 - 문제 규모가 (매우) 클 수 있음
 - 매우 비선형적(non-linear) 구조를 가질 수 있음
- 국소 최적해를 찾는 알고리즘들은 일반적으로 비선형 최적화 영역에 속함
- 높은 정확도(high accuracy)의 해가 필수적; 즉, 비실현성이 매우 작아야 함(small infeasibility)

Gurobi 의 비선형 배리어(Nonlinear Barrier)

버전	비선형 요소	문제 유형	NL Barrier
4.0	Quadratic Objective	QP, MIQP	
5.0	Convex Quadratic constraints (SOC)	QCP, SOCP, MIQCP	
9.0	Non-Convex Quadratic	NCQCP, MINCQCP	9.5에서 <i>NLPheur</i> 휴리스틱에 도입
9.0	Static PWL approximations	MILP	
11.0	Univariate function	MINLP	단변수(univariate) 비선형 함수에서 동작
12.0	Nonlinear Expressions	MINLP	
13.0	Nonlinear Expressions	NLP	직접 호출 가능



Dr. Andreas Wächter

배리어 알고리즘(Barrier Algorithm)

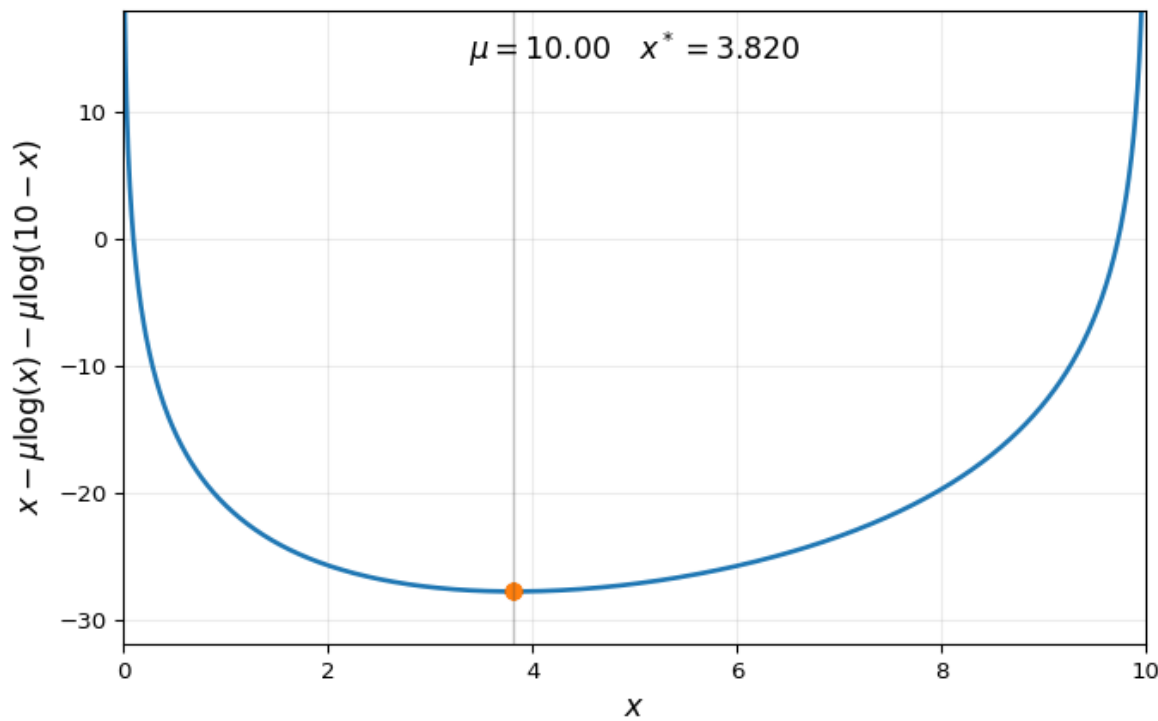
$$\begin{array}{ll}
 \min_{x \in \mathbb{R}^n} & f(x) \\
 \text{s.t.} & c(x) = 0 \\
 & x \geq 0
 \end{array}
 \quad \longrightarrow \quad
 \begin{array}{ll}
 \min_{x \in \mathbb{R}^n} & f(x) - \mu \log x \\
 \text{s.t.} & c(x) = 0
 \end{array}
 \quad (\text{BP}_\mu)$$

기본 알고리즘:

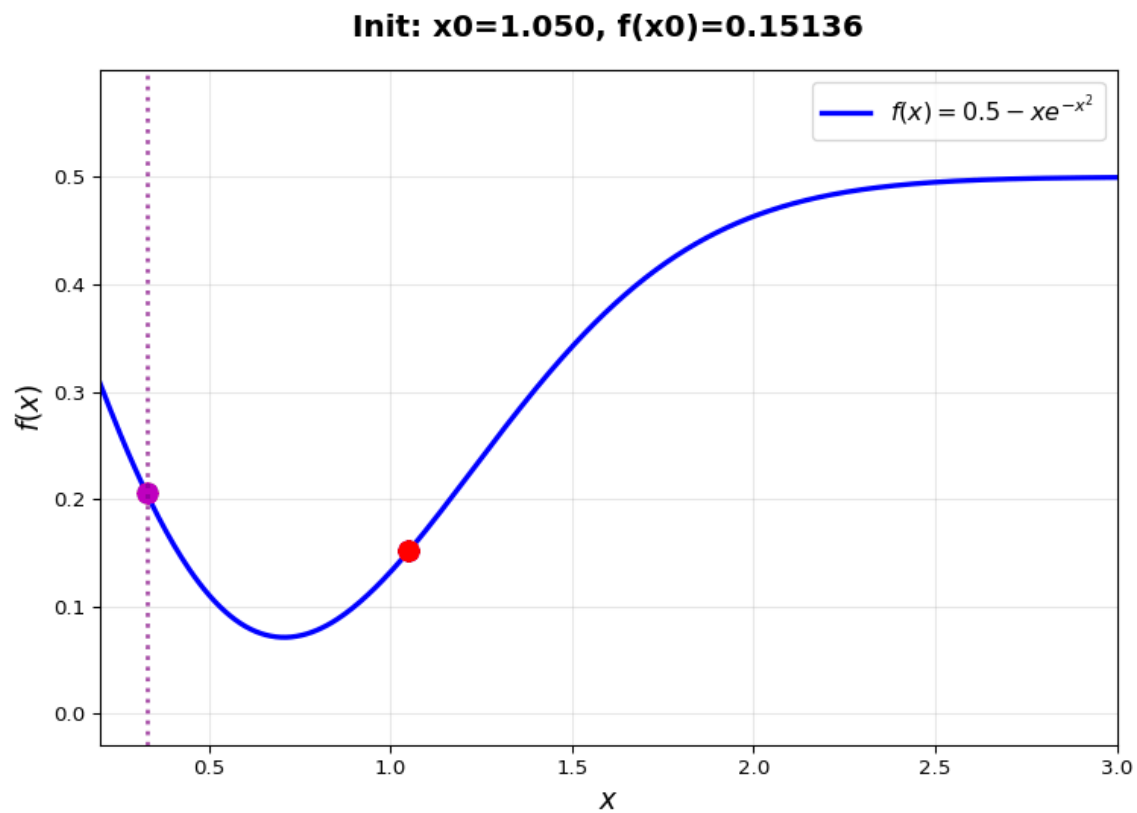
1. 초기값 $x_0 \in \mathbb{R}^n, \mu_0 > 0, k \leftarrow 0$ 으로 설정
2. (BP_μ) 에 대한 스텝(step) d_k 계산
3. 새로운 반복점 x_{k+1} 생성, $x_{k+1} = x_k + \alpha_k d_k$
4. (BP_μ) 가 일정 정확도로 해결된다면, μ 를 감소시킴
5. $k \leftarrow k + 1$ 하고; 단계 2 로 돌아감

배리어 함수(Barrier Function) 예시

$$\begin{array}{ll} \min & x \\ \text{s.t.} & 0 \leq x \leq 10 \end{array} \quad \longrightarrow \quad \min x - \mu \log x - \mu \log(10 - x)$$



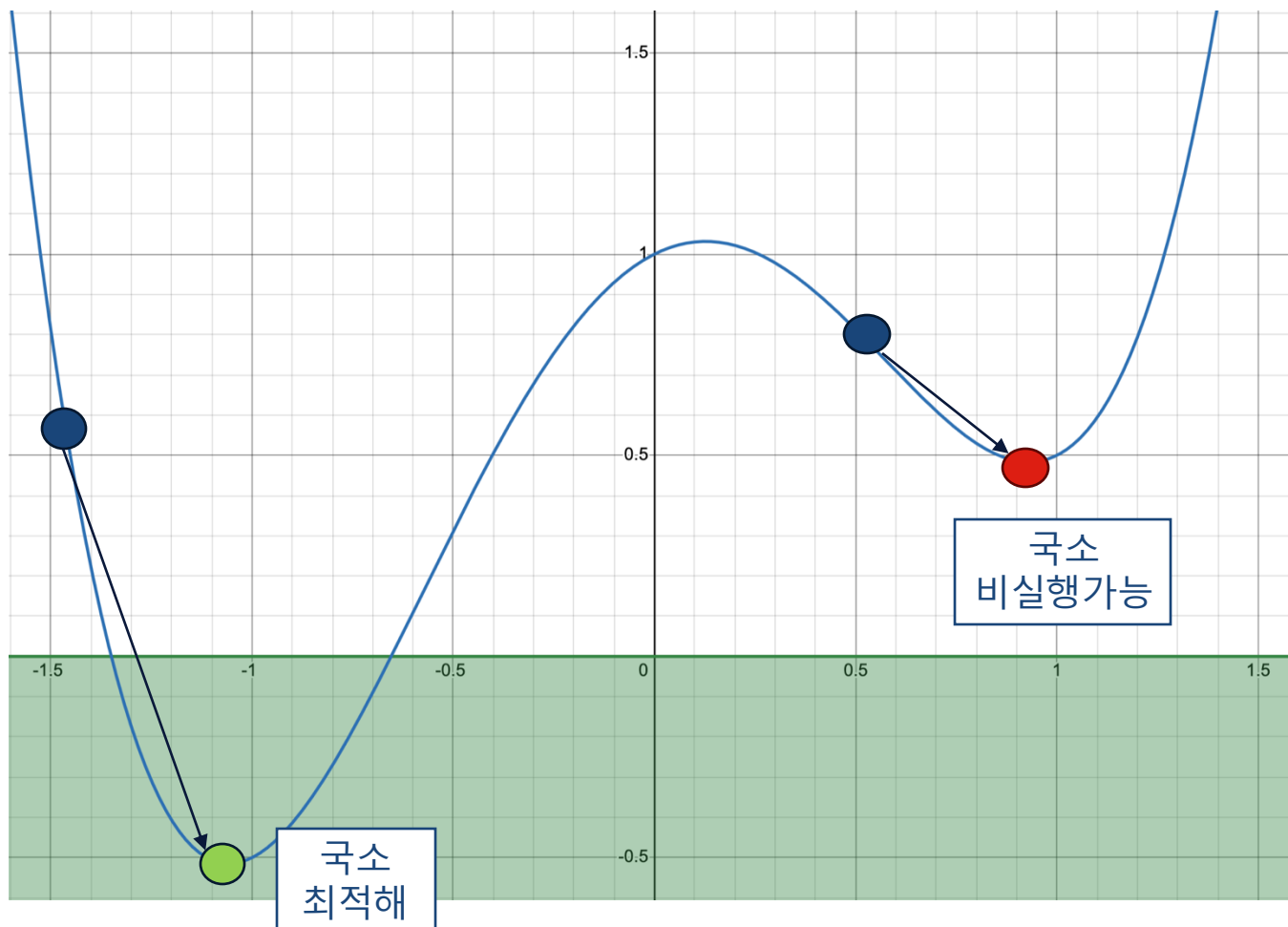
뉴턴 방법(Newton's Method) 예시



비선형 배리어(Nonlinear Barrier)

- 비선형 최적화(nonlinear programming)를 위한 내부점(interior-point) 알고리즘
- 1차 최적성 조건(1st order optimality conditions) 으로 수렴
 - 초선형(superlinear) 수렴률: *매우 빠른 국소(local) 수렴*
 - 높은 정확도의 해 제공
- 비볼록(non-convex) 문제에 대비한 정규화(regularization) 기법 포함
- KKT 시스템의 부정정부호(indefinite) 분해에 의존
- 라인 서치(line search), 실행가능성 복원(feasibility restoration) 등의 기법 사용
- 실무에서 매우 견고함이 입증된 접근 방식
 - IPOPT, Knitro 등 여러 NLP 솔버들이 유사한 방식 사용

국소 최적(Locally Optimal) / 국소 비실행가능(Locally Infeasible)



$$\begin{aligned} \min y \\ \text{s.t. } y &= x^4 - 2x^2 + 0.5x + 1 \\ y &\leq 0 \end{aligned}$$

- 시작점 $x = -1.5$
→ 국소 최적해
- 시작점 $x = 0.5$
→ 국소 비실행가능 발생
→ 알고리즘이 실행가능성 복원 (feasibility recovery) 절차로 전환되어 제약 위반을 최소화 시도
→ 결과적으로 국소 비실행가능 해 도출

Gurobi 13에서 비선형 배리어(NL Barrier) 사용하기



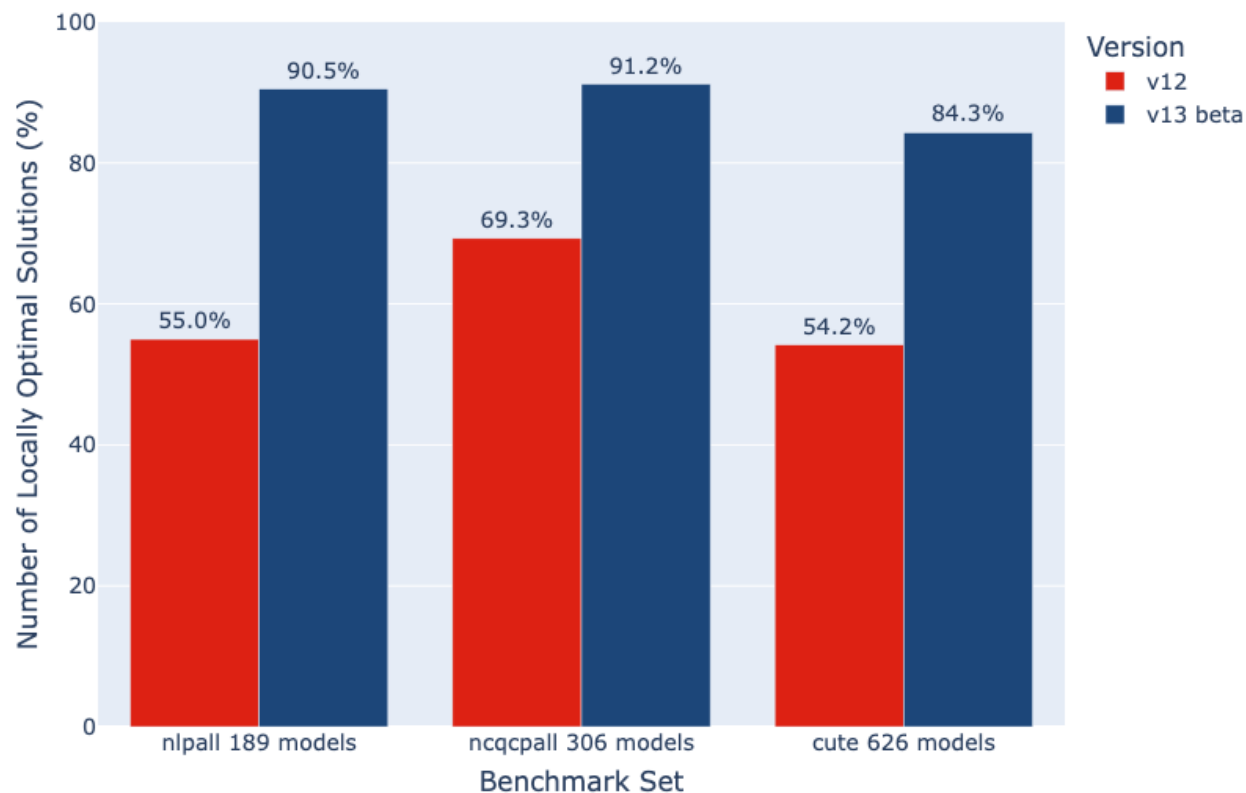
OptimalityTarget=1

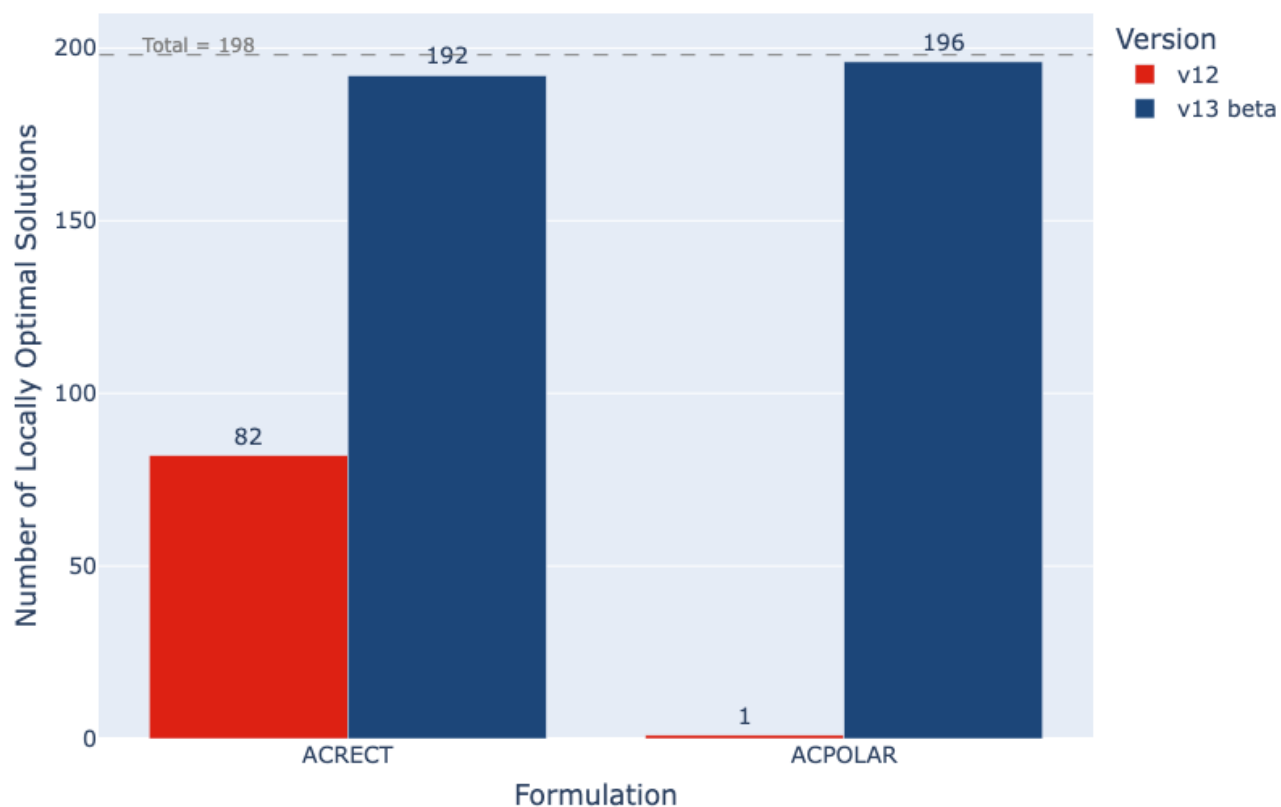
- 다른 값 옵션:
 - -1: 기본값(0으로 설정됨)
 - 0: 전역 MINLP 솔버 사용
 - 1: 비선형 배리어 사용(로컬 최적화)
- 비선형 연속(non-linear continuous) 문제에서만 사용 가능
 - 문제가 LP, convex QP, QCP인 경우 무시됨
 - 정수 변수(discrete variables) 또는 SOS 제약이 있으면 오류 발생
- Gurobi 13에서 "Preview" 기능으로 제공
 - 향후 버전에서 API 동작이 변경될 수 있음
 - 듀얼 정보(dual information)를 반환하지 않음
 - 품질 관련 속성들이 모두 지원되지 않음

NL Barrier - v13 개선 사항

- 실행가능성 복원
(feasibility restoration)
- 단순 라인서치
(simple line search)
- 반복적 정밀화
(iterative refinement)
- 알고리즘적 미분
(algorithmic differentiation)
- 보다 정확한 해를 얻기 위한
수치적 어려움 처리 개선

해 해결률(solve rate) 비교





Instances from <https://github.com/power-grid-lib/pglib-opf>
 generated using <https://github.com/Gurobi/gurobi-optimods>

ACOPF 성능

핵심 요소:

- 더 나은 시작점 처리
(Better handling of starting points)
- 실행가능성 복원
(Feasibility restoration)
- 알고리즘적 미분
(Algorithmic differentiation)

```
CPU model: AMD EPYC 7402P 24-Core Processor, instruction set [SSE2|AVX|AVX2]
Thread count: 12 physical cores, 12 logical processors, using up to 12
threads
```

```
Non-default parameters:
OptimalityTarget 1
```

```
Optimize a model with 4025 rows, 32604 columns and 5167 nonzeros (Min)
```

```
Model fingerprint: 0x1d675a52
```

```
Model has 2 linear objective coefficients
```

```
Model has 177 quadratic objective terms
```

```
Model has 37064 quadratic constraints
```

```
Coefficient statistics:
```

```
Matrix range [1e+00, 8e+03]
```

```
QMatrix range [4e-07, 1e+03]
```

```
QLMatrix range [1e+00, 1e+00]
```

```
Objective range [1e+00, 2e+03]
```

```
QObjective range [2e+01, 2e+03]
```

```
Bounds range [5e-04, 1e+03]
```

```
RHS range [5e-04, 1e+00]
```

```
QRHS range [5e-04, 1e+06]
```

```
Presolve removed 4007 rows and 14193 columns
```

```
Continuous model is non-convex -- solving with NL barrier
```

ACOPF 사례, 6468_rte

- 이 모델은 2013년 프랑스의 초고압(VHV) 및 고압(HV) 송전망을 정확하게 나타냅니다:
 - 399개 발전기(generators)
 - 18,000개 송전선(branches)
 - 6,468개 노드(nodes)

```
Presolve removed 5857 rows and 45606 columns
Presolve time: 0.59s
Presolved: 7080 rows, 43326 columns, 28005 nonzeros
Presolved model has 77652 quadratic constraint(s)
Warning, incomplete warm-start solution
Ordering time: 0.09s
```

NL barrier statistics:

```
Hessian NZ : 7.194e+04
Jacobian NZ : 3.180e+05
Factor NZ : 1.679e+06
Factor Ops : 2.426e+07
Threads : 1
```

Iter	Objective	Residuals			Step	Time
		Primal	Dual	Compl		
0	4.25101256e+05	3.11e+01	1.73e+02	1.84e+04	0.00e+00	1s
1	4.25101259e+05	3.11e+01	1.73e+02	1.84e+04	4.75e-10	1s
2	4.49822250e+05	3.09e+01	1.72e+02	1.83e+04	6.04e-03	1s
3	4.85968510e+05	3.06e+01	1.72e+02	1.83e+04	9.64e-03	1s
4	5.49358735e+05	3.00e+01	1.70e+02	1.82e+04	1.83e-02	2s
5	6.21857031e+05	2.93e+01	1.68e+02	1.79e+04	2.30e-02	2s

...

ACOPF 사례, 6468_rte

- 이 모델은 2013년 프랑스의 초고압(VHV) 및 고압(HV) 송전망을 정확하게 나타냅니다:
 - 399개 발전기(generators)
 - 18,000개 송전선(branches)
 - 6,468개 노드(nodes)

ACOPF 사례, 6468_rte

- 이 모델은 2013년 프랑스의 초고압(VHV) 및 고압(HV) 송전망을 정확하게 나타냅니다:
 - 399개 발전기(generators)
 - 18,000개 송전선(branches)
 - 6,468개 노드(nodes)

...							
41	2.83887656e+06	9.67e+00	1.89e+05	7.51e+02	1.52e-03	6s	
42*	2.83887656e+06	9.67e+00	3.10e+04	1.42e+04	0.00e+00	6s	
43*	2.83302648e+06	7.78e+00	2.10e+04	1.02e+04	1.95e-01	7s	
44*	2.83009023e+06	6.82e+00	1.14e+04	6.90e+03	1.23e-01	7s	
45*	2.82881891e+06	6.32e+00	1.01e+04	6.24e+03	7.45e-02	7s	
46*	2.82844843e+06	5.14e+00	8.90e+03	5.16e+03	2.23e-01	7s	
47*	2.82843805e+06	5.12e+00	8.60e+03	5.07e+03	2.85e-03	8s	
48*	2.82736002e+06	4.54e+00	6.83e+03	4.26e+03	1.13e-01	8s	
...							
63*	2.97552582e+06	1.80e+00	3.41e+02	1.06e+02	4.59e-01	11s	
64*	3.06728379e+06	3.52e+00	1.48e+02	4.62e+01	6.76e-01	11s	
65*	3.17581783e+06	7.05e+00	6.50e+00	1.48e+00	9.66e-01	11s	
66*	3.20837525e+06	3.52e+00	1.32e+00	1.00e+00	1.00e+00	11s	
67*	3.21003568e+06	3.11e-02	1.77e-02	1.00e+00	1.00e+00	11s	
68	3.21721689e+06	2.79e-01	6.98e+02	1.84e+04	6.00e-02	11s	
69	3.21721689e+06	2.79e-01	5.12e+02	1.35e+04	4.75e-10	11s	
...							

120	2.45283950e+06	3.79e-01	2.34e+00	3.51e-04	3.52e-01	17s
121	2.45280236e+06	2.19e-01	1.12e+00	3.48e-04	5.59e-01	17s
122	2.45278005e+06	2.44e-02	1.01e-03	3.46e-04	1.00e+00	17s
123	2.45278173e+06	3.40e-04	1.38e-05	3.46e-04	1.00e+00	17s
124	2.45278169e+06	1.50e-05	9.54e-07	3.46e-04	1.00e+00	17s
125	2.45276480e+06	1.51e-02	4.93e-01	1.32e-04	4.00e-01	17s
126	2.45275827e+06	1.40e-02	1.18e+00	2.09e-05	3.05e-01	17s
127	2.45275017e+06	1.28e-02	6.30e-01	1.51e-05	6.14e-01	17s
128	2.45274731e+06	5.30e-03	2.71e-01	1.44e-05	6.96e-01	18s
129	2.45274624e+06	4.37e-04	3.87e-05	1.43e-05	1.00e+00	18s
130	2.45274627e+06	1.65e-05	8.74e-06	1.43e-05	1.00e+00	18s
131	2.45274627e+06	3.83e-07	5.96e-08	1.43e-05	1.00e+00	18s
132	2.45274549e+06	1.71e-04	7.43e-02	1.52e-06	6.72e-01	18s
133	2.45274523e+06	9.70e-05	3.80e-02	1.70e-07	7.43e-01	18s
134	2.45274515e+06	3.81e-05	6.63e-03	1.66e-07	8.78e-01	18s
135	2.45274514e+06	1.50e-06	3.02e-06	1.65e-07	1.00e+00	18s
136	2.45274514e+06	1.86e-09	1.94e-09	1.65e-07	1.00e+00	18s
137	2.45274513e+06	2.88e-07	8.60e-04	1.11e-09	8.50e-01	19s
138	2.45274513e+06	1.26e-08	5.51e-09	1.00e-09	1.00e+00	19s
139	2.45274513e+06	1.86e-09	2.09e-10	1.00e-09	1.00e+00	19s

NL barrier solved model in 139 iterations and 18.85 seconds (13.08 work units)

First-order optimal solution

Solution objective 2.452745130706e+06

ACOPF 사례, 6468_rte

- 이 모델은 2013년 프랑스의 초고압(VHV) 및 고압(HV) 송전망을 정확하게 나타냅니다:
 - 399개 발전기(generators)
 - 18,000개 송전선(branches)
 - 6,468개 노드(nodes)
- 전역 최적해까지 풀려면 시간이 거의 무한히 걸림

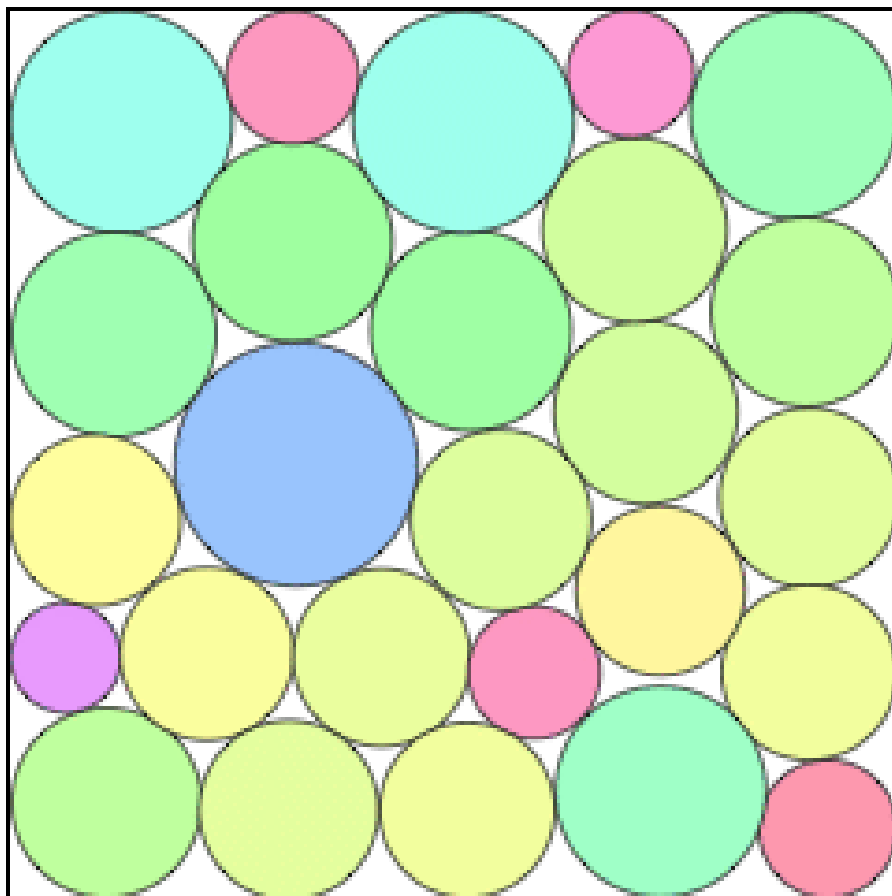
글로벌 솔버(Global Solver)에서의 배리어 통합

- NL 배리어(NL Barrier)는 글로벌 솔버 내에서 "휴리스틱(heuristic)"으로 사용됨
- 목표: 좋은 국소 최적해(locally optimal solution) 를 찾는 것
- NLPheur 파라미터로 제어되며, 새로운 값들은 다음과 같음:
 - -1: 기본값
 - 0: 비활성화
 - 1: 약하게 적용 (루트 노드에서 1~2회 정도 실행)
 - 2: 중간 강도 (정수 변수가 있는 모델에서, 모든 assignment 지점마다 실행)
 - 3: 강하게 적용 (각 노드에서 node local bounds 기반으로 실행)

Circle Packing Problem

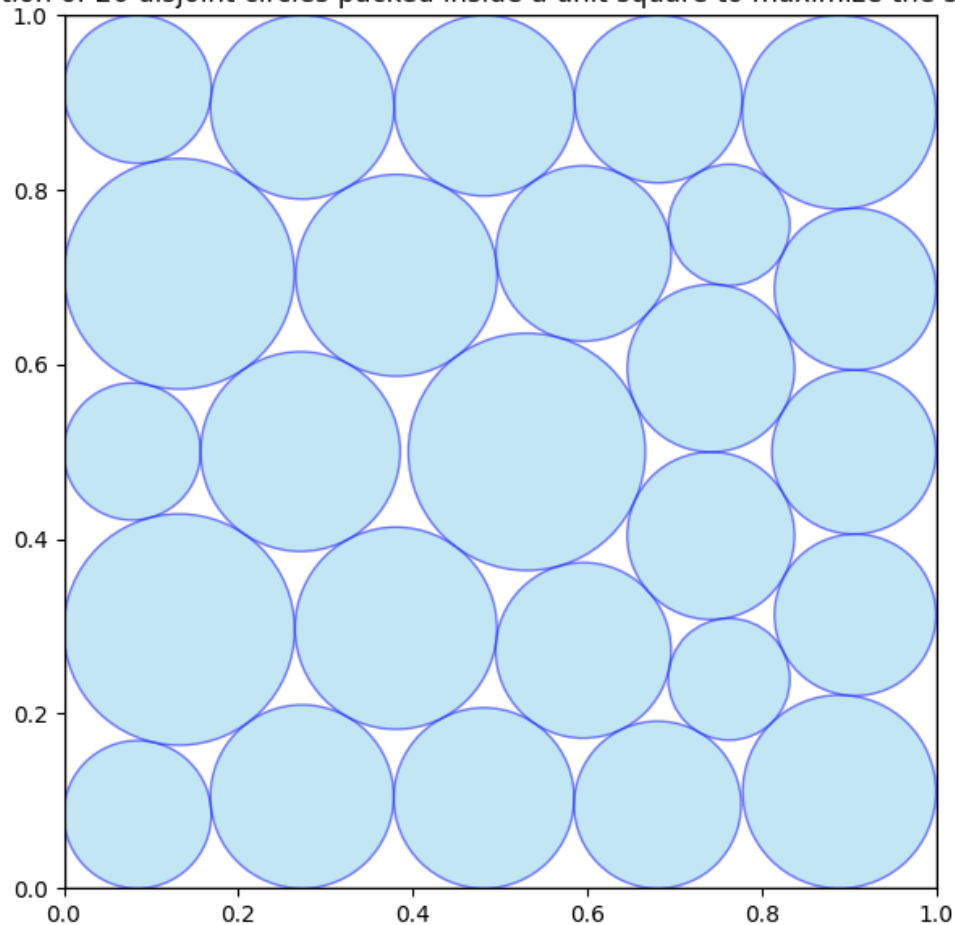
Previous Best

2.634+



<https://erich-friedman.github.io/packing/cirRsqu/>

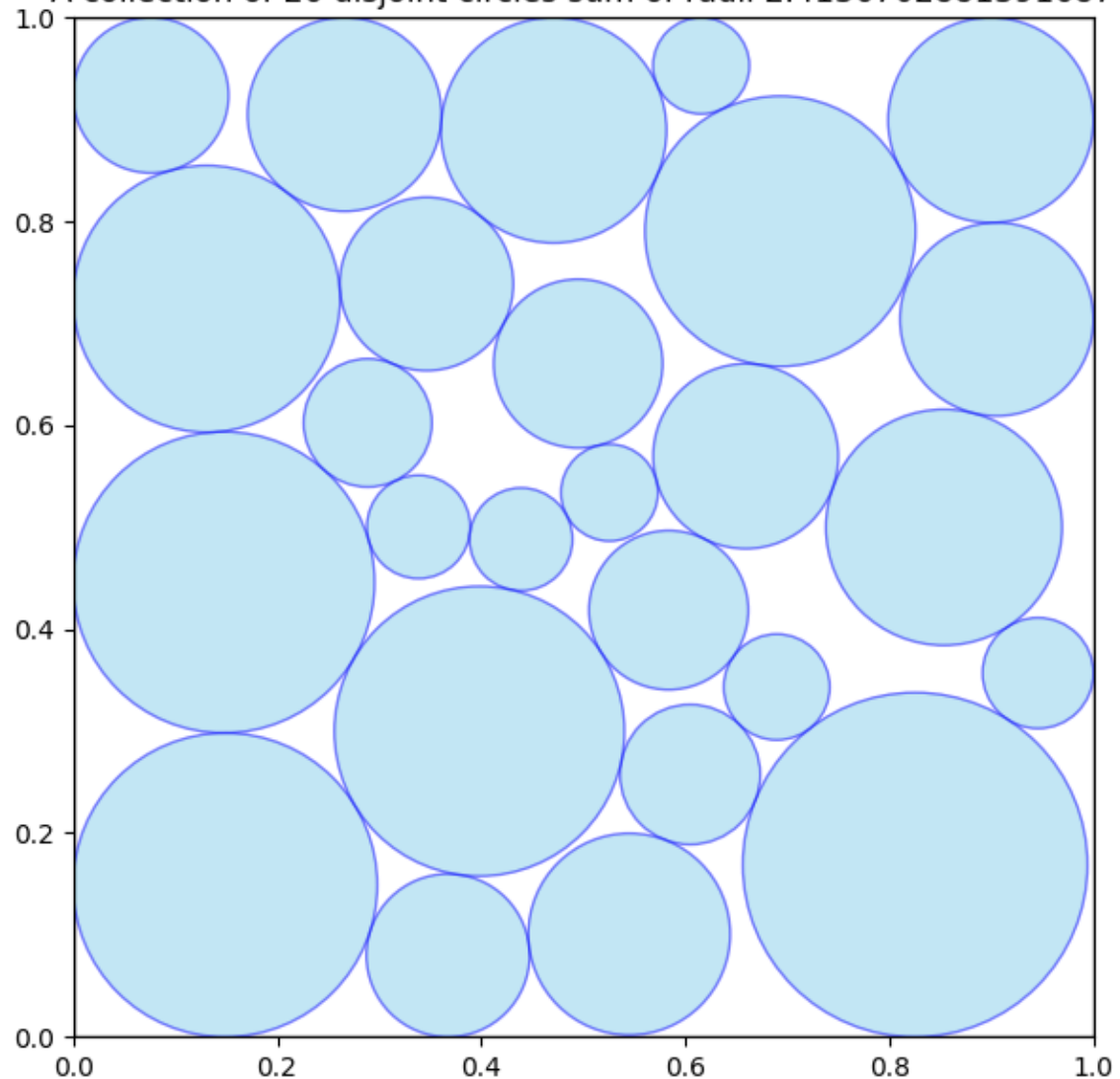
A collection of 26 disjoint circles packed inside a unit square to maximize the sum of radii



Circle Packing Problem

Previous Best	2.634+
AlphaEvolve	2.635 86275

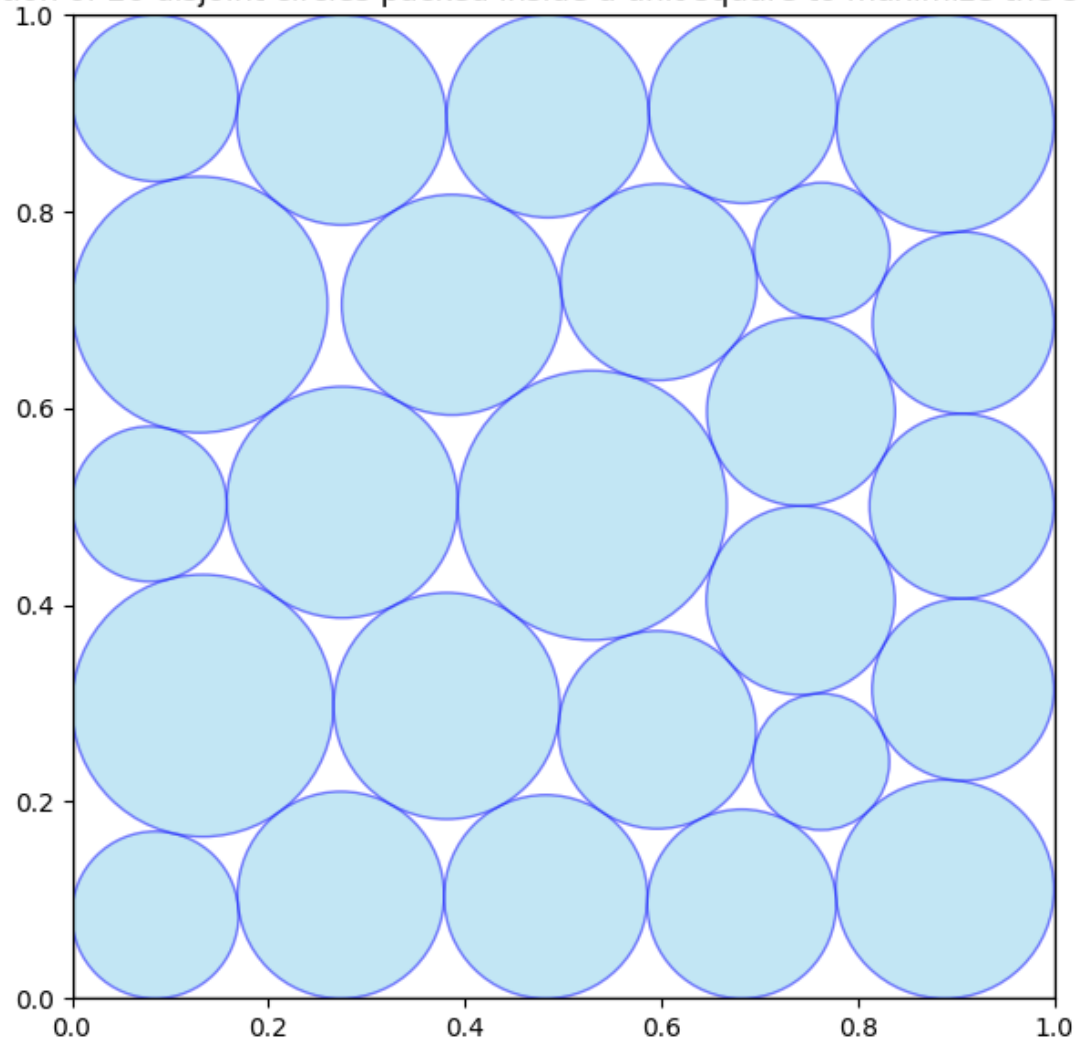
A collection of 26 disjoint circles sum of radii 2.4136702881391687



Circle Packing Problem

Previous Best	2.634+
AlphaEvolve	2.635 86275
Gurobi 12	2.41367028

A collection of 26 disjoint circles packed inside a unit square to maximize the sum of radii

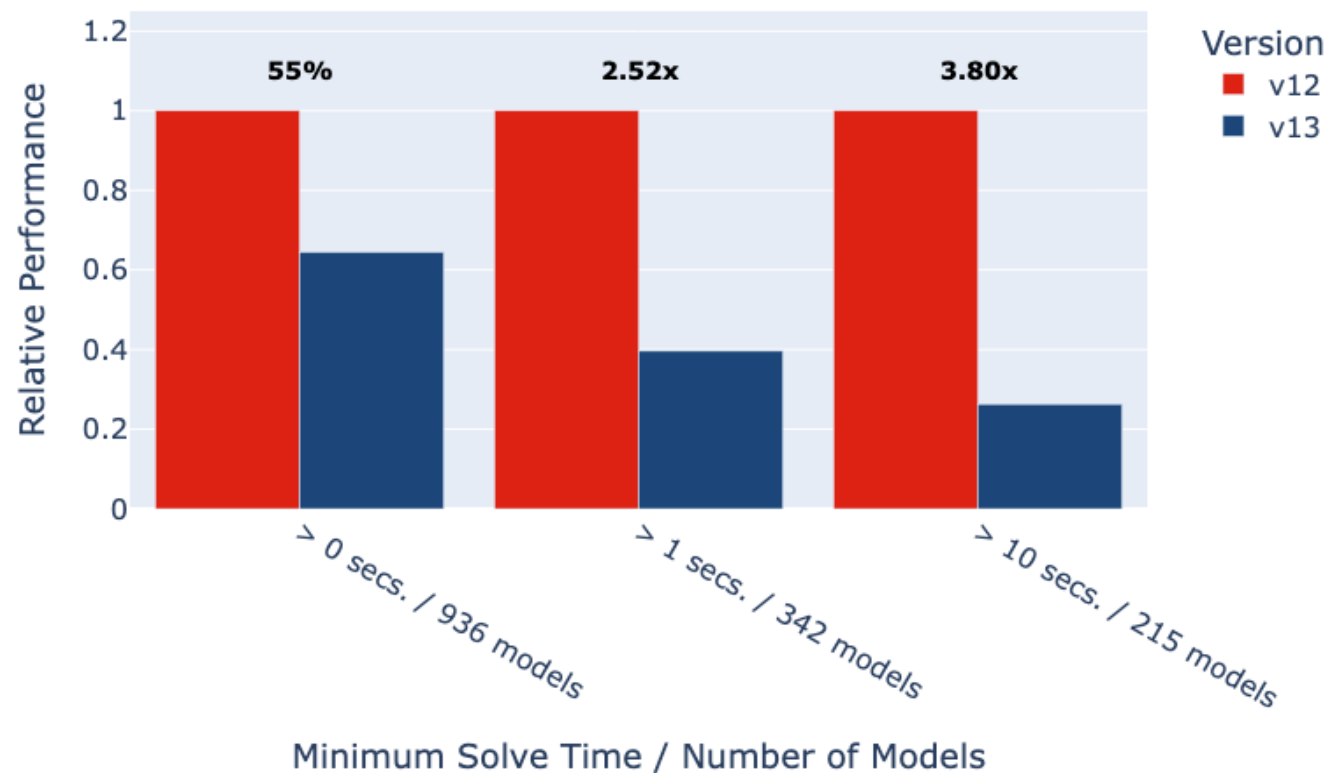


Circle Packing Problem

Previous Best	2.634+
AlphaEvolve	2.635 86275
Gurobi 12	2.41367028
Gurobi 13	2.635 98301

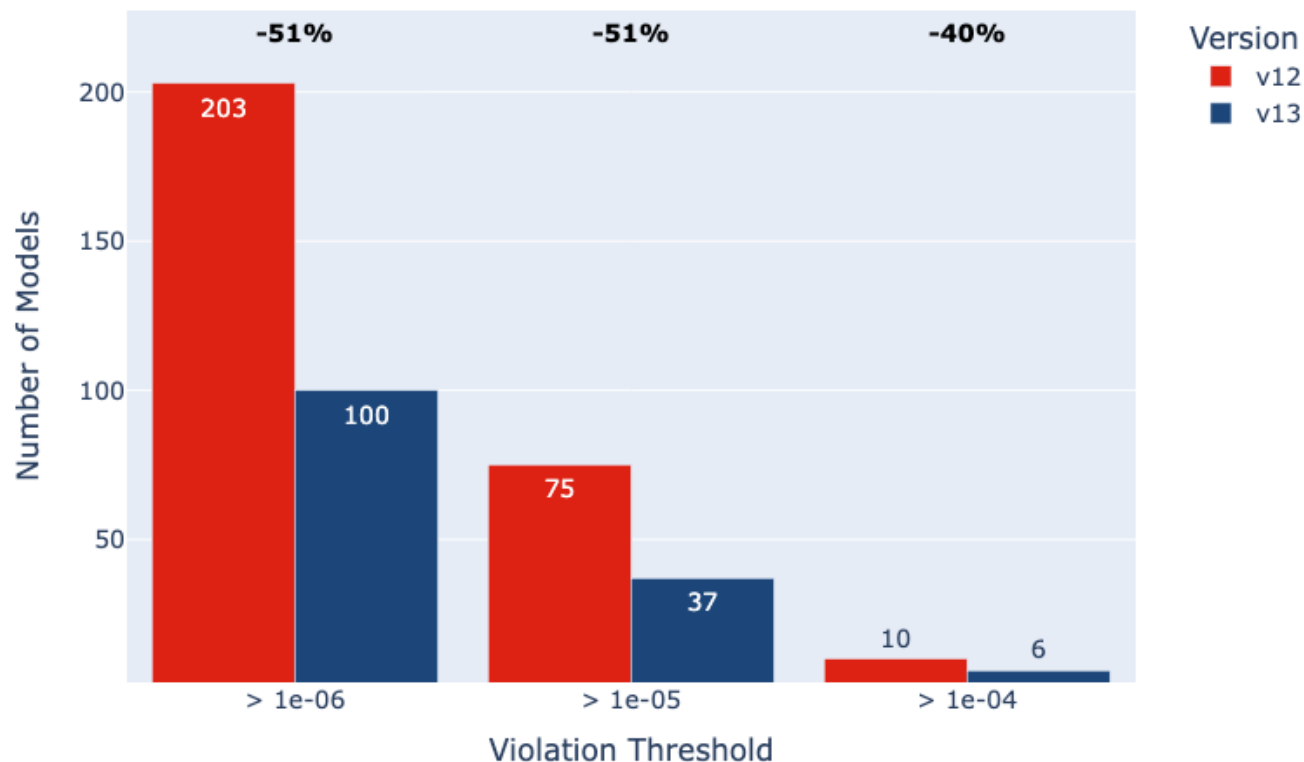
MINLP 성능 향상

- 해결 시간 기준(solve time threshold)에 따른 성능 비교
- 내부 테스트 세트 1394개 모델 기반 통계
- 두 버전(v12, v13) 모두 풀지 못한 458개 모델은 제외



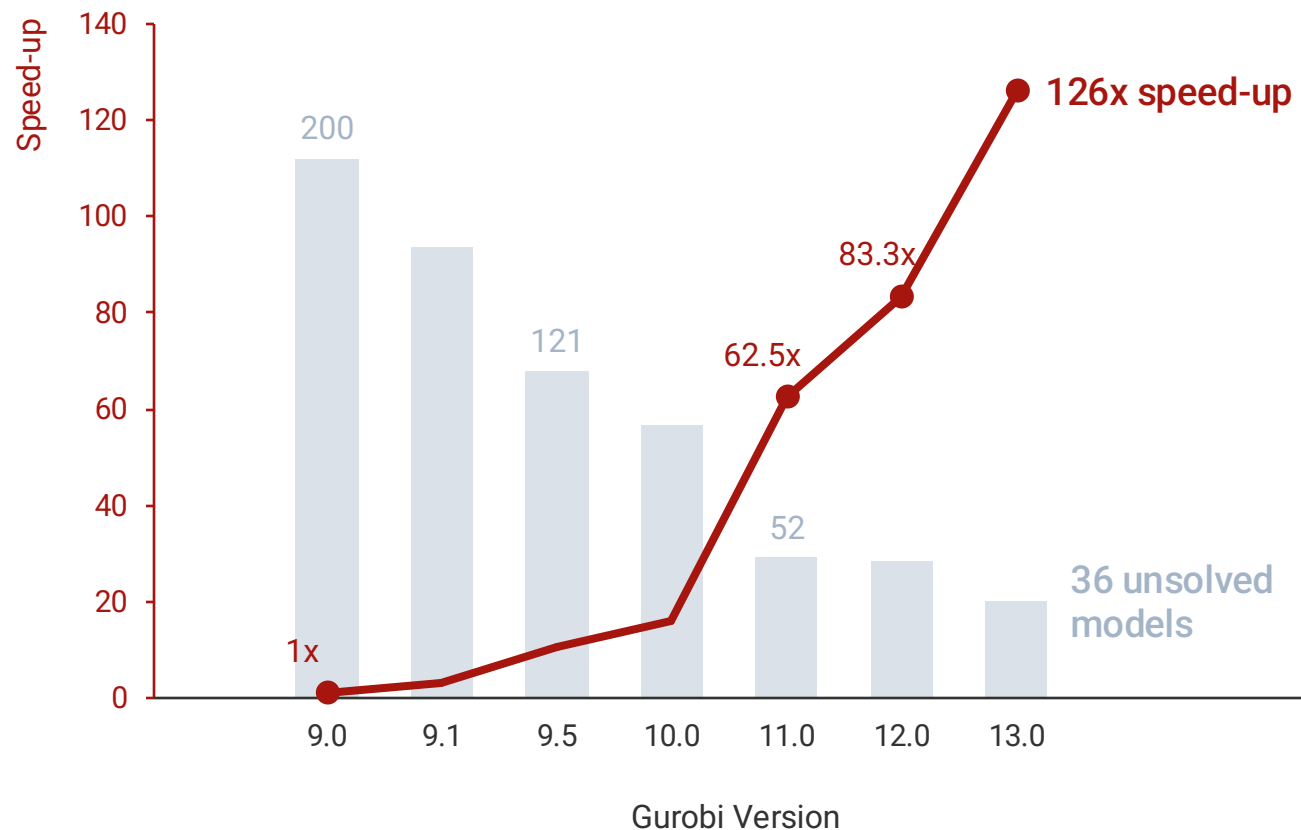
MINLP 해 품질 향상

- 제약 위반(cumulative constraint violation)을 기준으로 품질 비교
 - 제약을 많이 위반할수록 품질이 낮음
 - 제약 위반이 작은 해를 더 많이 만든다면 품질 개선이라는 뜻
- 내부 검증 세트 1394개 모델 기반 통계
- Gurobi v12는 1219개 모델에서 해를 생성
- Gurobi v13은 1309개 모델에서 해를 생성



Gurobi Version Comparison: Speed and Solvability (PAR-10)

Gurobi Nonconvex MIQCP Benchmark Suite



Nonconvex MIQCP

성능 향상

Gurobi의 Nonconvex MIQCP 벤치마크 세트에서 최신 버전은 다음과 같은 성능을 제공합니다:

- 버전 9.0 대비 126배 속도 향상 (실행시간 기하평균 기준, PAR-10)
- 최신 버전에서도 10,000초 내에 풀리지 않은 미해결 모델은 단 36개뿐, (이 테스트 세트는 최소 한 버전에서라도 풀 수 있는 모든 모델들로 구성되어 있음)

Time limit: 10000 sec.
Intel Xeon CPU E3-1240 v5 @ 3.50GHz
4 cores, 8 hyper-threads
32 GB RAM

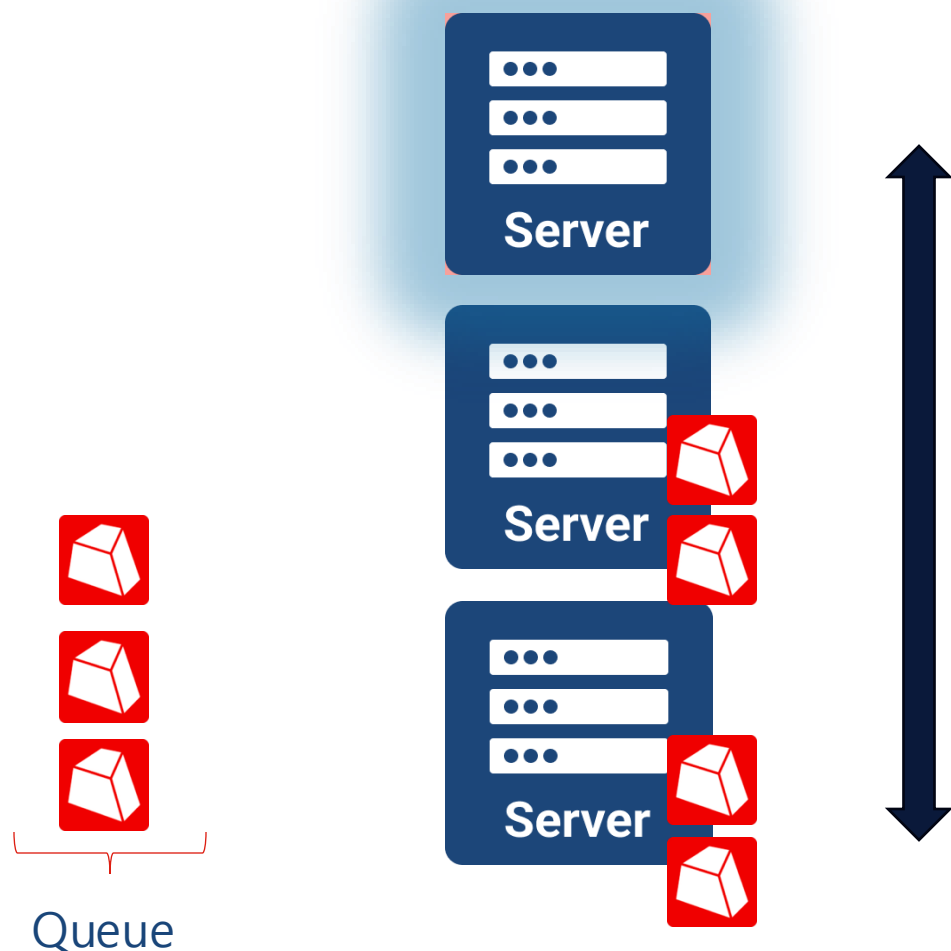
Test set has 1226 models:
- 89 discarded due to inconsistent answers
- 375 discarded that none of the versions can solve
- speed-up measured on >100s bracket: 337 models

플랫폼 기능 개선 & 결론

© 2025 Gurobi Optimization, LLC. Confidential, All Rights Reserved



GUROBI
OPTIMIZATION



Kubernetes Autoscaler Operator

- Kubernetes를 통해 Gurobi Compute Server를 자동으로 스케일링할 수 있음
- 지능형 스케일링 동작은 다음 파라미터로 구성 가능:
 - 작업 대기열(job queue)
 - 대기 시간(waiting time)
 - 유휴 시간(idle time)
 - 대기 중인 작업 수(number of queued jobs)
- 안전한 스케일 다운 로직을 적용하여 실행 중인 작업이 중단되지 않도록 보장함
- DockerHub에서 이용 가능

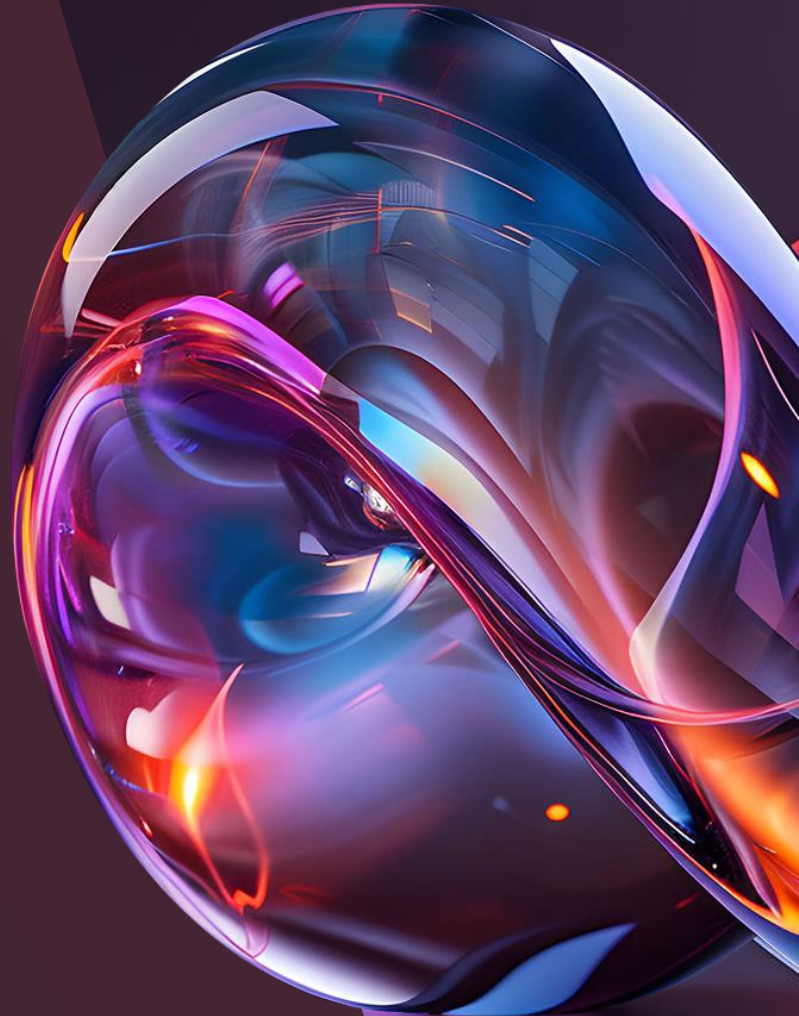
접근성(Accessibility)

버전 13의 Cluster Manager 접근성 개선 사항

- 전체 키보드 내비게이션 지원
- 모든 기능이 화면 읽기(screen-reader) 지원 가능
- 더 명확한 구조와 향상된 텍스트 대비(text contrast)

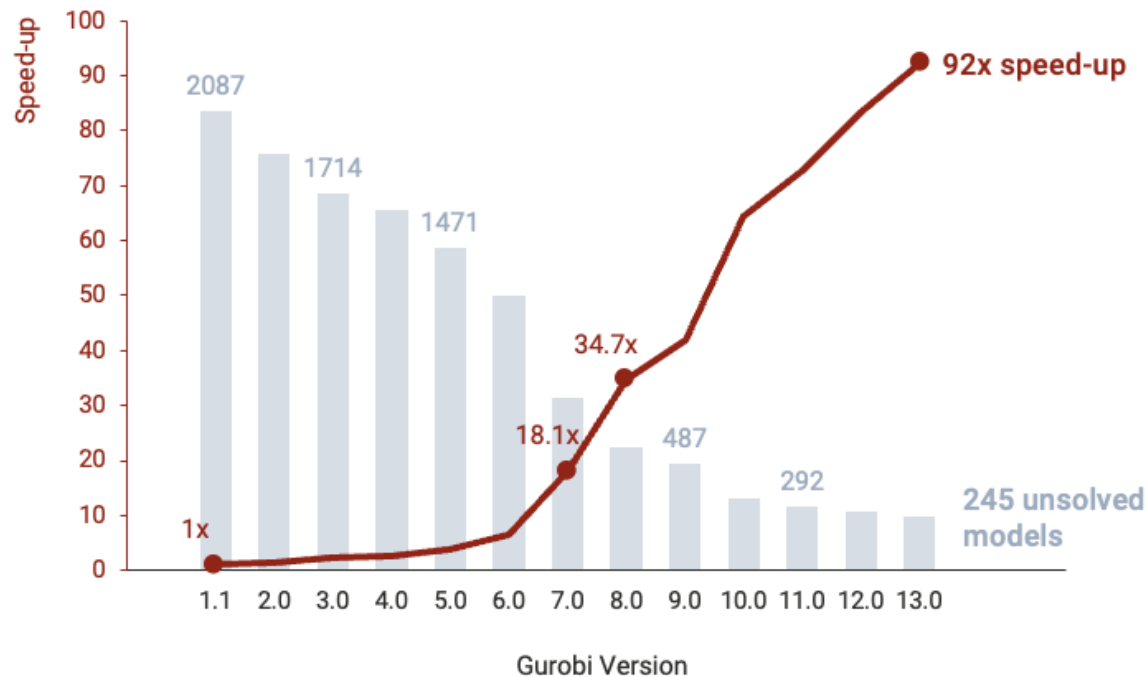
플랫폼 전반의 일관된 접근성 제공

- 웹사이트와 사용자 포털 개선에 기반하여 확장됨
- Gurobi 플랫폼 전반에서 일관되고 포용적인 사용자 경험 제공
- WCAG 2.1 AA 및 유럽 접근성 지침(European Accessibility Act, EN 301 549) 기준을 완전히 준수

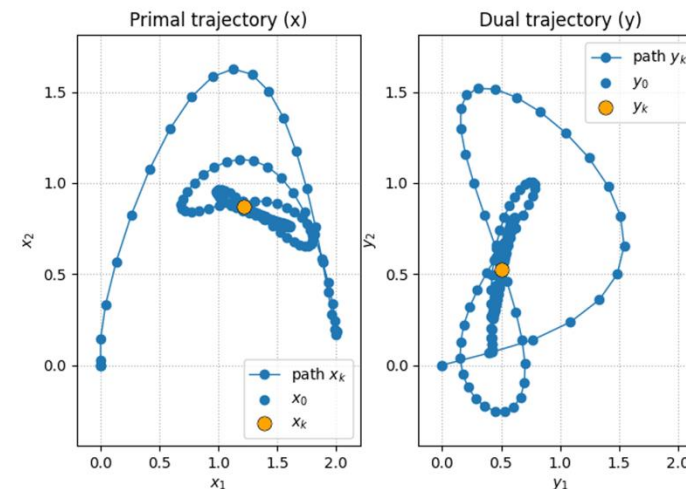
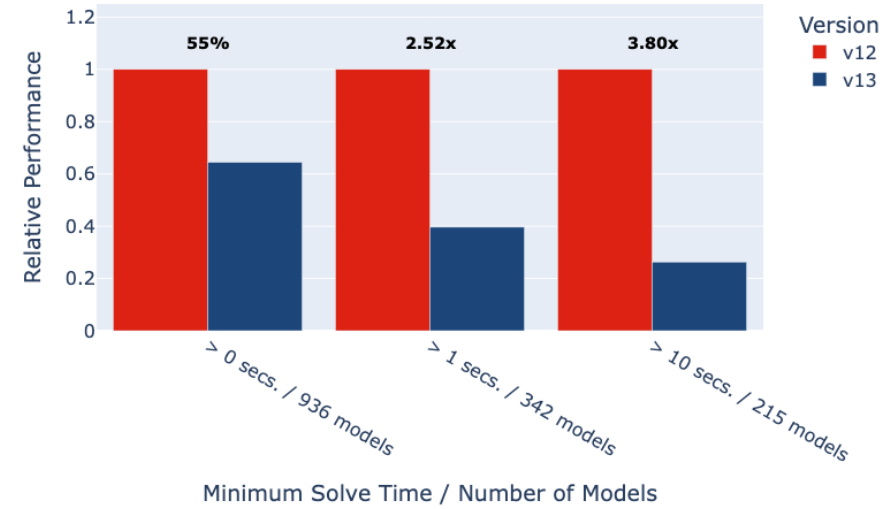


결론

MILP Performance



MINLP (+ NL Barrier)



**PDHG
+ GPU**

웨비나 시리즈

2026 – 13.0 웨비나 시리즈 계속됩니다:

- 로컬 NL 배리어(Local NL Barrier)
- 글로벌 MINLP(Global MINLP)
- GPU 기반 PDHG(PDHG on GPU)
- Gurobi Intelligence – GenAI 업데이트

[Gurobi.com/events](https://gurobi.com/events)에서 계속 소식을 확인하세요!



감사합니다

