Flow-based Heuristics for Optimal Planning: Landmarks and Merges

Blai Bonet
Universidad Simon Bolivar
Caracas, Venezuela
bonet@ldc.usd.ve

Menkes van den Briel
NICTA & Australian National University
Canberra, Australia
menkes@nicta.com.au

USB

NICTA

Outline

- Background and motivation
- Base model
- Strengthening the base model
  - Adding constraints derived from landmarks
  - Adding constraints derived from merges
- Results
- Conclusions

Background

An LP-Based Heuristic for Optimal Planning**. Van den Briel, Benton, Kambhampati, and Vossen. CP2007.**


An LP-Based Heuristic for Optimal Planning

Menkes van den Briel[1], J. Benton[2], Subbarao Kambhampati[2], and Thomas Vossen[3]

[1] Arizona State University, Department of Industrial Engineering,
[2] Department of Computer Science and Engineering,
Tempe AZ, 85287, USA
{menkes,j.benton,rao}@asu.edu
[3] University of Colorado, Leeds School of Business,
Boulder CO, 80309, USA
vossen@colorado.edu

An Admissible Heuristic for SAS+ Planning Obtained from the State Equation**. Bonet. IJCAI2013.**


Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence

An Admissible Heuristic for SAS$^+$ Planning Obtained from the State Equation

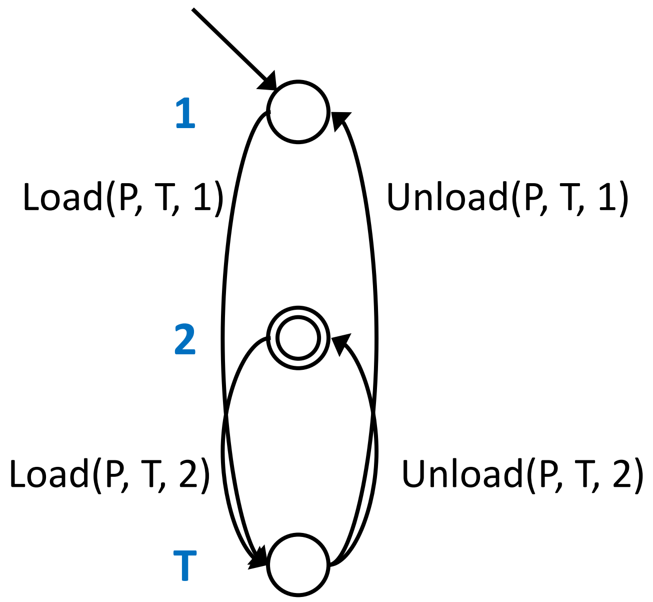
Blai Bonet
Departamento de Computación
Universidad Simón Bolívar
Caracas, Venezuela
bonet@ldc.usb.ve

Flow-based linear programming model

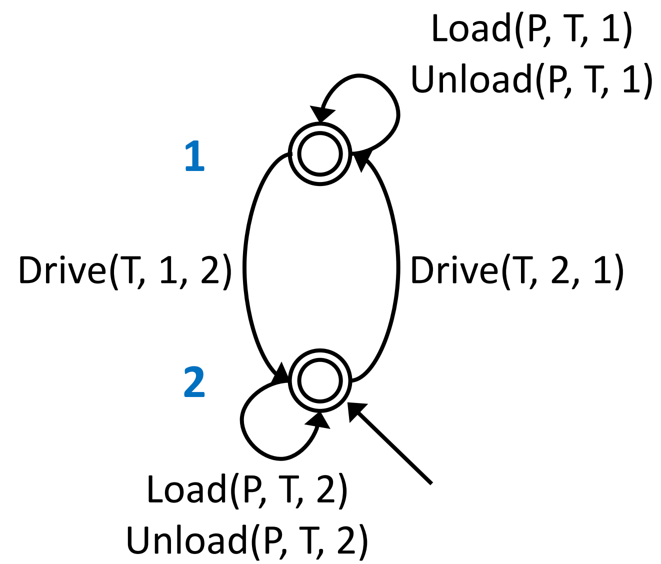
- **SAS+ planning** task with action cost $P =< V, A, s_0, s_*, c >$
  - V set of variables each with a finite domain D_X
  - A set of actions $< Pre, Post, Prev >$
  - s_0 initial state
  - s_* goal state (partial valuation)
  - c non-negative action cost
- **Domain transition graph** (DTG)
  - Node for each value in D_X
  - Labeled arc for each transition T_X

$$
\begin{aligned}
T_X = \; & \{(x, a, x'): X = x \in Pre, X = x' \in Post\} \cup \\
& \{(\perp, a, x'): X \notin Var(Pre), X = x' \in Post\} \cup \\
& \{(x, a, x): X = x \in Prev\}
\end{aligned}
$$

Example



DTG(P)



DTG(T)

Flow-based linear programming model

- A **flow** for a planning task $P = < V, A, s_0, s_*, c >$ is a function $f: A \rightarrow \mathbb{R}^+$ mapping action labels into non-negative real numbers

- Minimize

$$\sum_{a \in A} c(a) f(a)$$

- Subject to

$$\sum_{(x', a, x) \in T_X} f(a) - \sum_{(x, a, x') \in T_X} f(a) = 0$$

Flow-based linear programming model

- Minimize

$$\sum_{a \in A} c(a) f(a)$$

- Subject to

$$\sum_{(x',a,x) \in T_X} f(a) - \sum_{(x,a,x') \in T_X} f(a) = \begin{cases} 1 & \text{if } x \notin s_0, x \in s_* \\ -1 & \text{if } x \in s_0, x \notin s_* \\ 0 & \text{otherwise} \end{cases}$$

Flow-based linear programming model

- Minimize

$$\sum_{a \in A} c(a) f(a)$$

- Subject to

$$\sum_{(x',a,x) \in T_X} f(a) - \sum_{(x,a,x') \in T_X} f(a) \geq \begin{cases} 1 & \text{if } x \notin s_0, x \in s_* \\ -1 & \text{if } x \in s_0, x \notin s_* \\ 0 & \text{otherwise} \end{cases}$$

Due to incomplete actions (see paper for details)

Theorem 1. The solution to the base model h_{base} can be used as an admissible heuristic

Flow-based heuristic

- Base heuristic is effective and efficient
  - **Bonet. IJCAI2013**

Domain	LM-cut	base
Airport(50)	28	20
Barman-opt11(20)	4	4
Blocks(35)	28	28
Depot(22)	7	7
Driverlog(20)	13	11
Elevators-opt08(30)	22	9
Elevators-opt11(20)	17	7
Floortile-opt11(20)	6	4
Freecell(80)	15	35
Grid(5)	2	1
Gripper(20)	7	6
Logistics00(28)	20	15
Logistics98(35)	6	2
Miconic(150)	141	50
Mprime(35)	22	18
Mystery(30)	19	15
Nomystery-opt11(20)	14	10
Openstacks-opt08(30	20	15
Openstacks-opt11(20)	15	7
Openstacks(30)	7	7
Parcprinter-08(30)	18	28
Parcprinter-opt11(20)	13	20

#Problems solved in 30mins

Domain	LM-cut	base
Parking-opt11(20)	2	1
Pathways-noneg(30)	5	4
Pegsol-08(30)	27	28
Pegsol-opt11(30)	17	18
Pipesworld-notank(50)	17	15
Pipesworld-tank(50)	11	10
Psr-small(50)	49	50
Rovers(40)	7	6
Satellite(36)	7	6
Scanalyzer-08(30)	15	13
Scanalyzere-opt11(20)	12	10
Sokoban-opt08(30)	28	16
Sokoban-opt11(20)	20	15
Tidybot-opt11(20)	13	5
Tpp(30)	6	8
Transport-opt08(30)	11	10
Transport-opt11(20)	6	6
Trucks(30)	10	9
Visitall-opt11(20)	10	17
Woodworking-opt08(30)	16	12
Woodworking-opt11(20)	11	7
Zenotravel(20)	12	9

Total(1396)	**756**	**594**

Flow-based heuristic

– **Van den Briel, Benton, Kambhampati, and Vossen. CP2007**

Problem	Opt	*h*+	LP–
logitstics4-0	20	19	16
logitstics4-1	19	17	14
logitstics4-2	15	13	10
logitstics5-1	17	15	12
logitstics5-2	8	8	6
logistics6-1	14	13	10
logistics6-9	24	21	18
logistics12-0	42	39	32
logistics15-1		66	54
driverlog1	7	6	3
driverlog2	19	14	12
driverlog3	12	11	8
driverlog4	16	12	11
driverlog6	11	10	8
driverlog7	13	12	11
driverlog13	26	21	15
driverlog19		89	60
driverlog20		84	60

Problem	Opt	*h*+	LP–
zenotravel1	1	1	1
zenotravel2	6	4	3
zenotravel3	6	5	4
zenotravel4	8	6	5
zenotravel5	11	11	8
zenotravel6	11	11	8
zenotravel13	26	23	18
zenotravel19		62	46
zenotravel20			50
tpp1	5	4	3
tpp2	8	7	6
tpp3	11	10	9
tpp4	14	13	12
tpp5	19	17	15
tpp6	25	21	21
tpp28			150
tpp29			
tpp30			174

Heuristic value at initial state

Flow-based heuristic

- Base heuristic is weak and can be improved significantly
  - Van den Briel, Benton, Kambhampati, and Vossen. CP2007

Problem	Opt	*h+*	LP–	LP
logitstics4-0	20	19	16	20
logitstics4-1	19	17	14	19
logitstics4-2	15	13	10	15
logitstics5-1	17	15	12	17
logitstics5-2	8	8	6	8
logistics6-1	14	13	10	14
logistics6-9	24	21	18	24
logistics12-0	42	39	32	42
logistics15-1		66	54	67
driverlog1	7	6	3	7
driverlog2	19	14	12	19
driverlog3	12	11	8	11
driverlog4	16	12	11	16
driverlog6	11	10	8	11
driverlog7	13	12	11	13
driverlog13	26	21	15	24
driverlog19		89	60	97
driverlog20		84	60	90

Problem	Opt	*h+*	LP–	LP
zenotravel1	1	1	1	1
zenotravel2	6	4	3	6
zenotravel3	6	5	4	6
zenotravel4	8	6	5	8
zenotravel5	11	11	8	11
zenotravel6	11	11	8	11
zenotravel13	26	23	18	24
zenotravel19		62	46	67
zenotravel20			50	69
tpp1	5	4	3	5
tpp2	8	7	6	8
tpp3	11	10	9	11
tpp4	14	13	12	14
tpp5	19	17	15	19
tpp6	25	21	21	25
tpp28			150	
tpp29				
tpp30			174	

Heuristic value at initial state

Motivation

- Base heuristic is effective and efficient yet can be improved significantly
  - Adding constraints derived from **landmarks**
    Bonet. IJCAI2013
  - Adding constraints derived from **merges**
    Van den Briel, Benton, Kambhampati, and Vossen. CP2007

Landmarks

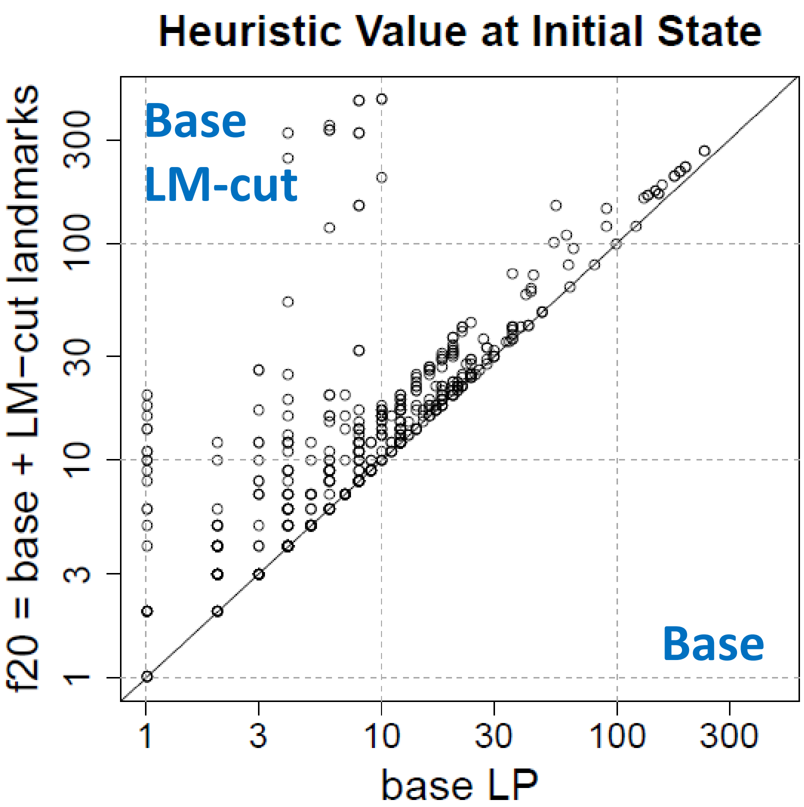
- A landmark $L \subseteq A$ for a planning task $P =< V, A, s_0, s_*, c >$ is a subset of actions such that every plan P contains at least one action in L
- A set of landmarks $\mathcal{L}$ (**state dependent**) introduces the following constraints

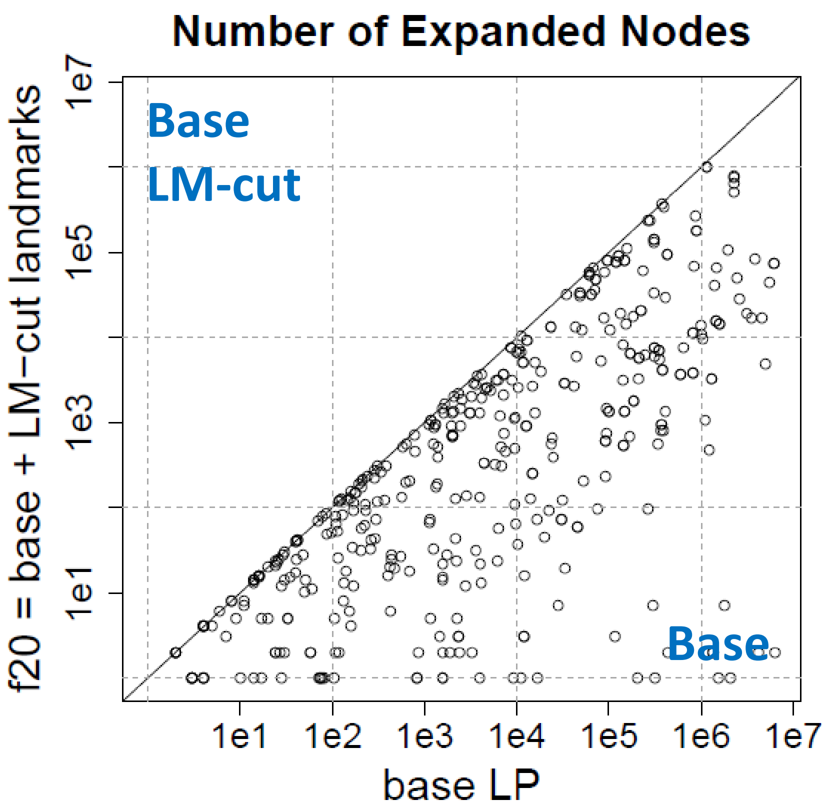
$$\sum_{a \in L} f(a) \geq 1 \qquad \forall L \in \mathcal{L}$$

Theorem 2. The set of landmark constraints is admissible. As a result, the solution to the base model with landmark constraints $h_{base}^{\mathcal{L}}$ can be used as an admissible heuristic.
Further, $h_{base}^{\mathcal{L}}$ dominates $h_{\mathcal{L}}^{*}$. In particular, $h_{base}^{LM-cut} \geq h_{LM-cut}^{*} \geq h_{LM-cut}$

Results

- Heuristic performance across common solved tasks
  - Base model versus Base model with LM-cut landmarks





Heuristic comparison

Results

- Coverage results
  - f10 **Zhu-Givan** method, f20 **LM-cut** method

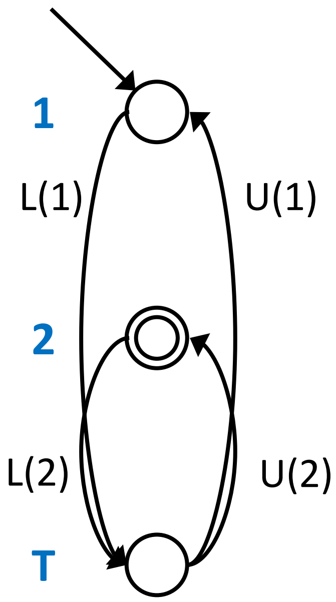
Domain	base	f 10	f 20
Airport(50)	20	26	25
Barman-opt11(20)	4	4	4
Blocks(35)	28	28	29
Depot(22)	7	7	7
Driverlog(20)	11	11	13
Elevators-opt08(30)	9	9	18
Elevators-opt11(20)	7	7	15
Floortile-opt11(20)	4	4	6
Freecell(80)	35	47	27
Grid(5)	1	2	1
Gripper(20)	6	7	5
Logistics00(28)	15	14	20
Logistics98(35)	2	3	6
Miconic(150)	50	57	140
Mprime(35)	18	20	20
Mystery(30)	15	16	16
Nomystery-opt11(20)	10	8	12
Openstacks-opt08(30	15	12	15
Openstacks-opt11(20)	7	7	7
Openstacks(30)	7	8	7
Parcprinter-08(30)	28	28	29
Parcprinter-opt11(20)	20	20	20

Domain	base	f 10	f 20
Parking-opt11(20)	1	1	1
Pathways-noneg(30)	4	4	5
Pegsol-08(30)	28	26	27
Pegsol-opt11(30)	18	16	17
Pipesworld-notank(50)	15	16	11
Pipesworld-tank(50)	10	10	9
Psr-small(50)	50	50	50
Rovers(40)	6	6	7
Satellite(36)	6	6	7
Scanalyzer-08(30)	13	13	11
Scanalyzer-opt11(20)	10	9	8
Sokoban-opt08(30)	16	20	27
Sokoban-opt11(20)	15	16	20
Tidybot-opt11(20)	5	11	8
Tpp(30)	8	8	8
Transport-opt08(30)	10	10	11
Transport-opt11(20)	6	5	6
Trucks(30)	9	9	9
Visitall-opt11(20)	17	17	19
Woodworking-opt08(30)	12	14	20
Woodworking-opt11(20)	7	9	15
Zenotravel(20)	9	9	11

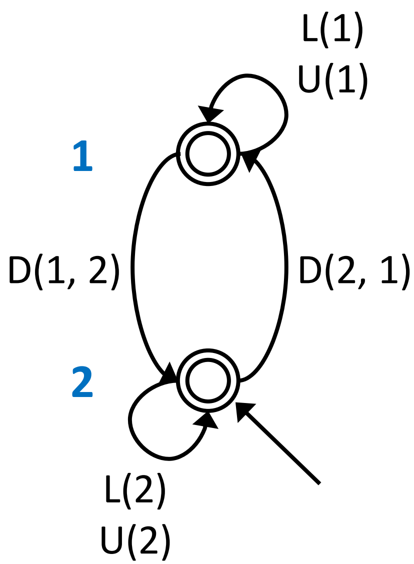
Total(1396)	**594**	**630**	**749**

#Problems solved in 30mins

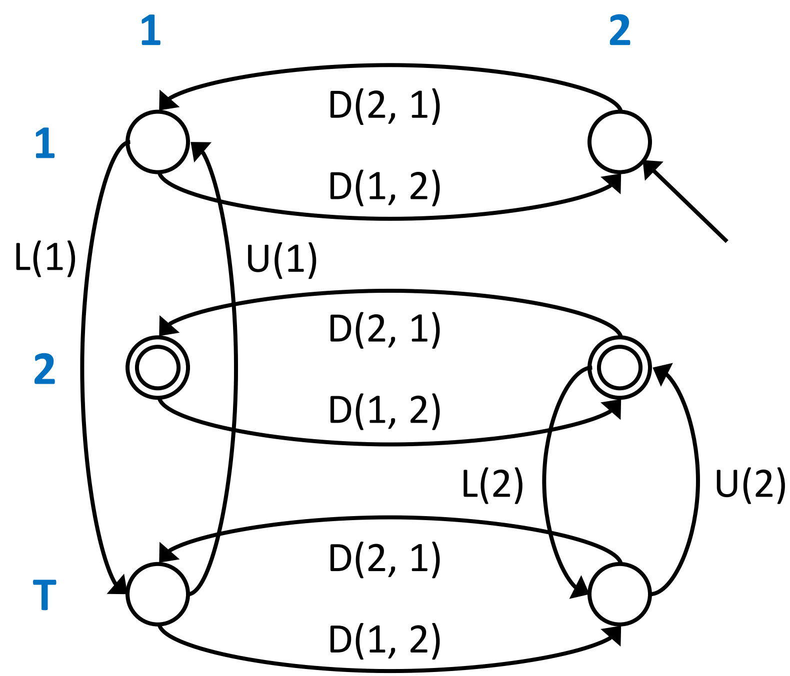
Example



DTG(P)



DTG(T)



DTG(P, T)

Merges

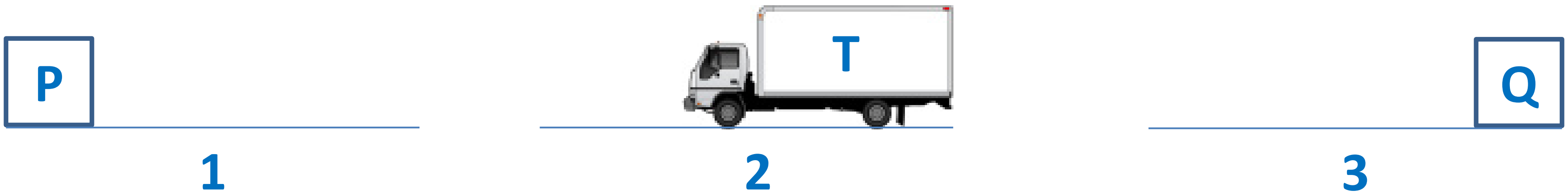
- A merge combines two or more variables into one “super” variable.
- A merged variable introduces the following constraints

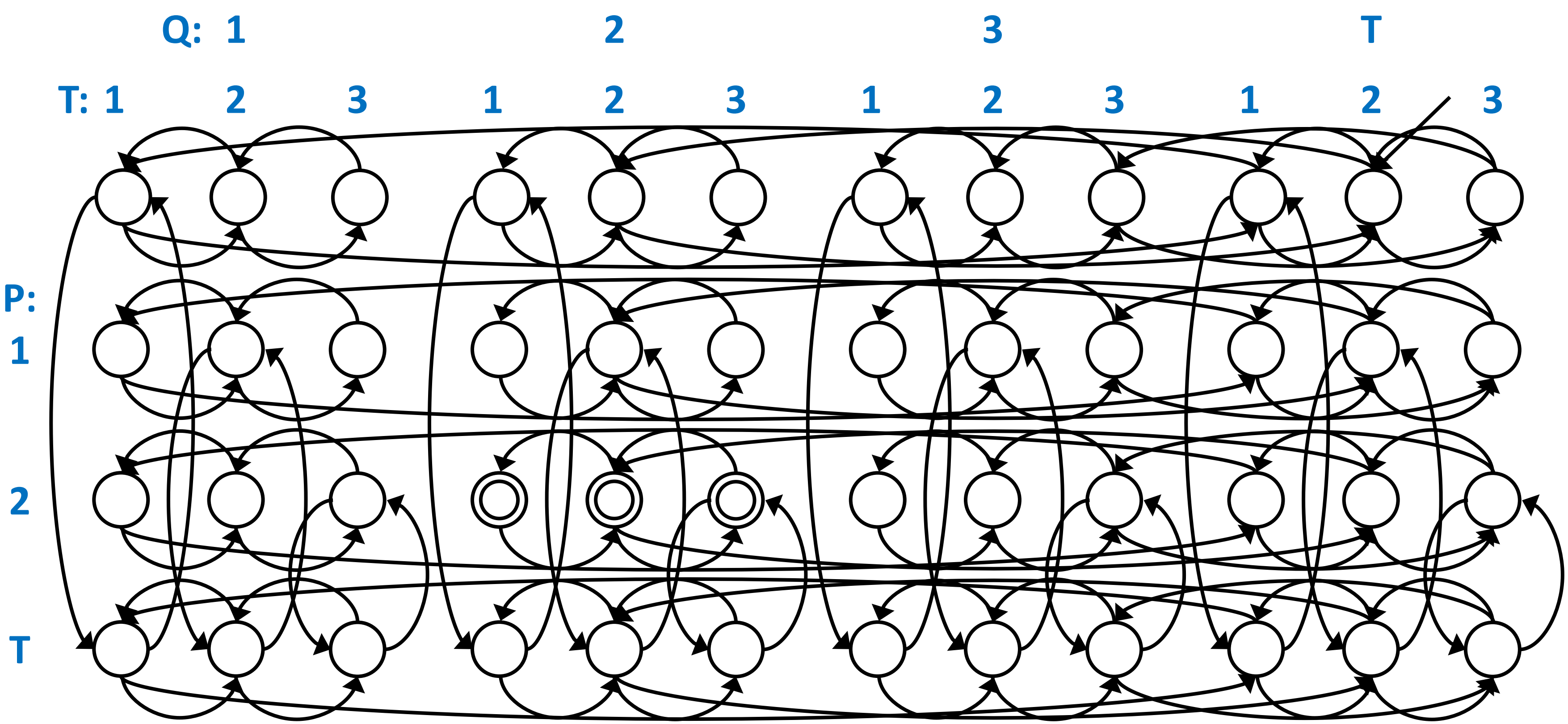
$$\sum_{(x',a,x)\in T_X} f(a) - \sum_{(x,a,x')\in T_X} f(a) \geq \begin{cases} 1 \text{ if } x \notin s_0, x \in s_* \\ -1 \text{ if } x \in s_0, x \notin s_* \\ 0 \text{ otherwise} \end{cases}$$

$$f(a) \geq \sum_{a' \in Copies(a,Z)} f(a')$$

Theorem 3. The set of merge constraints is admissible. As a result, the solution to the base model with merge constraints can be used as an admissible heuristic

Merges



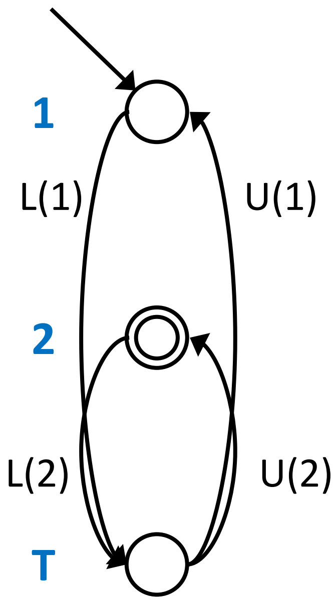


DTG(P, T, Q)

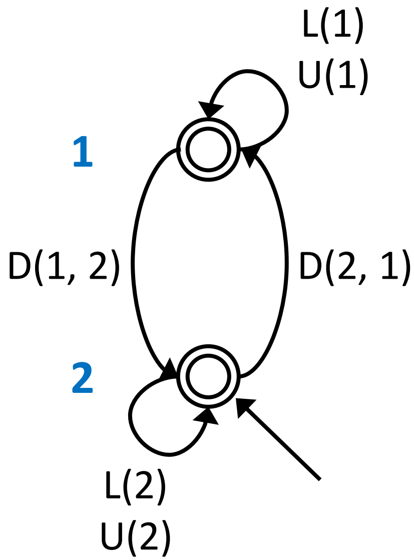
Dynamic merging

- **Dynamic merging** is dynamic in all ways
  - Incrementally merge more variables
  - Incrementally merge variables
- Very simple merge strategy
  - Merge prevail conditions with the preconditions of an action $merge(p, q)$ for all atoms $p \in Prev(a)$ and $q \in Pre(a)$

Example

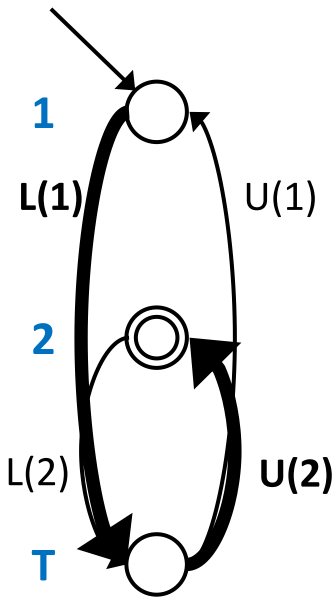


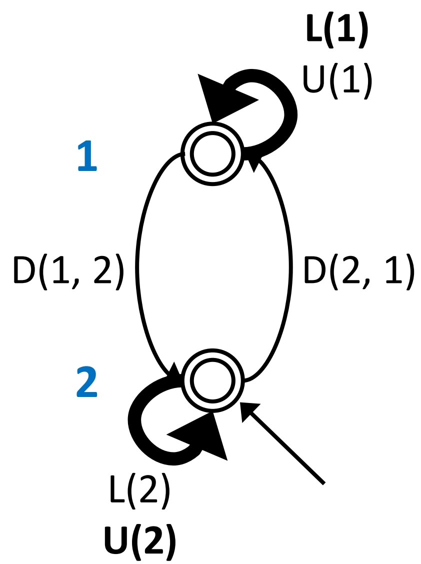
DTG(P)



DTG(T)

Example

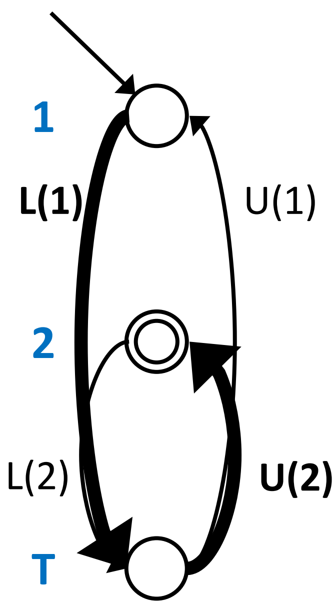




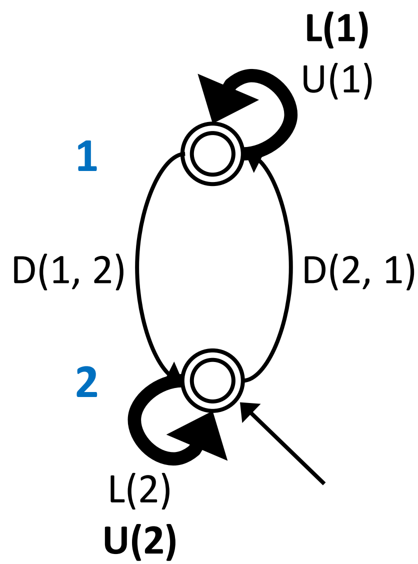
***h* = 2**

DTG(P) **DTG(T)**

Example

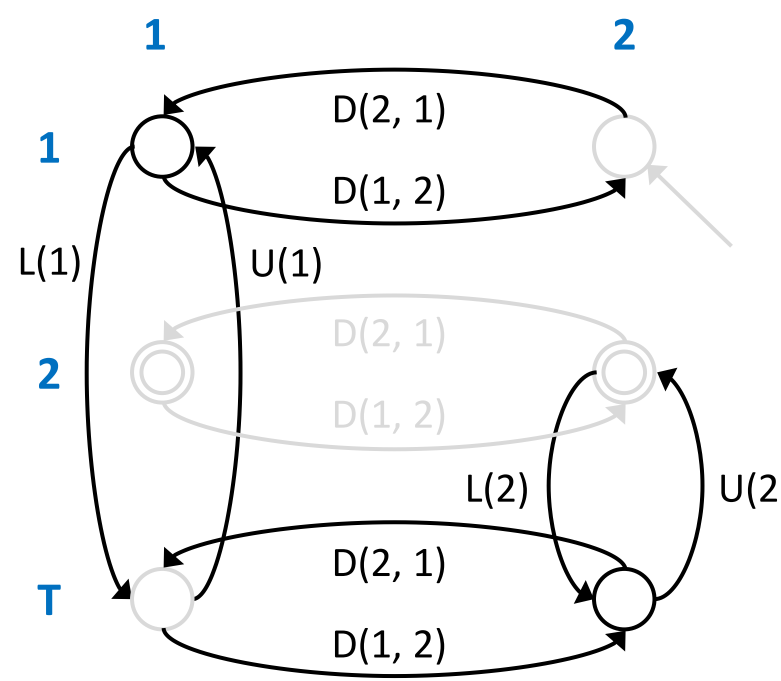


DTG(P)



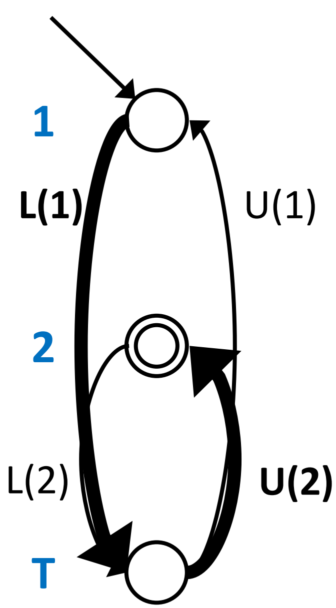
DTG(T)

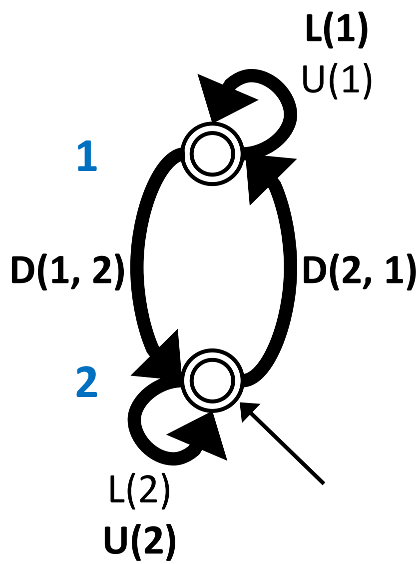
h = 2

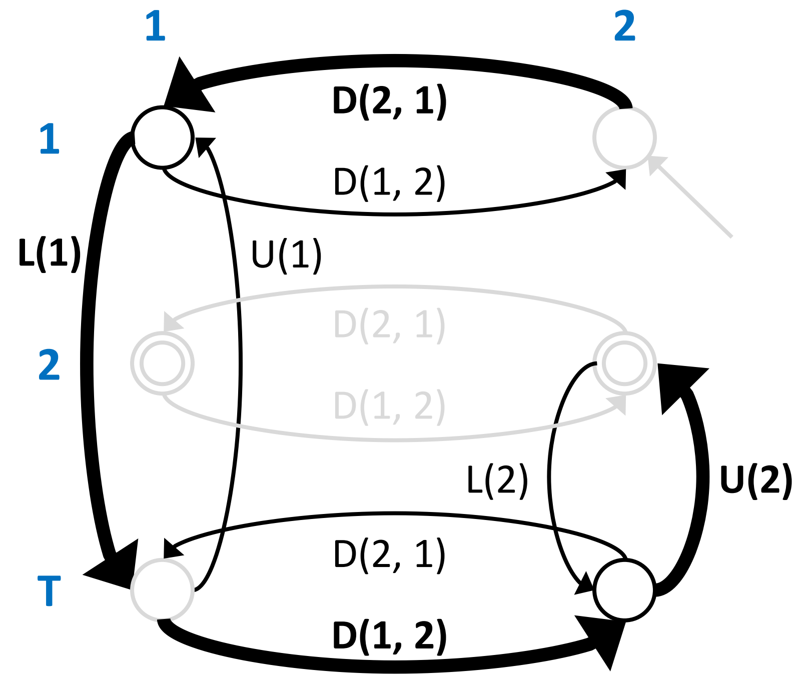


DTG(P, T)

Example







h = 4

DTG(P)

DTG(T)

DTG(P, T)

Results

- All experiments run with **Fast Downward**
  - **Helmert, Journal of Artificial Intelligence Research, 2006**
- Setup
  - AMD Opteron 6378 CPUs running at 2.4GHz
  - 2Gb memory limit
  - 1800 seconds timeout
  - Using IBM CPLEX 12.5.1 as the LP solver
- Heuristics
  - LM-cut
  - $h^{*}_{\text{LM-cut}}$
  - base
  - fXY
    - X = 0, without landmarks, X = 1, with Zhu-Givan, X = 2, with LM-cut
    - Y = 0, without merge, Y = 1, with merge

Results

- Coverage results

	LM-cut			Base Merge			Base LM-cut	Base LM-cut Merge
Domain	**LM-cut**	**$h^*_{\text{LM-cut}}$**	**base**	**f 01**	**f 10**	**f 11**	**f 20**	**f 21**
Airport(50)	28	28	20	22	26	22	25	22
Barman-opt11(20)	4	4	4	0	4	0	4	0
Blocks(35)	28	28	28	28	28	28	29	29
Depot(22)	7	7	7	6	7	5	7	5
Driverlog(20)	13	13	11	15	11	15	13	15
Elevators-opt08(30)	22	19	9	21	9	21	18	21
Elevators-opt11(20)	17	16	7	17	7	17	15	17
Floortile-opt11(20)	6	6	4	2	4	2	6	5
Freecell(80)	15	15	35	33	47	28	27	29
Grid(5)	2	2	1	2	2	2	1	2
Gripper(20)	7	6	6	20	7	20	5	20
Logistics00(28)	20	20	15	22	14	22	20	22
Logistics98(35)	6	6	2	7	3	7	6	10
Miconic(150)	141	141	50	58	57	140	140	141
Mprime(35)	22	22	18	24	20	24	20	25
Mystery(30)	19	18	15	20	16	20	16	17
Nomystery-opt11(20)	14	14	10	14	8	14	12	14
Openstacks-opt08(30	20	17	15	10	12	10	15	10
Openstacks-opt11(20)	15	12	7	5	7	5	7	5
Openstacks(30)	7	7	7	7	8	7	7	5
Parcprinter-08(30)	18	18	28	30	28	30	29	30
Parcprinter-opt11(20)	13	13	20	20	20	20	20	20

#Problems solved in 30mins

Results

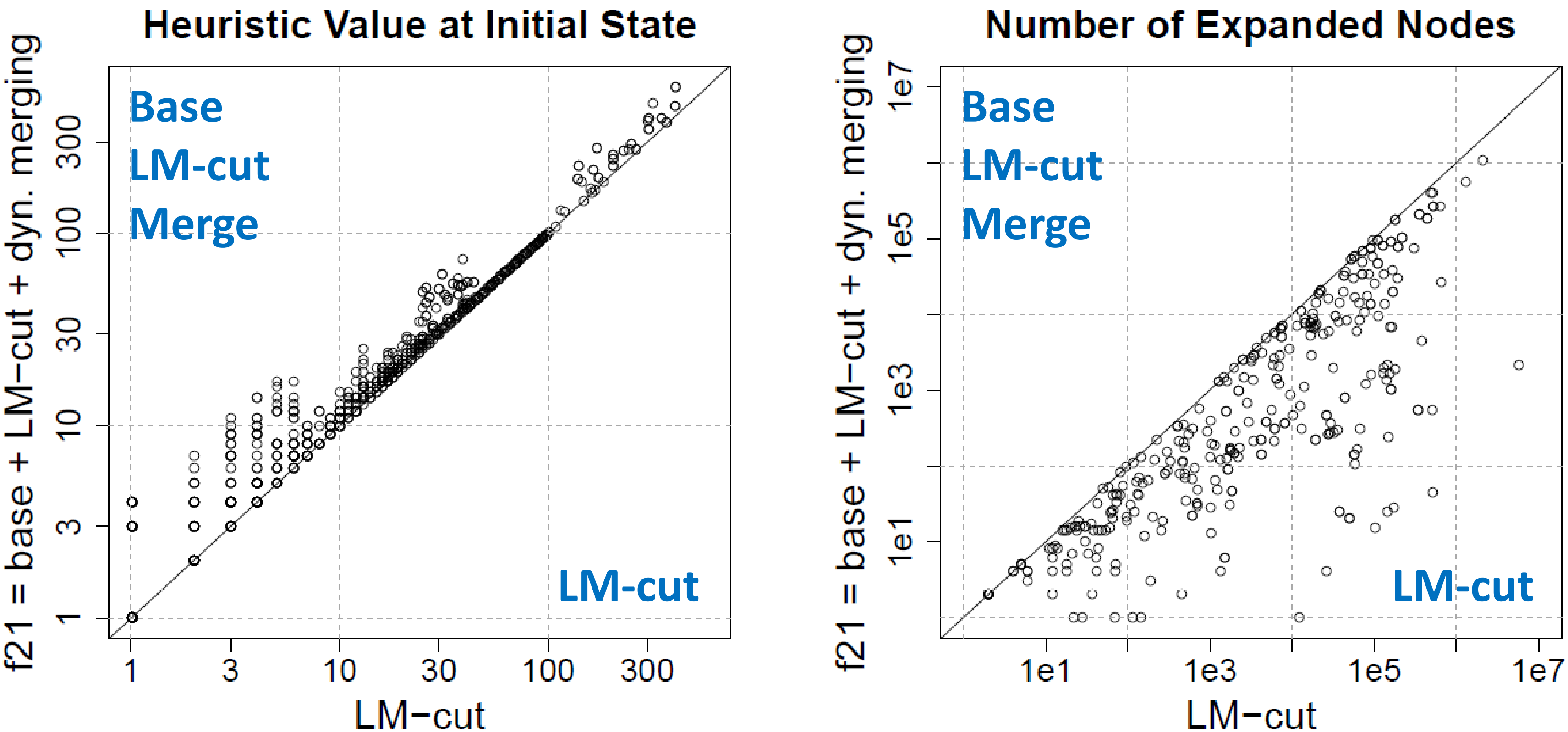
$$h_{base}^{LM-cut} \geq h_{LM-cut}^{*} \geq h_{LM-cut}$$

- Coverage results

	LM-cut			Base Merge			Base LM-cut	Base LM-cut Merge
Domain	**LM-cut**	**$h^*_{LM\text{-}cut}$**	**base**	**f 01**	**f 10**	**f 11**	**f 20**	**f 21**
Parking-opt11(20)	2	1	1	1	1	1	1	1
Pathways-noneg(30)	5	5	4	4	4	4	5	5
Pegsol-08(30)	27	27	28	28	26	26	27	27
Pegsol-opt11(30)	17	17	18	18	16	16	17	17
Pipesworld-notank(50)	17	16	15	13	16	15	11	13
Pipesworld-tank(50)	11	9	10	10	10	10	9	10
Psr-small(50)	49	49	50	50	50	50	50	50
Rovers(40)	7	7	6	6	6	7	7	7
Satellite(36)	7	7	6	6	6	6	7	7
Scanalyzer-08(30)	15	13	13	12	13	11	11	11
Scanalyzer-opt11(20)	12	10	10	9	9	8	8	8
Sokoban-opt08(30)	28	28	16	18	20	20	27	27
Sokoban-opt11(20)	20	20	15	15	16	16	20	19
Tidybot-opt11(20)	13	13	5	1	11	1	8	1
Tpp(30)	6	6	8	11	8	11	8	10
Transport-opt08(30)	11	11	10	10	10	10	11	11
Transport-opt11(20)	6	6	6	5	5	5	6	6
Trucks(30)	10	10	9	10	9	12	9	11
Visitall-opt11(20)	10	10	17	17	17	17	19	19
Woodworking-opt08(30)	16	16	12	25	14	28	20	28
Woodworking-opt11(20)	11	11	7	18	9	20	15	20
Zenotravel(20)	12	12	9	13	9	13	11	13
Total(1396)	**756**	**736**	**594**	**683**	**630**	**766**	**749**	**785**

Results

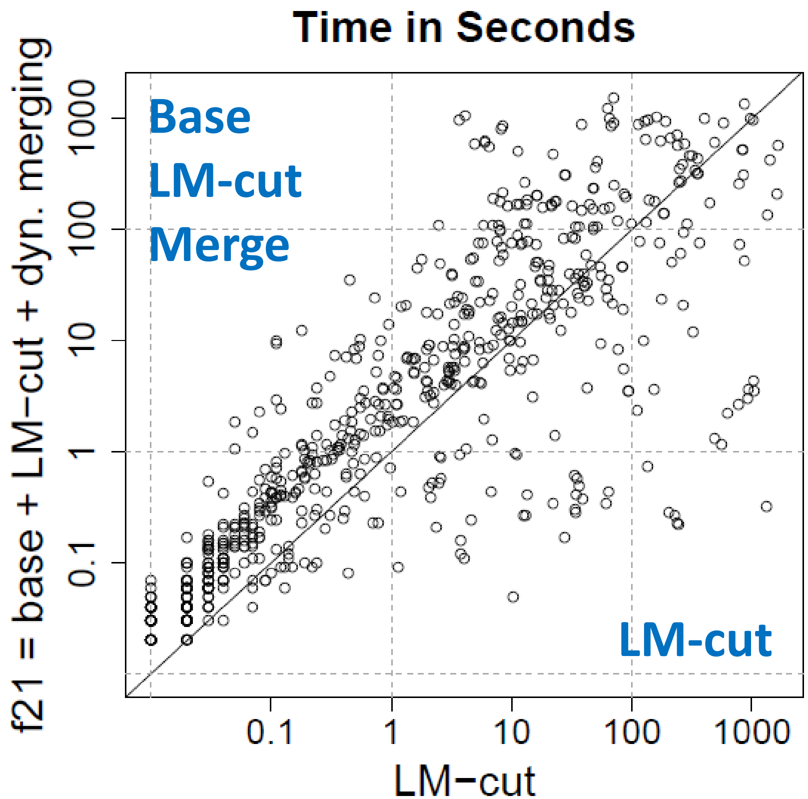
- Heuristic performance across common solved tasks
  - LM-cut versus Base model with landmarks and merges



Heuristic comparison

Results

- Time spent by A^* search for finding a solution
  - LM-cut versus Base model with LM-cut landmarks and merges



Time comparison

Conclusions

- **Flow-based heuristics** are effective and efficient yet can be improved significantly
  - Adding constraints derived from landmarks
  - Adding constraints derived from merges
- **Dynamic merging** incrementally merges more and more variables
  - So far, only incorporated a very simple merge strategy
- Flow-based heuristics provide a **rich area for future research**

Blai Bonet
Universidad Simon Bolivar
Caracas, Venezuela
bonet@ldc.usd.ve

Menkes van den Briel
NICTA & Australian National University
Canberra, Australia
menkes@nicta.com.au

USB

NICTA