



ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta stavební
Katedra mechaniky

Paralelní evoluční strategie pro vyhodnocení miniMax kritéria

A Parallel Evolutionary Strategy for miniMax Evaluation

Soutěžní práce

Studijní program: Stavební inženýrství
Studijní obor: Materiálové inženýrství

Vedoucí práce: doc. Ing. Matěj Lepš, Ph.D.

Bc. Eva Myšáková

Praha 2013

Abstrakt: *Návrh experimentů - the design of (computer) experiments (DoE) - tvoří základní součást každého meta-modelu. Jako takový musí mít určité vlastnosti, důraz je kladen především na dvě - ortogonalitu a rovnoměrnost pokrytí. Na splnění druhého z požadavků se v posledním desetiletí zaměřilo několik metod. Zajímavou oblast pro výzkum tvoří jedno z kritérií používaných pro hodnocení rovnoměrnosti pokrytí - kritérium miniMax (mM). Toto kritérium se zdá být nejvhodnějším pro praktické účely; jeho vyčíslení ve vyšších dimenzích je však extrémně výpočetně náročné. Proto je v této práci představena nová metoda. Je založena na použití algoritmu evoluční strategie (ES), pomocí kterého jsou nalezeny potenciální středy největších nepokrytých koulí. Hodnota mM je poté poloměr největší z těchto koulí. Navrhovaná metoda tedy nezaručuje přesné řešení, je však výpočetně nenáročná a, jak bude dále ukázáno, je možné ji efektivně zparalelizovat. Následně bude srovnána sériová a paralelní verze navrženého přístupu z pohledu časové náročnosti a kvality řešení.*

Abstract: *Space-Filling Design Strategies known as Design of (Computer) Experiments (DoE) create an essential part of a surrogate modeling. Two main objectives are usually placed on the resulting designs - orthogonality and space-filling properties. The last decade has witnessed the development of several methods for the latter objective. One of the metrics, called miniMax (mM) represents very interesting research area. Authors think that this metric seems as the most suitable one for practical purposes; however its computation for higher dimensions is intractable. Hence our contribution presents a new method on this subject. The core is the usage of the Evolution Strategy (ES) algorithm to estimate potential centers of the largest empty spheres. The miniMax is then the radius of the biggest empty sphere. Therefore the proposed procedure is inexact, however has no memory problem and as is shown in the text, is suitable for parallelization. The serial and parallel versions of the proposed approach are then compared from time and precision points of view.*

Obsah

1	Úvod	3
2	Přesné řešení	3
3	Sériová evoluční strategie	4
4	Paralelní evoluční strategie	6
5	Výsledky	8
6	Závěr	9
7	Poděkování	10

1 Úvod

Návrh experimentů - *the design of experiments (DoE)* tvoří základní součást vývoje každého meta-modelu [Simpson et al., 2001, Jin, 2005]. Cílem je získat co nejvíce informací o chování daného systému při minimálním počtu provedených experimentů. Jelikož předpokládáme, že finální meta-model je *a priori* neznámý, je žádoucí, aby návrh experimentů co nejlépe pokrýval celý návrhový prostor. Kvalitu návrhu můžeme hodnotit řadou kritérií zaměřených především na ortogonalitu [Cioppa and Lucas, 2007, Hofwing and Strömberg, 2010] nebo rovnoměrnost pokrytí [Crombecq et al., 2009, Myšáková and Lepš, 2011, Janouchová and Kučerová, 2013]. V této práci jsme zvolili kritérium *miniMax (mM)* pro jeho jednoduchost a snadnost zobrazení. Je-li dána sada n (návrhových) bodů v d -dimenzionální hyperkrychli, hodnota *miniMax* je poloměr největší hyperkoule, která neobsahuje žádný z n bodů sady a jejíž střed leží uvnitř dané hyperkrychle. Tento problém je v literatuře též označován jako *the largest empty sphere problem (LES)* [Dickerson and Eppstein, 1995] - „problém největší prázdné koule“. Jinými slovy, *miniMax* slouží k hodnocení kvality pokrytí návrhového prostoru tím, že ukazuje největší nepokrytou oblast. Jednou z možností pro nalezení středu největší prázdné koule (a poté hodnoty *mM*) je prověření všech vrcholů Voronoiova diagramu [Okabe et al., 2000] sestaveného pro danou sadu návrhových bodů. Počet těchto vrcholů však roste $O(n^{\lceil d/2 \rceil})$ v případě, kdy prostor není ohraničen, a v ohraničeném prostoru (což je případ sady bodů v hyperkrychli) je počet bodů nutných k prověření ještě vyšší. Ačkoli můžeme problém hraničních oblastí domény efektivně řešit pomocí zrcadlení [Pronzato and Müller, 2012], ke spolehlivému sestavení Voronoiova diagramu v reálném čase ve vyšších dimenzích jsou stále třeba nové algoritmy.

Tato práce je inspirována článkem [Lee et al., 2004], kde byla pro nalezení středu největší nepokryté koule použita evoluční strategie. Ukážeme, že tato metoda není schopná nalézt přesné řešení v prostoru nad 6D. Proto představíme vylepšený algoritmus, v němž je evoluční strategie spuštěna paralelně na jednotlivých částech řešené hyperkrychle. Výsledky ukazují, že paralelní metoda je robustnější než sériová verze popsaná ve zmíněném článku. Pro použití ve vysokých dimenzích (stovky až tisíce proměnných) ji však ještě bude nutné zdokonalit.

Členění práce je následující. Nejprve jsou popsány dvě exaktní metody pro výpočet hodnoty kritéria *miniMax*. Následně je představen algoritmus evoluční strategie v sériové verzi a popis jeho zparalelizování. Metody jsou dále porovnány z pohledu výpočetního času a kvality řešení.

2 Přesné řešení

Přesnou hodnotu kritéria *miniMax* lze získat pomocí Voronoiova diagramu. Střed největší prázdné koule (jejíž poloměr odpovídá hodnotě *mM*) leží v některém z vrcholů Voronoiova diagramu, nebo průniku hrany Voronoiova diagramu s hranicí řešené domény.

Ve 2D je všechny tyto kandidátní body možné najít „ručně“. Jsou to vrcholy Voronoiova diagramu, průsečíky hran Voronoiova diagramu se čtyřmi stranami řešené domény (ve 2D čtverce - $d = 2$) a čtyři vrcholy řešené domény [Schuster, 2008, Toussaint, 1983]. Zde tuto metodu označujeme jako *miniMax I* a její průběh je znázorněn na Obrázku 1. Ve vyšších dimenzích je však použití této metody obtížnější. Nalezení všech průníků Voronoiova diagramu s hraničními objekty řešené domény není triviální a je nutné vzít v úvahu i všechny vrcholy řešené hyperkrychle.

V článku [Pronzato and Müller, 2012] je popsána jiná metoda pro nalezení přesného řešení. Návrhové body jsou zrcadleny skrz všechny $(d - 1)$ -facety a až následně je sestrojen Voronoioův

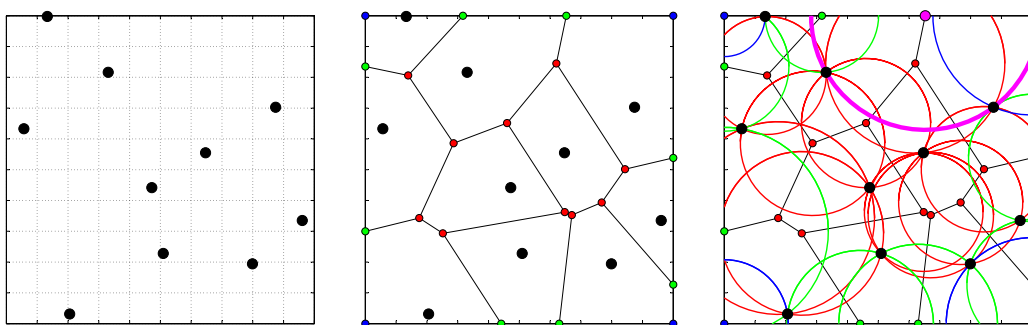
diagram. Kandidátní body jsou pak ty ležící uvnitř, nebo na hranici řešené domény. Tuto metodu ilustruje Obrázek 2 a označujeme ji jako miniMax II. Zrcadlení zaručuje, že i body na hranici a ve vrcholech domény budou uvedeny mezi kandidátními body. I tato metoda však má své limity. Sestrojení Voronoiova diagramu je ve vyšších dimenzích velice náročné (časově i paměťově) a zrcadlení (každý bod musí být ozrcadlen $(2d)$ -krát) tyto nároky ještě zvyšuje. To dokumentuje i Tabulka 1 v Sekci 5 - v nižších dimenzích jsou exaktní metody efektivnější, ve vyšších dimenzích ale jejich paměťové nároky výrazně rostou.

Problémy, se kterými se setkáváme v inženýrské praxi, jsou však zpravidla multidimenzionální (desítky, stovky, někdy i tisíce proměnných), proto potřebujeme metody schopné pracovat i v takových návrhových prostorech.

3 Sériová evoluční strategie

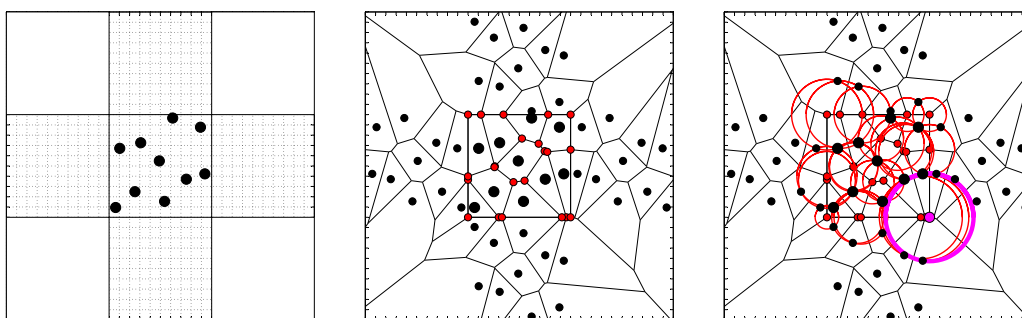
Jednou z možností je odhadnout hodnotu miniMax pomocí heuristické nebo meta-heuristické metody. My jsme zvolili evoluční strategii. Jedná se o metodu založenou na principech známých z přírody - přizpůsobování (*adaptation*), mutace (*matation*), křížení (*crossover*) a výběr (*selection*). Tato technika byla vyvinuta v 60. a 70. letech 20. století Rechenbergem, Schwefelem a jejich spolupracovníky, v [Rechenberg, 1973] nebo [Bäck and Schwefel, 1995] lze nalézt více informací .

Základem procedury je cyklus, jehož iterace nazýváme generace. Každá generace obsahuje populaci složenou z jednotlivých chromosomů - bodů, které pokrývají řešenou doménu. Cílová funkce je pak vyhodnocena pro každý chromosom (s parametry odpovídajícími souřadnicím chromosomu). Nová populace („potomci“) je odvozena z předchozí populace („rodiče“) pomocí mutace a křížení. Rozlišují se dva typy: v (μ, λ) -ES je nová generace odvozena z potomků z předchozí generace, v $(\mu + \lambda)$ -ES ze sjednocení rodičů a potomků z předchozí generace. Počet generací může být stanoven uživatelem pevně, nebo je pro algoritmus určeno zastavovací kritérium.

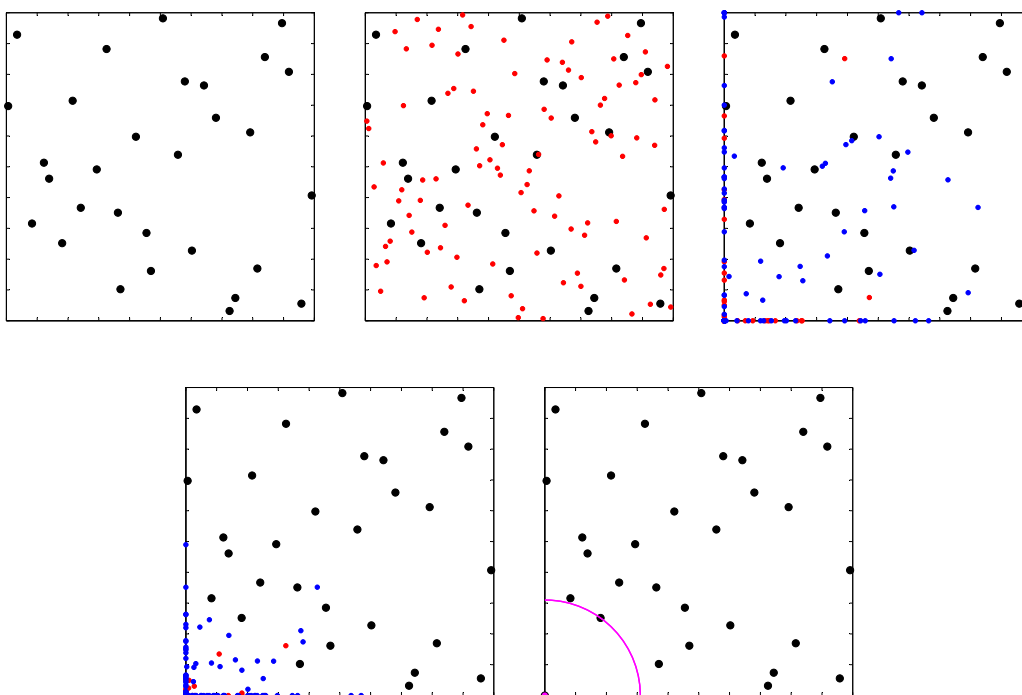


Obrázek 1: miniMax I. Legenda: černé body - návrhové body; černé úsečky - Voronoiova diagram; červené body - vrcholy Voronoiova diagramu; zelené body - průsečky hran Voronoiova diagramu s hranicemi řešené domény; modré body - vrcholy domény; červené, zelené a modré kružnice - největší prázdné kruhy se středy v červených, zelených a modrých bodech; purpurová kružnice - největší prázdný kruh; purpurový bod - střed největšího prázdného kruhu. Obrázky slouží pro ilustraci průběhu metody, popisky os jsou proto vynechány.

3. SÉRIOVÁ EVOLUČNÍ STRATEGIE

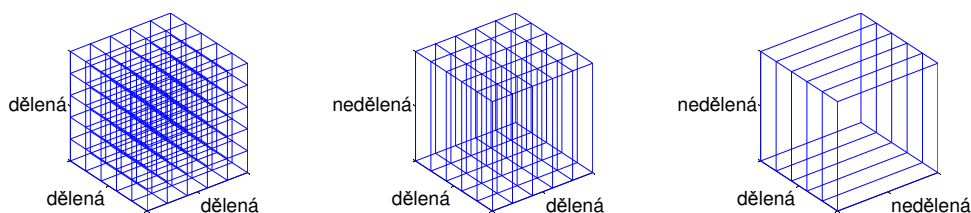


Obrázek 2: miniMax II. Legenda: velké černé body - návrhové body; malé černé body - ozrcadlené návrhové body; černé úsečky - Voronoiův diagram; červené body - vrcholy Voronoiova diagramu ležící uvnitř nebo na hranici řešené domény; červené kružnice - největší prázdné kruhy se středy v červených bodech; purpurová kružnice - největší prázdný kruh; purpurový bod - střed největšího prázdného kruhu.

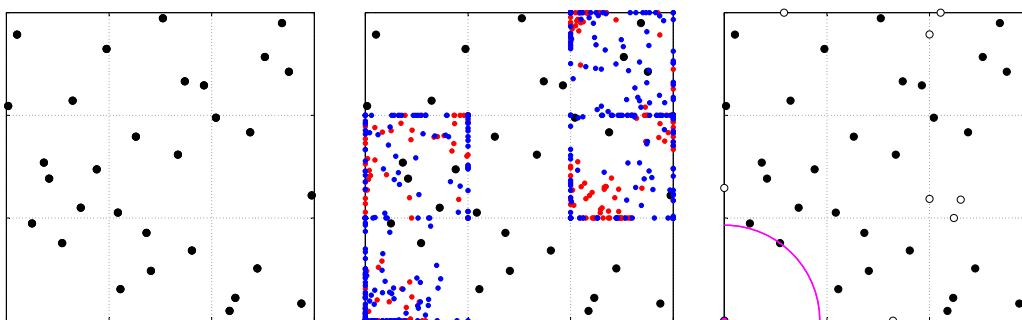


Obrázek 3: Sériová evoluční strategie. Legenda: černé body - návrhové body; červené body - „rodiče“; modré body - „potomci“; purpurový bod - střed největšího prázdného kruhu; purpurová kružnice - největší prázdný kruh.

$(\mu + \lambda)$ -evoluční strategie představená v této práci je inspirována článkem [Lee et al., 2004]. Nejprve je vytvořena výchozí populace. Ta by měla rovnoměrně pokrývat celý prostor, proto je vytvořena pomocí LHS (*latin hypercube sampling*). Jelikož se však v této práci zaměřujeme na porovnání časové náročnosti vlastního cyklu evoluční strategie (sériové vs. paralelní), je



Obrázek 4: Paralelní evoluční strategie - způsoby dělení na subdomény ve 3D s $k = 5$, kde k je počet intervalů na dělené hraně domény.



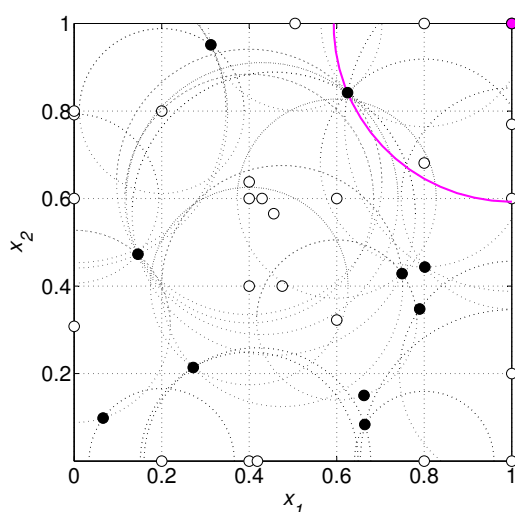
Obrázek 5: Paralelní evoluční strategie ve 2D s $k = 3$. Legenda: černé body - návrhové body; červené body - „rodiče“; modré body - „potomci“; bílé body - v každé subdoméně bod nejvíce vzdálený od návrhových bodů; purpurový bod - střed největšího prázdného kruhu; purpurová kružnice - největší prázdný kruh.

počáteční populace vytvořena pomocí funkce `lhsdesign` ze statistického toolboxu programu Matlab ve výchozím nastavení tak, aby nebyl celkový čas tímto procesem výrazně ovlivněn. Potomci jsou odvozováni pomocí mutace - přičtením normálně rozdělených (průměr roven nule, standardní odchylka klesající s pořadím cyklu) náhodných čísel k rodičovské populaci. Následně jsou ze sjednocení rodičovské populace a potomků vytvořeny náhodné páry a z každého tohoto páru je do nové rodičovské populace vybrán ten chromosom, který má delší vzdálenost k nejbližšímu návrhovému bodu. Toto výběrové schéma upřednostňuje lepší jednotlivce k postupu do další generace. Metoda je znázorněna na Obrázku 3. Je na něm patrné, jak se populace přibližuje k hledanému řešení.

Metoda je robustní a uživatel může volit velikost populace i počet generací podle dimenze řešeného problému. Přesto je ve vyšších dimenzích obtížné prozkoumat celou řešenou doménu. To by měla zajistit dále popsaná paralelní metoda.

4 Paralelní evoluční strategie

Paralelizovat evoluční strategii lze dvěma způsoby: za prvé, může probíhat několikeré hledání v celé doméně paralelně nezávisle na sobě; za druhé, řešenou doménu můžeme rozdělit na menší celky - subdomény - a hledání pomocí ES pak probíhá paralelně v těchto subdoménách. V této práci je zvolena druhá z možností.



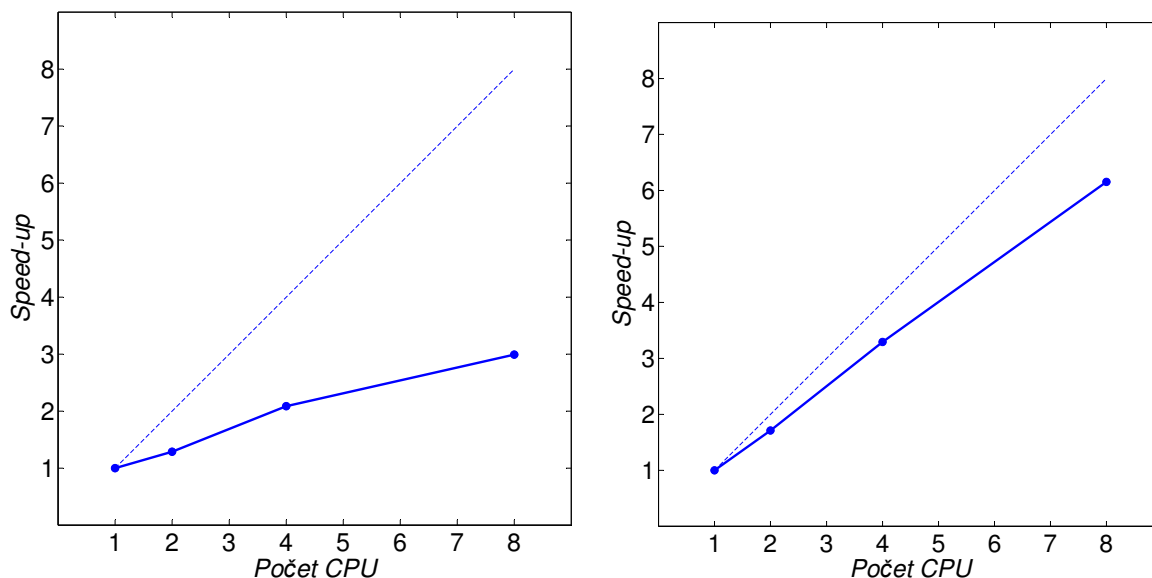
Obrázek 6: Paralelní evoluční strategie. Legenda: černé body - návrhové body; bílé body - v každé subdoméně bod nejvíce vzdálený od návrhových bodů; tečkované kružnice - největší prázdné kruhy se středy v bílých bodech; purpurový bod - střed největšího prázdného kruhu; purpurová kružnice - největší prázdný kruh.

Doménu lze dělit několika způsoby, jak ukazuje Obrázek 4. První z uvedených variant je pravděpodobně nejvhodnější k zaručení prohledání celého prostoru. Počet vzniklých subdomén je potom k^d , kde k označuje počet intervalů na dělené hraně domény. Množství subdomén tedy roste velice rychle s dimenzí řešené domény a v případech vyšších domén je tedy nutné použít některou z dalších alternativ dělení. V příkladech uvedených v této práci je použita první varianta dělení na subdomény s $k = 2$. Počet chromosomů v populaci je $10d$.

Popsanou metodu znázorňuje Obrázek 5. Několik subdomén je tedy řešeno zároveň paralelně na jednotlivých CPU. Z každé subdomény obdržíme kandidátní řešení/bod (Obrázek 6), jako střed největší prázdné koule (její poloměr je pak hodnota mM) je z těchto bodů označen ten, který má nejdelší vzdálenost k bodům návrhu.

	2D	6D	12D
metoda	čas [s]		
miniMax I	0.007	–	–
miniMax II	0.004	2.265	- (vyčerpáno 12 GB RAM)
ES miniMax - sériová	0.183	47.193	2489.557
ES miniMax - paralelní - 2 CPU	7.338	36.634	1455.274
ES miniMax - paralelní - 4 CPU	7.489	22.617	756.122
ES miniMax - paralelní - 8 CPU	7.935	15.774	404.693

Tabulka 1: Časová náročnost uvedených metod.



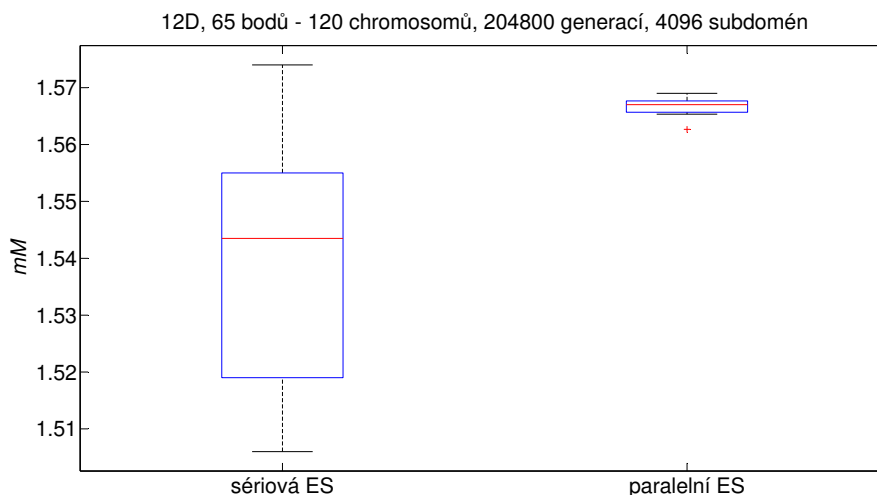
Obrázek 7: Speed-up paralelní ES pro výpočet hodnoty miniMax: 6D, 17 bodů - 60 chromosomů, 12800 generací, 64 subdomén (vlevo) a 12D, 65 bodů - 120 chromosomů, 204800 generací, 4096 subdomén (vpravo).

5 Výsledky

K porovnání představených metod jsou použity tři typové příklady: (i) 2D návrh s 27 body převzatý z [van Dam, 2005], (ii) 6D návrh se 17 body a (iii) 12D návrh s 65 body. Postup optimalizace tvorby uvedených 6D a 12D návrhů je popsán v [Cioppa and Lucas, 2007] a výsledné návrhy byly staženy ze [Sanches, 2005].

Nejprve porovnáme jednotlivé metody z pohledu výpočetních nároků. Ty jsou uvedeny v Tabulce 1. Jedná se o průměrné hodnoty z 10 nezávislých spuštění každé procedury. Je zřejmé, že v nízkých dimenzích jasně vítězí metody poskytující přesné řešení (označeny miniMax I a miniMax II). Již v 12D příkladu ovšem nestačí operační paměť kvůli extrémní náročnosti sestavení Voronoiova diagramu. Sériová verze evoluční strategie je při zvolené velikosti populace a počtu generací více než 20krát pomalejší než exaktní metody. Efektivita paralelní verze je pak ve 2D případě nulová. Komunikace v počítači nutná k rozdělení problému na jednotlivé subdomény zde časově vysoce převyšuje samotný výpočet a celkový potřebný čas tu dokonce roste s počtem použitých CPU. Zajímavější je však situace ve vyšších dimenzích. Obrázek 7 ukazuje speed-up (závislost zrychlení výpočtu na počtu využitých CPU) pro 6D a 12D problémy. Se zvyšující se dimenzí se čas potřebný ke komunikaci mezi jednotlivými CPU stává zanedbatelným a paralelní metoda zde dosahuje téměř lineárního speed-upu.

Nyní metody porovnáme z pohledu kvality získaného řešení - hodnoty kritéria miniMax. Připomeňme, že evoluční strategie je stochastický algoritmus s náhodným chováním a hodnota jí získána je tedy pouze odhadem - přesná hodnota může být vyšší. V nižších dimenzích, tedy ve 2D a 6D příkladu poskytují naše metody založené na ES (sériová a paralelní) přesné řešení při všech 10 spuštěních. Pro 12D problém přesnou hodnotu mM neznáme. Obrázek 8 ukazuje boxploty hodnot kritéria miniMax získaných pomocí obou verzí evoluční strategie. Sériová verze je někdy schopná nalézt přesnější řešení (vyšší hodnotu odhadu mM), vykazuje však vyšší rozptyl v porovnání s verzí paralelní řešící problém v jednotlivých subdoménách. Poznamenejme, že



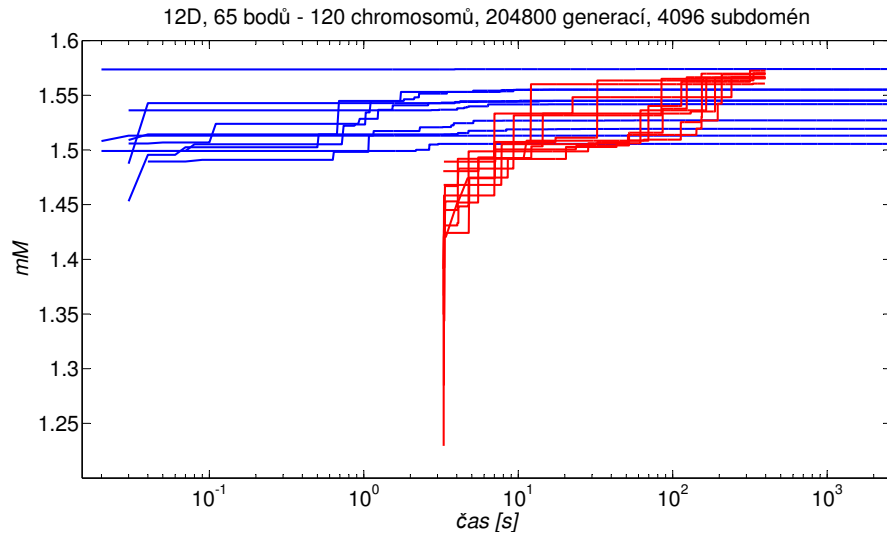
Obrázek 8: Boxploty hodnot mM získaných uvedenými metodami pro problém ve 12D.

celkový počet provedených iterací a výpočtů vzdáleností byl shodný pro sériovou verzi i pro verzi paralelní (součet počtů generací v jednotlivých subdoménách je shodný s počtem generací v sériové verzi). Srovnání metod z hlediska reálného času ukazuje Obrázek 9. Sériová verze ES se zdá být velmi rychlá z pohledu nalezení kvalitního řešení. Výsledek je však ze značné části dán „štěstím“ v zasažení oblasti středu největší prázdné koule (ten se velmi často nachází na hranici řešené domény). Paralelní verze zase ztrácí určitý čas na začátku výpočtu rozdělením prostoru na subdomény a komunikací nutnou k paralelizaci. Poté však paralelní verze poskytuje robustnější řešení s menším rozptylem.

6 Závěr

Tato práce představuje dvě tradiční metody poskytující přesné řešení miniMax kritéria společně s algoritmem evoluční strategie v sériové a paralelní verzi. Ten poskytuje odhad hodnoty kritéria. Naše výsledky ukazují, že v nižších dimenzích navržené metody založené na ES přesné řešení naleznou. Ve vyšších dimenzích pak bez problémů s nedostatkem operační paměti (typické pro exaktní metody využívající Voronoiův diagram) poskytují odhad hodnoty mM . Paralelní verze představuje způsob „rozparcelování“, tedy rozdělení celé řešené domény na menší subdomény. Tato verze dosahuje ve vyšších dimenzích téměř lineárního speed-upu a odhady jí získané mají menší rozptyl v porovnání se sériovou verzí (kde je řešena celá doména najednou).

V této práci nebyly zkoumány všechny možnosti hledání přibližného a/nebo paralelního řešení problému hodnoty miniMax. Mezi další oblasti vhodné pro výzkum patří: (i) aproximace Voronoiova diagramu popsána např. v pracích [Arya et al., 2002] nebo [Brochhaus et al., 2006]; a (ii) využití GPU paralelního počítání, více v publikaci [Wang et al., 2002] nebo v článku [Fischer and Gotsman, 2006]. Náš další výzkum bude zaměřen na adaptivní testování se zaměřením na robustní metody optimalizace [Dubourg, 2011]. Kritérium miniMax se pro tyto účely zdá být nejvhodnějším. Střed největší nepokryté koule je přirozeným kandidátem pro testování v dalším kroku. Stejná myšlenka je již nyní používána v tzv. updatovaných modelech jako je např. Kriging, pro více informací [Jones, 2001].



Obrázek 9: Vývoj hodnoty mM v průběhu výpočtu pro 10 spuštění ES. Legenda: modré křivky - sériová verze; červené křivky - paralelní verze (8 CPU).

7 Poděkování

Autoři by rádi poděkovali za finanční podporu Grantové agentury České republiky GAČR v rámci grantu č. P105/12/1146.

Literatura

- [Arya et al., 2002] Arya, S., Malamatos, T., and Mount, D. M. (2002). Space-efficient approximate Voronoi diagrams. In *Proceedings of the thirty-fourth annual ACM symposium on Theory of computing*, STOC '02, pages 721–730, New York, NY, USA. ACM.
- [Bäck and Schwefel, 1995] Bäck, T. and Schwefel, H.-P. (1995). Evolution Strategies I: Variants and their computational implementation. In Periaux and Winter, editors, *Genetic Algorithms in Engineering and Computer Science*, chapter 6, pages 111–126. John Wiley & Sons Ltd.
- [Brochhaus et al., 2006] Brochhaus, C., Wichterich, M., and Seidl, T. (2006). Approximation techniques to enable dimensionality reduction for Voronoi-based nearest neighbor search. In *In Ioannidis Y. et al. (Eds.): Advances in Database Technology - Proc. 10th International Conference on Extending Data Base Technology (EDBT 2006), Munich, Germany. Springer LNCS 3896*, pages 204–221, Heidelberg, Germany. Springer.
- [Cioppa and Lucas, 2007] Cioppa, T. M. and Lucas, T. (2007). Efficient nearly orthogonal and space-filling latin hypercubes. *Technometrics*, 49(1):45–55.
- [Crombecq et al., 2009] Crombecq, K., Couckuyt, I., Gorissen, D., and Dhaene, T. (2009). Space-filling sequential design strategies for adaptive surrogate modelling. In Topping, B. H. V. and Tsompanakis, Y., editors, *Proceedings of the First International Conference on Soft Computing Technology in Civil, Structural and Environmental Engineering*. Civil-Comp Press, Stirlingshire, UK.
- [Dickerson and Eppstein, 1995] Dickerson, M. and Eppstein, D. (1995). Algorithms for proximity problems in higher dimensions. *Comput. Geom.*, 5:277–291.
- [Dubourg, 2011] Dubourg, V. (2011). *Adaptive surrogate models for reliability analysis and reliability-based design optimization*. PhD thesis, Université Blaise Pascal, Clermont-Ferrand, France.
- [Fischer and Gotsman, 2006] Fischer, I. and Gotsman, C. (2006). Fast approximation of high-order Voronoi diagrams and distance transforms on the GPU. *J. Graphics Tools*, pages 39–60.
- [Hofwing and Strömberg, 2010] Hofwing, M. and Strömberg, N. (2010). D-optimality of non-regular design spaces by using a bayesian modification and a hybrid method. *Structural and multidisciplinary optimization (Print)*, 42(1):73–88.

- [Janouchová and Kučerová, 2013] Janouchová, E. and Kučerová, A. (2013). Competitive comparison of optimal designs of experiments for sampling-based sensitivity analysis. *Computers & Structures (Accepted for publication)*.
- [Jin, 2005] Jin, Y. (2005). A comprehensive survey of fitness approximation in evolutionary computation. *Soft Computing*, 9:3–12.
- [Jones, 2001] Jones, D. R. (2001). A taxonomy of global optimization methods based on response surfaces. *Journal of Global Optimization*, 21:345–383.
- [Lee et al., 2004] Lee, J.-S., Cho, T.-S., Lee, J., Jang, M.-K., Jang, T.-K., Nam, D., and Park, C. H. (2004). A stochastic search approach for the multidimensional largest empty sphere problem.
- [Myšáková and Lepš, 2011] Myšáková, E. and Lepš, M. (2011). Comparison of Space-Filling Design Strategies. In *Engineering Mechanics 2011*, pages 399–402, Praha. Ústav termomechaniky AV ČR.
- [Okabe et al., 2000] Okabe, A., Boots, B., Sugihara, K., and Chiu, S. N. (2000). *Spatial tessellations: concepts and applications of Voronoi diagrams*. Wiley series in probability and statistics: Applied probability and statistics. Wiley.
- [Pronzato and Müller, 2012] Pronzato, L. and Müller, W. G. (2012). Design of computer experiments: space filling and beyond. *Statistics and Computing*, 22(3):681–701.
- [Rechenberg, 1973] Rechenberg, I. (1973). *Evolution strategy: Optimization of technical systems by means of biological evolution*. Fromman-Holzboog, Stuttgart.
- [Sanches, 2005] Sanches, S. M. (2005). Nohdesigns spreadsheet. Available online via http://diana.cs.nps.navy.mil/SeedLab/LinkedFiles/NOLHdesigns_v4.xls.
- [Schuster, 2008] Schuster, M. (2008). The largest empty circle problem. In *Proceedings of the Class of 2008 Senior Conference, Computer Science Department, Swarthmore College*, pages 28–37.
- [Simpson et al., 2001] Simpson, T. W., Peplinski, J. D., Koch, P. N., and Allen, J. K. (2001). Metamodels for computer-based engineering design: Survey and recommendations. *Engineering with Computers*, 17:129–150.
- [Toussaint, 1983] Toussaint, G. (1983). Computing largest empty circles with location constraints. *International Journal of Computer & Information Sciences*, 12:347–358.
- [van Dam, 2005] van Dam, E. R. (2005). Two-dimensional minimax latin hypercube designs. Discussion Paper 2005-105, Tilburg University, Center for Economic Research.
- [Wang et al., 2002] Wang, Y.-R., Horng, S.-J., Lee, Y.-H., and Lee, P.-Z. (2002). Optimal parallel algorithms for the 3D Euclidean distance transform on the CRCW and EREW PRAM models. In *Proceedings of the 19th Workshop on Combinatorial Mathematics and Computation Theory, Kaohsiung, Taiwan*, pages 209–218.