

# hexaly

î ð 14 Ë J g H i Δ ī L E i Δ ī Š G J Ë H G d J g 12 H G g d > 14 J i Ë H  
% ð g % 6 S S Δ S f i E i E S i  
J S E Ë C E Ë R S M i E Δ J % E 14 % E 14 A c i o @ ""

ç L s G G E i Δ d g E  
G A d g E d z L E R A 14 T d i

o o o ' L E R A 14 T d i

# LocalSolver

— is now —

# hexaly

We're excited to share that we are moving forward. We're leaving behind the LocalSolver brand and transitioning to our new identity: Hexaly. This represents a leap forward in our mission to enable every organization to make better decisions faster when faced with operational and strategic challenges.

# Mathematical solvers for scheduling problems

٪ؑا“مؑا“لؑا“ »ؑا“سؑا“ تؑا“مؑا“سؑا“ ءؑا“هؑا“شؑا“ مؑا“زؑا“خؑا“ مؑا“عؑا“قؑا“زؑا“عؑا“فؑا“ ءؑا“زؑا“

5 جـ هـشـهـهـهـهـ



٪ؑا“دؑا“شـلـ

5 جـ هـشـهـهـهـهـ



n ئـلـئـلـ ئـشـلـ ئـلـئـلـ ئـلـ

f(x)

٪ؑا“دؑا“شـلـ



# Mathematical solvers for scheduling problems

” $\text{M}\text{u}\text{l}$  »  $\text{s}^\circ$  ” $\text{T}\text{M}\text{u}\text{S}\text{E}$ »  $\pm\text{H}\text{u}\text{m}\text{u}\text{X}$   $\mu\text{u}\text{q}\text{E}\text{u}\text{f}$  »

- $\text{Z}\text{u}\text{u}\text{H}\text{u}\text{g}\text{u}\text{A}\text{u}\text{H}\text{u}\text{L}\text{u}\text{E}\text{u}\text{u}\text{S}$   $\text{G}\text{g}\text{d}$   $\text{u}\text{u}\text{u}\text{u}\text{u}$   $\text{H}$
- $\text{i}$   $\text{d}\text{i}$   $\text{E}\text{i}$   $\text{A}\text{i}\text{L}\text{i}$   $\text{A}\text{i}\text{S}\text{i}$   $\text{A}\text{i}\text{d}\text{S}\text{i}$   $\text{E}\text{i}\text{G}\text{i}\text{H}\text{i}\text{g}\text{i}$   $\text{H}\text{i}\text{L}\text{i}\text{d}\text{d}\text{i}$
- $\text{Z}\text{u}\text{u}\text{H}\text{u}\text{g}\text{u}\text{A}\text{u}\text{S}\text{o}$   $\text{S}\text{u}\text{L}\text{u}\text{W}\text{u}\text{E}\text{u}\text{R}\text{u}\text{A}$   $\text{G}\text{u}\text{S}\text{u}\text{E}\text{g}$ 
  - $n$   $\text{d}\text{E}\text{u}\text{S}$   $\text{A}\text{S}$   $\text{J}\text{A}$   $\text{E}$
  - $\text{c}\text{E}\text{g}$   $\text{d}\text{g}$   $\text{A}\text{S}\text{f}\text{E}$



# Mathematical solvers for scheduling problems

^ ūš≈μ‡ ūš» ū/’ Ņ· ‘ μÙš°

Qđđi „ E½i ŃŠTΔgř HfiŁđđ½

- oJi „ Egi
  - 6đi GΔgřHđS „ Eto ŃEŞSđj „ Egi
  - %%ššS TΔ 1½H
  - n J 1½Gđi TΔšS TΔ 1½H
- J „ X „ 3 „ -  
ε IXL  
ε K3 IVJ  
ε + 3 IVt
- J š  
J šIXēW  
S J š  
S ΔšJ š

# Mathematical solvers for scheduling problems

^ ūš≈μ‡ ūš» ū/’ Ŕ· ‘ μÙš°

Qđđi „ E½i ŒŠTΔgř HfiŁđđ½

- oJi „ Egi
  - 6đi GΔgř Hš, Eto ŒEşŞJi „ Egi
  - %œšŞTΔ ¼H
  - n JœGši ΔœŞTΔ ¼H
- J „ X „ 3 „ -  
ε I X i  
ε K 3 I V J  
ε + 3 I V t
- J š  
J š I X e W  
S J š  
S A š J š

■ n Z

# Mathematical solvers for scheduling problems

^ ūš≈μ‡ ūš» ū/’ Ŕ· ‘ μÙš°

Qđđi + HŒfidS%Agj H̄iLđđ½

- i ŒT̄H

”J ∈ ”a a” + ⊆ 5”GAḡššS + 5š + ”-

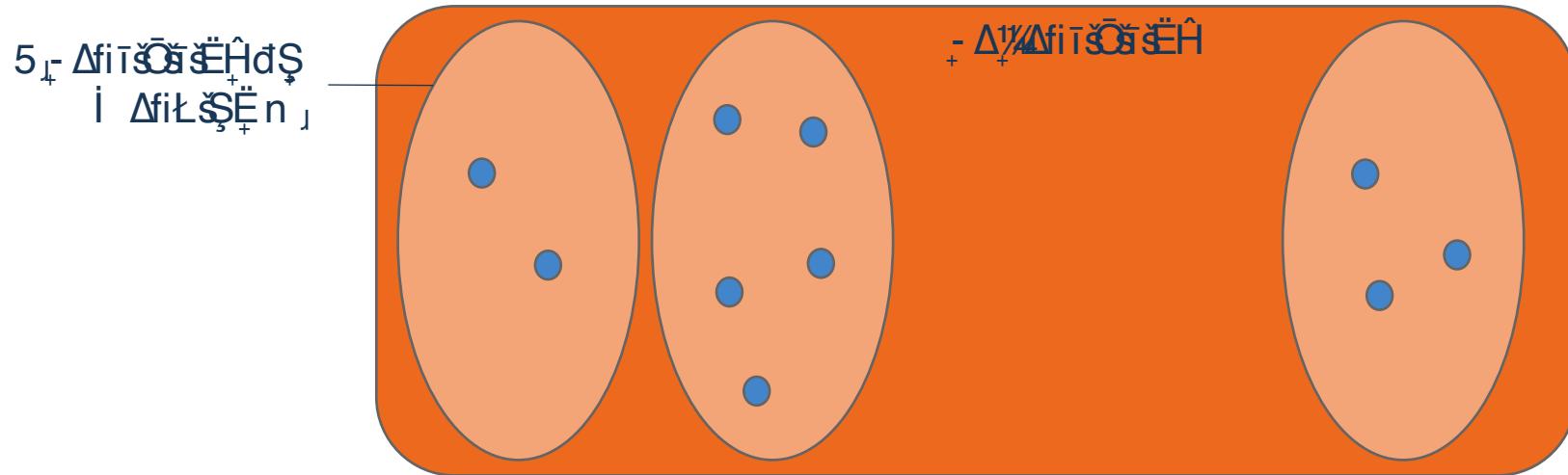
# Mathematical solvers for scheduling problems

^ ūš≈μ‡ ūš» ū/’ Ņ· ‘ μÙš°

Qđi + HËfidS%Agj HiLđđ½

- i EiH

"J ∈ "ă ă" + ⊆ 5" GAgjššS + 5š + "



# Mathematical solvers for scheduling problems

^ ūš≈μ‡ ūš» ū/’ Ņ· ‘ μÙš°

Qđđi + HËfiđS%Agř HíLđđ½

- i ËiH
- çËgj JíΔíššH

"J ∈ "ă ä" + ⊆ 5"ĞAgřššS " 5š " -  
i " " ∈ i

# Mathematical solvers for scheduling problems

^ ūš≈μ‡ ūš» ū/’ Ņ· ‘ μÙš°

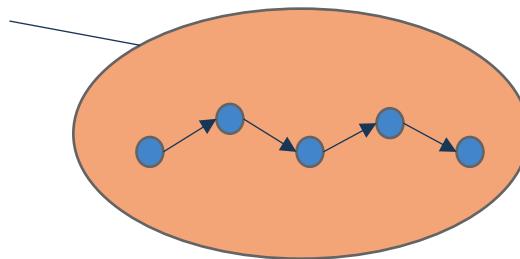
Qđđi + HËfiđS%Agđ HíLđđ½

- i EiH
- çEgj JíAíššH

"J ∈ "ă a" + ⊆ 5" GAgíššS + 5s "

i " + " ∈ i

i - HËGJ EŠfiEđ  
Δfi IššS ŠHđS  
Δi ΔfiLššEñ



# Mathematical solvers for scheduling problems

^ ūš≈μ‡ ūš» ū/’ Ŕ· ‘ μÙš°

Qđđi + HŒfiđS%Agř HřiLđđ¼

- i ŒTř
- çŒgř JřΔřšřSř
- ZřTŒgřOđ¼

”Jř ∈ ”a š” + ⊆ 5”GAgřšřSř + 5š ”-

i ” ” ∈ i

RřIV JřO ”ř ∈ R

# Mathematical solvers for scheduling problems

^ ūš≈μ‡ ūš» ū/’ Ŕ· ‘ μÙš

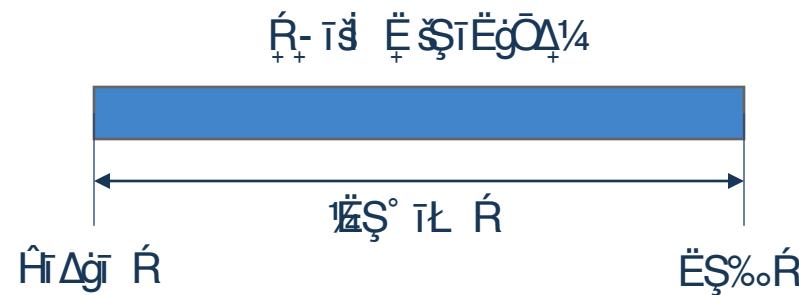
Qđđi + HŒfiđS%Agř HřiLđđ½

- i ŒiH
- çŒgj J̄ΔiňS̄H
- Z̄iŒgOđ½

”J̄ ∈ ”a a” + ⊆ 5”GAgřššS + 5š ”-

i ” + ” ∈ i

RIV J̄ O”i ∈ R



# Mathematical solvers for scheduling problems

^ ūš≈μ‡ ūš» ū/’ Ŕ· ‘ μÙš°

Qđđi + HŒfiđS%Agj H̄iLđđ¼

- i ŒT̄H
- çŒgj J̄iΔiňšŞH
- Z̄iŒgOΔ¼
- Q̄ŞfiňšŞH

”J ∈ ”ă ă” + ⊆ 5”GΔḡiňšŞS ” 5š ”-

i ” ” ∈ i

RIV J̄O ”i ∈ R

· T

# Mathematical solvers for scheduling problems

^ ūš≈μ‡ ūš» ū/’ ſ· ‘ μÙš°

Qđđi + HëfiđS%Agj HíLđđ¼

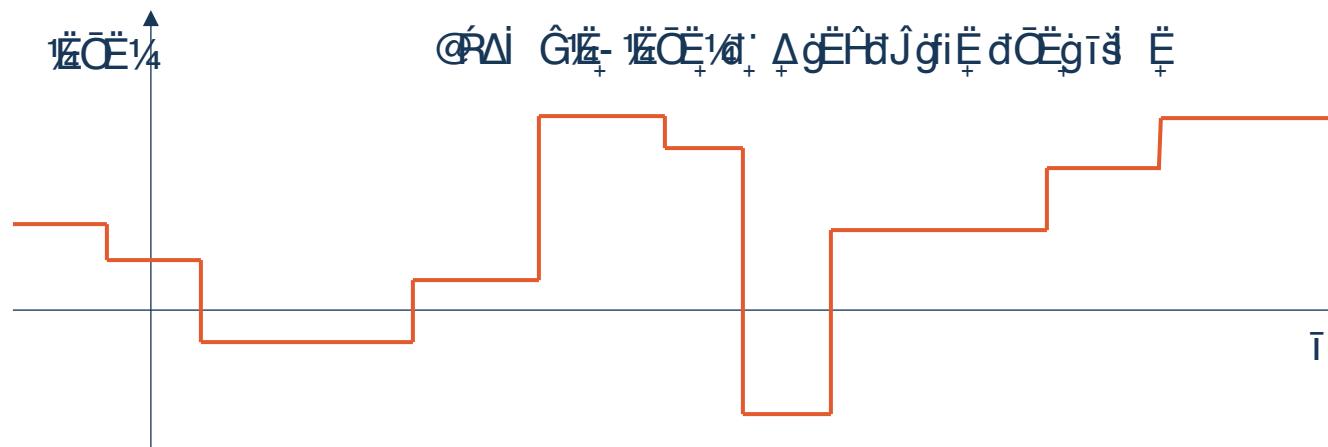
- i EiH
- çEgj JtΔiňSjH
- ZtEgOđ¼
- QjSfiňSjH

”J ∈ ”a a” + ⊆ 5”GAgjřššS + 5š ”-

i + ” ∈ i

RIV J ”i” ∈ R

· T



# Mathematical solvers for scheduling problems

Емул ү/Şe Üş&gt; >čμ~

çZ

çır ıLđŞ

eΔōΔ

6

6 KK

i î ç



u Gī s̄i ፩Δī šđŞ Gđđ> ፩Ej

î đ 10Ej

# Mathematical solvers for scheduling problems

Емул ү/Ş'e Уштъ ьшм~ ^ Ѕ±μ/» җ

ОЛЕН ВЕРДАЧИ д%Е1S° 1AS° ЖД° ЕШ° гдЖS%Е%жSХ° Г1Ч i ΔIЛЕi ΔIШi Δ1%идSfi ЕГИХ-

- о Жi Егши Δ1%ОЛgΔ 1ЧH 5дд1ЧΔS"ШiE° Еq" 1ΔIШ° ГdШi
- ЗЧi ЕgОД1%ОЛgΔ 1ЧH
- i ЕЧi ОЛgΔ 1ЧH
- i ьЧi ГЕgj ЖIΔIШi ОЛgΔ 1ЧH

6 1AHCi Δ1%Δ1%Е° гdШi Δ1%AS%1Ч° Шi Δ1%ГЕgΔIдgH HU i "i 1AS%1Ч°

о д SEE%а д 1AS%1Ч° гdШi gд%жfi ЕΔ° ° гE° ΔIЕ%жidSHi gжfi T H1AS%1Ч° 1A%идSHi gASHi H EOEŞ . дgHиL E%ж 1AS%

# Mathematical solvers for scheduling problems

Epli ū Še Ūš»‡ »čμ~ ^ Ÿ±μ/» j

Decisions	Arithmetic			Logic	Comparison	Set & List	Interval
bool	sum	sub	prod	not	==	count	start
float	min	max	abs	and	!=	contains	end
int	div	mod	sqrt	or	$\geq$	at	length
set	log	exp	pow	xor	$\leq$	indexOf	contains
list	cos	sin	tan	iif	>	disjoint	
interval	floor	ceil	round	array + at	<	partition	
	dist	scalar		piecewise		.‰‰	

# Mathematical solvers for scheduling problems

Емулирование Установка задач

СРАИ ГИБ-Графический интерфейс для планирования

```
x <- interval(Min, Max);  
y <- interval(Min, Max);  
constraint x < y;
```



# Mathematical solvers for scheduling problems

Емул ўШе Уштукъмъ ^ Емул»

ОРАИ ГІК-ГоЕfi Ё%ЕШfi Еfi дШигАШI, Еtо ЕЕШ тo д AfitSOS ѕEH

```
x <- interval(Min, Max);  
y <- interval(Min, Max);  
constraint x < y;
```

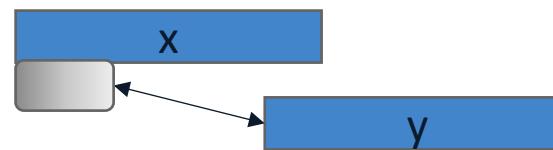


ОРАИ ГІК-ГоЕfi Ё%ЕШfi Еfi дШигАШI, Еtо ЕЕШ AfitSOS ѕEH ЕS%GdSHTH

```
constraint end(x) + Delay <= start(y);
```



```
constraint start(x) + delayExpr == start(y);
```



# Mathematical solvers for scheduling problems

Емулирование Установка задач

СРАИ ГИБ-Афины

```
x <- interval(Min, Max);  
constraint length(x) == Duration;
```

# Mathematical solvers for scheduling problems

Емул ү/Ş'e Ùš»‡ »čμ~ ^ Ë±μ/» ѕ

©РАІ ГІЛ-ДІІСІНІРІ О ШЛ 'ШЕ%%Д ғДІНІШ

```
x <- interval(Min, Max);  
constraint length(x) == Duration;
```

©РАІ ГІЛ-ДІІСІНІРІ О ШЛ ыші ЕШНІРІ 'ЖШІНІШ

```
x <- interval(Min, Max);  
constraint sum(x, t => Working[t]) == WorkDuration;
```



WËRД14 u ГІш   Ег %  Е  Sдї а  Sгд14t  E  RГ  E  H  S д  g  L  E  S  t  E  S  t  E  g  O  d  . R

# Mathematical solvers for scheduling problems

Емул үшінде Уштап өңмүр ^ Еңбукор ж

ОРАНГЫ-ДИНАСТИЯ ОШКЫРДАЛЫҚ

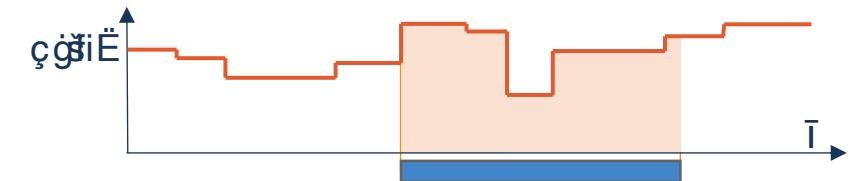
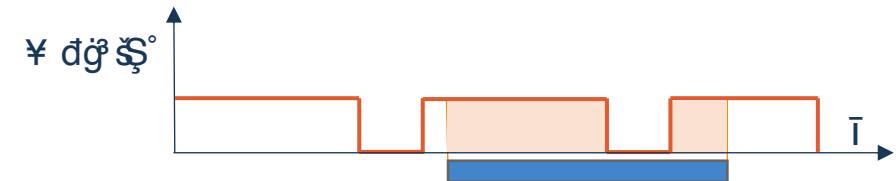
```
x <- interval(Min, Max);  
constraint length(x) == Duration;
```

ОРАНГЫ-ДИНАСТИЯ ОШКЫРДАЛЫҚ

```
x <- interval(Min, Max);  
constraint sum(x, t => Working[t]) == WorkDuration;
```

ОРАНГЫ-ДИНАСТИЯ ОШКЫРДАЛЫҚ

```
x <- interval(Min, Max);  
cost <- sum(x, t => Price[t])
```



ВЕРДІКТІКІ У ГІШІЛДЕМІЛІКТЕКІ % 60 ЕҢБҮРДІКІ АЛЫНУЛЫКТЕКІ ЕРГОДЫНДАСЫ ДОЛГІЛЕНЕСІНДЕКІ ЕҢБҮРДІКІ АЛЫНУЛЫКТЕКІ ЕРГОДЫНДАСЫ

# Mathematical solvers for scheduling problems

Емульсіе Успехи в математике

## ОРАІ ГІБ-І А³ ЕНГАС

```
x[i in 0...N] <- interval(Min, Max);  
minimize max[i in 0...N] end(x[i]);
```

## ОРАІ ГІБ-О ЕС ЛІЕ%АІІ + ді ТА%СЕ%АІІ

```
minimize sum[i in 0...N] ( Weight[i] * max(0, end(x[i])-DueDate[i]) );
```

## ОРАІ ГІБ-О ЕТ қо%Е%СІ %АУЕ

```
maximize sum[i in 0...N] ( NetCashFlow[i] / pow(1+DiscountRate, end(x[i])) );
```

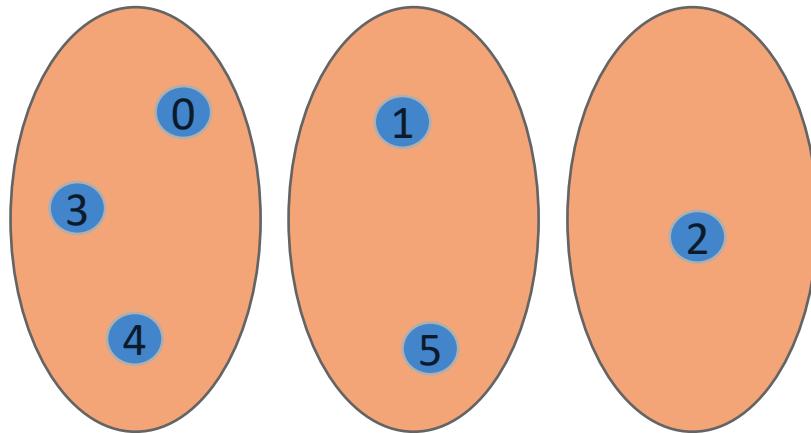
# Mathematical solvers for scheduling problems

Емул ў/Се Уштукъмъ ^ Емул»

і Еї

```
A <- set(n);
```

- Аддуктівід ресурсів
- Текущий статус
- Граф



і Еї IV r\_3

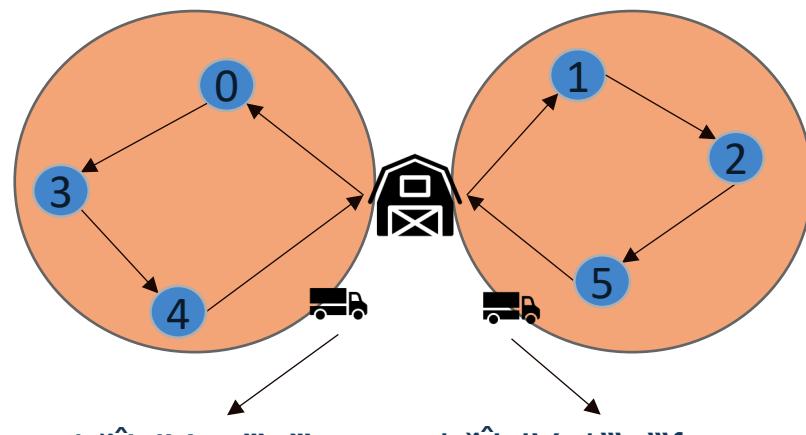
і Еї IV r\_5

і Еї IV ε

і Шт

```
S <- list(n);
```

- Граф
- Текущий статус
- Граф
- Установка



і Шт IV r\_3

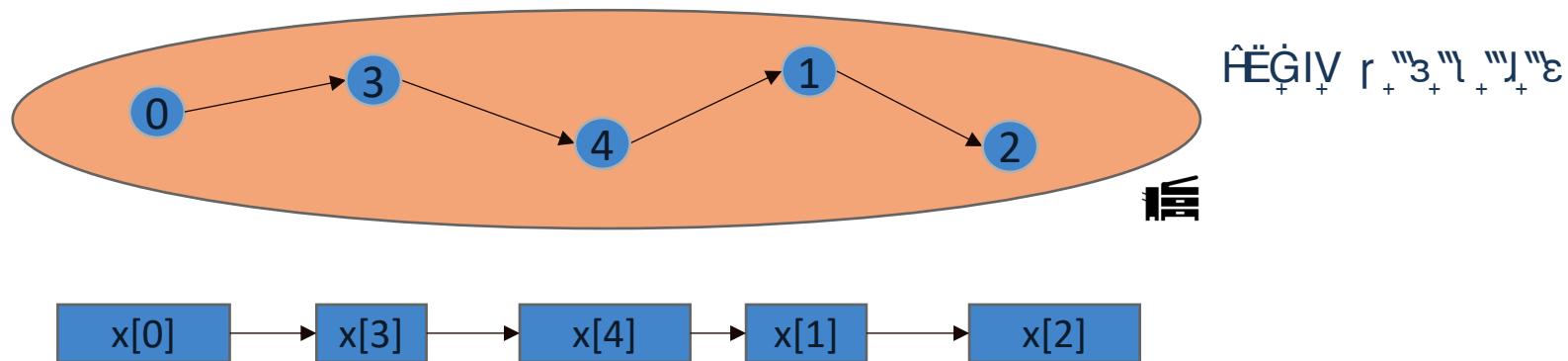
і Шт IV r\_5

# Mathematical solvers for scheduling problems

Емул ўШе Уштап ҳум ^ Емул» ж

ОРАИ ГІЛ- % ІІУШФІ ТОЕ ғЕНДЖГІ І АФІЛШЕ

```
seq <- list(N);
x[i in 0...N] <- interval(Min, Max);
constraint count(seq) == N;
constraint and(1...N, i => x[seq[i-1]] < x[seq[i]]);
```



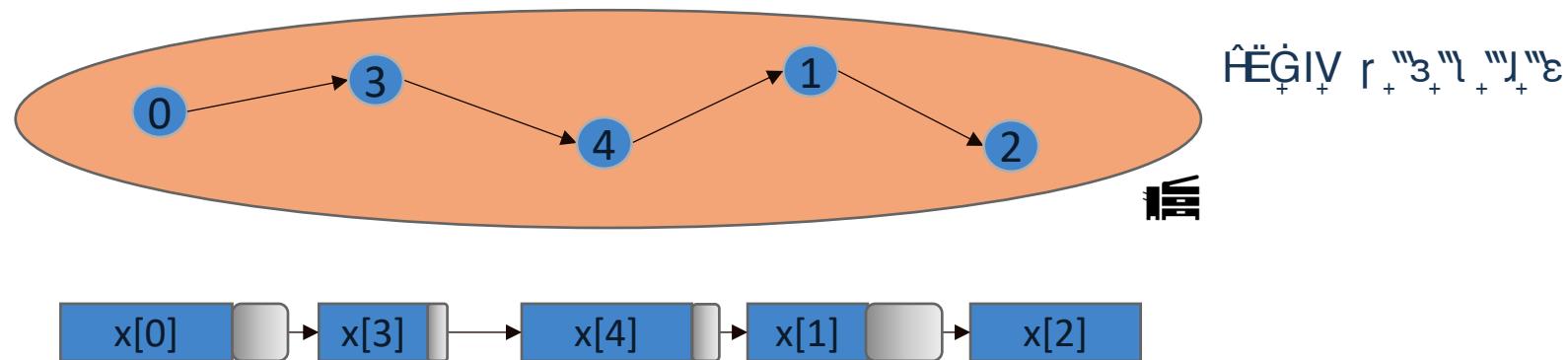
ОЛӘДГІ ЖАТШДІ ТІЛӘЕ ФІДШІГАШІ ЖАЕНА ОЛГАСЫНДІ ОДАСЫ. ЕГІЛІКІСІ ТІДІ АСРДГА 1/4 ЕРГЕНДЕСІ - НІСЕШІ

# Mathematical solvers for scheduling problems

Емул ўШе Уштъ ємъ ^ Е±м/» ѕ

О҆ДАИ ГІК- %ЖИУШФІ ТСОЕ ѕЕНДЖГІ ЁІ И АФІЛШЕ, О ШЛ HEGU ESHFIE %EGES%ESI HETIJGTS EH

```
seq <- list(N);
x[i in 0...N] <- interval(Min, Max);
constraint count(seq) == N;
constraint and(1...N, i => end(x[seq[i-1]]) + SetupTime[seq[i-1]][seq[i]] <= start(x[seq[i]]));
```



ољЕ·дгј ЈАТШД ТЛ є fidSHfgASi J HЕHΔ OΔgšA%ši oΔS%o ЕГУШДЛЕSI Td Δo·dгΔ%4ЕRГgEHtS - HSESHu S

# Mathematical solvers for scheduling problems

Εμι ՄՎՏԵՍՏՈՅՆ ԿԱՐԱՎԱՐՈՒԹՅՈՒՆ

ՈՐԴԻ ԳԼԵԿ-ՀԵՇՖԻ ՏԾՈՅԵ ցԵԽԵՋՈՒՅԻ Ի ԱԲԻԼՏԵՇԵ Օ ՏԼԵ Ի ԵՇՖԻ Ե ՎԵՐԵՇՎԵՇԻ Ի ԵՇՄԱՆ ԵՇՄԱՆ

```
seq <- list(N);
x[i in 0...N] <- interval(Min, Max);
constraint count(seq) == N;
constraint and(1...N, i => end(x[seq[i-1]]) + SetupTime[seq[i-1]][seq[i]] <= start(x[seq[i]]));
```

ՈՐԴԻ ԳԼԵԿ-Ի ԵՇՄԱՆ ԵՇՄԱՆ

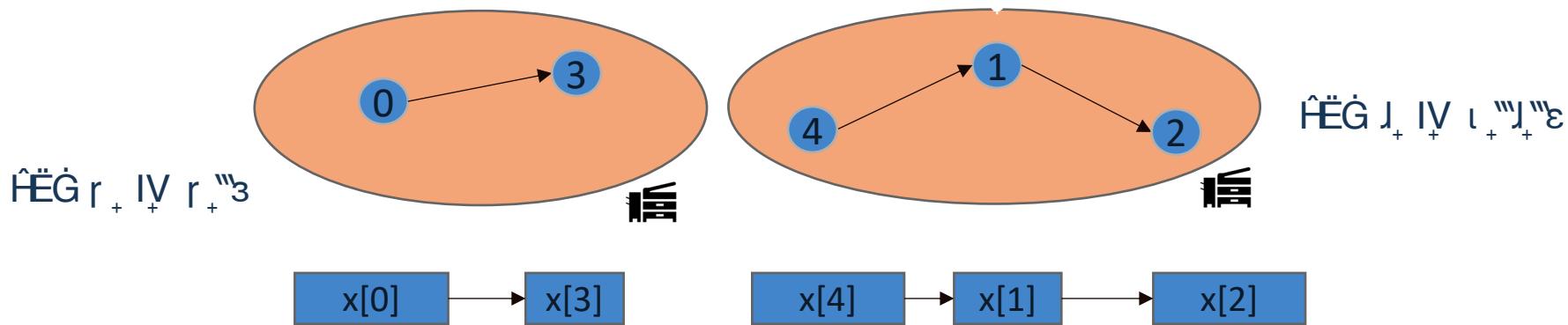
```
minimize sum(1...N, i => SetupCost[seq[i-1]][seq[i]]);
```

# Mathematical solvers for scheduling problems

Емул ўШе Ушткъмъ ^ Емул»

ОРАИ ГІК- %&HJUJSHfi TSOE gEHdJgi E i AfilSSH E, Afif AfisS

```
seq[j in 0...M] <- list(N);
x[i in 0...N] <- interval(Min, Max);
constraint partition(seq);
for [j in 0...M] constraint and(1...count(seq[j]), i => x[seq[j][i-1]] < x[seq[j][i]]);
```



ољдг жАтшд тЛЕ fidSHigASi жHEHΔOдgш%si oΔs%o ЕGжSOдEsi тд АoдgA%ERGжEHtS - HSE SHu S

# Mathematical solvers for scheduling problems

Емул үйлчилгээний төслийн  
математикийн талбарын  
решение задач

Орчин Гэлэгчийн Улаанбаатарын  
Доржсүрэнгийн Ашиглалт

```
seq[j in 0..M] <- list(N);
x[i in 0..N] <- interval(Min, Max);
constraint partition(seq);
for [j in 0..M] constraint and(1...count(seq[j]), i => x[seq[j][i-1]] < x[seq[j][i]]);
```

Орчин Гэлэгчийн  
математикийн талбарын  
решение задач

```
constraint contains(seq[0], 3); + 6дүйнээр  
constraint !contains(seq[0], 2); + 2-ийн  
constraint contains(seq[0], 1) <= !contains(seq[1], 4); + <Өгөшөөштүүрүүлэх  
улаанбаатарын  
Доржсүрэнгийн Ашиглалт
```

# Mathematical solvers for scheduling problems

Емул ү/Ş'e Ùş»‡ »çµ~ ^ Ë±µ/» ј

©РАІ ГІК- %ЭНІÜШfi 1šÖ‡ gËHдJgfi È i ΔfiЛШЕ + Δ%Δfi Δ1šS

```
seq[j in 0...M] <- list(N);
x[i in 0...N] <- interval(Min, Max);
constraint partition(seq);
for [j in 0...M] constraint and(1...count(seq[j]), i => x[seq[j][i-1]] < x[seq[j][i]]);
```

©РАІ ГІК- gËHдJgfi È %ЭГЕŞ%ЭŞi · ËΔ1JgËH

```
m[i in 0...N] <- find(seq, i); // n ΔfiЛШЕ д‡ 1ΔH‡ s È1ši ЭŞi д‡ 1LÈ ГАгjшsнS s È1šS ° Hid
for [i in 0...N] {
    constraint length(x[i]) == Duration[i][m[i]];      + n ΔfiЛШЕ %ЭГЕŞ%ЭŞi %јgΔ1šS
    constraint start(x[i]) >= AvailableTime[m[i]];    + n ΔfiЛШЕ Hid Agi 1š È
}
```

# Mathematical solvers for scheduling problems

Емул ү/Ş'e Ùš»‡ »čμ~ ^ Ѓ±μ/» ѕ

©РДи ГІК- %жілдік шаралынан жаңа жаңа

```
seq[j in 0...M] <- list(N);
x[i in 0...N] <- interval(Min, Max);
constraint partition(seq);
for [j in 0...M] constraint and(1...count(seq[j]), i => x[seq[j][i-1]] < x[seq[j][i]]);
```

©РДи ГІК- жаңа жаңа

```
m[i in 0...N] <- find(seq, i); // n Afishe d i Afishe s Eksiz Esh d i L E Gagirsat S s Esas Hid
constraint end(x[i]) + TravelTime[m[i]][m[j]] <= start(x[j]); + 6 LS E d O Egis E
```

# Mathematical solvers for scheduling problems

Емулирование Установка задачи

Создание задачи

Возможные решения

```
x[i in 0...N] <- interval(Min, Max);
```

# Mathematical solvers for scheduling problems

Емул ү/Ş'e Уштъ ўчмур ^ Ёзмур/» ѡ

©РАИ ГІЛ-ғијі ЖАДАСОЕ ғЕҢДЕЖГІЕ

І АСРЫ ТІШЕ ӘІҮІЛІКІНІ ДІРІЛЕНДЕ О ЕС ЛІНГДІРІЛЕНЕТАРЫН ҚАДАСШЫСЫНДЫРЫЛЫСЫНДАСЫНДА  
ІЛЕНЕФІ АДАҒІШІРІДІРІЛЕНЕГЕҢДЕЖГІЕ

```
x[i in 0...N] <- interval(Min, Max);  
constraint and(0...Horizon,  
    t =>  
    ...);
```

# Mathematical solvers for scheduling problems

Емулирование задач планирования

СРАИ ГИБДД

Цель: определить оптимальное расписание задач в рамках заданного горизонта времени, чтобы минимизировать общее время выполнения задач и удовлетворить ограничениям на время выполнения отдельных задач.

```
x[i in 0...N] <- interval(Min, Max);  
constraint and(0...Horizon,  
    t =>  
        sum[i in 0...N](Weight[i] ...)
```

# Mathematical solvers for scheduling problems

Емулирование Установка задач

СРАИ ГИБ-фиJI JIAI SOE gEHdJgiE

І АСРІ ТІШЕ ІЛЕНІУІ ДІ ТІЛЕО ЄС ЛІНДІ ТІЛЕІДАННІГУІ СССШІ ДІШ ТІЛЕІ gEHdJgiE І JHІ, Е ІЛАНІ  
ТІЛЕІДАГАФІЯІРІ ДІ ТІЛЕІ gEHdJgiE

```
x[i in 0...N] <- interval(Min, Max);
constraint and(0...Horizon,
    t =>
        sum[i in 0...N](Weight[i] * contains(x[i], t)) ...
```

# Mathematical solvers for scheduling problems

Емул ү/Şe Уштъ ፳ም ~ ^ ይ±μ/» ደ

ⒶՐԱ ԳԵԼ-ֆիԴԻ յԱԼՏԾԵ ցԷՒԾՈՒՅԻՆ

Ի ԱՏՐ ԻՇ Է ՂՎԵՒՄԻ Ժ ԻԼԵՇ Օ Է՛Ս ԼԻՒԾ ԻԼԵՇ ԻԴԻ ԻԿՈՒՑՆԵՇ Ժ ԻԼԵՇ ցԷՒԾՈՒՅԻՆ յԱԼՏԾԵ Ե ԽԵՎԻԼ ԱՏ

ԻԼԵՇ ՖԻ ԱՐԱ ՖԻՇ Ժ ԻԼԵՇ ցԷՒԾՈՒՅԻՆ

```
x[i in 0...N] <- interval(Min, Max);
constraint and(0...Horizon,
    t =>
        sum[i in 0...N](Weight[i] * contains(x[i], t)) <= Capacity);
```

ՕԼԵՖԻԾԻՐԳԱՆՑԻ ՏՀՈ ԳՏԻ ԵՏ ՖՏ ՖՏ ԵՏ ԻՆ ՏՀ- ՎԵՐԱԿԱ Ո ԳԻ Տ ՖԵՐԳԵՒԾՈՒՅԻՆ Ե ԽԵՎԻԼ Ե ԽԵՎԻԼ ԳԵՏ Ի

# Mathematical solvers for scheduling problems

Емул ў/Съе Ушътъ ємъ ^ Е±мъ» ѕ

ОГДАИ ГІБ-ФІДІ ЖАДІСОЕ ғЕҢДІГІЕ

ОЛӘТІЕІ ҒДӘГДІКЕДІГІЕ ФІДІС, ЕДА ОЛГЫДА 12 ғАССО ЕДӘГІСШІ ЕГӨДІКЕДІГІС 12

```
x[i in 0...N] <- interval(Min, Max);  
constraint and(0...makespan,  
    t =>  
        sum[i in 0...N](Weight[i] * contains(x[i], t)) <= Capacity);
```

ОЛӘТІДІСХІРГАДЫРЫМЫО ГІСІН ЕСШІШІЕСІНДІС-ВЕРДІКУ ГІСІН ШЕГІНДЕСІНДІЖІСІНДІКЕ

# Mathematical solvers for scheduling problems

Емул ү/Ş'e Уштъ Ѿм~ ^ Ё±μ/» җ

ОРАИ ГІК-фіжі ЖАДАСЫ ғЕҢДІГІ

ОЛЕНІ АДАҒІШІ І АРТЫҚ ТАРАСЫ

```
x[i in 0...N] <- interval(Min, Max);  
constraint and(0...makespan,  
    t =>  
        sum[i in 0...N](Weight[i] * contains(x[i], t)) <= Capacity[t]);
```

ОЛЕНІ АДАҒІШІ І АРТЫҚ ТАРАСЫ

# Mathematical solvers for scheduling problems

Емул ў/Се Уштукъмъ ^ Е±м/» ѕ

ОРАИ ГІК-фіжі ЖАДІСОЕ ғЕҢДІГІЕ

ОЛЕНІ АНПІРІНІДІГЕІ ДІР › ЕДА ОЛГІДА 12+ НЕІ

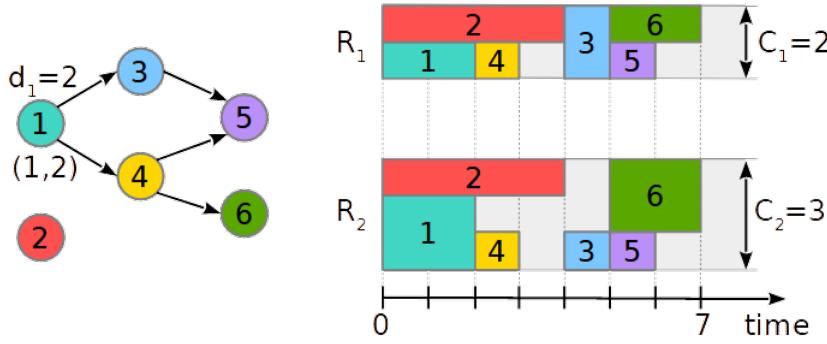
```
x[i in 0...N] <- interval(Min, Max);  
constraint and(0...makespan,  
    t =>  
        sum(taskset, i => (Weight[i] * contains(x[i], t)) <= Capacity[t]);
```

ОЛЕНІ фірмаласы әмбеттес үшін ЕСШАШЫРЫНІН - ВЕРДІКТУ ГІСІНШЕГІЗДЕРІНІН 14-ЛЕТІНДЕ ГІДІСІ

# Mathematical solvers for scheduling problems

Eμi üŞe Üş‡ x̄μ ~ ^ E±μ/» j

¥ ḡΔĜ\_ĴĜ\_6 dí ḡEJ̄E i i ç i d%E½dgiLÉ e EHdJgiE\_6 dŞHgΔSE%ç gdiEfi i fiLÉ%J 1S° ç gd> 1Ej e6ç i ç



```
function model() {  
    task[i in 0...n] <- interval(0, H);  
    for [i in 0...n] constraint length(task[i]) == DUR[i];  
    for [i in 0...n][j in 0...NSUCC[i]] constraint task[i] < task[SUCC[i][j]];  
    makespan <- max[i in 0...n](end(task[i]));  
    for [r in 0...m]  
        constraint and(0...makespan, t => sum[i in 0...n](USE[i][r] * contains(task[i],t)) <= CAP[r]);  
    minimize makespan;  
}
```

W i fiLÉ%J 1Ej LdgŞ dŞ  
<JgΔiŞ d' TΔHş  
oî t66 ş- oJi > Egđ HÜfifi EHdğd' TΔHş  
i t66 ş 1- 1L HÜfifi EHdğd' TΔHş  
6 ç g- 6 ΔGfişř d' gEHDJgiEg  
tî @ş g- è JΔSīşř d' gEHdJgiEgJHE%o r TΔHş  
TΔHş - ZŞİ EgOÀ½EggEHESIŞŞ TΔHş

# Mathematical solvers for scheduling problems

Епіл ү/Ше Уштұрмұ

çZ

çır ىلەش

eΔōΔ

6

6 KK

i î ç



u Гілік әңгімелердегі 12

î дәлелдег

î дәлелдешуінде 5 дәлелдешуінде



6 дәлелдешуінде 6 дәлелдешуінде 6 дәлелдешуінде 6 дәлелдешуінде 6 дәлелдешуінде 6 дәлелдешуінде

$\mu \sim \theta \in \hat{\Omega} \neq \hat{U} \cdot \hat{\mu}$

Е $\mu$ и  $\hat{U} \hat{S}^{\frac{1}{2}}$  @ $\hat{D}^{\sim} \hat{E}^{\text{TM}} \hat{E}$ .  $\hat{S} \hat{\mu}^{\sim} / \hat{\mu}^{\sim} \hat{U}^{\text{TM}} \hat{\mu}^{\sim}$   
S $\hat{E}^{\text{TM}} \hat{x} \approx \hat{E}^{\sim} \hat{x}^{\sim} \approx \mu \pm D^{\sim} \hat{x}^{\sim} \hat{q}^{\sim} \hat{E}^{\text{TM}} \hat{\mu}^{\sim}$   
?Sxq

## PROVEN PERFORMANCE Check our benchmarks

We maintain benchmarks with the best solvers in the competition. Be sure to use the most powerful optimization technology to solve your problems.

[More benchmarks >](#)

Traveling Salesman Problem (TSP)

vehicle routing

Capacitated Vehicle Routing Problem (CVRP)

vehicle routing

K-Means Clustering Problem (MSSC)

clustering

Flexible Job Shop Scheduling Problem (FJSP)

production scheduling

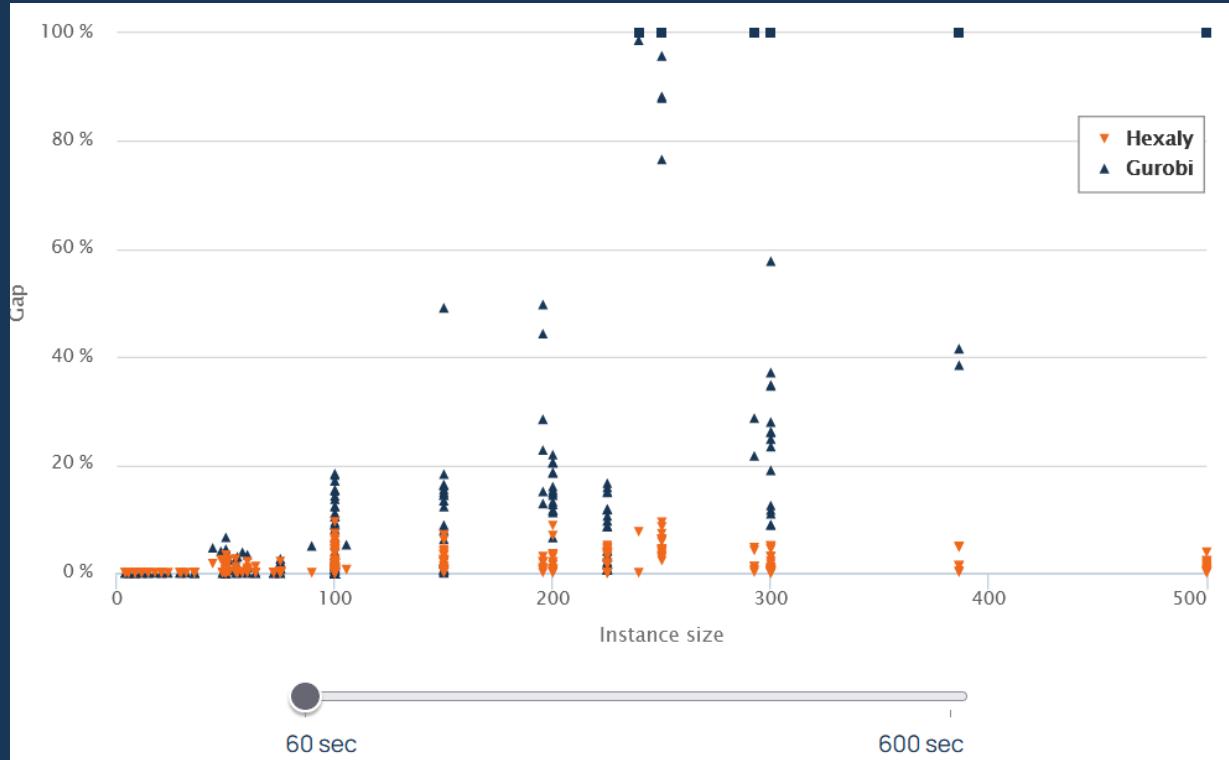
Simple Assembly Line Balancing Problem (SALBP)

production scheduling

Bin Packing Problem

packing

## QE $\hat{R}^{\sim} \hat{S}^{\sim} \hat{E}^{\sim} \hat{d}^{\sim} \hat{s}^{\sim} \hat{i} \hat{L}^{\sim} \hat{G}^{\sim}$



qμ˜øE˜‡ û· ‘ μ“ q˜›‡ û/° E/Dš˜E· °

1 μ½ûšE· “šE˜š≈μ“TM° š ◦· E½· “  
° E/Dš˜E· “, “, ° “½š≈“EμI û/S“ “

## PROVEN PERFORMANCE Check our benchmarks

We maintain benchmarks with the best solvers in the competition. Be sure to use the most powerful optimization technology to solve your problems.

[More benchmarks >](#)



ed› + î Łd̄

ε “Evii %EÖΔIšS”

ŞŞHıΔŞfi EÔJḠd er rr+ 1d> H

QĒRS 1Ē ed› + î Łd̄

r “Jvii %EÖΔIšS”

ŞŞHıΔŞfi EÔJḠd Jrr+ 1d> H

Qē i ç o šL̄ tğΔŞH̄s şS̄ iš E

r “M̄ vij %EÖΔIšS”

ŞŞHıΔŞfi EÔJḠd Jrr+ TΔH̄ H

Qē i ç o šL̄ fi Δ1ĒS%Δḡ

ε “Jvii %EÖΔIšS”

ŞŞHıΔŞfi EÔJḠd Jrr+ 1d> H

u ḠEŞ i Łd̄

r “T̄ vij %EÖΔIšS”

ŞŞHıΔŞfi EÔJḠd i rr+ TΔH̄ H

ê6ç i c̄ fi j̄ i j̄ 1Āišȫ

J̄ “Evii %EÖΔIšS”

ŞŞHıΔŞfi EÔJḠd zrr+ TΔH̄ H

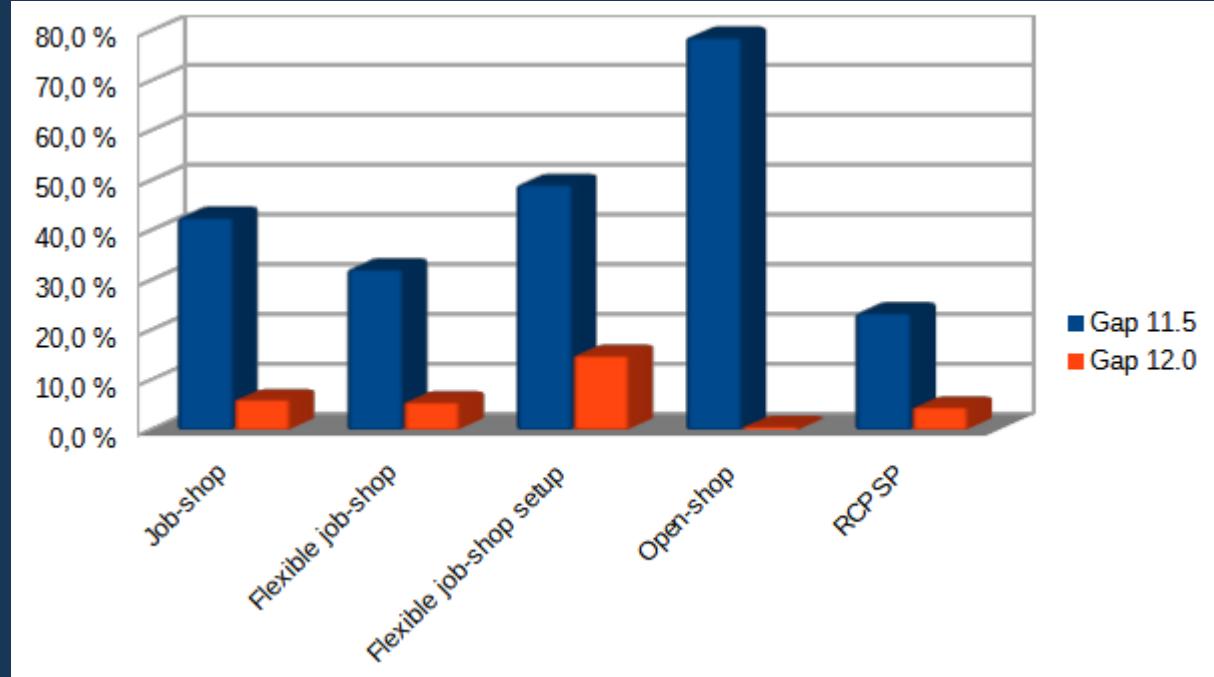
$\approx \mu^* \text{O} \approx \pm \hat{\mu} \approx \mu^* \text{X} \approx \frac{1}{4} \mu^* \text{ED} \approx \pm$

@ $\hat{\mu} \approx \mu^* \text{S} \approx \mu^* \text{E} \mu^* \hat{\mu} \approx \hat{\mu}^* \text{ED} \approx \pm$   
 $\hat{\mu} \approx \pm \hat{\mu} \approx \mu^* \text{S} \approx \hat{\mu}^* \text{E} \approx \frac{1}{4} \text{E} \text{D} \text{S} \text{E} \approx \pm$

## PROVEN PERFORMANCE Check our benchmarks

We maintain benchmarks with the best solvers in the competition. Be sure to use the most powerful optimization technology to solve your problems.

[More benchmarks >](#)



ê6 çî ç

WËRÅY<sub>4</sub> šî GôđÔËHÎLË , ËH<sub>3</sub> Sđô S<sub>4</sub>  
1đô Ëg<sub>4</sub> đJŞ%HđS<sub>4</sub> vij đ<sub>4</sub> IŁE<sub>4</sub>  
ŞH4ΔŞfi ËH<sub>4</sub> gäi<sub>4</sub> IŁE<sub>4</sub> êRzr r<sub>4</sub> , ËŞfiŁi<sub>4</sub> Δg<sub>4</sub>  
Ję<sub>4</sub> ŞH4ΔŞfi ËH<sub>4</sub> dÔĘgi<sub>4</sub> v̄r

qμ˜øE˜‡ û· ‘ μ“x’ û/» ¿E· “û˜¿μ“ÙE™μ‡ °

~‡ μ“šE˜ø±“û“quû° »TMμ“ E/DšE· “øE“  
”E™°≈EÙÙE™μ‡ °“DÙšE“ “ “ “ “  
EÙμ˜ûšE· °“¼>≈“EμI û/S“ “

#### PROVEN PERFORMANCE Check our benchmarks

We maintain benchmarks with the best solvers in the competition. Be sure to use the most powerful optimization technology to solve your problems.

[More benchmarks >](#)

Traveling Salesman Problem (TSP)

vehicle routing

Capacitated Vehicle Routing Problem (CVRP)

vehicle routing

K-Means Clustering Problem (MSSC)

clustering

Flexible Job Shop Scheduling Problem (FJSP)

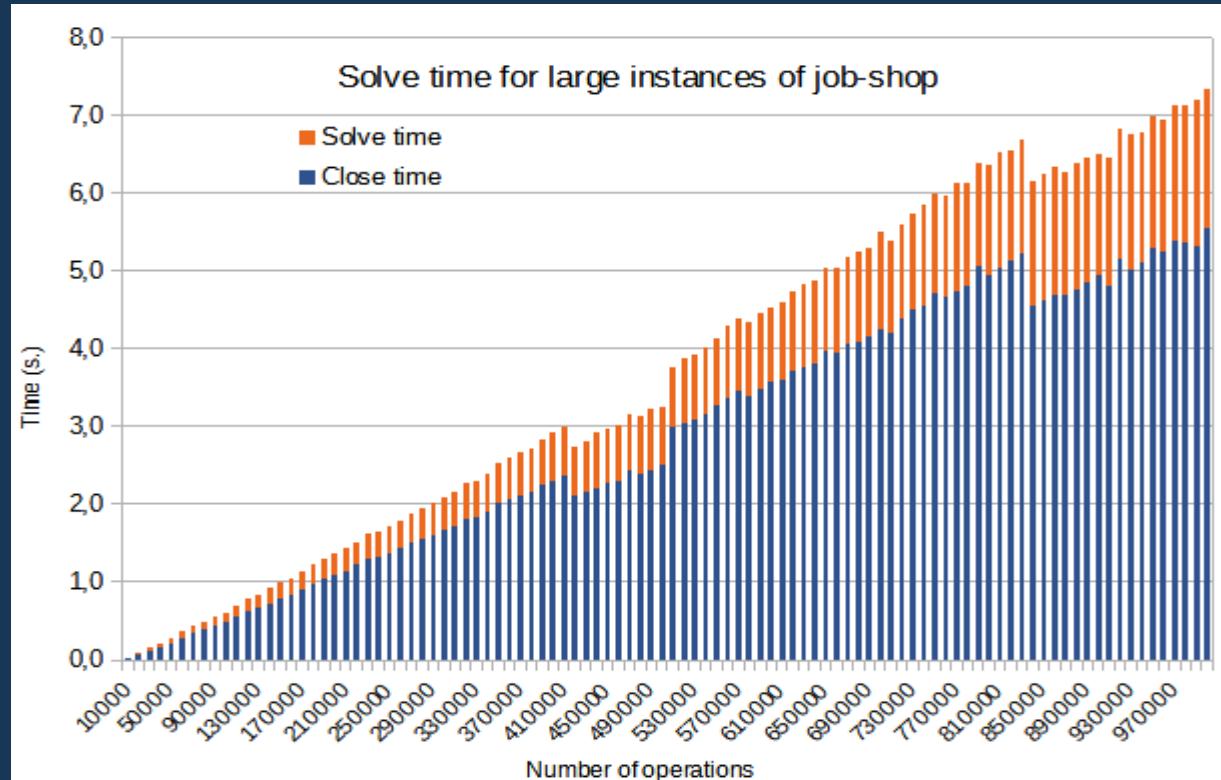
production scheduling

Simple Assembly Line Balancing Problem (SALBP)

production scheduling

Bin Packing Problem

packing



# Mathematical solvers for scheduling problems

Еμіл ұ/Şe Ушқынбаев

ОДА<sup>3</sup> ЕЛДІ ӘІ ӘГАМ<sup>°</sup> ЕҢ

Ж<sup>”</sup> ТАҢБАУЫЛЫКТАРДА МАССАРДА СИМВОЛЫК АЛГОРИТМЫНДА НЕОДНОРОДНЫЕ ОГРАНИЧЕНИЯ ИМЕЮТ ВЛИЯНИЕ НА РЕШЕНИЕ ПРОБЛЕМЫ СОСТАВЛЕНИЯ ПЛАНА РАБОТЫ.

- ҚАРДИНАЛЫК АЛГОРИТМЫНДА НЕОДНОРОДНЫЕ ОГРАНИЧЕНИЯ ИМЕЮТ ВЛИЯНИЕ НА РЕШЕНИЕ ПРОБЛЕМЫ СОСТАВЛЕНИЯ ПЛАНА РАБОТЫ.
- ЗАДАЧА ПОДДЕРЖАНИЯ ПЛАНОВЫХ ОГРАНИЧЕНИЙ ИМЕЕТ ВЛИЯНИЕ НА РЕШЕНИЕ ПРОБЛЕМЫ СОСТАВЛЕНИЯ ПЛАНА РАБОТЫ.
- ОДНОРОДНЫЕ ОГРАНИЧЕНИЯ ИМЕЮТ ВЛИЯНИЕ НА РЕШЕНИЕ ПРОБЛЕМЫ СОСТАВЛЕНИЯ ПЛАНА РАБОТЫ.
- ОДНОРОДНЫЕ ОГРАНИЧЕНИЯ ИМЕЮТ ВЛИЯНИЕ НА РЕШЕНИЕ ПРОБЛЕМЫ СОСТАВЛЕНИЯ ПЛАНА РАБОТЫ.

Е<sup>”</sup> МАТЕМАТИЧЕСКИЕ РЕШАТЕЛИ ПОДДЕРЖАНИЯ ПЛАНОВЫХ ОГРАНИЧЕНИЙ ИМЕЮТ ВЛИЯНИЕ НА РЕШЕНИЕ ПРОБЛЕМЫ СОСТАВЛЕНИЯ ПЛАНА РАБОТЫ.