



## Metropolis-Hastings: Sudoku

Aplikované evoluční algoritmy  
2022/2023

Autor: Vojtěch MIMOCHODEK (xmimoc01)  
Datum: 8. května 2023

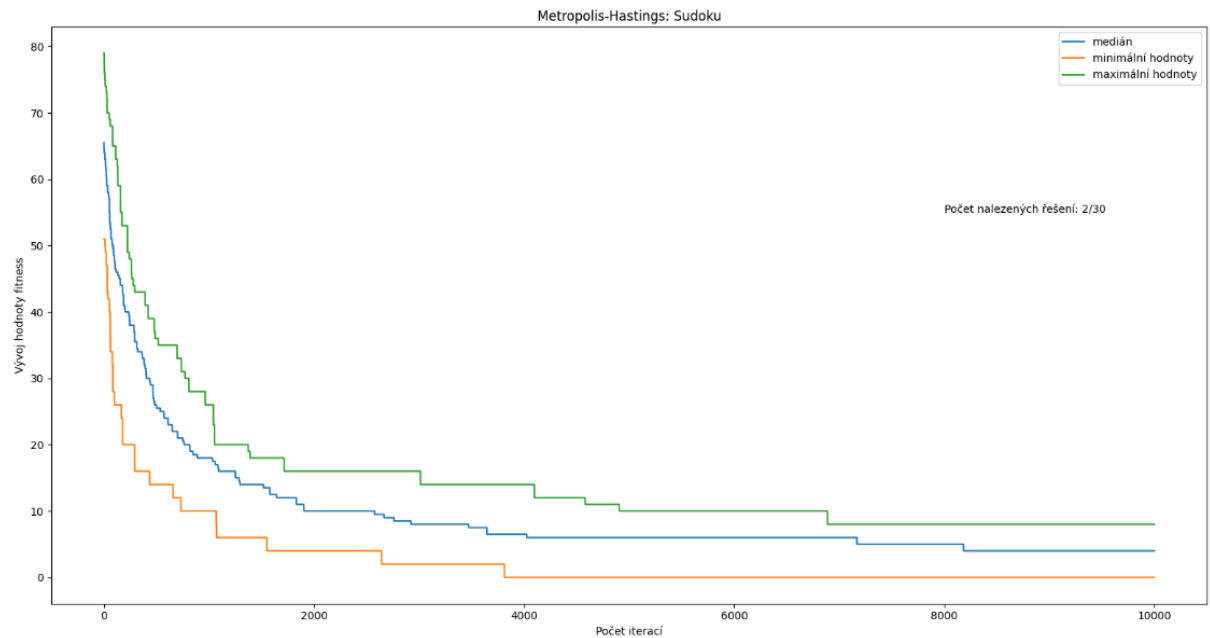
# Implementace

- Python
- Knihovna Py-Sudoku jako generátor validních desek
- Aplikace sestává ze souborů:
  - o *metropolis.py*, *main.py*, *service.py*
- Vstupní parametry:
  - o *--size*
  - o *--iterations*
  - o *--runs*
  - o *--difficulty*
  - o *--probability*
- Přijetí horšího stavu:
  - o  $\exp((actual\_fitness - new\_fitness) / probability) > random.random()$
- Hodnota fitness je reprezentována jako počet konfliktů na desce

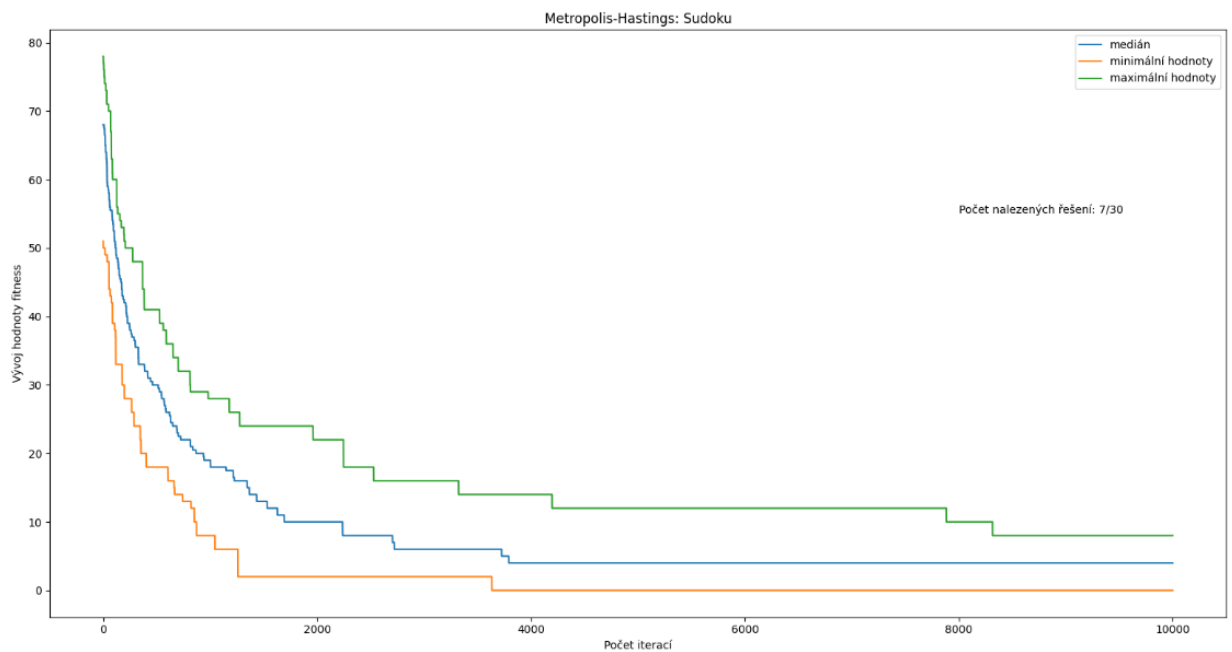
6	2	1	6	8	4	9	3	5
3	8	6	7	9	5	2	1	4
4	5	9	2	3	1	8	7	6
5	3	4	8	7	2	6	9	1
8	6	7	9	1	3	5	4	2
1	9	2	5	4	6	7	8	3
6	4	8	3	5	7	1	2	9
9	1	5	4	2	8	3	6	7
2	7	3	1	6	9	4	5	8

# Experimenty

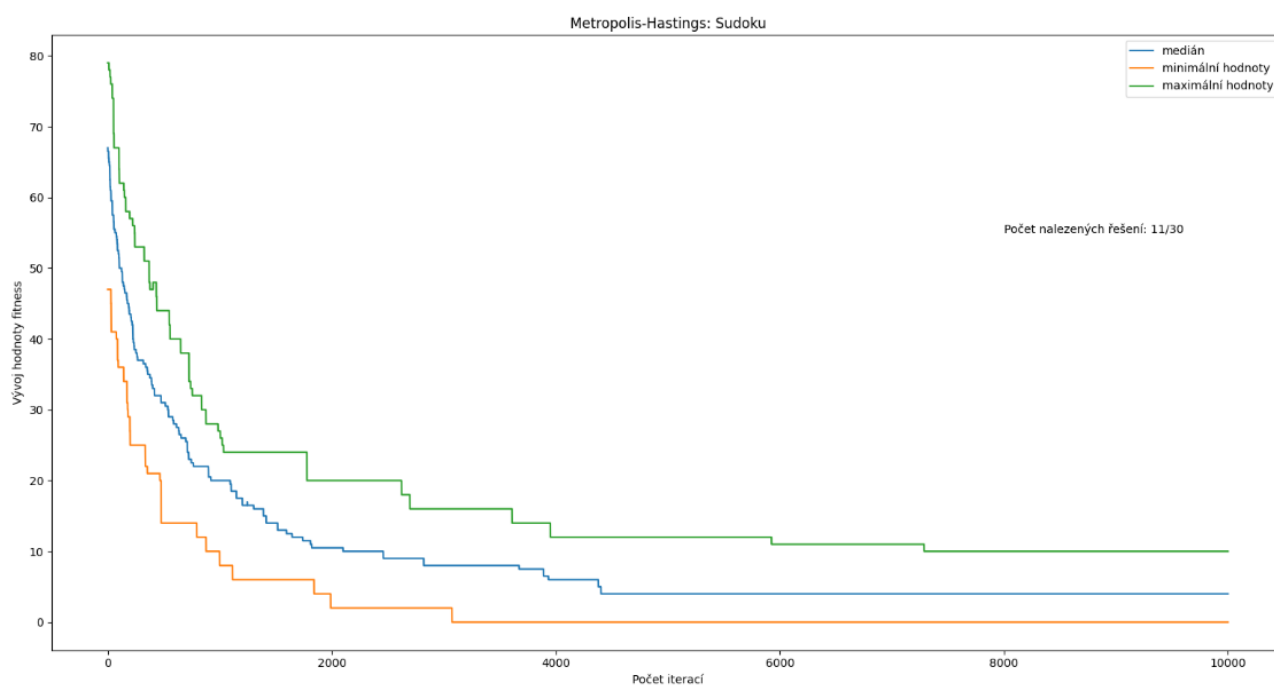
- Nalezení optimální hodnoty pravděpodobnosti přijetí horšího stavu
- Spouštěno s velikostí 9x9, 10,000 iterací, 30 běhů programu, obtížnost 0.5



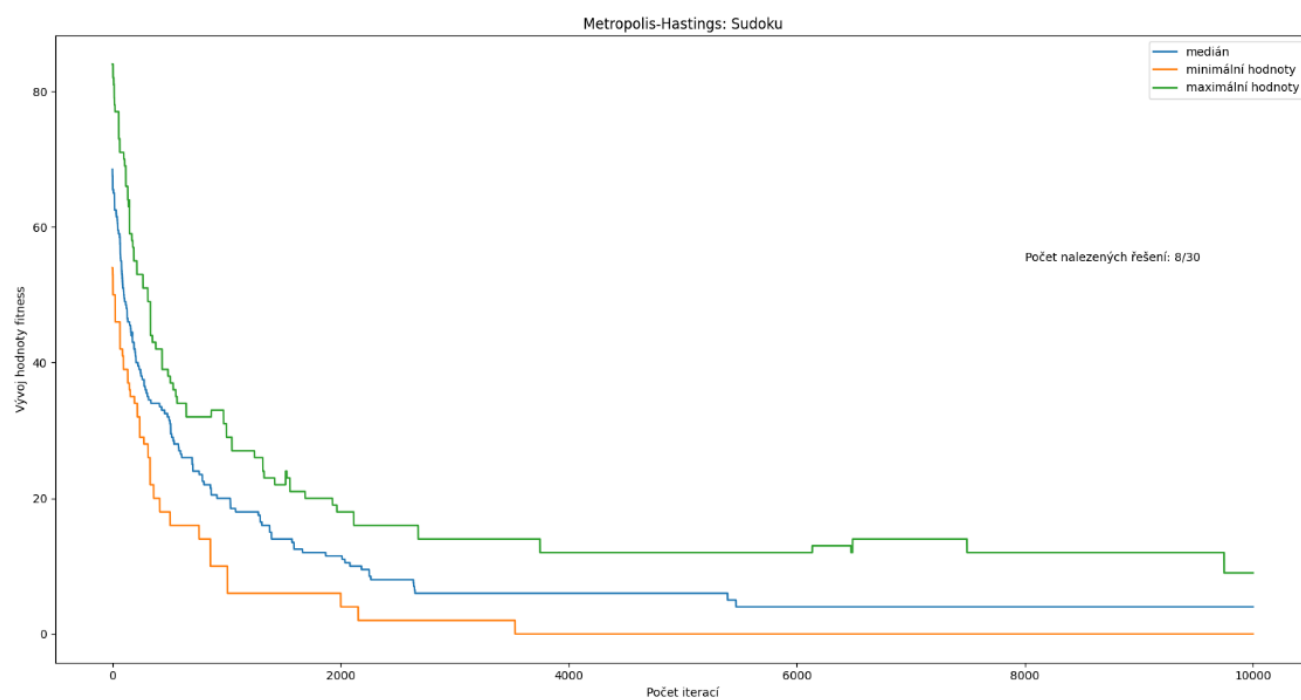
Pravděpodobnost: **0.05**



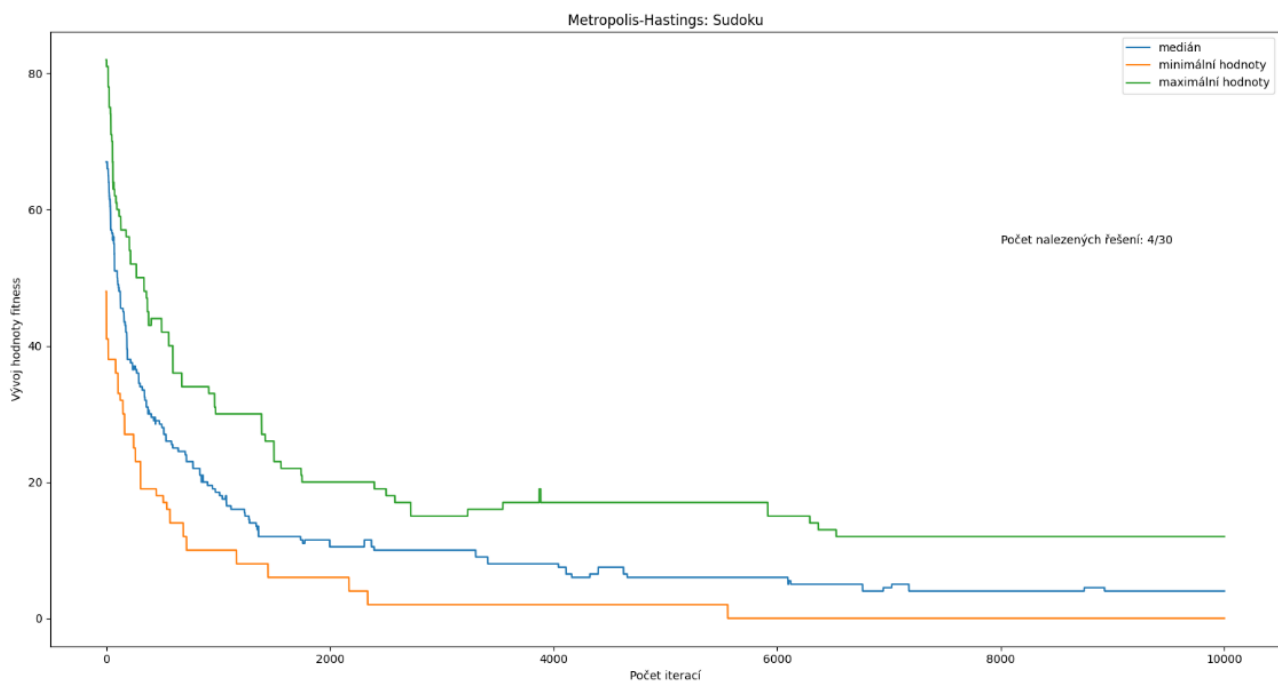
Pravděpodobnost **0.10**



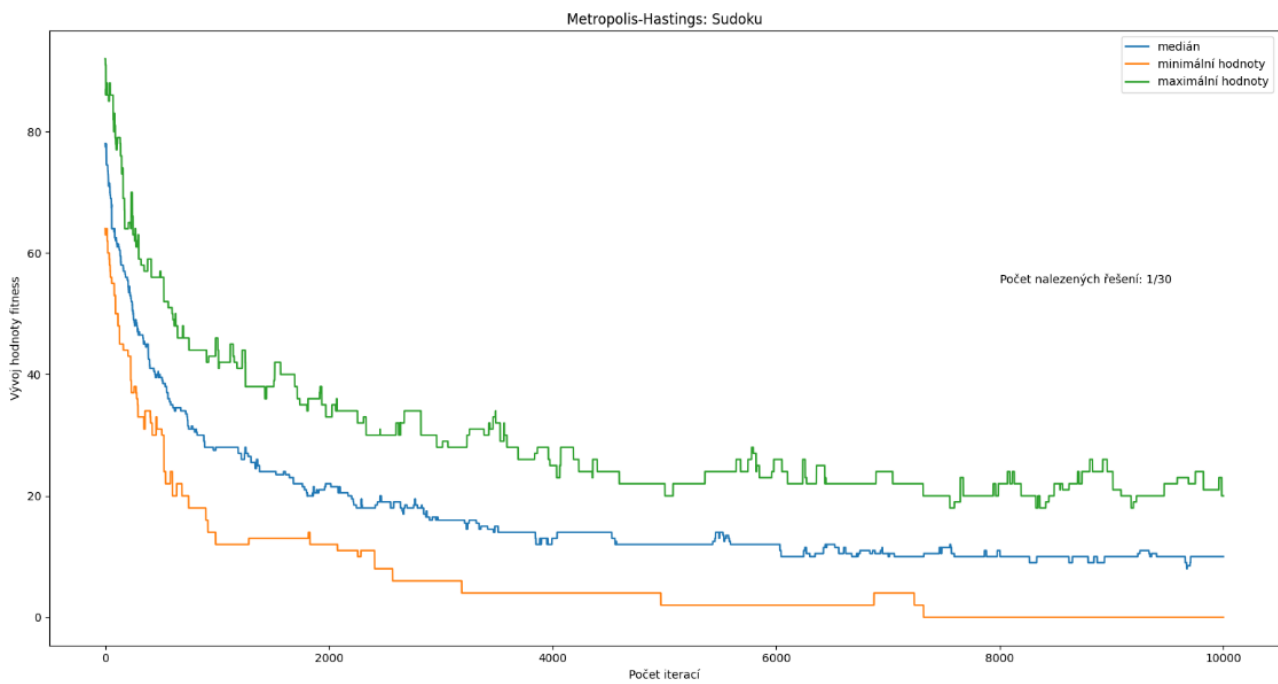
Pravděpodobnost **0.20**



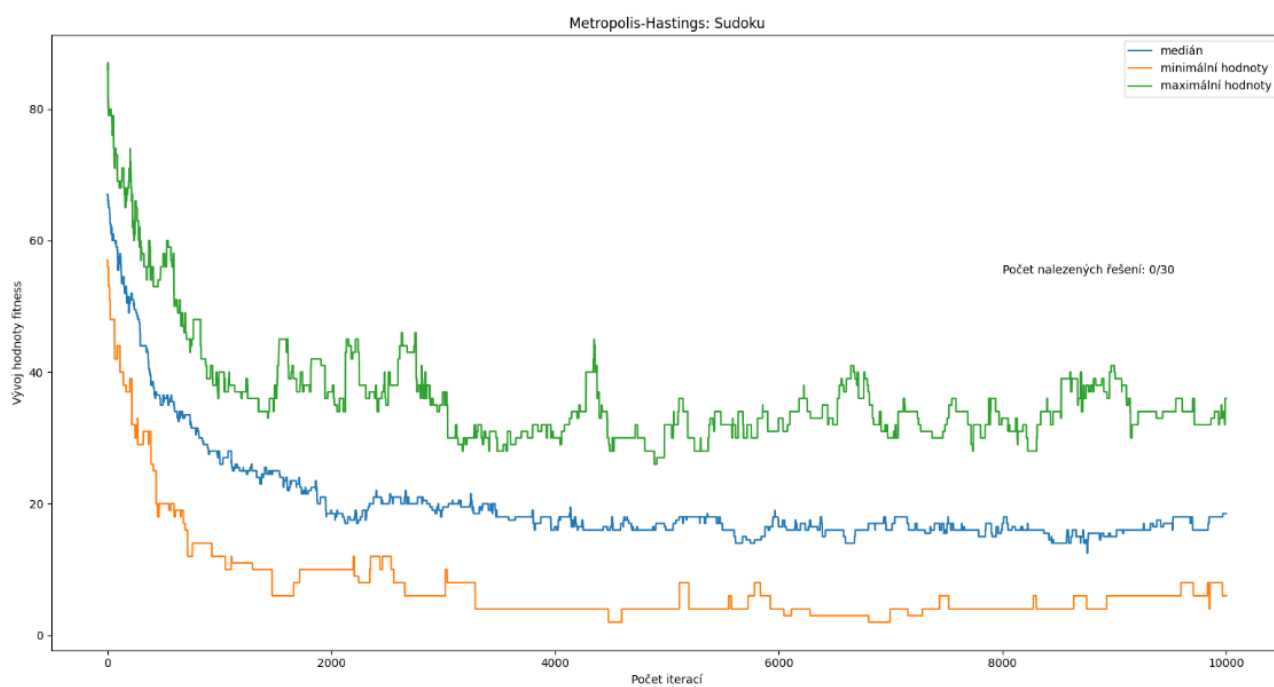
Pravděpodobnost **0.30**



Pravděpodobnost **0.40**

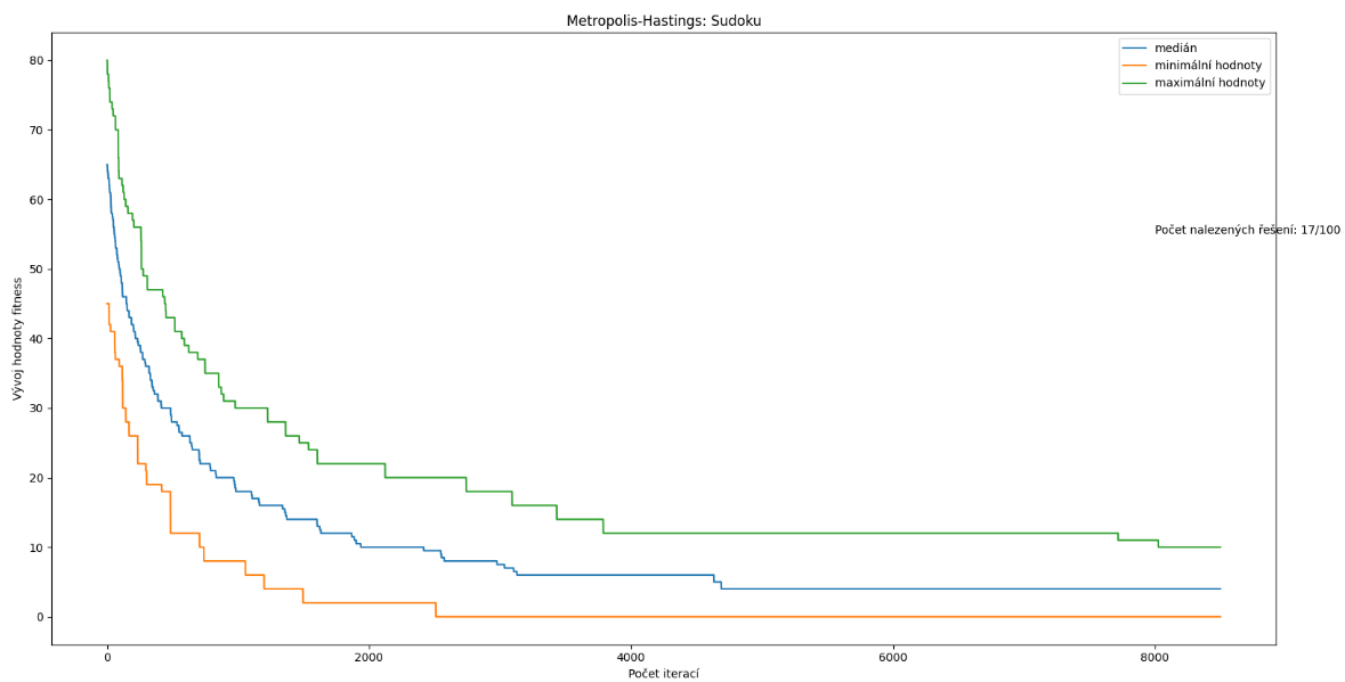


Pravděpodobnost **0.70**

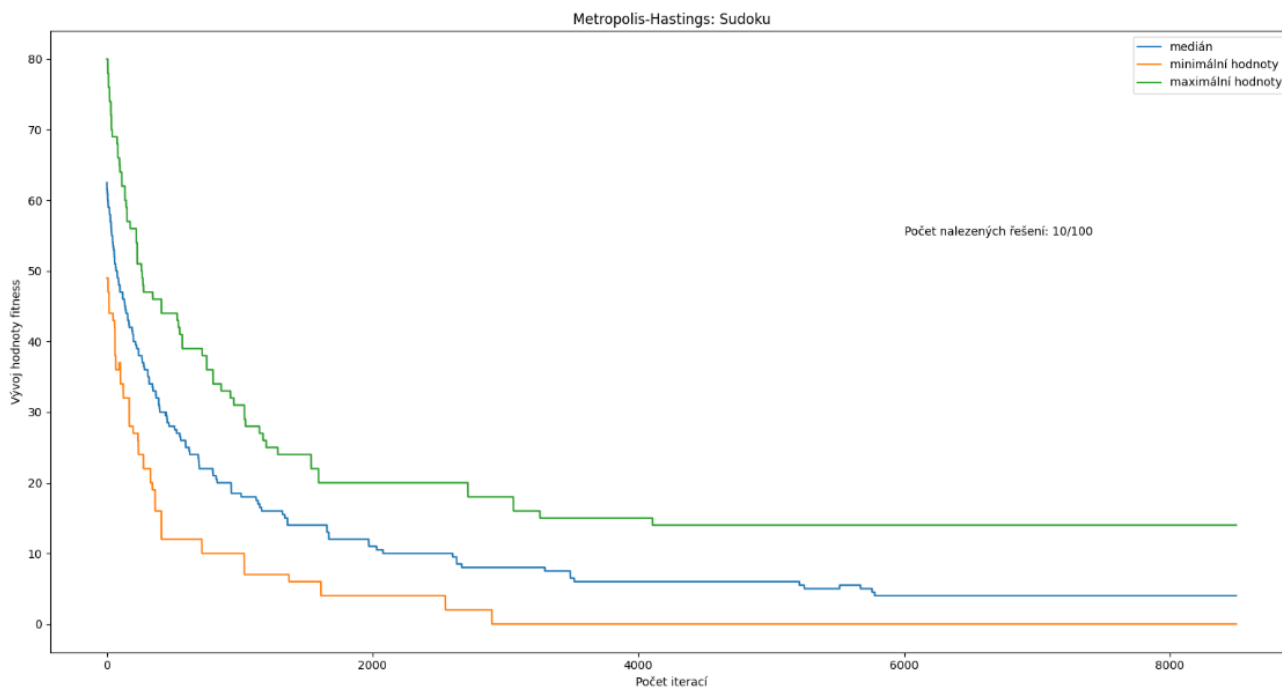


Pravděpodobnost **0.90**

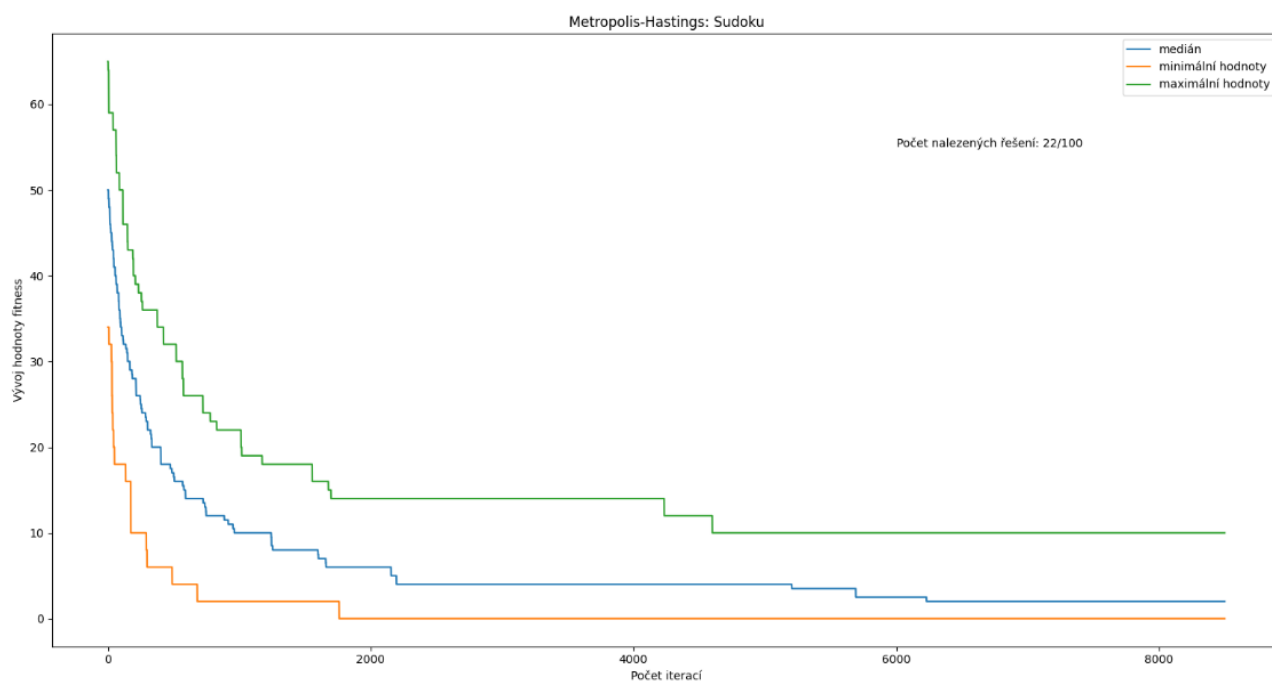
- Změna v parametrech pro upřesnění: 8500 iterací, 100 běhů programu



Pravděpodobnost **0.20**



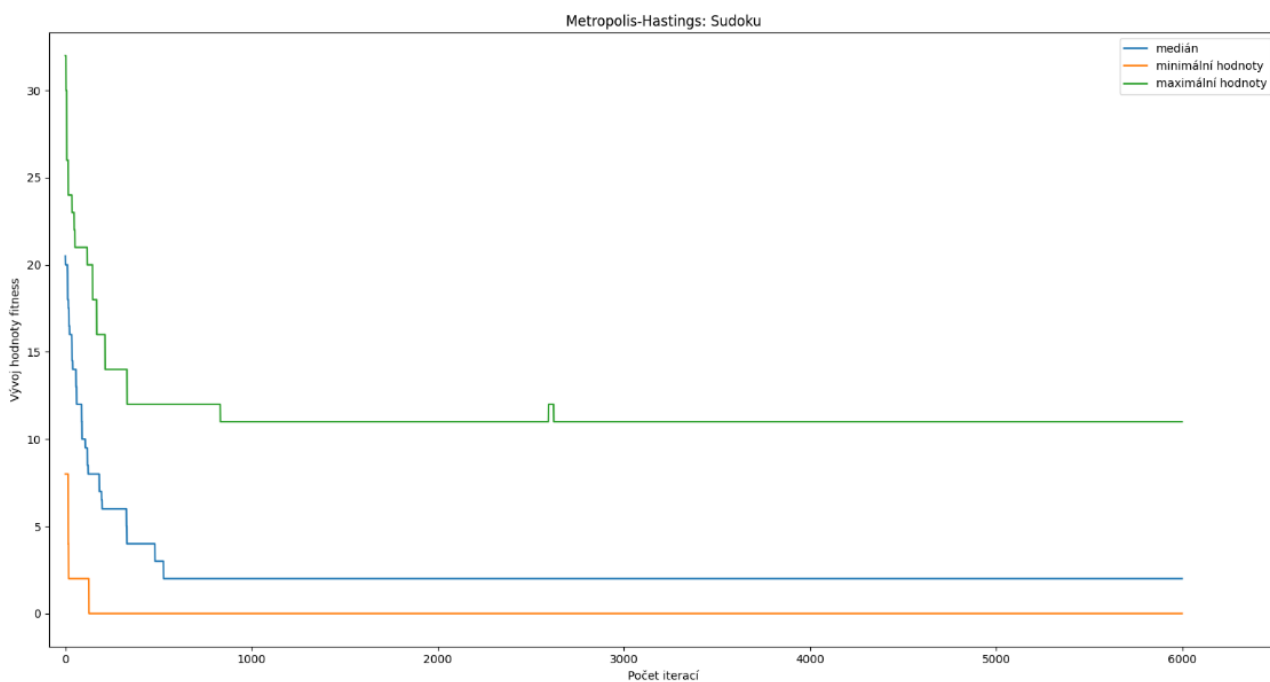
Pravděpodobnost **0.30**



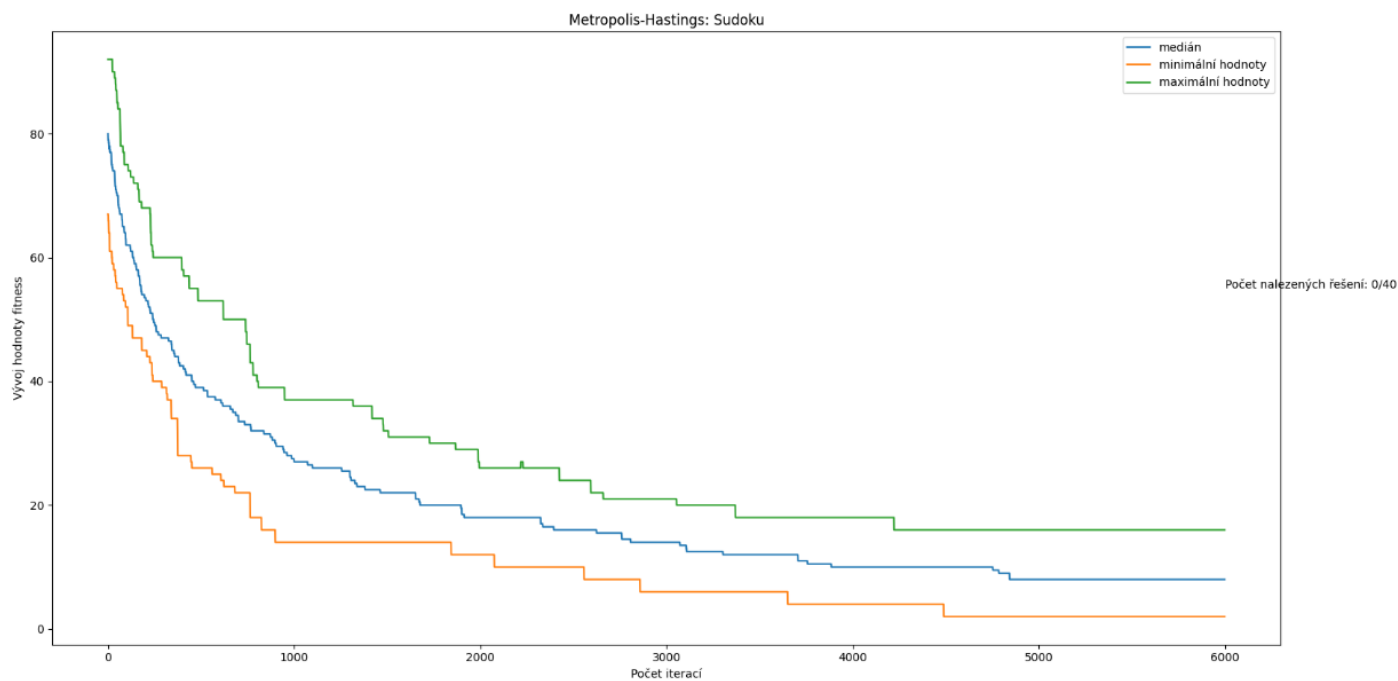
Pravděpodobnost **0.175**



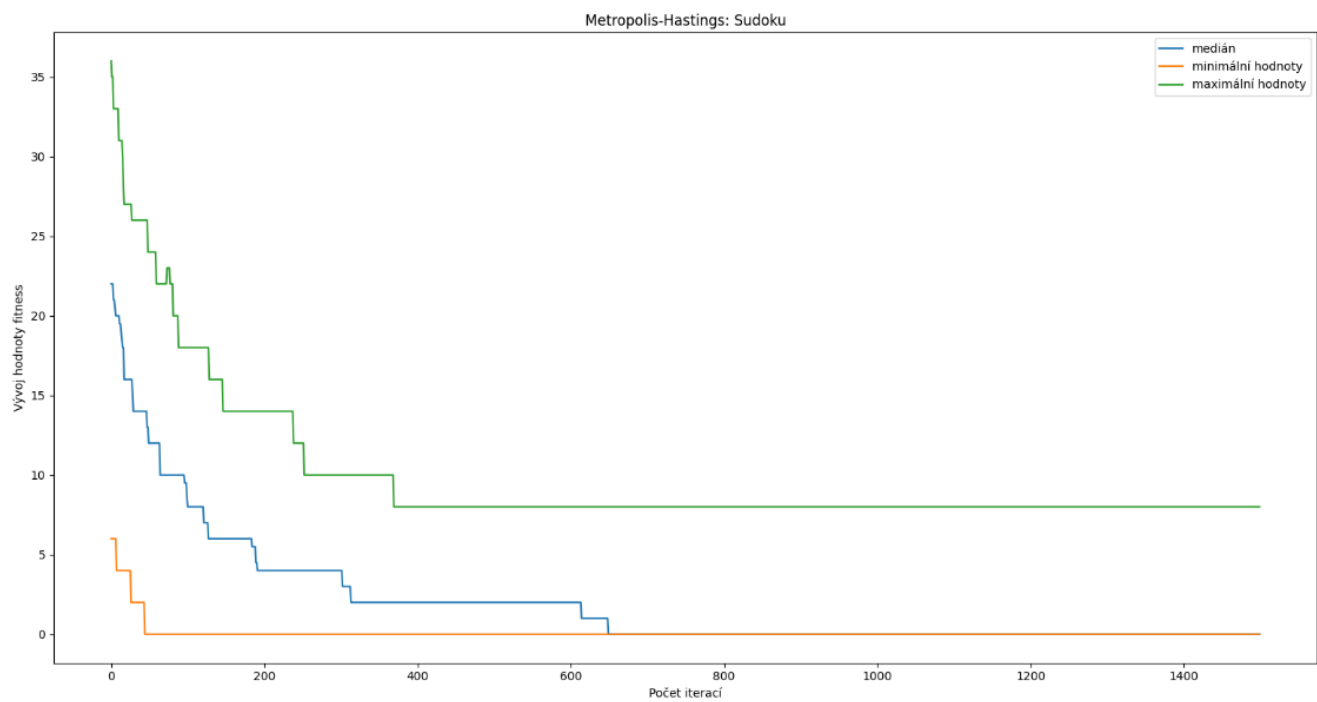
- Efektivita v závislosti na složitosti
- Spuštěno s 6000 iteracemi, 40 běhů, pravděpodobnost 0.18



Složitost: **0.2**, nalezených řešení 17/40



Složitost: **0.8**, nalezených řešení 0/40



Složitost **0.2**, pravděpodobnost **0.3**, nalezených řešení: 55/100

## Závěr

Bylo zjištěno, že vhodná pravděpodobnost pro přijetí horšího stavu se pohybuje v rozmezí od **0.16 - 0.30**. Vypozorována byla závislost mezi touto pravděpodobností a složitostí desky. S rostoucí složitostí je vhodné volit pravděpodobnost přijetí horšího stavu menší, a to **~0.20**. U jednodušších řešení je vhodné volit pravděpodobnost větší, protože algoritmus rychleji konverguje k řešení, ale může snáze uvíznout v lokálních minimech.

Dalším aspektem byl počet iterací, kdy počet potřebných iterací rostl se složitostí. U složitostí kolem **0.2 - 0.35** stačilo k dosažení lepších výsledků kolem **1000-2000** iterací. U složitostí pohybujících se u **0.5** pak kolem **8000-10000** iterací. U složitostí **0.8** pak zhruba **15000**.