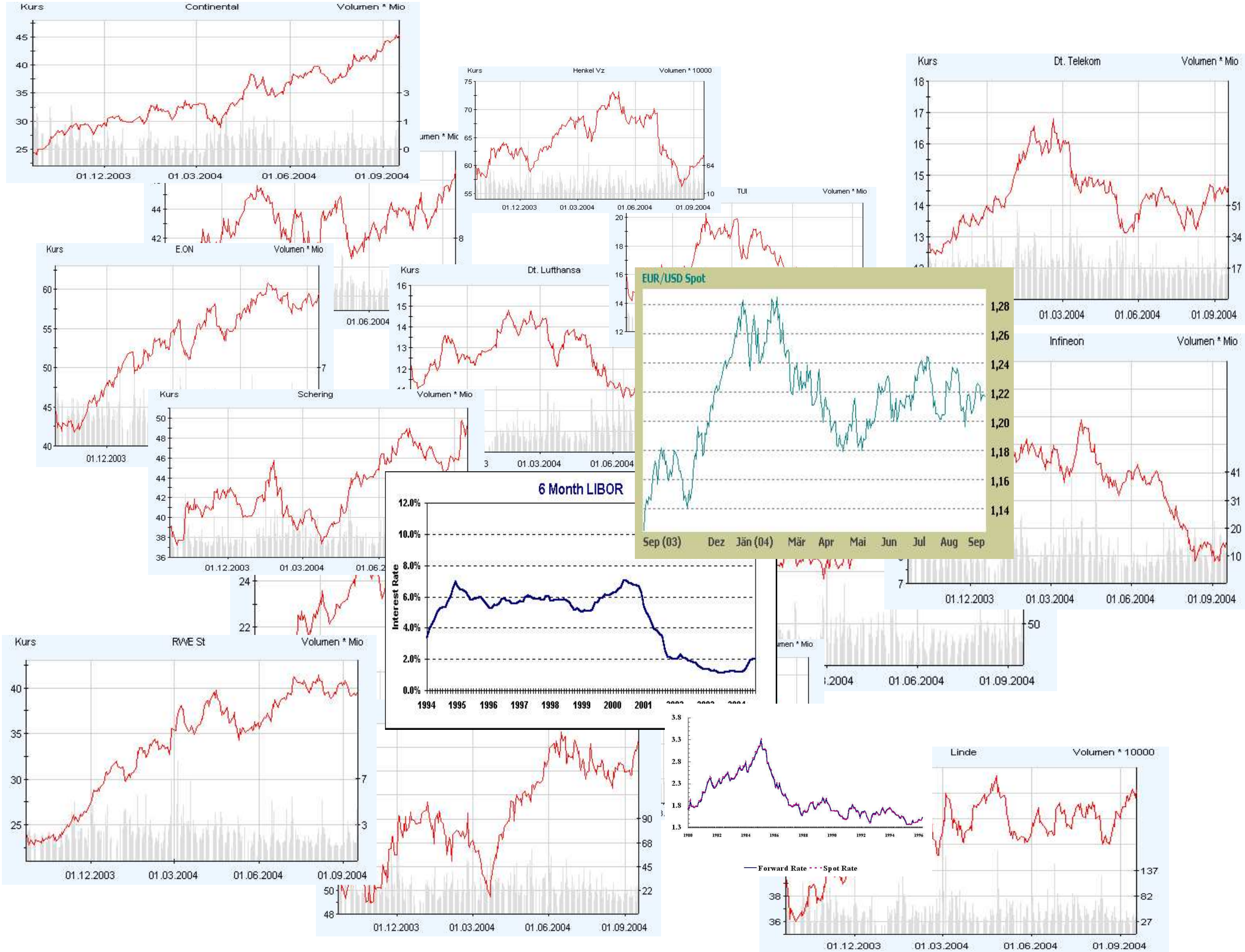
Computational Strategies for High Dimensional Option Pricing Problems

Christoph Reisinger

christoph.reisinger@maths.ox.ac.uk.

OCIAM, Mathematical Institute, Oxford University

Why high dimensional?



Potential factors

- equities $\quad \mathrm{d}S_i = r S_i \,\mathrm{d}t + \sigma_{S_i} S_i \,\mathrm{d}W_{S_i}$ (eg GBM)

  eg basket options $i \leq d = 10, 1000$

- FX-rates $\quad \mathrm{d}x = (r_d - r_f) x \,\mathrm{d}t + \sigma_x x \,\mathrm{d}W_x$

- short rates $\quad \mathrm{d}r = \kappa(\theta - r) \,\mathrm{d}t + \sigma_r \,\mathrm{d}W_r$ (eg Hull-White)

- LIBOR rates $\quad \mathrm{d}L_i = \mu_i(L_i) L_i \,\mathrm{d}t + \sigma_{L_i} L_i \,\mathrm{d}W_{L_i}$ (BGM)

  eg Bermudan swaptions $i \leq d = 10, 100$

Example I: FX option

- exchange rate x
- spot rates r_d (*domestic*) und r_f (*foreign*)
- European put u on x, strike K, expiry T

$$\frac{\partial u}{\partial t}+\frac{1}{2}\sigma_x^2 x^2\frac{\partial^2 u}{\partial x^2}+\frac{1}{2}\sum_{i=d,f}\sigma_{r_i}^2\frac{\partial^2 u}{\partial {r_i}^2}+(r_d-r_f)x\frac{\partial u}{\partial x}+\sum_{i=d,f}\kappa_i(\theta_i-r_i)\frac{\partial u}{\partial x_i}-r_d u = 0$$

terminal condition $\quad u(x,r_d,r_f,T)=(K-x)^+$

- $\sigma_{r_i}=0.15$, $\kappa_i=0.5$, $\theta_i=0.045$, $K=0.95$, $T=10\ a$
- spot price $S=0.9$, spot rates $r_d=r_f=0.05$

Example II: basket options

- stocks $S_1, \ldots, S_d$ with covariance $\mathbf{\Sigma} = (\sigma_{ij})$
- basket $\sum_{i=1}^{d} \mu_i S_i$ with strike K
- European/American put u on basket

Black-Scholes equation

$$\frac{\partial u}{\partial t} + \frac{1}{2} \sum_{i,j=1}^{d} \sigma_{ij} S_i S_j \frac{\partial^2 u}{\partial S_i \partial S_j} + r \sum_{i=1}^{d} S_i \frac{\partial u}{\partial S_i} - ru = 0$$

or associated obstacle problem

terminal condition $\quad u(\mathbf{S}, T) = \left(K - \sum_{i=1}^{d} \mu_i S_i\right)^+$

Example III: swaptions

- tenor structure $T_0 = 0, T_1, \ldots, T_d = T$
- LIBOR rates L_i with volatilities σ_i, correlation $\rho_{ij} = \mathrm{e}^{-\alpha|i-j|}$
- Bermudan swaption u

BGM model

$$\frac{\partial u}{\partial t} + \frac{1}{2} \sum_{i,j=1}^{d} \sigma_i \sigma_j \rho_{ij} L_i L_j \frac{\partial^2 u}{\partial L_i \partial L_j} - \sum_{j=1}^{d} \mu_j(\mathbf{L}) L_j \frac{\partial u}{\partial L_j} = 0$$

with

$$\mu_j(\mathbf{L}) = \begin{cases} \sigma_j \sum_{k=j+1}^{d} \frac{\delta_k L_k}{1+\delta_k L_k} \sigma_k \rho_{jk} & j < d \\ 0 & j = d \end{cases}$$

Model framework

Ito processes for underlyings (equities, FX-rates, interest rates,...) or parameters (volatilities, interest rates,...)

$$\mathrm{d}x_i = \beta_i(\mathbf{x}, t)\ \mathrm{d}t + \alpha_i(\mathbf{x}, t)\ \mathrm{d}W_i$$

(Anti-)Parabolic PDE for option in $(\underline{x}_1, \overline{x}_1) \times \ldots \times (\underline{x}_d, \overline{x}_d)$

$$\frac{\partial u}{\partial t} + \sum_{i,j=1}^{d} a_{ij}(\mathbf{x}) \frac{\partial^2 u}{\partial x_i \partial x_j} - \sum_{i=1}^{d} b_i(\mathbf{x}) \frac{\partial u}{\partial x_i} - c(\mathbf{x}) u = 0,$$

where

$$a_{ij}(\mathbf{x}) = 0 \quad \text{for} \quad x_i \in \{\underline{x}_i, \overline{x}_i\},$$

$$b_i(\mathbf{x}) = 0 \quad \text{for} \quad x_i \in \{\underline{x}_i, \overline{x}_i\} \text{ or at least}$$

$$b_i(\mathbf{x}) n_i(\mathbf{x}) \geq 0.$$

$\rightarrow$ **existence & uniqueness** without boundary conditions

Zhou, Li: *Multi-factor Financial Derivatives on Finite Domains*, 2003.

What is high dimensional?

Example DAX 30: If each asset is represented by only two states, the total number of variables is already

$$\mathbf{2^{30} = 1\ 073\ 741\ 824}.$$

What is high dimensional?

Example DAX 30: If each asset is represented by only two states, the total number of variables is already

$$\mathbf{2^{30} = 1\ 073\ 741\ 824}.$$

If we choose a reasonable number of points in each direction, say $32 = 2^5$, the same total number is already obtained for

$$\mathbf{dim = 6}.$$

Overall strategy

Model transformation and/or reduction:

Principal components
Asymptotic analysis
...

Optimal discrete approximation spaces:

Sparse grids
Dimensional adaptivity
...

Optimal complexity solution algorithm:

Multigrid solver
Robust relaxation
...

Fast software on parallel platform:

Efficient datastructures
Parallel programming
...

Expansions

Sparse grids

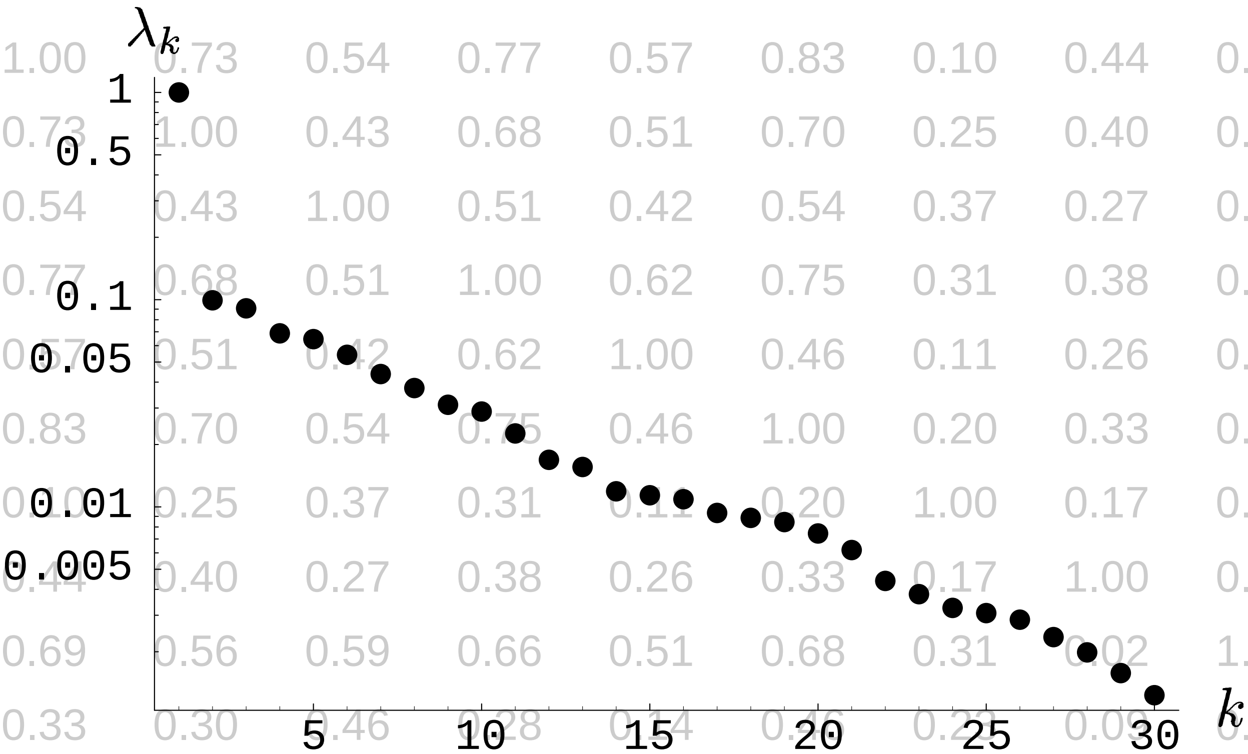
Multigrid

Implementation

Correlation data (DAX)

1.00	0.62	0.65	0.59	0.33	0.61	0.57	0.66	0.01	0.28	0.61	0.38	0
0.62	1.00	0.77	0.74	0.45	0.74	0.59	0.76	0.21	0.34	0.64	0.27	0
0.65	0.77	1.00	0.73	0.54	0.77	0.57	0.83	0.10	0.44	0.69	0.33	0
0.59	0.74	0.73	1.00	0.43	0.68	0.51	0.70	0.25	0.40	0.56	0.30	0
0.33	0.45	0.54	0.43	1.00	0.51	0.42	0.54	0.37	0.27	0.59	0.46	0
0.61	0.74	0.77	0.68	0.51	1.00	0.62	0.75	0.31	0.38	0.66	0.28	0
0.57	0.59	0.57	0.51	0.42	0.62	1.00	0.46	0.11	0.26	0.51	0.14	0
0.66	0.76	0.83	0.70	0.54	0.75	0.46	1.00	0.20	0.33	0.68	0.46	0
0.01	0.21	0.10	0.25	0.37	0.31	0.11	0.20	1.00	0.17	0.31	0.23	0
0.28	0.34	0.44	0.40	0.27	0.38	0.26	0.33	0.17	1.00	0.02	0.09	0
0.61	0.64	0.69	0.56	0.59	0.66	0.51	0.68	0.31	0.02	1.00	0.42	0
0.38	0.27	0.33	0.30	0.46	0.28	0.14	0.46	0.23	0.09	0.42	1.00	0
0.60	0.72	0.68	0.66	0.44	0.83	0.69	0.65	0.21	0.23	0.64	0.19	1
0.55	0.52	0.63	0.63	0.51	0.50	0.35	0.63	0.34	0.23	0.59	0.35	0
0.39	0.67	0.66	0.58	0.55	0.57	0.37	0.64	0.21	0.36	0.51	0.27	0
0.51	0.57	0.55	0.62	0.30	0.40	0.64	0.49	0.03	0.12	0.52	0.16	0
0.59	0.69	0.75	0.61	0.53	0.75	0.62	0.73	0.15	0.21	0.71	0.41	0
0.65	0.82	0.8	0.89	0.49	0.73	0.51	0.76	0.25	0.47	0.64	0.31	0
0.34	0.63	0.52	0.56	0.42	0.6	0.33	0.62	0.19	0.1	0.47	0.18	0

Correlation data (DAX)



$\frac{\sigma(\Sigma)}{\lambda_1}$, DAX (16. 1. 2003)

- spectral gap after λ_1
- exponential decay from λ_2

Example PCA

stock	weight	rel. vola	correlation				
Deutsche B.	38.1	0.518	1.00	0.79	0.82	0.91	0.84
Hypo-Vereins.	6.5	0.648	0.79	1.00	0.73	0.80	0.76
Commerzb.	5.7	0.623	0.82	0.73	1.00	0.77	0.72
Allianz	27.0	0.570	0.91	0.80	0.77	1.00	0.90
Münch. Rück	22.7	0.530	0.84	0.76	0.72	0.90	1.00

- eigenvalues: $\{1.409, 0.113, 0.101, 0.0388, 0.0213\}$
- 'principal component': $(0.185, 0.221, 0.209, 0.204, 0.182)$

Example PCA

stock	weight	rel. vola	correlation				
Deutsche B.	38.1	0.518	1.00	0.79	0.82	0.91	0.84
Hypo-Vereins.	6.5	0.648	0.79	1.00	0.73	0.80	0.76
Commerzb.	5.7	0.623	0.82	0.73	1.00	0.77	0.72
Allianz	27.0	0.570	0.91	0.80	0.77	1.00	0.90
Münch. Rück	22.7	0.530	0.84	0.76	0.72	0.90	1.00

- eigenvalues: $\{1.409, 0.113, 0.101, 0.0388, 0.0213\}$
- 'principal component': $(0.185, 0.221, 0.209, 0.204, 0.182)$

error of lower-dim. approximations, $T = 1a$, a-t-m:

1D	2D	3D	4D
6.24 %	4.99 %	2.50 %	0.87 %

Asymptotics

Expand the solution in the small parameters $\lambda_2, \ldots, \lambda_d$

- $u(\mathbf{S}, t, \boldsymbol{\lambda}) = u_1(\mathbf{S}, t, \lambda_1) + \sum_{k=2}^{d} \lambda_k \frac{\partial u}{\partial \lambda_k}\Big|_{(\mathbf{S}, t, \lambda_1)} + R(\boldsymbol{\lambda})$

Residual $R(\boldsymbol{\lambda})$ for 5D example $\mathbf{0.06\%}$, for DAX 30 $\mathbf{0.05\%}$!

Asymptotics

Expand the solution in the small parameters $\lambda_2, \ldots, \lambda_d$

- $u(\mathbf{S}, t, \boldsymbol{\lambda}) = u_1(\mathbf{S}, t, \lambda_1) + \sum_{k=2}^{d} \lambda_k \frac{\partial u}{\partial \lambda_k} \Big|_{(\mathbf{S}, t, \lambda_1)} + R(\boldsymbol{\lambda})$

Residual $R(\boldsymbol{\lambda})$ for 5D example 0.06%, for DAX 30 0.05%!

- $\frac{\partial u}{\partial \lambda_j}(\mathbf{S}, t, \boldsymbol{\lambda}) = \frac{\overbrace{u(\mathbf{S}, t, \{\lambda_1, 0, \ldots, 0, \lambda_j, 0, \ldots\})}^{=: u_j(\mathbf{S}, t, \lambda_1, \lambda_j)} - u_1(\mathbf{S}, t, \lambda_1)}{\lambda_j} + R(\boldsymbol{\lambda})$

- $u(\mathbf{S}, t, \boldsymbol{\lambda}) = (2 - d) \cdot \underbrace{u_1(\mathbf{S}, t, \lambda_1)}_{\text{1-dim}} + \sum_{j=2}^{d} \underbrace{u_j(\mathbf{S}, t, \lambda_1, \lambda_j)}_{\text{2-dim}} + R(\boldsymbol{\lambda})$

Asymptotics

Expand the solution in the small parameters $\lambda_2, \ldots, \lambda_d$

- $u(\mathbf{S}, t, \boldsymbol{\lambda}) = u_1(\mathbf{S}, t, \lambda_1) + \sum_{k=2}^{d} \lambda_k \frac{\partial u}{\partial \lambda_k}\Big|_{(\mathbf{S}, t, \lambda_1)} + R(\boldsymbol{\lambda})$

Residual $R(\boldsymbol{\lambda})$ for 5D example 0.06%, for DAX 30 0.05%!

- $\frac{\partial u}{\partial \lambda_j}(\mathbf{S}, t, \boldsymbol{\lambda}) = \frac{\overbrace{u(\mathbf{S}, t, \{\lambda_1, 0, \ldots, 0, \lambda_j, 0, \ldots\})}^{=: u_j(\mathbf{S}, t, \lambda_1, \lambda_j)} - u_1(\mathbf{S}, t, \lambda_1)}{\lambda_j} + R(\boldsymbol{\lambda})$
- $u(\mathbf{S}, t, \boldsymbol{\lambda}) = (2 - d) \cdot \underbrace{u_1(\mathbf{S}, t, \lambda_1)}_{\text{1-dim}} + \sum_{j=2}^{d} \underbrace{u_j(\mathbf{S}, t, \lambda_1, \lambda_j)}_{\text{2-dim}} + R(\boldsymbol{\lambda})$

$\Rightarrow$ only (numerical) solution of 2-dim PDEs required!

Expansions

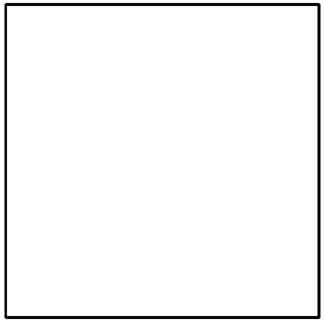
Sparse grids

Multigrid

Implementation

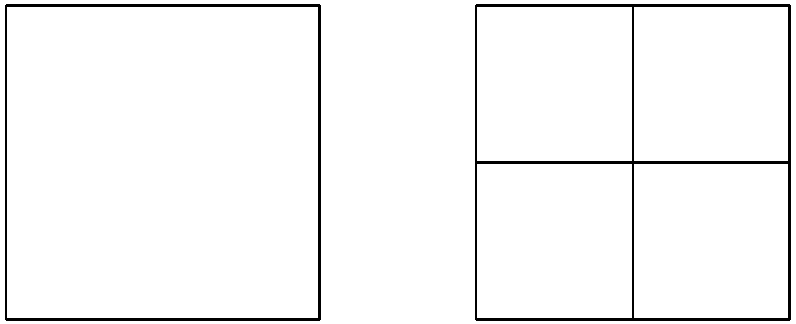
Grid-based methods

Isotropic Cartesian (‘full’) grid on cubic domain:

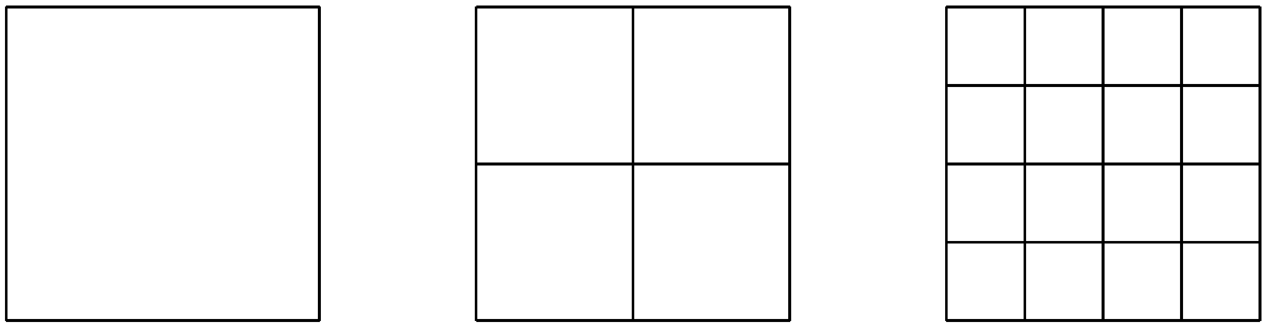


Grid-based methods

Isotropic Cartesian ('full') grid on cubic domain:

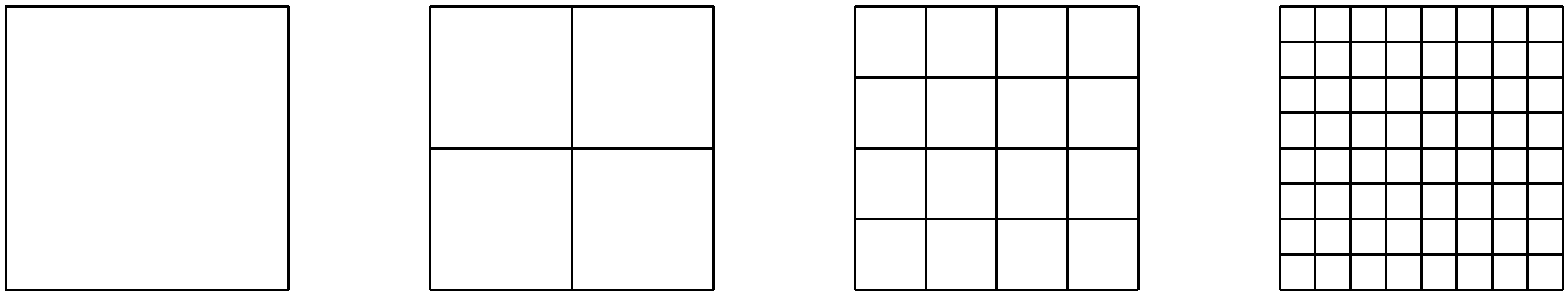


Isotropic Cartesian ('full') grid on cubic domain:



Grid-based methods

Isotropic Cartesian ('full') grid on cubic domain:

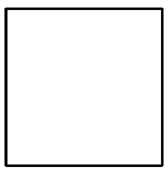


'Curse of dimensionality':

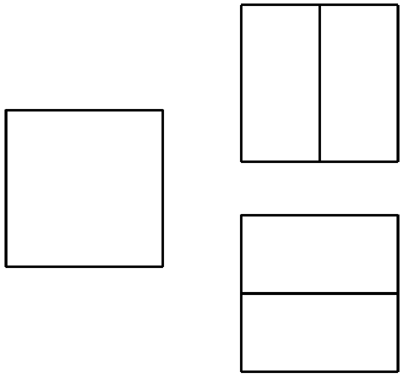
- on level n we face $N \sim 2^{nd}$ points/elements
- accuracy ϵ requires $N(\epsilon) \sim \epsilon^{-d/2}$ degrees of freedom (order 2)

$\rightarrow$ exponential growth of unknowns

Sparse grids



Sparse grids



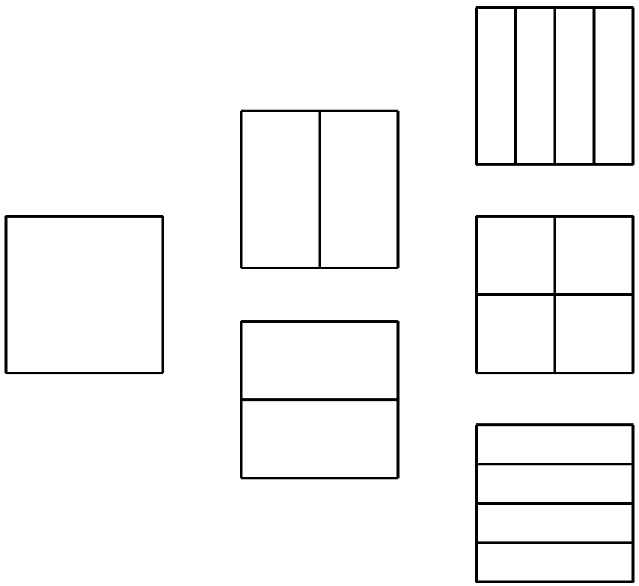
Sparse grids

Expansions

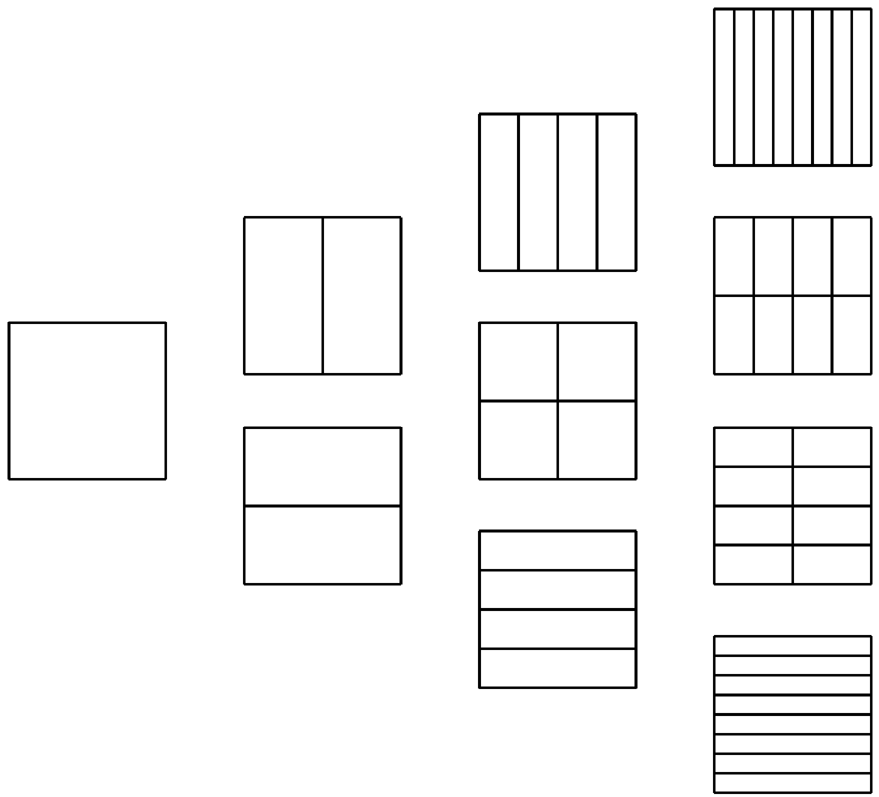
Sparse grids

Multigrid

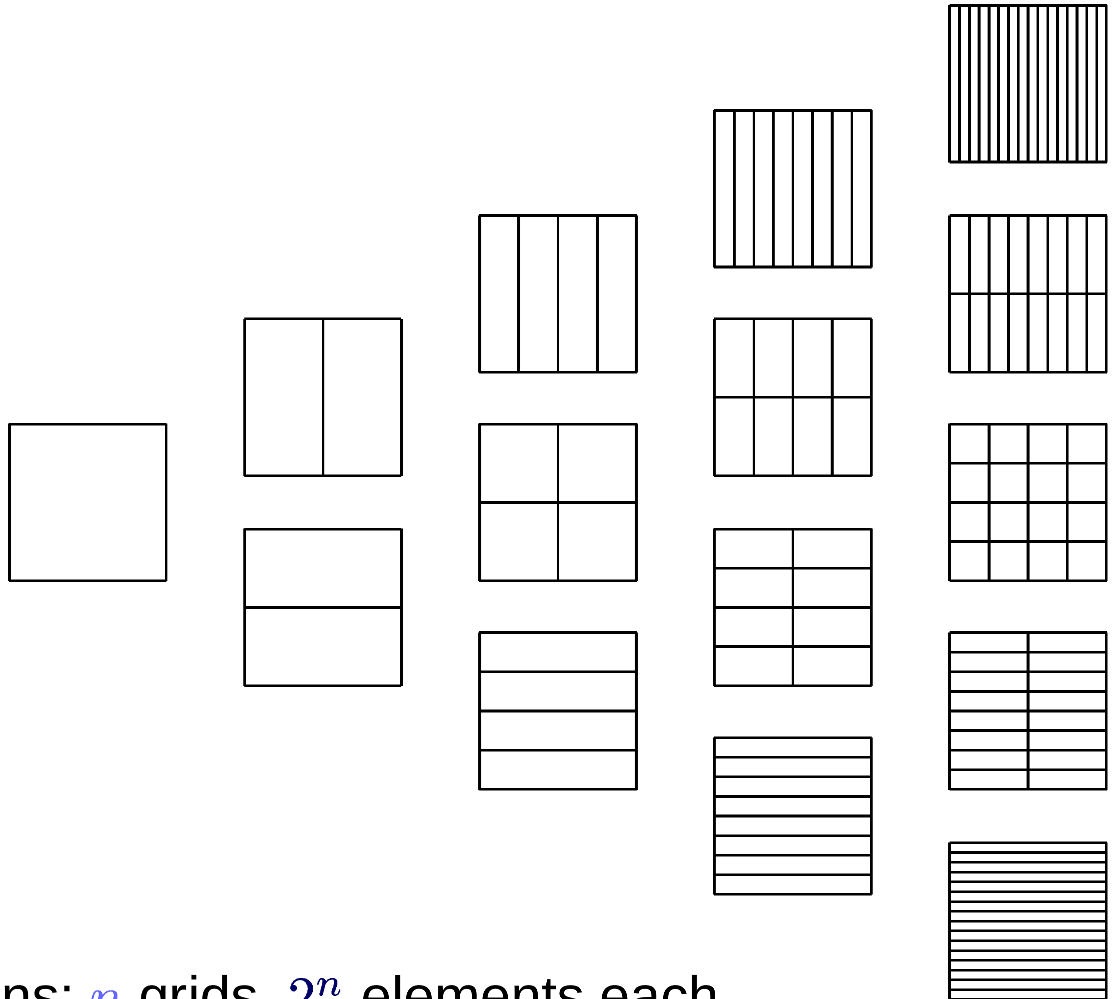
Implementation



Sparse grids

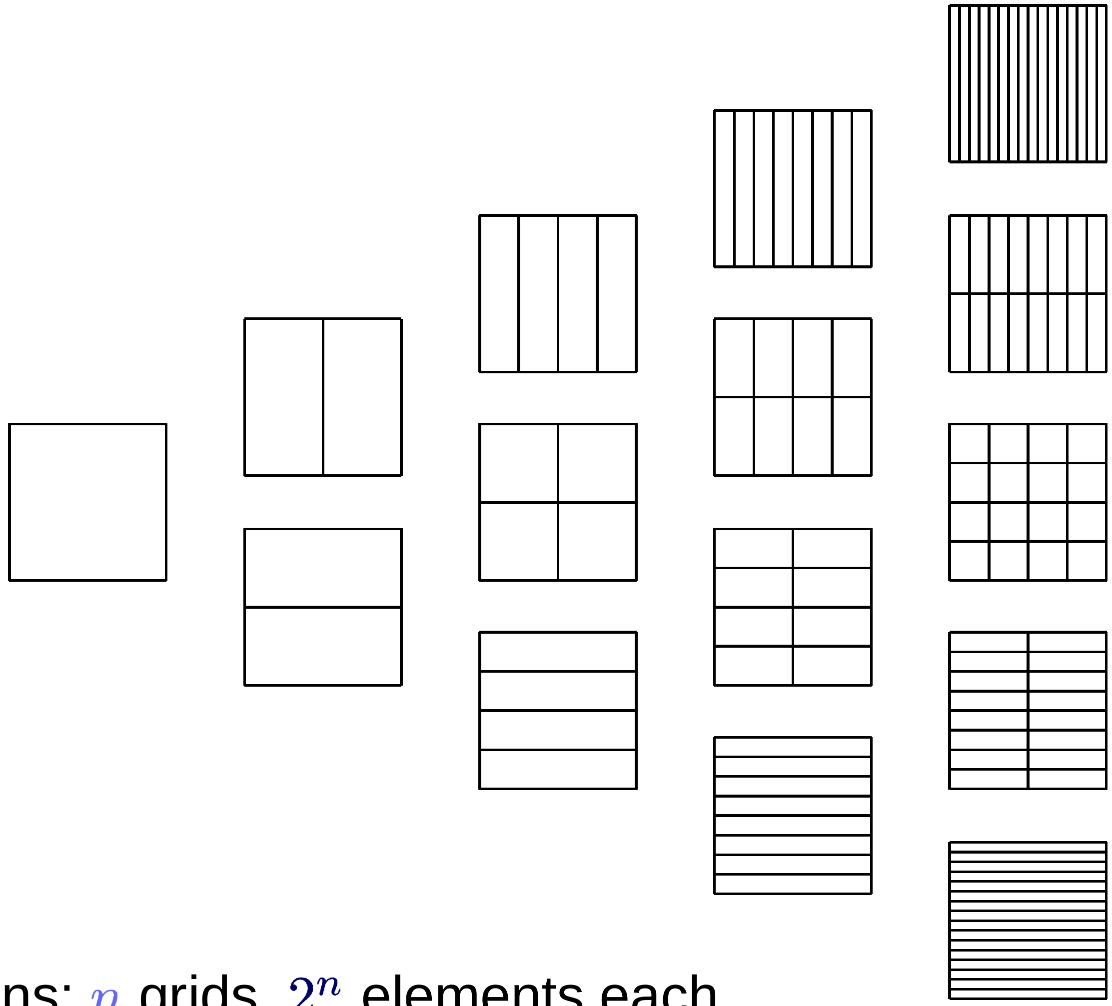


Sparse grids



2 dimensions: n grids, 2^n elements each

Sparse grids



2 dimensions: n grids, 2^n elements each

d dimensions: $\begin{pmatrix} n+d-1 \\ d-1 \end{pmatrix}$ grids, 2^n elements each

Approximation property

For sufficiently smooth functions [Bungartz, 1992]

$$\begin{aligned}\|u - u_n\|_\infty &\leq \frac{1}{6^d}\left\|\frac{\partial^{2d} u}{\partial x_1^2 \ldots \partial x_d^2}\right\|_\infty \left(1 + \sum_{i=1}^{d-1}\left(\frac{3}{4}\right)^i \binom{n+i-1}{i}\right) 2^{-2n} \\ &= \mathcal{O}(n^{d-1} 2^{-2n})\end{aligned}$$

for an interpolating multilinear spline on the sparse grid of level n.

The number of nodes/elements is

$$N_n = \mathcal{O}(n^{d-1} 2^n),$$

finest grid size $h = 2^{-n}$.

Sparse FE/wavelets

- Bungartz: *Finite Elements of Higher Order on Sparse Grids*, 1996.

$$\|u_h - u\|_E \leq c(d)h^p$$

for finite elements of order p and $\partial^{\boldsymbol{\alpha}} u \in \mathcal{C}, |\boldsymbol{\alpha}|_\infty \leq p+1$.

- Petersdorff, Schwab: *Numerical Solution of Parabolic Equations in High Dimensions*, 2002.

For wavelets of order p, $u(.,0) \in H^\epsilon$, the semidiscrete solution fulfills ($\theta_0 \in]0,1]$, $\delta > 0$)

$$\|u_h(t) - u(t)\|_2 \leq C h^{\theta_0 p + \delta} t^{-(p+1)d/2} \|u_0\|_2$$

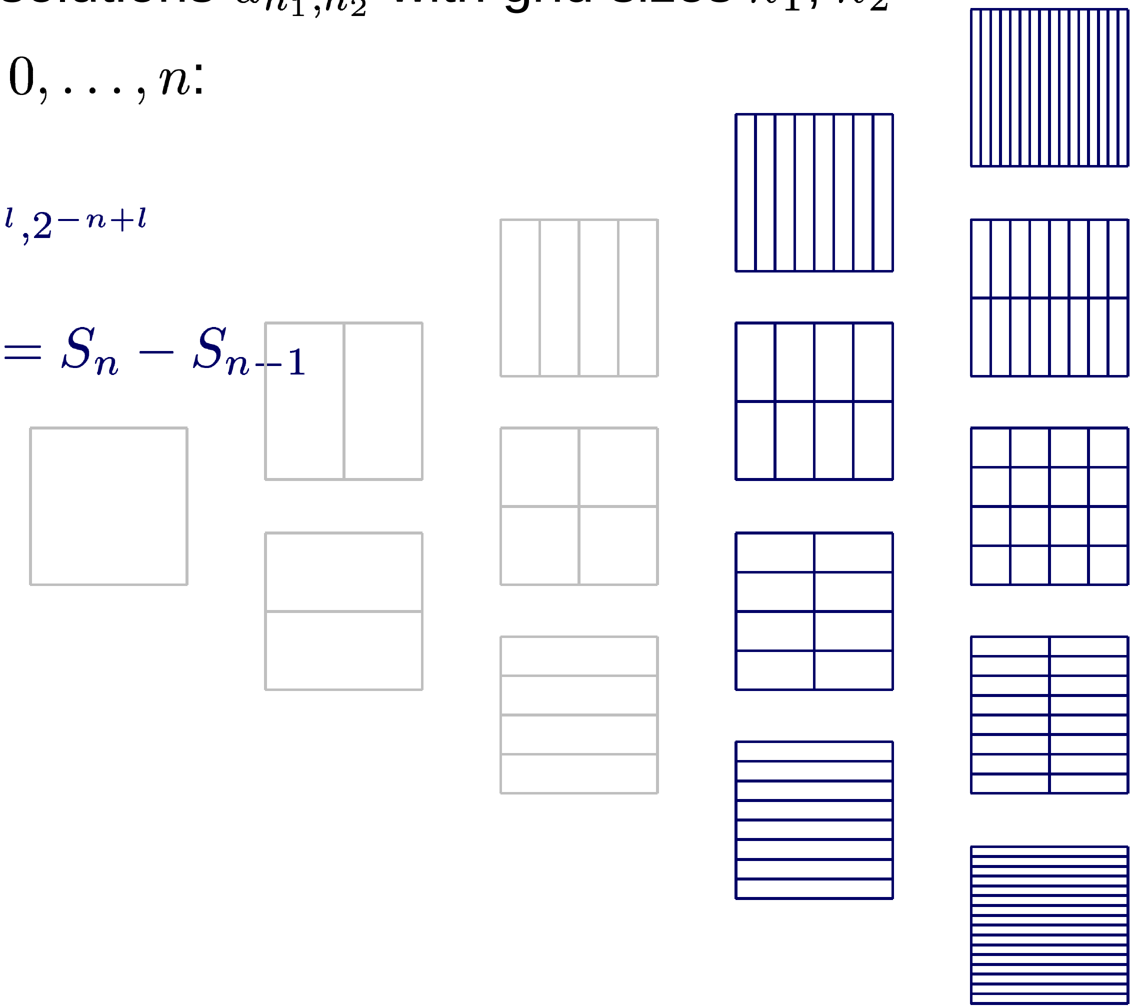
hp-Discontinuous-Galerkin time stepping, order '$r = \log h$':

$$\|\hat{u}_h(t) - u(t)\|_2 \leq C h^{\theta_0 p + \delta}.$$

Combination technique

2 dimensions:

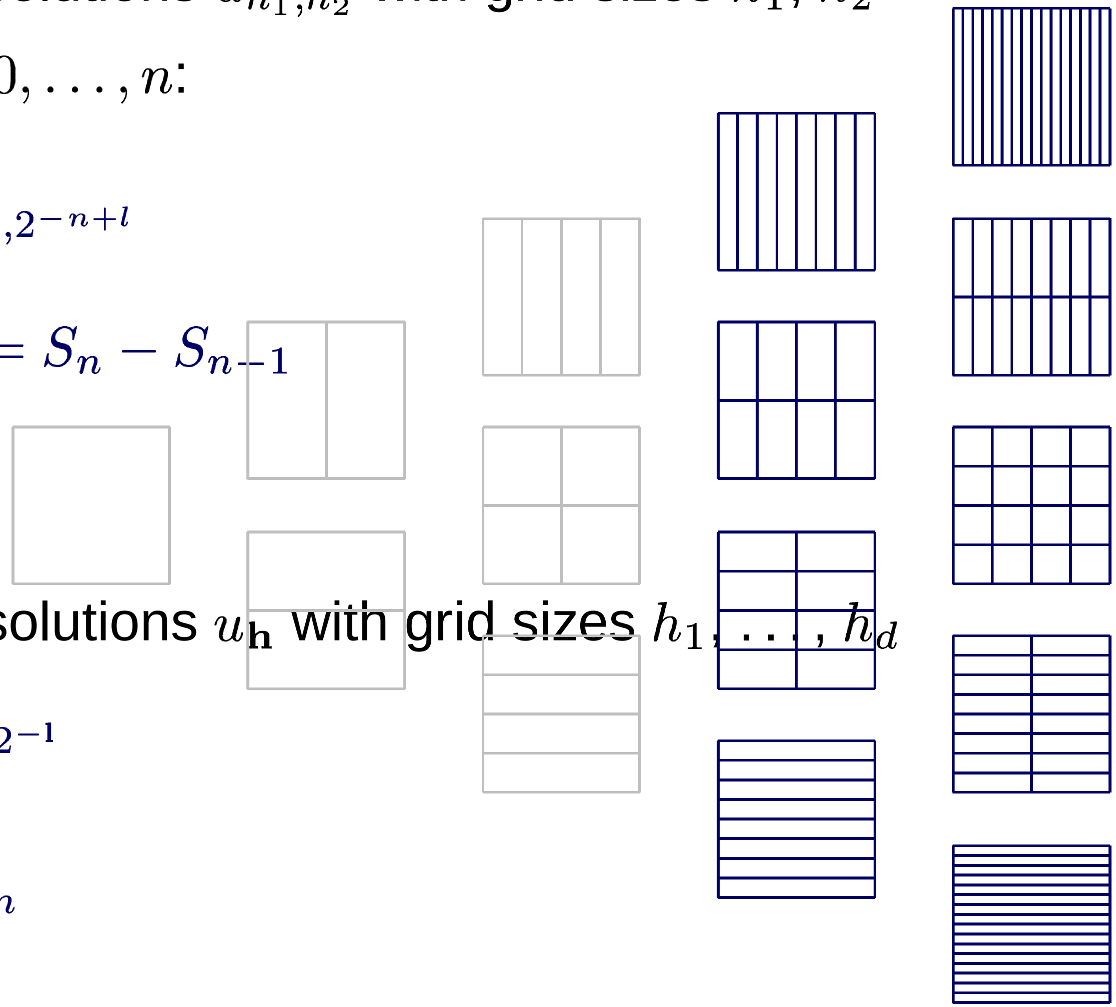
- given numerical solutions u_{h_1,h_2} with grid sizes h_1, h_2
- columnwise, $l = 0, \dots, n$:
- $$S_n := \sum_{l=0}^{n} u_{2^{-l},2^{-n+l}}$$
  $$u_n := \Delta_n S_n := S_n - S_{n-1}$$



Combination technique

2 dimensions:

- given numerical solutions u_{h_1,h_2} with grid sizes h_1, h_2
- columnwise, $l = 0, \ldots, n$:
- $$S_n := \sum_{l=0}^{n} u_{2^{-l},2^{-n+l}}$$ $$u_n := \Delta_n S_n := S_n - S_{n-1}$$



d dimensions:

- given numerical solutions $u_{\mathbf{h}}$ with grid sizes $h_1, \ldots, h_d$
- $$S_n := \sum_{|\mathbf{l}|_1 = n} u_{2^{-\mathbf{l}}}$$ $$u_n := \Delta_n^{d-1} S_n$$

Convergence

Assume for all solutions $u_{\mathbf{h}}$ a pointwise error estimate

$$u - u_{\mathbf{h}} = \sum_{m=1}^{d} \sum_{j_1,\dots,j_m} \gamma_{j_1,\dots,j_m}(h_{j_1},\dots,h_{j_m}) h_{j_1}^2 \cdot \ldots \cdot h_{j_m}^2,$$

where $|\gamma_{j_1,\dots,j_m}| \le K \quad \forall 1 \le m \le d \, \forall \{j_1,\dots,j_m\} \subset \{1,\dots,d\}$.

Convergence

Assume for all solutions $u_{\mathbf{h}}$ a pointwise error estimate

$$u - u_{\mathbf{h}} = \sum_{m=1}^{d} \sum_{j_1,\ldots,j_m} \gamma_{j_1,\ldots,j_m}(h_{j_1},\ldots,h_{j_m}) h_{j_1}^2 \cdot \ldots \cdot h_{j_m}^2,$$

where $|\gamma_{j_1,\ldots,j_m}| \leq K \quad \forall 1 \leq m \leq d \, \forall \{j_1,\ldots,j_m\} \subset \{1,\ldots,d\}$.

Then the combined solution satisfies (pointwise)

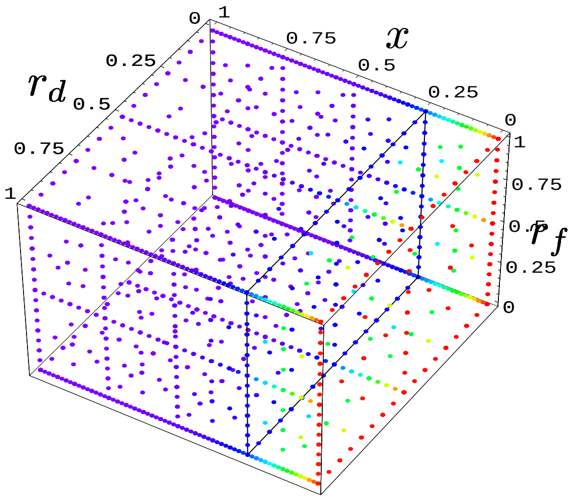
$$|u - u_n| \leq \frac{K}{(d-1)!} \left(\frac{5}{2}\right)^d (n+d-1)^{d-1} 2^{-2n}.$$

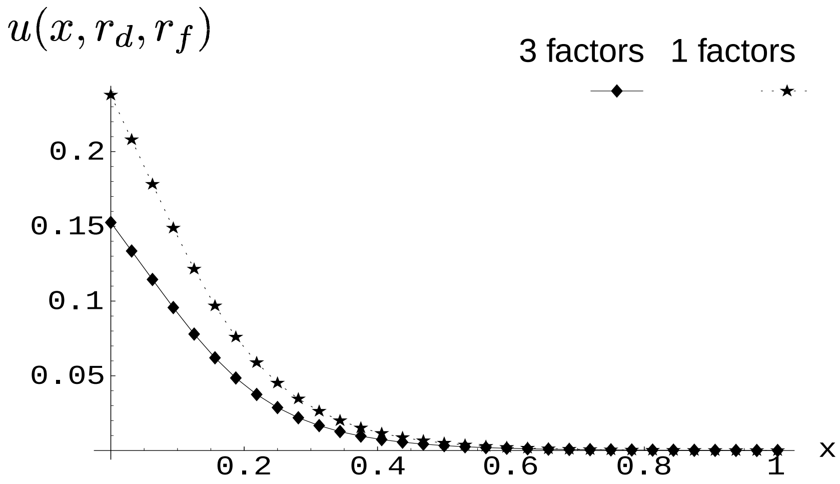
Griebel, Schneider, Zenger: *A Combination Technique for the Solution of Sparse Grid Problems*,1991. $(d = 2,\, d = 3)$

Reisinger, *PhD Thesis*, 2004. $(d \geq 2)$

Example: FX option, $d = 3$

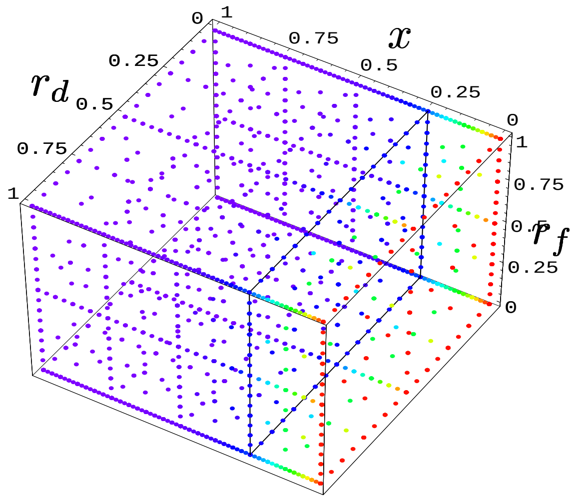
- Solution on a sparse grid and difference to one-factor model.

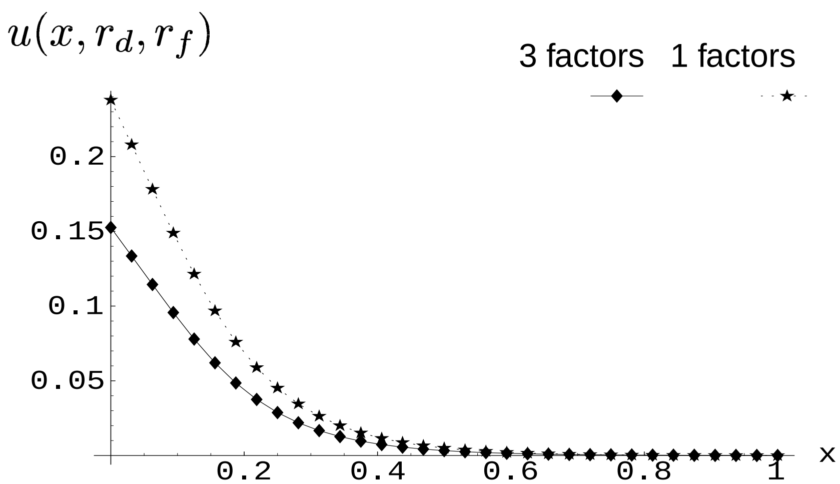




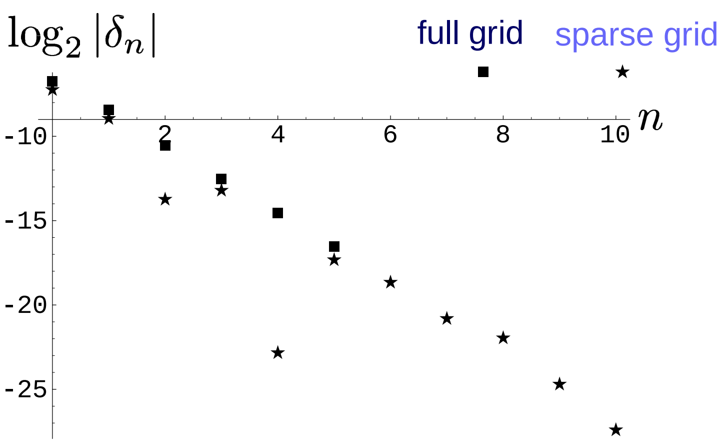
Example: FX option, $d = 3$

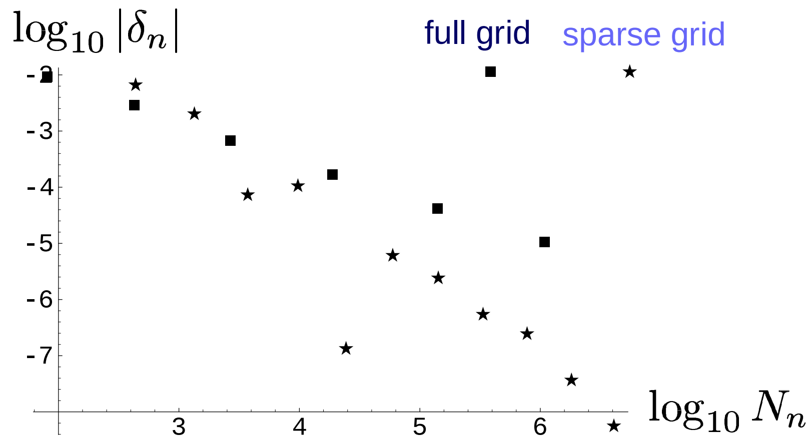
- Solution on a sparse grid and difference to one-factor model.





- Error δ_n of full and sparse grid on level n with N_n unknowns.





Example: Bermudan swaption, $d = 5$

$$
\begin{aligned}
V_{BSw}(T_i, \ldots, T_d) &= \max(V_{BSw}(T_{i+1}, \ldots, T_d; T_i), V_{Sw}(T_i, \ldots, T_d; T_i)) \\
V_{BSw}(T_d; T_d) &= 0
\end{aligned}
$$

- $T_0 = 0$ (29. 07. 04), $T_1 = 1a, \ldots, T_5 = 5a$ (29. 07. 09)
- forward LIBOR rates $2.423, 3.281, 3.931, 4.365, 4.680, 4.933$
- volatilities (%) $0, 24.73, 22.45, 19.36, 17.43, 16.15$

Example: Bermudan swaption, $d = 5$

$$
\begin{aligned}
V_{BSw}(T_i, \dots, T_d) &= \max(V_{BSw}(T_{i+1}, \dots, T_d; T_i), V_{Sw}(T_i, \dots, T_d; T_i)) \\
V_{BSw}(T_d; T_d) &= 0
\end{aligned}
$$

- $T_0 = 0$ (29. 07. 04), $T_1 = 1a, \dots, T_5 = 5a$ (29. 07. 09)
- forward LIBOR rates $2.423, 3.281, 3.931, 4.365, 4.680, 4.933$
- volatilities (%) $0, 24.73, 22.45, 19.36, 17.43, 16.15$

Level	Full grid solution	Sparse grid solution	Total full grid points	Total sparse grid points
3	70.4627	130.641	6561	15149
4	62.3506	50.5246	83521	107427
5		56.8943	1185921	273013
6		59.5418	17850625	678793
7		57.7917	276922881	1658035
8		58.7684	4.36×10^9	3990775

[Blackham, 2004]

Example: basket option, $d = 5$

5 components, $T = 1a$, $r = 0.05$

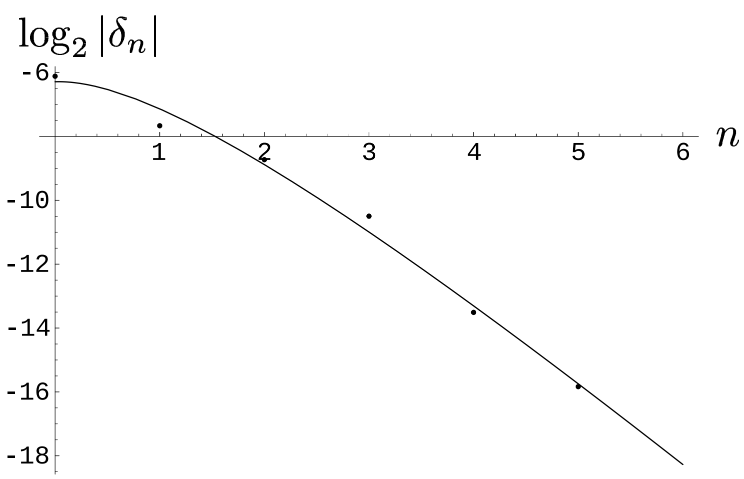
Equity	i	μ_i	σ_i	ρ_{ij}, $1 \leq j \leq 5$				
Deutsche Bank	1	38.1	0.518	1.00	0.79	0.82	0.91	0.84
Hypo-Vereinsbank	2	6.5	0.648	0.79	1.00	0.73	0.80	0.76
Commerzbank	3	5.7	0.623	0.82	0.73	1.00	0.77	0.72
Allianz	4	27.0	0.570	0.91	0.80	0.77	1.00	0.90
Münchner Rück	5	22.7	0.530	0.84	0.76	0.72	0.90	1.00

Example: basket option, $d = 5$

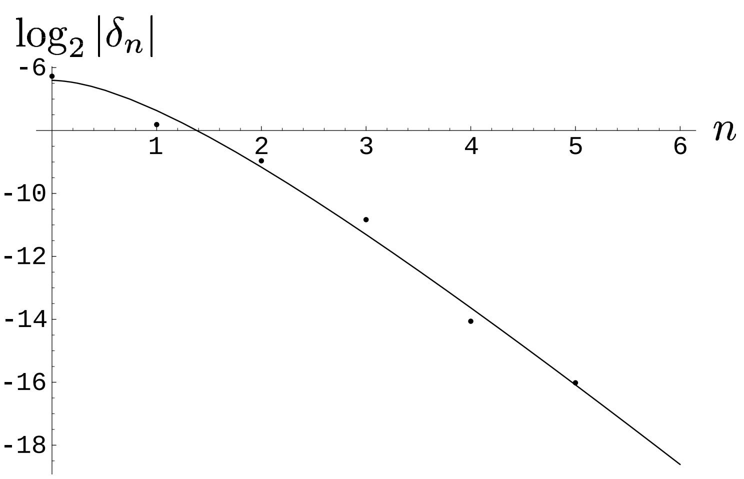
5 components, $T = 1a$, $r = 0.05$

Equity	i	μ_i	σ_i	ρ_{ij}, $1 \le j \le 5$				
Deutsche Bank	1	38.1	0.518	1.00	0.79	0.82	0.91	0.84
Hypo-Vereinsbank	2	6.5	0.648	0.79	1.00	0.73	0.80	0.76
Commerzbank	3	5.7	0.623	0.82	0.73	1.00	0.77	0.72
Allianz	4	27.0	0.570	0.91	0.80	0.77	1.00	0.90
Münchner Rück	5	22.7	0.530	0.84	0.76	0.72	0.90	1.00

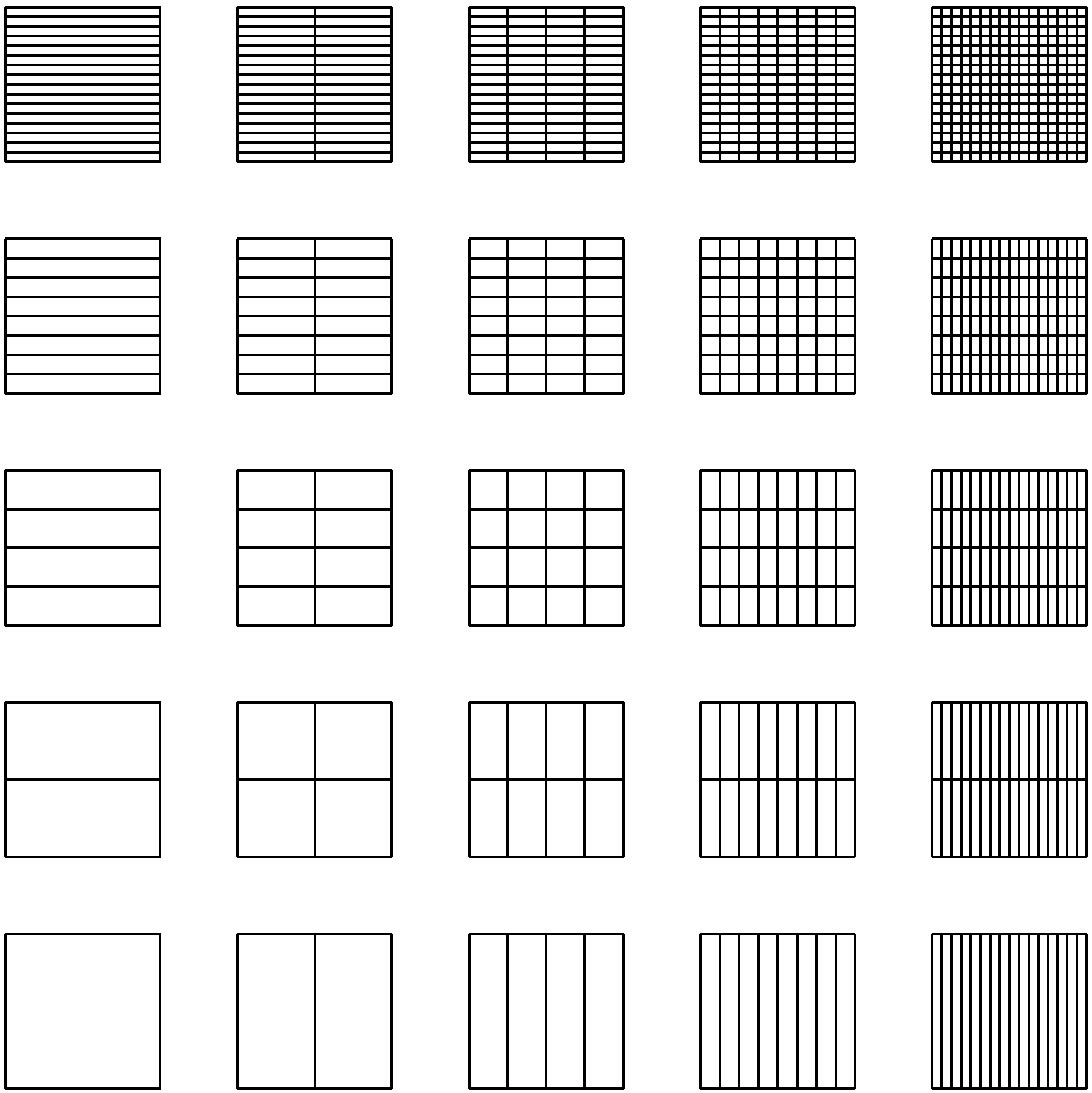
European option, order ~ 2.09



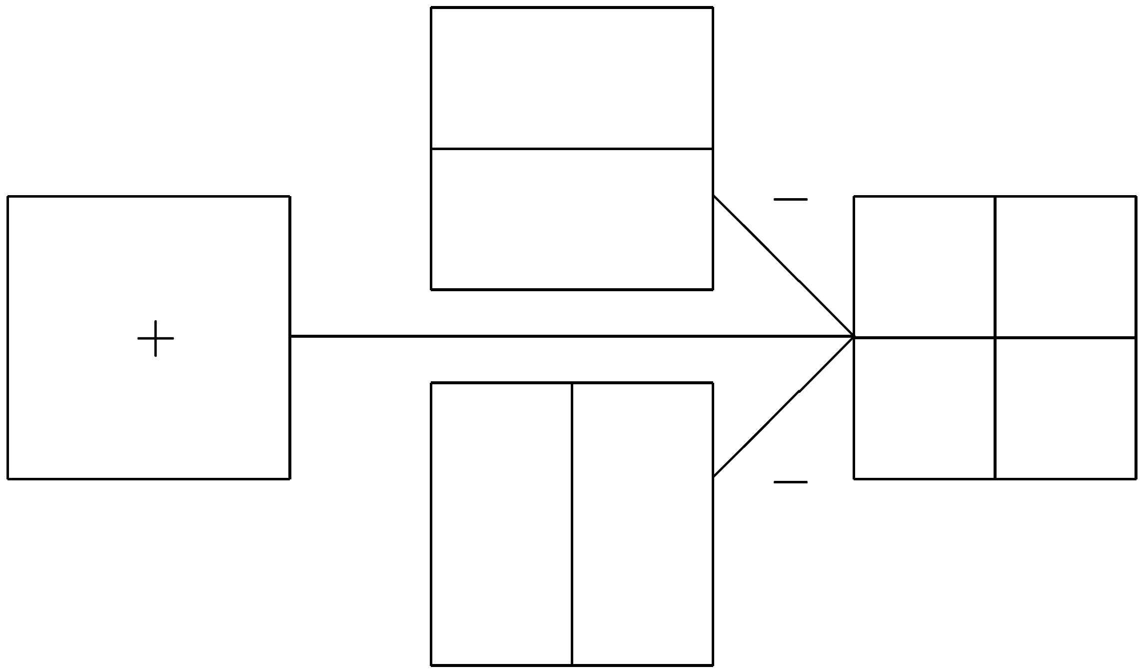
American option, order ~ 2.07



Grid of grids



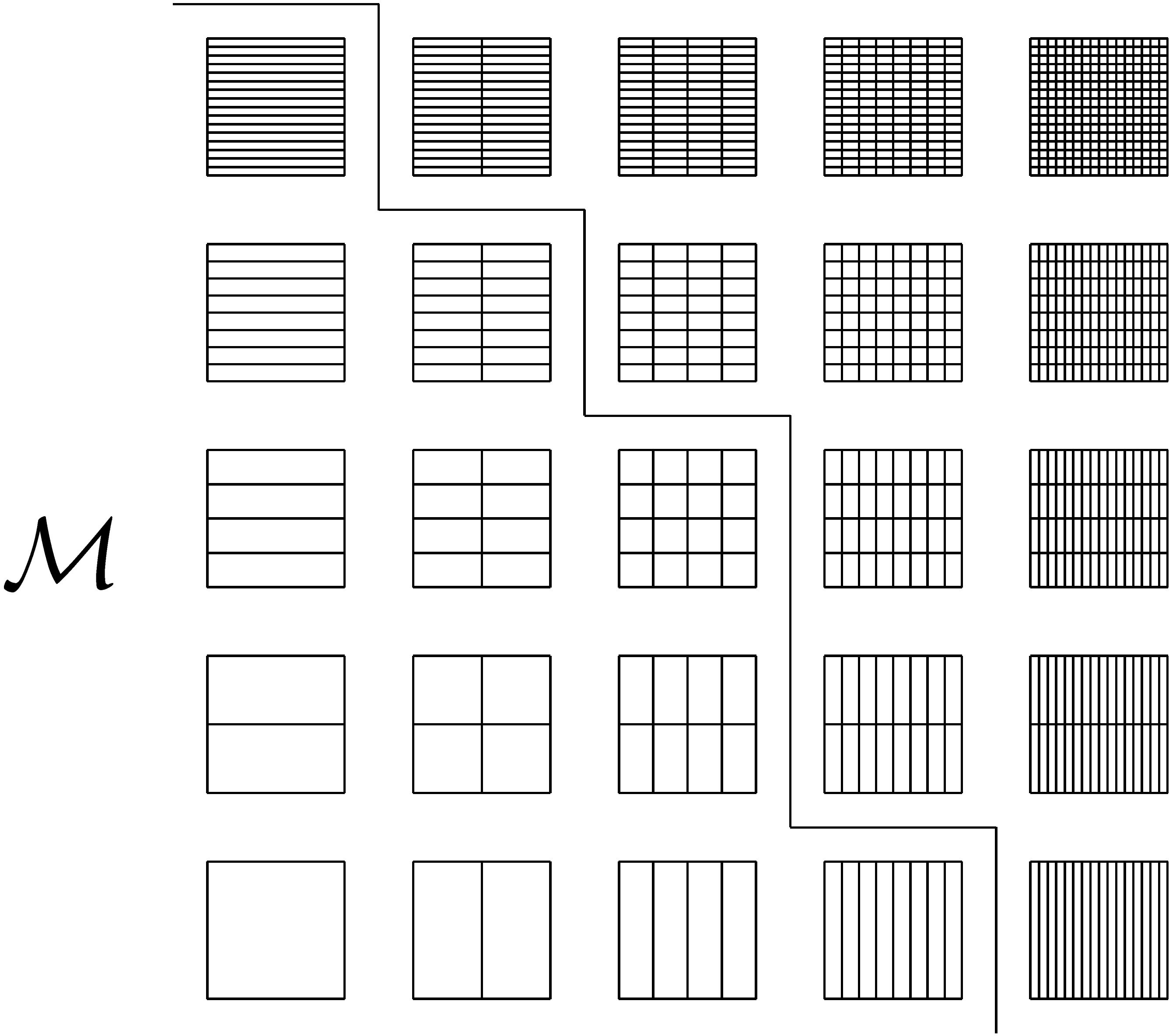
Hierarchical surplus



$$\Delta u_{i,j} = u_{i,j} - (u_{i-1,j} + u_{i,j-1}) + u_{i-1,j-1}$$
$$= \Delta_1 \Delta_2 u_{i,j}$$

$\Delta_1 u_{i,j} = u_{i,j} - u_{i-1,j}$ etc.

Grid of grids (2)



Hierarchical representation

For

$$u_n \quad = \sum_{(i,j)\in\mathcal{M}_n\subset\mathbb{N}_0^2} \Delta u_{i,j}$$

and a suitable discretisation, we expect

$$u_n \to u \quad \text{for} \quad \mathcal{M}_n \uparrow \mathbb{N}_0^2$$

and

$$|u - u_n| \leq \sum_{(i,j)\notin\mathcal{M}_n} |\Delta u_{i,j}|.$$

Hierarchical representation

For

$$u_n \quad = \sum_{(i,j)\in\mathcal{M}_n\subset\mathbb{N}_0^2} \Delta u_{i,j}$$

and a suitable discretisation, we expect

$$u_n \to u \quad \text{for} \quad \mathcal{M}_n \uparrow \mathbb{N}_0^2$$

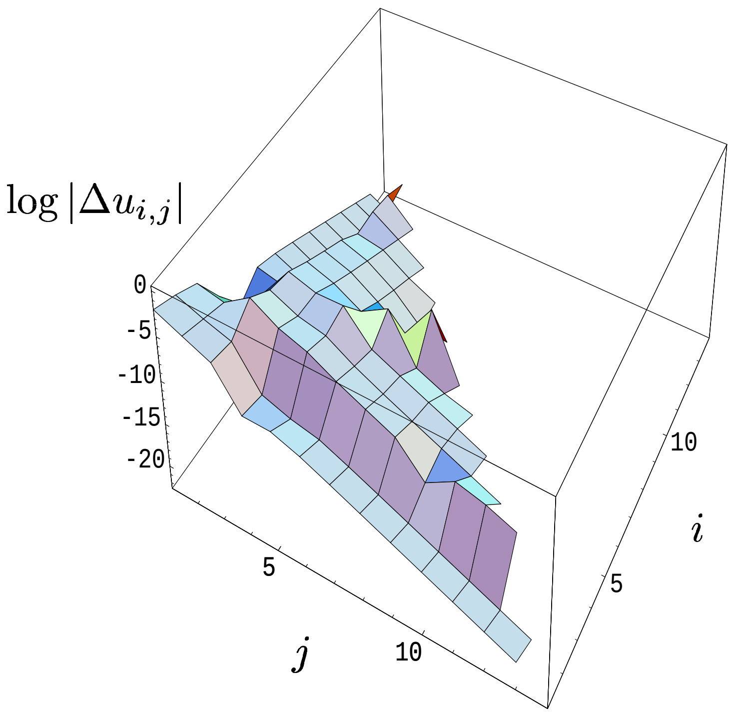
and

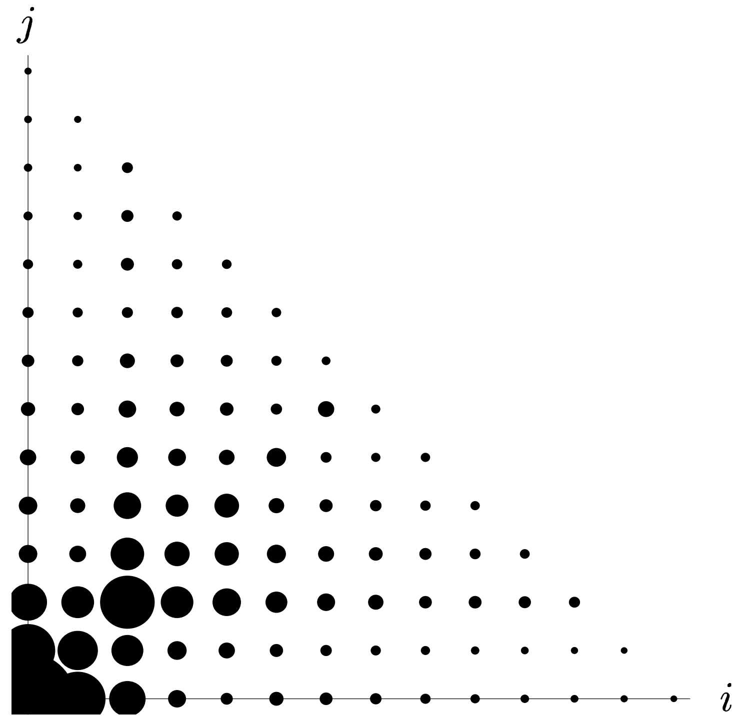
$$|u - u_n| \leq \sum_{(i,j)\notin\mathcal{M}_n} |\Delta u_{i,j}|.$$

- When and how fast will u_n converge?
- Optimal strategy for choice of $\mathcal{M}_n$?

Example: Black-Scholes 2d

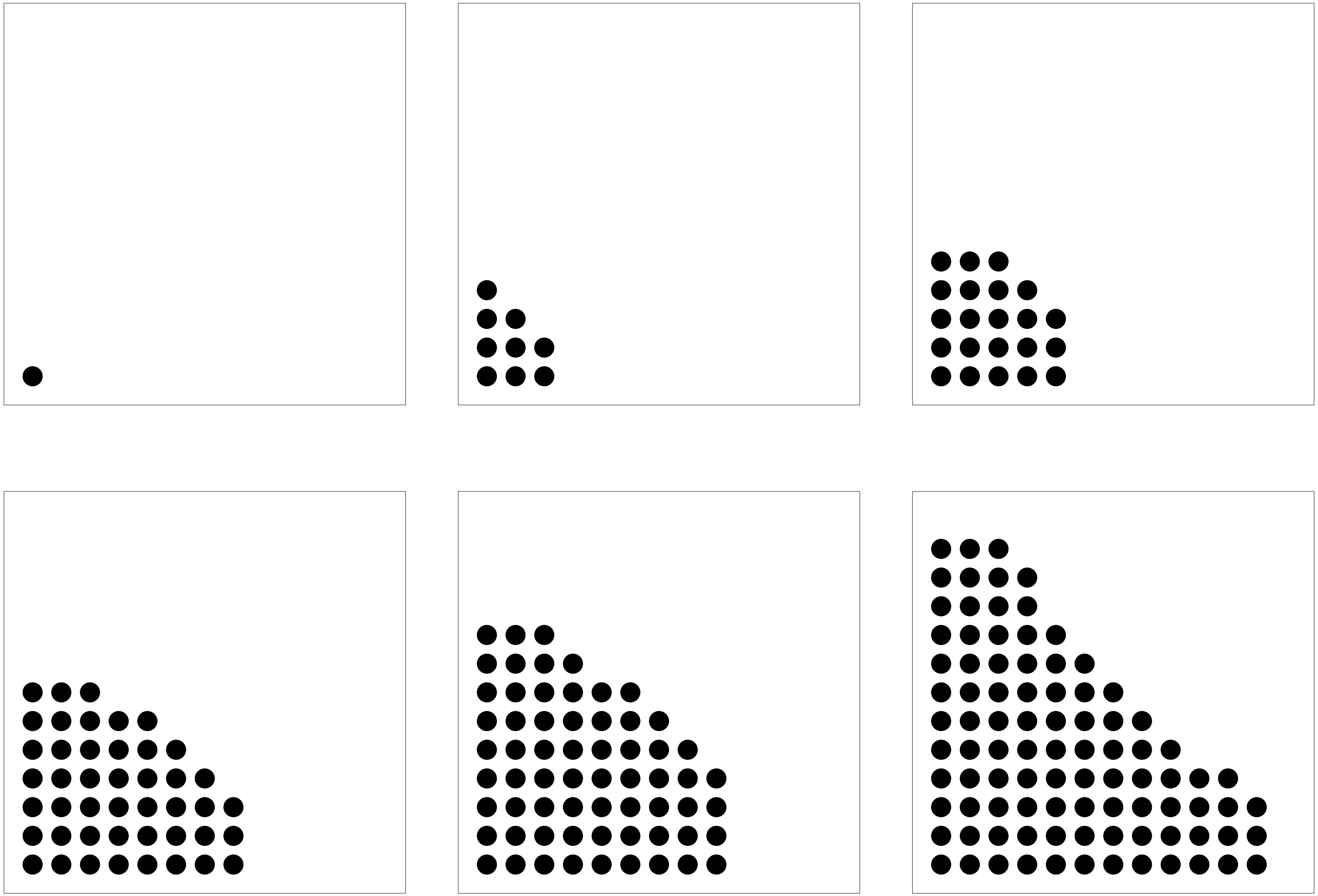
2 equity basket, BMW and Daimler, $\rho = 0.89$, $T = 1$



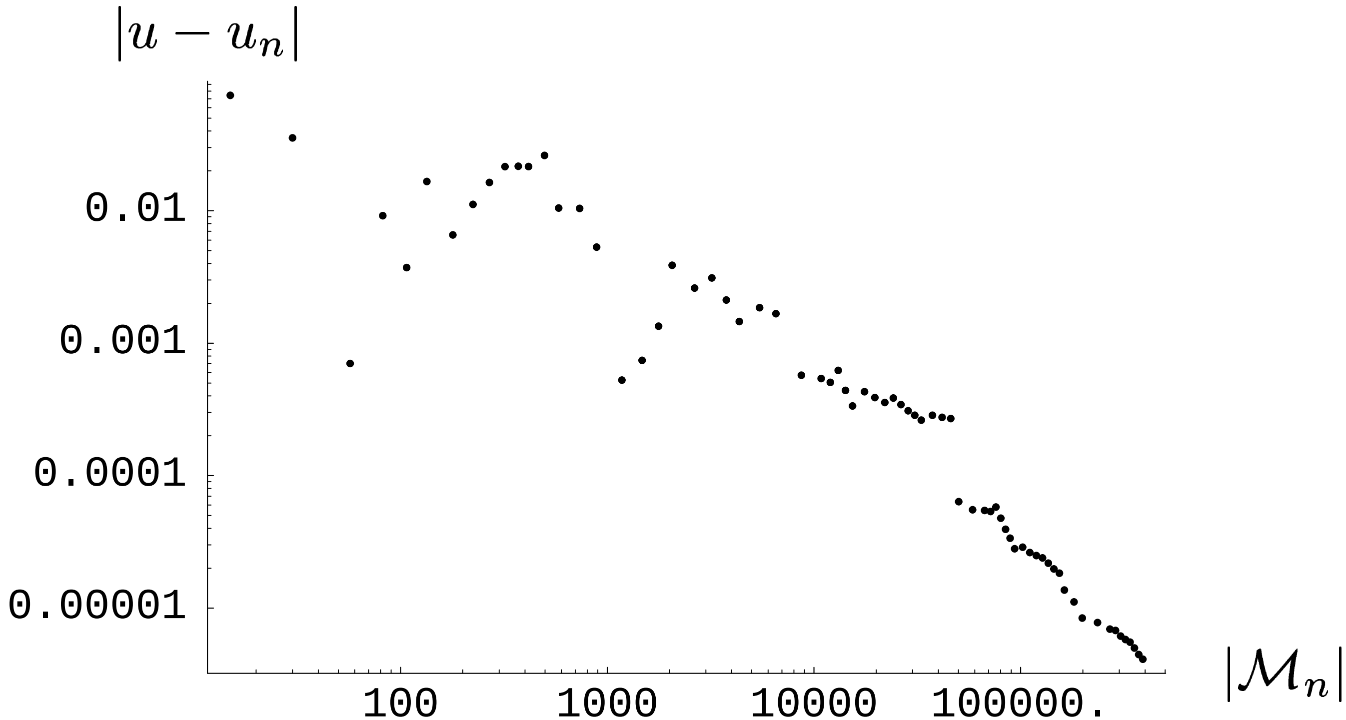


radius $\sim |\log|\Delta u_{i,j}||^{-1}$

Adaptive grid choice

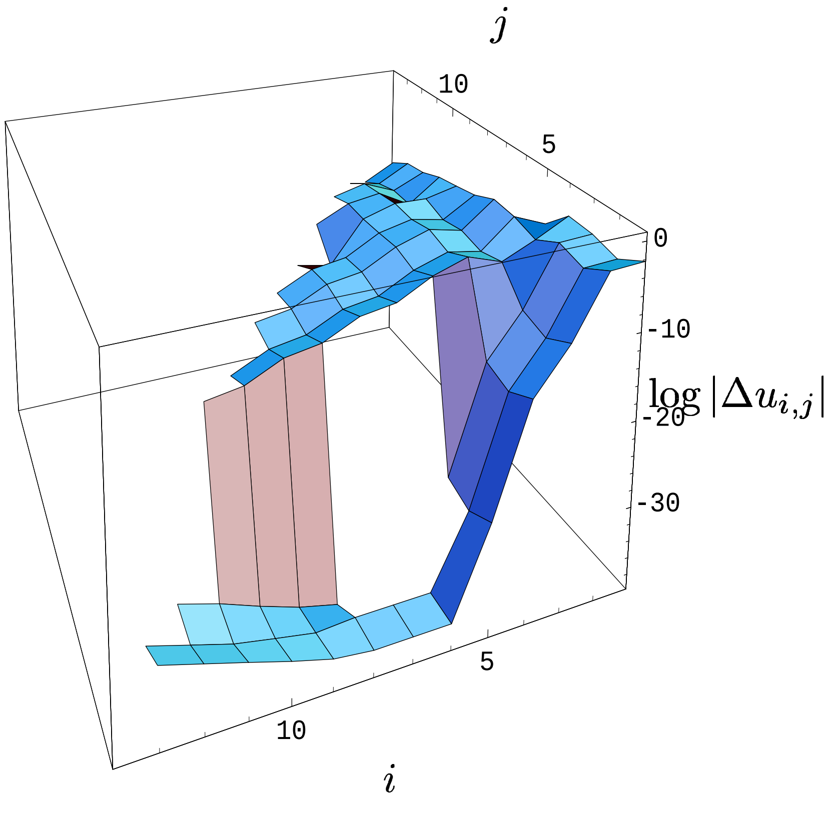


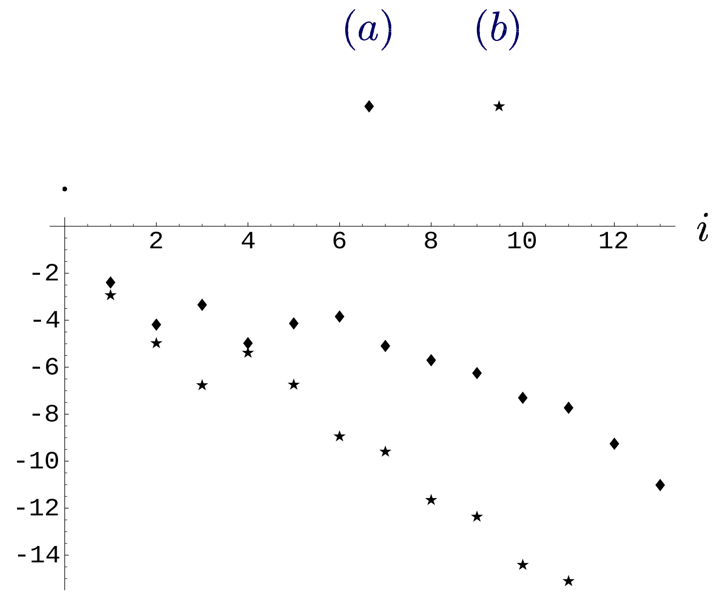
Error behaviour



Example: Black-Scholes 2d, PCA

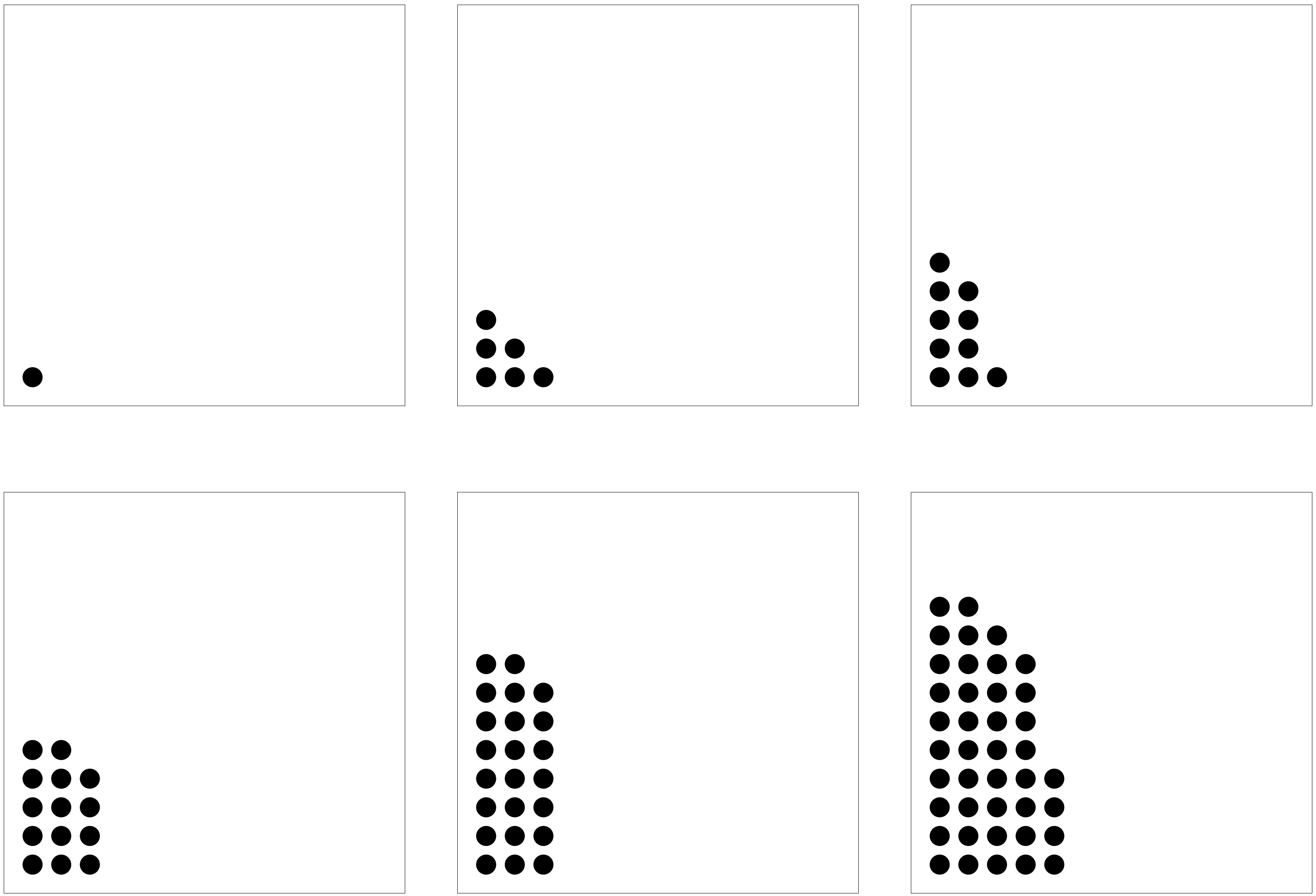
Same system expressed in eigenvectors, $\lambda_1 = 0.431$, $\lambda_2 = 0.024$





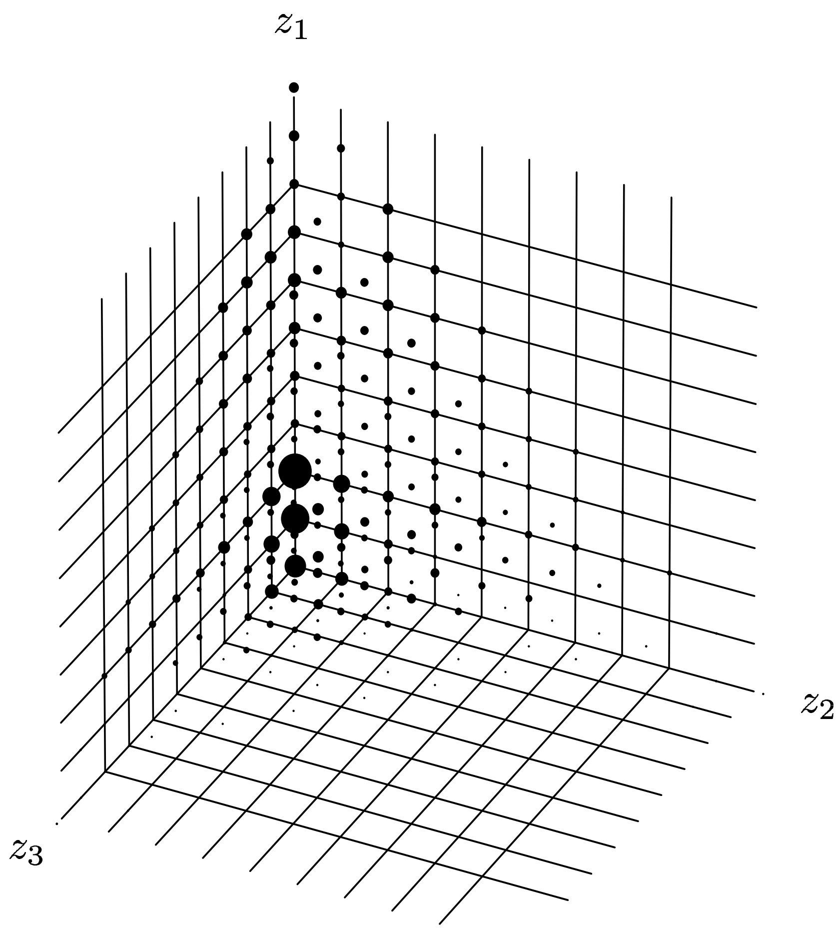
(a) $\log|\Delta u_{i,3}|$ (b) $\log|\Delta u_{2,i}|$

Adaptive grid choice



Example: Black-Scholes 3d, PCA

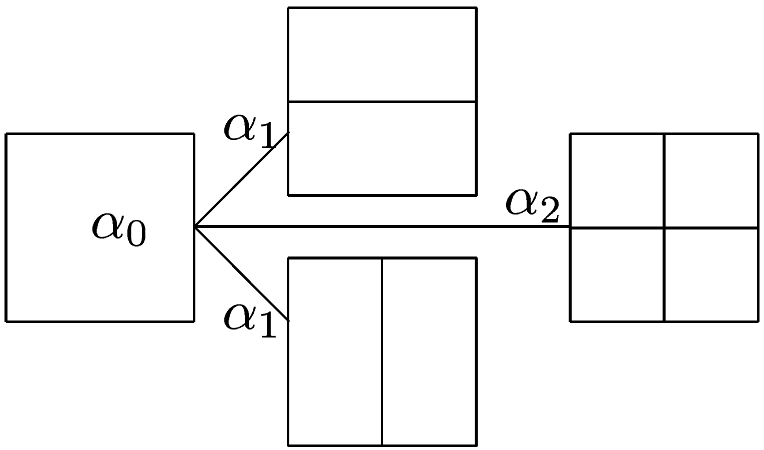
add VW, $\lambda_1 = 0.653$, $\lambda_2 = 0.069$, $\lambda_3 = 0.023$



- ‘points’ at boundary: 469
- ‘points’ in interior: 1728
- $\sum_{\mathbf{i}\in\partial\mathcal{M}} |\Delta u_{\mathbf{i}}| = 0.4369$
- $\sum_{\mathbf{i}\in\mathring{\mathcal{M}}} |\Delta u_{\mathbf{i}}| = 0.0035$

Multivariate extrapolation

Error expansion: $u_{h_1,h_2} = u + c_1 h_1^2 + c_2 h_2^2 + c_{12} h_1^2 h_2^2 + \mathcal{O}(h_1^4 + h_2^4)$



$$\hat{u}_{h_1,h_2} = \alpha_0 u_{h_1,h_2} + \alpha_1 \left(u_{h_1/2,h_2} + u_{h_1,h_2/2}\right) + \alpha_2 u_{h_1/2,h_2/2}$$

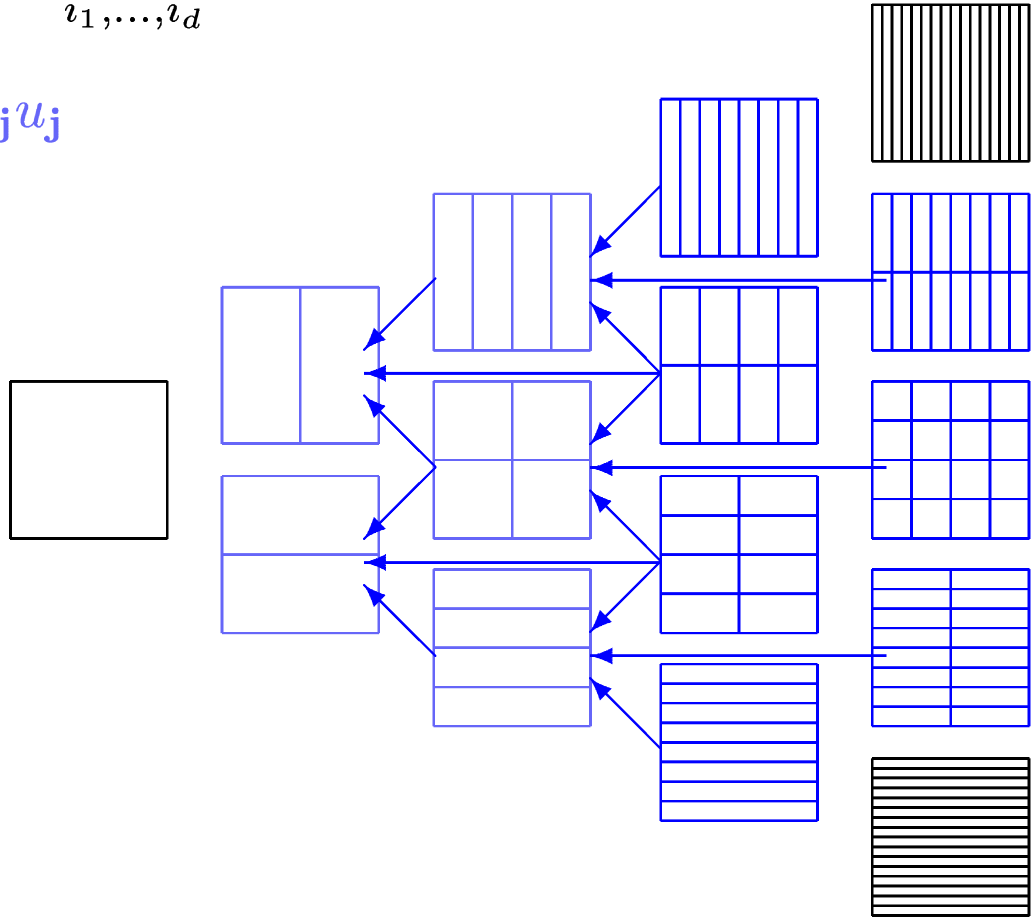
- Richardson: $\alpha_0 = -\frac{1}{3}$, $\alpha_1 = 0$, $\alpha_2 = \frac{4}{3}$, $\qquad \mathcal{O}(h_1^2 h_2^2 + h_1^4 + h_2^4)$
- Schüller, Lin: $\alpha_0 = -\frac{5}{3}$, $\alpha_1 = \frac{4}{3}$, $\alpha_2 = 0$ $\qquad \mathcal{O}(h_1^2 h_2^2 + h_1^4 + h_2^4)$
- CR: $\alpha_0 = \frac{1}{9}$, $\alpha_1 = \frac{-4}{9}$, $\alpha_2 = \frac{16}{9}$, $\qquad \mathcal{O}(h_1^4 + h_2^4)$

Higher order, Greeks

$$u_{\mathbf{h}} = u + \sum_{i=1}^{d} \beta_i(\mathbf{h} \backslash \{h_i\}) h_i^2 + \sum_{i_1,\ldots,i_d} \gamma_{i_1,\ldots,i_d}(h_{i_1},\ldots,h_{i_m}) h_{i_1}^4 \cdot \ldots \cdot h_{i_m}^4$$

$$u_n = \sum_{l=n-2d+1}^{n} \sum_{|\mathbf{j}|=l} a_{\mathbf{j}} u_{\mathbf{j}}$$

$$|u - u_n| \leq c(d) n^{d-1} 2^{-4n}$$

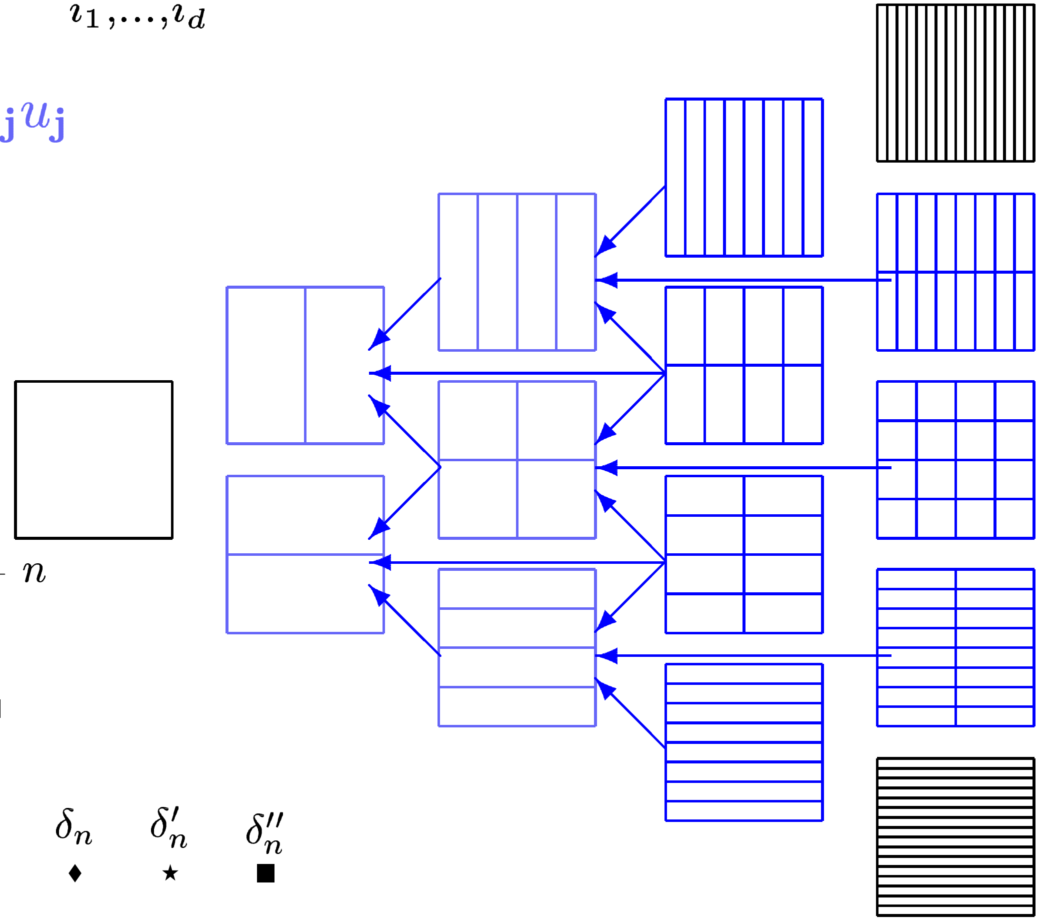


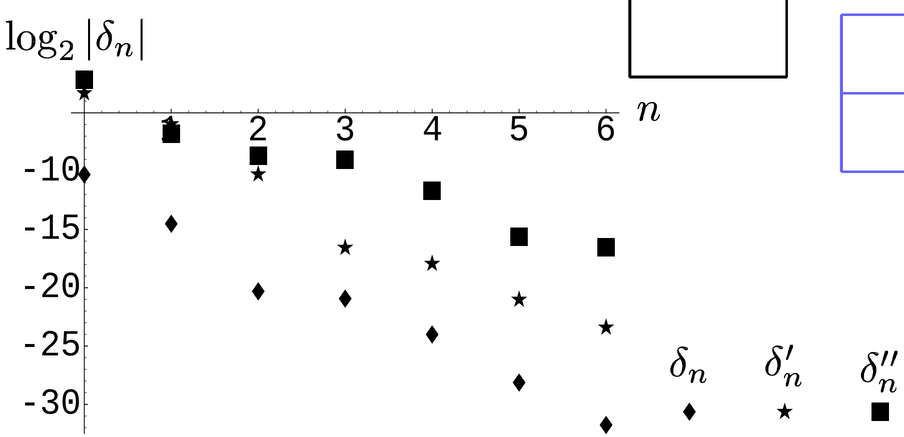
Higher order, Greeks

$$u_{\mathbf{h}} = u + \sum_{i=1}^{d} \beta_i(\mathbf{h}\backslash\{h_i\}) h_i^2 + \sum_{i_1,\ldots,i_d} \gamma_{i_1,\ldots,i_d}(h_{i_1},\ldots,h_{i_m}) h_{i_1}^4 \cdot \ldots \cdot h_{i_m}^4$$

$$u_n = \sum_{l=n-2d+1}^{n} \sum_{|\mathbf{j}|=l} a_{\mathbf{j}} u_{\mathbf{j}}$$

$$|u - u_n| \leq c(d) n^{d-1} 2^{-4n}$$





Discretisation error of FX option price (δ_n), its Δ and Γ

Expansions

Sparse grids

Multigrid

Implementation

Iterative solution - preconditioning

- discretisation (FD, FE, FV) $\rightarrow$ large, ill-conditioned systems
- multilevel algorithms provide asymptotically mesh size independent convergence rates
- simple variant: cascadic multigrid uses coarse grid solution as initial guess on next level

Iterative solution - preconditioning

- discretisation (FD, FE, FV) $\rightarrow$ large, ill-conditioned systems
- multilevel algorithms provide asymptotically mesh size independent convergence rates
- simple variant: cascadic multigrid uses coarse grid solution as initial guess on next level

eg 2D Black-Scholes:

- level l
- N_l unknowns
- initial residual $||\mathbf{r}_0^l||$
- final residual $||\mathbf{r}^l||$
- n_l iterations

l	N_l	$\|\|\mathbf{r}_0^l\|\|$	$\|\|\mathbf{r}^l\|\|$	n_l
3	81	1.94 10^{1}	2.67 10^{-4}	3
4	289	7.98 10^{0}	1.33 10^{-3}	3
5	1089	3.31 10^{0}	1.18 10^{-3}	3
6	4225	1.30 10^{0}	1.04 10^{-3}	3
7	16641	4.94 10^{-1}	6.52 10^{-3}	2
8	66049	1.93 10^{-1}	2.03 10^{-3}	2
9	263169	9.15 10^{-2}	5.94 10^{-4}	2

R., Wittum: *On Multigrid for Anisotropic Equations and Variational Inequalities*. 2004.

Iterative solution - robustness

- degeneracy of the equation at boundaries
- strongly anisotropic grids

eg 3D Black-Scholes:

- levels l_1, l_2, l_3
- $2^{l_i} + 1$ points
- N unknowns
- CPU time T

l_1	l_2	l_3	N	T	T/N
0	0	15	131076	106.88	0.000815405
0	3	12	73746	29.71	0.000402869
0	6	9	66690	21.46	0.000321787
1	2	12	61455	27.89	0.000453828
1	5	9	50787	21.7	0.000427275
2	2	11	51225	25.54	0.000498585
2	5	8	42405	20.12	0.000474472
3	4	8	39321	19.91	0.000506345
4	4	7	37281	33.61	0.000901532
5	5	5	35937	17.58	0.000489189

Expansions

Sparse grids

Multigrid

Implementation

Parallelisation

5D basket: Level n, M_n unknowns, max unknowns/grid m_n, grids ν_n

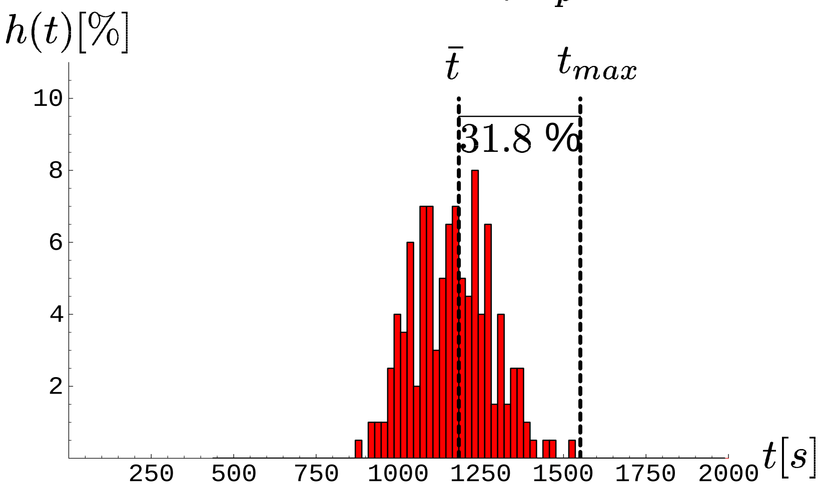
n	M_n	m_n	ν_n	n	M_n	m_n	ν_n	n	N_n	m_n
1	32	32	1	6	42363	528	126	11	6042330	16400
2	240	48	5	7	122125	1040	210	12	15185610	32784
3	1120	80	15	8	337755	2064	330	13	37600980	65552
4	4200	144	35	9	904745	4112	495	14	91913985	131088
5	13890	272	70	10	2362620	8208	715	15	222166875	262160

Parallelisation

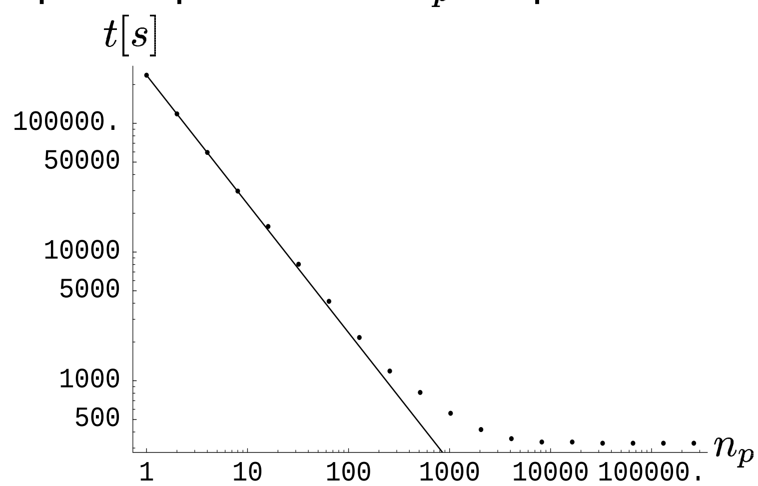
5D basket: Level n, M_n unknowns, max unknowns/grid m_n, grids ν_n

n	M_n	m_n	ν_n	n	M_n	m_n	ν_n	n	N_n	m_n
1	32	32	1	6	42363	528	126	11	6042330	16400
2	240	48	5	7	122125	1040	210	12	15185610	32784
3	1120	80	15	8	337755	2064	330	13	37600980	65552
4	4200	144	35	9	904745	4112	495	14	91913985	131088
5	13890	272	70	10	2362620	8208	715	15	222166875	262160

Distribution of CPU-time, $n_p = 200$



Speed-up for n_p processors



Conclusions

From an algorithmic point of view, the presented framework

- automatically detects and exploits lower dimensional structures
- chooses asymptotically optimal discrete approximation spaces
- solves the discrete systems in linear complexity
- is inherently parallel

Practically relevant features include

- easy estimation of the Greeks
- comparable efficiency for American and Bermudan contracts
- extensible to calibration and more general models