

Semidefinitní programování v geometrii počítačového vidění

Pavel Trutman

Vedoucí práce: Ing. Tomáš Pajdla, Ph.D.



Centrum strojového vnímání
Katedra kybernetiky
Fakulta elektrotechnická
České vysoké učení technické v Praze

- ◆ Motivace
- ◆ Metoda momentů a Lasserova hierarchie
- ◆ Implementace
- ◆ Experimenty
 - Poloha a orientace kalibrované kamery (P3P)
 - Poloha a orientace kalibrované kamery s neznámou ohniskovou vzdáleností (P3.5Pf)
- ◆ Přínosy práce

Motivace

- ◆ Mnoho problémů v geometrii počítačového vidění vede na řešení soustav polynomiálních rovnic
- ◆ Tyto soustavy umíme řešit algebraicky (Algoritmus F_4 [1], Automatický generátor [3])
 - jsou vypočtena všechna komplexní řešení, nereálná jsou poté vyřazena
 - nevhodné na přeuročené soustavy na datech se šumem
- ◆ Aplikace metod z polynomiální optimalizace řeší tyto problémy
 - umíme najít pouze reálná řešení
 - do systému lze přidat polynomiální nerovnice
 - lze optimalizovat kritériální polynomiální funkci na prostoru řešení

[1] J.-C. Faugère. A new efficient algorithm for computing gröbner bases (f_4).

[3] Z. Kukelova, M. Bujnak, T. Pajdla. Automatic generator of minimal problem solvers.

Metoda momentů [5]

- ◆ Založena na Lasserrově hierarchii [4]
- ◆ Problém polynomiální optimalizace (nekonvexní)

$$\begin{aligned}
 p^* &= \min_{x \in \mathbb{R}^n} f(x) \\
 \text{s.t. } & g_i(x) \geq 0 \quad (i = 1, \dots, k) \\
 & h_j(x) = 0 \quad (j = 1, \dots, l)
 \end{aligned} \tag{1}$$

- ◆ Každý monom nahradíme novou proměnnou: $\ell_y(x^\alpha) = y_\alpha$
- ◆ Semidefinitní problém nekonečné dimenze (konvexní)

$$\begin{aligned}
 p^* &= \min_{y \in \mathbb{R}^{\mathbb{N}^n}} \text{vec}(f)^\top y \\
 \text{s.t. } & y_{0\dots 0} = 1 \\
 & M(y) \succeq 0 \\
 & M(g_i y) \succeq 0 \quad (i = 1, \dots, k) \\
 & \text{vec}(h)^\top y = 0 \quad \forall h \in \langle h_1, \dots, h_l \rangle
 \end{aligned} \tag{2}$$

[4] J. B. Lasserre. Global optimization with polynomials and the problem of moments.

[5] J. B. Lasserre, M. Laurent, and P. Rostalski. Semidefinite characterization and computation of zero-dimensional real radical ideals.

Metoda momentů [5]

- ◆ Založena na Lasserrově hierarchii [4]
- ◆ Problém polynomiální optimalizace (nekonvexní)

$$\begin{aligned} p^* &= \min_{x \in \mathbb{R}^n} f(x) \\ \text{s.t. } & g_i(x) \geq 0 \quad (i = 1, \dots, k) \\ & h_j(x) = 0 \quad (j = 1, \dots, l) \end{aligned} \quad (1)$$

- ◆ Každý monom nahradíme novou proměnnou: $\ell_y(x^\alpha) = y_\alpha$
- ◆ Semidefinitní problém dimenze $\binom{n+2r}{n}$ pro stupeň relaxace $r \in \mathbb{N}$

$$\begin{aligned} p_r^* &= \min_{y \in \mathbb{R}^{\binom{n+2r}{n}}} \text{vec}(f)^\top y \\ \text{s.t. } & y_{0\dots 0} = 1 \\ & M_r(y) \succeq 0 \\ & M_{r - \lceil \frac{\deg(g_i)}{2} \rceil}(g_i y) \succeq 0 \quad (i = 1, \dots, k) \\ & \text{vec}(h)^\top y = 0 \quad \forall h \in \{h_j x^\alpha \mid j = 1, \dots, l, |\alpha| \leq 2r - \deg(h_j)\} \end{aligned} \quad (3)$$

- ◆ Konvergence zajištěna

$$p_r^* \leq p_{r+1}^* \leq p^* \quad \lim_{r \rightarrow +\infty} p_r^* = p^* \quad (4)$$

[4] J. B. Lasserre. Global optimization with polynomials and the problem of moments.

[5] J. B. Lasserre, M. Laurent, and P. Rostalski. Semidefinite characterization and computation of zero-dimensional real radical ideals.

Implementace

◆ Balíček Polyopt

- programovací jazyk Python
- implementace momentové metody
- implementace vlastního nástroje na řešení semidefinitních problémů: algoritmus vnitřních bodů s využitím bariérové funkce, dle [8]

◆ Implementace momentové metody v MATLABu

- využití nástroje MOSEK [7] na řešení semidefinitních problémů
- využití toolboxu YALMIP [6] jako interface

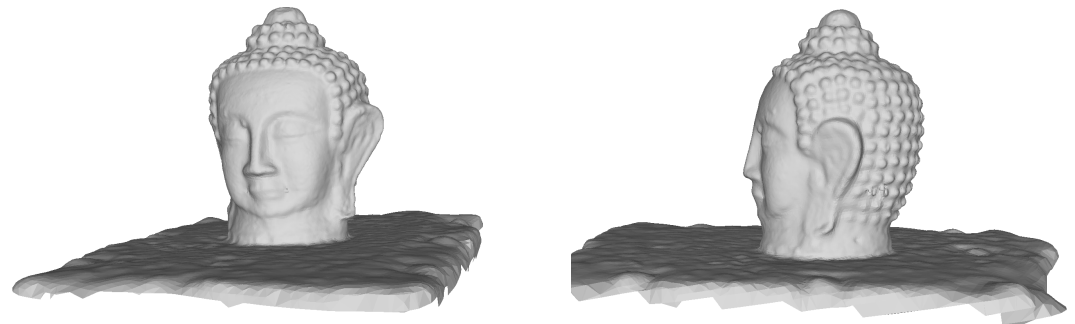
[6] Johan Löfberg. YALMIP: A toolbox for modeling and optimization in MATLAB.

[7] MOSEK ApS. The MOSEK optimization toolbox for MATLAB manual. Version 7.1 (Revision 28).

[8] Y. Nesterov. Introductory lectures on convex optimization: A basic course.

Experimenty

- ◆ Řešení soustav polynomiálních rovnic
- ◆ Aplikace na minimálních problémech z geometrie počítačového vidění
 - Poloha a orientace kalibrované kamery (P3P)
 - Poloha a orientace kalibrované kamery s neznámou ohniskovou vzdáleností (P3.5Pf)
- ◆ Testování na reálné 3D scéně
 - robustně zrekonstruovaná
 - 67 kamer, 145 001 bodů v prostoru
- ◆ Porovnání různých implementací
 - čistě algebraické řešení vygenerované Automatickým generátorem [3]
 - balíček Polyopt
 - implementace v MATLABu s využitím nástroje MOSEK [7]
 - optimalizační nástroj Gloptipoly [2]








[2] D. Henrion, J. B. Lasserre, and J. Löfberg. Gloptipoly 3: Moments, optimization and semidefinite programming.

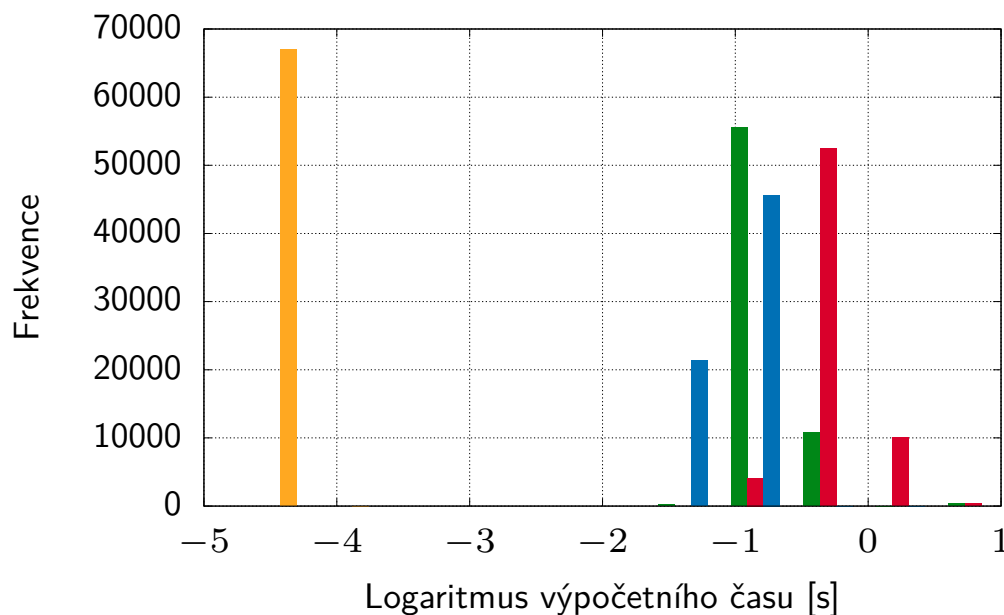
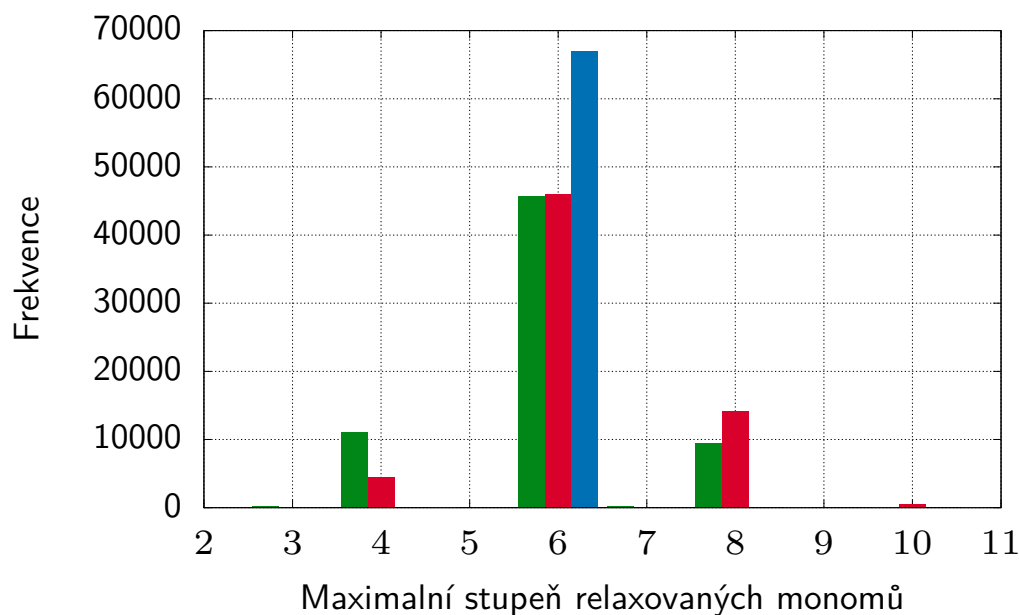
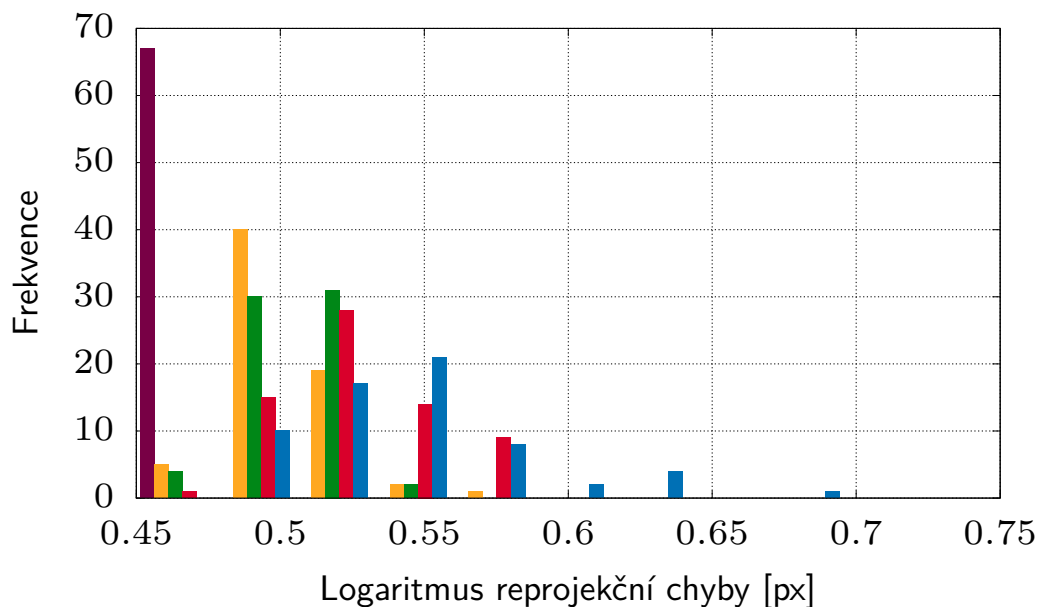
[3] Z. Kukelova, M. Bujnak, T. Pajdla. Automatic generator of minimal problem solvers.

[7] MOSEK ApS. The MOSEK optimization toolbox for MATLAB manual. Version 7.1 (Revision 28).

Poloha a orientace kalibrované kamery (P3P)






- ♦ Jedna rovnice čtvrtého stupně
- ♦ Jedna neznámá
- ♦ Vybrán nejlepší výsledek z 1000 různých konfigurací pro každou z 67 kamer (RANSAC-like)
- ♦ 60 % řešení je reálných

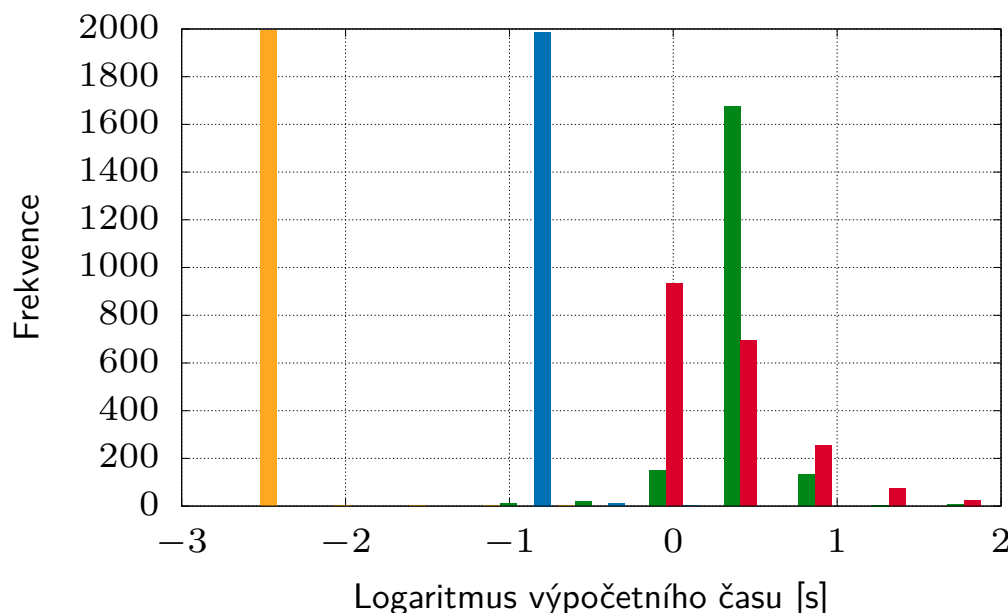
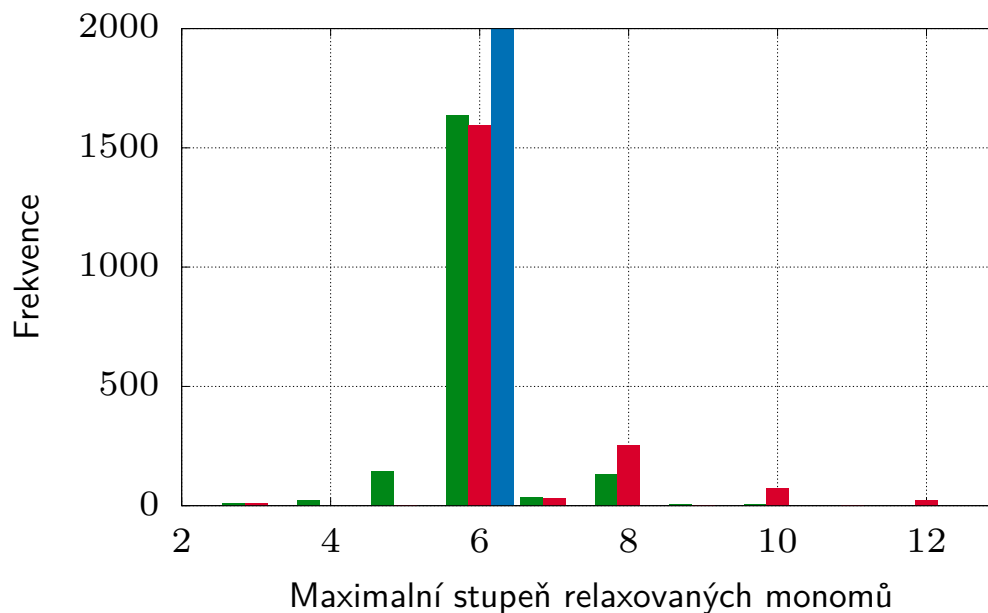
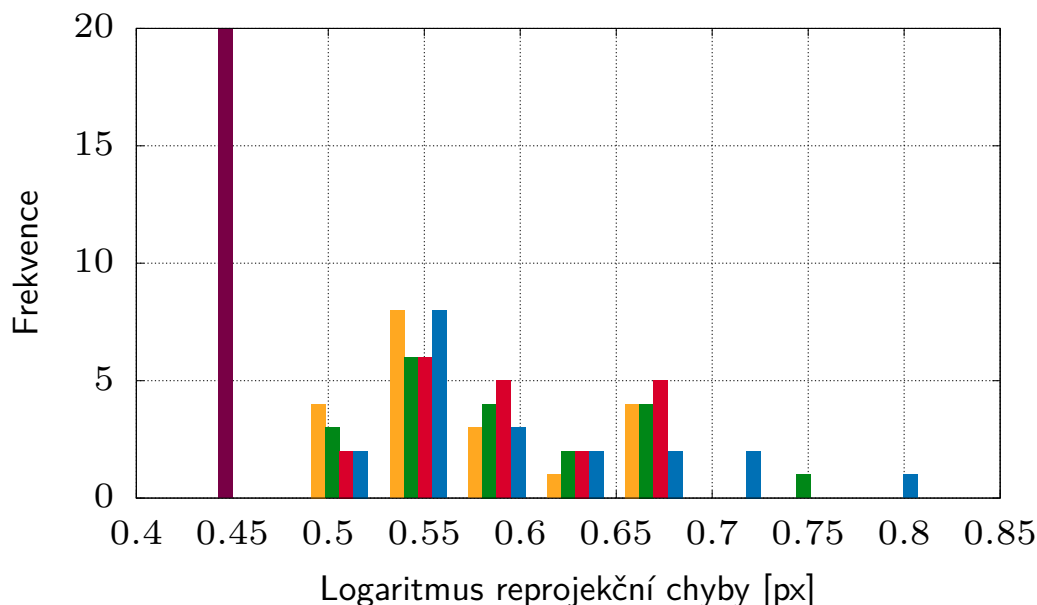
Reference 
Automatický generátor [3] 
Polyopt 
Implementace v MATLABu s nástrojem MOSEK [7] 
Gloptipoly [2] 



Poloha a orientace kalibrované kamery s neznámou ohniskovou vzdáleností (P3.5Pf)

- ◆ Devět rovnic třetího stupně
- ◆ Čtyři neznámé
- ◆ Vybrán nejlepší výsledek z 100 různých konfigurací pro každou z 20 vybraných kamer (RANSAC-like)
- ◆ 48 % řešení je reálných

Reference 
Automatický generátor [3] 
Polyopt 
Implementace v MATLABu s nástrojem MOSEK [7] 
Gloptipoly [2] 



Přínosy práce

- ◆ Metoda momentů
 - rozbor a implementace metody v Pythonu a MATLABu
 - použití metody na problémy z geometrie počítačového vidění
 - aplikace na úlohy z robotiky: řešení inverzní kinematické úlohy
- ◆ Nástroj na řešení semidefinitních problémů
 - implementace v Pythonu
 - porozumění typům semidefinitních problémů generovaných Lasserrovou hierarchií
 - využití implementace v rámci formální verifikace algoritmů v konvexní optimalizaci, D. Henrion, LAAS–CNRS v Toulouse
- ◆ Srovnání implementace momentové metody se současnými nástroji
 - stabilita odpovídá algebraickým metodám
 - nemůže konkurovat algebraickým metodám v rychlosti

Děkuji za pozornost

Použitá literatura

- [1] Jean-Charles Faugère. A new efficient algorithm for computing gröbner bases (f_4). *Journal of pure and applied algebra*, 139(1–3):61–88, July 1999.
- [2] Didier Henrion, Jean-Bernard Lasserre, and Johan Löfberg. Gloptipoly 3: Moments, optimization and semidefinite programming. *Optimization Methods Software*, 24(4–5):761–779, August 2009.
- [3] Zuzana Kukelova, Martin Bujnak, and Tomas Pajdla. Automatic generator of minimal problem solvers. In *Proceedings of The 10th European Conference on Computer Vision*, ECCV 2008, October 12–18 2008.
- [4] Jean B. Lasserre. Global optimization with polynomials and the problem of moments. *Society for Industrial and Applied Mathematics Journal on Optimization*, 11:796–817, 2001.
- [5] Jean Bernard Lasserre, Monique Laurent, and Philipp Rostalski. Semidefinite characterization and computation of zero-dimensional real radical ideals. *Foundations of Computational Mathematics*, 8(5):607–647, October 2008.
- [6] Johan Löfberg. YALMIP : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.
- [7] MOSEK ApS. *The MOSEK optimization toolbox for MATLAB manual. Version 7.1 (Revision 28)*, 2015. <http://docs.mosek.com/7.1/toolbox/index.html> [Online; accessed 2017-04-25].
- [8] Yurii Nesterov. *Introductory lectures on convex optimization : A basic course*. Springer, 2004.

Parametrizace P3P problému

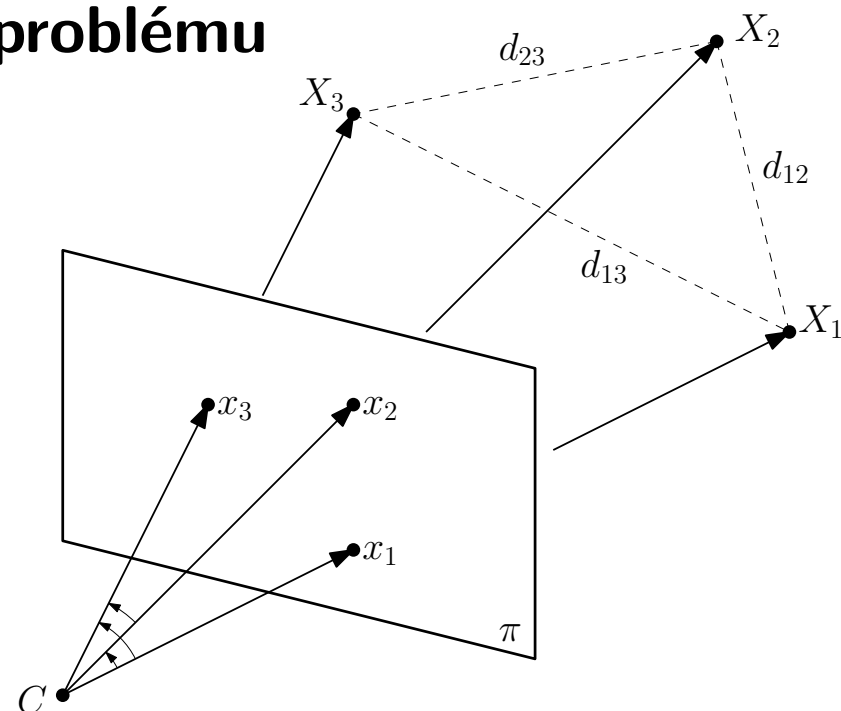
- ◆ Známe 3D body X_i a jejich projekce x_i
- ◆ Napočítáme vzdálenosti mezi 3D body d_{ij} a kosiny úhlů paprsků c_{ij}
- ◆ Z kosinovy věty dostáváme tři rovnice druhého stupně ve třech neznámých z_i

$$d_{ij}^2 = z_i^2 + z_j^2 - 2z_i z_j c_{ij} \quad (5)$$

- ◆ Manipulací lze upravit na jednu rovnici čtvrtého stupně v jedné neznámé ξ

$$a_4 \xi^4 + a_3 \xi^3 + a_2 \xi^2 + a_1 \xi + a_0 = 0 \quad (6)$$

- ◆ Každý parametr a_k je polynomiální funkcí hodnot d_{ij} a c_{ij}



Obrázek 1. Náskres P3P problému.

r	Soustava (5)	Soustava (6)
1	10	—
2	35	5
3	84	7

Tabulka 1. Počet neznámých řešeného semidefinitního problému.

Počet nalezených reálných řešení

- ◆ Implementace založené na metodě momentů nenaleznou všechna reálná řešení

Implementace	Počet nalezených reálných řešení	Procento nalezených reálných řešení
Automatický generátor [3]	158850	100.0 %
Polyopt	129394	81.5 %
Implementace v MATLABu s nástrojem MOSEK [7]	141702	89.2 %
Gloptipoly [2]	71934	45.3 %

Tabulka 1. Počet nalezených reálných řešení pro P3P problém.

Implementace	Počet nalezených reálných řešení	Procento nalezených reálných řešení
Automatický generátor [3]	9608	100.0 %
Polyopt	8110	84.4 %
Implementace v MATLABu s nástrojem MOSEK [7]	8698	90.5 %
Gloptipoly [2]	5907	61.5 %

Tabulka 2. Počet nalezených reálných řešení pro P3.5Pf problém.

Numerické problémy v balíčku Polyopt

- ◆ Nástroj na řešení semidefinitních problémů nevyřeší zadaný semidefinitní problém
- ◆ Metoda založena na minimalizaci bariérové funkce

$$F(y) = -\ln \det(A(y)) \quad (7)$$

- ◆ $A(y)$ je semidefinitně pozitivní matice

$$A(y) = A_0 + \sum_{k=1}^m A_k y_k \quad (8)$$

- ◆ Potřeba spočítat první a druhé derivace bariérové funkce

$$\frac{\partial F}{\partial y_i}(y) = -\operatorname{tr}(A(y)^{-1} A_i) \quad (9)$$

$$\frac{\partial^2 F}{\partial y_i \partial y_j}(y) = \operatorname{tr}\left((A(y)^{-1} A_i)(A(y)^{-1} A_j)\right) \quad (10)$$

- ◆ Použití upravené Newtonovy metody \rightarrow vyčíslení inverze Hessianovy matice v bodě

Momentová matice

- ◆ Momentová matice $M(y)$ pro $y \in \mathbb{R}^{N^n}$ má tvar

$$M(y)_{\alpha,\beta} = y_{\alpha+\beta} \quad (11)$$

- ◆ Příklad pro $n = 2$:

[illegible]

Lokalizační matice

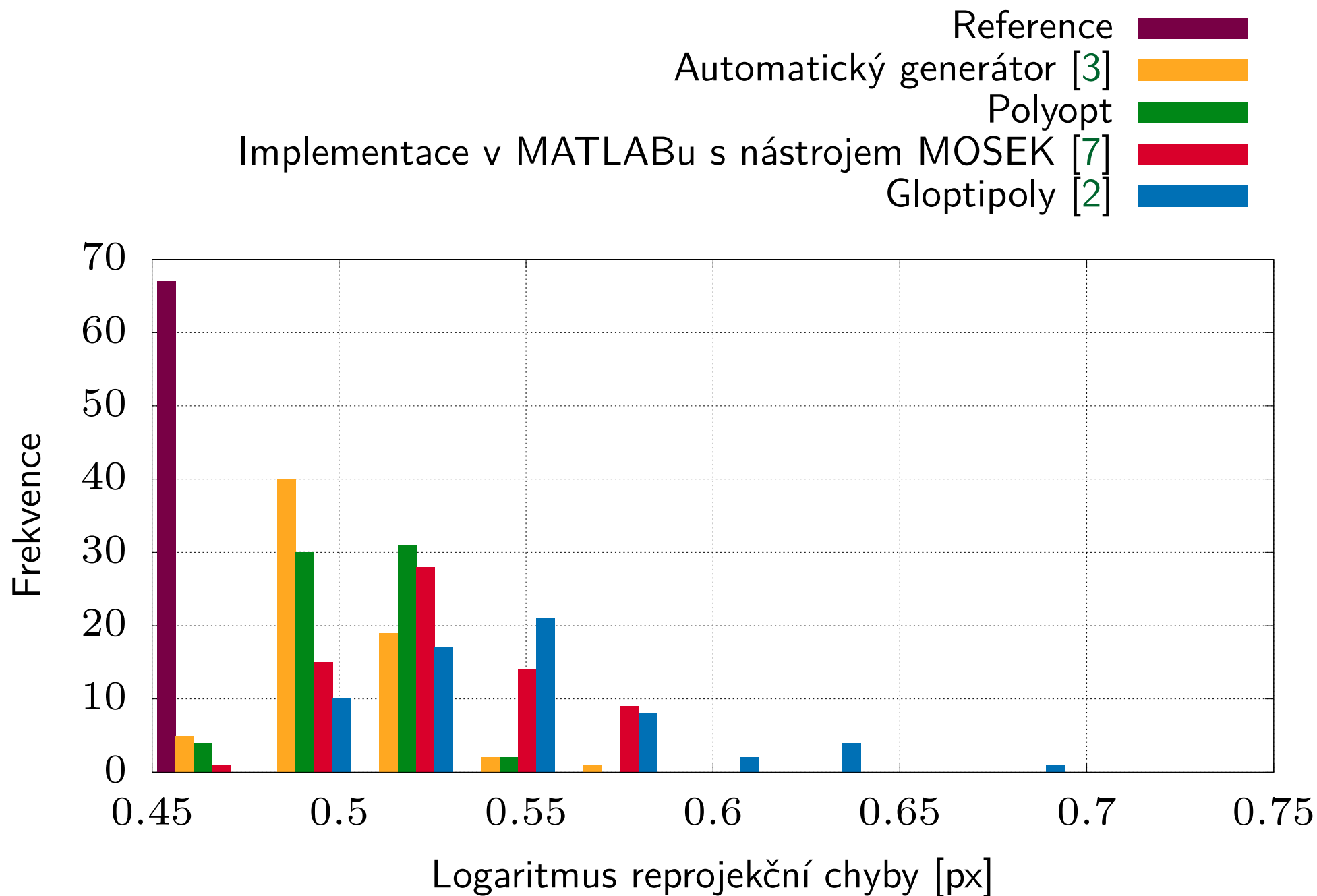
- ◆ Lokalizační matice $M(qy)$ pro $y \in \mathbb{R}^{\mathbb{N}^n}$ a polynom $q(x)$ má tvar

$$M(qy)_{\alpha,\beta} = \sum_{\gamma \in \mathbb{N}^n} q_{\gamma} y_{\alpha+\beta+\gamma} \quad (13)$$

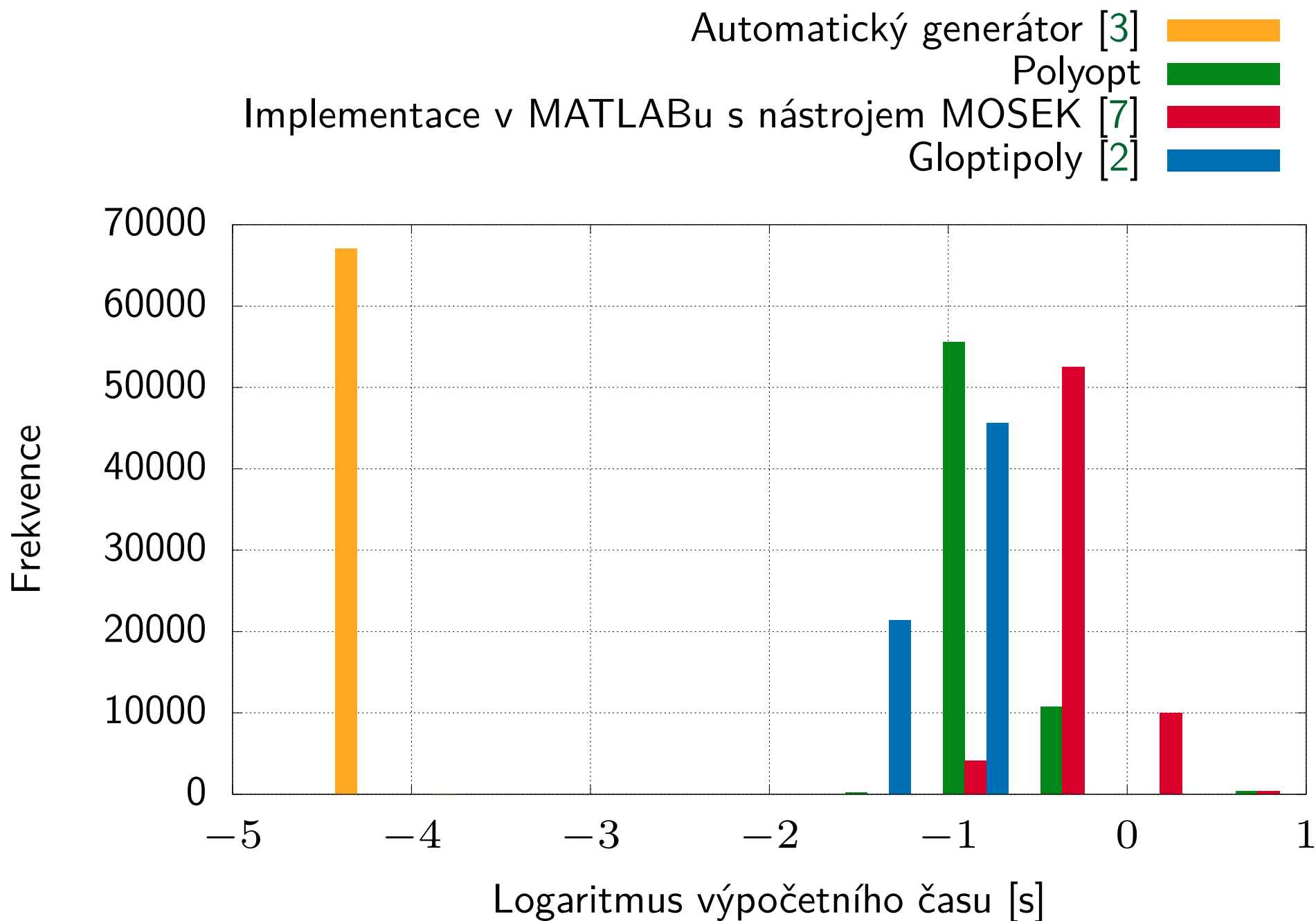
- ◆ Příklad pro $n = 2$ a $q(x) = 5x_1^2 + 8x_2 + 3$:

$$M(qy) = 5 \begin{bmatrix} y_{20} & y_{30} & y_{21} & \cdots \\ y_{30} & y_{40} & y_{31} & \cdots \\ y_{21} & y_{31} & y_{22} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} + 8 \begin{bmatrix} y_{01} & y_{11} & y_{02} & \cdots \\ y_{11} & y_{21} & y_{12} & \cdots \\ y_{02} & y_{12} & y_{03} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} + 3 \begin{bmatrix} y_{00} & y_{10} & y_{01} & \cdots \\ y_{10} & y_{20} & y_{11} & \cdots \\ y_{01} & y_{11} & y_{02} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad (14)$$

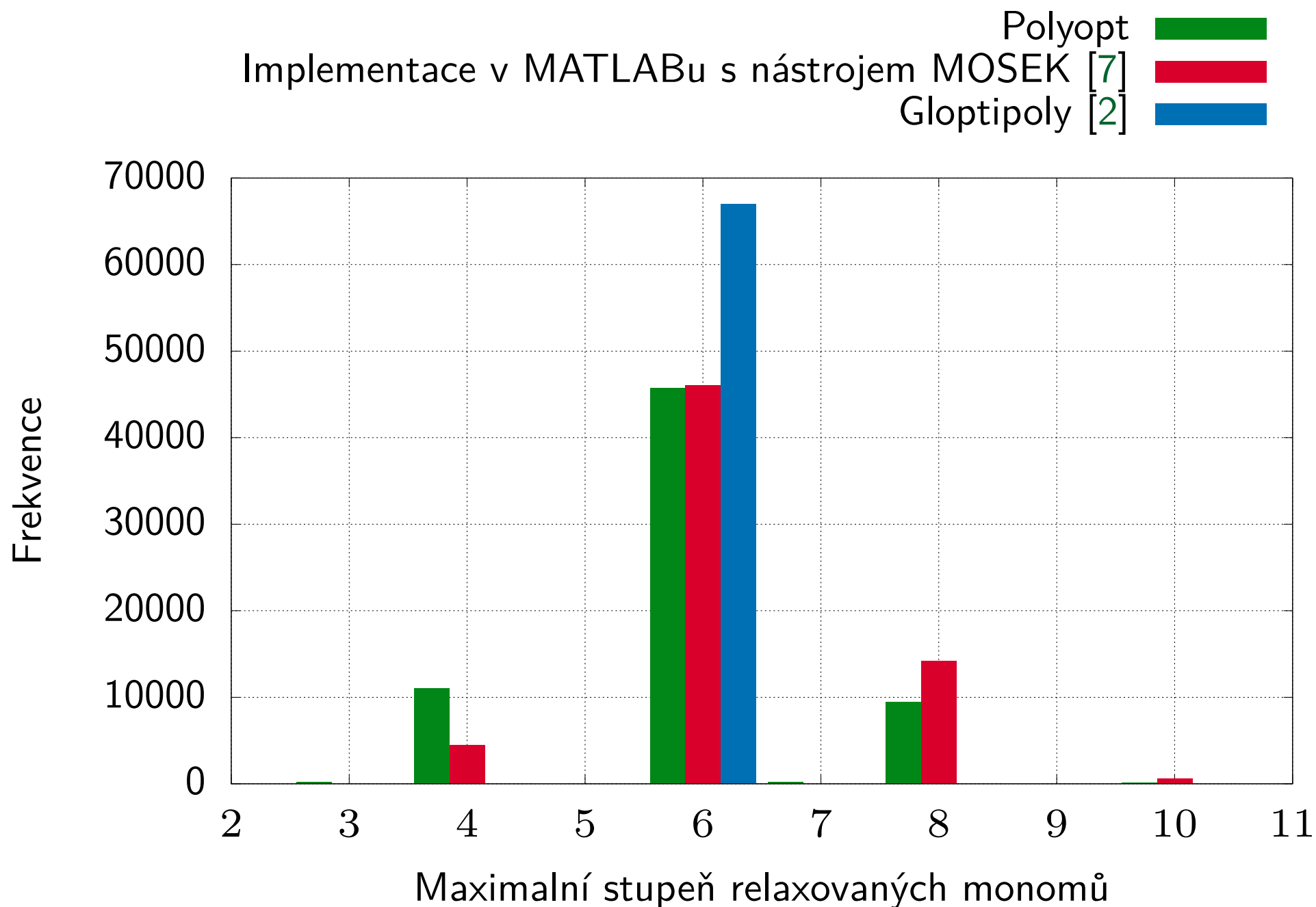
P3P: Histogram reprojekčních chyb



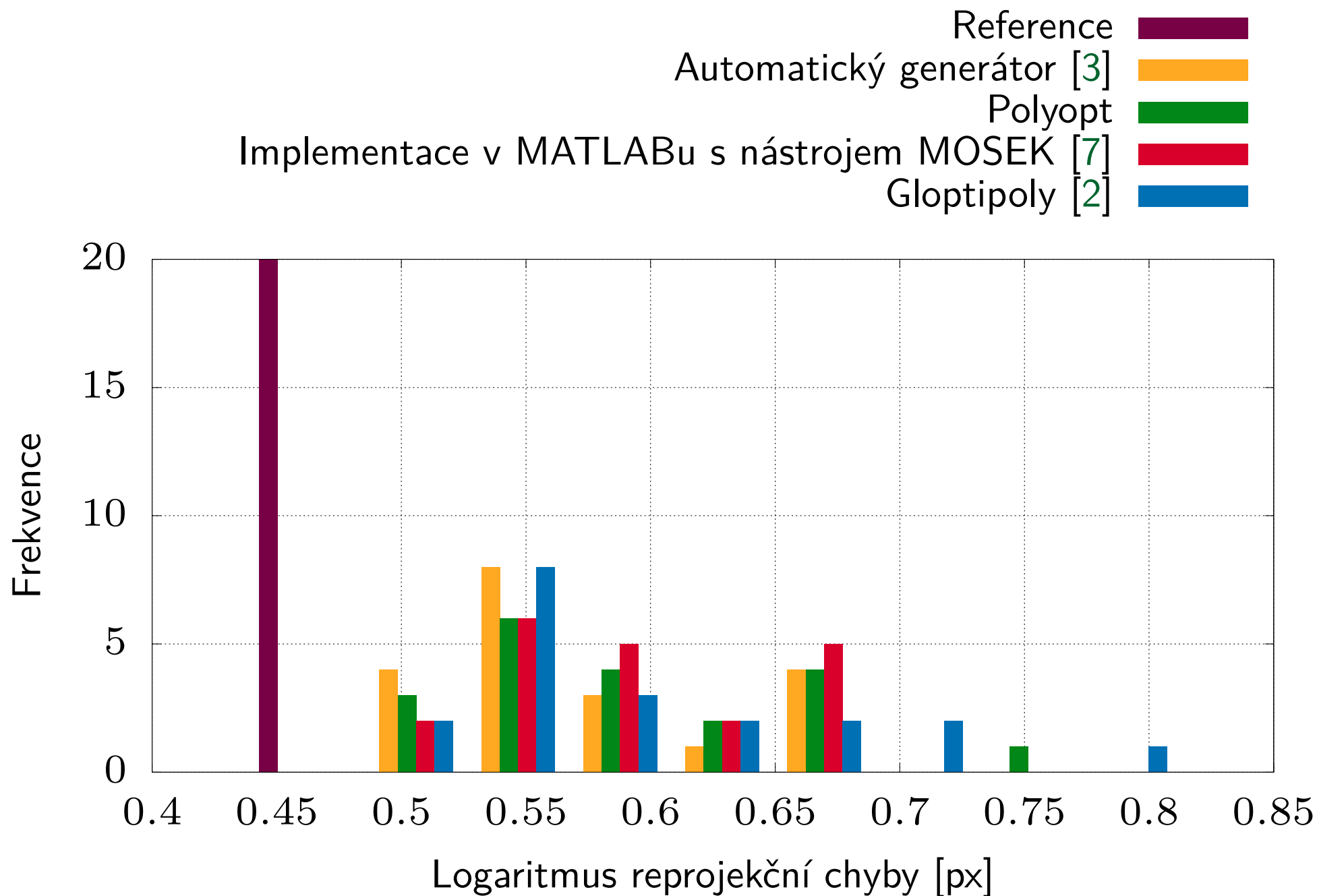
P3P: Histogram výpočetních časů



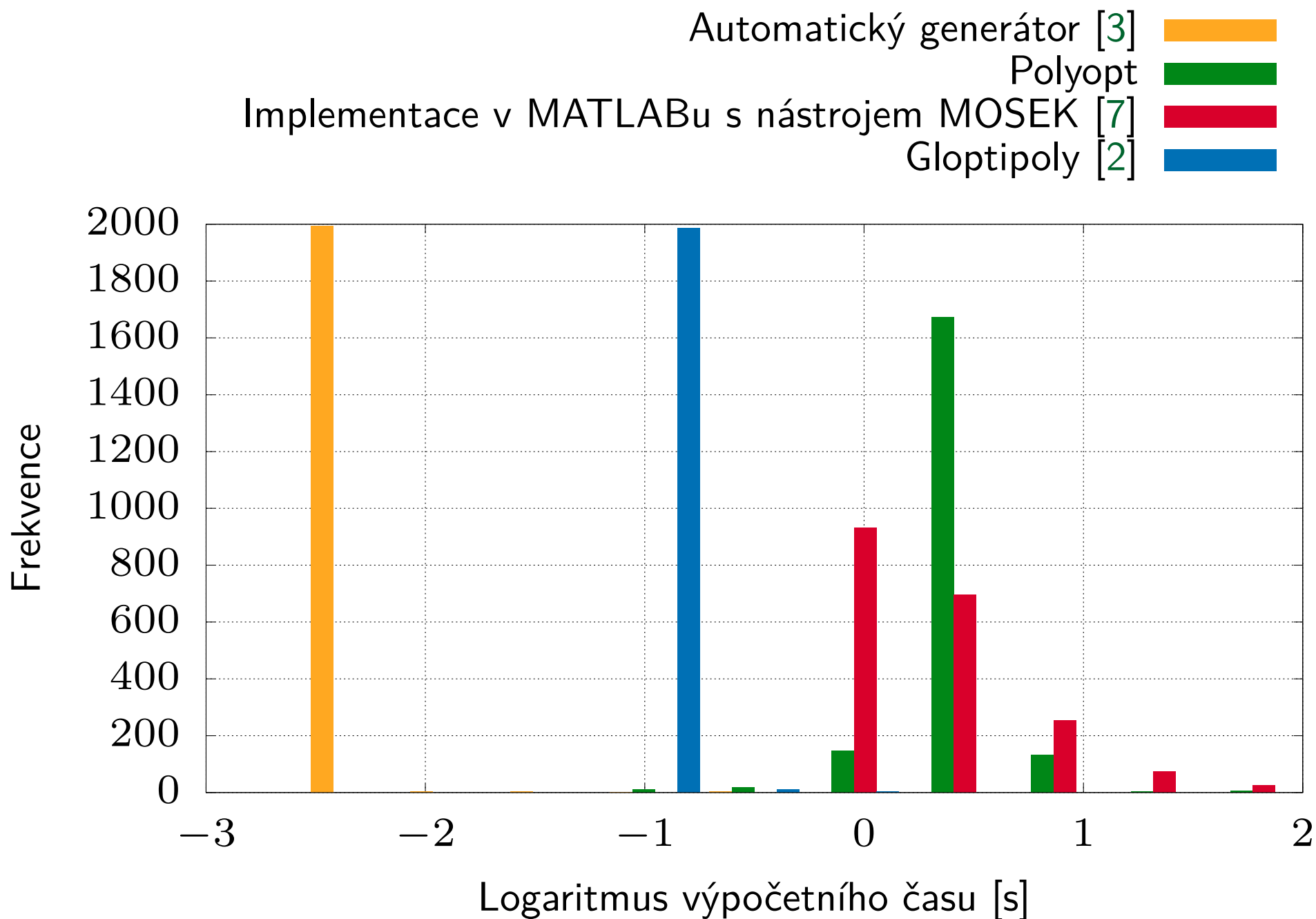
P3P: Histogram stupňů relaxovaných monomů



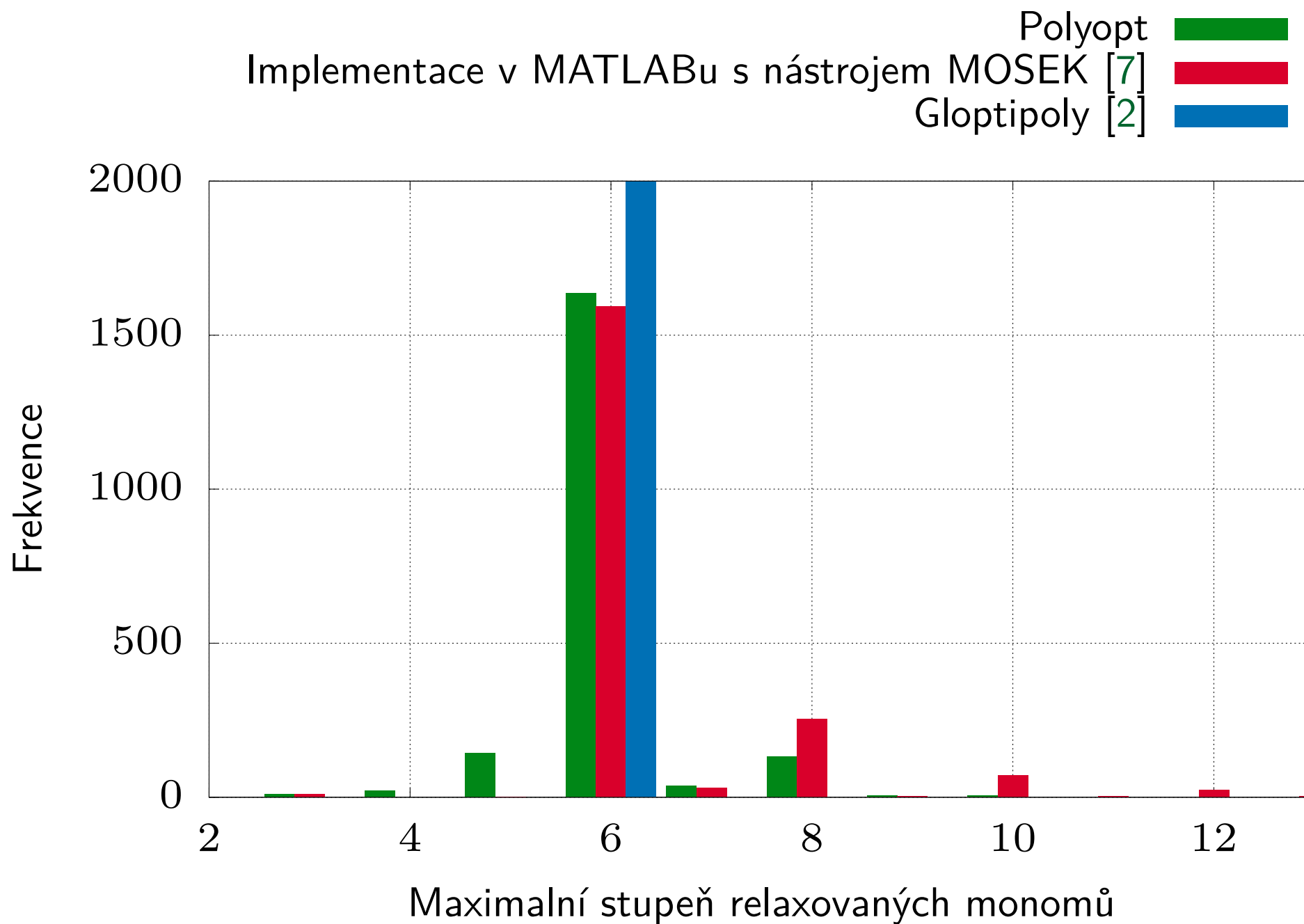
P3.5Pf: Histogram reprojekčních chyb



P3.5Pf: Histogram výpočetních časů



P3.5Pf: Histogram stupňů relaxovaných monomů



P3.5Pf: Histogram relativních chyb ohniskových vzdáleností

