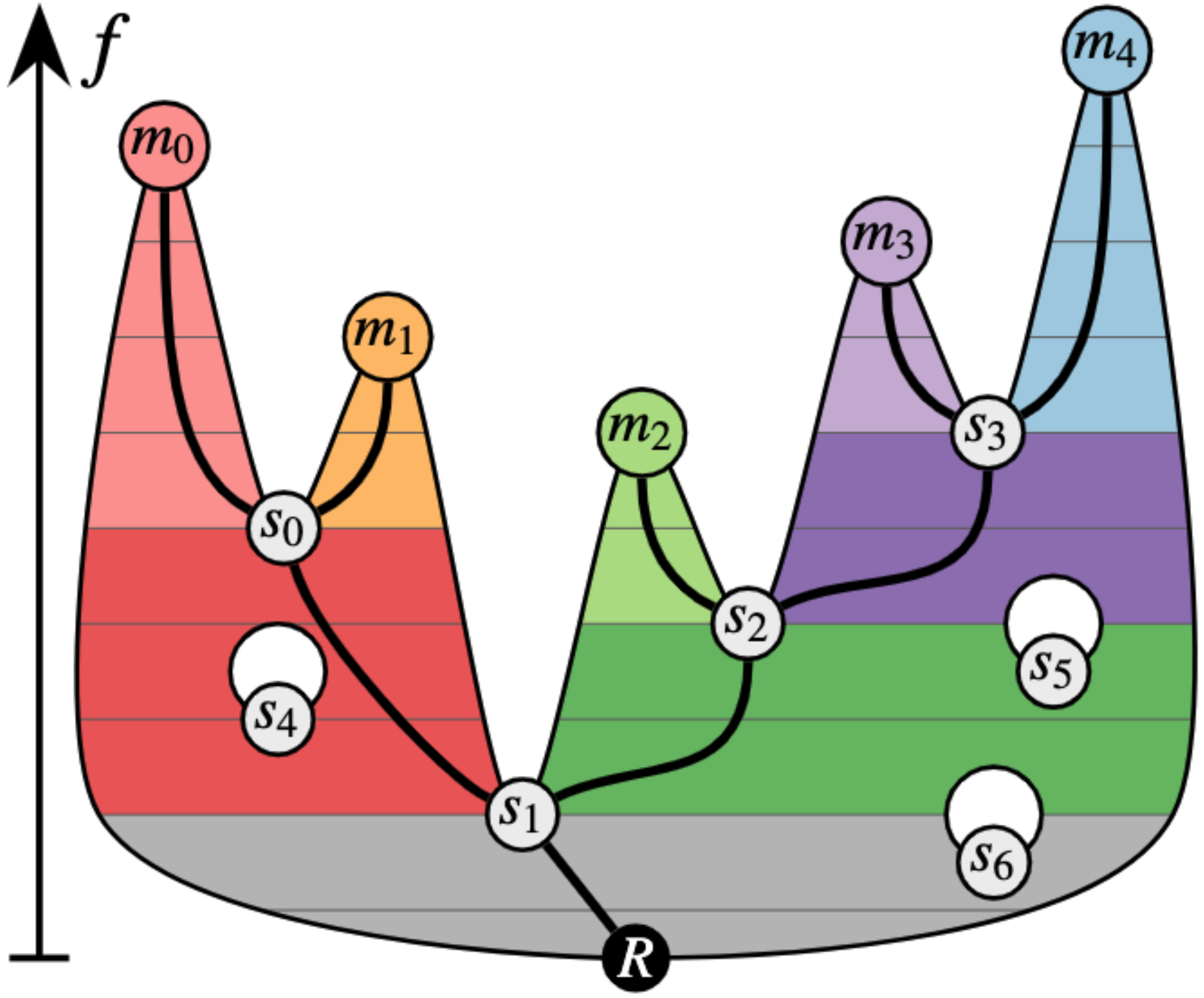
ExTreeM: Scalable Augmented Merge Tree Computation via Extremum Graphs

Jonas Lukasczyk, Michael Will, Florian Wetzels, Gunther H. Weber, and Christoph Garth

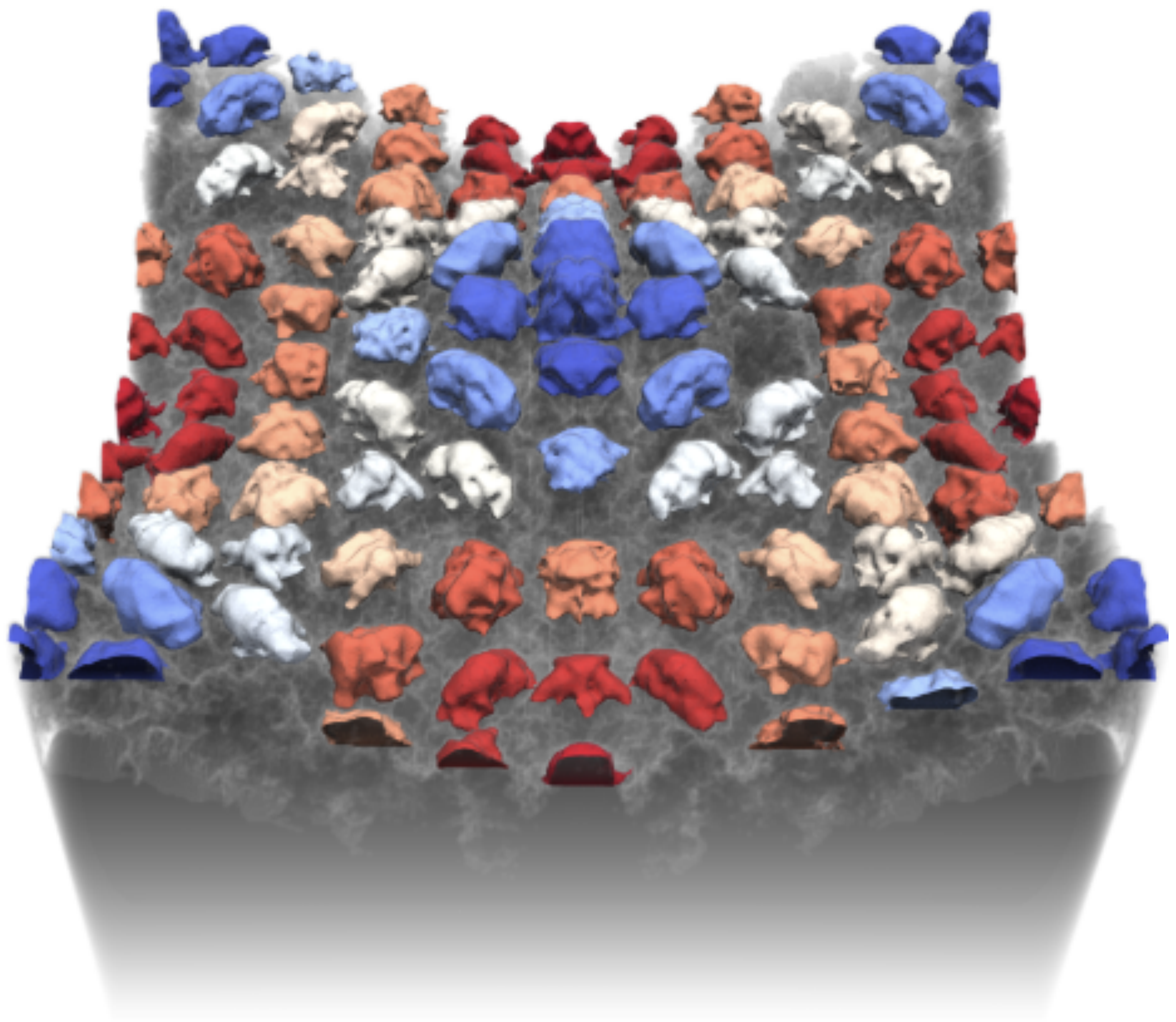
RPTU Rheinland-Pfälzische Technische Universität Kaiserslautern Landau

BERKELEY LAB

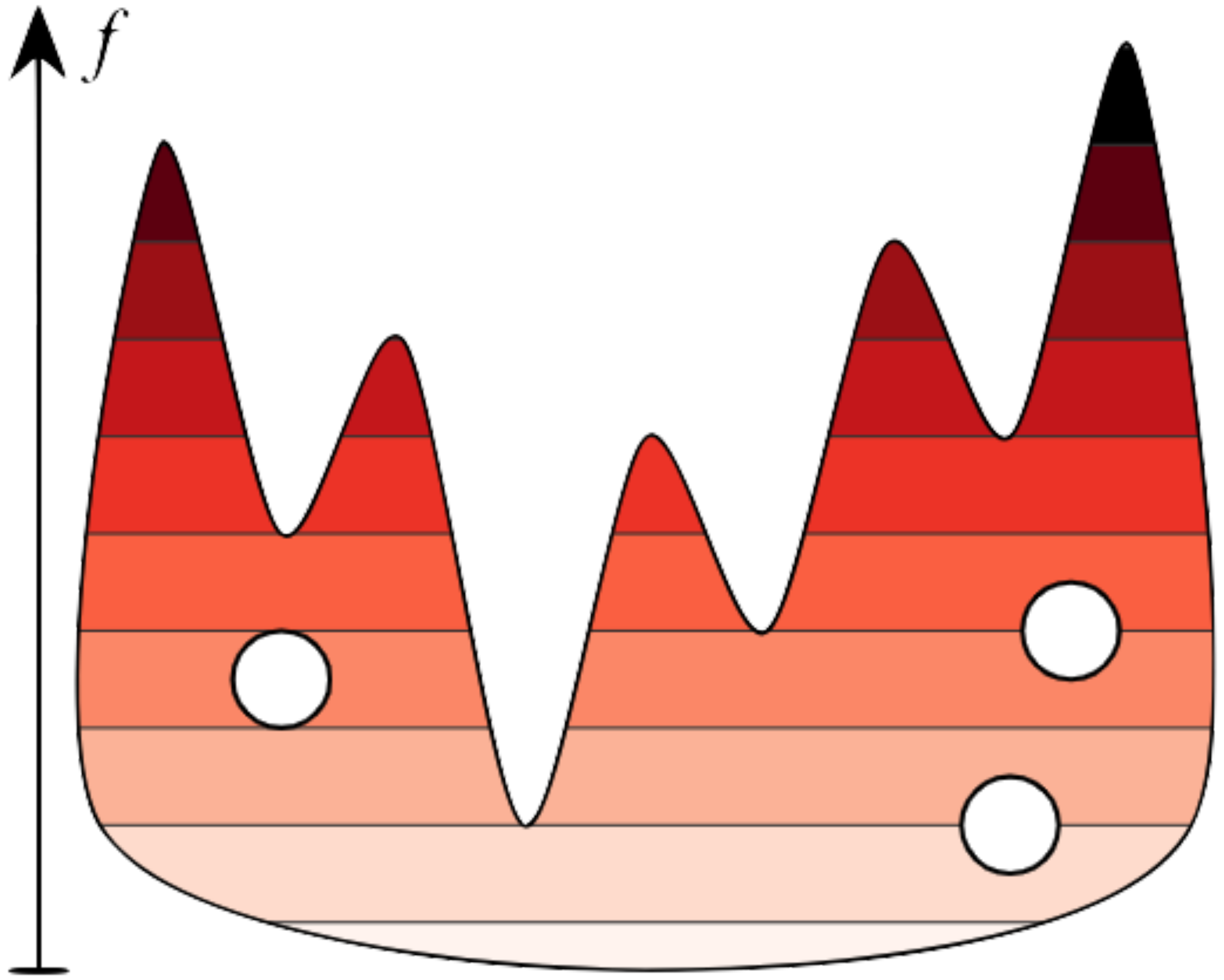
Motivation



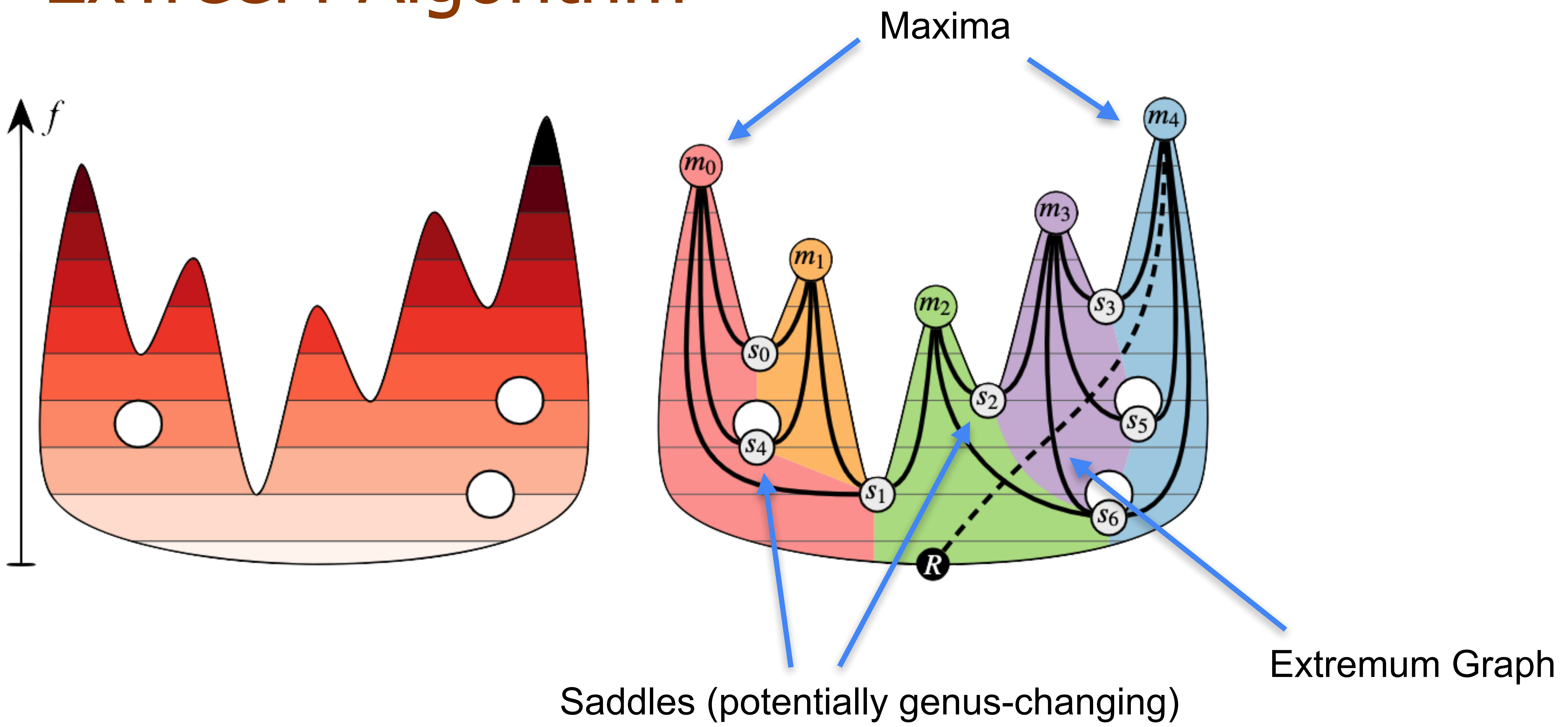
Motivation



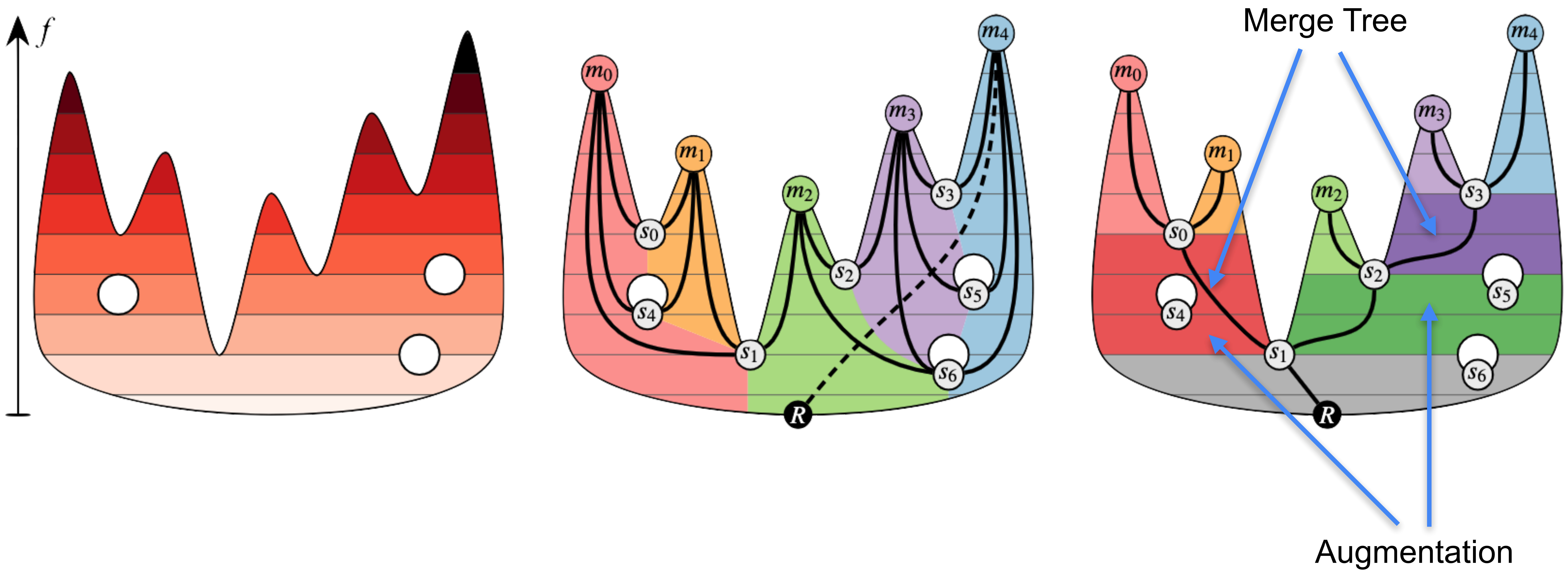
ExTreeM-Algorithm



ExTreeM-Algorithm

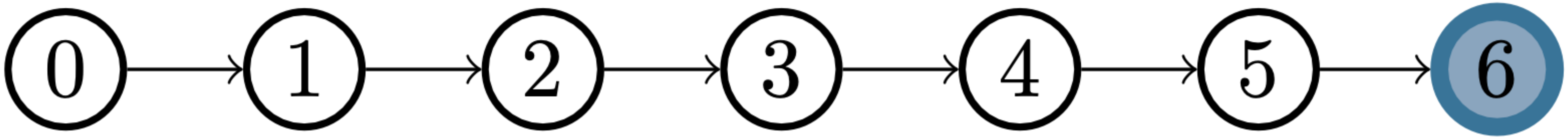


ExTreeM-Algorithm



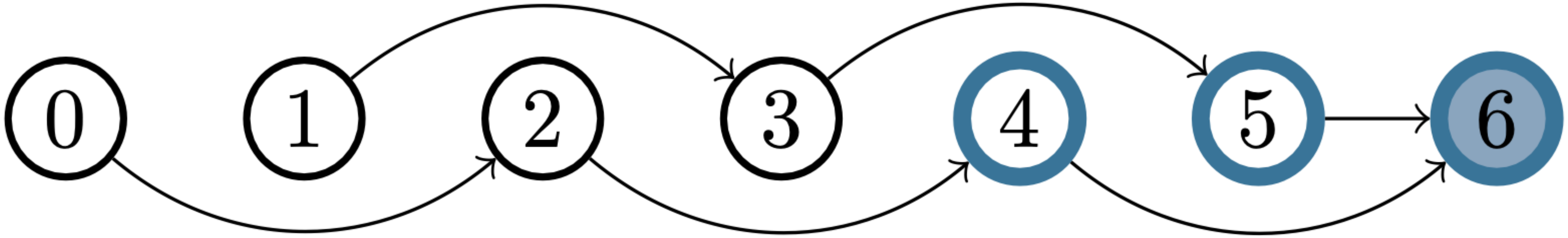
Descending Manifold

- use path compression:
  - each vertex points to largest neighbor
  - maxima point to themselves
  - in each step replace your pointer with the pointer of your pointer
  - repeat until all pointers converge to the maxima



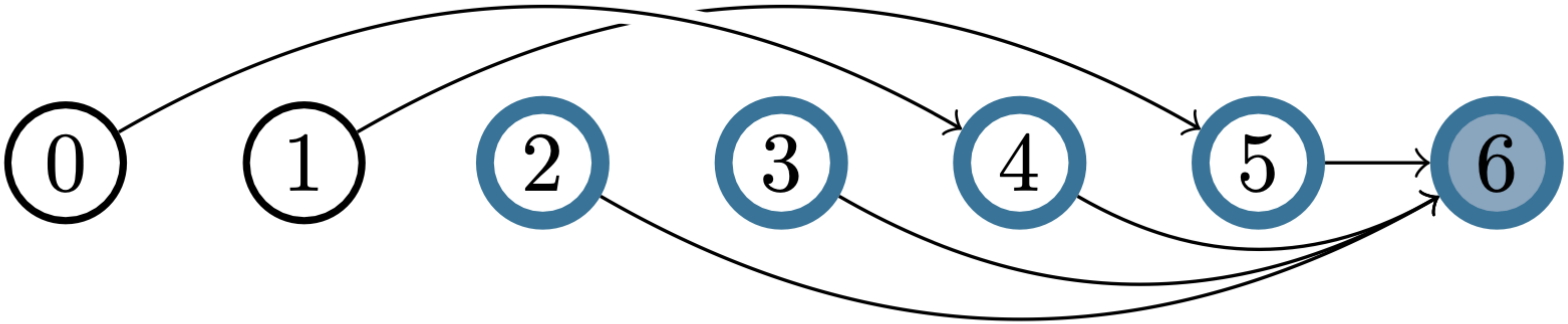
Descending Manifold

- use path compression:
  - each vertex points to largest neighbor
  - maxima point to themselves
  - in each step replace your pointer with the pointer of your pointer
  - repeat until all pointers converge to the maxima



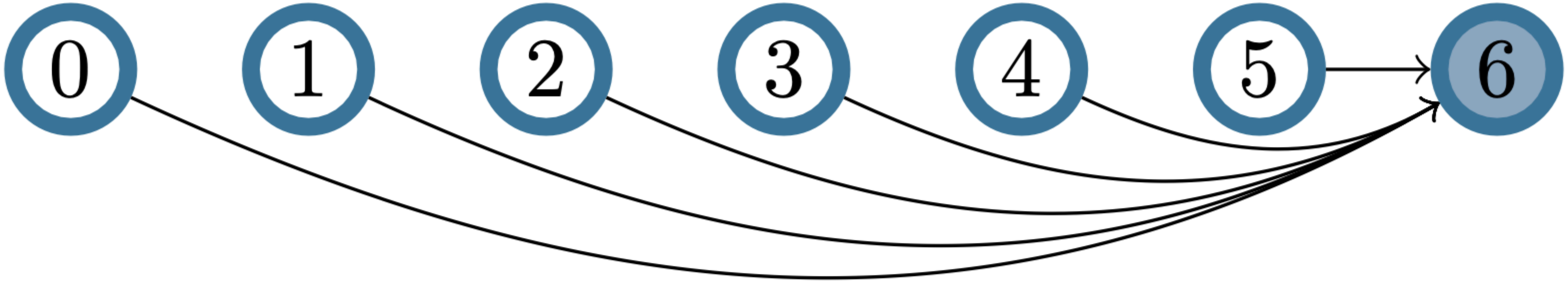
Descending Manifold

- use path compression:
  - each vertex points to largest neighbor
  - maxima point to themselves
  - in each step replace your pointer with the pointer of your pointer
  - repeat until all pointers converge to the maxima



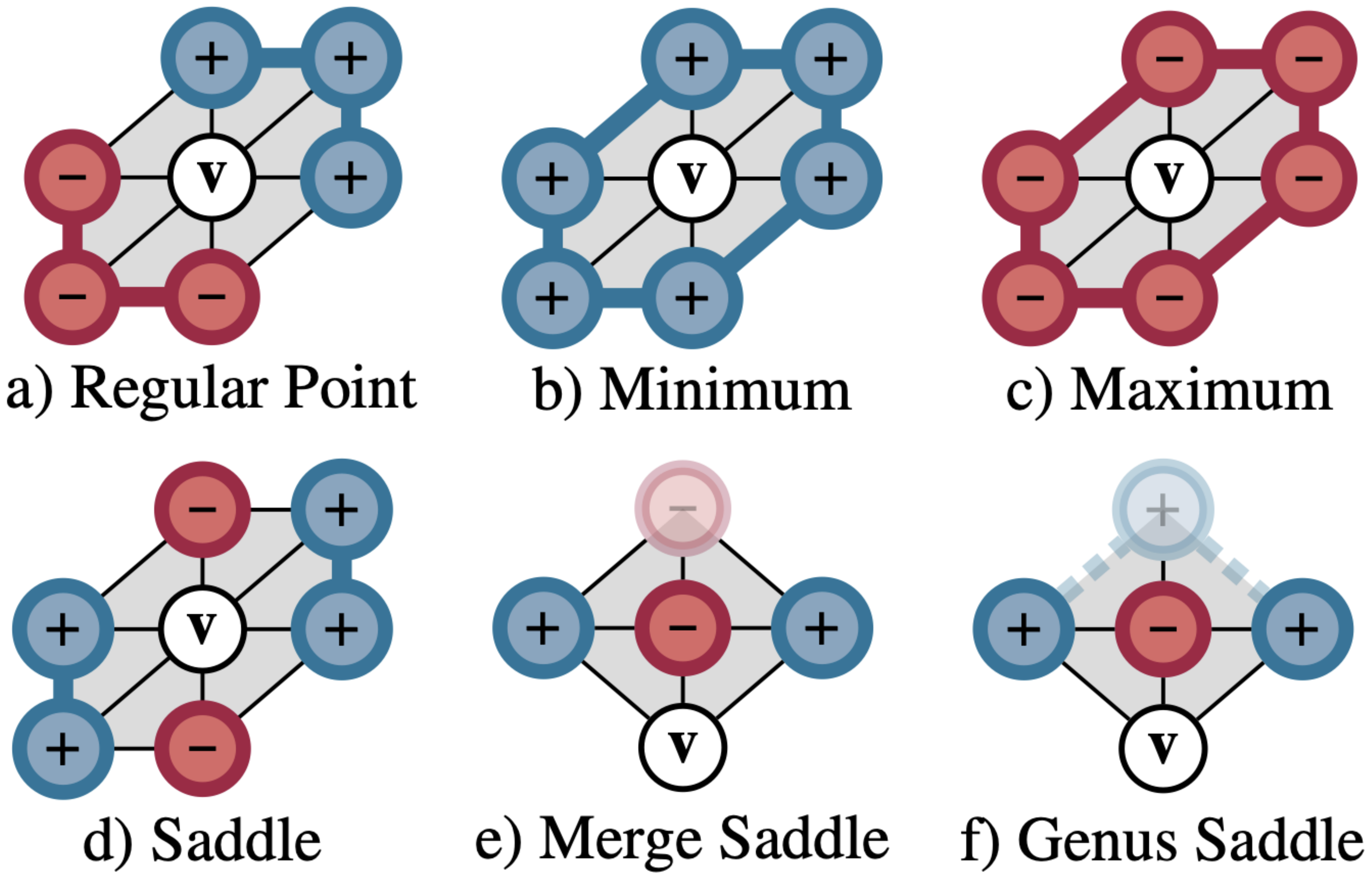
Descending Manifold

- use path compression:
  - each vertex points to largest neighbor
  - maxima point to themselves
  - in each step replace your pointer with the pointer of your pointer
  - repeat until all pointers converge to the maxima



Critical Point Computation

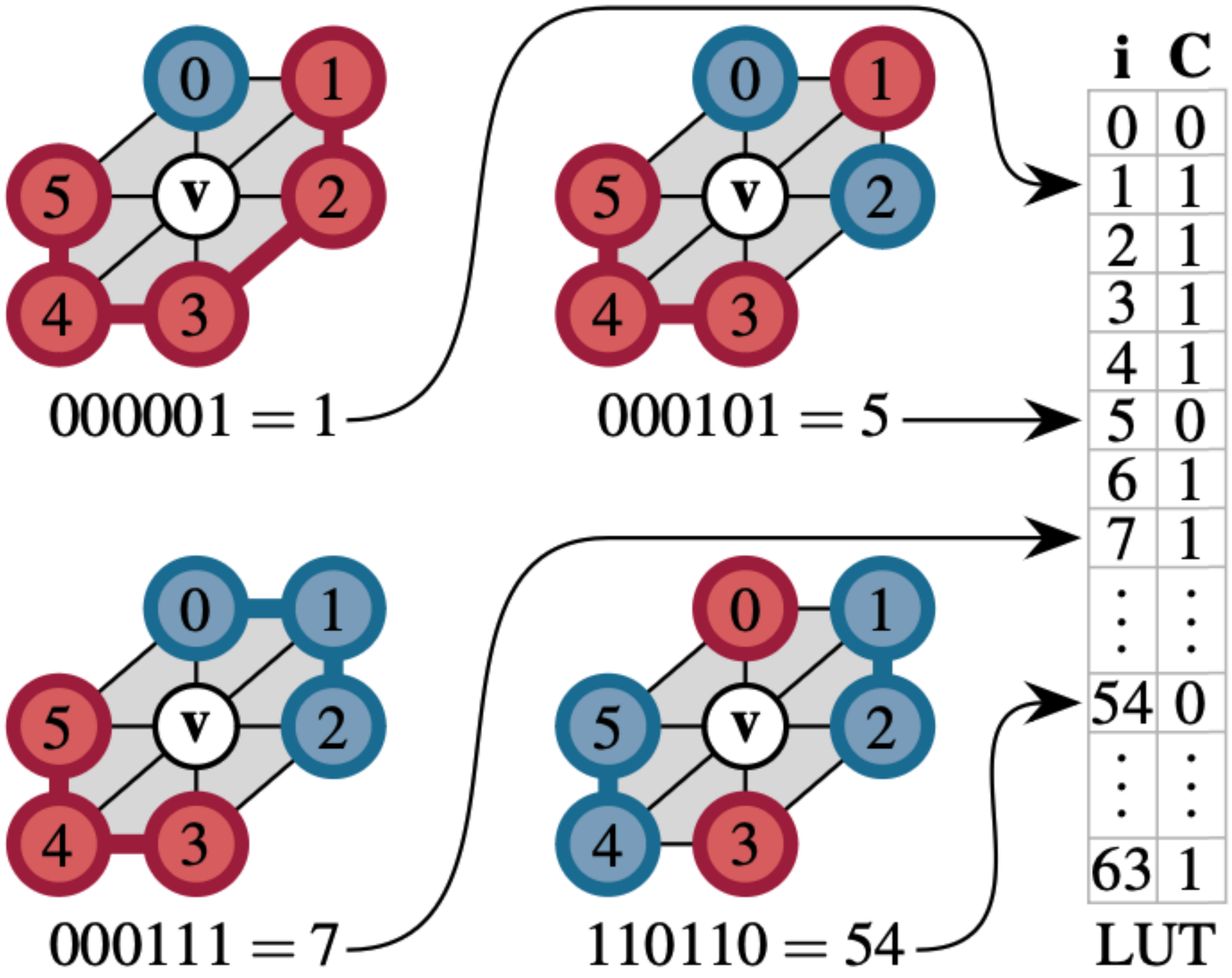
- critical points are characterized by the connectivity of their upper and lower link, for split tree upper link



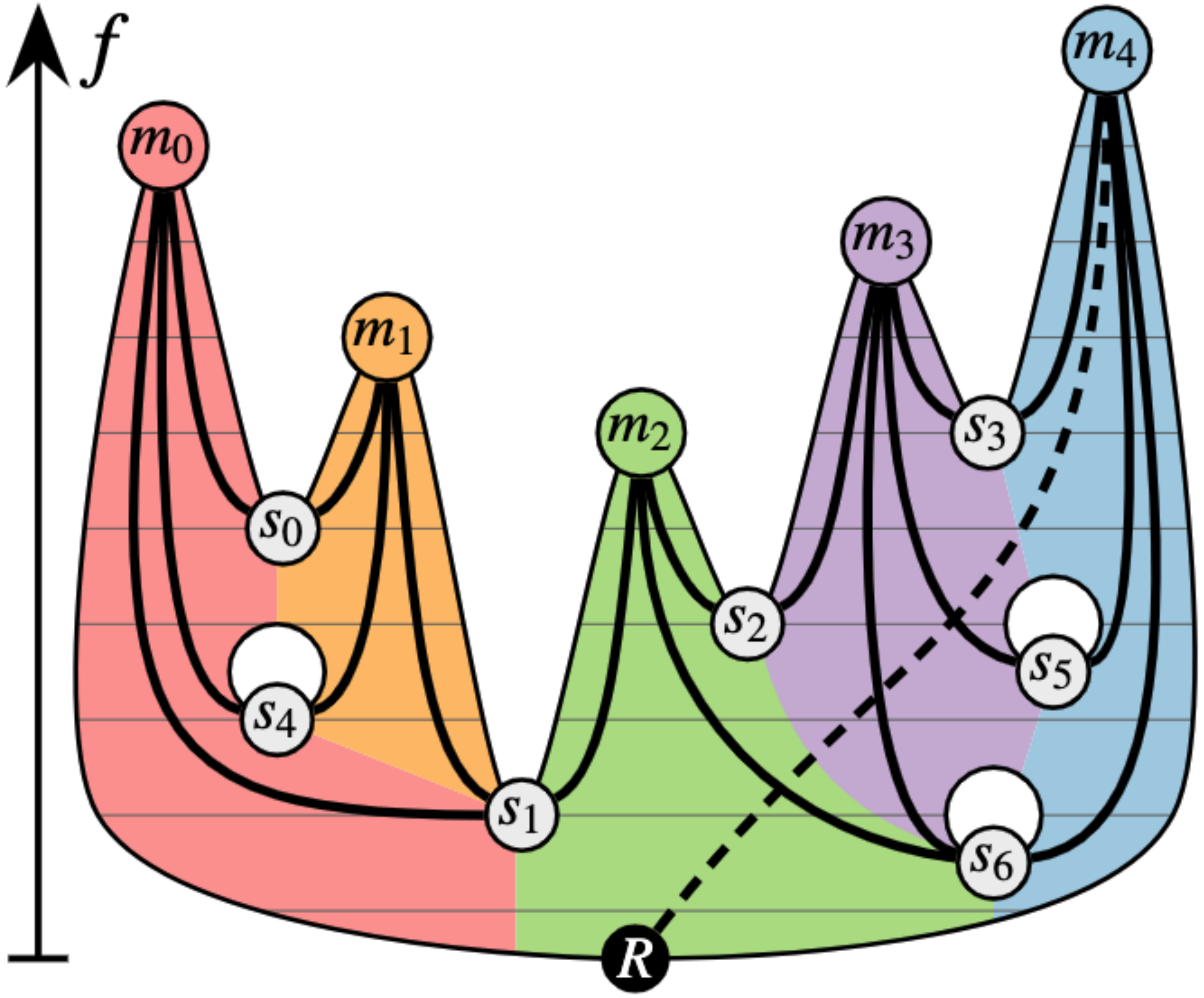
Critical Point Computation

- critical points are characterized by the connectivity of their upper and lower link, for split tree upper link
- a vertex can only be a split / merge saddle if its upper link vertices lead to at least two distinct maxima through a monotone path
- this is already recorded in the descending manifold

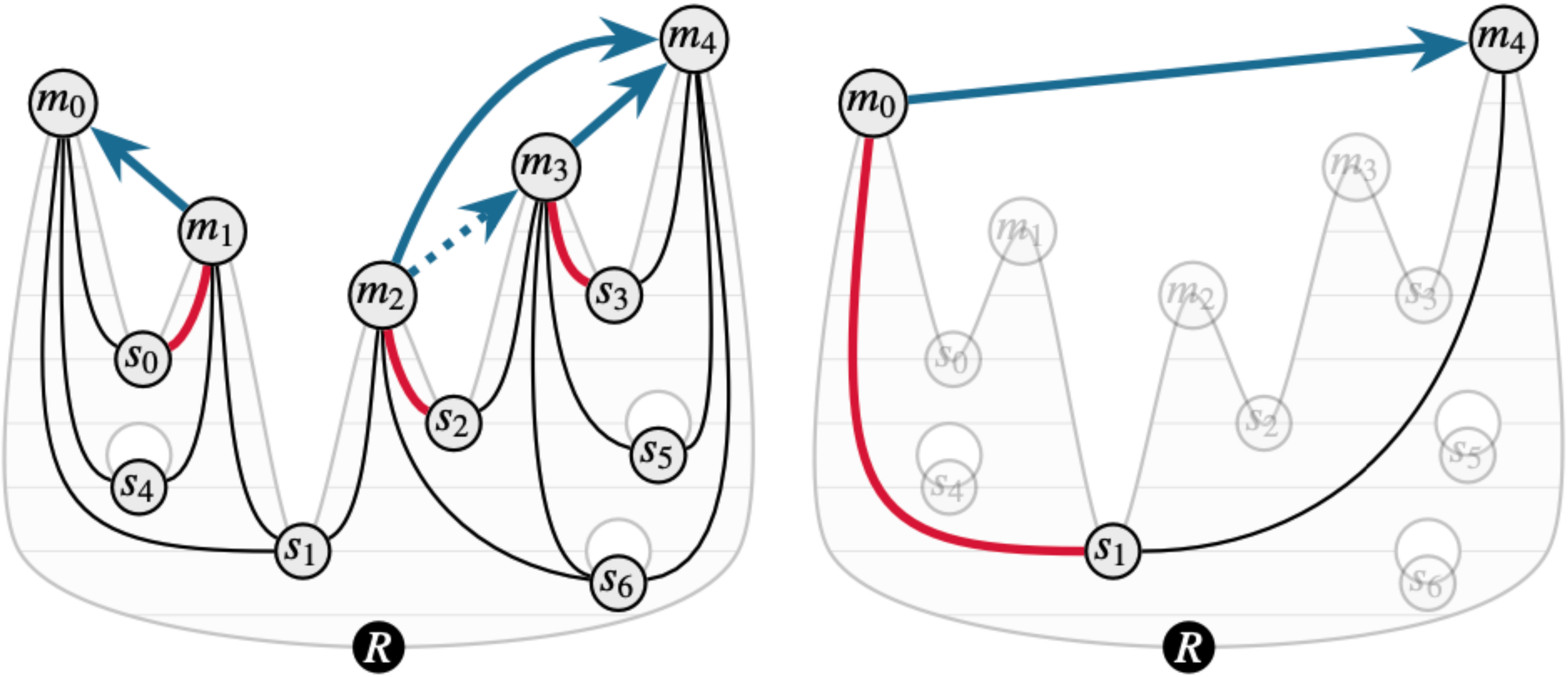
Critical Point Computation



Extremum Graph Computation

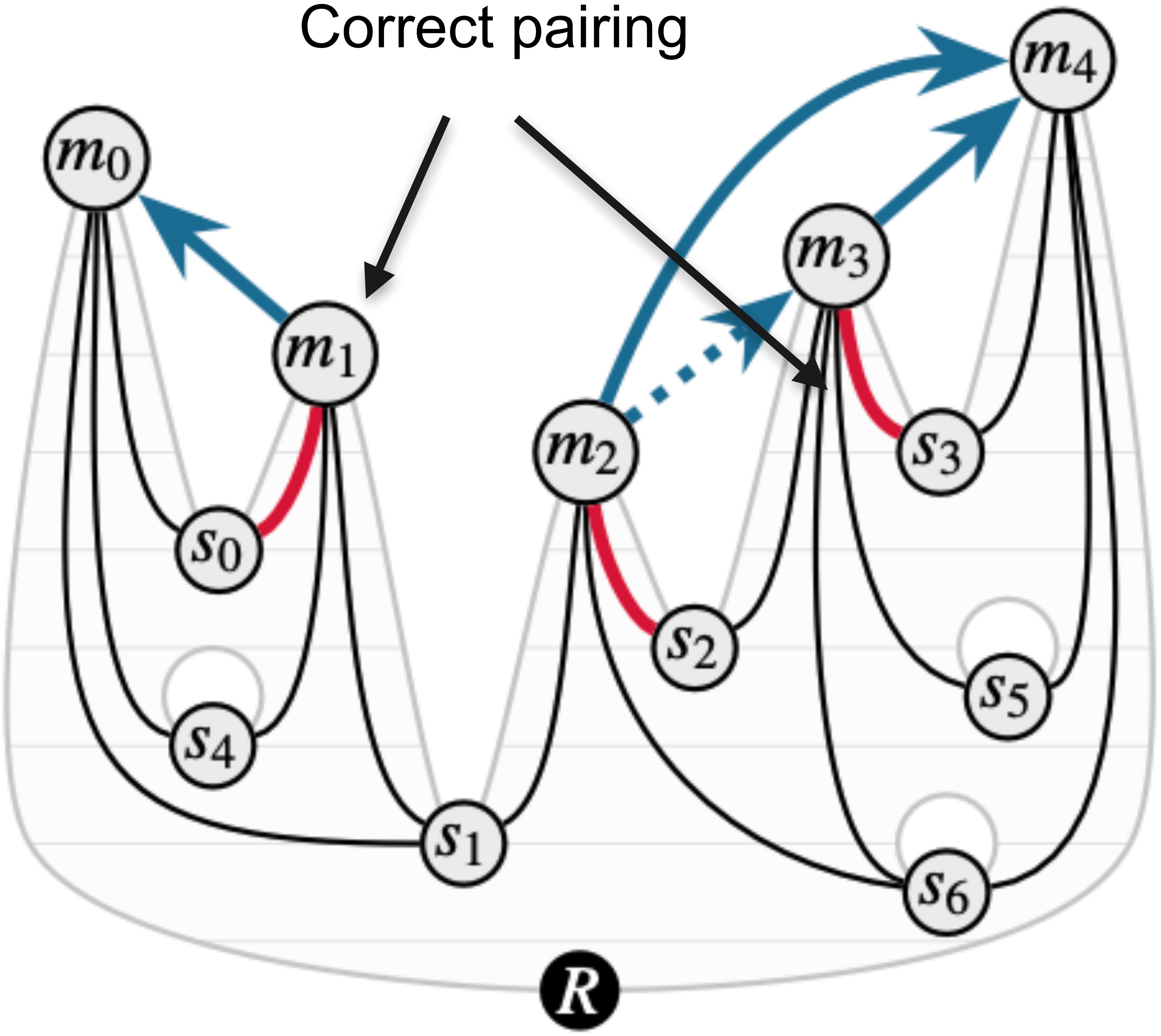


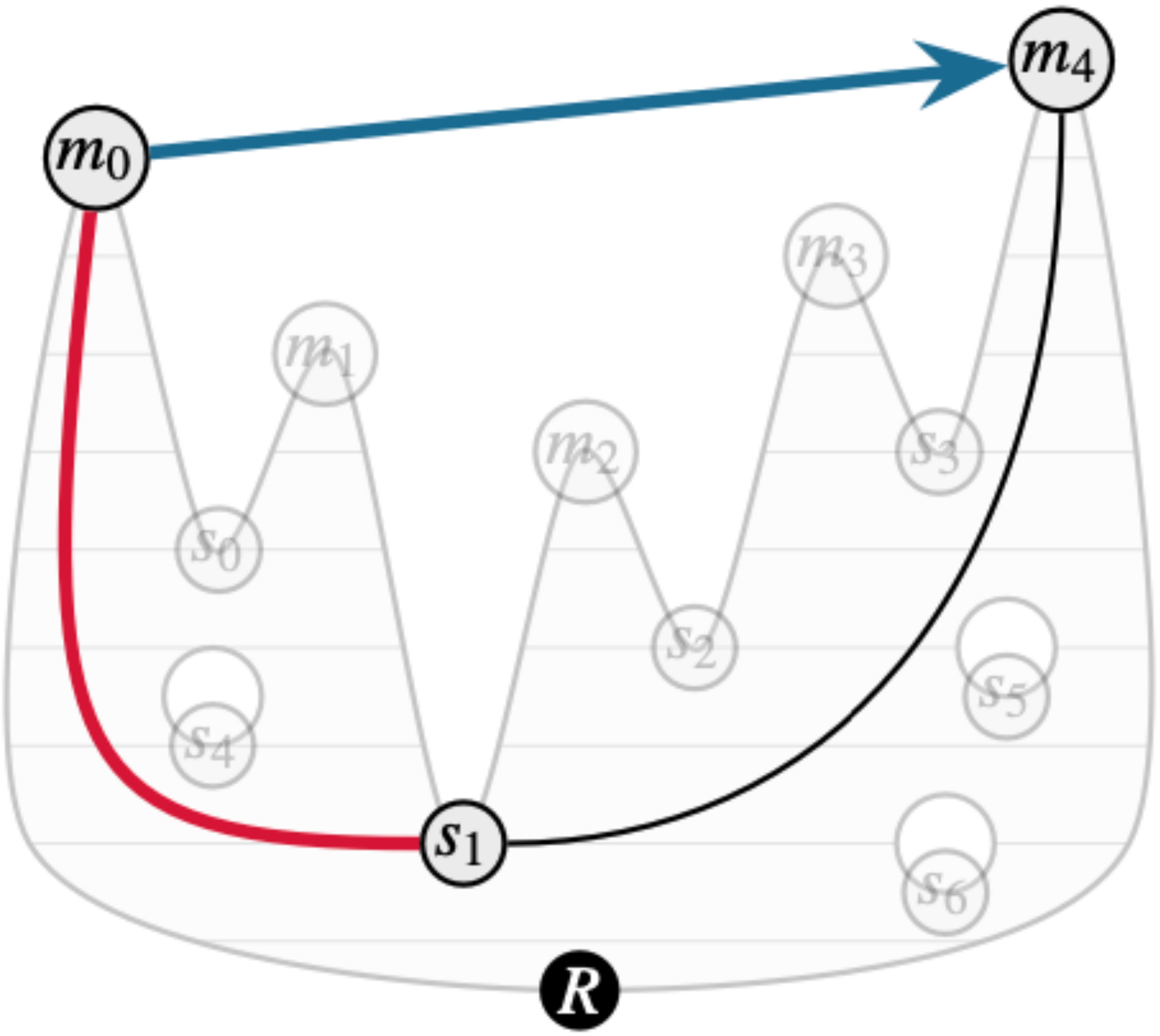
Merge Tree Computation



Merge Tree Computation

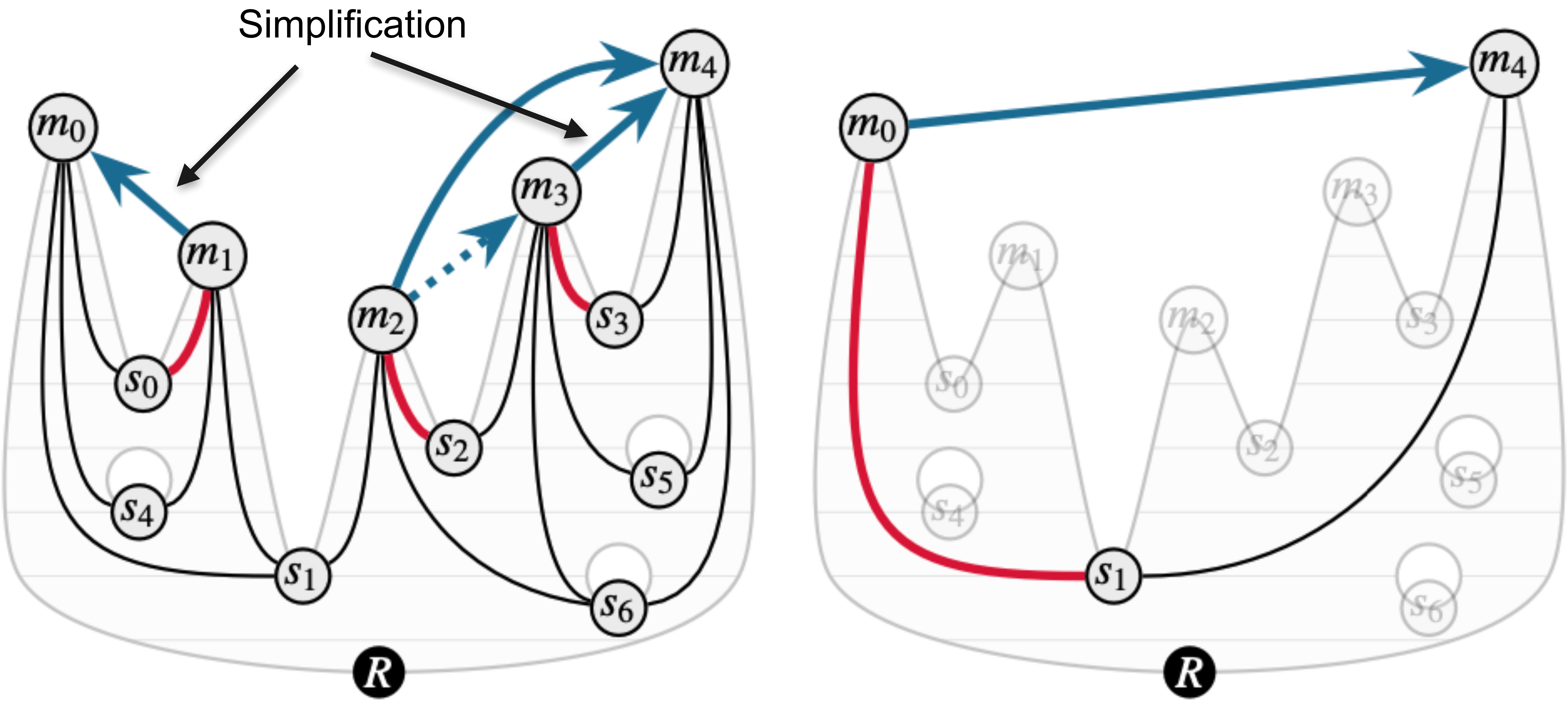
Correct pairing



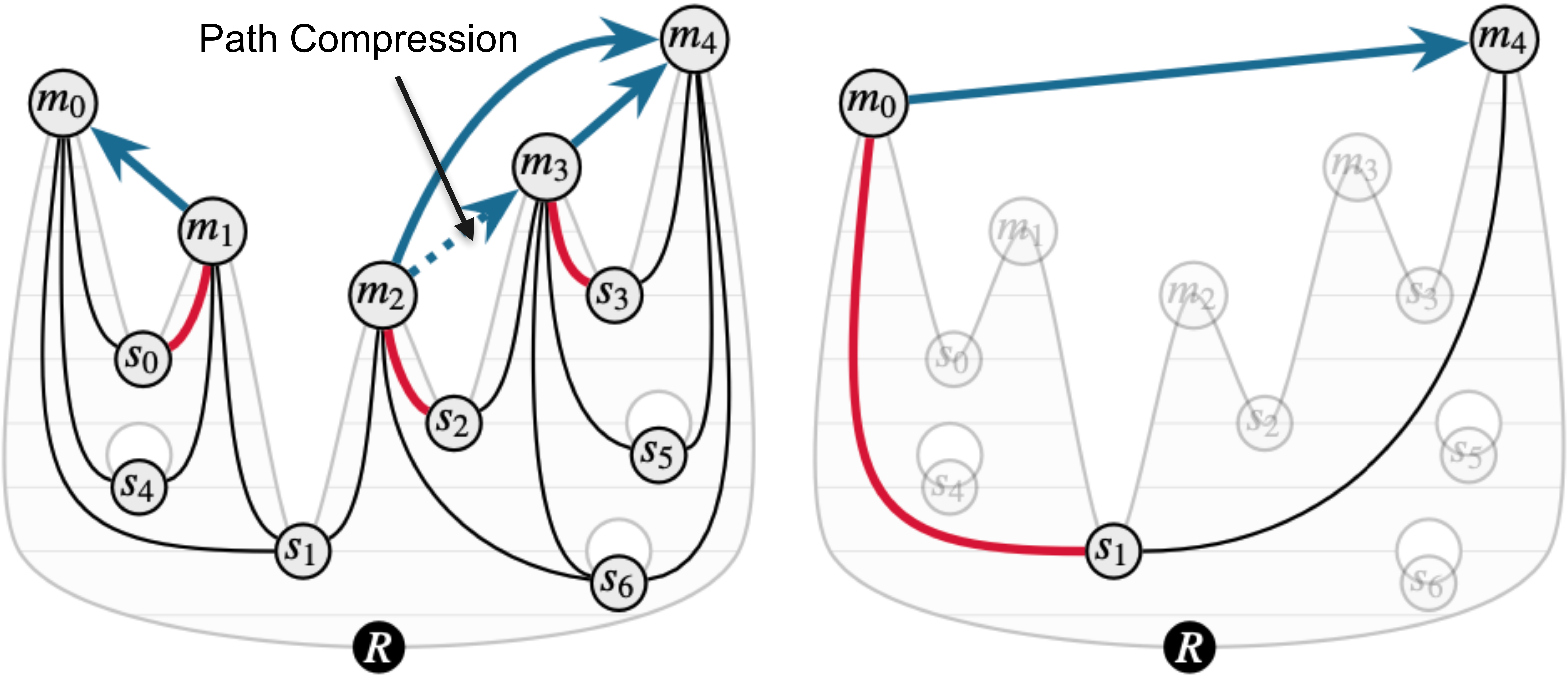


(m_i, s_i) where m_i is the smallest maximum of s_i and s_i is the largest saddle of m_i

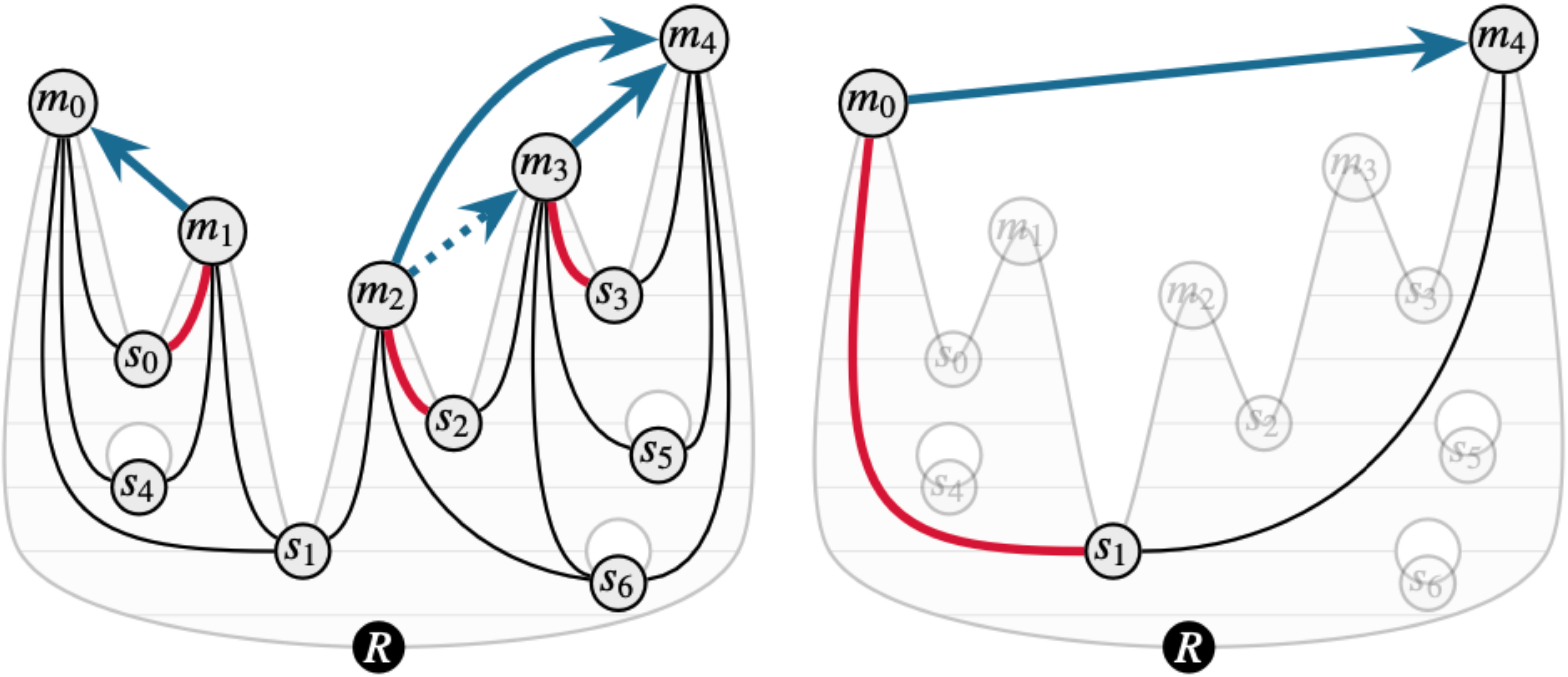
Merge Tree Computation



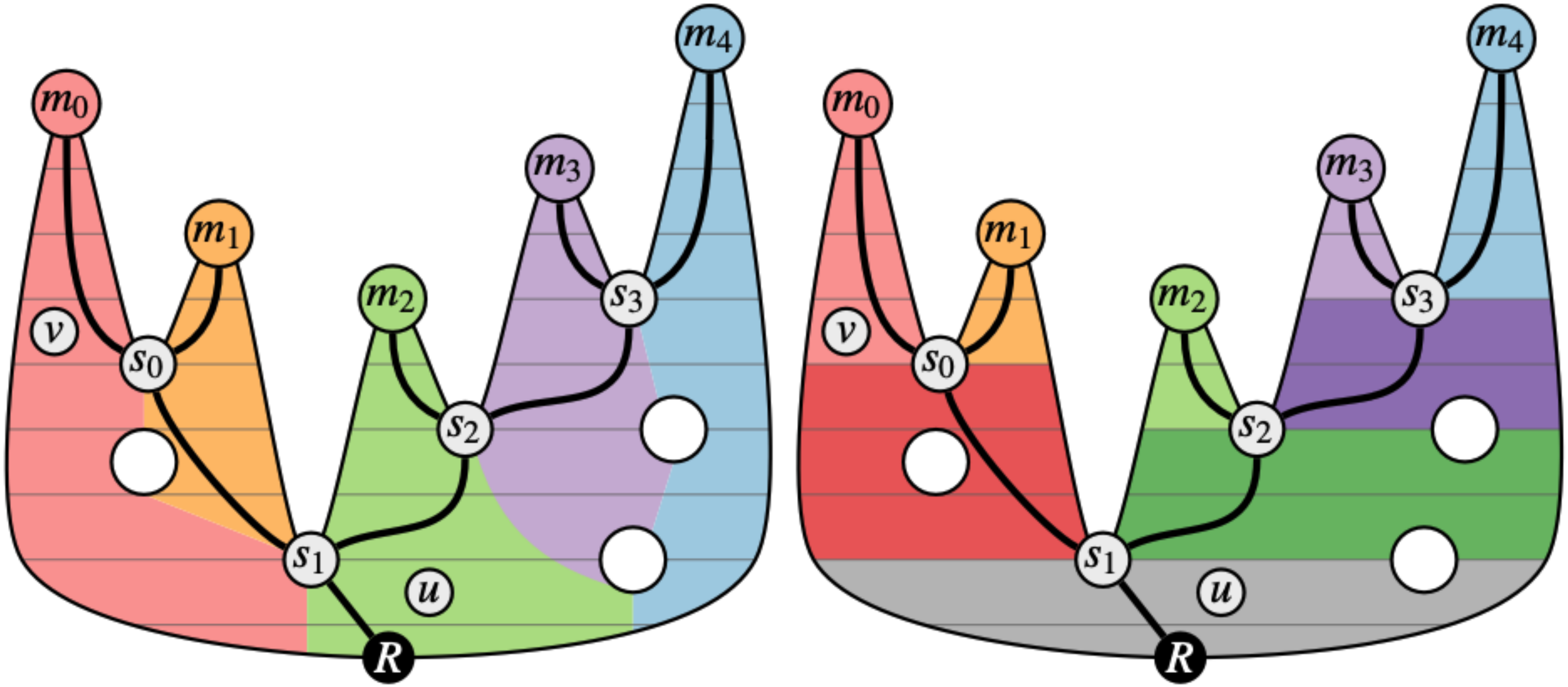
Merge Tree Computation



Merge Tree Computation



Merge Tree Augmentation



Results

Dataset	Algorithm	SU	56T	32T	16T	8T	4T	2T	1T
ctBones [41] 128^3 (3.26%)	ExTreeM	-	0.50	0.56	0.80	1.17	2.11	3.27	5.98
	PPP	**3.48**	1.74	1.90	2.50	3.96	5.74	10.86	20.36
	FTM-Tree	**14.34**	7.17	5.13	4.37	5.08	5.74	6.47	9.66
Backpack [24] $512 \times 512 \times 373$ (4.79%)	ExTreeM	-	4.42	4.67	6.81	10.58	14.01	25.25	45.23
	PPP	**3.03**	13.41	15.59	21.89	34.97	64.44	91.50	172.45
	FTM-Tree	**14.32**	63.33	45.28	40.27	44.69	41.73	51.31	79.02
Magnetic Reconnection [20] 512^3 (8.84%)	ExTreeM	-	21.58	23.92	33.15	48.13	60.79	105.52	195.81
	PPP	**2.95**	63.72	73.39	103.75	164.92	210.57	395.35	726.40
	FTM-Tree	**42.35**	914.22	787.64	865.48	830.96	820.62	924.67	931.89
Rayleigh-Taylor instability [11] 1024^3 (0.30%)	ExTreeM	-	14.21	21.50	39.16	72.56	137.31	268.47	503.67
	PPP	**8.01**	113.76	157.82	262.92	482.16	738.87	1513.28	3009.64
	FTM-Tree	**10.41**	147.88	147.78	167.88	207.60	376.77	725.51	1284.83
Neurons in Marmoset [16] $1024 \times 1024 \times 314$ (15.21%)	ExTreeM	-	32.34	37.39	48.04	67.86	95.01	149.20	262.18
	PPP	**2.05**	66.18	76.14	105.82	157.29	280.86	445.43	887.27
Kingsnake [36] $1024 \times 1024 \times 795$ (4.71%)	ExTreeM	-	36.34	43.99	59.17	93.39	128.17	235.03	434.40
	PPP	**2.79**	97.97	118.47	170.53	272.38	445.99	823.12	1641.87
Jet in Crossflow [18] $1408 \times 1080 \times 1100$ (0.03%)	ExTreeM	-	14.26	21.39	37.51	71.55	134.80	270.67	521.87
	PPP	11.20	159.71	226.46	386.06	718.04	1223.16	2386.21	4609.00
Richtmyer-Meshkov instability [10] $1536 \times 1536 \times 1408$ (0.31%)	ExTreeM	-	39.15	56.85	98.95	182.57	287.42	502.10	844.03
	PPP	**5.01**	196.00	258.66	422.94	756.77	1316.27	2110.14	4163.87
Unstructured Richtmyer-Meshkov [10] $\sim 7\times10^6$ vertices, $\sim 42\times10^6$ edges (12.82%)	ExTreeM	-	1.48	1.76	2.57	4.12	6.97	11.60	21.68
	FTM-Tree	**67.85**	100.42	93.76	87.38	92.12	94.69	94.48	100.61
Perlin Noise 256^3 (23.28%)	ExTreeM	-	2.24	2.60	3.41	4.58	5.58	9.67	16.60
	PPP	**1.83**	4.10	4.43	5.92	9.17	18.48	22.65	39.29
	FTM-Tree	**139.85**	313.26	281.10	273.31	207.78	272.25	204.69	273.59
Perlin Noise 1024^3 (23.20%)	ExTreeM	-	148.81	181.72	243.14	379.63	490.85	897.46	1811.55
	PPP	**1.41**	209.42	251.81	373.76	604.46	892.84	1766.66	3285.86

Dataset	Algorithm	SU	56T
Jet in Crossflow [18] $1408 \times 1080 \times 1100$ (0.03%)	ExTreeM	-	14.26
	PPP	**11.20**	159.71
Unstructured Richtmyer-Meshkov [10] $\sim 7\times10^6$ vertices, $\sim 42\times10^6$ edges (12.82%)	ExTreeM	-	1.48
	FTM-Tree	**67.85**	100.42
Perlin Noise 1024^3 (23.20%)	ExTreeM	-	148.81
	PPP	**1.41**	209.42

Conclusion

Contributions:

- generic merge tree computation scheme
- novel merge tree algorithm with high parallel efficiency and low memory footprint
- open-source implementation in the Topology ToolKit

Future Work:

- adapting the generic concept with other merge tree algorithms e.g. Triplet Merge Tree or Parallel Peak Pruning
- GPU and distributed versions of ExTreeM

References

- R. G. Maack, J. Lukasczyk, J. Tierny, H. Hagen, R. Maciejewski, and C. Garth. Parallel Computation of Piecewise Linear Morse-Smale Seg- mentations. *IEEE Transactions on Visualization and Computer Graphics*, 2023.
- C. Gueunet, P. Fortin, J. Jomier, and J. Tierny. Task-Based Augmented Merge Trees with Fibonacci Heaps. *IEEE Symposium on Large Data Analysis and Visualization*, 2017.
- H. A. Carr, O. Rübel, G. H. Weber, and J. P. Ahrens. Optimization and Augmentation for Data Parallel Contour Trees. *IEEE Transactions on Visualization and Computer Graphics*, 2021

Thanks to:

- Exascale Computing Project (17- SC-20-SC) for support

- You for listening