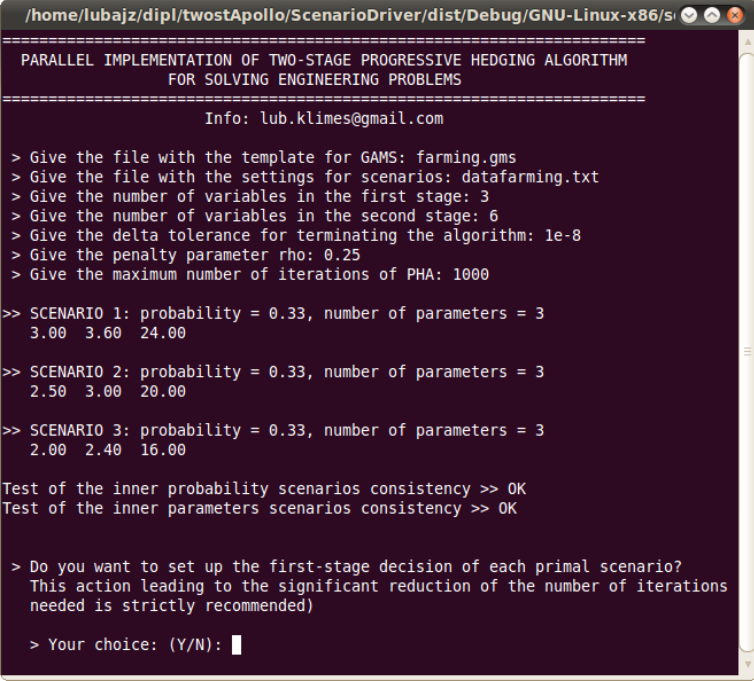


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA STROJNÍHO INŽENÝRSTVÍ

ENERGETICKÝ ÚSTAV
ODBOR TERMOMECHANIKY A TECHNIKY PROSTŘEDÍ

PARALLEL IMPLEMENTATION OF TWO-STAGE
PROGRESSIVE HEDGING ALGORITHM
FOR SOLVING ENGINEERING PROBLEMS



```
/home/lubajz/dipl/twostApollo/ScenarioDriver/dist/Debug/GNU-Linux-x86/s
=====
PARALLEL IMPLEMENTATION OF TWO-STAGE PROGRESSIVE HEDGING ALGORITHM
FOR SOLVING ENGINEERING PROBLEMS
=====
Info: lub.klimes@gmail.com

> Give the file with the template for GAMS: farming.gms
> Give the file with the settings for scenarios: datafarming.txt
> Give the number of variables in the first stage: 3
> Give the number of variables in the second stage: 6
> Give the delta tolerance for terminating the algorithm: 1e-8
> Give the penalty parameter rho: 0.25
> Give the maximum number of iterations of PHA: 1000

>> SCENARIO 1: probability = 0.33, number of parameters = 3
3.00 3.60 24.00

>> SCENARIO 2: probability = 0.33, number of parameters = 3
2.50 3.00 20.00

>> SCENARIO 3: probability = 0.33, number of parameters = 3
2.00 2.40 16.00

Test of the inner probability scenarios consistency >> OK
Test of the inner parameters scenarios consistency >> OK

> Do you want to set up the first-stage decision of each primal scenario?
This action leading to the significant reduction of the number of iterations
needed is strictly recommended)

> Your choice: (Y/N): █
```

Apollo ID: 24630
Datum: 15. října 2010
Typ projektu: R — software
Autor: Ing. Lubomír Klimeš

Popis funkce

Software *Parallel Implementation of Two-Stage Progressive Hedging Algorithm for Solving Engineering Problems* je určen pro řešení dvojestupňových stochastických optimalizačních úloh, které se často vyskytují při řešení inženýrských problémů.

Program je napsán v C++ a je primárně určen pro operační systém Linux. Pro řešení scénářových podúloh je používán optimalizační software GAMS¹, který musí být nainstalován včetně nelineárního řešiče (např. CONOPT) a jeho systémové umístění musí být uloženo v souboru `pathGAMS.txt`. Více o GAMSu v [3]. Dále je vyžadováno nainstalované Message Passing Interface, MPI, zahrnující program `mpiexec`. Více o MPI v [4]. Software byl testován s implementací OpenMPI².

Popis algoritmu

Algoritmus progressive hedging patří mezi dekompoziční algoritmy pro řešení stochastických optimalizačních úloh, jehož výhodou je rozklad původní úlohy na vzájemně nezávislé podúlohy, které jsou řešeny odděleně a umožňující paralelizaci. Algoritmus progressive hedging je podrobně popsán v [8, 10]. O stochastickém programování je pojednáno např. v [2, 5], o nelineární optimalizaci v [1]. Paralelní implementace použitá v tomto software je popsána v [7, 6].

Technické a programové požadavky

Operační systém Linux (testováno na Ubuntu Jaunty Jackalope a Lucid Lynx), optimalizační software GAMS včetně řešiče nelineárních úloh (např. CONOPT), Message Passing Interface (testováno na OpenMPI).

Popis použití

Program se spustí příkazem `./parpha` z Terminálu. Vstupní parametry pro program jsou následující:

1. ŠABLONA ÚLOHY PRO GAMS. Textový soubor se šablonou úlohy obsahující speciální značky, které program nahrazuje v dané iteraci konkrétními hodnotami. Speciální značka je uvozena znakem “zavináč” `@` a následuje přirozené číslo, např. `@102`. Čísla 0, 1, 2, ..., 99 jsou použita pro hodnoty náhodných veličin jednotlivých scénářů dle hodnot v souboru s nastavením scénářů. Čísla 100, 101, ..., 199 jsou použita pro váhové proměnné, viz [8, 10, 6]. Čísla 200, 201, ..., 299 jsou vyhrazena pro “průměrná” řešení, opět viz [8, 10, 6]. Číslo 300 je vyhrazeno penalizačnímu parametru. Šablona musí po příkazu `solve` obsahovat následující tři řádky:

```
file done / done@400 /;
put done;
put 'finish';
```

2. SOUBOR S NASTAVENÍM SCÉNÁŘŮ. Textový soubor s nastavením hodnot náhodných veličin úlohy pro jednotlivé scénáře. Soubor musí obsahovat právě tolik řádků³, kolik má

¹<http://www.gams.com>

²OpenMPI (<http://www.open-mpi.org>) v Ubuntu nainstalujete z repozitářů:

```
sudo apt-get install libopenmpi-dev openmpi-bin openmpi-doc
sudo apt-get install ssh
```

a SSH nakonfigurujete:

```
ssh-keygen -t dsa
cd ~/.ssh
cat id_dsa.pub >> authorized_keys
```

³kromě řádků s komentáři, které začínají znakem `%`

úloha scénářů a každý řádek představuje nastavení konkrétního scénáře. V prvním sloupci je vždy uvedena pravděpodobnost realizace daného scénáře, další sloupce představují jednotlivé hodnoty náhodných parametrů úlohy. Je-li tedy počet náhodných parametrů úlohy n , pak šablona musí obsahovat speciální značky s hodnotami $0, 1, \dots, n - 1$ a soubor nastavení scénářů musí mít $n + 1$ sloupců (včetně prvního s pravděpodobností scénáře).

3. POČET PROMĚNNÝCH V PRVNÍM STUPNI ÚLOHY. Dle konkrétní řešené úlohy.
4. POČET PROMĚNNÝCH V DRUHÉM STUPNI ÚLOHY. Dle konkrétní řešené úlohy.
5. TOLERANCE PRO ZASTAVENÍ ALGORITMU. Dle volby uživatele, představuje nejmenší změnu řešení během dvou po sobě následujících iterací, při které dojde k zastavení algoritmu. Více o výpočtu změny řešení viz [6].
6. PENALIZAČNÍ PARAMETR. Dle konkrétní úlohy a volby uživatele. Více viz [8, 10, 6, 9].
7. MAXIMÁLNÍ POČET ITERACÍ ALGORITMU. Dle konkrétní řešené úlohy a volby uživatele.

Po zadání všech vstupních parametrů program nabízí možnost vložení řešení prvního stupně jednotlivých původních scénářů. Pokud jsou tato řešení známa, je doporučeno je do programu zadat, čímž dojde k počáteční inicializaci “průměrného” řešení úlohy a zejména pak ke snížení počtu iterací algoritmu potřebného k dosažení řešení se zadanou přesností.

Pokud algoritmus dosáhne optimálního řešení úlohy, je toto řešení zobrazeno v okně Terminálu s běžícím programem. Dále jsou dílčí výsledky algoritmu ve všech iteracích zaznamenány do souboru `results.txt`.

Pro zastavení programu během výpočtu stiskněte `Ctrl-C`.

Archiv s programem obsahuje vzorové vstupy pro řešení “farmářovy úlohy”, viz [2, 6]: šablona `farming.gms`, soubor s nastavením scénářů `datafarming.txt`, počet proměnných v prvním stupni úlohy = 3, počet proměnných v druhém stupni úlohy = 6. S tolerancí pro zastavení algoritmu $1e-8$, penalizačním parametrem 0.25 a vložení řešení prvního stupně původních scénářů $\mathbf{x}_1 = (183.33, 66.67, 250)^T$, $\mathbf{x}_2 = (120, 80, 300)^T$ a $\mathbf{x}_3 = (100, 25, 375)^T$ algoritmus vyžaduje pro nalezení optimálního řešení 130 iterací. Bez vložení řešení \mathbf{x}_1 , \mathbf{x}_2 a \mathbf{x}_3 do programu vyžaduje algoritmus 145 iterací.

Vazba na projekty

1. FSI-J-10-8 *Matematické modelování a optimalizace v průmyslových aplikacích*
2. GA106/09/0940 *Numerický a stochastický model plynule odlévaných ocelových předlitků obdélníkového profilu*
3. 1M06047 *Centrum pro jakost a spolehlivost výroby*
4. GA103/08/1658 *Pokročilá optimalizace návrhu složených betonových konstrukcí*

Licenční podmínky

K využití software jiným subjektem je vždy nutné nabytí licence. Poskytovatel licence na software nepožaduje licenční poplatek.

Kontaktní osoba

Ing. Lubomír Klimeš, lub.klimes@gmail.com

Dokumentace grafického uživatelského prostředí

```
Terminal
Terminal
Terminal
Soubor Upravit Zobrazit Terminál Nápověda
Reading Data
Iter Phase Ninf Infeasibility RGmax NSB Step InItr MX OK
0 0 1.3459068983E+04 (Input point)
Pre-triangular equations: 0
Post-triangular equations: 1
1 0 4.4000000000E+02 (After pre-processing)
2 0 3.4375000000E+00 (After scaling)
** Feasible solution. Value of objective = 128598.576214
Iter Phase Ninf Objective RGmax NSB Step InItr MX OK
5 3 -4.7549631017E+04 0.0E+00 0
** Optimal solution. There are no superbasic variables.
--- Restarting execution
--- scenario3.gms(69) 0 Mb
--- Reading solution for model uloha
--- scenario3.gms(73) 3 Mb
--- Putfile done /home/lubajz/dipl/twostApollo/ScenarioDriver/done3
*** Status: Normal completion
--- Job scenario3.gms Stop 10/14/10 14:33:36 elapsed 0:00:00.424

e:/lubajz/dipl/twostApollo/ScenarioDriver/dist/Debug/GNU-Linux-x86/s
0.00000
30.00000
0.00000
stage decision for scenario 2:
30.00000
0.00000
0.00000
0.00000
30.00000
0.00000
0.00000
Second-stage decision for scenario 3:
0.00000
0.00000
0.00000
0.00000
4800.00000
0.00000
Distance parameter delta in 5-th iteration is 3099.595093730.
=====
Iteration 6
=====
> GAMS source file 'scenario1.gms' with scenario 1 has been created.
> GAMS source file 'scenario2.gms' with scenario 2 has been created.
> GAMS source file 'scenario3.gms' with scenario 3 has been created.
```

Stažení software

Software je možné stáhnout na <http://ottp.fme.vutbr.cz/vysledkyvyzkumu/parpha.zip>

Reference

- [1] M. S. Bazaraa, H. D. Sherali, and C. M. Sheety. *Nonlinear Programming: Theory and Algorithms*. John Wiley & Sons, New York, second edition, 1993. ISBN 0-471-55793-5.
- [2] J. R. Birge and F. Louveaux. *Introduction to Stochastic Programming*. Springer Series in Operations Research. Springer Verlag, New York, 1997. ISBN 0-387-98217-5.
- [3] A. Brooke, D. Kendrick, A. Meeraus, and R. Raman. *GAMS — A User's Guide*. GAMS Development Corporation, Washington, DC, USA, 2008.
- [4] W. Gropp, E. Lusk, and A. Skjellum. *Using MPI: Portable Parallel Programming with the Message Passing Interface*. MIT Press, second edition, 1999. ISBN 978-0262571326.

- [5] P. Kall and S. W. Wallace. *Stochastic Programming*. John Wiley & Sons, Inc., Chichester, second edition, 1994.
- [6] L. Klimeš. Stochastic programming algorithms. Master's thesis, Brno University of Technology, 2010.
- [7] L. Klimeš and P. Popela. An implementation of progressive hedging algorithm for engineering problems. In R. Matoušek, editor, *Mendel 2010*, 16th International Conference on Soft Computing, pages 459–464. Brno University of Technology, 2010.
- [8] R. T. Rockafellar and R. J.-B. Wets. Scenarios and policy aggregation in optimization under uncertainty. In *Mathematics of Operation Research*, volume 16, pages 119–147. INFORMS, Linthicum, Maryland, 1991.
- [9] J.-P. Watson, D. L. Woodruff, and D. R. Strip. Progressive hedging innovations for a class of stochastic resource allocation problems. Technical Report 2007-3722J 2007-3722J, Sandia National Laboratories, 2008.
- [10] R. J.-B. Wets. An aggregation principle in scenario analysis and stochastic optimization. In S. W. Wallace, editor, *Algorithms and Model Formulations in Mathematical Programming*. Springer Verlag, New York, 1989.

Prohlašuji, že popsaný výsledek naplňuje definici uvedenou v Příloze č. 1 *Metodiky hodnocení výsledků výzkumu a vývoje* v roce 2010 a že jsem si vědom důsledků plynoucích z porušení § 14 zákona č. 130/2002 Sb. (ve znění platném od 1. července 2009). Prohlašuji rovněž, že na požádání předložím technickou dokumentaci výsledku.

Ing. Lubomír Klimeš