Doktorantūros ataskaita

Mindaugas Kepalas (III kursas, I semestras)

2023-03-15

Doktorantūros suvestinė

Doktorantas	**Mindaugas Kepalas**
Disertacijos tema	**Optimalus vietų išdėstymas tinkle**
Doktorantūros vadovas	**prof. dr. (HP) Julius Žilinskas**
Doktorantūros laikotarpis	**2020 spalio 1 d. — 2024 rugsėjo 30 d.**
Kursas, semestras	**III kursas I semestras**

Studijų plano suvestinė

Studijų metai	Egzaminai		Dalyvavimas konferencijose		Publikacijos		
	Planas	Įvykdyta	Planas	Įvykdyta	Planas	Įvykdyta	Būklė
I (2020/2021)	2	2					
II (2021/2022)	2	2	1	2			
III (2022/2023)			**1**		**1**		
IV (2023/2024)					1		

Ataskaitinio pusmečio suvestinė

Dalyvavimas konferencijose		
Planas	**Įvykdyta**	**Konferencijos tipas**
DAMSS konferencija, 2022 gruodžio 1-3, Druskininkai, Lietuva	Mindaugas Kepalas, Julius Žilinskas, “Handling Two-Dimensional Net Constraint: Its Formulation in MIP Terms and Empirical Studies of Solvers”	Nacionalinė

Mokslinių tyrimų ir disertacijos rengimo etapai

Darbo pavadinimas	Atlikimo terminai	Pastabos
Disertacijos tikslų ir uždavinių formulavimas	2021 rugsėjo mėn.	Atlikta
Literatūros apžvalga	Visu doktorantūros metu	Atliekama
Uždavinių teorinis tyrimas	Visu doktorantūros metu	Atliekama
Optimizavimo algoritmų programavimas (uždavinių sprendimas)	Visu doktorantūros metu	Atliekama
Mokslinės literatūros (straipsnių, knygų) skaitymas, sisteminimas, analizė	Visu doktorantūros metu	Atliekama
Dalyvavimas konferencijoje (-ose)	2022 rugsėjo mėn.	Atlikta
Moksliniai straipsniai	2023, 2024 pavasaris	Nepradėta
Disertacijos rašymas	2023 rugsėjo mėn. - 2024 birželio mėn.	
Disertacijos įteikimas	2024 birželio mėn.	

Gauti moksliniai rezultatai ir planai kitam semestrui

- **III kurso I semestre gauti moksliniai rezultatai**: žr. mokslinę ataskaitą, straipsnių šiuo metu nėra (planuota III kurso I semestrą)
- **Planai III kurso II semestrui:**
  - Sukurti disertacijos uždavinio modeliavimo įrankį
  - Išbandyti įvairius algoritmus ir strategijas uždaviniui spręsti
  - Paruošti pirmąjį straipsnį

Inžinierių uždavinys

- **Uždavinys.** Duotam pastatui išdėlioti atramas taip, kad "apkrova" kiekvienai atramai būtų kuo labiau vienoda (lygi).
- **Daugiau apie uždavinį straipsnyje** Belevičius, R., Ivanikovas, S., Šešok, D., Valentinavičius, S. and Žilinskas, J., 2011. Optimal placement of piles in real grillages: experimental comparison of optimization algorithms. *Information Technology and Control, 40*(2), pp.123-132

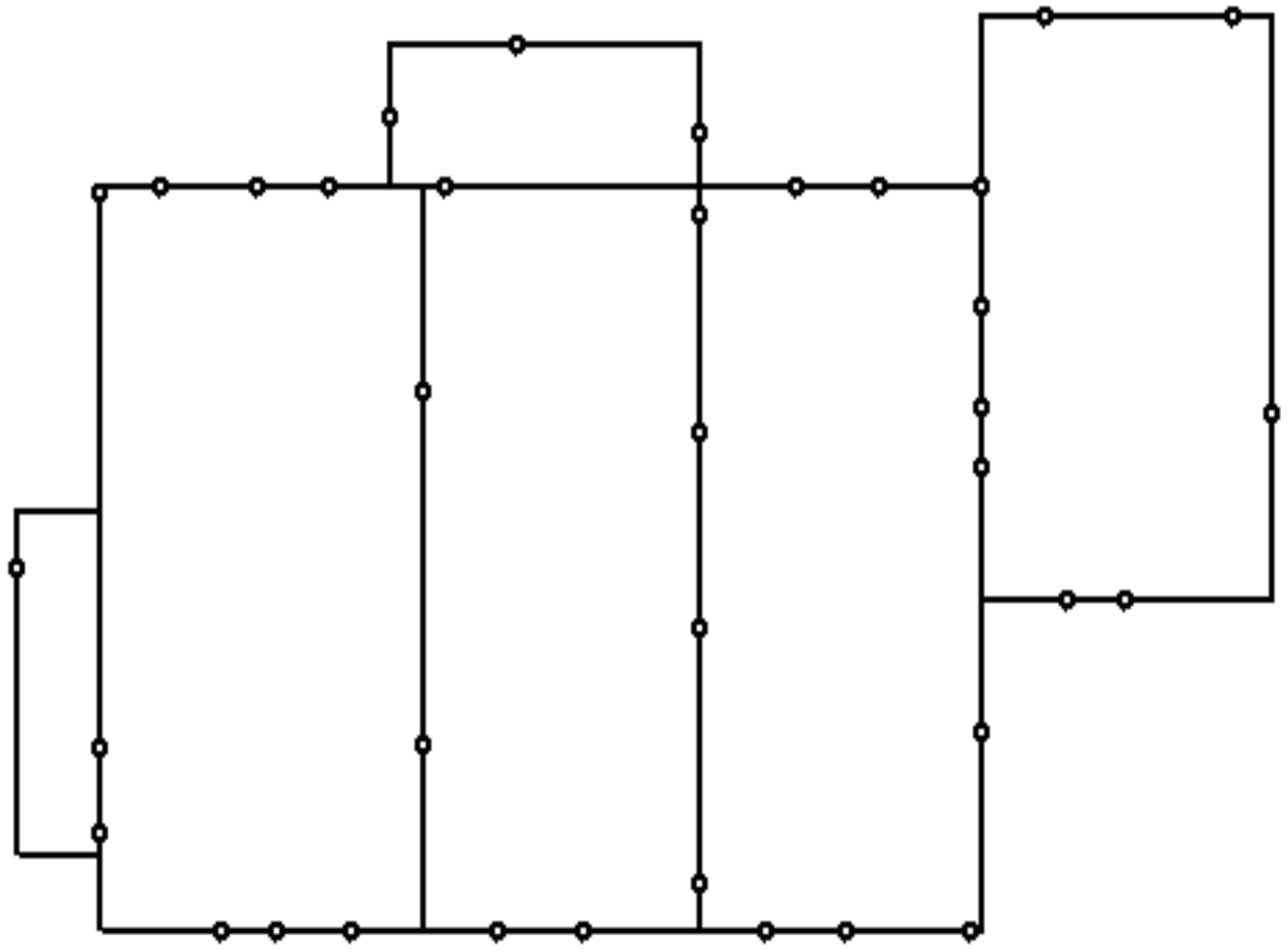


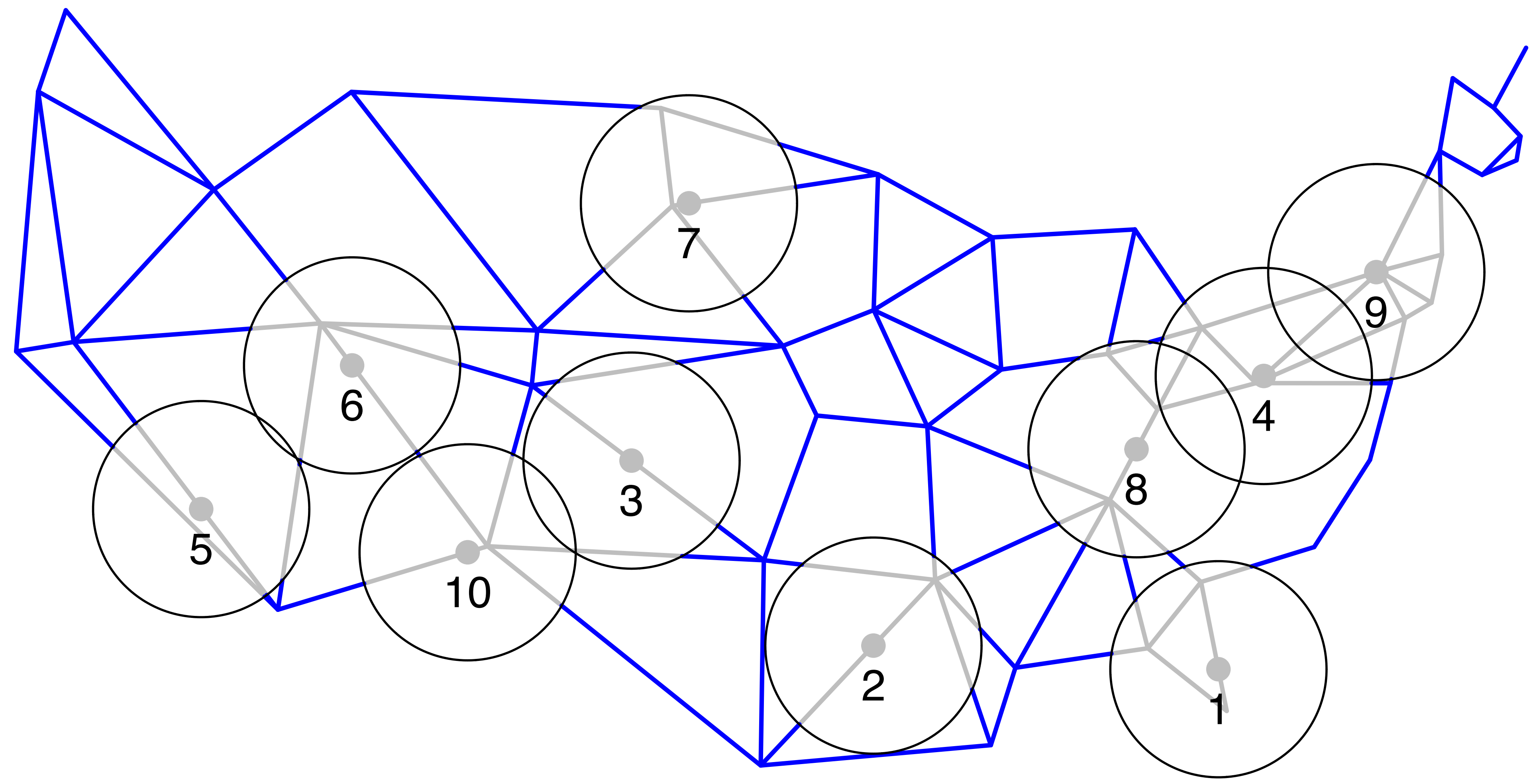
Figure 8. The best found solution of grillage No 6

Geriausias rastas atramų išdėstymas vienam iš pastatų (paveiksliukas iš straipsnio)

Uždavinio apibendrinimas: Network Locations

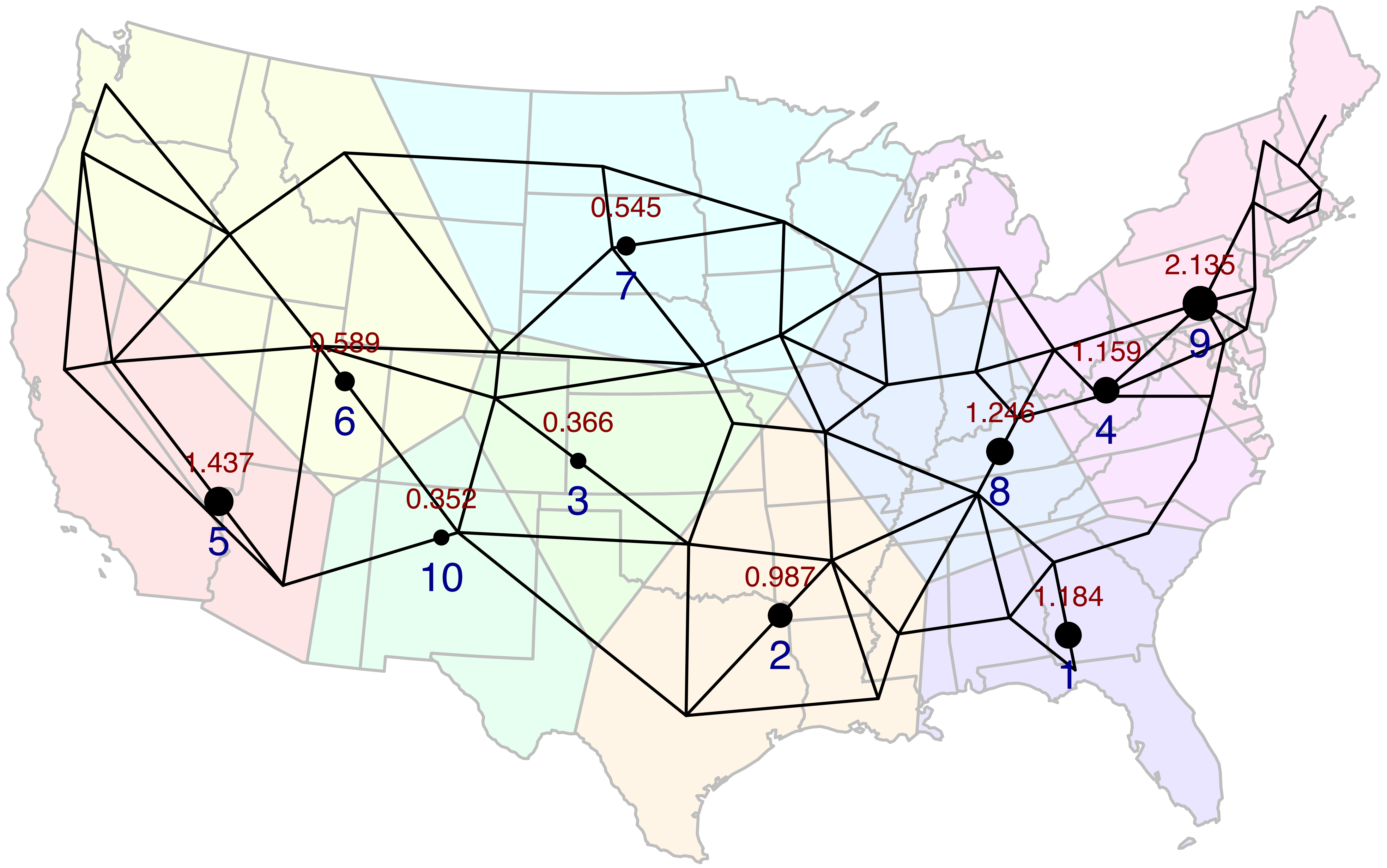
- **Duota:**
  - Tinklas plokštumoje $\mathcal{N}$ ("Network");
  - Taškų skaičius M ("Points"), taškai turi priklausyti tinklui: $P_i \in \mathcal{N}, \;\; i = 1,\ldots,M$;
  - Minimalus atstumas r, kuris turi skirti bet kuriuos du skirtingus taškus ("Radius");

Tinklas ir jame sugeneruoti taškai, tenkinantys uždavinio sąlygas



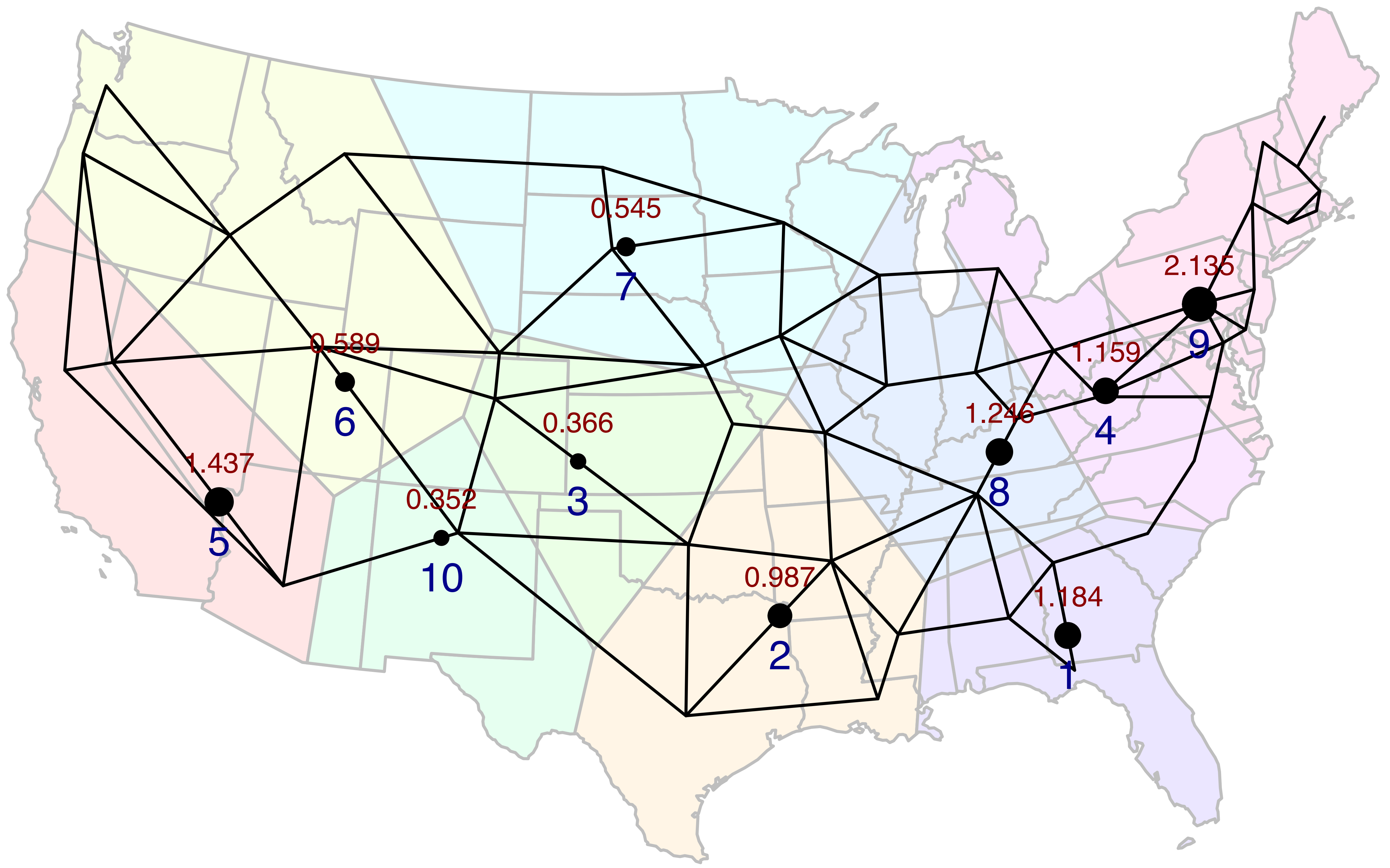
Uždavinio apibendrinimas: Network Locations

- **Duota:**
  - Vektorinė "apkrovos" funkcija $F : \{P_1, P_2, \ldots, P_M\} \mapsto \mathbb{R}^M$ ("Objective");
  - Nuostolių (kaštų) funkcija $cost : \mathbb{R}^M \mapsto \mathbb{R}$ ("Loss");
- **Uždavinys.** Rasti M tinklui priklausančių taškų išdėstymą, kad nuostoliai (kaštai) būtų **minimalūs**.



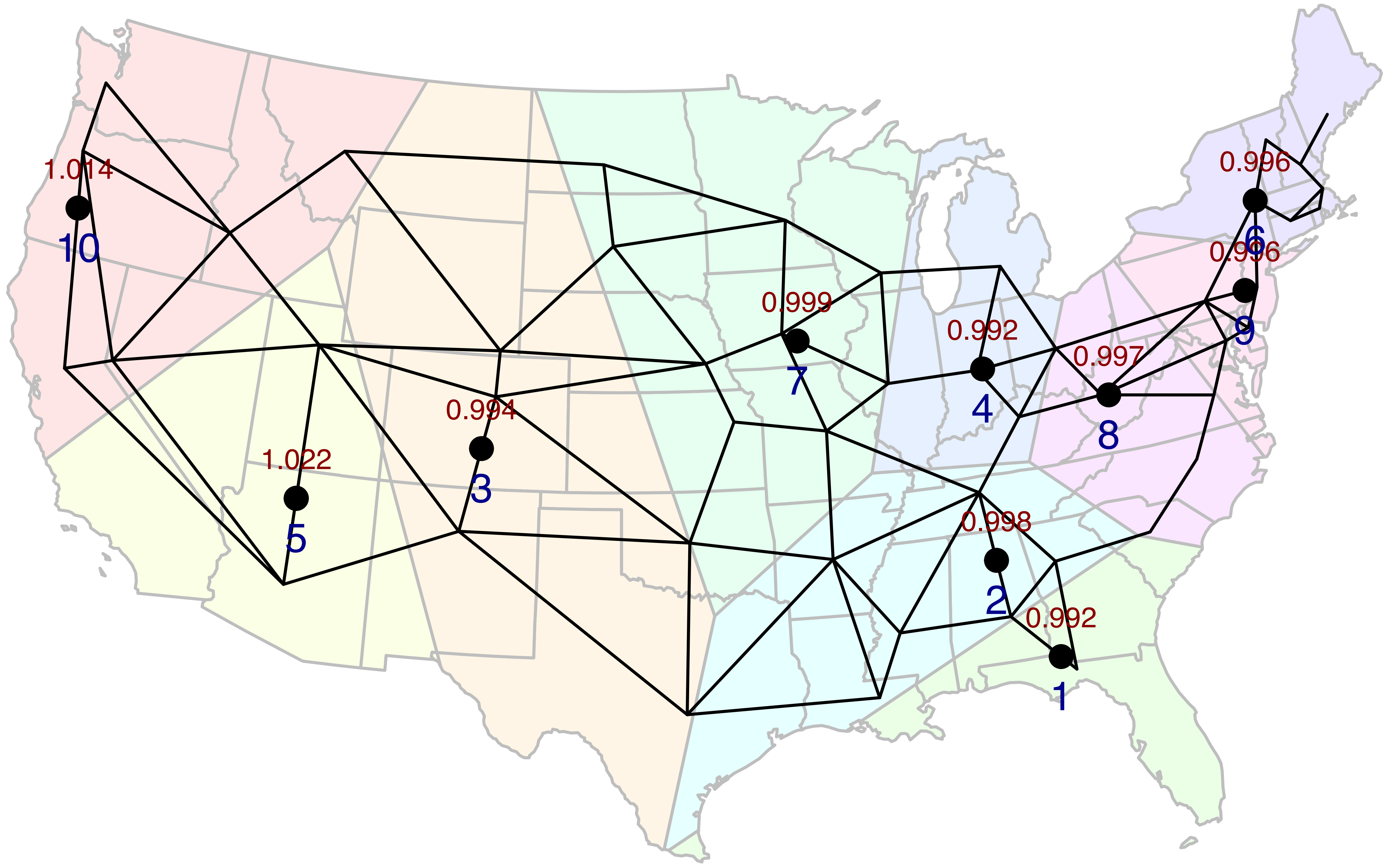
Mokslinio darbo kryptis

- **Ateities planai:** norėčiau sukurti modeliavimo įrankį (interaktyvią programą), kuris leistų/ padėtų optimizuoti funkcijas, apibrėžtas taškų pozicijoms tinkle arba plokštumos dalyje.
  - Optimizuojama funkcija nebūtinai turi turėti analitinę išraišką.
  - Optimizuojama funkcija gali turėti trūkius.
- Aktualūs uždaviniai optimizuojant:
  - **Lokalaus sprendinio radimas.**
  - **Globalaus sprendinio paieška.**



Išsikelti tikslai III kurso I semestrui (neįgyvendinti)

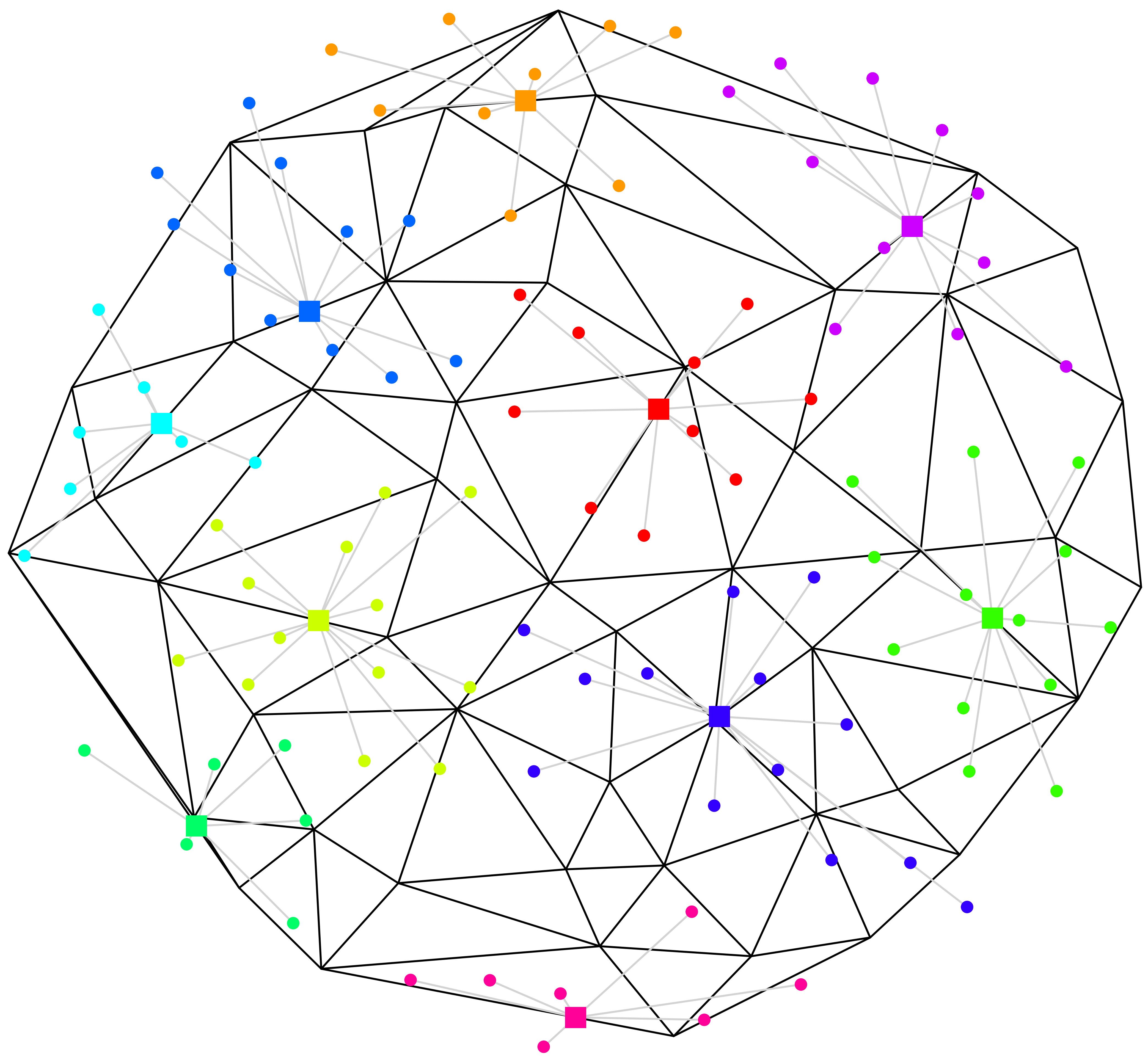
- Interaktyvi programa, kuri leidžia **įterpti, išstrinti, perkelti** arba **fiksuoti** taškų padėtis tinkle ir keisti kitus uždavinio nustatymus.
- Išbandyti įvairias optimizavimo strategijas, siekiant, kad "geras" sprendinys būtų rastas pakankamai efektyviai (greitai).
- Įgyvendinti lokalaus sprendinio radimo metodus (Niutono metodas).
- Palyginti optimizavimo rezultatus (algoritmus) statistiškai (algoritmo sprendinio kokybė *vs* sugaištas laikas sprendiniui rasti).



"DAMSS 2022" pranešimo uždavinys

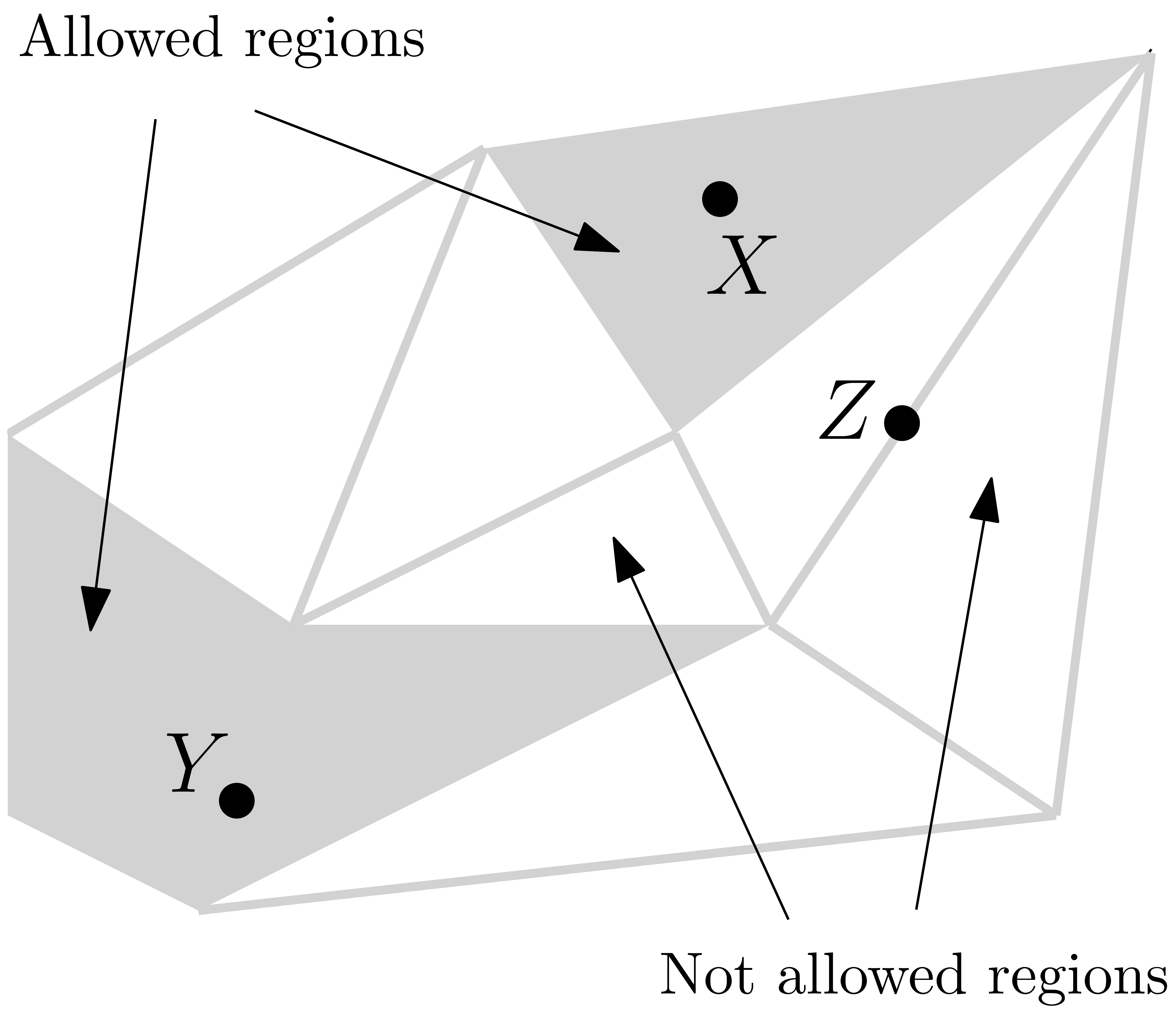
- **Duota:**
  - Tinklas plokštumoje $\mathcal{N}$ ("Network");
  - Sandėlių skaičius M ("Facilities"), juos atitinkantys taškai turi priklausyti tinklui: $P_i \in \mathcal{N}, \;\; i = 1,\ldots,M$;
  - N klietų ("Customers") su koordinatėmis $Q_j \in \mathbb{R}^2$ ir svoriais $w_j > 0, \;\; j = 1,\ldots,N$;
- **Uždavinys.** Rasti sandėlių pozicijas $P_i \in \mathcal{N}$, $i = 1,\ldots,M$ ir priskirti klientus sandėliams taip, kad kaštai aptarnauti klientus būtų mažiausi:

$$\min_{\{P_i \in \mathcal{N},\, i=1,\ldots,M\}} \sum_{i=1}^{M} \sum_{j \in \mathcal{C}_i} w_j \, d(P_i, Q_j)$$



Tinklo ribojimai kaip MIP nelygybės

- Taško priklausymą tinklui ($P_i \in \mathcal{N}$, $i = 1, \ldots, M$) galima užrašyti **MIP (mixed-integer-programming) nelygybėmis.**
- Toks ribojimas dažnai nekeičia uždavinio tipo: pavyzdžiui, **k-means uždavinys su tinklo ribojimu lieka MIQCP uždaviniu.**
- Iš tikrųjų, ribojimai nebūtinai turi būti segmentų sąjunga: **bet kokią atkarpų ir trikampių sąjungą galima užrašyti MIP nelygybėmis.**
- Kadangi bet kokį daugiakampį galima suskaidyti į trikampius, tai **bet kokią atkarpų ir daugiakampių sąjungą galima užrašyti MIP nelygybėmis.**



Apibendrintas uždavinys

- **Duota:**
  - aibė taškų $Q_1, Q_2, \ldots, Q_N$ ir jų svoriai $w_1, w_2, \ldots, w_N > 0$
  - ribojimai galimoms sandėlių pozicijoms $P_1, P_2, \ldots, P_M$
  - ribojimai ("barriers to travel") galimam kelionės maršrutui iš taškų $Q_1, Q_2, \ldots, Q_N$ į taškus $P_1, P_2, \ldots, P_M$, nulemiantys atstumo "metriką" $d(P, Q)$ taškams $P, Q \in \mathbb{R}^2$
- **Rasti:** taškus $P_1, P_2, \ldots, P_M$ kurie tenkina ribojimus ir kad atstumų suma (atsižvelgiant į barjerus) iki taškų $Q_1, Q_2, \ldots Q_N$ būtų mažiausia: $\min_{P_1, P_2, \ldots, P_M} \sum_{i=1}^{M} \sum_{j \in \mathscr{C}_i} d(P_i, Q_j)$