Ant Colony Optimization for Path Planning in Search and Rescue Operations

Michael Morin*, Irène Abi-Zeid*, Claude-Guy Quimper**
* Department of Operations and Decision Systems
** Department of Computer Science and Software Engineering
Université Laval, Québec, Canada

UNIVERSITÉ LAVAL
Faculté des sciences de l'administration

INFORMS 2024, Seattle, WA, USA

European Journal of Operational Research 305 (2023) 53–63

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Discrete Optimization

Ant colony optimization for path planning in search and rescue operations

Michael Morin[a,*], Irène Abi-Zeid[a], Claude-Guy Quimper[b]

[a] *Department of Operations and Decision Systems, Université Laval, Québec, Canada*
[b] *Department of Computer Science and Software Engineering, Université Laval, Québec, Canada*

Morin, M., I. Abi-Zeid, C.-G. Quimper (2023).
Ant Colony Optimization for Path Planning in Search and Rescue Operations.
In: European Journal of Operational Research 305(1), 53–63.
https://doi.org/10.1016/j.ejor.2022.06.019

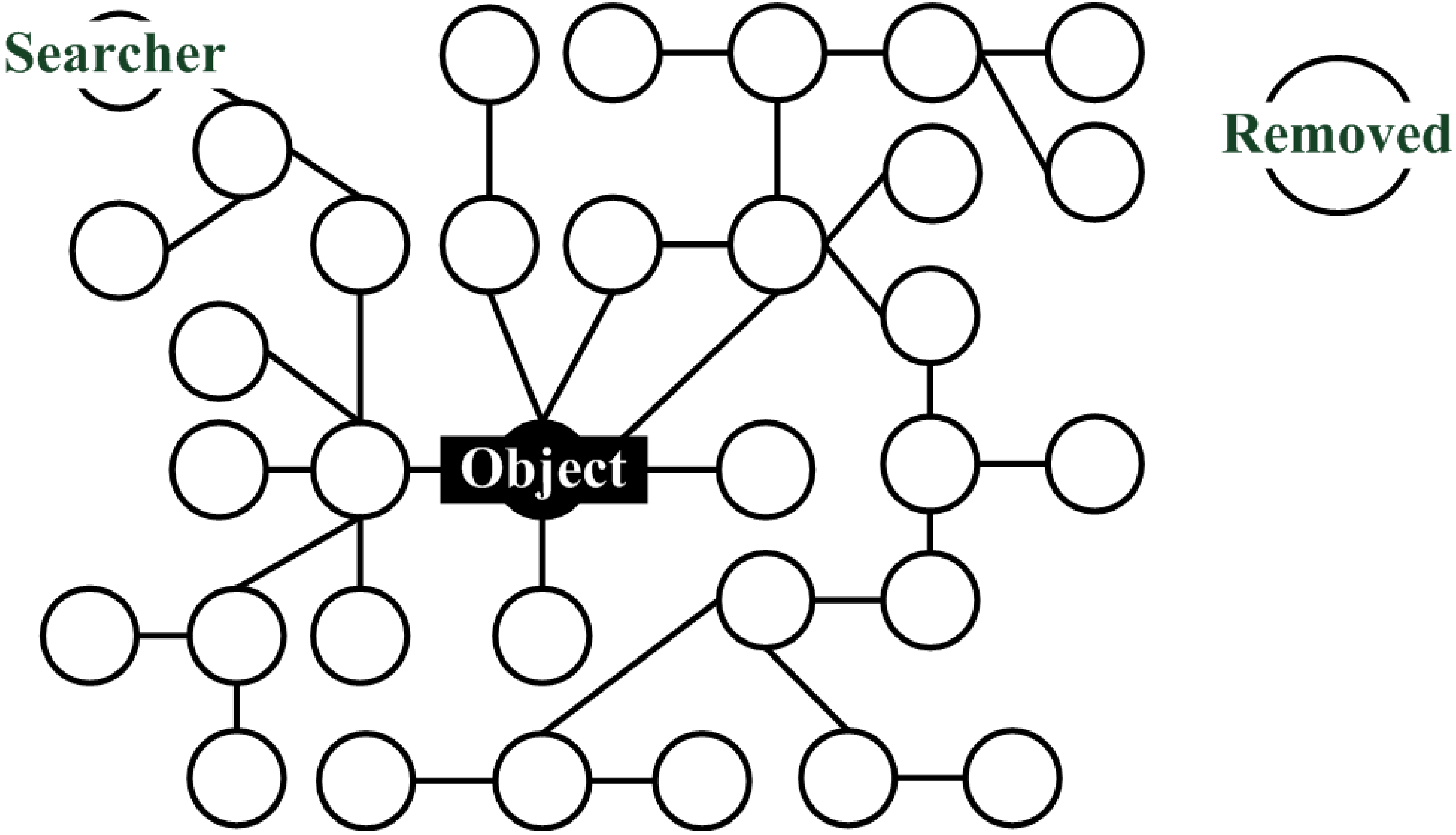
Search theory

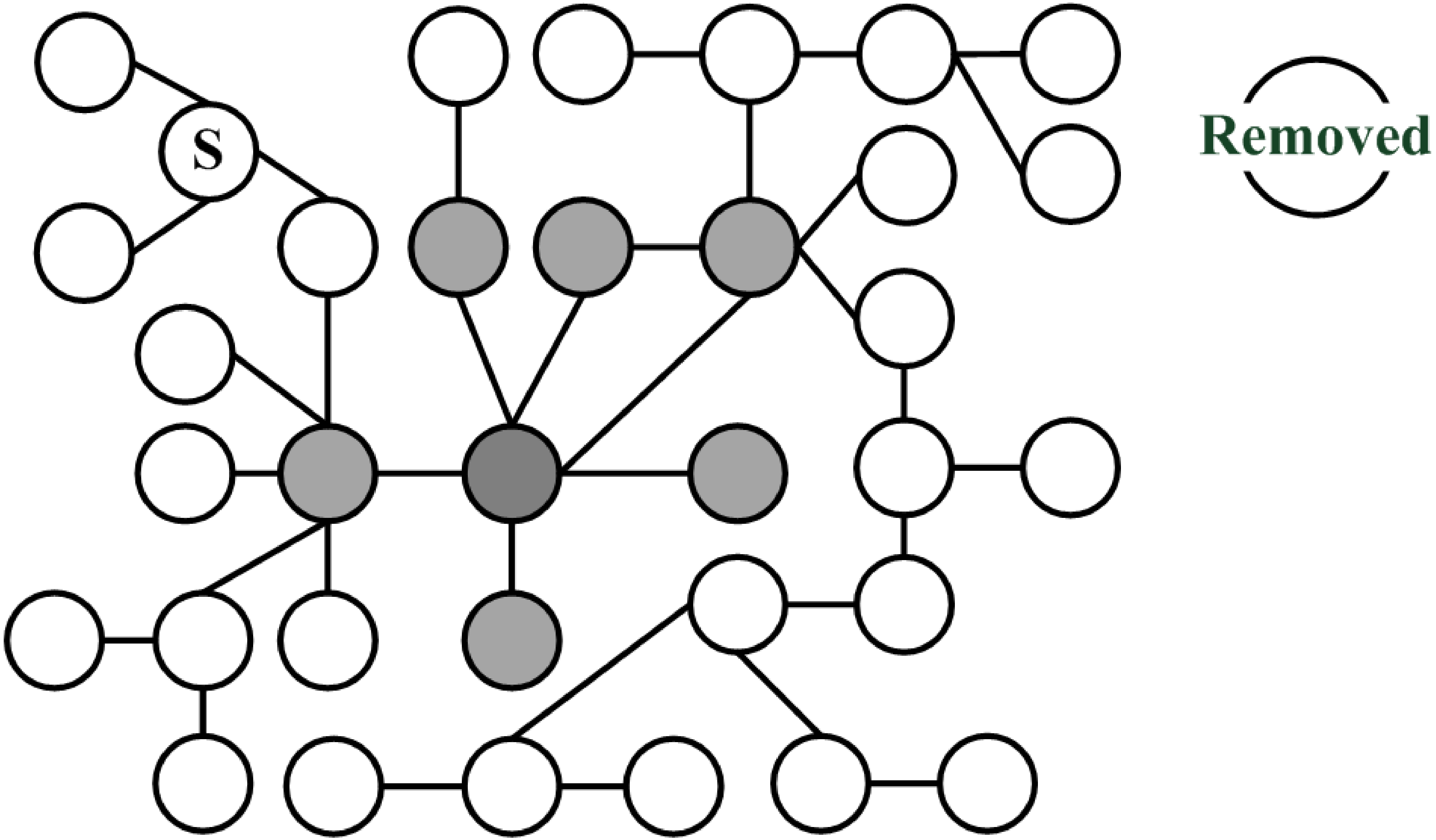
- Also called the Theory of Optimal Search and Screening (first developed by B.O. Koopman in 1946)
- To enhance U-boat search methodologies during the Battle of the Atlantic (1939-1945)
- Main problem classes:
  - Optimal Search Density problems
  - Optimal Searcher Path problems

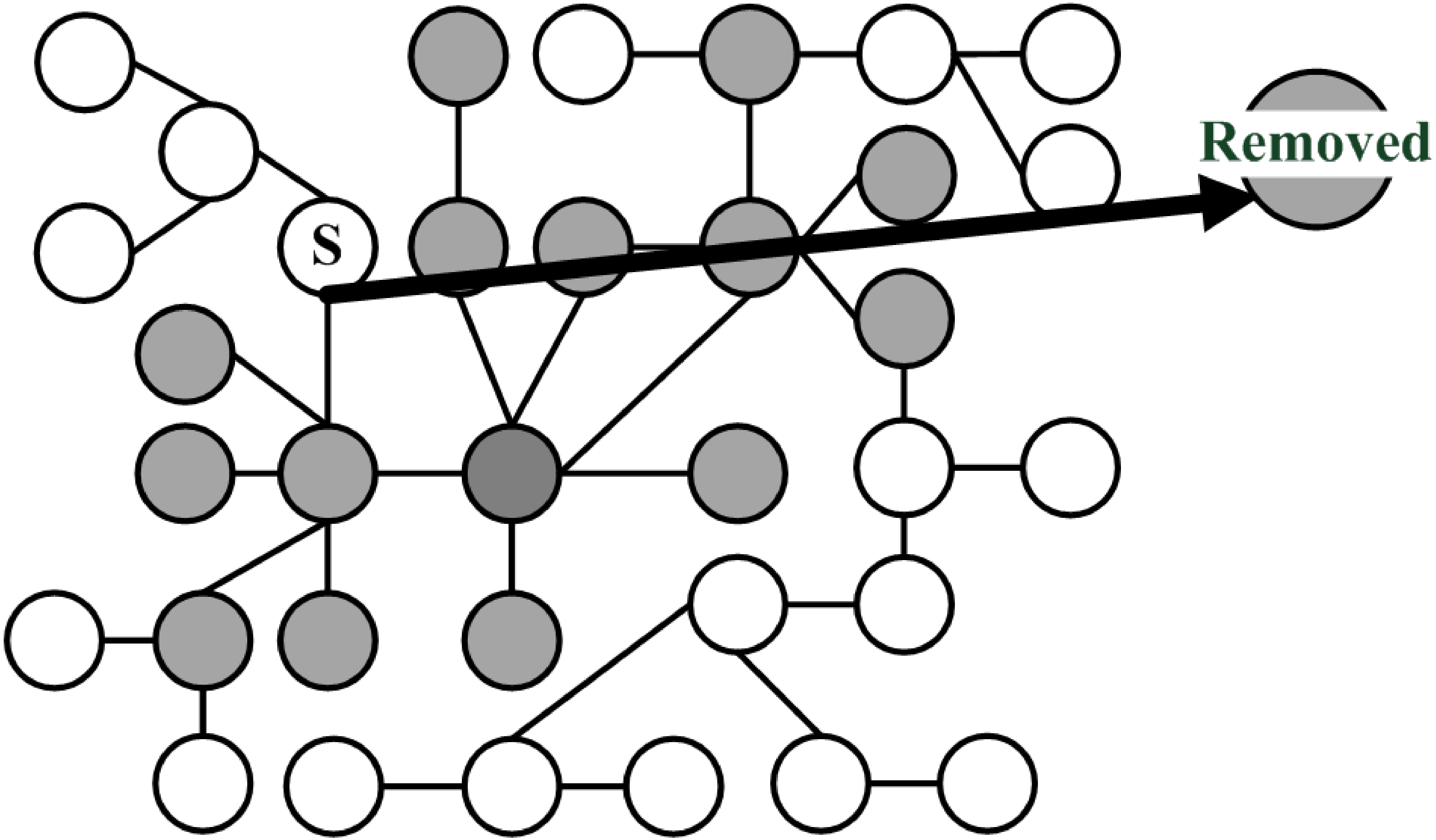
Path planning in search theory

- Optimal searcher path (OSP) problem
  - Operational-level path planning in graph/grid *search environment*
  - Imperfect detection ⇒ *Probability of Detection*
  - Moving search object ⇒ *Probability of Containment* (whereabouts)
    Motion model
  - Maximize the Cumulative Overall p*robability of Success* (*COS*)

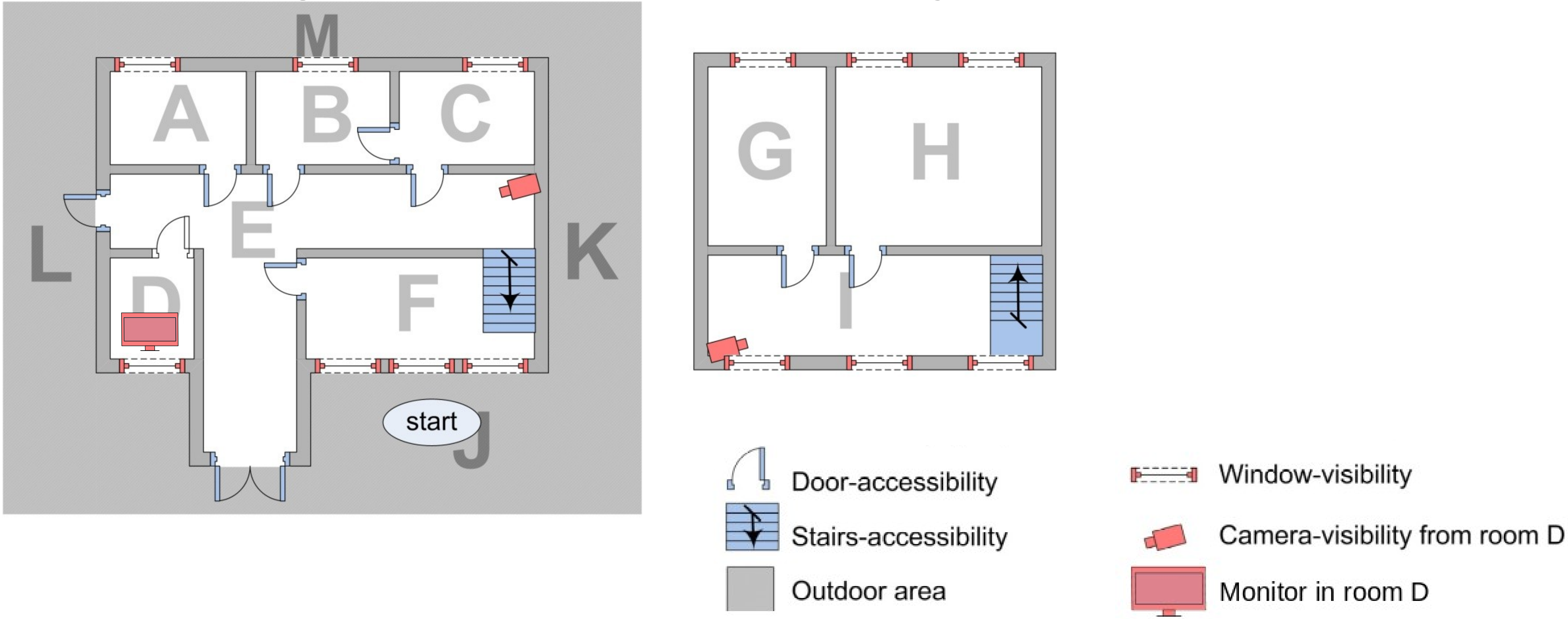
Cumulative in time Across space

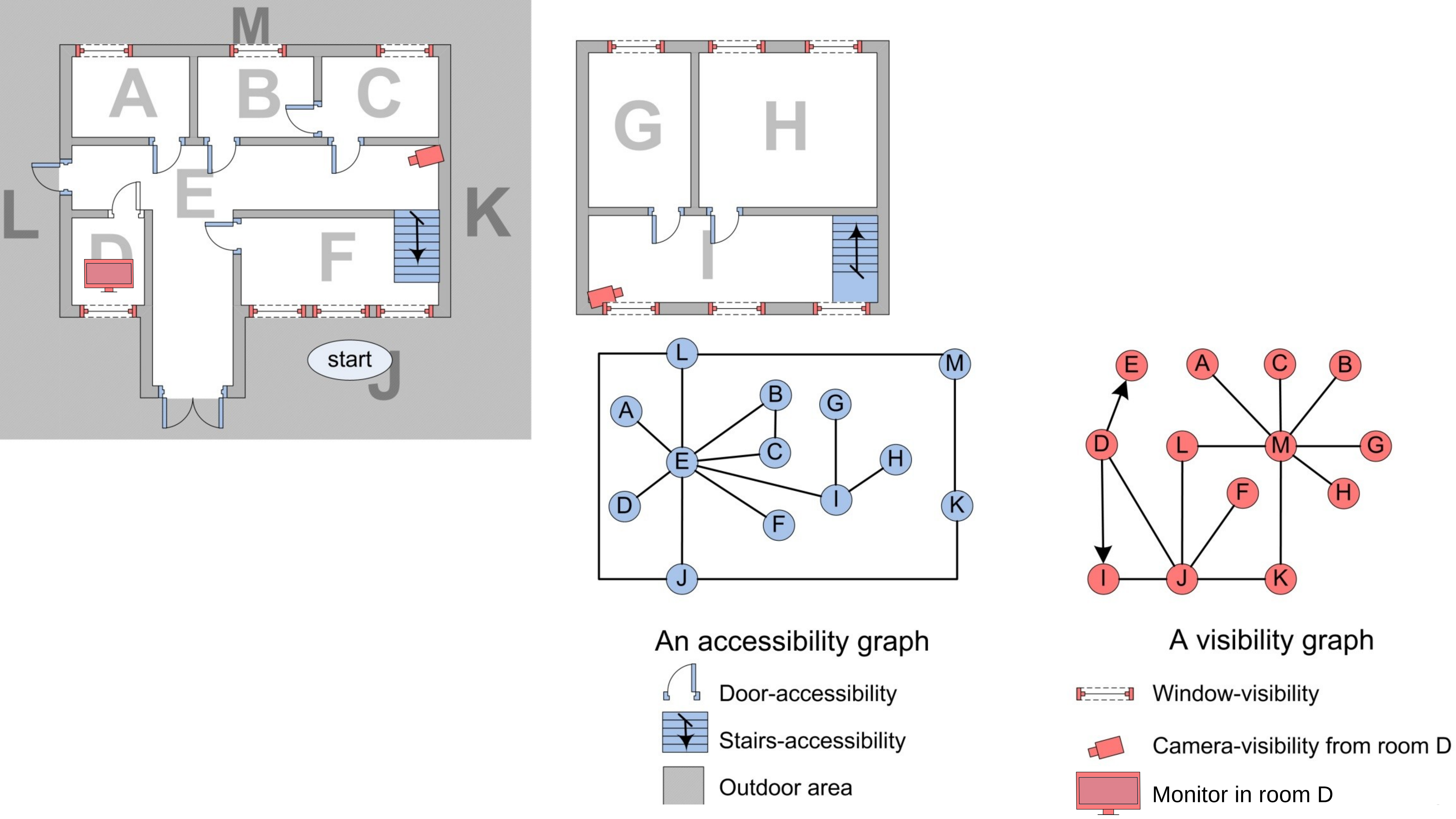

Searcher
Object
Removed


S
Removed

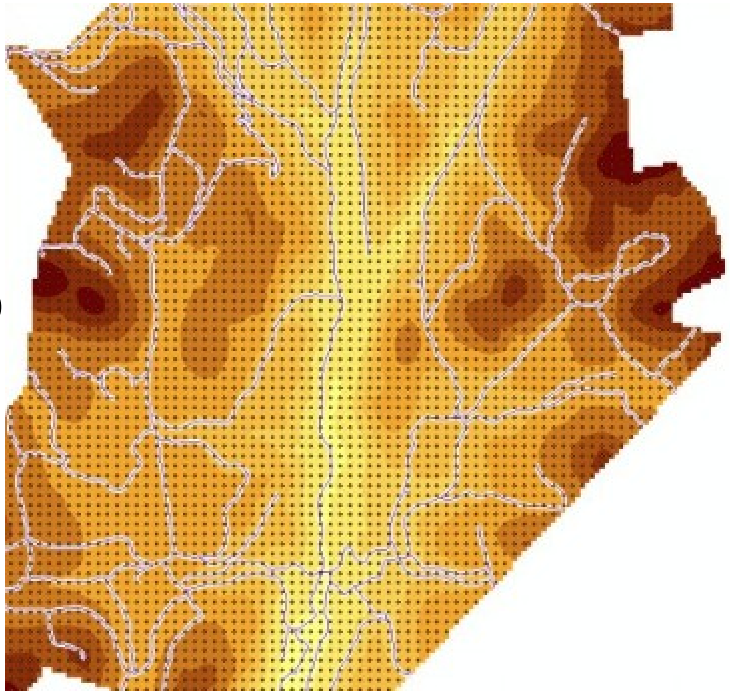

S
Removed

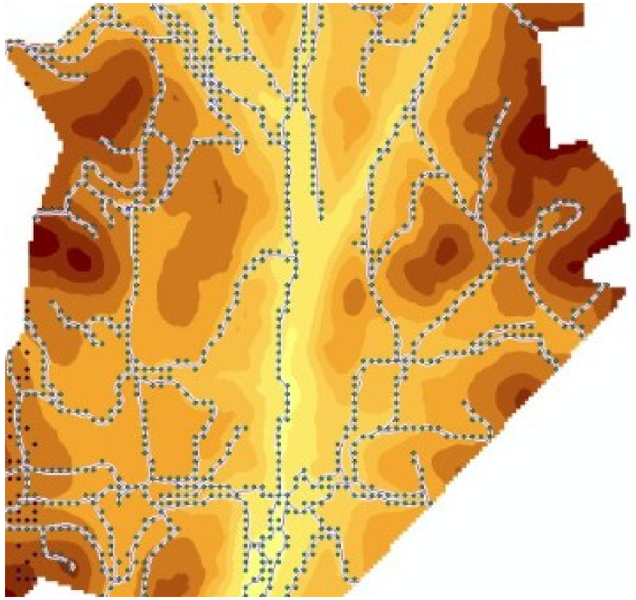
Searching with extended visibility




M
A
B
C
E
L
K
D
F
start
J
G
H
I
An accessibility graph
Door-accessibility
Stairs-accessibility
Outdoor area
A visibility graph
Window-visibility
Camera-visibility from room D
Monitor in room D

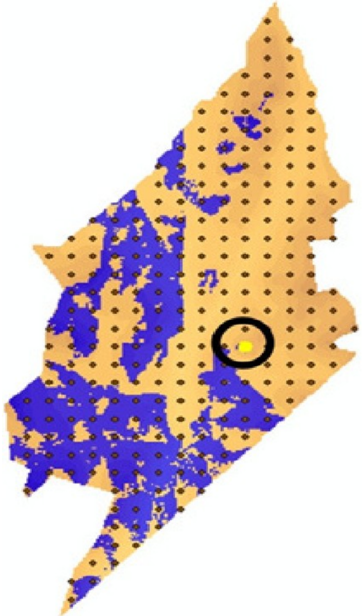
GIS map

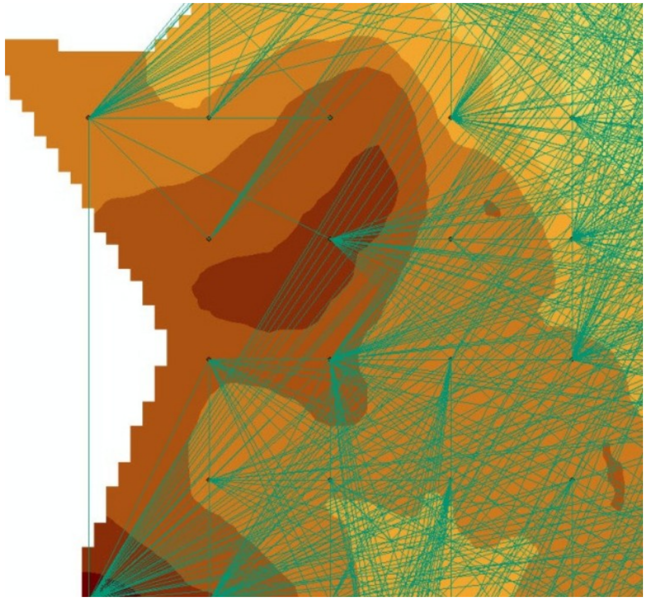




Accessibility graph

Viewshed analysis

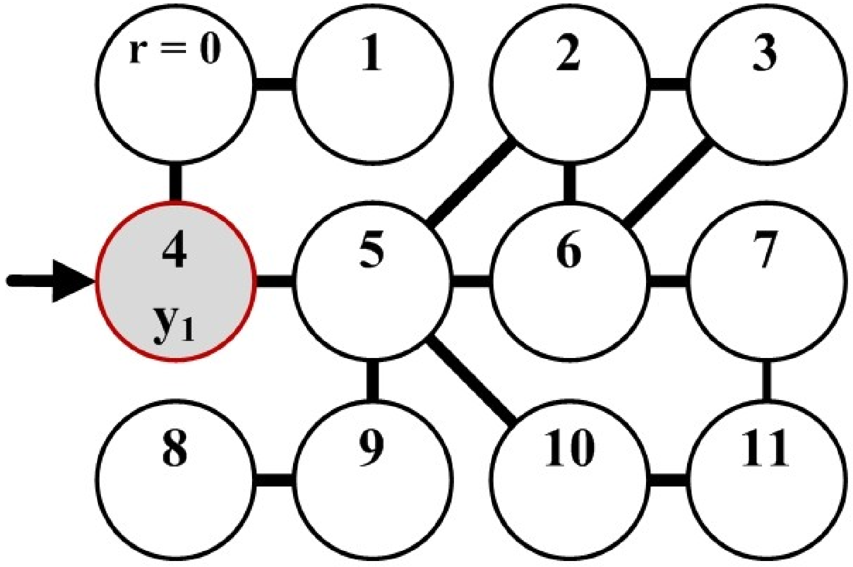




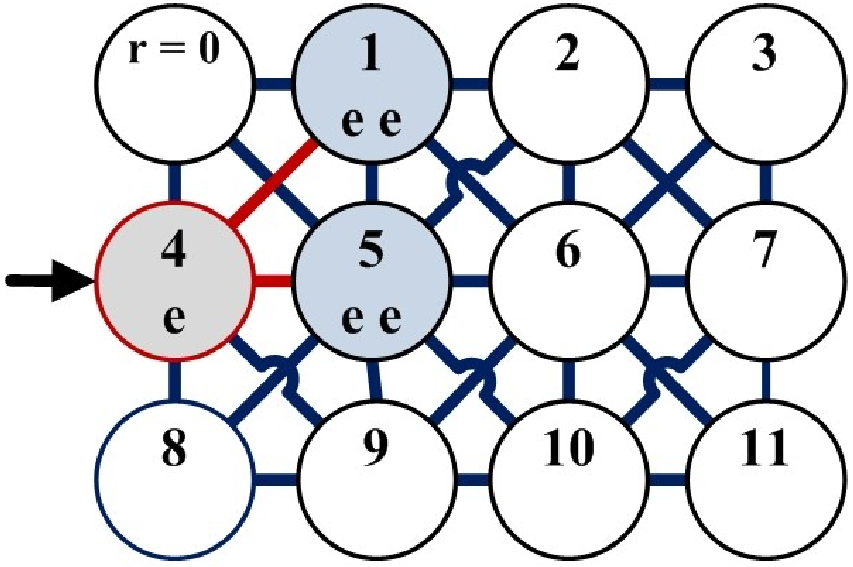
Visibility graph

Optimal Search Path with Visibility (OSPV)

- Search from a distance, ex. number of looks (scans)



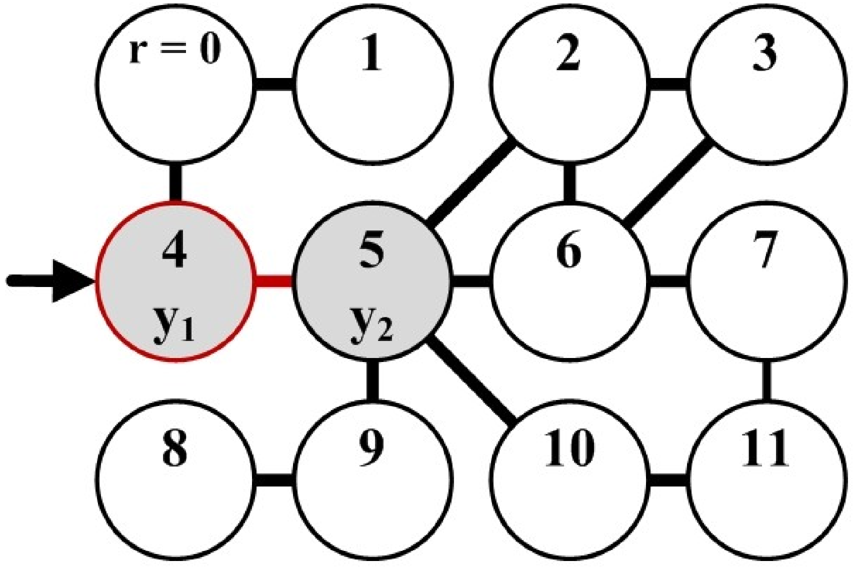
$t = 1, T = 6$



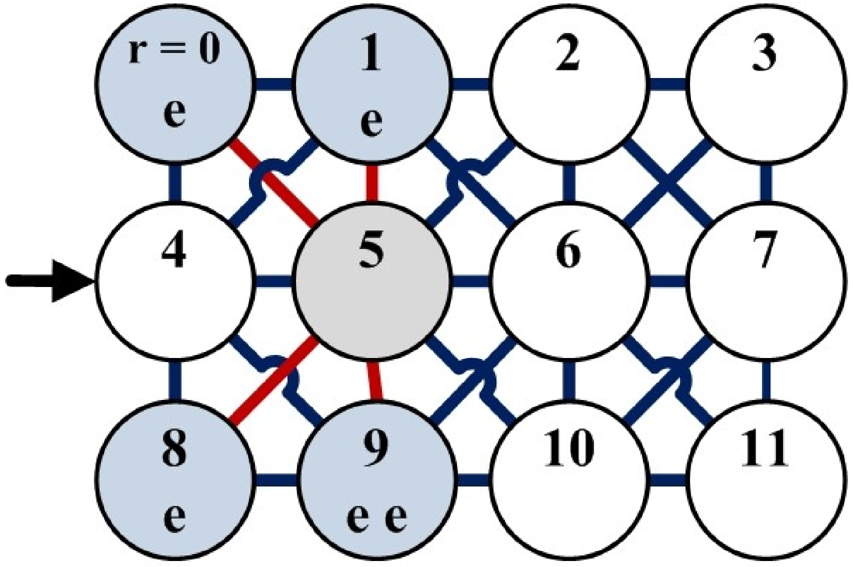
$t = 1, Q = 5$

Optimal Search Path with Visibility (OSPV)

- Search from a distance, ex. number of looks (scans)



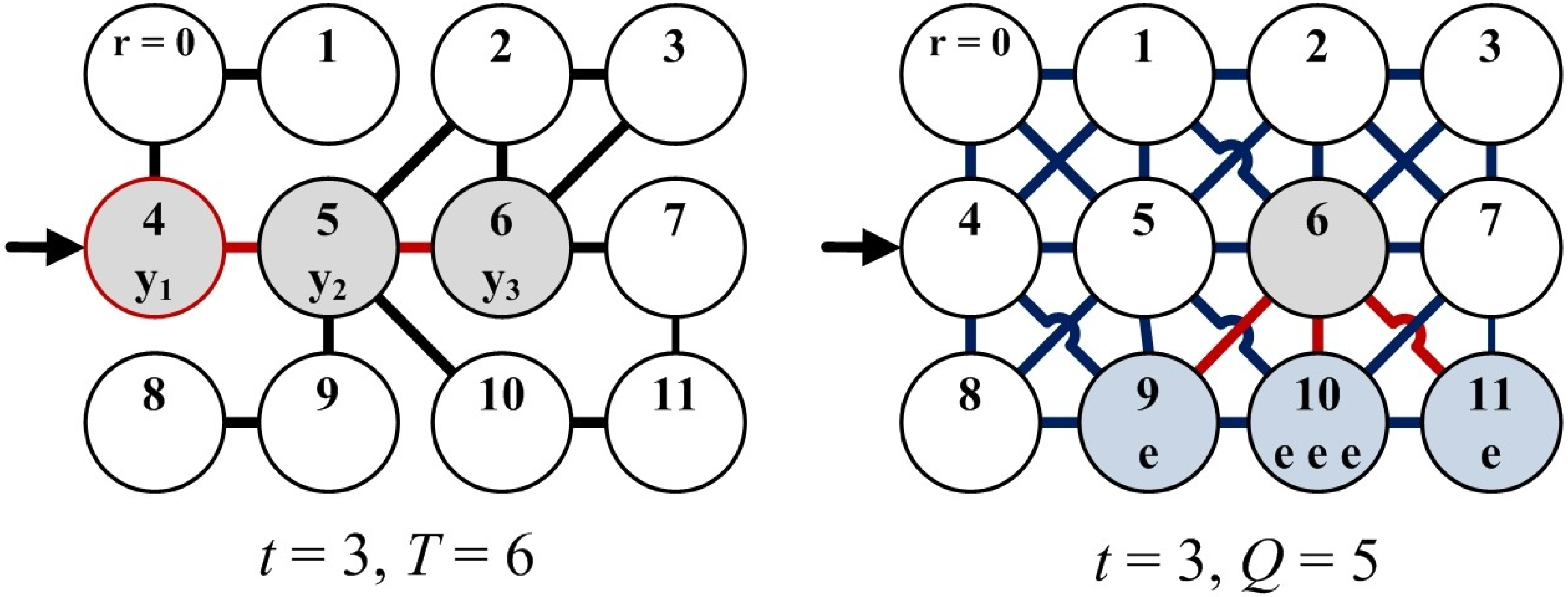
$t = 2, T = 6$



$t = 2, Q = 5$

Optimal Search Path with Visibility (OSPV)

- Search from a distance, ex. number of looks (scans)



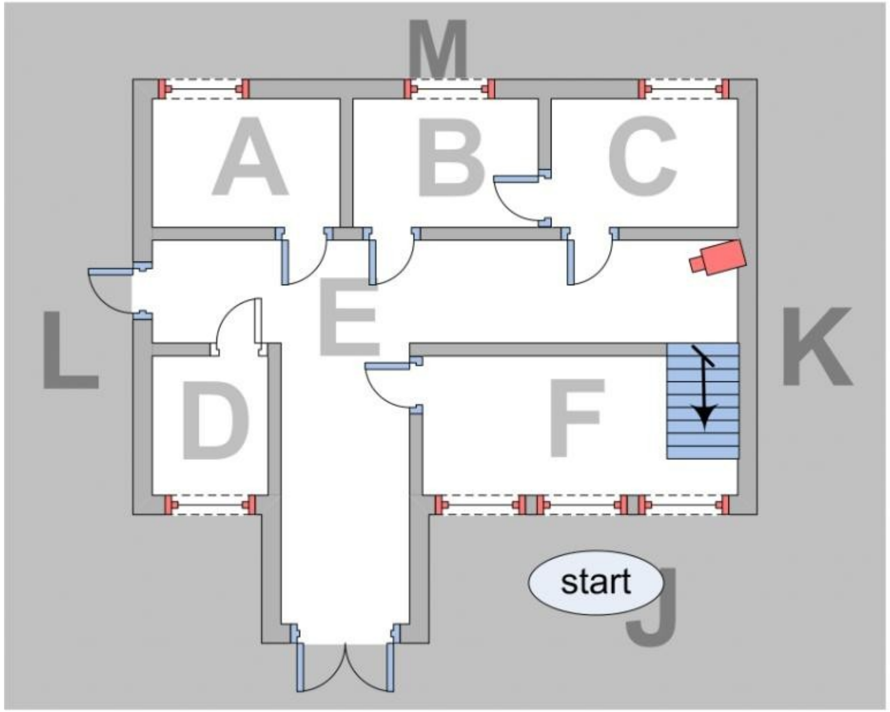
Ant Search Path with Visibility

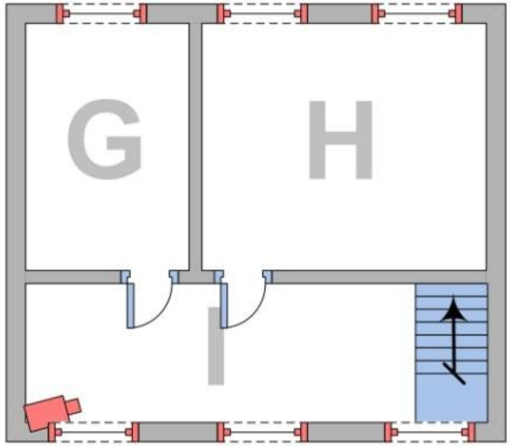
Algorithm 1: ASPV($Ospv$, C, ρ)

Input: An OSPV problem $Ospv$, the size of the colony C, and the evaporation rate ρ.
Output: The incumbent search plan P^{best}.

begin
  $\tau^{\text{path}}, \tau^{\text{eff}} \leftarrow$ `Initialize()` ;
  while *stopping criterion is not met* **do**
    $\mathcal{C} \leftarrow$ `Generate()` ;
    $P^{\text{best}}, \tau^{\text{path}}, \tau^{\text{eff}} \leftarrow$ `Update()` ;
  end
  return P^{best} ;
end

2 types of decision = 2 pheromone tables





T	Path's pheromone values												
1	A	B	C	D	E	F	G	H	I	**J**	K	L	M
2	A	B	C	D	**E**	F	G	H	I	J	K	L	M
3	A	B	C	D	E	F	G	H	**I**	J	K	L	M
4	A	B	C	D	E	F	G	**H**	I	J	K	L	M

T	Effort's pheromone values												
1	A	B	C	D	E	**F**	G	H	I	J	K	L	M
2	A	B	C	D	**E**	F	G	H	I	J	K	L	M
3	A	B	C	D	E	F	G	H	I	**J**	K	L	M
4	A	B	C	D	E	F	G	**H**	I	J	K	L	M

Components = 96 variants of the algorithm

Type	Name	Name
Initialization	iU: Uniform iR: Random	iO: OSPV
Updates	uAA: All-Ants uIB: Iteration-best uRB: Restart-best	uGB: Global-best uORBU: On restart-best upd. uOGBU: On global-best upd.
Restarts	rG: Geometric rL: Luby	rN: Without
Boosting	bY: With	bN: Without

1 initialization component is problem specific

Components = 96 variants of the algorithm

Type	Name	Name
Initialization	iU: Uniform iR: Random	iO: OSPV
Updates	uAA: All-Ants uIB: Iteration-best uRB: Restart-best	uGB: Global-best uORBU: On restart-best upd. uOGBU: On global-best upd.
Restarts	rG: Geometric rL: Luby	rN: Without
Boosting	bY: With	bN: Without

2 update components used only with restart

Components = 96 variants of the algorithm

Type	Name	Name
Initialization	iU: Uniform iR: Random	iO: OSPV
Updates	uAA: All-Ants uIB: Iteration-best uRB: Restart-best	uGB: Global-best uORBU: On restart-best upd. uOGBU: On global-best upd.
Restarts	rG: Geometric rL: Luby	rN: Without
Boosting	bY: With	bN: Without

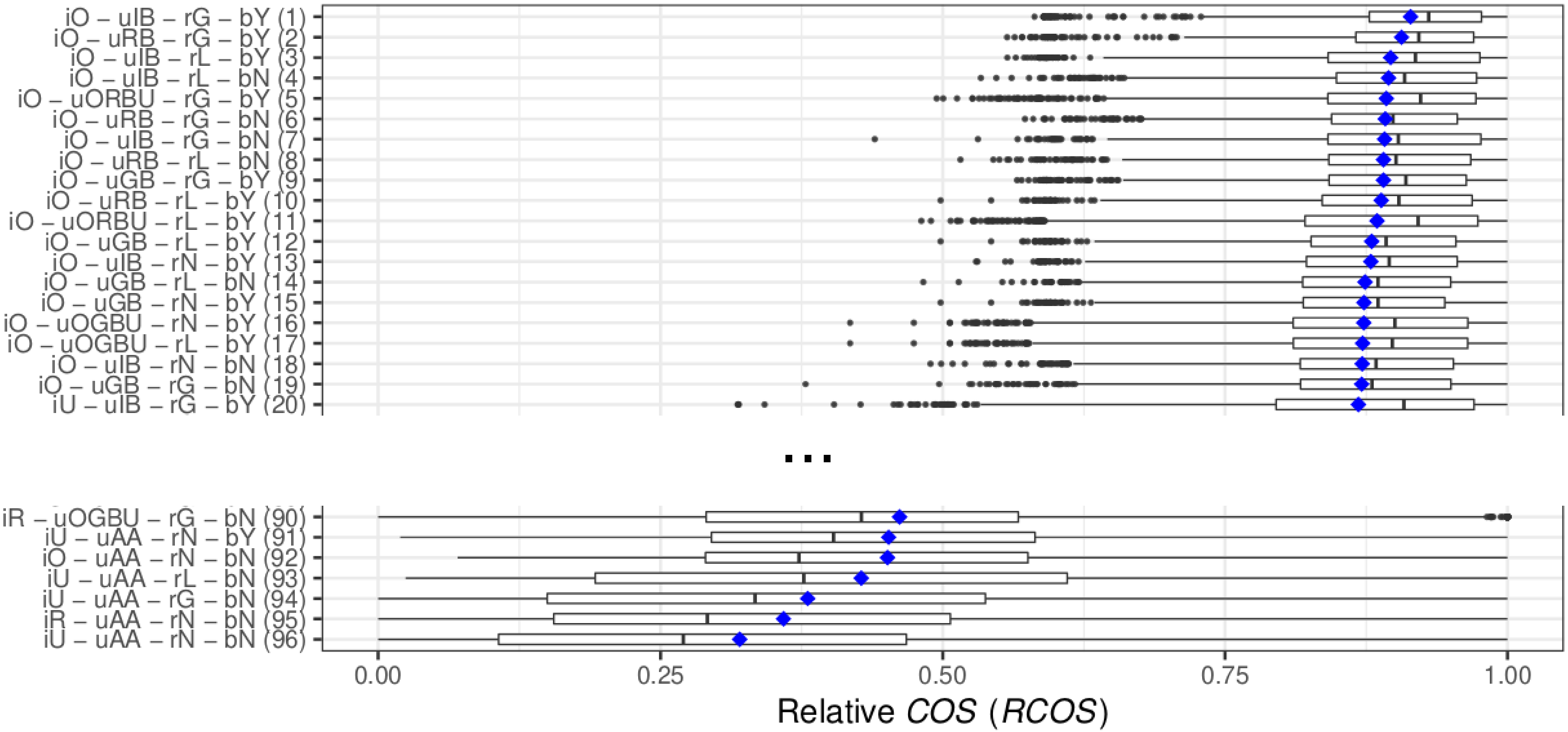
2 restart strategies based on the number of iterations without improvement

Components = 96 variants of the algorithm

Type	Name	Name
Initialization	iU: Uniform iR: Random	iO: OSPV
Updates	uAA: All-Ants uIB: Iteration-best uRB: Restart-best	uGB: Global-best uORBU: On restart-best upd. uOGBU: On global-best upd.
Restarts	rG: Geometric rL: Luby	rN: Without
Boosting	bY: With	bN: Without

Boosting use a greedy solution to increase the pheromone

100 random instances, Q ϵ {1, 2, ..., 5}, T ϵ {4, 9, 16, ... 121}



Relative Cumulative Overall Probability of Success (50 instances)

		Number of scans (Q)				
Method	T	1	2	3	4	5
iO - uIB - rG - bY	9	[*1*, *1*]	[.99, .99]	[*1*, *1*]	[**.97**, **.97**]	[**1**, **1**]
	25	[.96, .98]	[**.95**, **.97**]	[**.94**, **.96**]	[**.9**, **.92**]	[*.94*, *.96*]
	49	[**.97**, **.97**]	[**.96**, **.97**]	[**.96**, **.97**]	[**.97**, **.98**]	[**.95**, **.96**]
	81	[**.88**, **.90**]	[**.92**, **.94**]	[**.95**, **.96**]	[**.97**, **.98**]	[**.97**, **.98**]
	121	[**.82**, **.86**]	[*.87*, *.90*]	[**.95**, **.96**]	[**.92**, **.93**]	[**.94**, **.95**]
Greedy	9	.98	.95	.96	.89	.94
	25	.47	.89	.92	.89	*.96*
	49	.66	.68	.92	.93	.80
	81	.64	.35	.58	.81	.93
	121	.49	*.87*	.35	.80	.82
Best of CPLEX	9	*1*	**1**	*1*	.87	.88
	25	**1**	.35	.41	.30	–
	≥ 36	–	–	–	–	–

Empirical contributions to ACO

- Problem-specific pheromone initialization works
- Restarts pay
- Targeted pheromone updates using quality candidates are best (especially *iteration-best* in our case)
- Boosting may be beneficial

Contributions to search operations planning

- Optimal search path:

  - More flexible with the « V » for visibility
  - So far : ASPV is the best algorithm for the OSPV

- We made steps in the direction of lower-level path planning for operational systems

Version of record:

Morin, M., I. Abi-Zeid, C.-G. Quimper (2023).
Ant Colony Optimization for Path Planning in Search and Rescue Operations. In: European Journal of Operational Research 305(1), 53–63.
https://doi.org/10.1016/j.ejor.2022.06.019