Doktorantūros ataskaita

Mindaugas Kepalas (III kursas, II semestras)

2023-09-23

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 II 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**	**1**	**0**	**Ruošiamas**
IV (2023/2024)					1		

Ataskaitinio pusmečio suvestinė

Dalyvavimas konferencijose		
Planas	**Įvykdyta**	**Konferencijos tipas**
The 2023 World Congress on Global Optimization (WCGO 2023), Atėnai, Graikija, liepos 10-14 d.	Mindaugas Kepalas, Julius Žilinkas “2-Dimensional Net-Constrained Clustering Problem”	Tarptautinė

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	Straipsnis ruošiamas
Disertacijos rašymas	2023 rugsėjo mėn. - 2024 birželio mėn.	Planuojama pradėti išspausdinus pirmąjį straipsnį
Disertacijos įteikimas	2024 birželio mėn.	

Gauti moksliniai rezultatai ir planai kitam semestrui

- **III kurso II semestre gauti moksliniai rezultatai**: žr. mokslinę ataskaitą, straipsnių šiuo metu neturiu (nuo doktorantūros plano atsilieku)
- **Planai IV kurso I semestrui:**
  - Paruošti pirmąjį straipsnį
  - Pradėti rašyti disertaciją

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

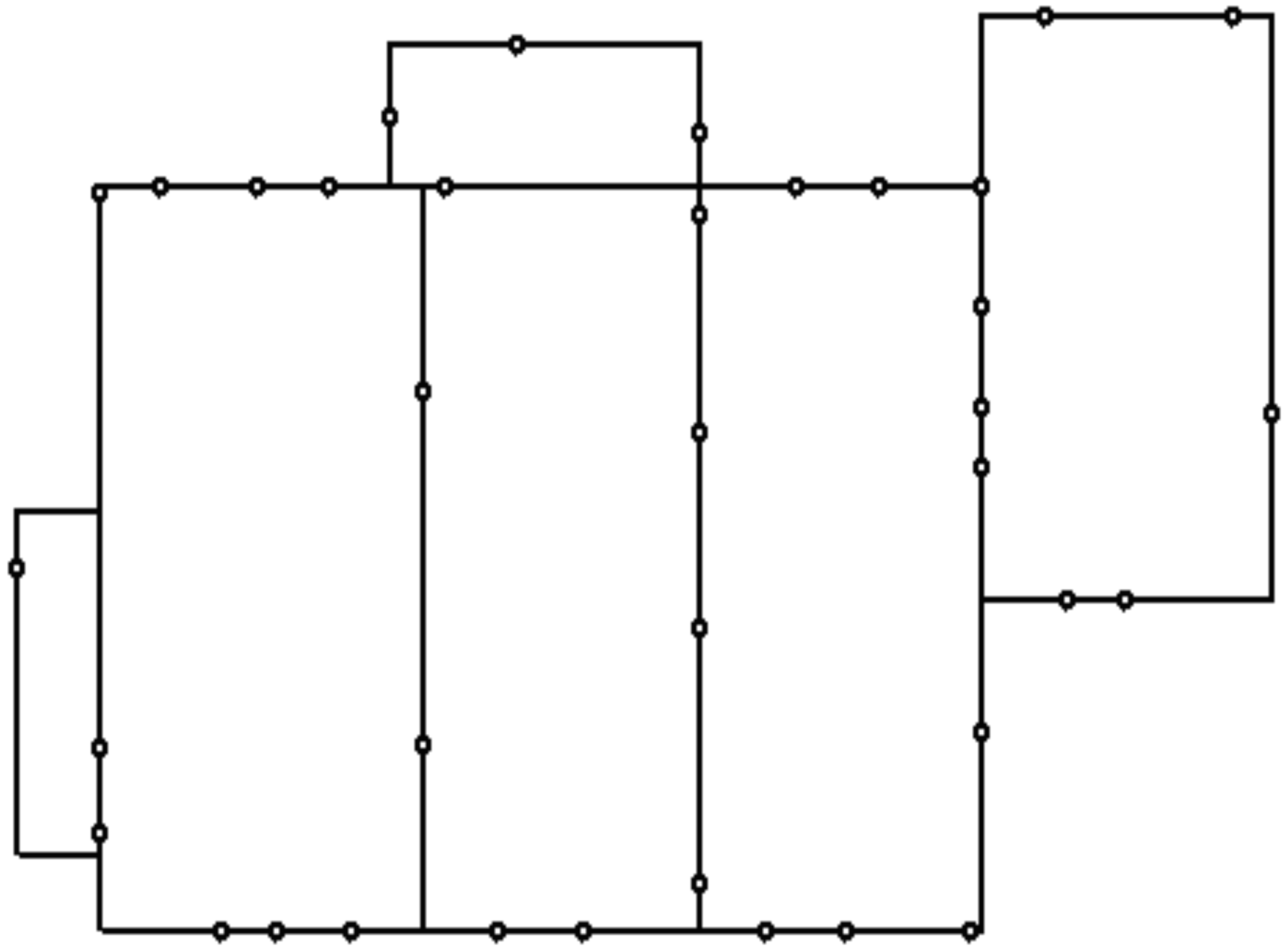


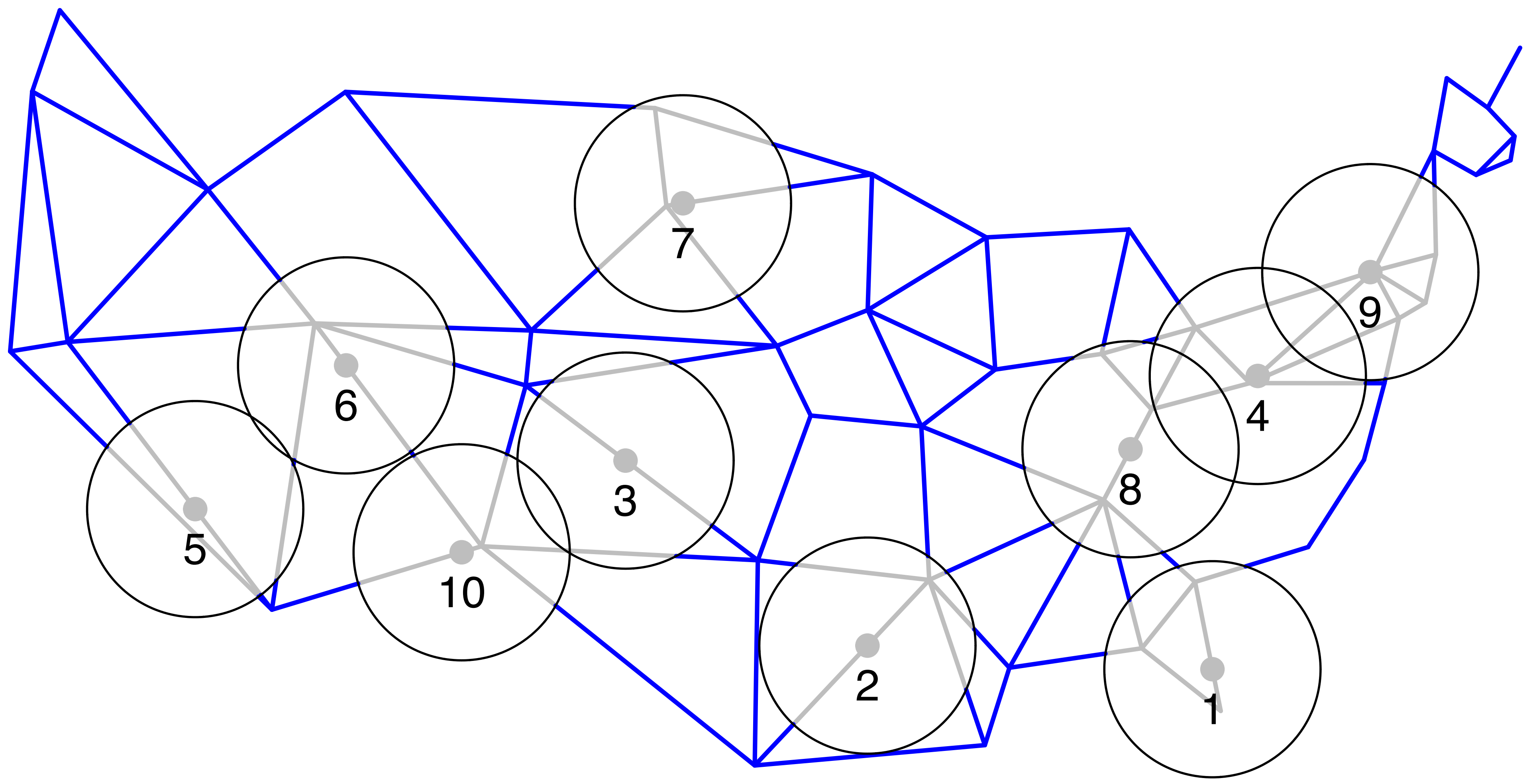
Figure 8. The best found solution of grillage No 6

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

Uždavinio apibendrinimas: Network Locations

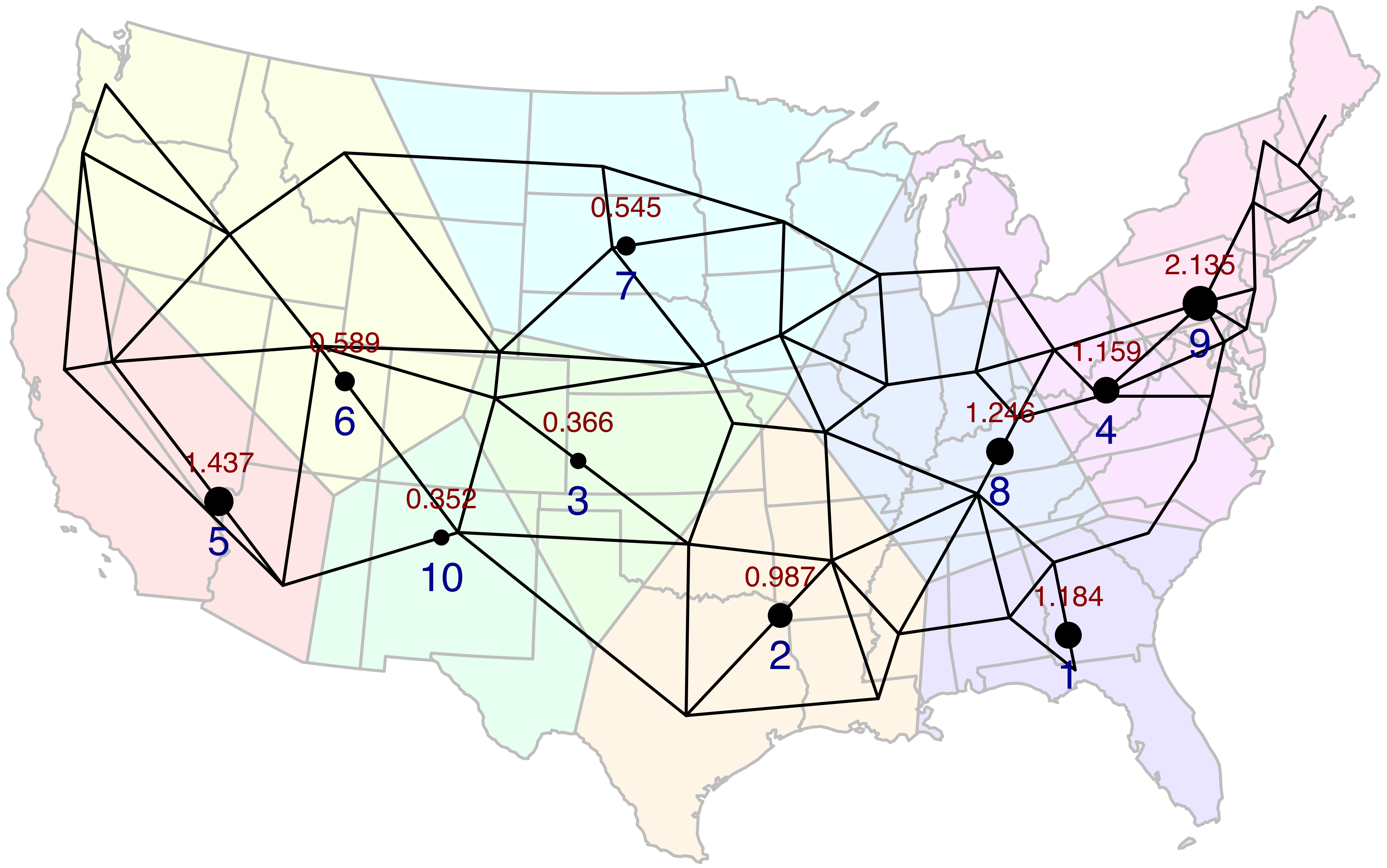
- **Duota:**
  - Tinklas plokštumoje $\mathcal{N}$ ("Network");
  - Taškų skaičius M ("Points"), taškai turi priklausyti tinklui: $P_i \in \mathcal{N}, \;\; i = 1,\ldots,M$;
  - Minimalus atstumas r, kuris turi skirti bet kuriuos du skirtingus taškus ("Radius");

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 jų išdėstymo “kaina” būtų **minimali**.



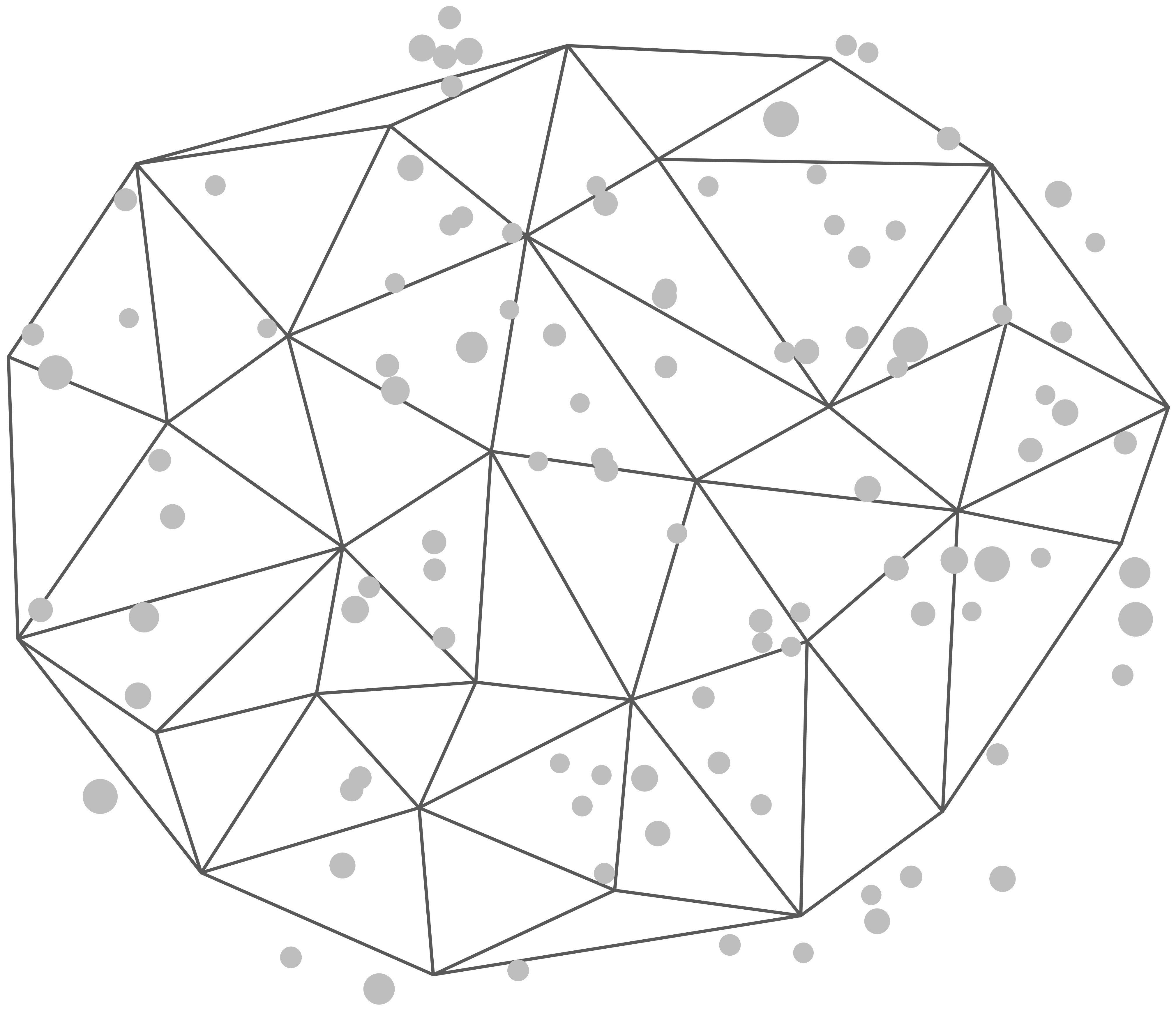
“WCGO 2023” nagrinėtas uždavinys

“WCGO 2023” nagrinėtas uždavinys

- **Duota:**
  - Taškų aibė $\{P_1, P_2, \ldots, P_N\}$ su svoriais $\{w_1, w_2, \ldots, w_N\}$
  - “Tinklas” $\mathcal{N}$ (šis “tinklas” yra M segmentų sąjunga)
  - Klasterių skaičius K

“WCGO 2023” nagrinėtas uždavinys

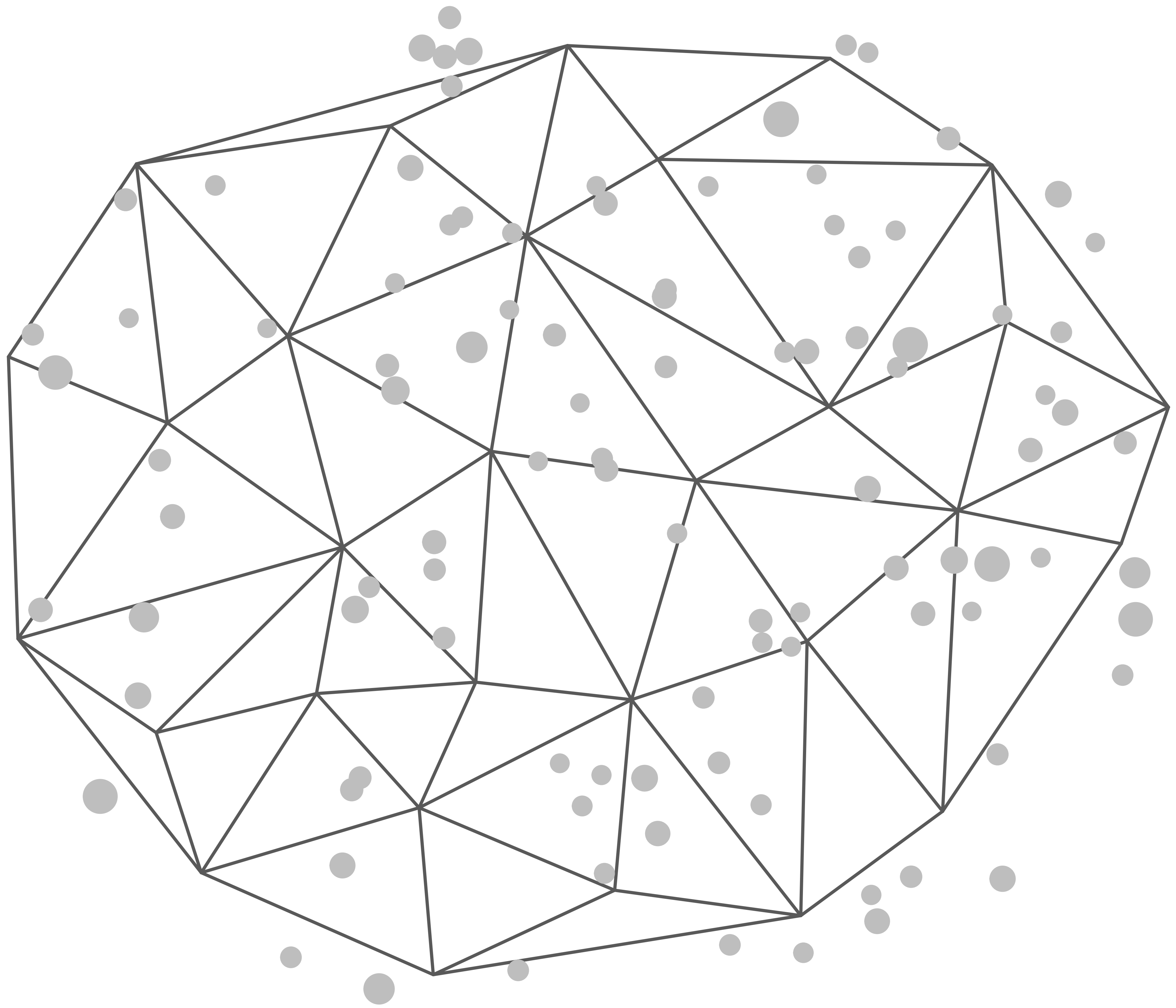
- **Duota:**
  - Taškų aibė $\{P_1, P_2, \ldots, P_N\}$ su svoriais $\{w_1, w_2, \ldots, w_N\}$
  - “Tinklas” $\mathcal{N}$ (šis “tinklas” yra M segmentų sąjunga)
  - Klasterių skaičius K



"WCGO 2023" nagrinėtas uždavinys

- **Duota:**
  - Taškų aibė $\{P_1, P_2, \ldots, P_N\}$ su svoriais $\{w_1, w_2, \ldots, w_N\}$
  - "Tinklas" $\mathscr{N}$ (šis "tinklas" yra M segmentų sąjunga)
  - Klasterių skaičius K
- **Uždavinys.** Apibrėžti K klasterių $\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_K$ su atitinkamais centras $Q_1, Q_2, \ldots, Q_K$ randančių globalų sprendinį uždaviniui

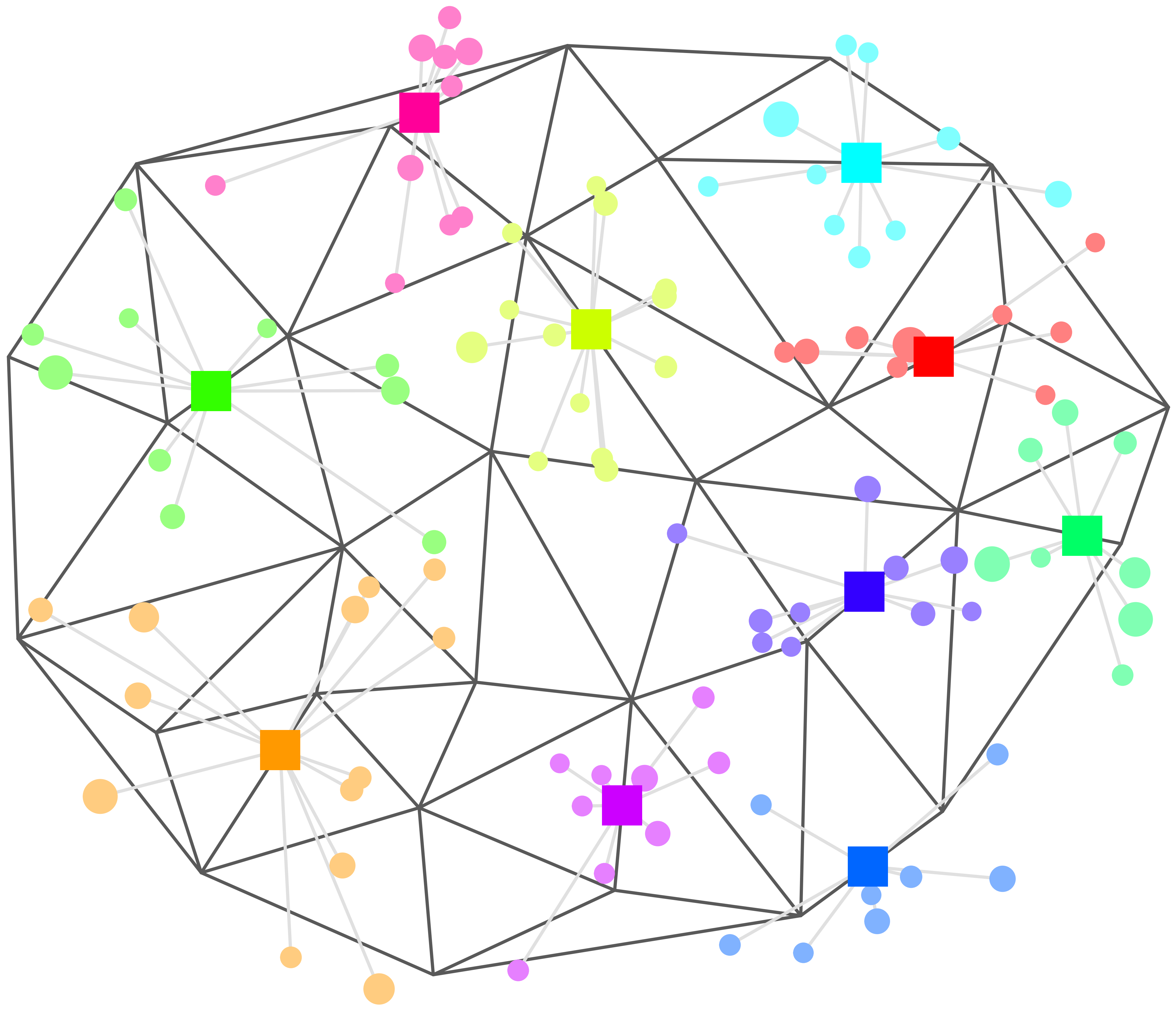
$$\min_{Q_1, Q_2, \ldots Q_K} \sum_{k=1}^{K} \sum_{i \in \mathscr{C}_k} w_i \left\| P_i - Q_k \right\|_2^2 \quad \text{s.t.} \quad Q_k \in \mathscr{N},\ k = 1, \ldots, K$$



“WCGO 2023” nagrinėtas uždavinys

- **Duota:**
  - Taškų aibė $\{P_1, P_2, \ldots, P_N\}$ su svoriais $\{w_1, w_2, \ldots, w_N\}$
  - “Tinklas” $\mathcal{N}$ (šis “tinklas” yra M segmentų sąjunga)
  - Klasterių skaičius K
- **Uždavinys.** Apibrėžti K klasterių $\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_K$ su atitinkamais centras $Q_1, Q_2, \ldots, Q_K$ randančių globalų sprendinį uždaviniui

$$\min_{Q_1, Q_2, \ldots Q_K} \sum_{k=1}^{K} \sum_{i \in \mathscr{C}_k} w_i \left\| P_i - Q_k \right\|_2^2 \quad \text{s.t.} \quad Q_k \in \mathcal{N},\ k = 1, \ldots, K$$

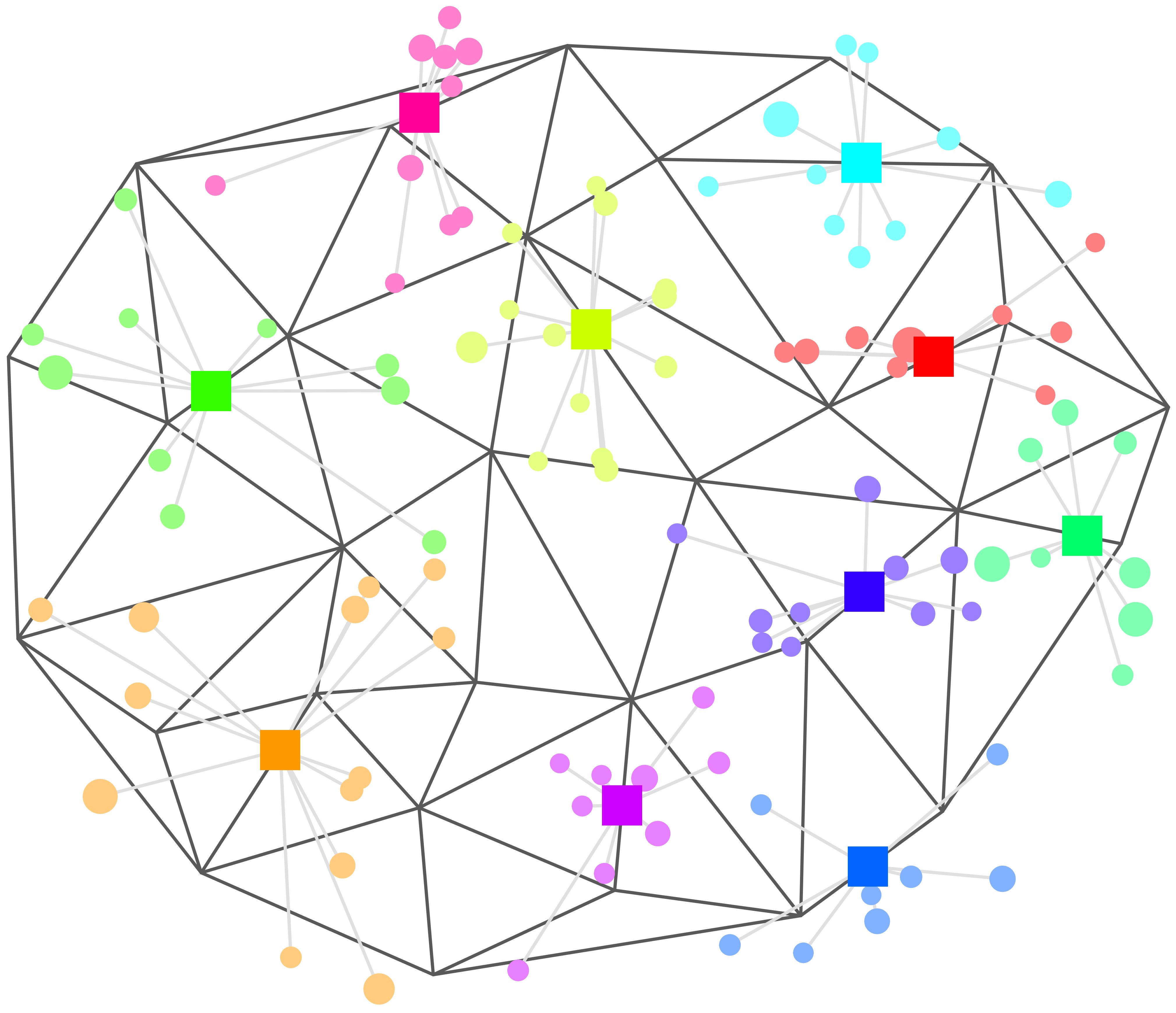


“WCGO 2023” nagrinėtas uždavinys

- **Duota:**
  - Taškų aibė $\{P_1, P_2, \ldots, P_N\}$ su svoriais $\{w_1, w_2, \ldots, w_N\}$
  - “Tinklas” $\mathcal{N}$ (šis “tinklas” yra M segmentų sąjunga)
  - Klasterių skaičius K
- **Uždavinys.** Apibrėžti K klasterių $\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_K$ su atitinkamais centras $Q_1, Q_2, \ldots, Q_K$ randančių globalų sprendinį uždaviniui

$$\min_{Q_1, Q_2, \ldots Q_K} \sum_{k=1}^{K} \sum_{i \in \mathscr{C}_k} w_i \left\| P_i - Q_k \right\|_2^2 \quad \text{s.t.} \quad Q_k \in \mathcal{N},\ k = 1, \ldots, K$$

- **Pastebėkite tinklo ribojimą klasterių centrams: $Q_k \in \mathcal{N}$**

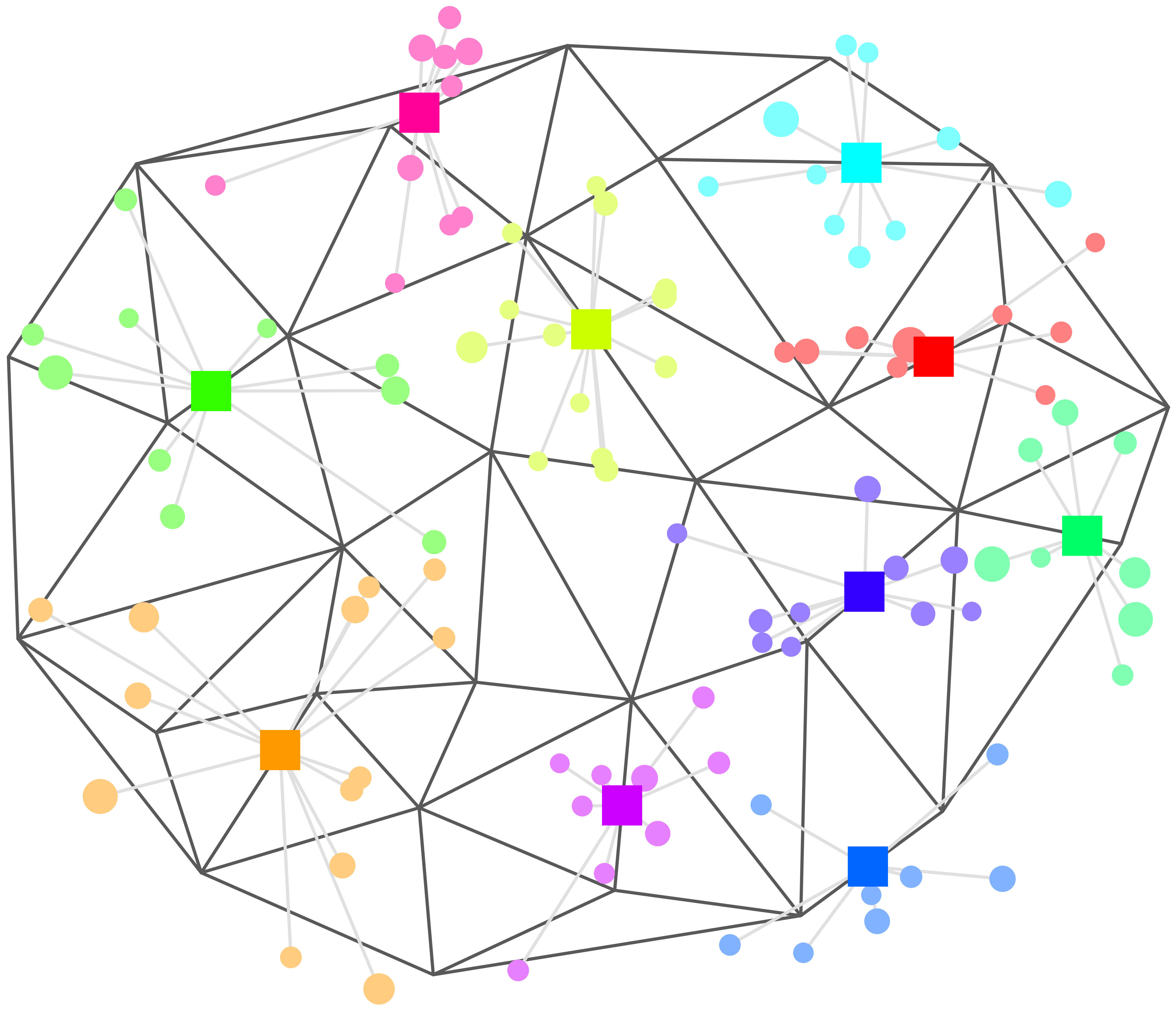


"WCGO 2023" nagrinėtas uždavinys

- **Duota:**
  - Taškų aibė $\{P_1, P_2, \ldots, P_N\}$ su svoriais $\{w_1, w_2, \ldots, w_N\}$
  - "Tinklas" $\mathcal{N}$ (šis "tinklas" yra M segmentų sąjunga)
  - Klasterių skaičius K
- **Uždavinys.** Apibrėžti K klasterių $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K$ su atitinkamais centras $Q_1, Q_2, \ldots, Q_K$ randančių globalų sprendinį uždaviniui

$$\min_{Q_1, Q_2, \ldots Q_K} \sum_{k=1}^{K} \sum_{i \in \mathcal{C}_k} w_i \left\| P_i - Q_k \right\|_2^2 \quad \text{s.t.} \quad Q_k \in \mathcal{N},\ k = 1, \ldots, K$$

- **Pastebėkite tinklo ribojimą klasterių centrams:** $Q_k \in \mathcal{N}$
- Jeigu šio ribojimo nebūtų, turėtume uždavinį kuris literatūroje vadinamas minimum-sum-of-squares-clustering (MSSC), kuriam spręsti naudojamas garsusis **k-means algoritmas**



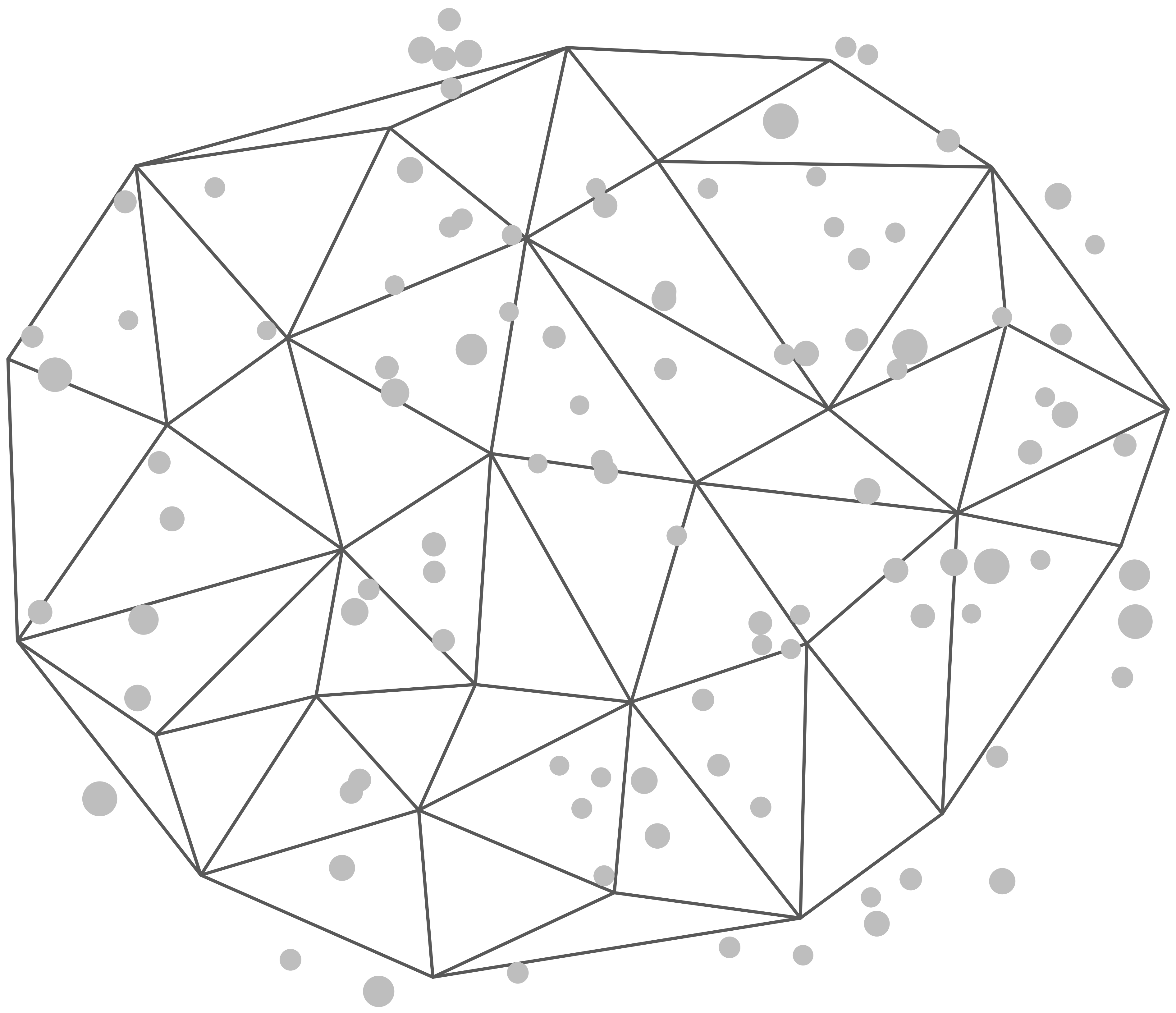
"WCGO 2023" pasiūlytas algoritmas

“WCGO 2023” pasiūlytas algoritmas

- **Net-constrained k-means algoritmas** skirtas rasti lokaliam klasteriavimo uždavinio su tinklo ribojimu sprendiniui. **Šablonas:**

“WCGO 2023” pasiūlytas algoritmas

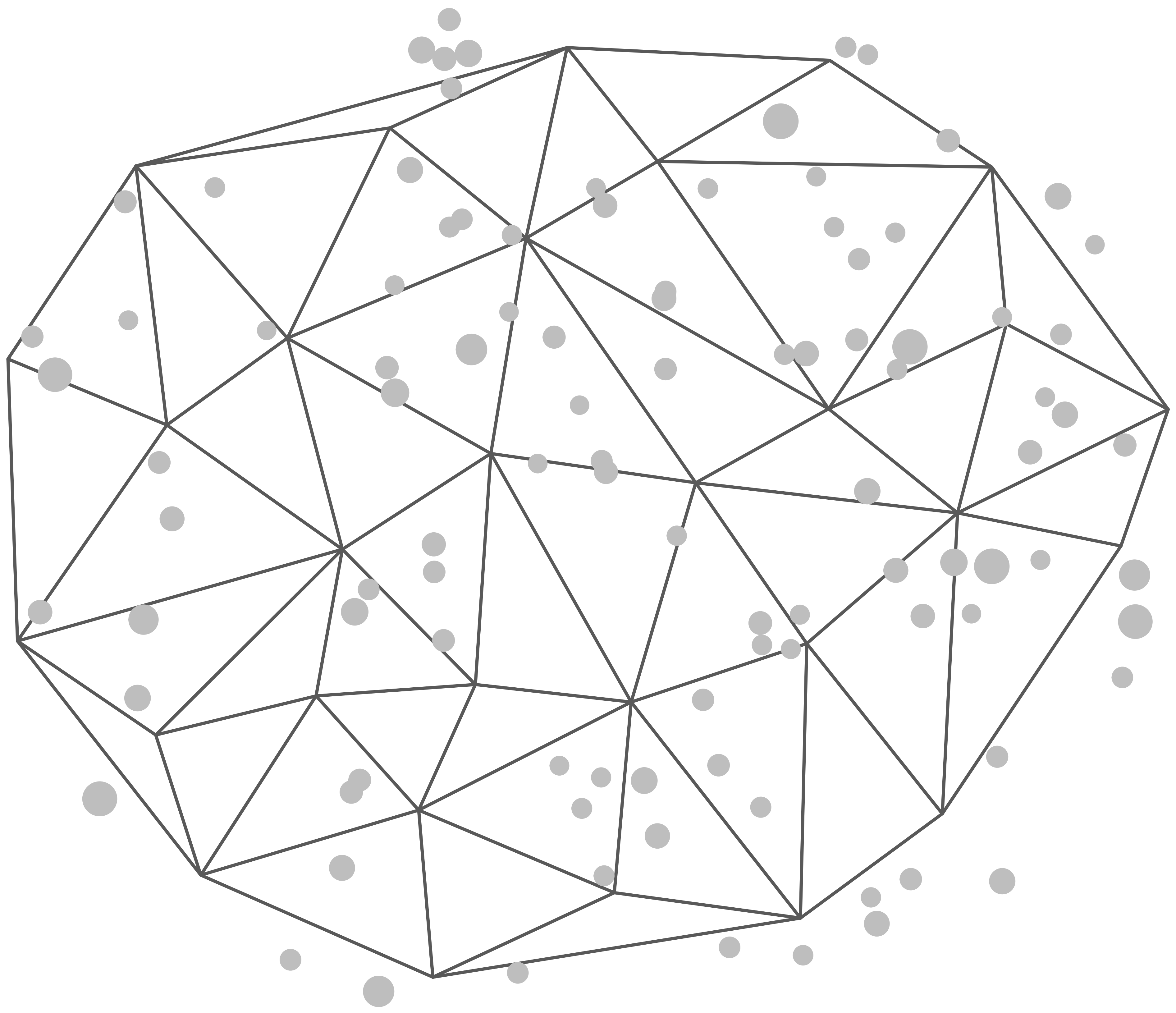
- **Net-constrained k-means algoritmas** skirtas rasti lokaliam klasteriavimo uždavinio su tinklo ribojimu sprendiniui. **Šablonas:**



“WCGO 2023” pasiūlytas algoritmas

- **Net-constrained k-means algoritmas** skirtas rasti lokaliam klasteriavimo uždavinio su tinklo ribojimu sprendiniui. **Šablonas:**

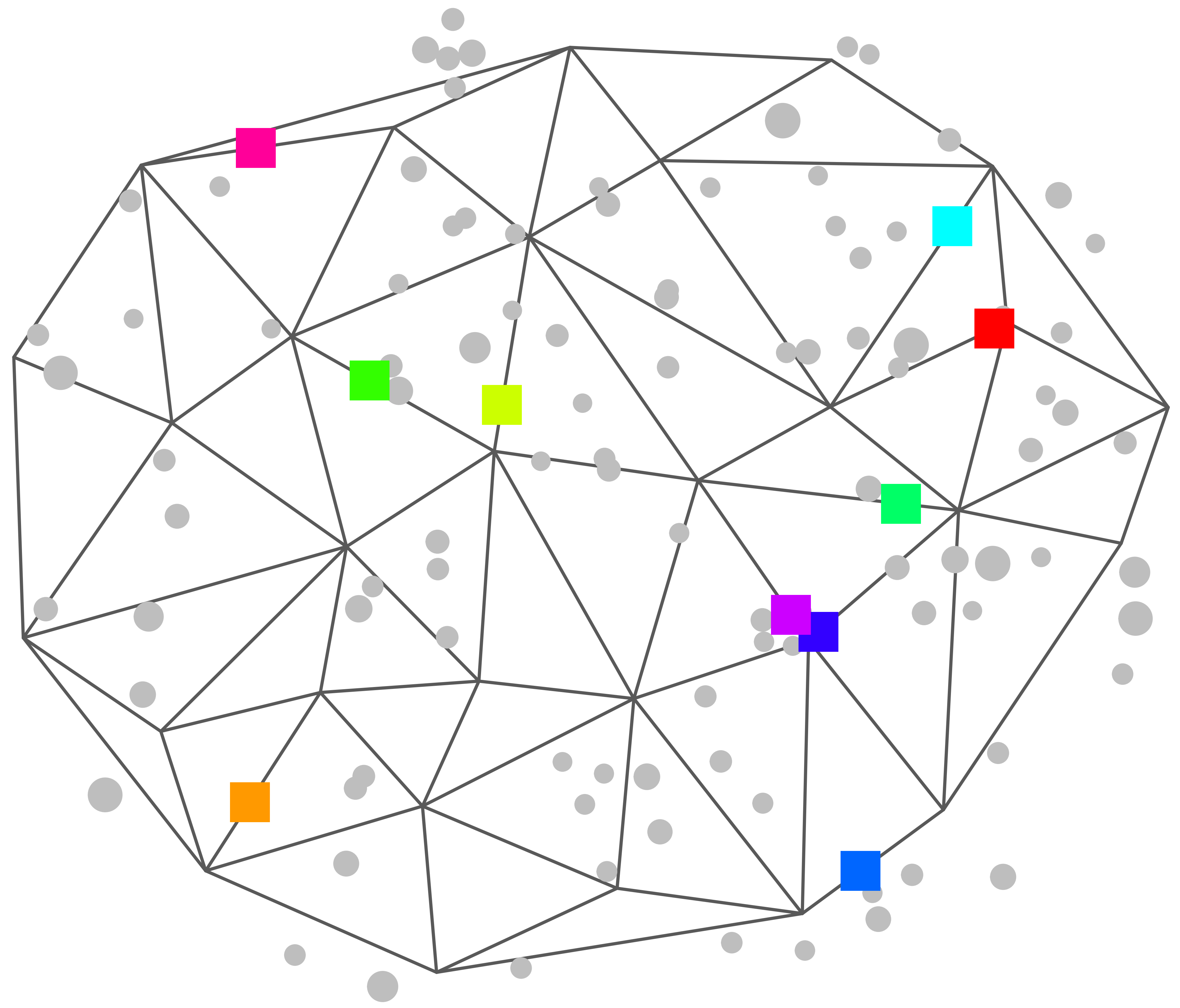
1. Atsitiktinai parenkam pradinius centrus $Q_1, Q_2, \ldots, Q_K$ (kuriuos vėliau tikslinsim)



“WCGO 2023” pasiūlytas algoritmas

- **Net-constrained k-means algoritmas** skirtas rasti lokaliam klasteriavimo uždavinio su tinklo ribojimu sprendiniui. **Šablonas:**

1. Atsitiktinai parenkam pradinius centrus $Q_1, Q_2, \ldots, Q_K$ (kuriuos vėliau tikslinsim)

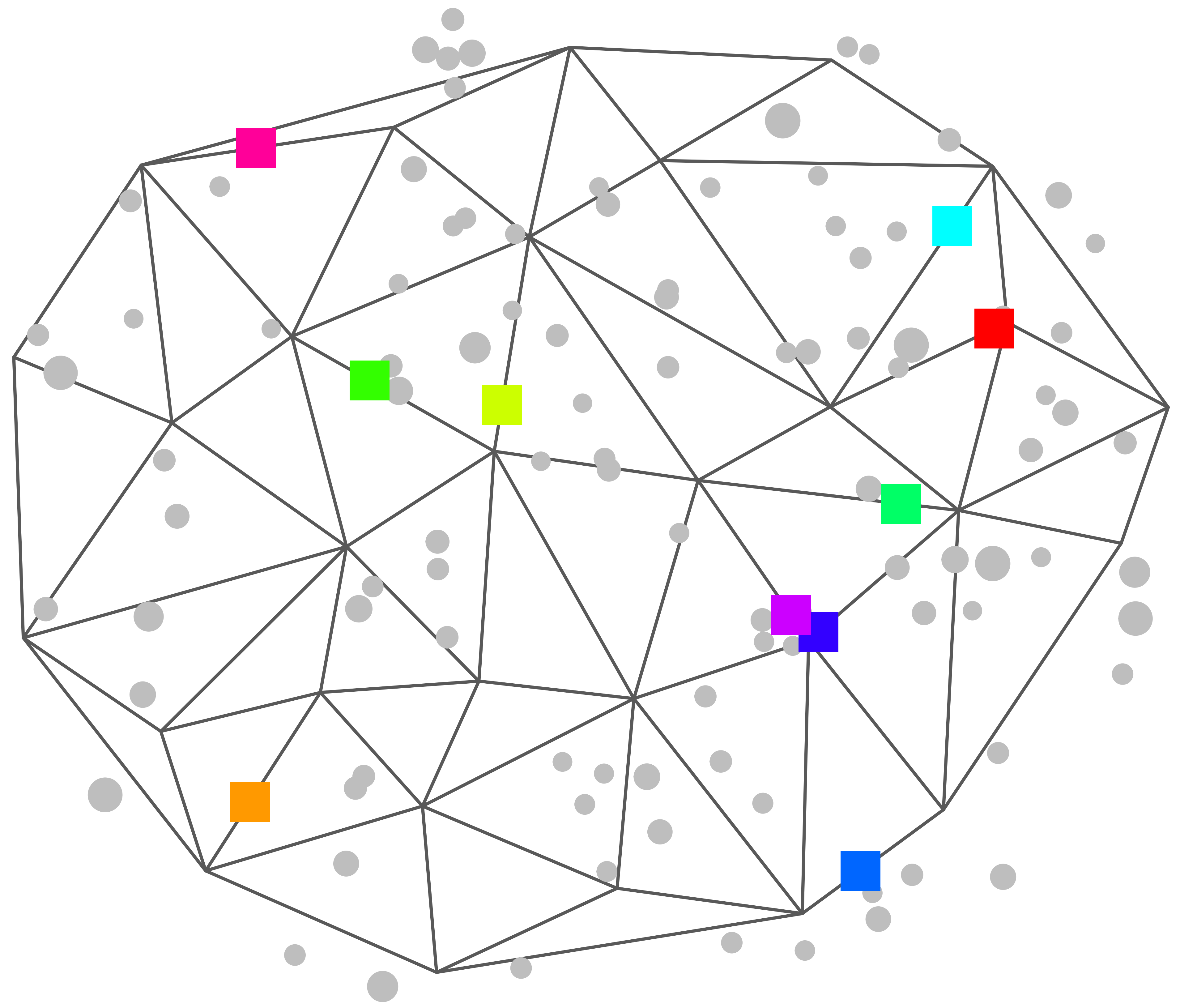


“WCGO 2023” pasiūlytas algoritmas

- **Net-constrained k-means algoritmas** skirtas rasti lokaliam klasteriavimo uždavinio su tinklo ribojimu sprendiniui. **Šablonas:**

1. Atsitiktinai parenkam pradinius centrus $Q_1, Q_2, \ldots, Q_K$ (kuriuos vėliau tikslinsim)

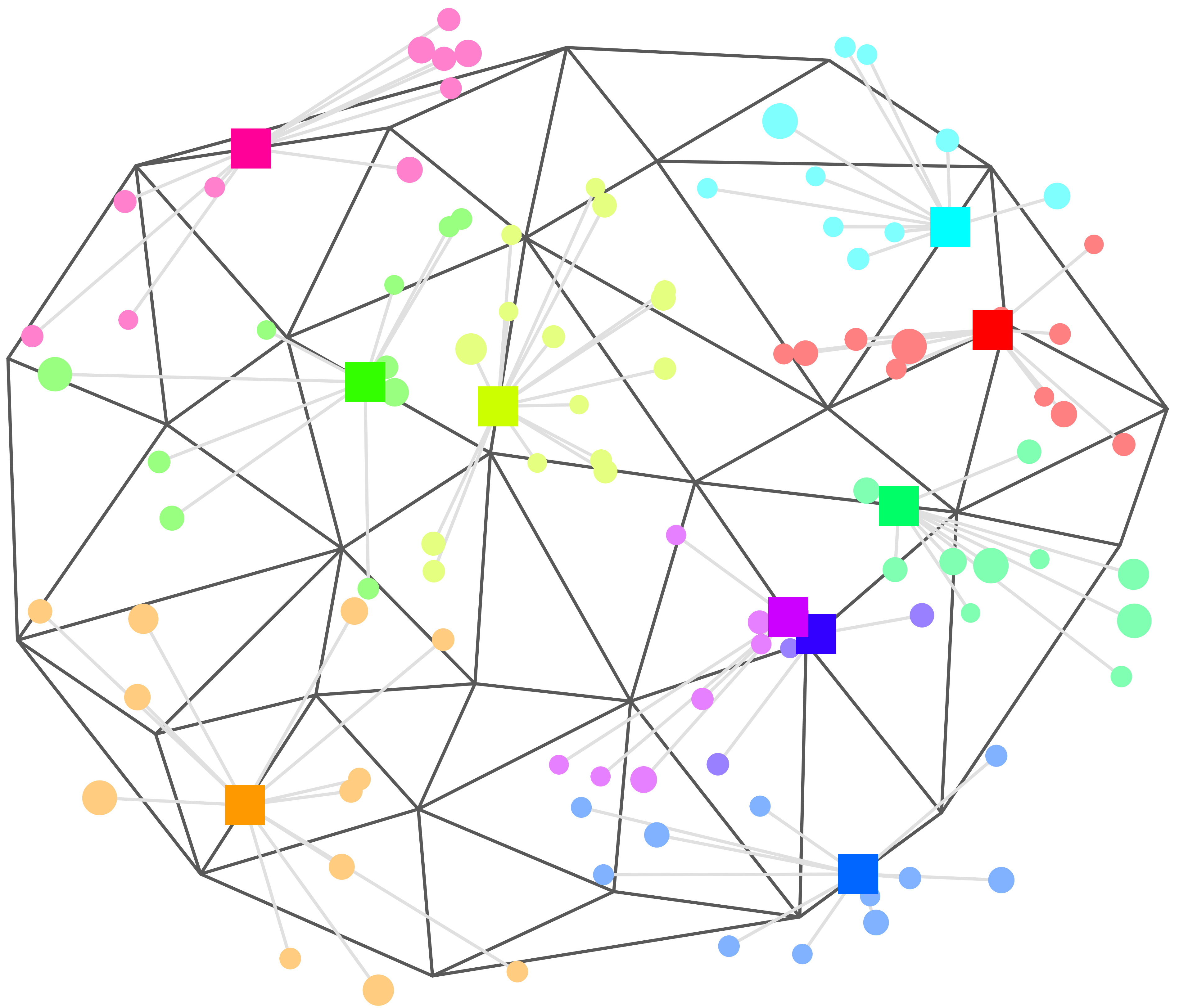
2. **Priskyrimo Žingsnis.** Randam klasterius $\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_K$: kiekvienas taškas P_i yra priskiriamas artimiausiam centrui Q_k (ir atitinkamam klasteriui)



“WCGO 2023” pasiūlytas algoritmas

- **Net-constrained k-means algoritmas** skirtas rasti lokaliam klasteriavimo uždavinio su tinklo ribojimu sprendiniui. **Šablonas:**

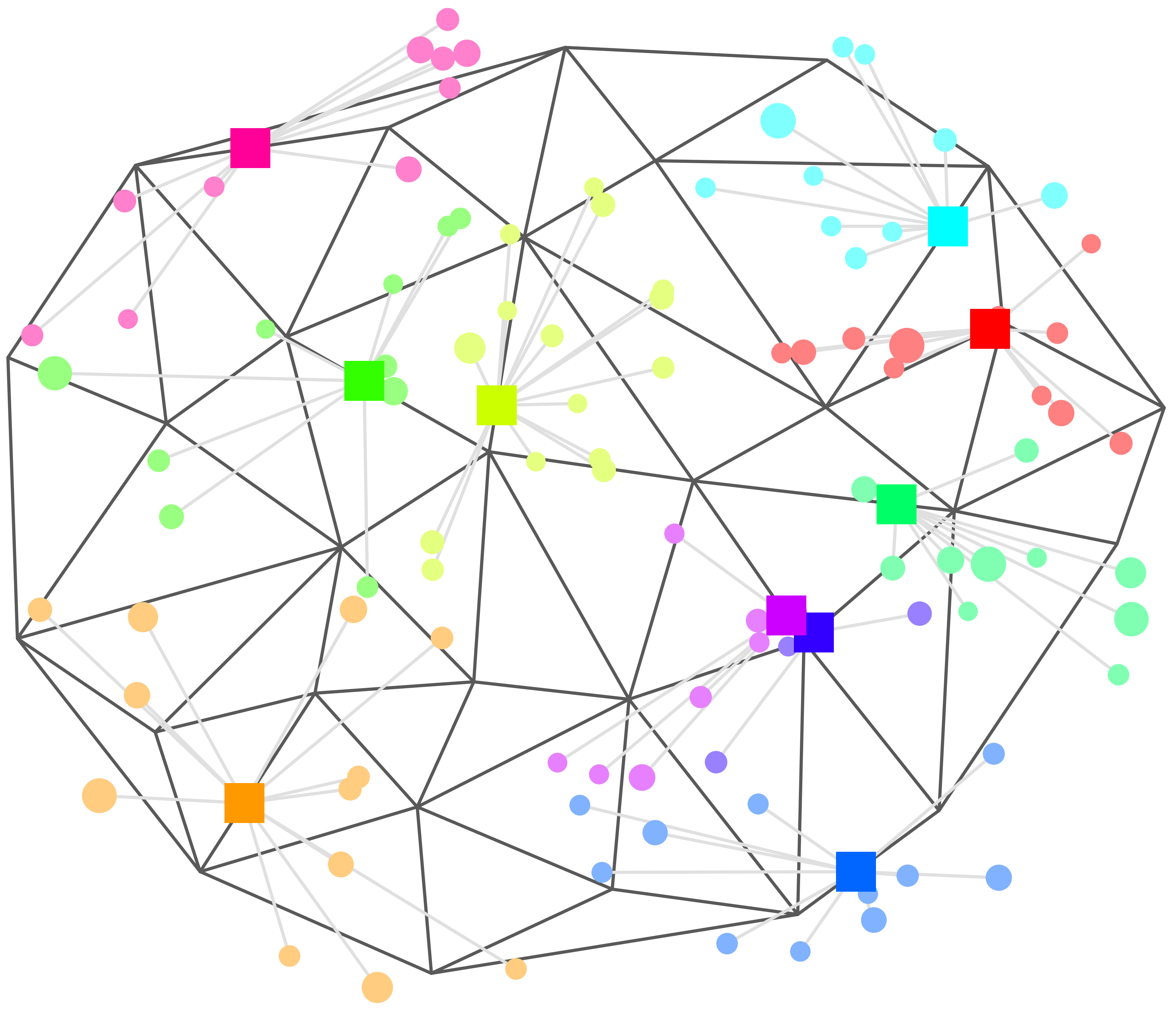
1. Atsitiktinai parenkam pradinius centrus $Q_1, Q_2, \ldots, Q_K$ (kuriuos vėliau tikslinsim)
2. **Priskyrimo Žingsnis.** Randam klasterius $\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_K$: kiekvienas taškas P_i yra priskiriamas artimiausiam centrui Q_k (ir atitinkamam klasteriui)



“WCGO 2023” pasiūlytas algoritmas

- **Net-constrained k-means algoritmas** skirtas rasti lokaliam klasteriavimo uždavinio su tinklo ribojimu sprendiniui. **Šablonas:**

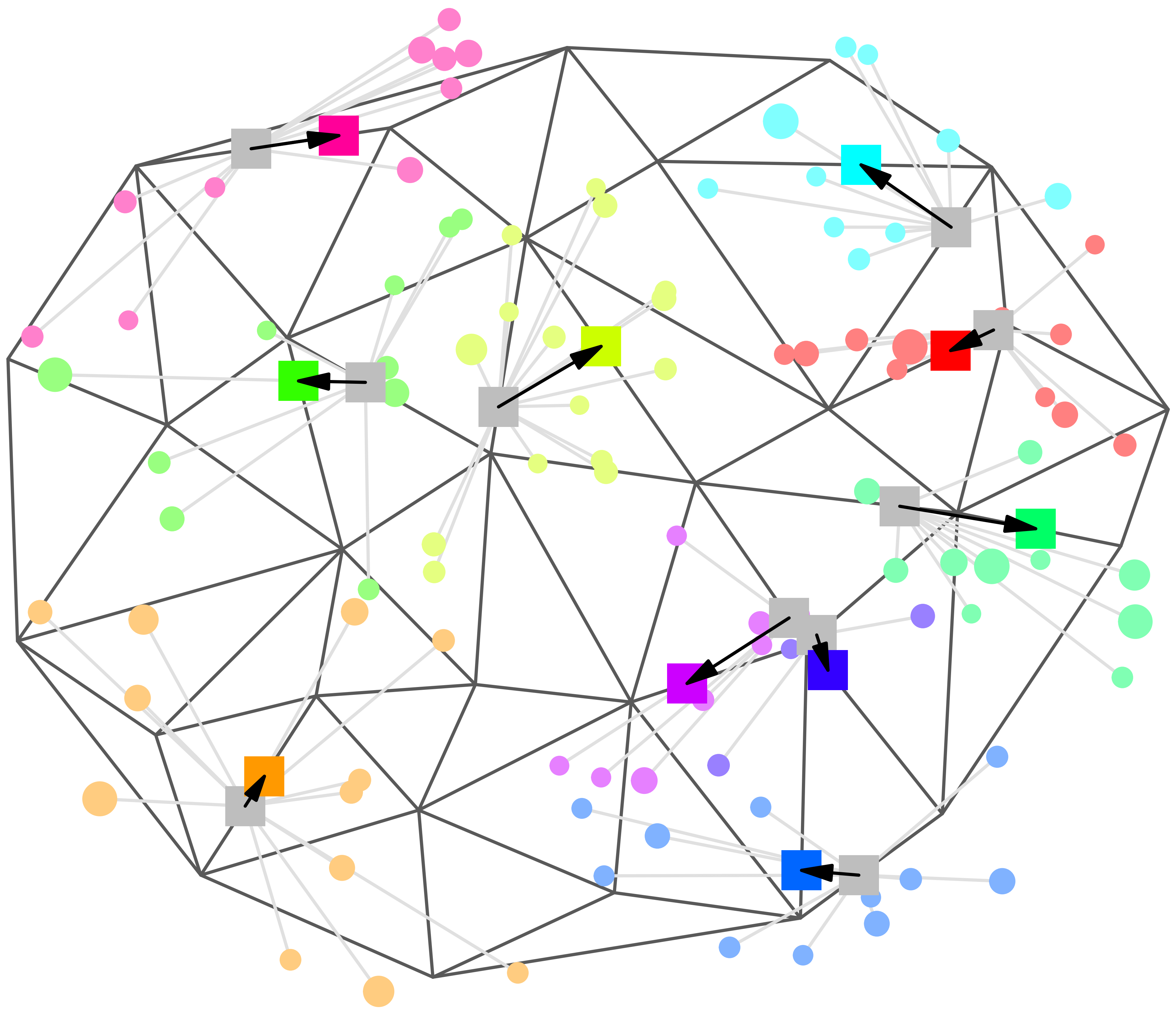
1. Atsitiktinai parenkam pradinius centrus $Q_1, Q_2, \ldots, Q_K$ (kuriuos vėliau tikslinsim)
2. **Priskyrimo Žingsnis.** Randam klasterius $\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_K$: kiekvienas taškas P_i yra priskiriamas artimiausiam centrui Q_k (ir atitinkamam klasteriui)
3. **Centrų Tikslinimo Žingsnis.** Kiekvienam klasteriui $\mathscr{C}_k$ randame centrą Q_k kuris tenkina tinklo ribojimą $Q_k \in \mathscr{N}$ ir minimizuoja vidinius klasterio nuostolius



“WCGO 2023” pasiūlytas algoritmas

- **Net-constrained k-means algoritmas** skirtas rasti lokaliam klasteriavimo uždavinio su tinklo ribojimu sprendiniui. **Šablonas:**

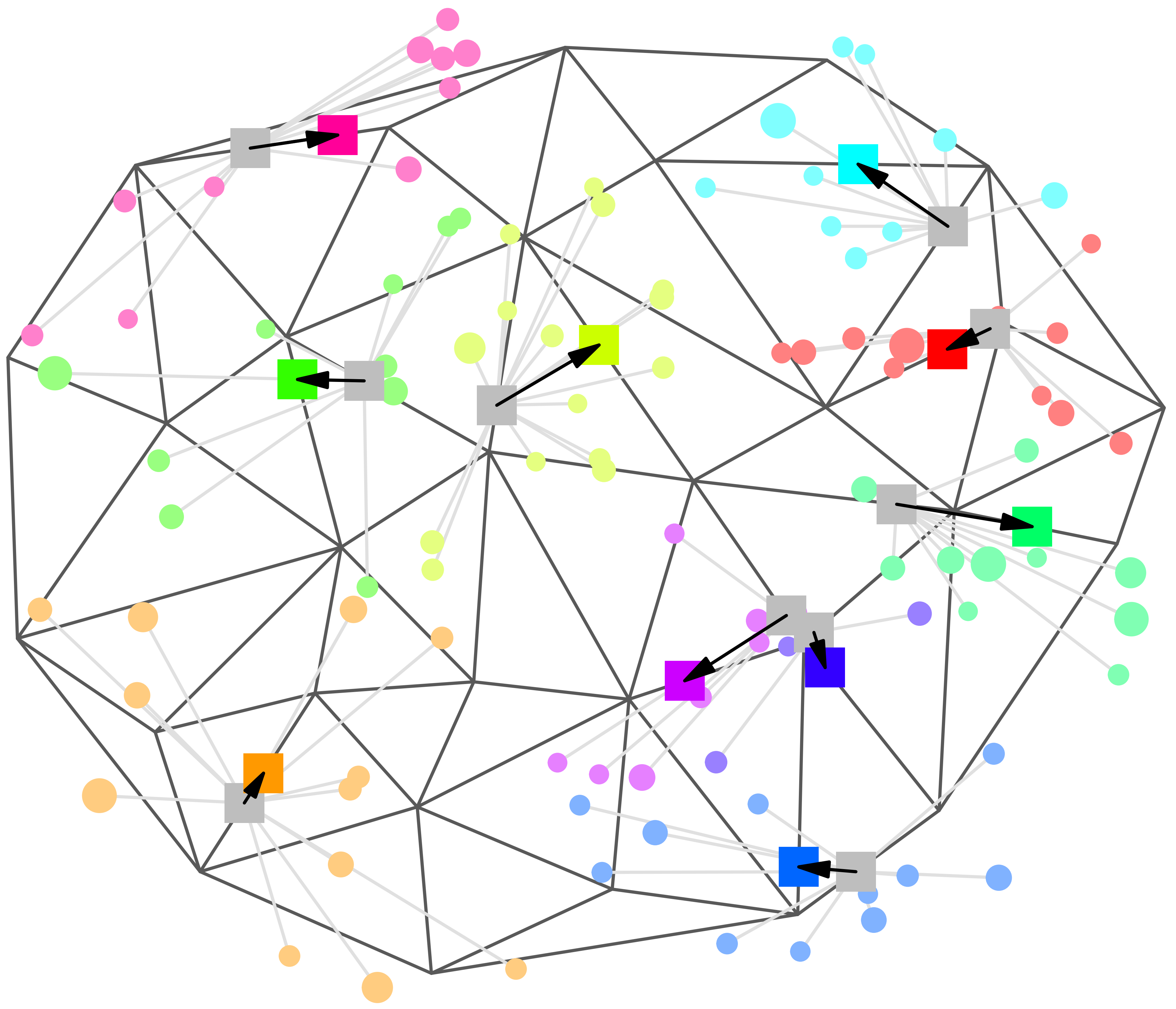
1. Atsitiktinai parenkam pradinius centrus $Q_1, Q_2, \ldots, Q_K$ (kuriuos vėliau tikslinsim)
2. **Priskyrimo Žingsnis.** Randam klasterius $\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_K$: kiekvienas taškas P_i yra priskiriamas artimiausiam centrui Q_k (ir atitinkamam klasteriui)
3. **Centrų Tikslinimo Žingsnis.** Kiekvienam klasteriui $\mathscr{C}_k$ randame centrą Q_k kuris tenkina tinklo ribojimą $Q_k \in \mathscr{N}$ ir minimizuoja vidinius klasterio nuostolius



“WCGO 2023” pasiūlytas algoritmas

- **Net-constrained k-means algoritmas** skirtas rasti lokaliam klasteriavimo uždavinio su tinklo ribojimu sprendiniui. **Šablonas:**

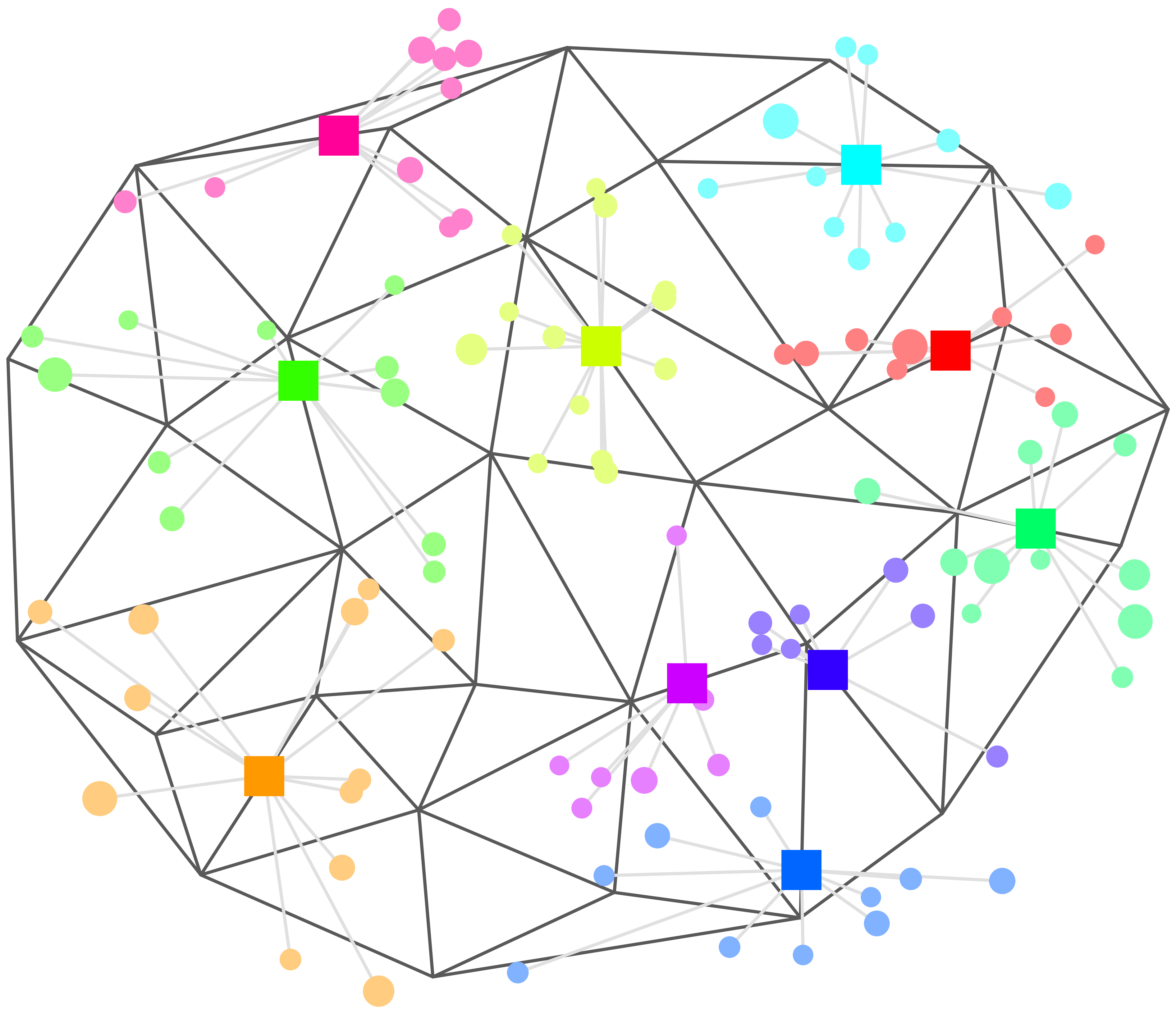
1. Atsitiktinai parenkam pradinius centrus $Q_1, Q_2, \ldots, Q_K$ (kuriuos vėliau tikslinsim)
2. **Priskyrimo Žingsnis.** Randam klasterius $\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_K$: kiekvienas taškas P_i yra priskiriamas artimiausiam centrui Q_k (ir atitinkamam klasteriui)
3. **Centrų Tikslinimo Žingsnis.** Kiekvienam klasteriui $\mathscr{C}_k$ randame centrą Q_k kuris tenkina tinklo ribojimą $Q_k \in \mathscr{N}$ ir minimizuoja vidinius klasterio nuostolius
4. Kartojame **Priskyrimo ir Centrų Tikslinimo Žingsnius** kol klasteriai ir jų centrai stabilizuojasi



“WCGO 2023” pasiūlytas algoritmas

- **Net-constrained k-means algoritmas** skirtas rasti lokaliam klasteriavimo uždavinio su tinklo ribojimu sprendiniui. **Šablonas:**

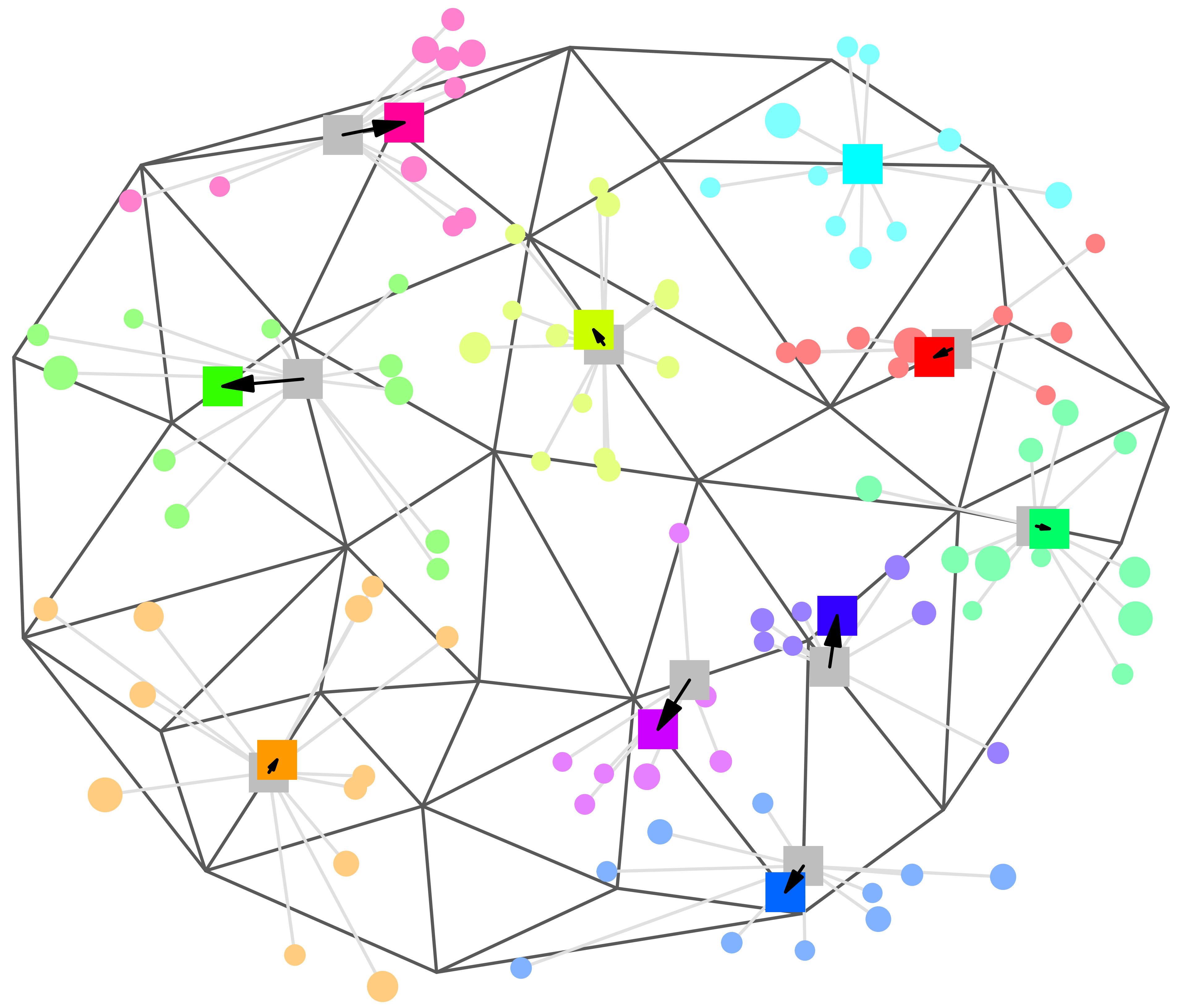
1. Atsitiktinai parenkam pradinius centrus $Q_1, Q_2, \ldots, Q_K$ (kuriuos vėliau tikslinsim)
2. **Priskyrimo Žingsnis.** Randam klasterius $\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_K$: kiekvienas taškas P_i yra priskiriamas artimiausiam centrui Q_k (ir atitinkamam klasteriui)
3. **Centrų Tikslinimo Žingsnis.** Kiekvienam klasteriui $\mathscr{C}_k$ randame centrą Q_k kuris tenkina tinklo ribojimą $Q_k \in \mathscr{N}$ ir minimizuoja vidinius klasterio nuostolius
4. Kartojame **Priskyrimo ir Centrų Tikslinimo Žingsnius** kol klasteriai ir jų centrai stabilizuojasi



“WCGO 2023” pasiūlytas algoritmas

- **Net-constrained k-means algoritmas** skirtas rasti lokaliam klasteriavimo uždavinio su tinklo ribojimu sprendiniui. **Šablonas:**

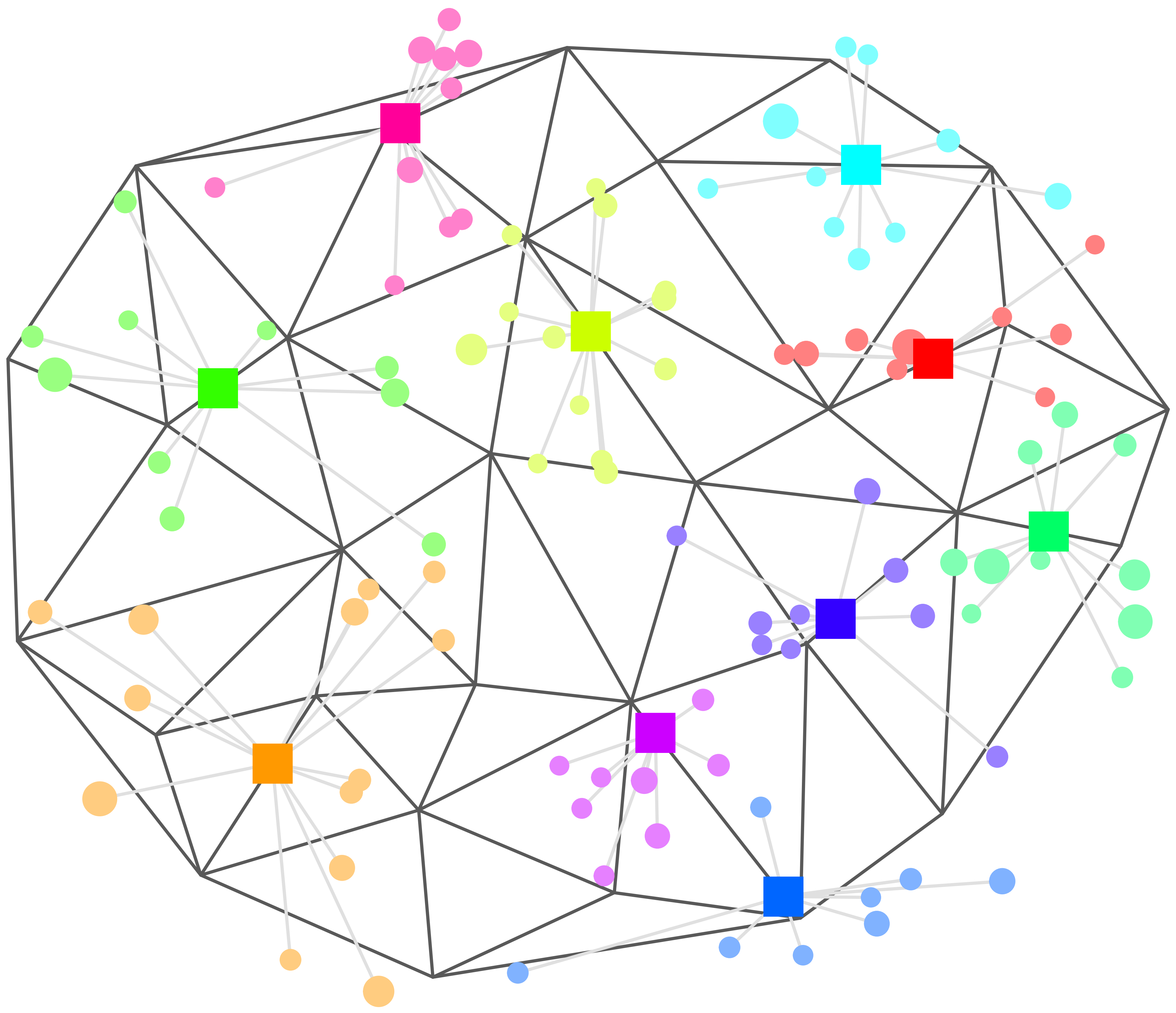
1. Atsitiktinai parenkam pradinius centrus $Q_1, Q_2, \ldots, Q_K$ (kuriuos vėliau tikslinsim)
2. **Priskyrimo Žingsnis.** Randam klasterius $\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_K$: kiekvienas taškas P_i yra priskiriamas artimiausiam centrui Q_k (ir atitinkamam klasteriui)
3. **Centrų Tikslinimo Žingsnis.** Kiekvienam klasteriui $\mathscr{C}_k$ randame centrą Q_k kuris tenkina tinklo ribojimą $Q_k \in \mathscr{N}$ ir minimizuoja vidinius klasterio nuostolius
4. Kartojame **Priskyrimo ir Centrų Tikslinimo Žingsnius** kol klasteriai ir jų centrai stabilizuojasi



"WCGO 2023" pasiūlytas algoritmas

- **Net-constrained k-means algoritmas** skirtas rasti lokaliam klasteriavimo uždavinio su tinklo ribojimu sprendiniui. **Šablonas:**

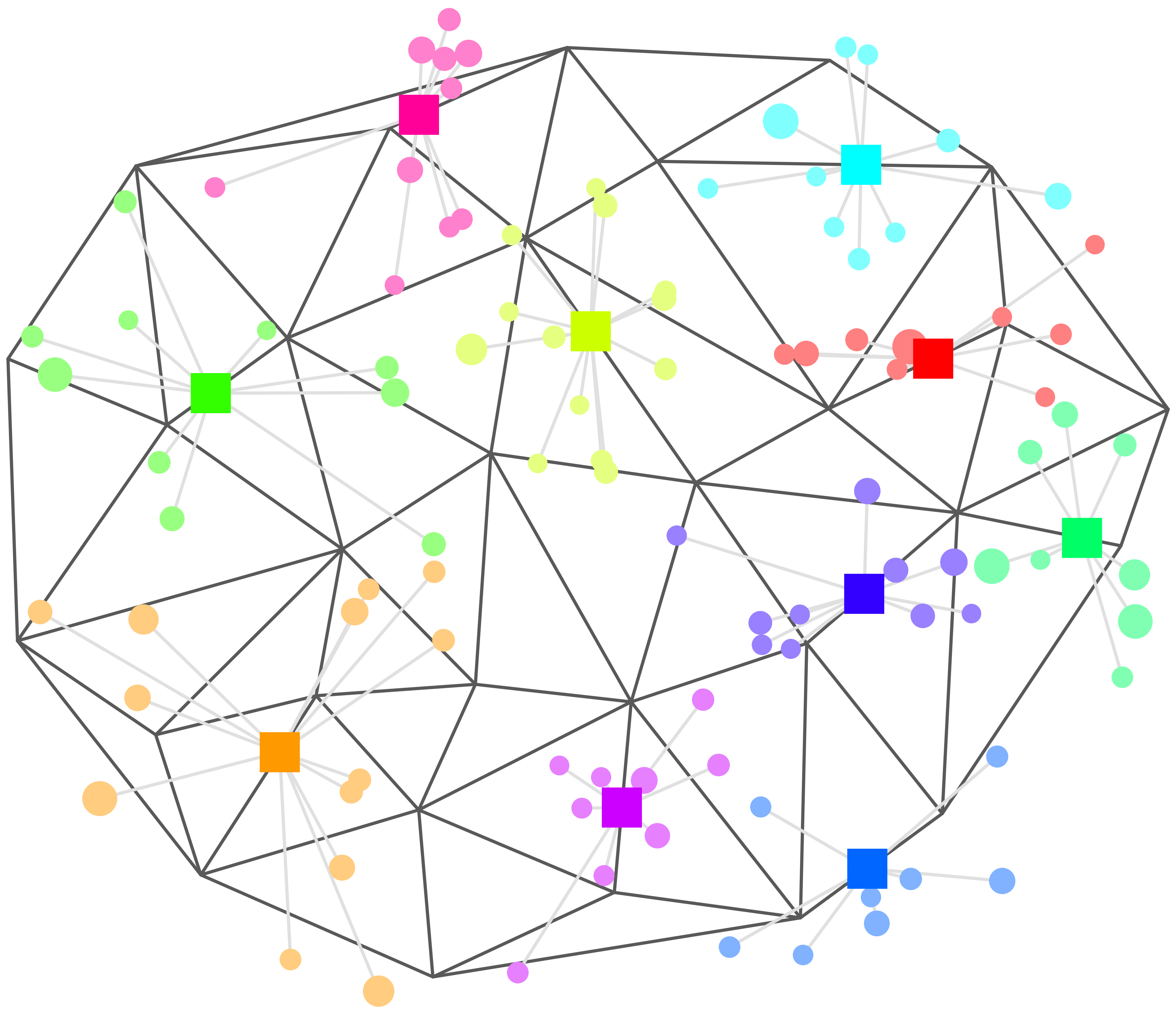
1. Atsitiktinai parenkam pradinius centrus $Q_1, Q_2, \ldots, Q_K$ (kuriuos vėliau tikslinsim)
2. **Priskyrimo Žingsnis.** Randam klasterius $\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_K$: kiekvienas taškas P_i yra priskiriamas artimiausiam centrui Q_k (ir atitinkamam klasteriui)
3. **Centrų Tikslinimo Žingsnis.** Kiekvienam klasteriui $\mathscr{C}_k$ randame centrą Q_k kuris tenkina tinklo ribojimą $Q_k \in \mathscr{N}$ ir minimizuoja vidinius klasterio nuostolius
4. Kartojame **Priskyrimo ir Centrų Tikslinimo Žingsnius** kol klasteriai ir jų centrai stabilizuojasi



“WCGO 2023” pasiūlytas algoritmas

- **Net-constrained k-means algoritmas** skirtas rasti lokaliam klasteriavimo uždavinio su tinklo ribojimu sprendiniui. **Šablonas:**

1. Atsitiktinai parenkam pradinius centrus $Q_1, Q_2, \ldots, Q_K$ (kuriuos vėliau tikslinsim)
2. **Priskyrimo Žingsnis.** Randam klasterius $\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_K$: kiekvienas taškas P_i yra priskiriamas artimiausiam centrui Q_k (ir atitinkamam klasteriui)
3. **Centrų Tikslinimo Žingsnis.** Kiekvienam klasteriui $\mathscr{C}_k$ randame centrą Q_k kuris tenkina tinklo ribojimą $Q_k \in \mathscr{N}$ ir minimizuoja vidinius klasterio nuostolius
4. Kartojame **Priskyrimo ir Centrų Tikslinimo Žingsnius** kol klasteriai ir jų centrai stabilizuojasi



Bandymai rasti globalų sprendinį su “gurobi” įrankiu

Bandymai rasti globalų sprendinį su “gurobi” įrankiu

- Suformulavom uždavinį “gurobi” kalba

Bandymai rasti globalų sprendinį su “gurobi” įrankiu

- Suformulavom uždavinį “gurobi” kalba
- Paleidom “kompiuterinį” eksperimentą:

Bandymai rasti globalų sprendinį su “gurobi” įrankiu

- Suformulavom uždavinį “gurobi” kalba
- Paleidom “kompiuterinį” eksperimentą:

for $N \in \{10, 15, \ldots, 45, 50\}$ # taškų skaičius

Bandymai rasti globalų sprendinį su “gurobi” įrankiu

- Suformulavom uždavinį “gurobi” kalba
- Paleidom “kompiuterinį” eksperimentą:

for $N \in \{10, 15, \ldots, 45, 50\}$ # taškų skaičius

for $M \in \{20, 30, \ldots, 90, 100\}$ # tinklo briaunų skaičius

Bandymai rasti globalų sprendinį su “gurobi” įrankiu

- Suformulavom uždavinį “gurobi” kalba
- Paleidom “kompiuterinį” eksperimentą:

for $N \in \{10, 15, \ldots, 45, 50\}$ # taškų skaičius

    for $M \in \{20, 30, \ldots, 90, 100\}$ # tinklo briaunų skaičius

        for $K \in \{2, 3, 4, 5, 6, 7, 8, 9, 10\}$ # klasterių skaičius

Bandymai rasti globalų sprendinį su “gurobi” įrankiu

- Suformulavom uždavinį “gurobi” kalba
- Paleidom “kompiuterinį” eksperimentą:

for $N \in \{10, 15, \ldots, 45, 50\}$ # taškų skaičius

  for $M \in \{20, 30, \ldots, 90, 100\}$ # tinklo briaunų skaičius

    for $K \in \{2, 3, 4, 5, 6, 7, 8, 9, 10\}$ # klasterių skaičius

      solve_MIQCP_program_with_gurobi(N, M, K**)**

Bandymai rasti globalų sprendinį su “gurobi” įrankiu

- Suformulavom uždavinį “gurobi” kalba
- Paleidom “kompiuterinį” eksperimentą:

for $N \in \{10, 15, \ldots, 45, 50\}$ # taškų skaičius

  for $M \in \{20, 30, \ldots, 90, 100\}$ # tinklo briaunų skaičius

    for $K \in \{2, 3, 4, 5, 6, 7, 8, 9, 10\}$ # klasterių skaičius

      solve_MIQCP_program_with_gurobi(N, M, K**)**

- Rezultatai ne itin įspūdingi

Bandymai rasti globalų sprendinį su “gurobi” įrankiu

- Suformulavom uždavinį “gurobi” kalba
- Paleidom “kompiuterinį” eksperimentą:

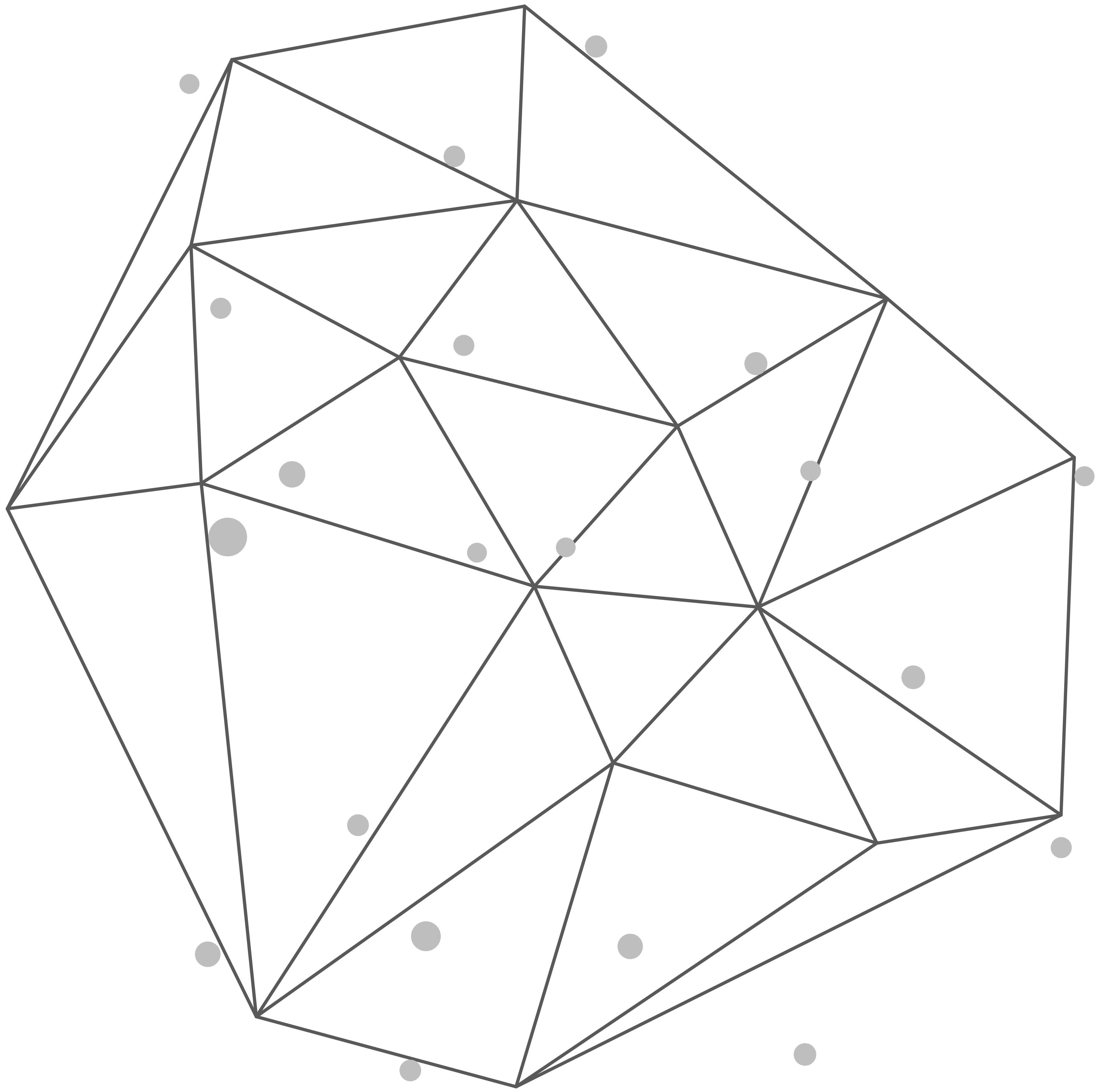
for $N \in \{10, 15, \ldots, 45, 50\}$ # taškų skaičius

for $M \in \{20, 30, \ldots, 90, 100\}$ # tinklo briaunų skaičius

for $K \in \{2, 3, 4, 5, 6, 7, 8, 9, 10\}$ # klasterių skaičius

solve_MIQCP_program_with_gurobi(N, M, K**)**

- Rezultatai ne itin įspūdingi



Bandymai rasti globalų sprendinį su “gurobi” įrankiu

- Suformulavom uždavinį “gurobi” kalba
- Paleidom “kompiuterinį” eksperimentą:

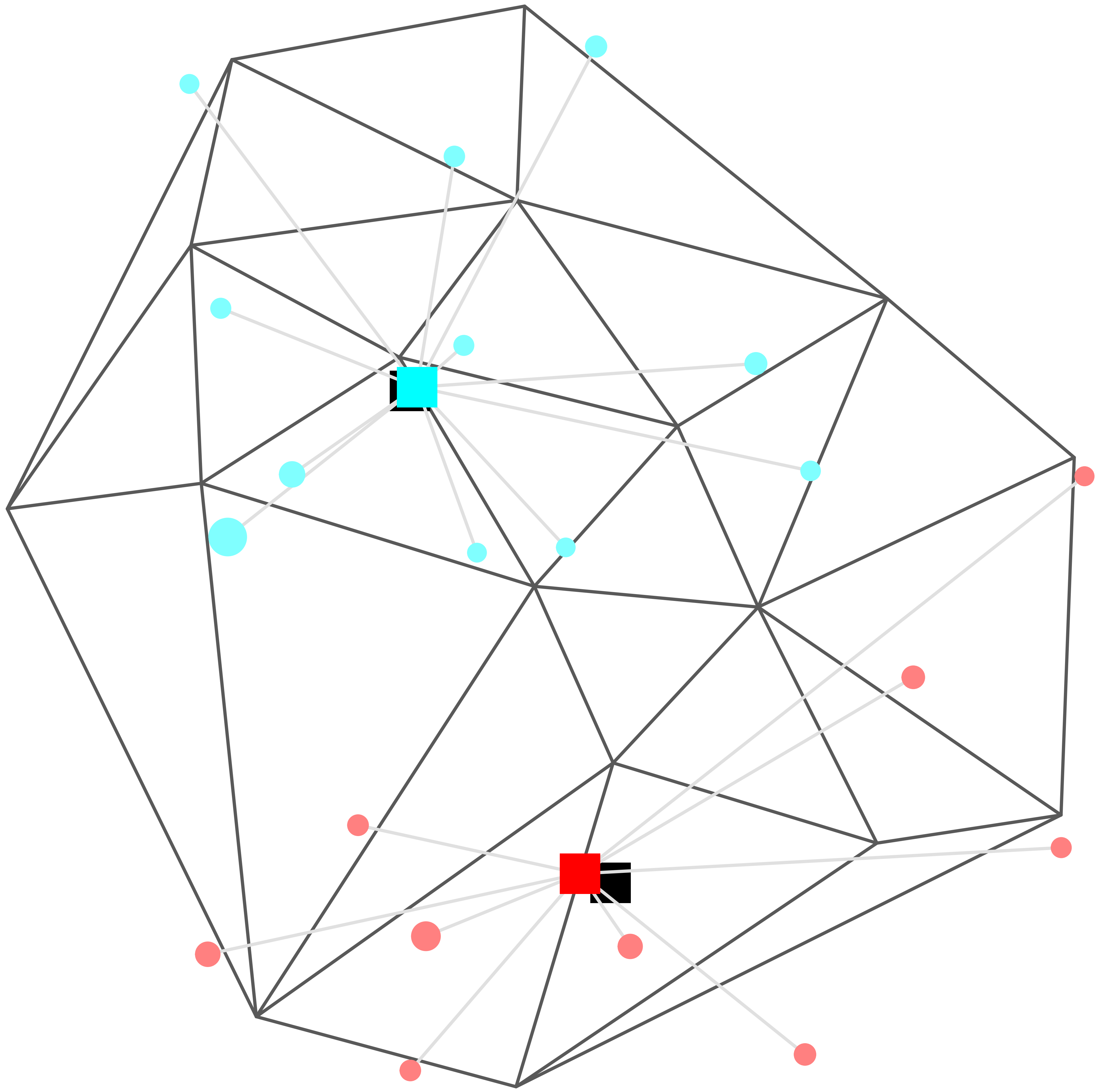
for $N \in \{10, 15, \ldots, 45, 50\}$ # taškų skaičius

    for $M \in \{20, 30, \ldots, 90, 100\}$ # tinklo briaunų skaičius

        for $K \in \{2, 3, 4, 5, 6, 7, 8, 9, 10\}$ # klasterių skaičius

            solve_MIQCP_program_with_gurobi(N, M, K)

- Rezultatai ne itin įspūdingi



Bandymai rasti globalų sprendinį su “gurobi” įrankiu

- Suformulavom uždavinį “gurobi” kalba
- Paleidom “kompiuterinį” eksperimentą:

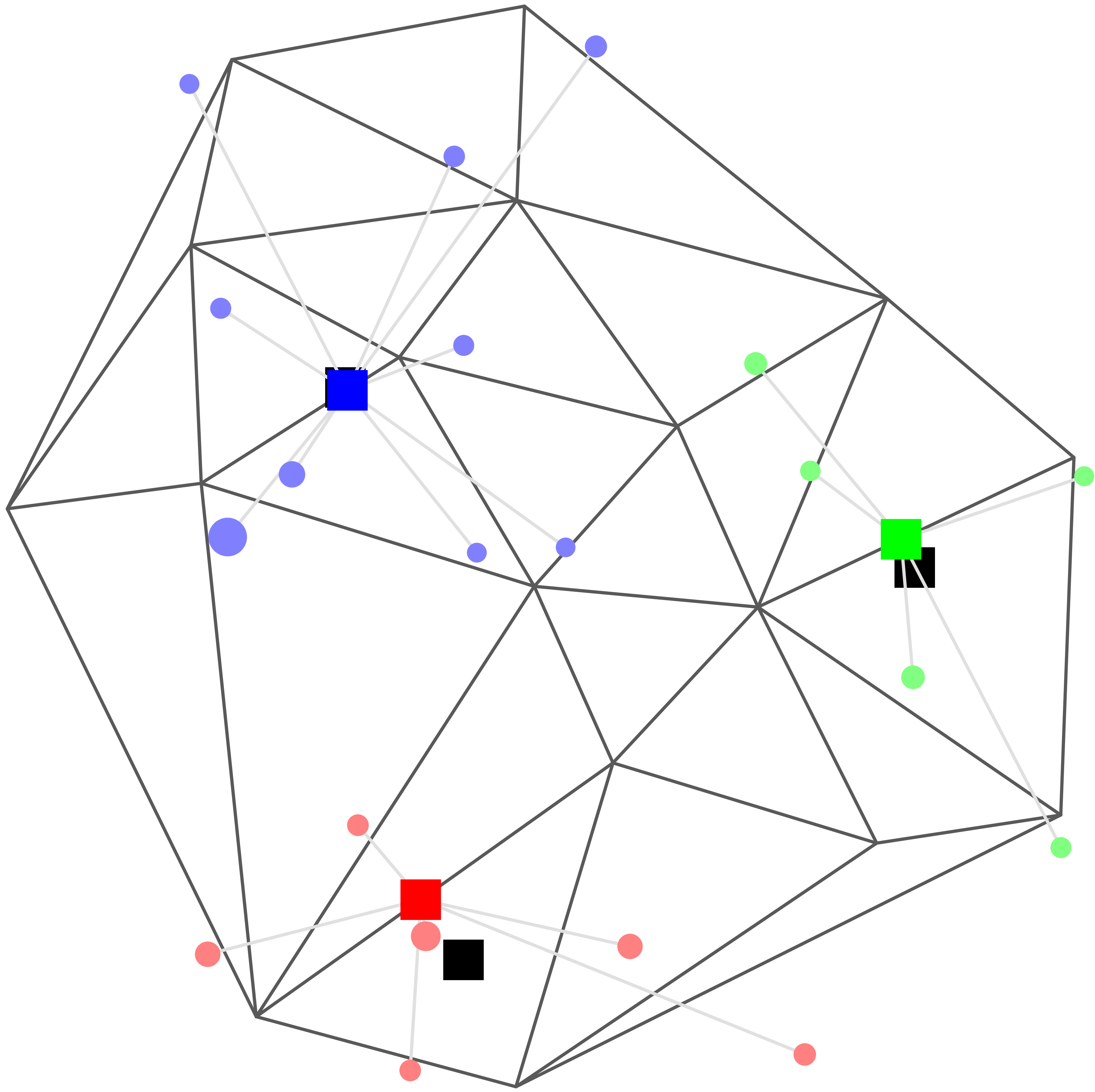
for $N \in \{10, 15, \ldots, 45, 50\}$ # taškų skaičius

for $M \in \{20, 30, \ldots, 90, 100\}$ # tinklo briaunų skaičius

for $K \in \{2, 3, 4, 5, 6, 7, 8, 9, 10\}$ # klasterių skaičius

solve_MIQCP_program_with_gurobi(N, M, K)

- Rezultatai ne itin įspūdingi



Bandymai rasti globalų sprendinį su “gurobi” įrankiu

- Suformulavom uždavinį “gurobi” kalba
- Paleidom “kompiuterinį” eksperimentą:

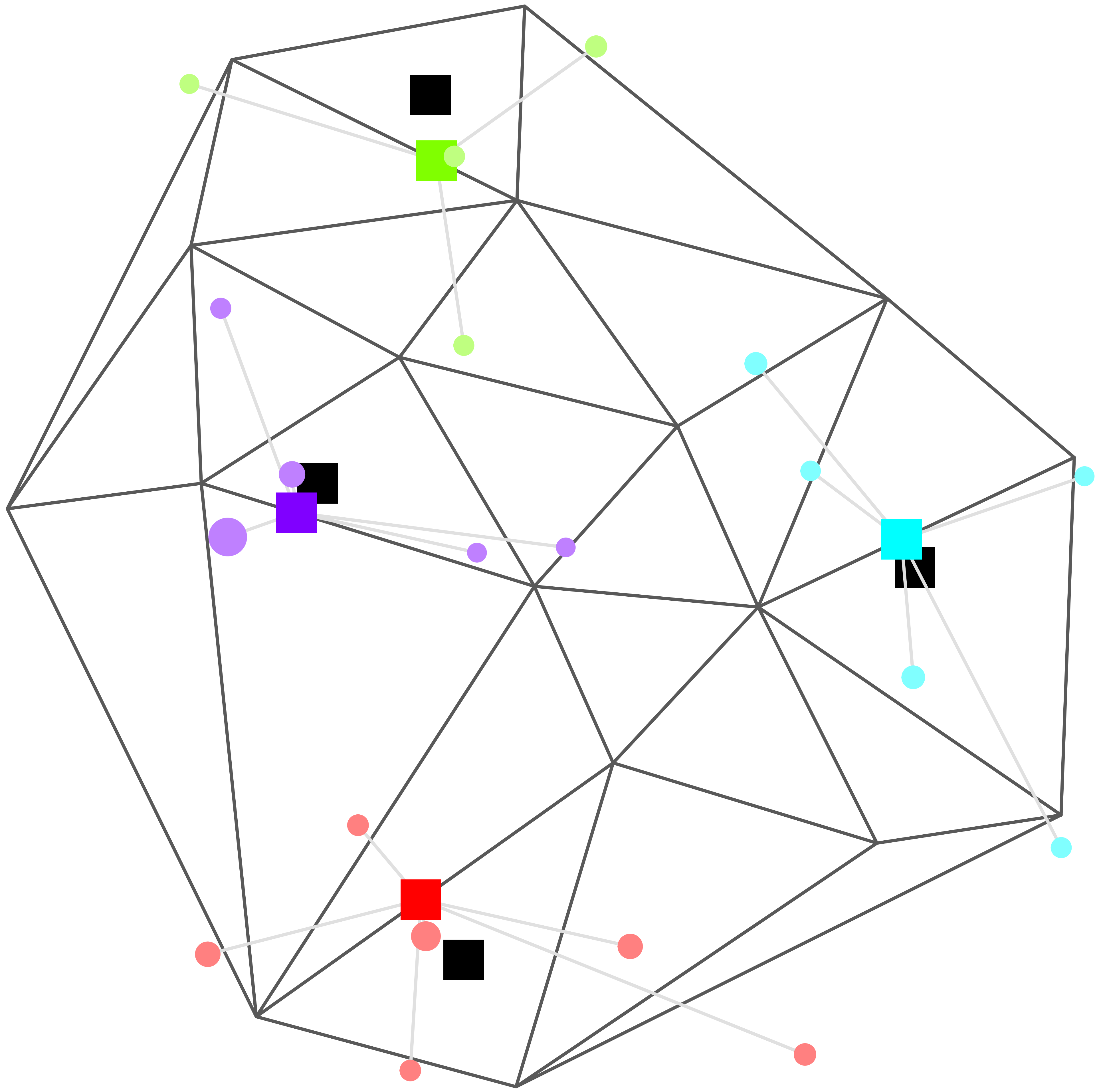
for $N \in \{10, 15, \ldots, 45, 50\}$ # taškų skaičius

for $M \in \{20, 30, \ldots, 90, 100\}$ # tinklo briaunų skaičius

for $K \in \{2, 3, 4, 5, 6, 7, 8, 9, 10\}$ # klasterių skaičius

solve_MIQCP_program_with_gurobi(N, M, K**)**

- Rezultatai ne itin įspūdingi



Laiko modelis “gurobi” įrankiui

Laiko modelis “gurobi” įrankiui

- Iš atlikto eksperimento laiko statistikos sudarėm modelį, kuris prognozuoja kiek “gurobis” užtruks išspręsti tam tikrą uždavinį

Laiko modelis “gurobi” įrankiui

- Iš atlikto eksperimento laiko statistikos sudarėm modelį, kuris prognozuoja kiek “gurobis” užtruks išspręsti tam tikrą uždavinį
- Laiko modelis (sekundėmis) ir mažiausių kvadratų metodu įvertinti koeficientai:

Laiko modelis “gurobi” įrankiui

- Iš atlikto eksperimento laiko statistikos sudarėm modelį, kuris prognozuoja kiek “gurobis” užtruks išspręsti tam tikrą uždavinį
- Laiko modelis (sekundėmis) ir mažiausių kvadratų metodu įvertinti koeficientai:

$$T(K, N, M) \approx 0.0044 \cdot 1.92^{K} \cdot 1.079^{K \times N} \cdot 1.028^{M}$$

Laiko modelis “gurobi” įrankiui

- Iš atlikto eksperimento laiko statistikos sudarėm modelį, kuris prognozuoja kiek “gurobis” užtruks išspręsti tam tikrą uždavinį
- Laiko modelis (sekundėmis) ir mažiausių kvadratų metodu įvertinti koeficientai:

$$T(K, N, M) \approx 0.0044 \cdot 1.92^{K} \cdot 1.079^{K \times N} \cdot 1.028^{M}$$

  - K - klasterių skaičius

Laiko modelis “gurobi” įrankiui

- Iš atlikto eksperimento laiko statistikos sudarėm modelį, kuris prognozuoja kiek “gurobis” užtruks išspręsti tam tikrą uždavinį
- Laiko modelis (sekundėmis) ir mažiausių kvadratų metodu įvertinti koeficientai:

$$T(K, N, M) \approx 0.0044 \cdot 1.92^{K} \cdot 1.079^{K \times N} \cdot 1.028^{M}$$

  - K - klasterių skaičius
  - N - (klasteriuojamų) taškų skaičius

Laiko modelis “gurobi” įrankiui

- Iš atlikto eksperimento laiko statistikos sudarėm modelį, kuris prognozuoja kiek “gurobis” užtruks išspręsti tam tikrą uždavinį
- Laiko modelis (sekundėmis) ir mažiausių kvadratų metodu įvertinti koeficientai:

$$T(K, N, M) \approx 0.0044 \cdot 1.92^{K} \cdot 1.079^{K \times N} \cdot 1.028^{M}$$

  - K - klasterių skaičius
  - N - (klasteriuojamų) taškų skaičius
  - M - tinklo sudėtingumas (segmentų skaičius tinkle)

Laiko modelis “gurobi” įrankiui

- Iš atlikto eksperimento laiko statistikos sudarėm modelį, kuris prognozuoja kiek “gurobis” užtruks išspręsti tam tikrą uždavinį
- Laiko modelis (sekundėmis) ir mažiausių kvadratų metodu įvertinti koeficientai:

$$T(K, N, M) \approx 0.0044 \cdot 1.92^{K} \cdot 1.079^{K \times N} \cdot 1.028^{M}$$

  - K - klasterių skaičius
  - N - (klasteriuojamų) taškų skaičius
  - M - tinklo sudėtingumas (segmentų skaičius tinkle)

$$T(K, N, M) \approx 2^{-c} \cdot 2^{\frac{19}{20}K} \cdot 2^{\frac{K \times N}{9}} \cdot 2^{\frac{M}{25}}, \quad c \approx 7.84$$

Laiko modelis “gurobi” įrankiui

- Iš atlikto eksperimento laiko statistikos sudarėm modelį, kuris prognozuoja kiek “gurobis” užtruks išspręsti tam tikrą uždavinį
- Laiko modelis (sekundėmis) ir mažiausių kvadratų metodu įvertinti koeficientai:

$$T(K, N, M) \approx 0.0044 \cdot 1.92^{K} \cdot 1.079^{K \times N} \cdot 1.028^{M}$$

  - K - klasterių skaičius
  - N - (klasteriuojamų) taškų skaičius
  - M - tinklo sudėtingumas (segmentų skaičius tinkle)

$$T(K, N, M) \approx 2^{-c} \cdot 2^{\frac{19}{20}K} \cdot 2^{\frac{K \times N}{9}} \cdot 2^{\frac{M}{25}}, \quad c \approx 7.84$$

- Nedidelis komentaras: jeigu tinklo ribojimo centrams nėra, laiko prognozės modelyje naudoti $M = 0$

Laiko modelis “gurobi” įrankiui

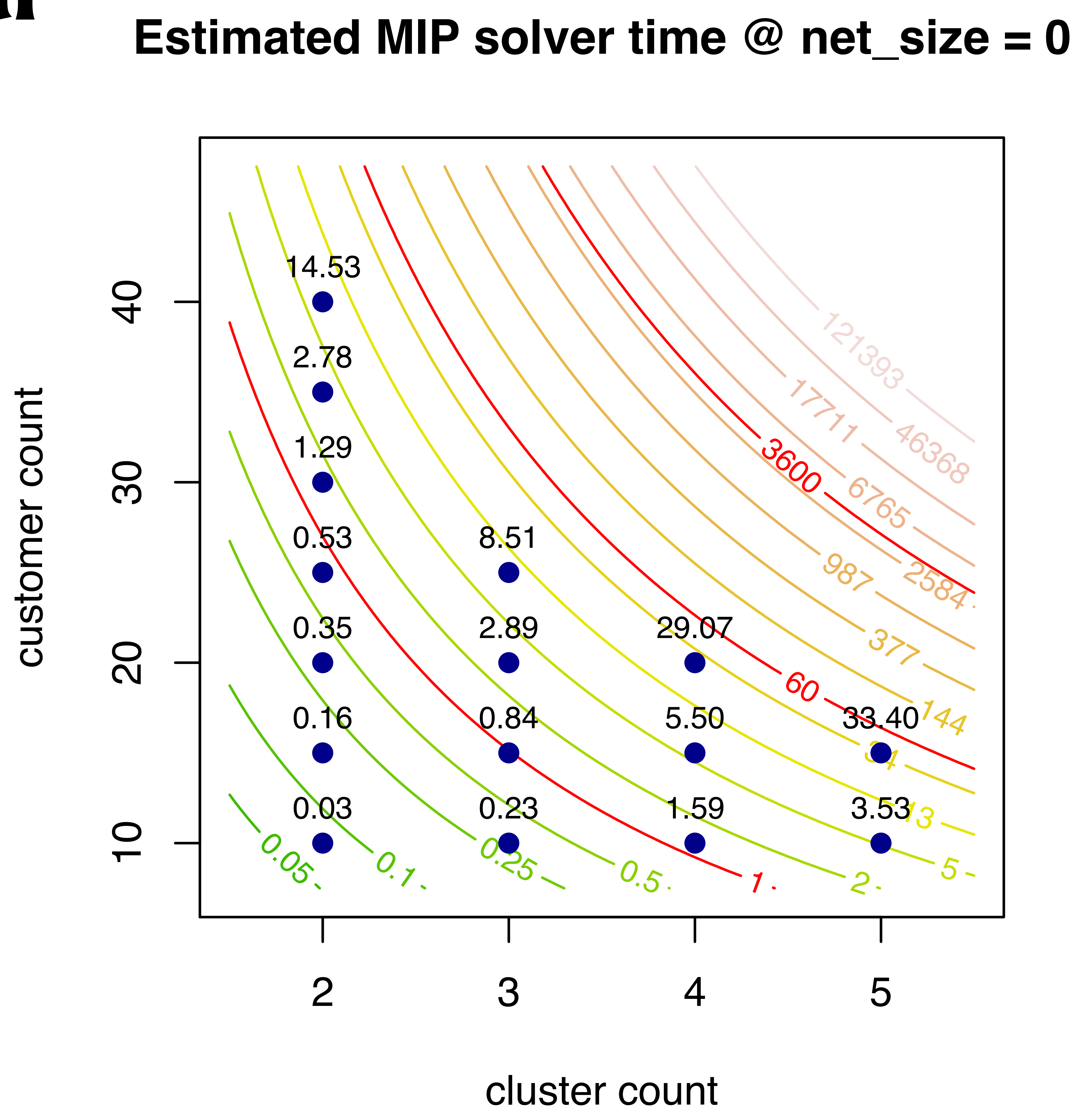
- Iš atlikto eksperimento laiko statistikos sudarėm modelį, kuris prognozuoja kiek “gurobis” užtruks išspręsti tam tikrą uždavinį
- Laiko modelis (sekundėmis) ir mažiausių kvadratų metodu įvertinti koeficientai:

$$T(K, N, M) \approx 0.0044 \cdot 1.92^{K} \cdot 1.079^{K\times N} \cdot 1.028^{M}$$

  - K - klasterių skaičius
  - N - (klasteriuojamų) taškų skaičius
  - M - tinklo sudėtingumas (segmentų skaičius tinkle)

$$T(K, N, M) \approx 2^{-c} \cdot 2^{\frac{19}{20}K} \cdot 2^{\frac{K\times N}{9}} \cdot 2^{\frac{M}{25}}, \quad c \approx 7.84$$

- Nedidelis komentaras: jeigu tinklo ribojimo centrams nėra, laiko prognozės modelyje naudoti $M = 0$



Laiko modelis “gurobi” įrankiui

- Iš atlikto eksperimento laiko statistikos sudarėm modelį, kuris prognozuoja kiek “gurobis” užtruks išspręsti tam tikrą uždavinį
- Laiko modelis (sekundėmis) ir mažiausių kvadratų metodu įvertinti koeficientai:

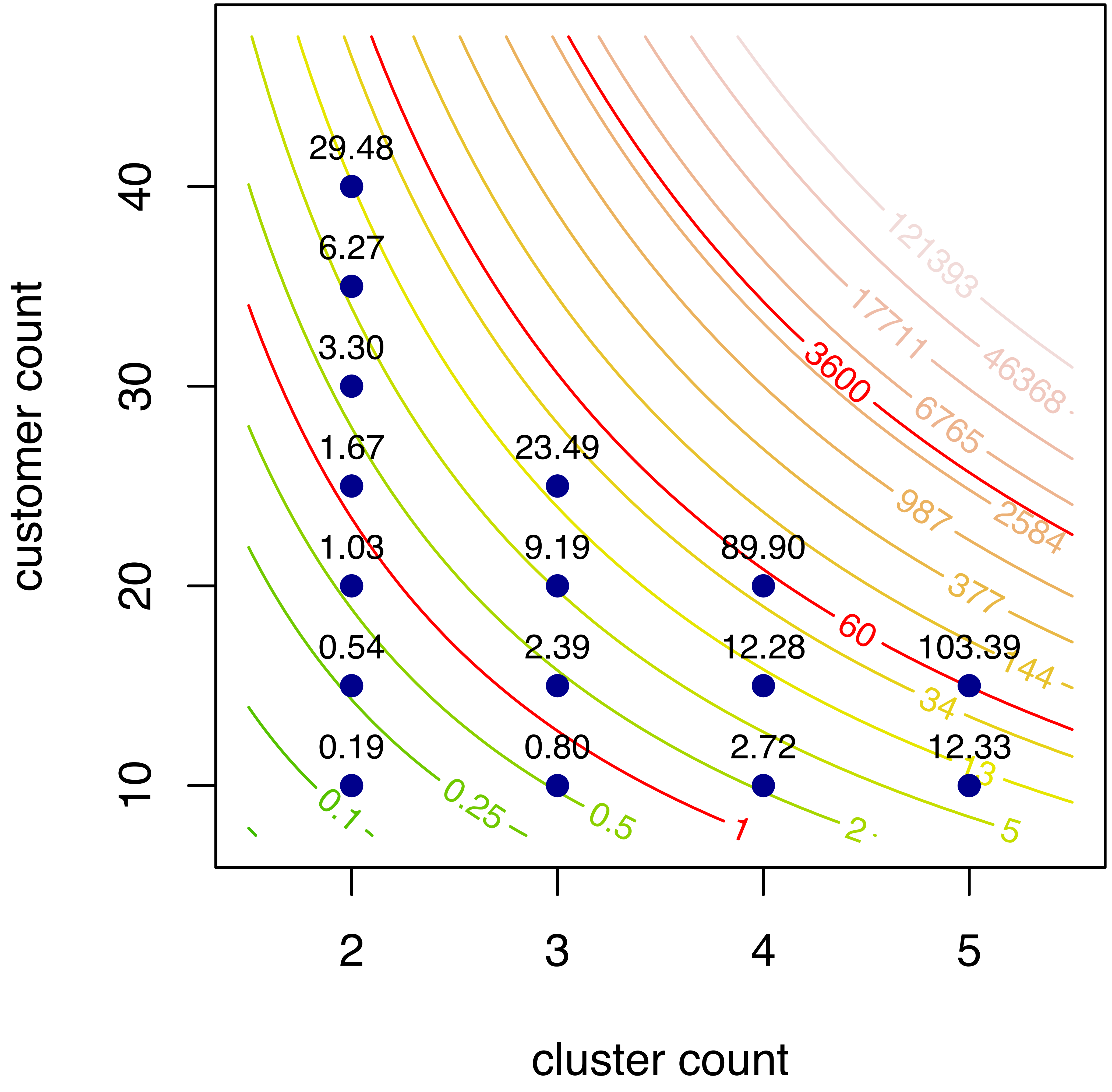
$$T(K, N, M) \approx 0.0044 \cdot 1.92^{K} \cdot 1.079^{K \times N} \cdot 1.028^{M}$$

  - K - klasterių skaičius
  - N - (klasteriuojamų) taškų skaičius
  - M - tinklo sudėtingumas (segmentų skaičius tinkle)

$$T(K, N, M) \approx 2^{-c} \cdot 2^{\frac{19}{20}K} \cdot 2^{\frac{K \times N}{9}} \cdot 2^{\frac{M}{25}}, \quad c \approx 7.84$$

- Nedidelis komentaras: jeigu tinklo ribojimo centrams nėra, laiko prognozės modelyje naudoti $M = 0$

Estimated MIP solver time @ net_size = 20



Laiko modelis “gurobi” įrankiui

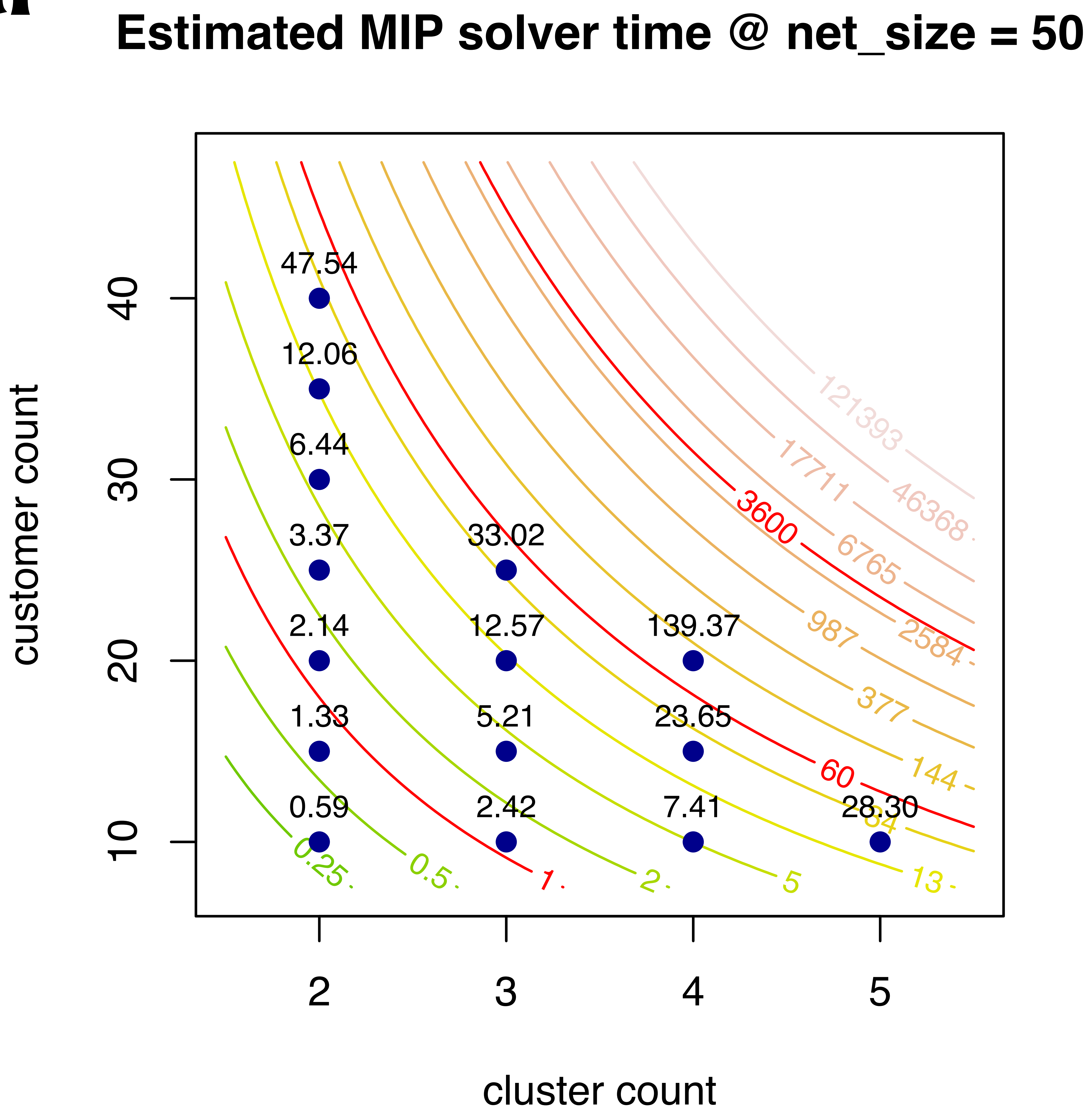
- Iš atlikto eksperimento laiko statistikos sudarėm modelį, kuris prognozuoja kiek “gurobis” užtruks išspręsti tam tikrą uždavinį
- Laiko modelis (sekundėmis) ir mažiausių kvadratų metodu įvertinti koeficientai:

$$T(K,N,M) \approx 0.0044 \cdot 1.92^{K} \cdot 1.079^{K\times N} \cdot 1.028^{M}$$

  - K - klasterių skaičius
  - N - (klasteriuojamų) taškų skaičius
  - M - tinklo sudėtingumas (segmentų skaičius tinkle)

$$T(K,N,M) \approx 2^{-c} \cdot 2^{\frac{19}{20}K} \cdot 2^{\frac{K\times N}{9}} \cdot 2^{\frac{M}{25}}, \quad c \approx 7.84$$

- Nedidelis komentaras: jeigu tinklo ribojimo centrams nėra, laiko prognozės modelyje naudoti $M = 0$



Laiko modelis “gurobi” įrankiui

- Iš atlikto eksperimento laiko statistikos sudarėm modelį, kuris prognozuoja kiek “gurobis” užtruks išspręsti tam tikrą uždavinį
- Laiko modelis (sekundėmis) ir mažiausių kvadratų metodu įvertinti koeficientai:

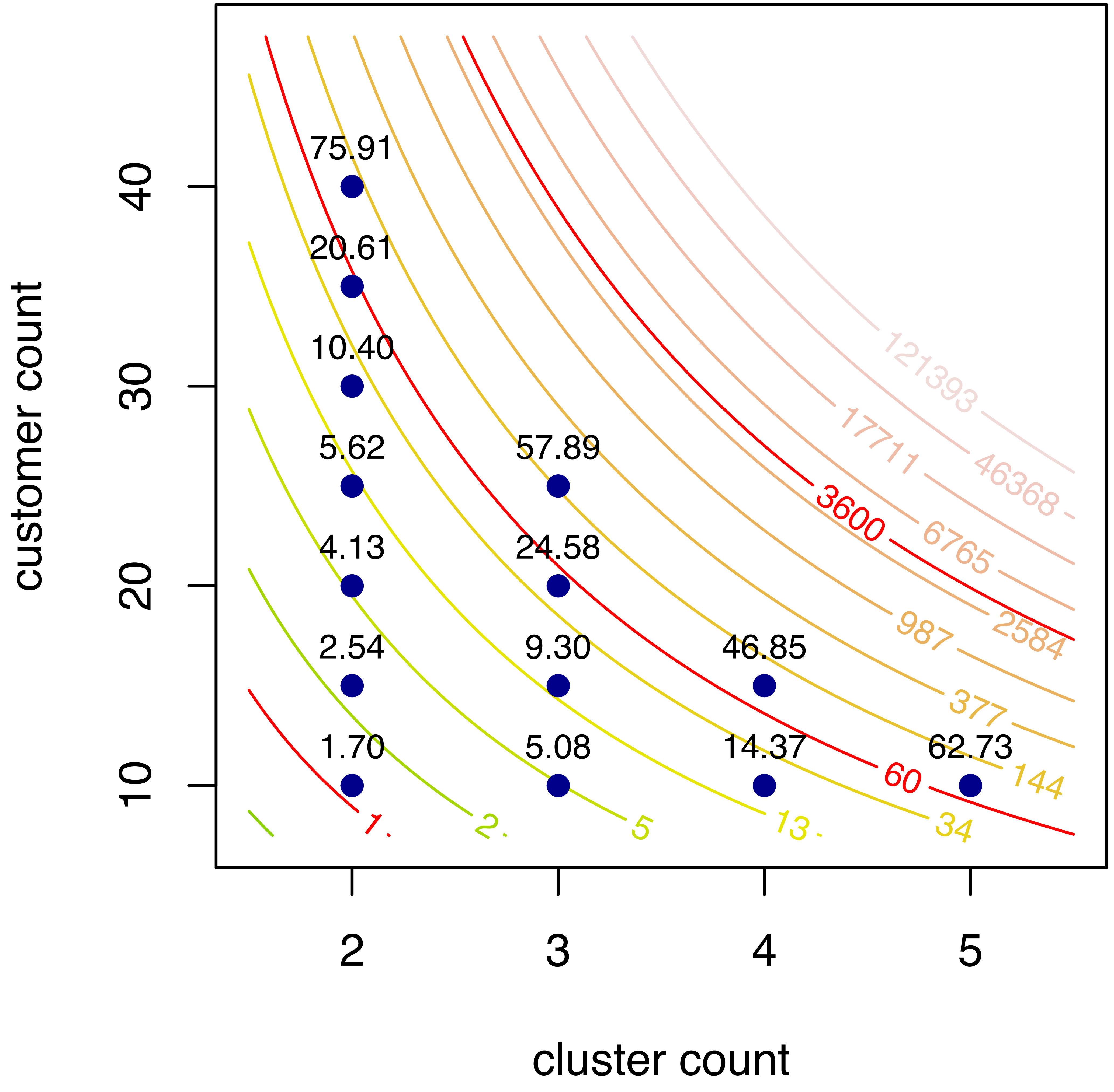
$$T(K, N, M) \approx 0.0044 \cdot 1.92^{K} \cdot 1.079^{K\times N} \cdot 1.028^{M}$$

  - K - klasterių skaičius
  - N - (klasteriuojamų) taškų skaičius
  - M - tinklo sudėtingumas (segmentų skaičius tinkle)

$$T(K, N, M) \approx 2^{-c} \cdot 2^{\frac{19}{20}K} \cdot 2^{\frac{K\times N}{9}} \cdot 2^{\frac{M}{25}}, \quad c \approx 7.84$$

- Nedidelis komentaras: jeigu tinklo ribojimo centrams nėra, laiko prognozės modelyje naudoti $M = 0$

Estimated MIP solver time @ net_size = 100



Paskutinio mėnesio darbas

Paskutinio mėnesio darbas

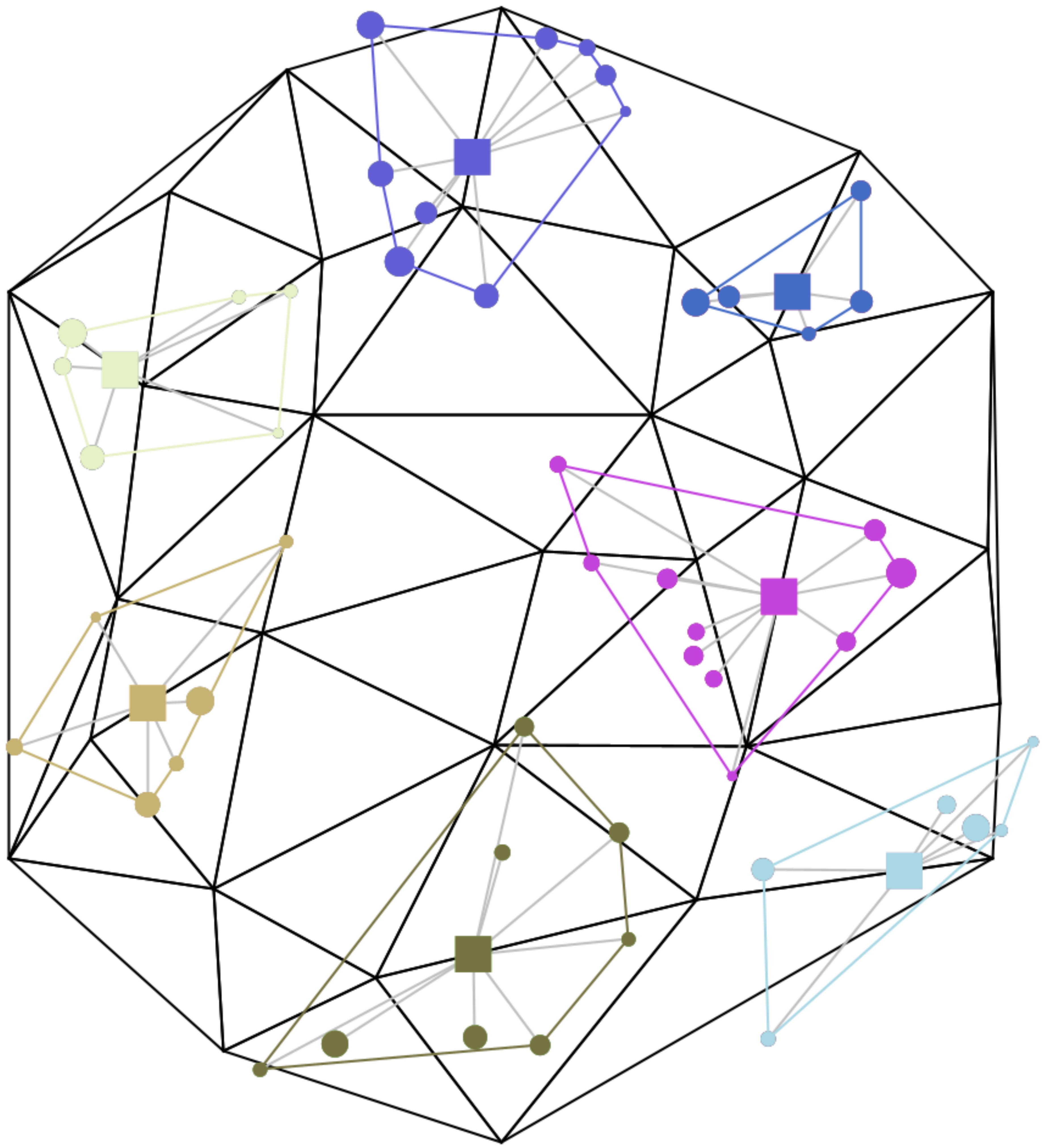
- Kadangi “gurobio” rezultatai nuvylė, pabandžiau suprogramuoti “protingos perrankos” algoritmą uždaviniui spręsti

Paskutinio mėnesio darbas

- Kadangi “gurobio” rezultatai nuvylė, pabandžiau suprogramuoti “protingos perrankos” algoritmą uždaviniui spręsti
- Pavyko išspręsti uždavinį su 7 klasteriais ir 50 taškų. Algoritmo laikas: ≈ 680 s. Laiko modelis prognozuoja, kad “gurobis” šiam uždaviniui būtų “užtrukęs” daugiau nei 1000 metų...

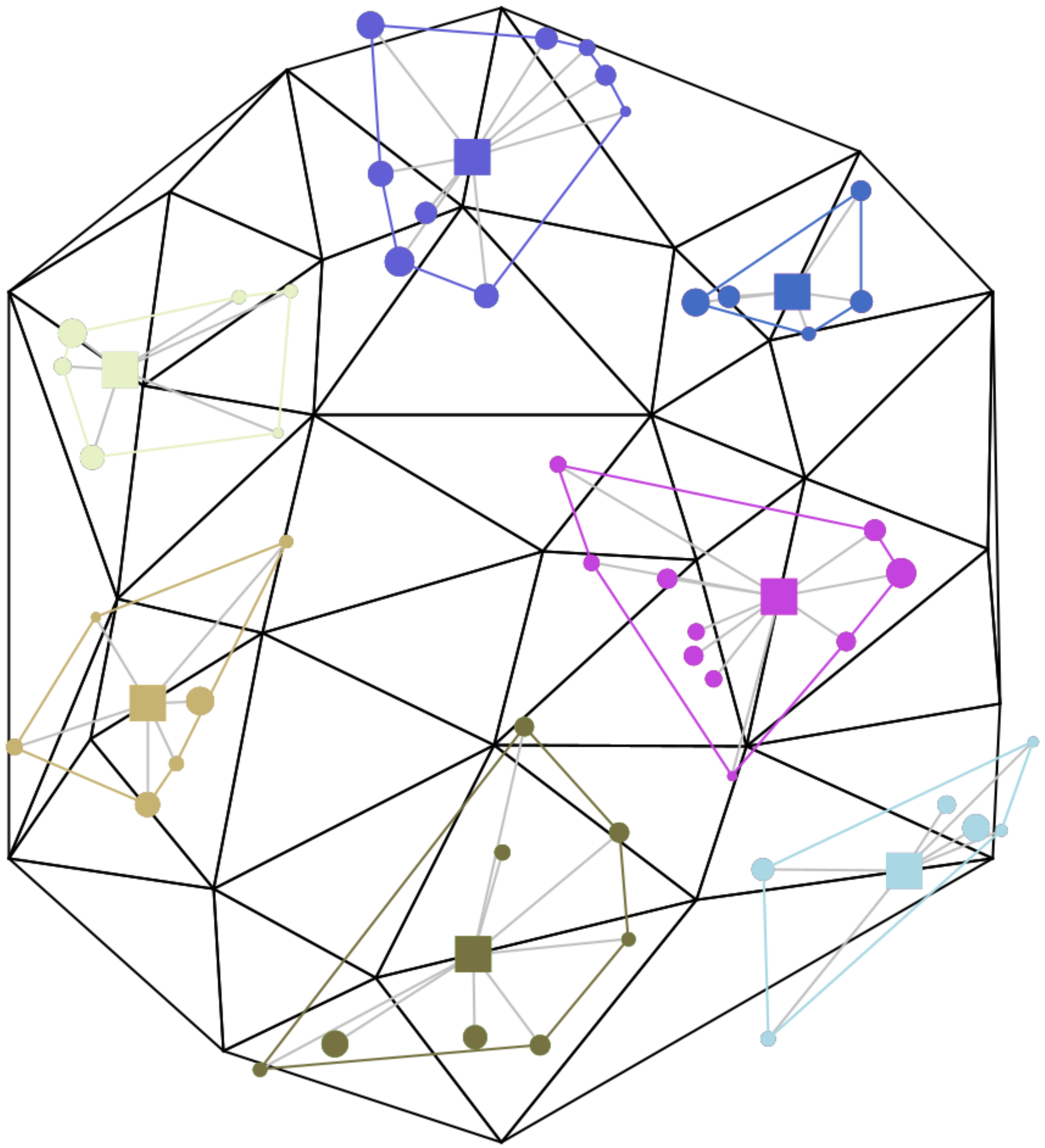
Paskutinio mėnesio darbas

- Kadangi “gurobio” rezultatai nuvylė, pabandžiau suprogramuoti “protingos perrankos” algoritmą uždaviniui spręsti
- Pavyko išpręsti uždavinį su 7 klasteriais ir 50 taškų. Algoritmo laikas: ≈ 680 s. Laiko modelis prognozuoja, kad “gurobis” šiam uždaviniui būtų “užtrukęs” daugiau nei 1000 metų...



Paskutinio mėnesio darbas

- Kadangi “gurobio” rezultatai nuvylė, pabandžiau suprogramuoti “protingos perrankos” algoritmą uždaviniui spręsti
- Pavyko išpręsti uždavinį su 7 klasteriais ir 50 taškų. Algoritmo laikas: ≈ 680 s. Laiko modelis prognozuoja, kad “gurobis” šiam uždaviniui būtų “užtrukęs” daugiau nei 1000 metų...
- Vis dėlto lyginant su neapribotu klasteriavimo uždaviniu, šis rezultatas nėra įspūdingas: neseniai išėjęs straipsns išsprendžia uždavinį su daugiau nei 1000 taškų ir palyginti nedideliu klasterių skaičiumi (≤ 10)



Ačiū už dėmesį!