



# Programinių našumo robotų metaeuristinių modelių kūrimas ir tyrimas

Parengė: Donatas Kavaliauskas

Darbo vadovas: Prof. habil. dr. Leonidas Sakalauskas

Doktorantūros laikotarpis 2015-2019 m.

# Įvadas

- Tyrimo objektas

- Disertacijos tyrimo objektas yra gamybos tvarkaraščių optimalaus planavimo realiu laiku automatizavimas (Edge AI). Tam pasiekti yra tiriama galimybė sudaryti gamybos tvarkaraščius artimus optimaliam tvarkaraščiui.

Analizuojamos optimalių tvarkaraščių sudarymo uždaviniai, jų ypatumai, bei matematiniai algoritmai, kurie NP klasės problemos uždaviniams gali pateikti artimą optimaliam sprendinį. Kadangi tokie algoritmai turi hyper-parametrus, nagrinėjamos šių algoritmų kalibravimo metodai, bei jų realizavimas programinio našumo robotų pavidalu.

- Tikslas

Disertacijos tikslas yra sukurti programinio produktyvumo roboto architektūrą, panaudojant metaeuristinius algoritmus sudaryti gamybos tvarkaraščius.

# Uždaviniai

- Sukurti programinio našumo roboto prototipinę architektūrą.
- Iširti skirtingų gamybos modelių optimizavimo uždavinių formuluotes skirtus tvarkaraščių sudarymui.
- Sukurti skirtingų metaeuristinių algoritmų arba metaeuristinių algoritmų su skirtingais parametru rinkiniais palyginimo metodologiją.
- Parinkti ir pritaikyti metaeuristinį algoritmą, tinkamą sudaryti artimą optimaliam sprendiniui gamybos tvarkaraštį.
- Pasiūlyti tinkamą algoritmų hyper-parametru kalibravimo metodologiją.

# Mokslinis naujumas

- Pasiūlyta programinio našumo roboto architektūra.
- Pasiūlyta Monte Carlo metodas metaeuristinių algoritmų palyginimui.
- Pasiūlytas modeliuojamo atkaitinimo algoritmo modifikacija tvarkaraščių sudarymui remiantis Pareto tipo tikimybiniais skirstiniais.
- Pasiūlytas metaeuristinių algoritmų automatizuotas hyper parametrų kalibravimo metodas.

# Ginamieji teiginiai

- Optimalių tvarkaraščių sudarymo problema yra opi įvairioms sritims, kaip mokykloms, transporto pervežimui, gamybos planavimui, statybų projektavimui ir kitoms.
- Iki šiol dar nėra sukurtos programinės įrangos, kuri visiems tvarkaraščiams rastų optimalų arba artimą optimaliam sprendinį.
- 4.0 industrijos revoliucijos metu automatizuojant gamybos linijas nebeužtenka turėti tik tvarkaraštį, kuris yra leistinas. Reikia teikti optimizuotus tvarkaraščius, kad būtų išnaudojama automatizavimo įrenginių sparta.
- Metaeuristiniai algoritmai gali pagerinti vidutiniškai iki 30% gamybos tvarkaraščio trukmę, lyginant su standartiniais algoritmais.
- Kai kurių metaeuristinių algoritmų konvergavimas yra įrodytas, tačiau beveik visų metaeuristinių algoritmų konvergavimo greitis yra nežinomas.

# Darbo rezultatų apibavimas, periodiniai leidiniai

- Donatas KAVALIAUSKAS, Leonidas SAKALAUSKAS, 201. Study of Convergence in Metaheuristics Algorithms. *Journal on Baltic J. Modern Computing*, Vol. 7 (2019), No. 3, 436-443  
<https://doi.org/10.22364/bjmc.2019.7.3.10>
- D. Kavaliauskas, L. Sakalauskas, G. Felinskas, 2019 . ENHANCEMENT OF THE SIMULATED ANNEALING ALGORITHM AND METODOLIGY OF CONVERGENCE STUDY BY STATISTICAL SIMULATION. *International Journal of Computer Systems Science and Engineering*. **ISI WOS** (Recenzuojamas).
- Recenzuojami konferencijų leidiniai
  - D. Kavaliauskas, L. Sakalauskas, 2019. Conceptual model of productivity bot for smart construction planning. *Procedia Engineering* ISSN 1877-7058. (1-10 pp.)(Priimta spausdinimui).

# Darbo rezultatų apibavimas, konferencijos

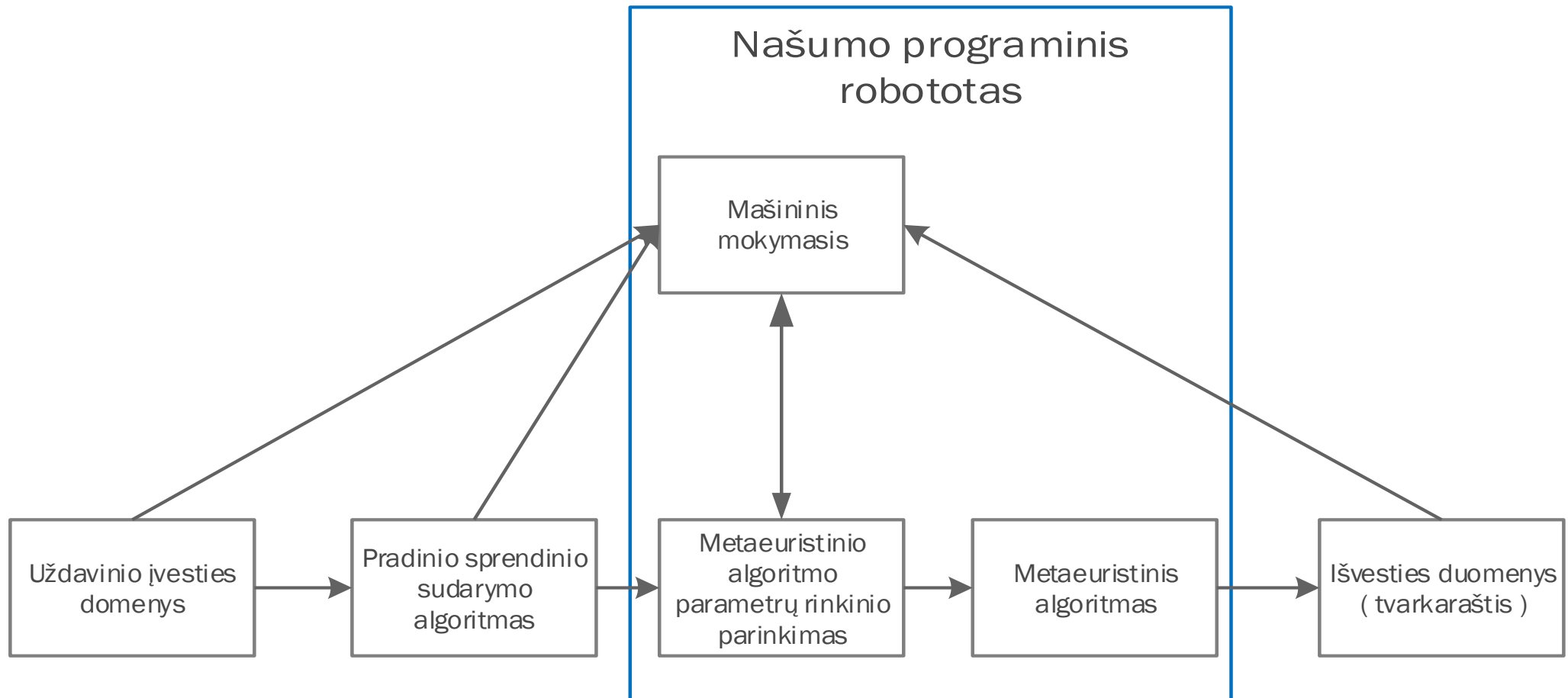
- Tarptautinės konferencijos:
  - 13-toji tarptautinė mokslinė konferencija: „Modern Buildings materials, Structures and Techniques“. Pranešimas: „Conceptual model of productivity bot for smart construction planning“. Lietuva, Vilnius, 2019-05-16 – 2019-05-17
  - 3-čioji tarptautinė mokslinė konferencija: „INNOVATIONS AND CREATIVITE“. Pranešimas: „Study of convergence in metaheuristics algorithms“. Latvija, Liepoja, 2019-06-06 – 2019-06-08
- Nacionalinės konferencijos:
  - 8-oji mokslinė konferencija „Operacijų tyrimas ir taikymas“. Pranešimas: „Modeliuojamo atkaitinimo algoritmo konvergavimo tyrimas“. Kaunas, Lietuva, 2016-04-8
  - Dalyvauta 8-oje tarptautinėje konferencijoje: „Data Analysis Methods for Software Systems“. Stendinis pranešimas: „Simulated Annealing Algorithm Optimization Method Convergence“. Druskininkai, Lietuva, 2016-12-1 – 2016-12-03
  - Respublikinė mokslinės- praktikos konferencija: “Informacinių technologijų iššūkiai kūrybos ekonomikoje”. Pranešimas: „Modeliuojamo atkaitinimo algoritmo optimizacijos konvergavimo tyrimas“. Šiauliai, Lietuva, 2017-03-17

# Disertacijos praktinė reikšmė

- Lietuvos verslo paramos agentūra. Intelektas LT. Bendri mokslo- verslo projektai. LVPA-K-828:
- Pavadinimas: Intelektinės, save apsimokančios gamybos procesų planavimo sistemos, skirtos lanksčiai ir kompleksiškai valdyti gamybos procesus, kūrimas panaudojant metaeuristinius metodus.
- Projekto numeris: J05-LVPA-K-01-0026

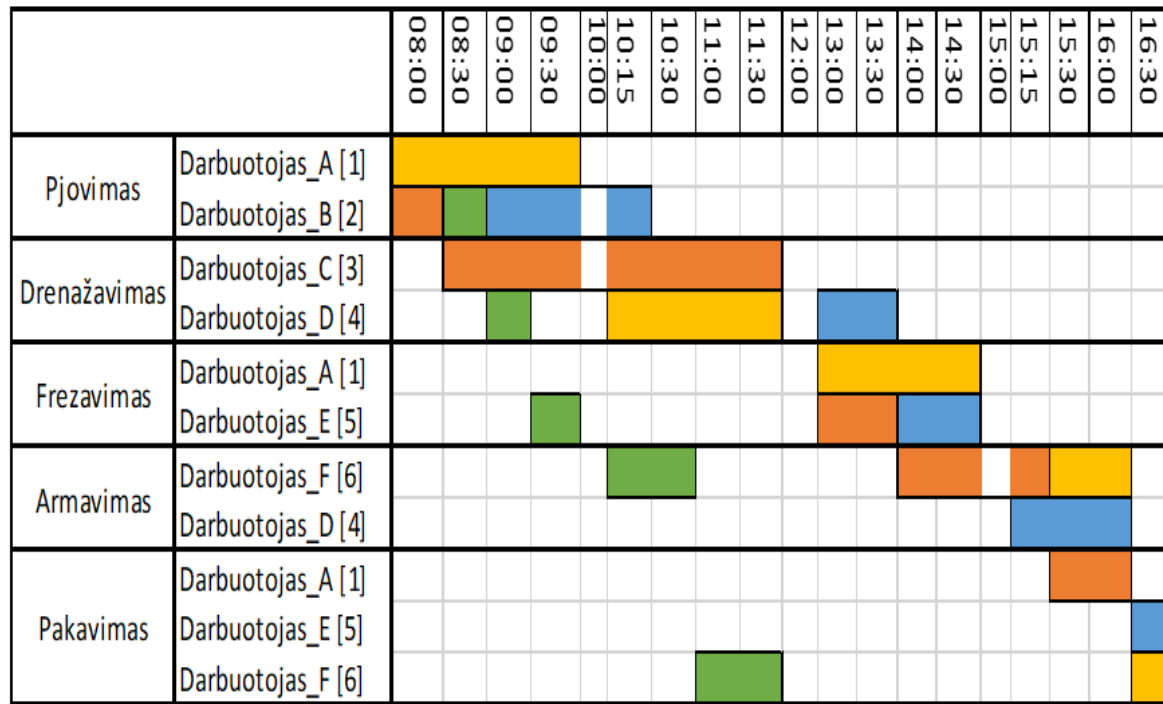


# Programinio roboto architektūra

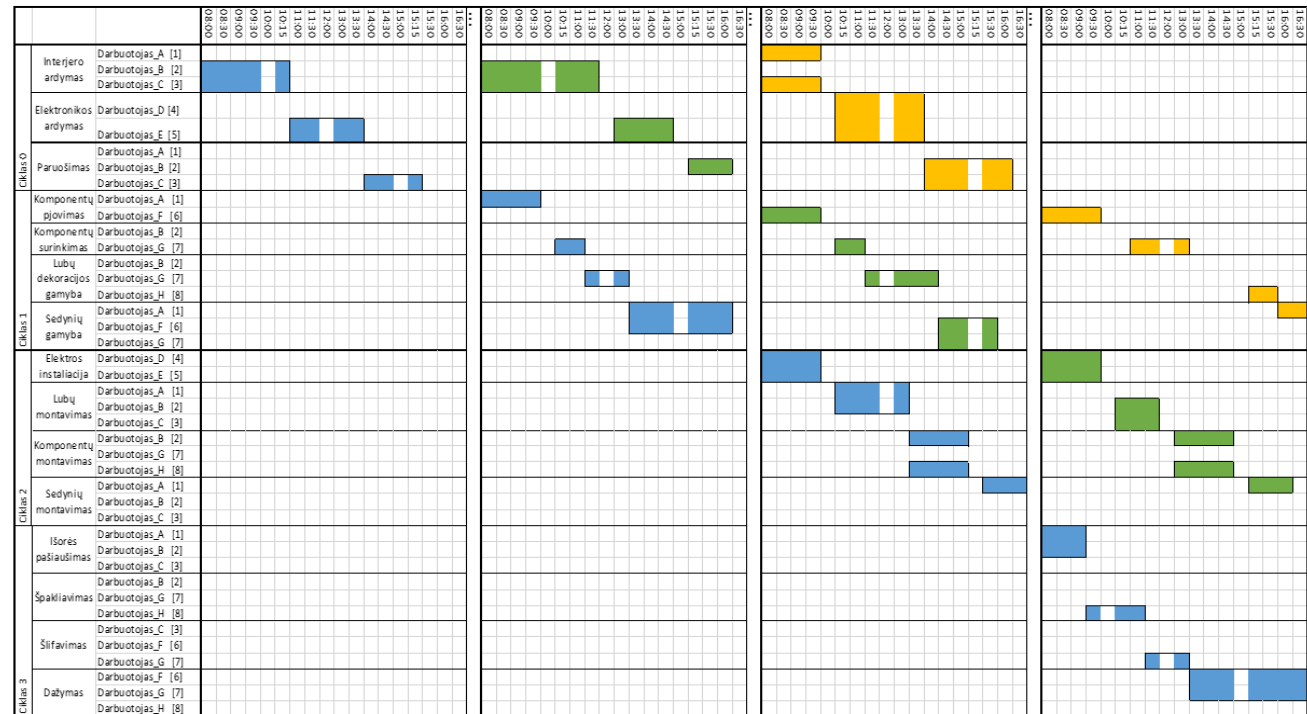


# Gamybos modeliai (1)

## Vieno srauto gamybos modelis

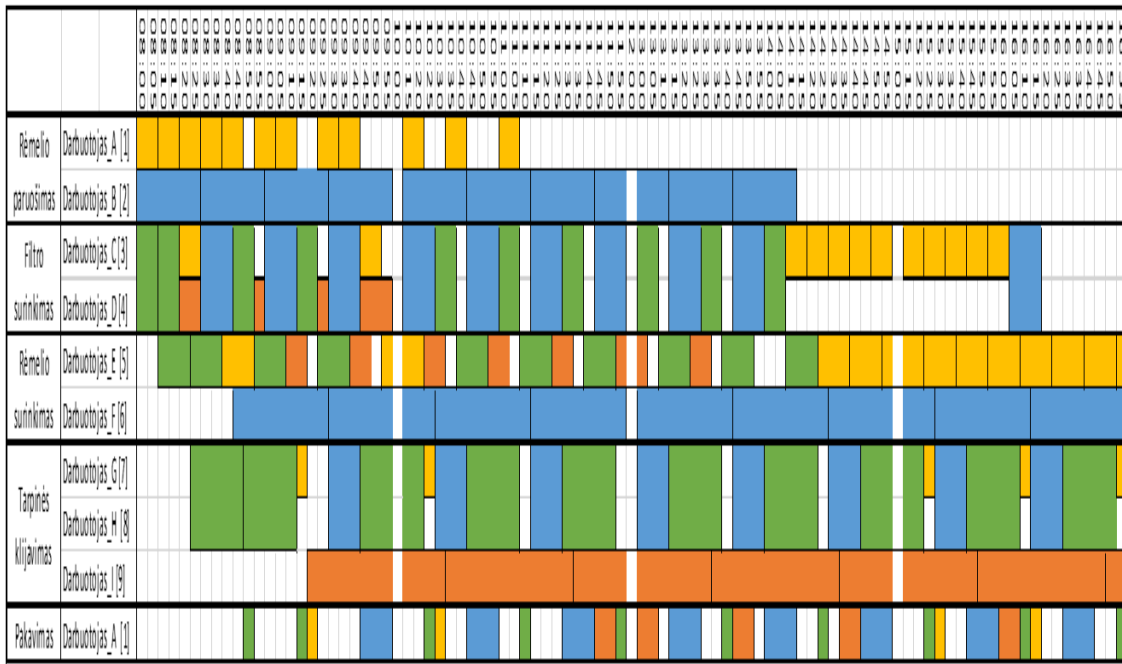


## Gamybos ciklais modelis

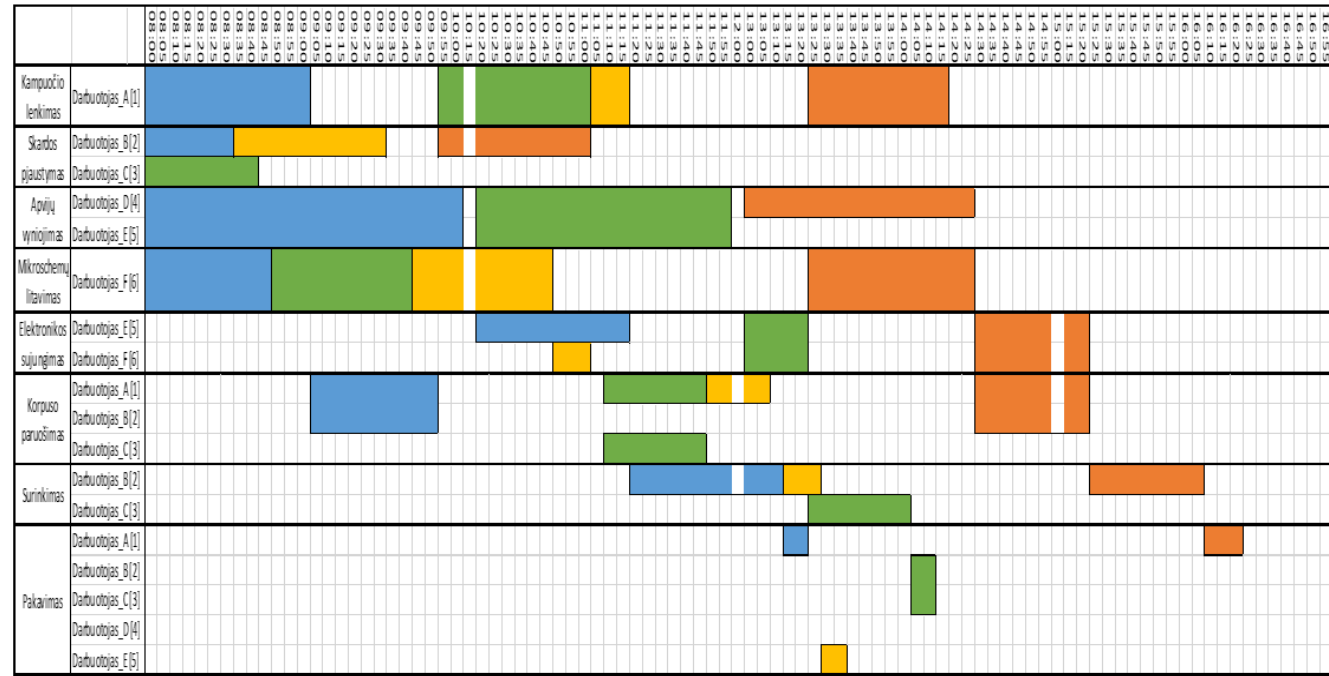


# Gamybos modeliai (2)

## Gamybos partijomis modelis



## Lygiagrečios gamybos modelis



# Optimizavimo formuluotė

Tikslo funkcija:

$$\sum_{t \in T} (t - 1) * x_{(n+1,t)}$$

Apribojimai:

$$\sum_{t \in T} x_{(j,t)} = 1 \quad \forall j \in J \cup \{n + 1\}$$

$$\sum_{j \in J} \sum_{t' = t - p_j + 1} u_{(j,r)} * x_{(j,t')} \leq c_r \quad \forall t \in T, r \in R$$

$$\sum_{\substack{t \in T \\ \cup \{n + 1\}}} t * x_{(s,t)} - \sum_{t \in T} t * x_{(j,t)} \geq p_j \quad \forall (j, s) \in S, x_{(j,t)} \in \{0, 1\}, \forall j \in J$$

- J – užduotys
- R – atsinaujinantys resursai
- S – Pirmumo sąrašas tarp užduočių  $(i, j) \in J \times J$
- T – užduočių laikų matrica.
- $P_j$  – j užduoties atlikimo laikas
- $U_{(r,j)}$  Resursų kiekis reikalingas atlikti darbą j.
- $C_r$  – resursų atsinaujinimo kiekis

# Algoritmų sudėtingumas

- Bendru atveju tvarkaraščių uždavinys yra NP sudėtingumo J. D. Ullman (1975), Karp (1972), Graham (1966), Mirshekarian, Sadegh; Šormaz, Dušan (2016) ir kiti.
- Pvz.: open shop uždavinys dviem procesoriams be nuoseklumo sąryšio bendru atveju yra eksponentinio sudėtingumo.
- Butelio kakliuko problema atsirandą tik uždaviniuose su nuoseklumo sąryšiais. Jie palengvina uždavinio sprendimą.

# Statistinis metodas metaeuristinių algoritimų palyginimui remiantis Monte Carlo metodu

1. Generuojamos nagrinėjamos klasės uždavinių parametų atsitiktinės imtys  $W(x_i)$ ;
2. Skaičiuojamos išėjimo charakteristikų  $y$  reikšmės, kiekvienam atsitiktinių parametų rinkiniams;
3. Modeliavimo rezultatai apdorojami statistiniais metodais.

Gautų charakteristikų skirstiniams nustatyti buvo naudojamas didžiausios tikėtinumo metodas. Ši metodika susideda iš etapų:

1. Sudarome ir įvertiname Monte Carlo metodų tikėtinumo funkciją;
2. Apskaičiuojame tikėtinumo funkcijos dalinių išvestinių Monte Carlo įverčius pagal visus parametrus;
3. Prilyginame rastas išvestines nuliui ir išsprendžiame gautą lygčių sistemą nagrinėjamų parametų atžvilgiu;
4. Gauti lygčių sistemos sprendiniai laikomi ieškomaisiais įverčiais.

Taigi, sudarome funkciją  $g(x)$ , kurioje palyginame Weibull'o skirstinį su Pareto skirstiniu

Disertacijoje sudarytas kriterijus prinkti empirinį skirstinį tarp Weibyulo ir pareto

$$g(x) = \left( \frac{\sum_{i=1}^n \hat{a}_i}{n} \right)^2 / \frac{\sum_{i=1}^n (\hat{a}_i)^2}{n}$$

# Statistinio modeliavimo metodo teikiama nauda

- Geriausios pasiektos tikslo funkcijos vidutinės priklausomybės;
- Geriausių sprendimų atstumus nuo optimalaus sprendimo iki vidutinės priklausomybės;
- Bendros priklausomybės tikimybė rasti globalų sprendimą arba patekti į visuotinio optimalumo zoną;
- Geriausiai pasiektos tikslo funkcijos ir geriausio sprendimo atstumo priklausomybė nuo optimalių sprendimo empirinių pasiskirstymų;
- Algoritmo hyper-parametrų priklausomybės.
- Panašumus su gerai žinomais skirstiniais.

# Modeliuojamo atkaitinimo algoritmas (1)

**Step 0.** Let  $x_0 \in X$  be a given starting point,  $z_0 = \{x_0\}$  and  $k = 0$ ;

**Step 1.** We calculate the temperature parameter  $T$  at iteration  $k$  on which the number of elementary transformation operations depends. We constructed special neighborhood depth ( $\rho_k$ ) generating algorithm, based on generation of stable Pareto values:

**Step 1. 1.** Set initial  $i := 1$  and  $T := 0$ .

**Step 1. 2.** Generate  $U1$  and  $U2$ , uniformly distributed in (Gelatt et al., 1983), and (Weise, 2011). Then calculate:

$$Z := \left( \frac{\sin((\alpha-1) \cdot U1 \cdot \pi)}{-\ln(U2)} \right)^{\frac{1-a}{a}} \cdot \frac{\sin(\alpha \cdot U1 \cdot \pi)}{\left( \cos(\alpha \cdot \frac{\pi}{2}) \cdot \sin(U1 \cdot \pi) \right)^{\frac{1}{\alpha}}}$$

$$T := T + Z$$

**Step 1. 3.** Calculating a  $\rho_k$  value:

$$\rho_k = \begin{cases} i, & T \geq T_0 \\ i + 1, & T < T_0 \end{cases}$$

**Step 2.** A new solution is being developed by carrying out elementary rearrangements up to the permitted temperature by volume to the solution set In the following calculations as simply rearrangements were used randomly shift and swap operation in the solution set.



# Modeliuojamo atkaitinimo algoritmas (2)

**Step 3.** We calculate the objective function with new meanings of values.

**Step 4.** The resulting value is checked using the Metropolis-Hastings (MH) criterion algorithm, which is expressed in:

**Step 4.1.** Select the initial value  $\theta$ ;

**Step 4.2.** Iteration  $t$  proposes to go to the value of  $\theta^*$  with the probability  $J_t(\theta^* | \theta^{(t-1)})$ ;

**Step 4.3.** Calculate the acceptable ratio (opportunity) (Weise, 2011):

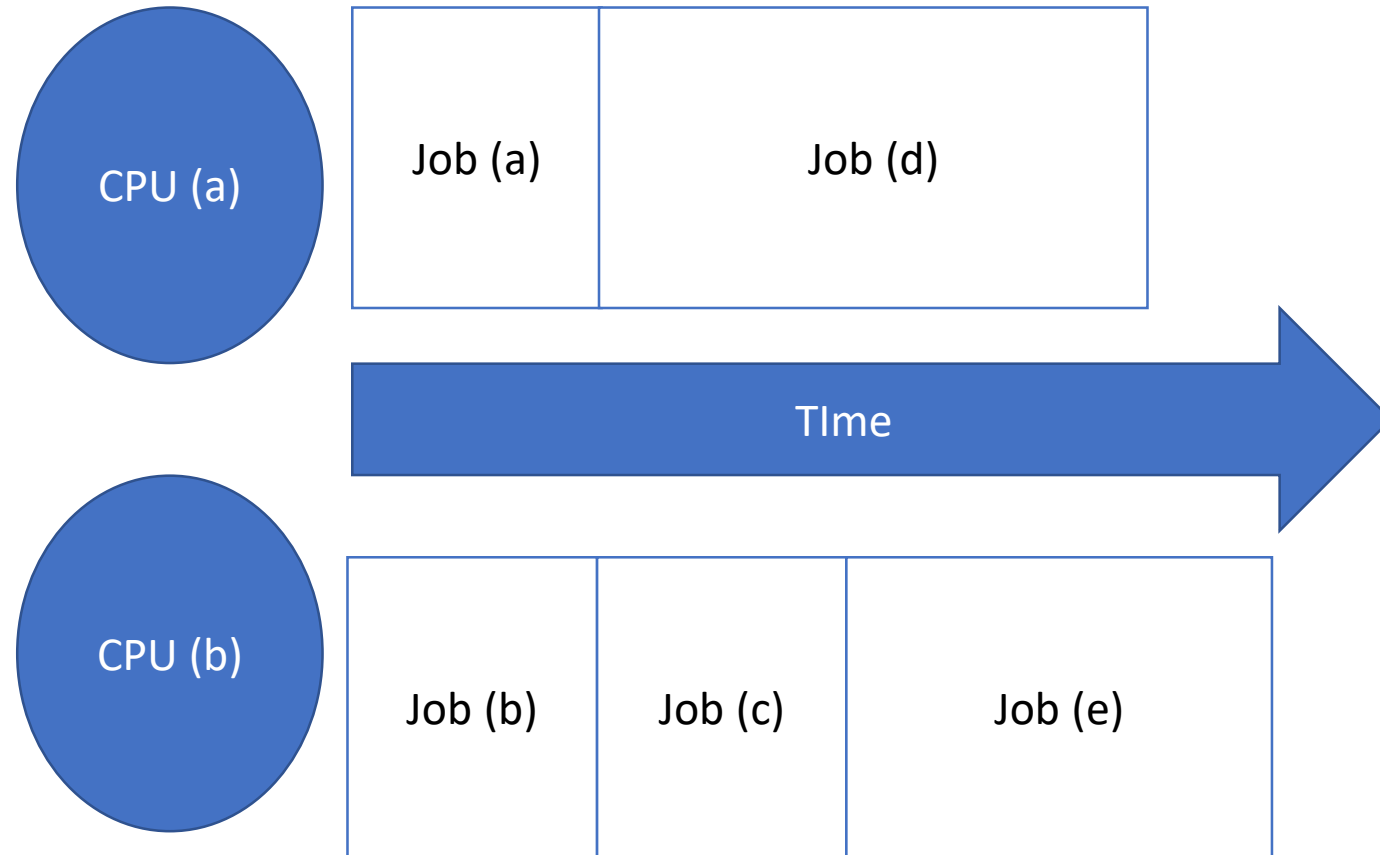
$$r = \frac{p(\theta^0 | y) / J_t(\theta^* | \theta^{(t-1)})}{p(\theta^{(t-1)} | y) / J_t(\theta^{(t-1)} | \theta^*)}$$

**Step 4.4.** We accept  $\theta^*$  as  $\theta(t)$  with the option  $\min(r, 1)$ . If  $\theta^*$  is not accepted, then  $\theta(t) = \theta(t-1)$ .

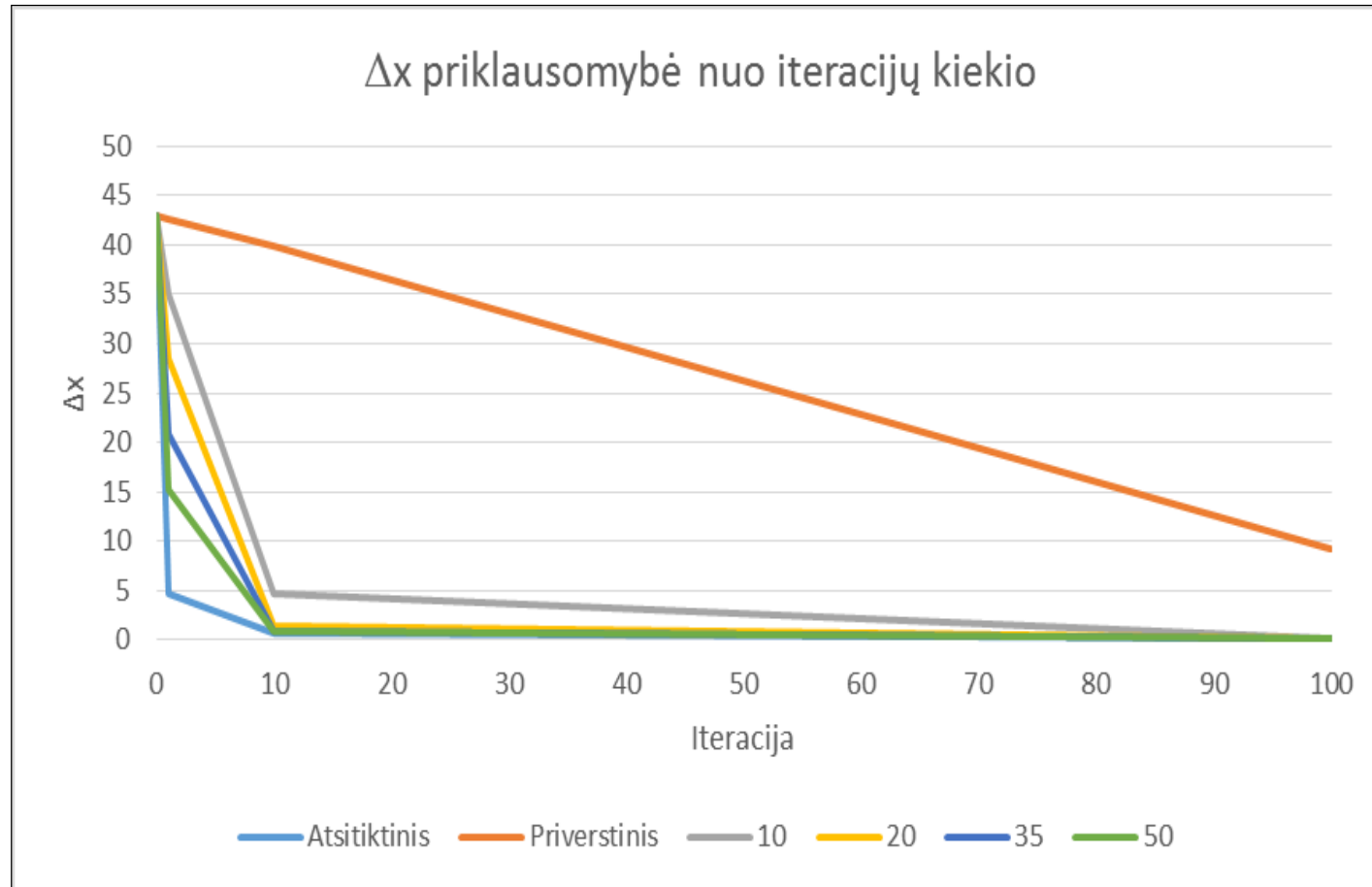
**Step 5.** Check a stopping criterion and if it fails set  $k = k + 1$  and go back to **Step 1**.

# Tvarkaraščių sudarymo be nuoseklumo sąryšių testinis uždavinys (angl. open shop)

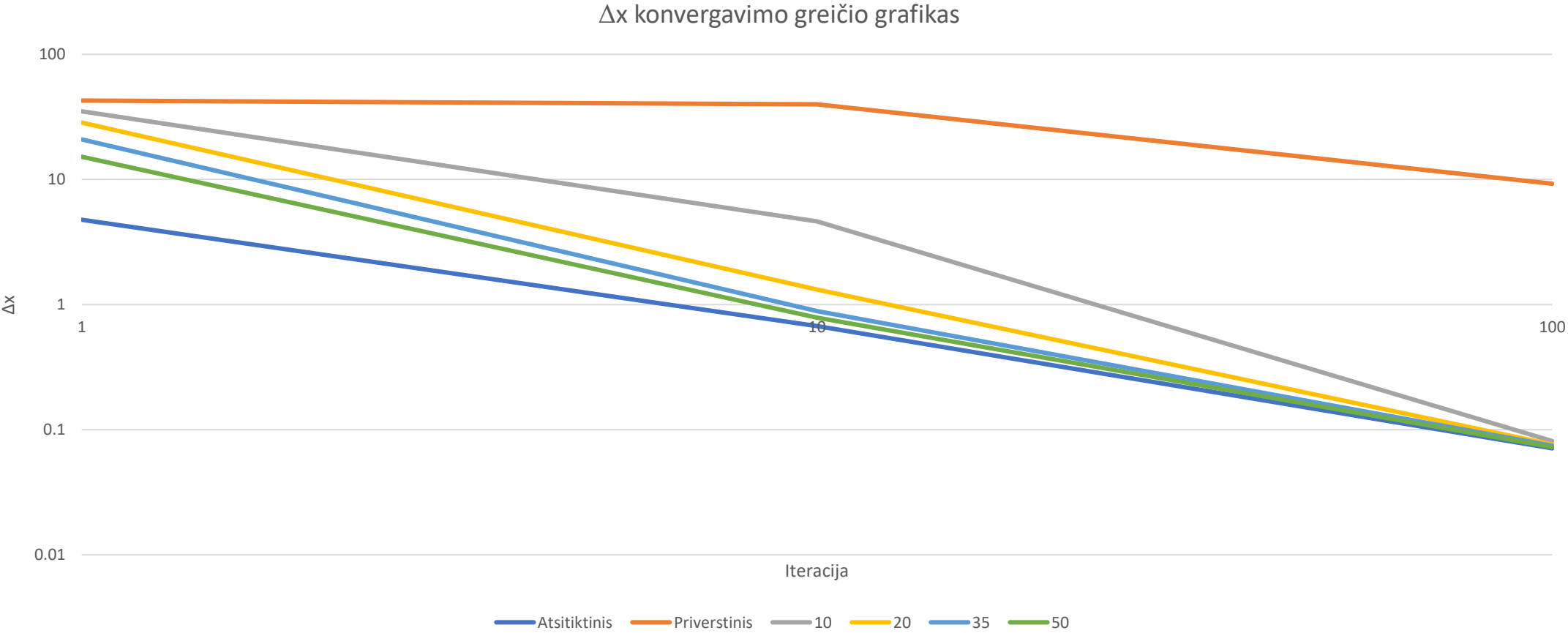
- Turime du procesorius, kurie dalijasi darbais. Yra 100 užduočių. Kiekvienai užduočiai trukmė yra atsitiktinai sugeneruojama pagal Gauso skirtinį. Jei pirmasis procesorius imasi užduoties iš sąrašo, tada antrasis procesorius negali atlikti tos pačios užduoties. Reikia paminėti, kad procesoriai yra vienodo pajėgumo.
- Testinio uždavinio **tikslo funkcija** yra sudaryti kuo trumpesnę tvarkaraštį.
- Bandymai pakartoti po 2000 kartų.



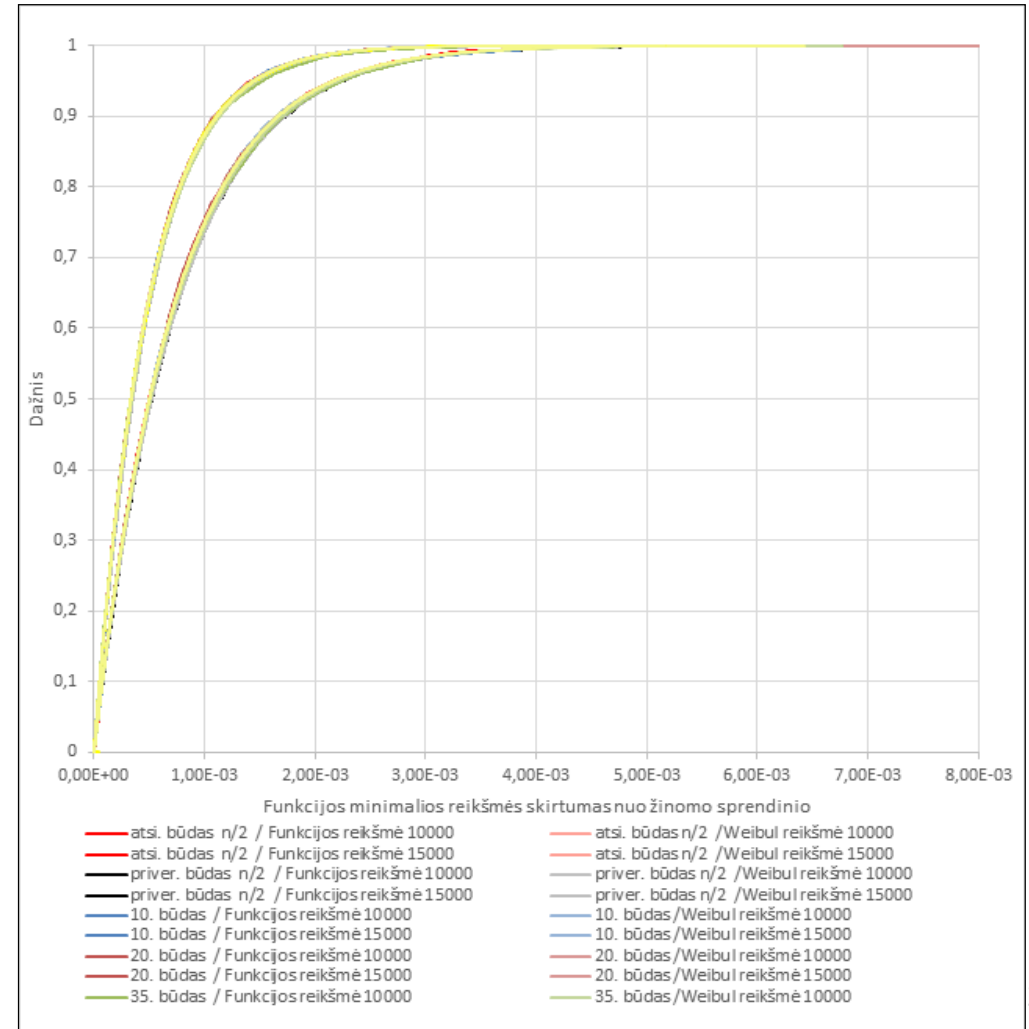
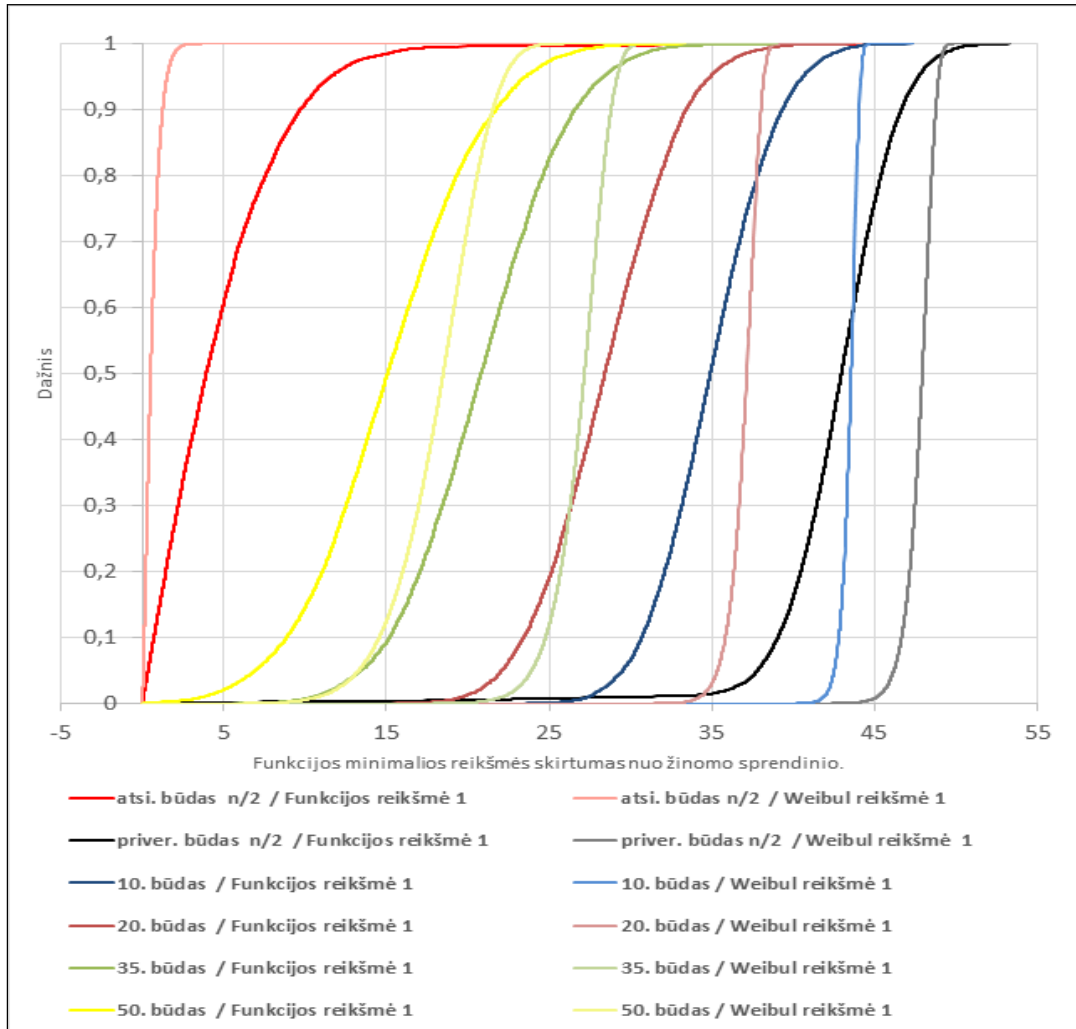
# Tikslo funkcijos atstumo nuo optimumo grafikas (M=2000 atsitiktinių imčių)



# Tikslo funkcijos konvergavimo greičio grafikas



# Weibulo skirtinio ir tikslo funkcijos empiriniai grafikai



# Tarptautinės PSPLib bibliotekos sprendinių palyginimas

- Bibliotekoje PSPLib galima rasti skirtingų uždavinių rinkinius skirtus įvairaus tipo tvarkaraščių su ribotais ištekliais sudarymo uždaviniams spręsti. Šiems uždaviniams yra žinomos optimalus sprendiniai.
- Spręsta 600 uždavinių iš PSPLib bibliotekos, kur kiekviename uždavinyje yra po 120 procesų, kurie gali naudoti 4 resursus. Kiekvieno proceso įvykdymo laikas ir resursų panaudojimas yra skirtingas

# Algoritmo palyginimas pagal vidutinį nuokrypį

- Vidutinis nuokrypis apskaičiuojamas pagal formulę:

$$\sigma = \sqrt{\frac{1}{n} * \sum_{i=1}^n (X_i - X_{vid})^2}$$

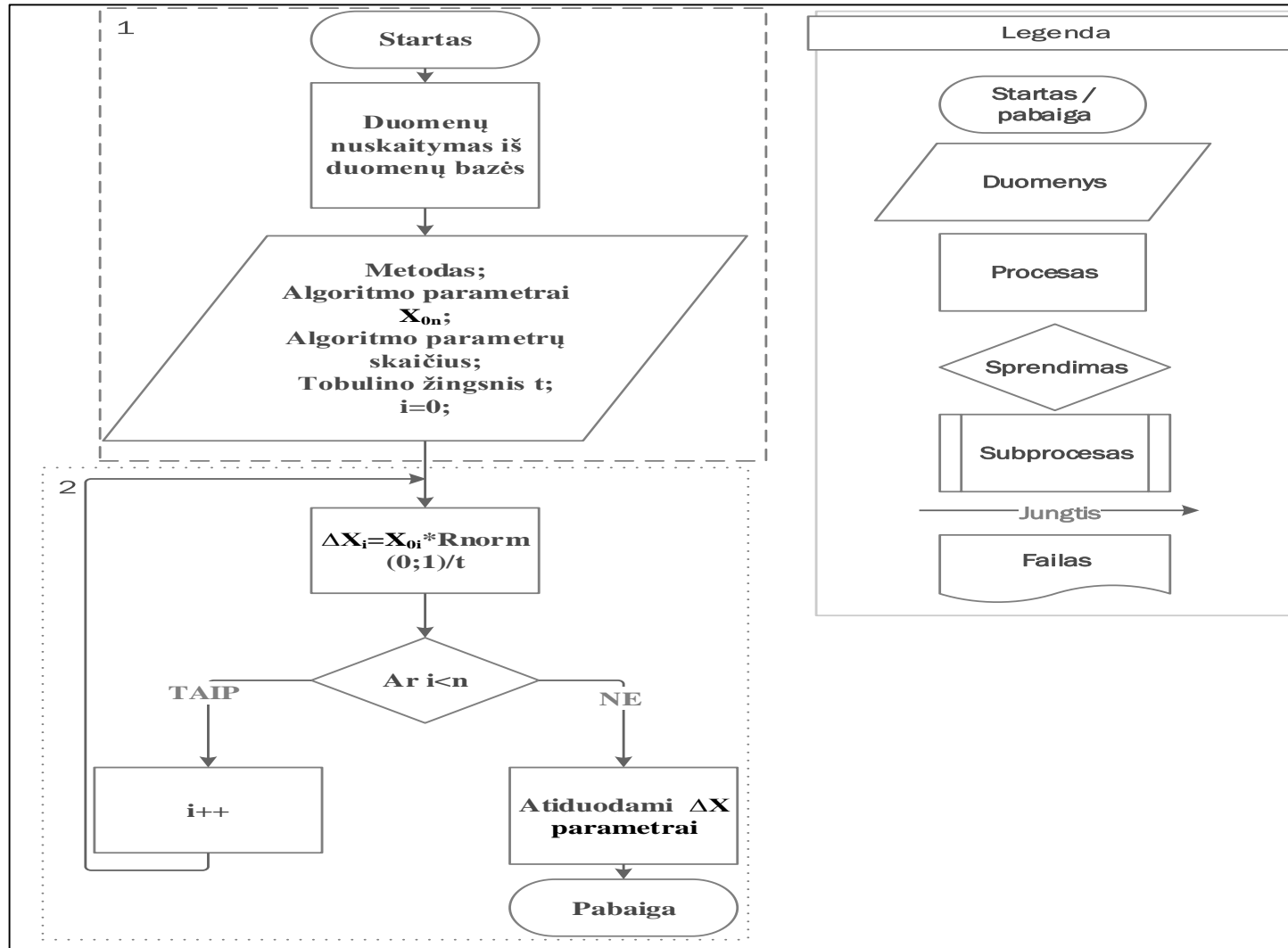
- Remiantis 31 autoriaus mokslinių publikacijų pateiktomis vidutinio nuokrypio reikšmėmis buvo palygintas su sukurto modeliuojamo atkaitinimo algoritmo rezultatais. Sukurtas modeliuojamo atkaitinimo algoritmo rezultatas po 1000 iteracijų patenka į geriausiųjų 15, kuris atsilieka nuo geriausio rezultato 12 %. Po 5000 iteracijų algoritmo rezultatai patenka į 12 vietą ir atsilieka nuo geriausio rezultato tik 13,5 %.

# Metaeuristinių algoritimų automatizuotas hyper parametrų kalibravimo metodai

- Uždavinių klasterizavimas į klases pagal charakteristikas.
- Kalibravimo metodai:
  - Pokoordinatinis metaeuristinių algoritimų parametrų kalibravimas.
  - Stochastinės aproksimacijos metodas:



Metaeuristinių algoritmų automatizuotas hyperparametru kalibravimo metodas remiantis daugiarankio bandito algoritmu.



# Išvados

- Nors metaeuristinių algoritmų taikymas optimaliam sprendimui reikalauja labai daug kompiuterinio laiko ir operacijų, tačiau pagrindinis tikslo funkcijos pagerėjimas pasiekiamas jau per pirmąsias veikimo iteracijas. Todėl yra tikslinga taikyti euristinius algoritmus su nedideliu iteracijų žingsniu.
- Algoritmo efektyvumas priklauso nuo jo hyper-parametrų parinkimo, todėl turi būti įgyvendintas tų parametrų parinkimas ir pritaikymas sprendžiamų uždavinių klasei, realizuojant išmaniojo programinio roboto kūrimą.
- Sukurta metaeuristinio algoritmo efektyvumo tyrimo metodika galima taikyti diskrečių ir tolydžių problemų sprendimo metodams. Ši metodika leidžia ištirti algoritmų konvergencijos greitį ir nustatyti jo eilę.
- Metaeuristiniai algoritmai konvergavimo greitis yra atvirkščiai proporcingas iteracijų skaičiui pakeltame laipsnyje  $n^{-\gamma}$ . Čia laipsnis  $\gamma$  yra tarp 0 ir 1.