Numerical Experiments with AMLET, a New Monte-Carlo Algorithm for Estimating Mixed Logit Models

Monte-Carlo methods with variable sampling size for mixed logit

Fabian Bastin, Cinzia Cirillo, Philippe L. Toint

Department of Mathematics

University of Namur

Namur, Belgium

Contact: `philippe.toint@fundp.ac.be`

10th International Conference on Travel Behaviour Research

The University of Namur

www.fundp.ac.be

Outline

PART I: *Theoretical investigations*

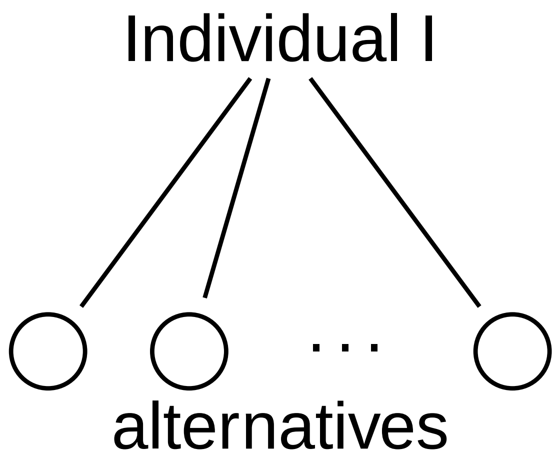
- Introduction: the mixed logit problem
- Properties
  - Convergence of solutions
  - Asymptotic properties: bias and error estimation
- Estimation algorithm
  - Trust region methods
  - Variable sampling size technique

Outline (2)

PART II: *Numerical investigations*

- The software `AMLET`
- Optimization framework: linesearch vs trust-region
- Tests on simulated data
- Comparison with Halton sequences
- Tests on real data (MobiDrive)
- Conclusions and research perspectives

Discrete choice models



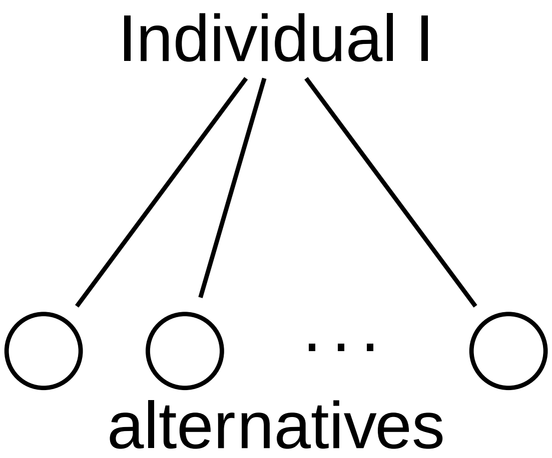
Set of alternatives available for individual i: $\mathcal{A}(i)$.

Utility U_{ij} of $A_j \in \mathcal{A}(i)$:

$$U_{ij} = V(\beta) + \boldsymbol{\epsilon}_{ij}.$$

β: parameters to be estimated.

Discrete choice models



Set of alternatives available for individual i: $\mathcal{A}(i)$.

Utility U_{ij} of $A_j \in \mathcal{A}(i)$:

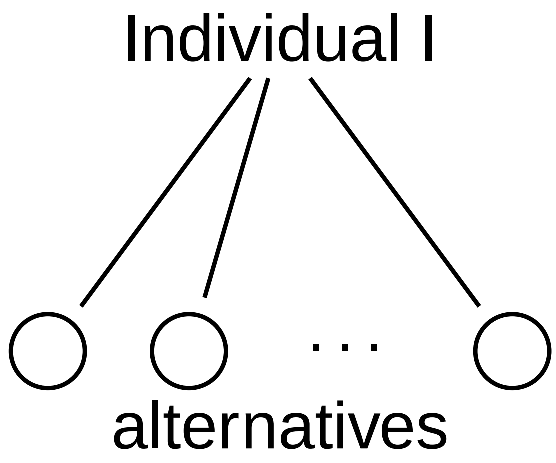
$$U_{ij} = V(\beta) + \boldsymbol{\epsilon}_{ij}.$$

β: parameters to be estimated.

Utility maximization principle: choice of A_j if $U_{ij} \geq U_{in}, \forall A_n \in \mathcal{A}(i)$.

Discrete choice models

Outline
→ ***Introduction***
- Properties
- Trust-region
- Variable size
- AMLET
- Simulated data
- Halton
- Real data
- Conclusions



Set of alternatives available for individual i: $\mathcal{A}(i)$.

Utility U_{ij} of $A_j \in \mathcal{A}(i)$:

$$U_{ij} = V(\beta) + \boldsymbol{\epsilon}_{ij}.$$

β: parameters to be estimated.

Utility maximization principle: choice of A_j if $U_{ij} \geq U_{in}, \forall A_n \in \mathcal{A}(i)$.

Gumbel distributed residuals ϵ_{ij} (mean 0, scale factor μ): multinomial logit (MNL).

Probability that individual i choose A_j:

$$P_{ij} = \frac{e^{\mu V_{ij}(\beta)}}{\sum_{n=1}^{N} e^{\mu V_{in}(\beta)}}.$$

The University of Namur
www.fundp.ac.be

Mixed Logit Models

Allow heterogeneity in parameters inside the population.

$$\boldsymbol{\beta} = \beta(\boldsymbol{\gamma}, \theta),$$

γ: random vector, e.g. vector of independant $N(0,1)$);

θ: vector of parameters, e.g. vector of means and std dev.

Mixed Logit Models

Allow heterogeneity in parameters inside the population.

$$\boldsymbol{\beta} = \beta(\boldsymbol{\gamma}, \theta),$$

γ: random vector, e.g. vector of independant $N(0,1)$);

θ: vector of parameters, e.g. vector of means and std dev.

Probability choice of A_j by individual i:

$$P_{ij}(\theta) = E_P\left[L_{ij}(\gamma, \theta)\right] = \int L_{ij}(\gamma, \theta) f(\gamma) d\gamma$$

Mixed Logit Models

Allow heterogeneity in parameters inside the population.

$$\boldsymbol{\beta} = \beta(\boldsymbol{\gamma}, \theta),$$

γ: random vector, e.g. vector of independant $N(0,1)$);

θ: vector of parameters, e.g. vector of means and std dev.

Probability choice of A_j by individual i:

$$P_{ij}(\theta) = E_P\left[L_{ij}(\gamma, \theta)\right] = \int L_{ij}(\gamma, \theta) f(\gamma) d\gamma$$

This integral is approximated by

$$SP_{ij}^R = \frac{1}{R} \sum_{r=1}^{R} L_{ij}(\gamma_r, \theta)$$

Mixed Logit Models

Allow heterogeneity in parameters inside the population.

$$\boldsymbol{\beta} = \beta(\boldsymbol{\gamma}, \theta),$$

γ: random vector, e.g. vector of independant $N(0,1)$);

θ: vector of parameters, e.g. vector of means and std dev.

Probability choice of A_j by individual i:

$$P_{ij}(\theta) = E_P\left[L_{ij}(\gamma, \theta)\right] = \int L_{ij}(\gamma, \theta) f(\gamma) d\gamma$$

This integral is approximated by

$$SP_{ij}^R = \frac{1}{R} \sum_{r=1}^R L_{ij}(\gamma_r, \theta)$$

Sample average approximation (SAA) problem:

$$\max_\theta \hat{g}_R(\theta) = \max_\theta SLL(\theta) = \max_\theta \frac{1}{I} \sum_{i=1}^I \ln SP_{ij_i}^R$$

Outline
- → *Introduction*
- Properties
- Trust-region
- Variable size
- AMLET
- Simulated data
- Halton
- Real data
- Conclusions

The University of Namur

www.fundp.ac.be

Numerical issues

Reduction of numerical cost:

1. work on the objective form: quasi Monte-Carlo;
2. adapt the optimization method: little exploration.

Numerical issues

Reduction of numerical cost:

1. work on the objective form: quasi Monte-Carlo;
2. adapt the optimization method: little exploration.

Two ideas to develop point 2:

Numerical issues

Reduction of numerical cost:

1. work on the objective form: quasi Monte-Carlo;
2. adapt the optimization method: little exploration.

Two ideas to develop point 2:

1. do not use full precision when far of the solution;

Numerical issues

Reduction of numerical cost:

1. work on the objective form: quasi Monte-Carlo;
2. adapt the optimization method: little exploration.

Two ideas to develop point 2:

1. do not use full precision when far of the solution;
2. stop optimization when no signification reduction can be obtained.

Numerical issues

Reduction of numerical cost:

1. work on the objective form: quasi Monte-Carlo;
2. adapt the optimization method: little exploration.

Two ideas to develop point 2:

1. do not use full precision when far of the solution;
2. stop optimization when no signification reduction can be obtained.

Requirement: ability to estimate the approximation's accuracy.

The University of Namur
www.fundp.ac.be

Properties of Mixed Logit

Analogy to stochastic programming:

$$\min_x g(x) = \min_x E\left[f(x, \boldsymbol{\omega})\right]$$

Properties of Mixed Logit

Analogy to stochastic programming:

$$\min_x g(x) = \min_x E\left[f(x, \boldsymbol{\omega})\right]$$

Mixed logit:

$$\max_\theta g\left(\theta\right) = \max_\theta LL(\theta) = \frac{1}{I}\sum_i \ln E_P\left[L_{ij_i}\left(\boldsymbol{\gamma}, \theta\right)\right]$$

Properties of Mixed Logit

Analogy to stochastic programming:

$$\min_x g(x) = \min_x E\left[f(x, \boldsymbol{\omega})\right]$$

Mixed logit:

$$\max_\theta g\left(\theta\right) = \max_\theta LL(\theta) = \frac{1}{I}\sum_i \ln E_P\left[L_{ij_i}\left(\boldsymbol{\gamma}, \theta\right)\right]$$

Known properties can be adapted.

Assume I fixed and R grows toward ∞.

Properties of Mixed Logit

Analogy to stochastic programming:

$$\min_x g(x) = \min_x E\left[f(x, \boldsymbol{\omega})\right]$$

Mixed logit:

$$\max_\theta g\left(\theta\right) = \max_\theta LL(\theta) = \frac{1}{I}\sum_i \ln E_P\left[L_{ij_i}\left(\boldsymbol{\gamma}, \theta\right)\right]$$

Known properties can be adapted.

Assume I fixed and R grows toward ∞.

If θ^*_R, $R = 1, \ldots$, is first-order critical for the corresponding SAA problem, any limit point θ^* of $(\theta^*_R)_{R=1}^{\infty}$ is first-order critical, almost surely.

Error and bias of simulation

With an i.i.d. sample for each individual, we have, from the delta method (see Shapiro and Rubinstein):

$$LL(\theta) - SLL^R(\theta) \Rightarrow N\left(0, \frac{1}{I}\sqrt{\sum_{i=1}^{I} \frac{\sigma^2_{ij_i}(\theta)}{R\big(P_{ij_i}(\theta)\big)^2}}\right)$$

Error and bias of simulation

With an i.i.d. sample for each individual, we have, from the delta method (see Shapiro and Rubinstein):

$$LL(\theta) - SLL^R(\theta) \Rightarrow N\left(0, \frac{1}{I}\sqrt{\sum_{i=1}^{I} \frac{\sigma^2_{ij_i}(\theta)}{R\big(P_{ij_i}(\theta)\big)^2}}\right)$$

Asymptotic value of the confidence interval radius:

$$\epsilon_\delta = \alpha_\delta \frac{1}{I}\sqrt{\sum_{i=1}^{I} \frac{\sigma^2_{ij_i}(\theta)}{R\big(P_{ij_i}(\theta)\big)^2}}$$

δ: signification level; $\alpha_{0.9} \approx 1.65$.

Error and bias of simulation

With an i.i.d. sample for each individual, we have, from the delta method (see Shapiro and Rubinstein):

$$LL(\theta) - SLL^R(\theta) \Rightarrow N\left(0, \frac{1}{I}\sqrt{\sum_{i=1}^{I} \frac{\sigma^2_{ij_i}(\theta)}{R\big(P_{ij_i}(\theta)\big)^2}}\right)$$

Asymptotic value of the confidence interval radius:

$$\epsilon_\delta = \alpha_\delta \frac{1}{I}\sqrt{\sum_{i=1}^{I} \frac{\sigma^2_{ij_i}(\theta)}{R\big(P_{ij_i}(\theta)\big)^2}}$$

δ: signification level; $\alpha_{0.9} \approx 1.65$.

Bias of simulation (Taylor expansion):

$$B := E[SLL^R(\theta)] - LL(\theta) = -\frac{I\epsilon^2_\delta}{2\alpha^2_\delta}$$

Error and bias of simulation

With an i.i.d. sample for each individual, we have, from the delta method (see Shapiro and Rubinstein):

$$LL(\theta) - SLL^R(\theta) \Rightarrow N\left(0, \frac{1}{I}\sqrt{\sum_{i=1}^{I} \frac{\sigma^2_{ij_i}(\theta)}{R\big(P_{ij_i}(\theta)\big)^2}}\right)$$

Asymptotic value of the confidence interval radius:

$$\epsilon_\delta = \alpha_\delta \frac{1}{I}\sqrt{\sum_{i=1}^{I} \frac{\sigma^2_{ij_i}(\theta)}{R\big(P_{ij_i}(\theta)\big)^2}}$$

δ: signification level; $\alpha_{0.9} \approx 1.65$.

Bias of simulation (Taylor expansion):

$$B := E[SLL^R(\theta)] - LL(\theta) = -\frac{I\epsilon^2_\delta}{2\alpha^2_\delta}$$

In practice, use of SAA estimators $\sigma^R_{ij_i}(\theta)$ and $P^R_{ij_i}(\theta)$.

The University of Namur
www.fundp.ac.be

Trust-region methods

Basic principle: at iteration k, $k = 1, \ldots$, approximately minimize a model of the objective over a trust region ($\mathcal{B}_k$). It gives a (candidate iterate).

Trust-region methods

Basic principle: at iteration k, $k = 1, \ldots$, approximately minimize a model of the objective over a trust region ($\mathcal{B}_k$).

It gives a (candidate iterate).

Compute the following ratio:

$$\rho = \frac{\text{real reduction}}{\text{predicted reduction}}$$

Trust-region methods

Basic principle: at iteration k, $k = 1, \ldots$, approximately minimize a model of the objective over a trust region ($\mathcal{B}_k$). It gives a (candidate iterate).

Compute the following ratio:

$$\rho = \frac{\text{real reduction}}{\text{predicted reduction}}$$

- If $\rho \geq \eta_1$, accept the candidate.
  If $\rho \geq \eta_2$, enlarge $\mathcal{B}_k$, otherwise keep it the same, or reduce it.
- If $\rho < \eta_1$, reject the candidate and reduce $\mathcal{B}_k$.

For instance, $\eta_1 = 0.01$ and $\eta_2 = 0.75$.

Trust-Region methods

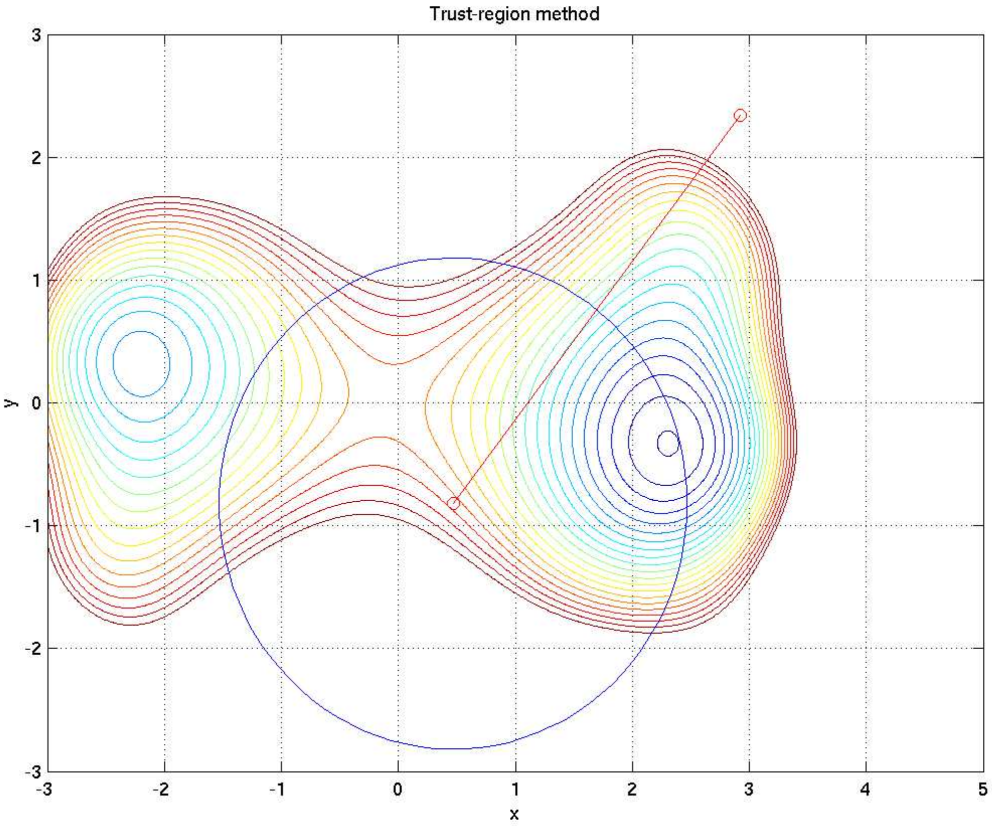
Example (Conn, Gould, Toint 2000):

$$\min_{x,y} -10x^2 + 10y^2 + 4\sin(xy) - 2x + x^4$$

Trust-Region methods

Example (Conn, Gould, Toint 2000):

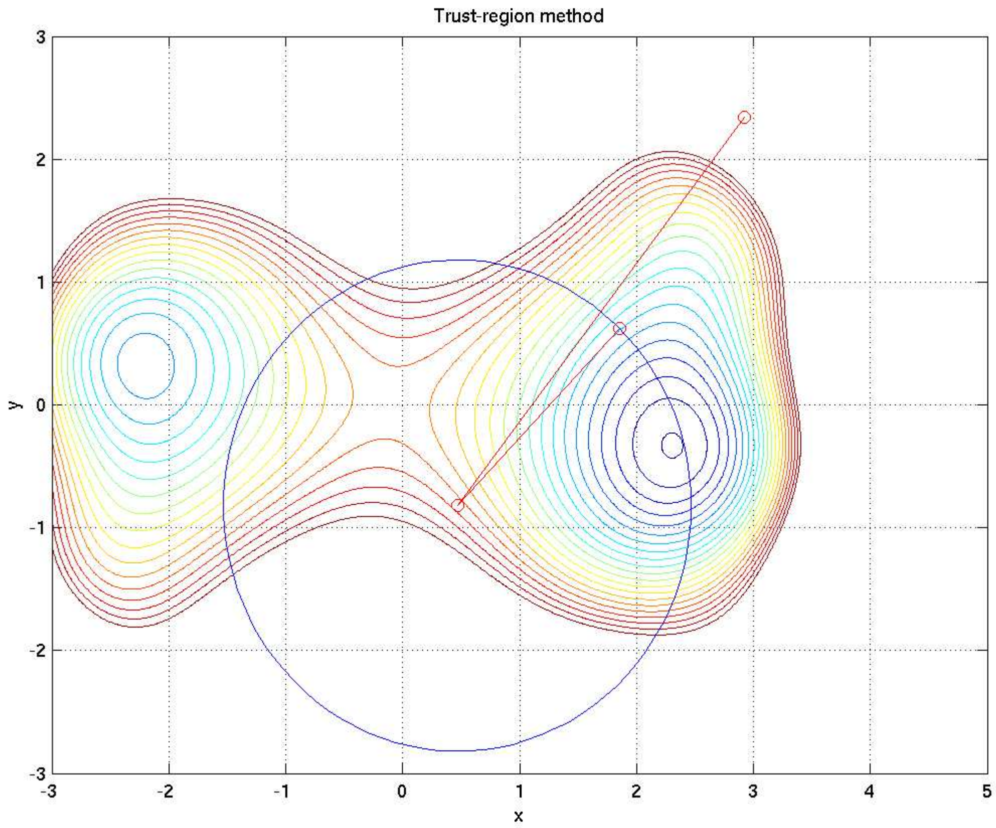
$$\min_{x,y} -10x^2 + 10y^2 + 4\sin(xy) - 2x + x^4$$



Trust-Region methods

Example (Conn, Gould, Toint 2000):

$$\min_{x,y} -10x^2 + 10y^2 + 4\sin(xy) - 2x + x^4$$



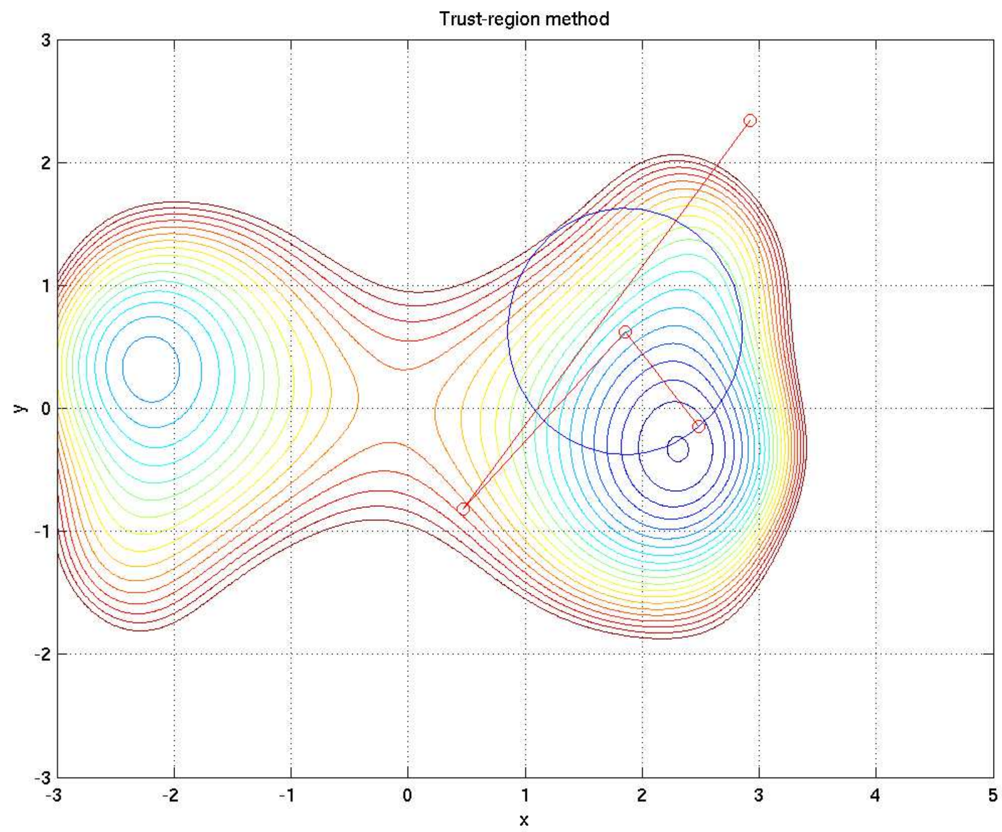
Outline
- Introduction
- Properties
- → ***Trust-region***
- Variable size
- AMLET
- Simulated data
- Halton
- Real data
- Conclusions

The University of Namur
www.fundp.ac.be

Trust-Region methods

Example (Conn, Gould, Toint 2000):

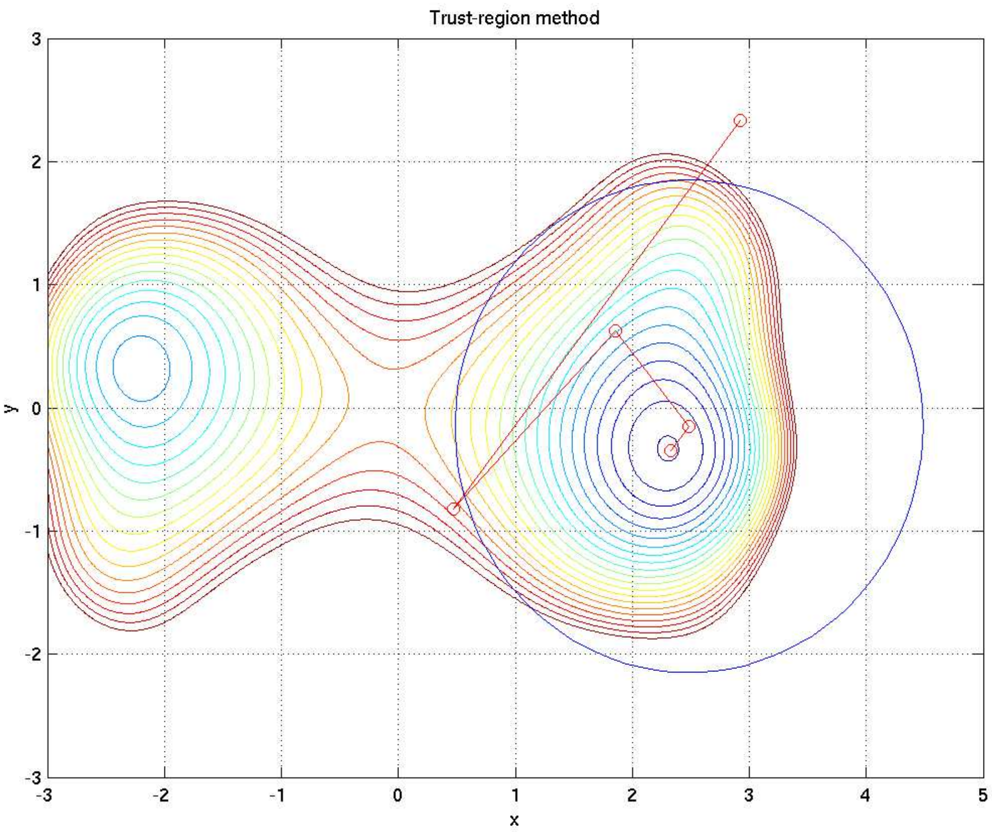
$$\min_{x,y} -10x^2 + 10y^2 + 4\sin(xy) - 2x + x^4$$



Trust-Region methods

Example (Conn, Gould, Toint 2000):

$$\min_{x,y} -10x^2 + 10y^2 + 4\sin(xy) - 2x + x^4$$



Outline
- Introduction
- Properties
- $\longrightarrow$ ***Trust-region***
- Variable size
- AMLET
- Simulated data
- Halton
- Real data
- Conclusions

The University of Namur

www.fundp.ac.be

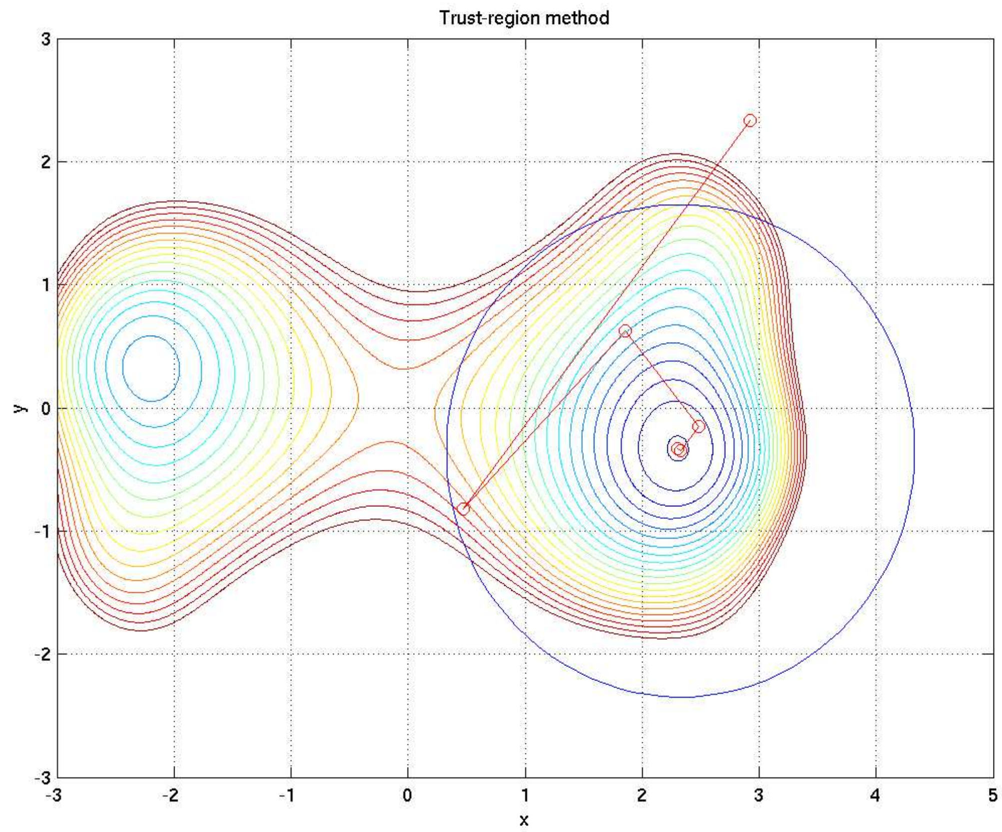
Trust-Region methods

Outline

- Introduction
- Properties
- → ***Trust-region***
- Variable size
- AMLET
- Simulated data
- Halton
- Real data
- Conclusions

Example (Conn, Gould, Toint 2000):

$$\min_{x,y} -10x^2 + 10y^2 + 4\sin(xy) - 2x + x^4$$

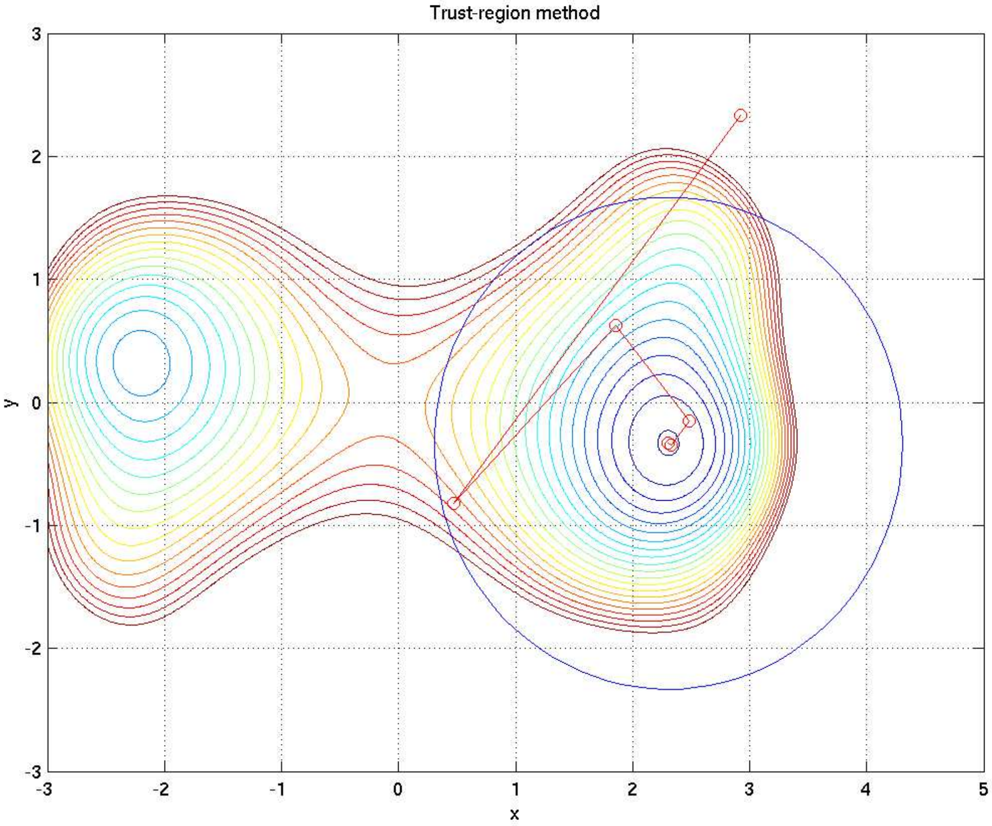


The University of Namur
www.fundp.ac.be

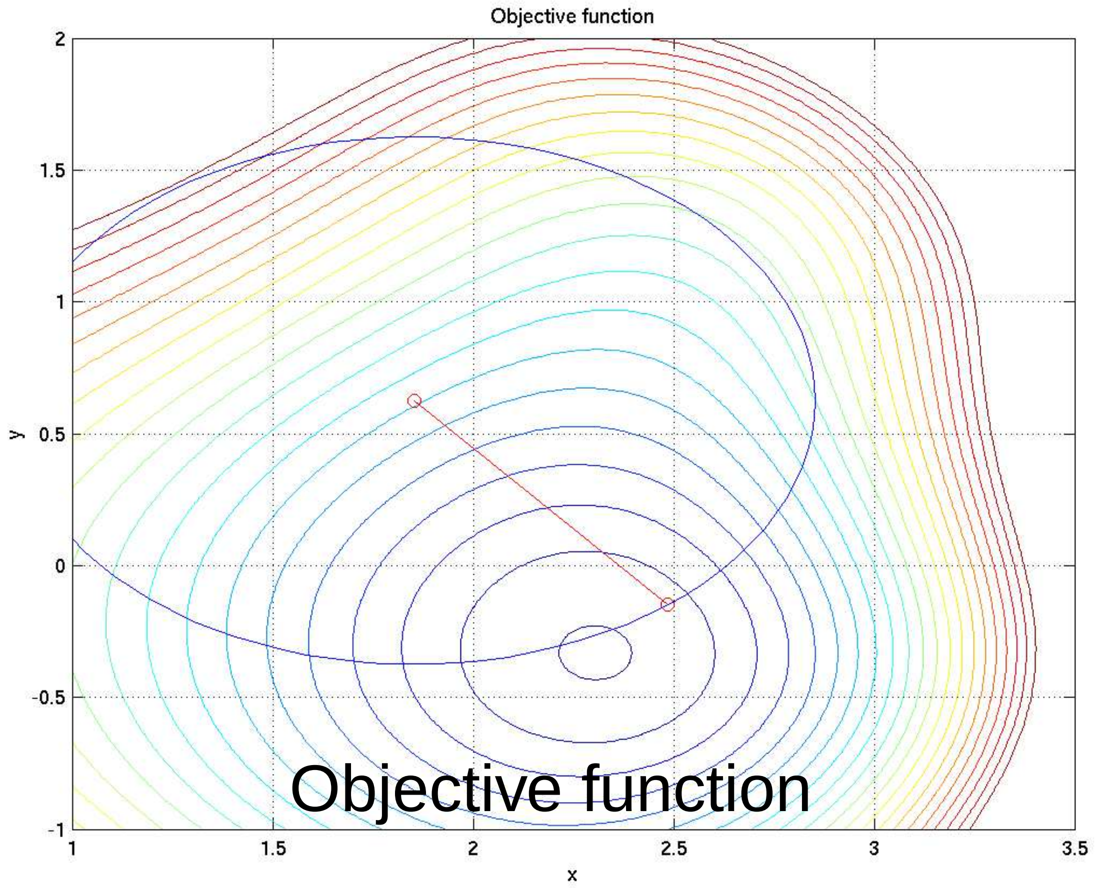
Trust-Region methods

Example (Conn, Gould, Toint 2000):

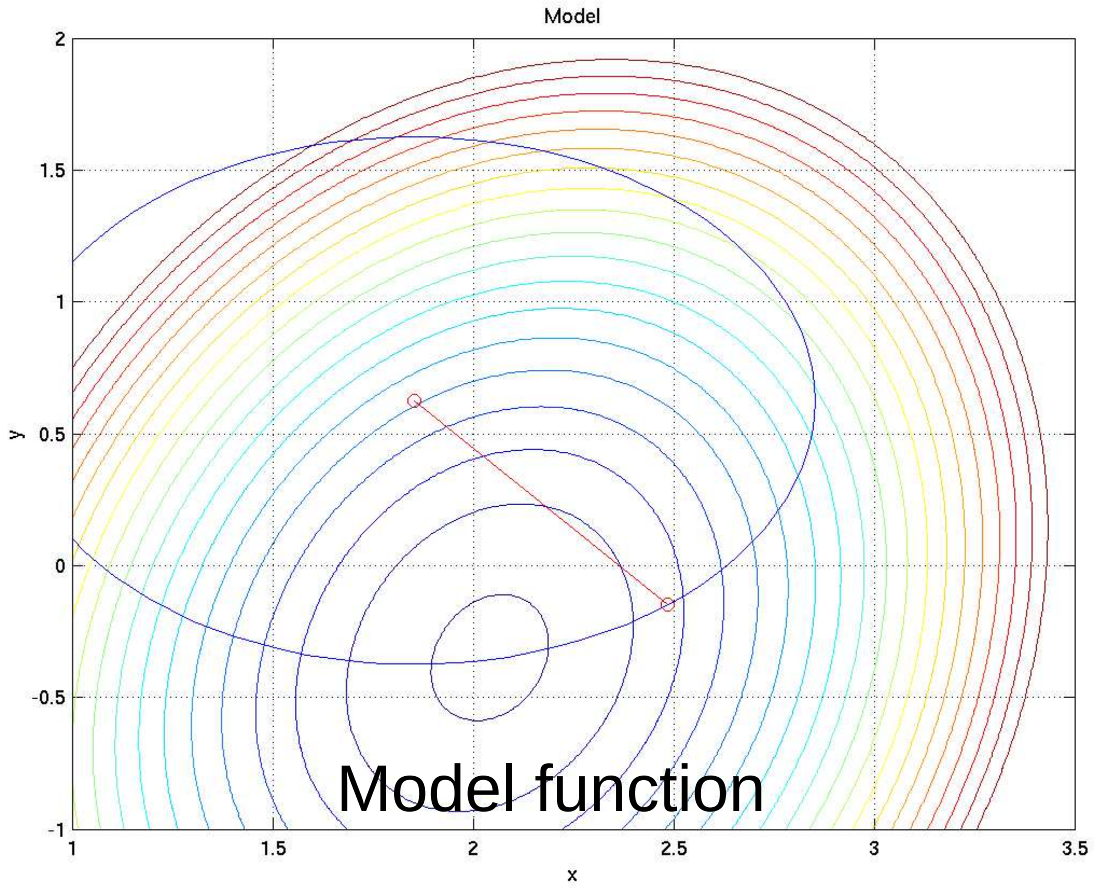
$$\min_{x,y} -10x^2 + 10y^2 + 4\sin(xy) - 2x + x^4$$



Model and objective comparison



Model and objective comparison



Outline
- Introduction
- Properties
- → ***Trust-region***
- Variable size
- AMLET
- Simulated data
- Halton
- Real data
- Conclusions

The University of Namur
www.fundp.ac.be

Variable sampling size technique

AMLET

New software: AMLET
(**A**nother **M**ixed **L**ogit **E**stimation **T**ool)

AMLET

New software: AMLET
(**A**nother **M**ixed **L**ogit **E**stimation **T**ool)

- multinomial logit and mixed logit models support;

AMLET

New software: AMLET
(**A**nother **M**ixed **L**ogit **E**stimation **T**ool)

- multinomial logit and mixed logit models support;
- works on real and simulated data;

AMLET

New software: AMLET
(**A**nother **M**ixed **L**ogit **E**stimation **T**ool)

- multinomial logit and mixed logit models support;
- works on real and simulated data;
- optimization algorithm: BFGS, BTR, BTRDA;

AMLET

New software: AMLET
(**A**nother **M**ixed **L**ogit **E**stimation **T**ool)

- multinomial logit and mixed logit models support;
- works on real and simulated data;
- optimization algorithm: BFGS, BTR, BTRDA;
- written in C, tested on a Linux system.

Outline

- Introduction
- Properties
- Trust-region
- Variable size
- → ***AMLET***
- Simulated data
- Halton
- Real data
- Conclusions

The University of Namur

www.fundp.ac.be

AMLET

New software: AMLET
(**A**nother **M**ixed **L**ogit **E**stimation **T**ool)

- multinomial logit and mixed logit models support;
- works on real and simulated data;
- optimization algorithm: BFGS, BTR, BTRDA;
- written in C, tested on a Linux system.

Working environment: Pentium IV, 2Ghz.

Tests on simulated data

Experimental design:

Tests on simulated data

Experimental design:

- attribute values drawn drom a $N(0,1)$;
- random parameters $\sim N(0.5,1)$;
- linear utilities;
- number of alternatives: 2, 3, 5, 10;
- observations: 2000, 5000, 7500, 1000.

Outline

The University of Namur
www.fundp.ac.be

Tests on simulated data

Experimental design:

- attribute values drawn drom a $N(0,1)$;
- random parameters $\sim N(0.5,1)$;
- linear utilities;
- number of alternatives: 2, 3, 5, 10;
- observations: 2000, 5000, 7500, 1000.

Comparisons with `Gauss` 5.0 with `MaxLik` modules, and code of Train (Halton, tol=10^{-6}).

Note: Gauss tested on Pentium III (1 licence); reported times are accordingly corrected.

Linesearch versus Trust-Region

Is the trust-region choice appropriate?

Linesearch versus Trust-Region

Is the trust-region choice appropriate?

Gauss	`MaxLik` module: BFGS with Dennis and Schnabel linesearch (STEPBT)
AMLET	BFGS with More-Thuente linesearch (recommendation of Nocedal and Wright) Basic Trust-Region (BTR) BTR with dynamic accuracy (BTRDA)

Outline

- Introduction
- Properties
- Trust-region
- Variable size
- AMLET
- → *Simulated data*
- Halton
- Real data
- Conclusions

The University of Namur

www.fundp.ac.be

Linesearch versus Trust-Region

Is the trust-region choice appropriate?

Gauss	`MaxLik` module: BFGS with Dennis and Schnabel linesearch (STEPBT)
AMLET	BFGS with More-Thuente linesearch (recommendation of Nocedal and Wright) Basic Trust-Region (BTR) BTR with dynamic accuracy (BTRDA)

Simulation experiment:

▷ 5000 individuals, 5 alternatives, 5 independant random parameters ($N(0.5, 1.0)$);

▷ minimization over 10 random samples of size 2000.

Linesearch versus Trust-Region (2)

Set	BFGS		BTR		BTRDA	
	Likelihood	Time (s)	Likelihood	Time (s)	Likelihood	Time (s)
1	-1.40529	1421	-1.40529	1103	-1.40529	722
2	-1.40532	1455	-1.40532	997	-1.40532	618
3	-1.40519	1691	-1.40519	1090	-1.40519	730
4	-1.40409	1736	-1.40409	997	-1.40409	564
5	-1.40525	1702	-1.40525	1017	-1.40525	531
6	-1.40499	1707	-1.40499	1089	-1.40499	587
7	-1.40437	1475	-1.40437	1045	-1.40437	560
8	-1.40530	1706	-1.40530	998	-1.40530	781
9	-1.40532	1701	-1.40532	1028	-1.40532	724
10	-1.40441	1729	-1.40441	1044	-1.40441	546
Mean	-1.40495	1632	-1.40495	1040	-1.40495	636

Linesearch versus Trust-Region (2)

Set	BFGS		BTR		BTRDA	
	Likelihood	Time (s)	Likelihood	Time (s)	Likelihood	Time (s)
1	-1.40529	1421	-1.40529	1103	-1.40529	722
2	-1.40532	1455	-1.40532	997	-1.40532	618
3	-1.40519	1691	-1.40519	1090	-1.40519	730
4	-1.40409	1736	-1.40409	997	-1.40409	564
5	-1.40525	1702	-1.40525	1017	-1.40525	531
6	-1.40499	1707	-1.40499	1089	-1.40499	587
7	-1.40437	1475	-1.40437	1045	-1.40437	560
8	-1.40530	1706	-1.40530	998	-1.40530	781
9	-1.40532	1701	-1.40532	1028	-1.40532	724
10	-1.40441	1729	-1.40441	1044	-1.40441	546
Mean	-1.40495	1632	-1.40495	1040	-1.40495	636

▷ Clear advantage to the trust-region approach.

Tests on simulated data

The University of Namur
www.fundp.ac.be

Comparison with Halton sequences

Halton sequences are popular: allow smaller sizes.

Comparison with Halton sequences

Halton sequences are popular: allow smaller sizes.

Comparisons with Gauss for low dimensions (≤ 5 random parameters): similar results, but AMLET remains faster (2 to 5 times for 1000 Monte-Carlo, compared to 125 Halton).

Comparison with Halton sequences

Halton sequences are popular: allow smaller sizes.

Comparisons with `Gauss` for low dimensions (≤ 5 random parameters): similar results, but `AMLET` remains faster (2 to 5 times for 1000 Monte-Carlo, compared to 125 Halton).

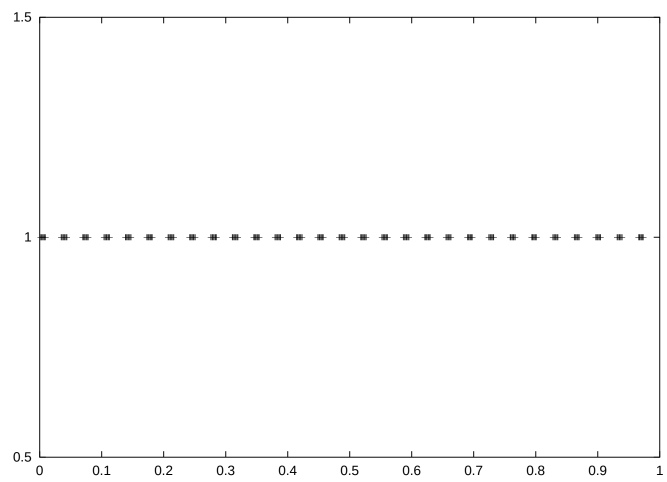
High dimensions with Halton sequences: loss of uniform coverage. Graphs with 250 Halton numbers:

Comparison with Halton sequences

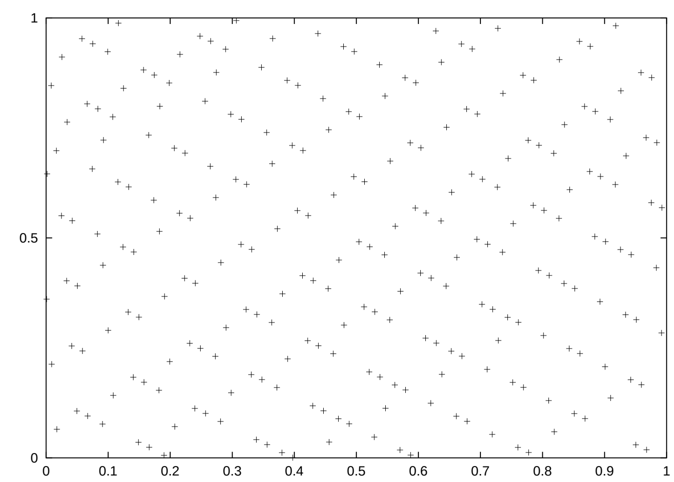
Halton sequences are popular: allow smaller sizes.

Comparisons with `Gauss` for low dimensions (≤ 5 random parameters): similar results, but `AMLET` remains faster (2 to 5 times for 1000 Monte-Carlo, compared to 125 Halton).

High dimensions with Halton sequences: loss of uniform coverage. Graphs with 250 Halton numbers:



10th prime number (29)



5th an 6th components

Outline

- Introduction
- Properties
- Trust-region
- Variable size
- AMLET
- Simulated data
- → *Halton*
- Real data
- Conclusions

The University of Namur

www.fundp.ac.be

Halton vs Monte-Carlo

Variable	125 Halt.	250 Halt.	1000 MC	2000 MC	3000 MC
P1 mean	0.4022	0.4371	0.437926	0.451574	0.452668
P1 std. dev.	0.9575	1.0546	1.06785	1.10345	1.10275
P2 mean	0.4237	0.4586	0.460888	0.47644	0.476496
P2 std. dev.	0.8423	0.8646	0.889336	0.924625	0.929872
P3 mean	0.3903	0.4305	0.428926	0.442901	0.444273
P3 std. dev.	0.7959	0.8812	0.881386	0.934353	0.934465
P4 mean	0.4700	0.5129	0.51648	0.533209	0.534616
P4 std. dev.	0.6744	0.7205	0.744546	0.77631	0.780702
P5 mean	0.4808	0.5308	0.531721	0.549948	0.5521
P5 std. dev.	0.7027	0.8312	0.847629	0.883452	0.886892
P6 mean	0.3782	0.4100	0.412724	0.425807	0.426386
P6 std. dev.	0.8297	0.9163	0.947523	0.987107	0.986711
P7 mean	0.3920	0.4337	0.435504	0.450424	0.450725
P7 std. dev.	0.9116	1.0398	1.03509	1.07814	1.0786
P8 mean	0.4895	0.5310	0.534592	0.551502	0.552986
P8 std. dev.	0.9198	1.0022	1.02172	1.054758	1.06168
P9 mean	0.4441	0.4768	0.481642	0.498321	0.49854
P9 std. dev.	0.9048	0.9694	0.977834	1.02296	1.02425
P10 mean	0.3702	0.4149	0.412852	0.426316	0.426021
P10 std. dev.	0.6250	0.8482	0.808399	0.835485	0.840841
Log-likelihood	-1.44244	-1.43990	-1.44132	-1.44086	-1.44077
Accuracy	NA	NA	0.001032	0.0007402	0.0006049
Bias	NA	NA	-0.0009844	-0.0005062	-0.0003381
Time (s)	2077	4252	435.8	890	1349.1

Halton vs Monte-Carlo: remarks

- AMLET very fast compared to Gauss;

Halton vs Monte-Carlo: remarks

- AMLET very fast compared to Gauss;
- Poor results with 125 Halton;

Halton vs Monte-Carlo: remarks

- AMLET very fast compared to Gauss;
- Poor results with 125 Halton;
- 250 Halton give similar solution to 1000 Monte-Carlo, but results change with 2000 Monte-Carlo, and Gauss underevaluates the objective;

Halton vs Monte-Carlo: remarks

- AMLET very fast compared to Gauss;
- Poor results with 125 Halton;
- 250 Halton give similar solution to 1000 Monte-Carlo, but results change with 2000 Monte-Carlo, and Gauss underevaluates the objective;
- Tests with 2000 individuals has given better results with 125 Halton than 250 Halton.

The University of Namur
www.fundp.ac.be

Halton vs Monte-Carlo: remarks

- AMLET very fast compared to Gauss;
- Poor results with 125 Halton;
- 250 Halton give similar solution to 1000 Monte-Carlo, but results change with 2000 Monte-Carlo, and Gauss underevaluates the objective;
- Tests with 2000 individuals has given better results with 125 Halton than 250 Halton.

Quid of Halton sequences? Our results incite us to be careful concerning their usage, but we still guess that good results can be obtained with scrambled sequences (see Bhat).

Tests on real data

Mode choice model: `Mobidrive` data (Axhausen and al.)

→ 5799 observations;

→ 5 alternatives;

→ 3 random parameters (normally distributed).

Tests on real data

Mode choice model: `Mobidrive` data (Axhausen and al.)

→ 5799 observations;

→ 5 alternatives;

→ 3 random parameters (normally distributed).

Note: correlation between observations not considered (feature planned for next `AMLET` release).

Tests on real data

Mode choice model: `Mobidrive` data (Axhausen and al.)

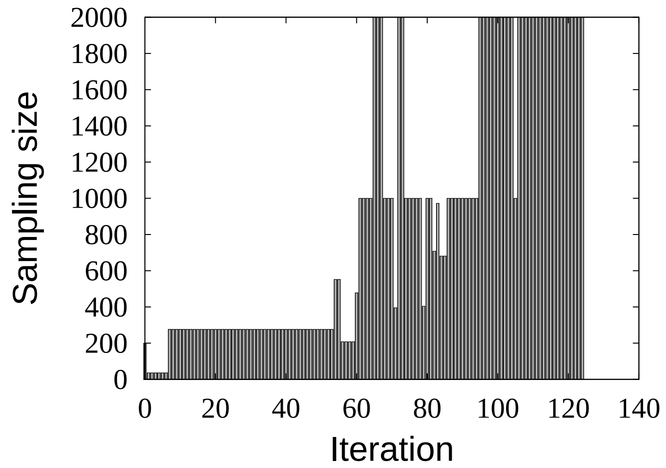
→ 5799 observations;

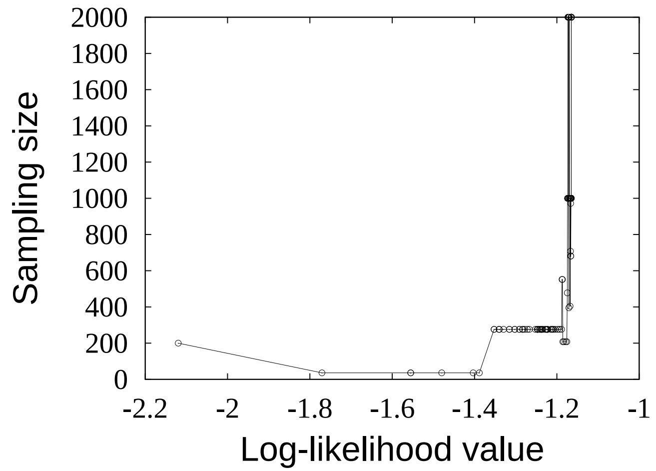
→ 5 alternatives;

→ 3 random parameters (normally distributed).

Note: correlation between observations not considered (feature planned for next `AMLET` release).

Evolution of sample sizes (2000 Monte-Carlo)





Tests on real data: results

Variable	Gauss (125 Halt.)	AMLET (1000 MC)	AMLET (2000 MC)
Car Passenger (CD)	-1.4511	-1.45104	-1.4527
Public Transport (PT)	-0.9355	-0.932594	-0.932458
Walk (W)	0.1081	0.109186	0.108793
Bike (B)	-0.6355	-0.634269	-0.635217
Urban household location (PT)	0.560609	0.557866	0.561515
Suburban household location (W, B)	-0.3451	-0.345403	-0.345113
Full-time worker (PT)	0.2690	0.269265	0.268996
Female and part-time (CP)	0.9133	0.912925	0.913835
Married with children (CD)	0.9716	0.970755	0.971656
Annual mileage (CD)	0.0518	0.0518679	0.0519161
Number of stop (CD)	0.1349	0.135187	0.135817
Time mean	-0.0268	-0.0268882	-0.0269985
Time std. dev.	0.0205	0.0206265	0.0208197
Cost mean	-0.1683	-0.168923	-0.169365
Cost std. dev.	-0.0452	0.0465829	0.0465628
Time budget/100 mean (CD, CP)	-0.1249	-0.124816	-0.125128
Time budget/100 std. dev. (CD, CP)	-0.1136	0.112801	0.113803
Log-likelihood	-1.16489	-1.16479	-1.16470
Bias	Not available	-0.00009117	-0.0000463
Accuracy	Not available	0.0002916	0.0002086
Time (s)	2439	936	1549

Outline

- Introduction
- Properties
- Trust-region
- Variable size
- AMLET
- Simulated data
- Halton
- → ***Real data***
- Conclusions

The University of Namur

www.fundp.ac.be

Conclusions

- New Monte-Carlo algorithm for nonlinear stochastic programming;

The University of Namur
www.fundp.ac.be

Conclusions

- New Monte-Carlo algorithm for nonlinear stochastic programming;
- Estimation of error and bias in the objective
  - allow variable sampling size techniques;
  - not sufficient to know stability of solutions;

The University of Namur
www.fundp.ac.be

Conclusions

- New Monte-Carlo algorithm for nonlinear stochastic programming;
- Estimation of error and bias in the objective
  - allow variable sampling size techniques;
  - not sufficient to know stability of solutions;
- Behaviour seems to be better when the number of alternatives rises;

Conclusions

- New Monte-Carlo algorithm for nonlinear stochastic programming;
- Estimation of error and bias in the objective
  - allow variable sampling size techniques;
  - not sufficient to know stability of solutions;
- Behaviour seems to be better when the number of alternatives rises;
- Implementation issues are important:
  - choice of optimization method;
  - implementation tricks (e.g.: possible to compute objective and gradient simultaneously).

Perspectives

- AMLET still on a prototype level:
  - more tests on real data would be useful;
  - planned features:
    - other distributions than normals;
    - correlation between observations;

Perspectives

- AMLET still on a prototype level:
  - more tests on real data would be useful;
  - planned features:
    - other distributions than normals;
    - correlation between observations;
- variable sampling size algorithm could be certainly refined;

Perspectives

- AMLET still on a prototype level:
  - more tests on real data would be useful;
  - planned features:
    - other distributions than normals;
    - correlation between observations;
- variable sampling size algorithm could be certainly refined;
- investigation of quasi Monte-Carlo methods:
  - exploitable error bounds?
  - exploration of scrambled sequences;
  - could reduce computation time and memory needs.

The University of Namur
www.fundp.ac.be