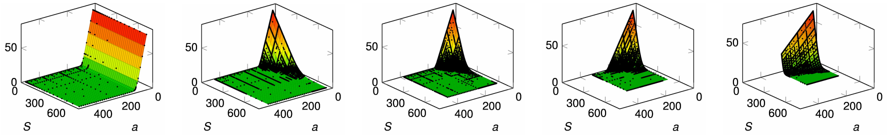
PRMIA Munich: Workshop on Variable Annuities

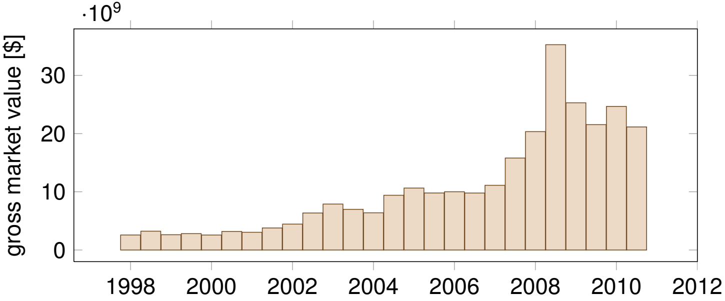
A High-Dimensional PDE-Based Approach for Capital Market Guarantees

Stefanie Schraufstetter

3. April 2012

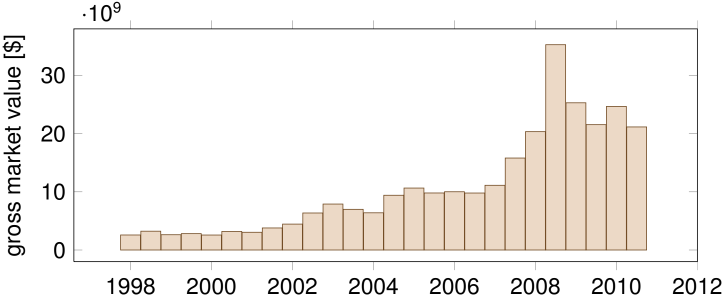


Motivation: Development of the Financial Market



total volume of all OTC derivates in the G10 countries and Switzerland

Motivation: Development of the Financial Market

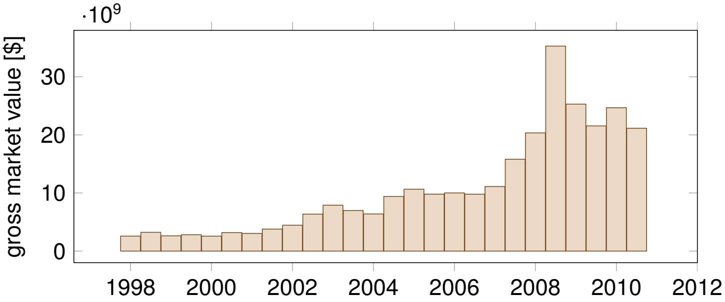


total volume of all OTC derivates in the G10 countries and Switzerland

problems:

- increasing variety
- increasing complexity

Motivation: Development of the Financial Market



total volume of all OTC derivates in the G10 countries and Switzerland

problems:

- increasing variety
- increasing complexity

→ need for a unique software for evaluation of financial derivatives

Challenge „Pricing Framework“

Basic idea: model driven architecture

- unique product specification
- automated evaluation

Challenge „Pricing Framework“

Basic idea: model driven architecture

- unique product specification
- automated evaluation

Modeling Language *ThetaML*:

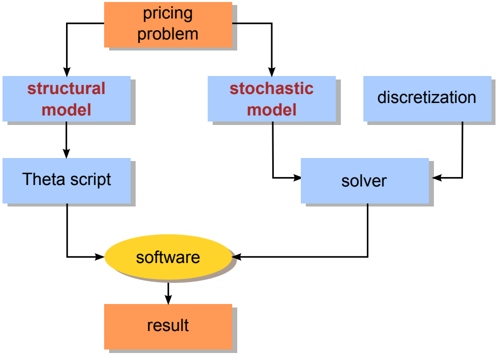
- virtual timing model
- mainly 3 basic effects:
  - waiting
  - transacting
  - deciding
- independent of pricing algorithm

The Pricing Framework

```
model asianOption
  import S "underlying"
  import K "strike"
  export V "option price"

  a=0
  loop j=1:n
    Theta 1/n
    a = a+S
  end
  V=max(a/n-K,0)
end
```

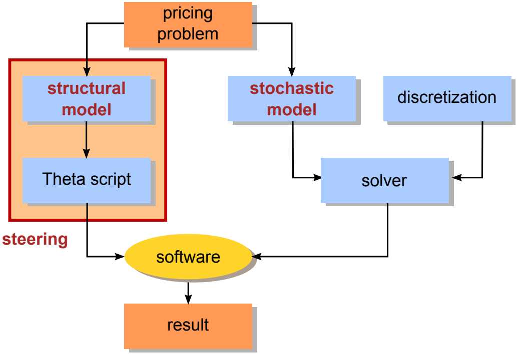
The Pricing Framework

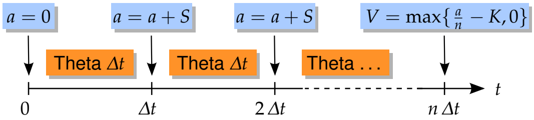
```
model asianOption
   import S "underlying"
   import K "strike"
   export V "option price"

   a=0
   loop j=1:n
      Theta 1/n
      a = a+S
   end
   V=max(a/n-K,0)
end
```



The Pricing Framework

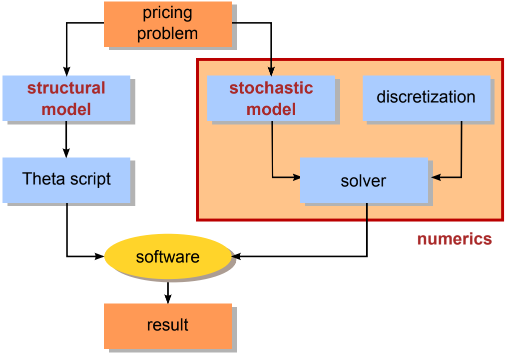
```
model asianOption
   import S "underlying"
   import K "strike"
   export V "option price"

   a=0
   loop j=1:n
      Theta 1/n
      a = a+S
   end
   V=max(a/n-K,0)
end
```

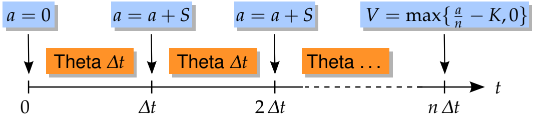




The Pricing Framework

```
model asianOption
   import S "underlying"
   import K "strike"
   export V "option price"

   a=0
   loop j=1:n
      Theta 1/n
      a = a+S
   end
   V=max(a/n-K,0)
end
```





MC vs. PDE – State of the Art

MC:

- independent of dimension ("nearly")
- (theoretical) slow speed of convergence: $\mathcal{O}(N^{\frac{1}{2}})$
- different accelerating techniques, e.g.
  - quasi-MC method
  - multi-level MC
- estimation of conditional expectation difficult ($\rightarrow$ least-squares MC!)

PDE:

- (theoretical) high speed of convergence
- strongly dimension-dependent: $\mathcal{O}(N^d)$ dofs!
- typically complex algorithms
- well-suited for hedging
- topic of current research

MC vs. PDE – State of the Art

MC:

- independent of dimension (“nearly”)
- (theoretical) slow speed of convergence: $\mathcal{O}(N^{\frac{1}{2}})$
- different accelerating techniques, e.g.
  - quasi-MC method
  - multi-level MC
- estimation of conditional expectation difficult ($\rightarrow$ least-squares MC!)

THETARIS
Engineering in Finance

PDE:

- (theoretical) high speed of convergence
- strongly dimension-dependent: $\mathcal{O}(N^d)$ dofs!
- typically complex algorithms
- well-suited for hedging
- topic of current research

MC vs. PDE – State of the Art

MC:

- independent of dimension (“nearly”)
- (theoretical) slow speed of convergence: $\mathcal{O}(N^{\frac{1}{2}})$
- different accelerating techniques, e.g.
  - quasi-MC method
  - multi-level MC
- estimation of conditional expectation difficult ($\rightarrow$ least-squares MC!)

THETARIS
Engineering in Finance

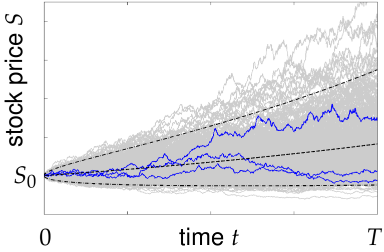
PDE:

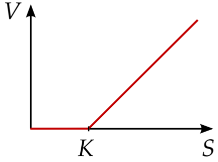
- (theoretical) high speed of convergence
- strongly dimension-dependent: $\mathcal{O}(N^d)$ dofs!
- typically complex algorithms
- well-suited for hedging
- topic of current research

Option Pricing with PDEs

underlying: stochastic process, e.g. geometric Brownian motion

$$dS_t = \mu S_t \, dt + \sigma S_t \, dW_t, \quad W_t \sim \mathcal{N}(0, t)$$



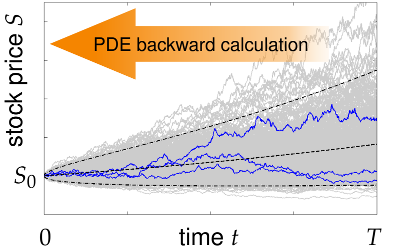


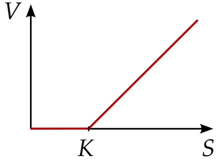
option price at
end time $t = T$

Option Pricing with PDEs

underlying: stochastic process, e.g. geometric Brownian motion

$$dS_t = \mu S_t \, dt + \sigma S_t \, dW_t, \quad W_t \sim \mathcal{N}(0, t)$$





option price at end time $t = T$

From the Stock Price to the Option Value

1. Assume an underlying following a geometric Brownian motion:
   $dS_t = \mu S\, dt + \sigma S\, dW_t$
2. Apply Itô's lemma
3. Eliminate stochastic term by considering a risk-free portfolio
   $\Pi = V - \Delta \cdot S \qquad \Rightarrow d\Pi = r\Pi\, dt$

From the Stock Price to the Option Value

1. Assume an underlying following a geometric Brownian motion:
   $dS_t = \mu S\, dt + \sigma S\, dW_t$
2. Apply Itô's lemma
3. Eliminate stochastic term by considering a risk-free portfolio
   $\Pi = V - \Delta \cdot S \qquad \Rightarrow d\Pi = r\Pi\, dt$

Itô's lemma

Let S_t follow the SDE

$$dS_t = a(t, S_t)\, dt + b(t, S_t)\, dW_t$$

with a Wiener process W_t and $V = f(t, S_t)$ be twice differentiable. Then, the stochastic process for V is given by

$$dV = \left(\frac{\partial V}{\partial t} + a(t, S_t) \frac{\partial V}{\partial S} + (b(t, S_t))^2 \frac{\partial^2 V}{\partial S^2} \right) dt + b(t, S_t) \frac{\partial V}{\partial S}\, dW_t.$$

From the Stock Price to the Option Value

1. Assume an underlying following a geometric Brownian motion:
   $dS_t = \mu S\,dt + \sigma S\,dW_t$
2. Apply Itô's lemma
3. Eliminate stochastic term by considering a risk-free portfolio
   $\Pi = V - \Delta \cdot S \qquad \Rightarrow d\Pi = r\Pi\,dt$

From the Stock Price to the Option Value

1. Assume an underlying following a geometric Brownian motion:
   $dS_t = \mu S\,dt + \sigma S\,dW_t$
2. Apply Itô's lemma
3. Eliminate stochastic term by considering a risk-free portfolio
   $\Pi = V - \Delta \cdot S \qquad \Rightarrow d\Pi = r\Pi\,dt$

Black-Scholes equation (1D)

$$\frac{\partial V(t,S)}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V(t,S)}{\partial S^2} + rS\frac{\partial V(t,S)}{\partial S} - rV(t,S) = 0$$

Notation:

V	option price	σ	volatility
S	stock price	r	risk-free rate

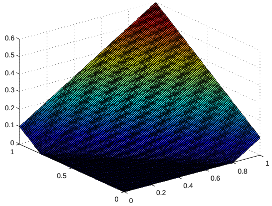
The Multi-Dimensional Black-Scholes Equation

Black-Scholes Equation

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sum_{i,j=1}^{d} \sigma_i \sigma_j \rho_{ij} S_i S_j \frac{\partial^2 V}{\partial S_i \partial S_j} + \sum_{i=1}^{d} (r - \delta_i) S_i \frac{\partial V}{\partial S_i} - rV = 0$$

final condition, e.g. for a call option:

$$V(\mathbf{S}, T) = \max\{\frac{1}{d}\sum_{i=1}^{d} S_i - K, 0\}$$



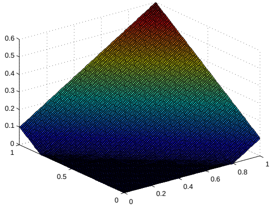
The Multi-Dimensional Black-Scholes Equation

Black-Scholes Equation

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sum_{i,j=1}^{d} \sigma_i \sigma_j \rho_{ij} S_i S_j \frac{\partial^2 V}{\partial S_i \partial S_j} + \sum_{i=1}^{d} (r - \delta_i) S_i \frac{\partial V}{\partial S_i} - rV = 0$$

final condition, e.g. for a call option:

$$V(\mathbf{S}, T) = \max\{\frac{1}{d}\sum_{i=1}^{d} S_i - K, 0\}$$



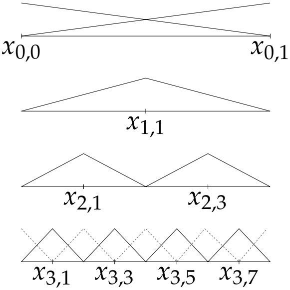
boundary condition:

$$V(\bar{\mathbf{S}}, t) = e^{-r(T-t)} V(\bar{\mathbf{S}}, T) \quad \forall \bar{\mathbf{S}} \in \partial\Omega$$

with $\Omega := [S_{1,\text{min}}, S_{1,\text{max}}] \times \ldots \times [S_{d,\text{min}}, S_{d,\text{max}}]$

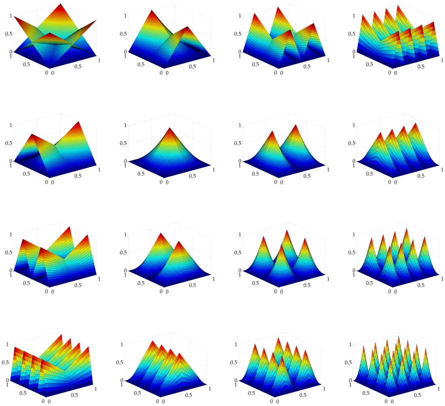
Discretization with Sparse Grids

hierarchical basis 1D



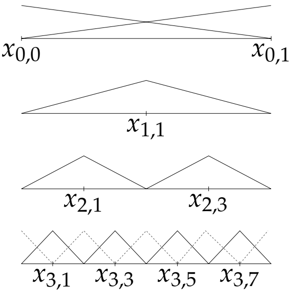
$\varphi_{l,i}(x)$: **function of level** l
with index i

hierarchical basis 2D



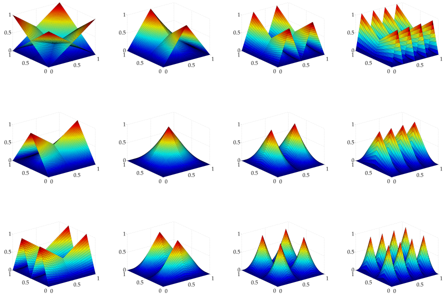
Discretization with Sparse Grids

hierarchical basis 1D



$\varphi_{l,i}(x)$: **function of level l with index i**

hierarchical basis 2D

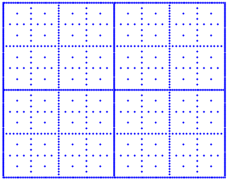


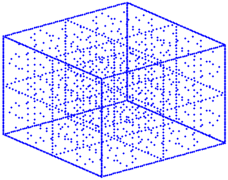
defined via tensor product:

$$\varphi_{\mathbf{l},\mathbf{i}}(\mathbf{x}) = \prod_k \varphi_{l_k,i_k}(x_k)$$

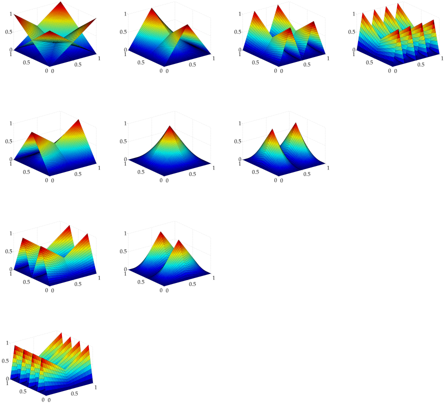
Discretization with Sparse Grids

sparse grid 2D/3D





hierarchical basis 2D

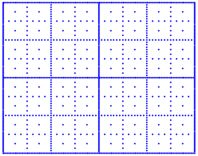


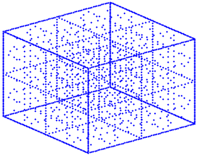
Discretization with Sparse Grids

Sparse grids vs. full grids:

	full grid	sparse grid
number of points	$\mathcal{O}(n^d)$	$\mathcal{O}(n(\log n)^{d-1})$
approximation accuracy	$\mathcal{O}(h^2)$	$\mathcal{O}(h^2(\log \frac{1}{h})^{d-1})$

with mesh width h and $n = 1/h$





Solving PDEs with Sparse Grids

‘standard’ approach:

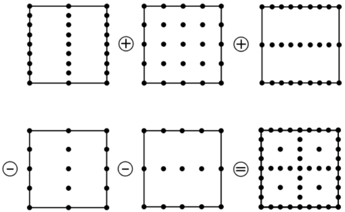
combination technique

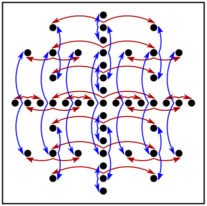
- extrapolation of solutions on smaller full grids
- simple to implement
- no spatial adaptivity

our approach:

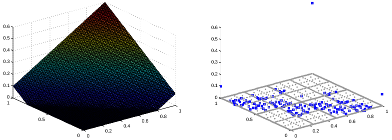
direct sparse grid approach

- works directly on the hierarchical basis
- complex algorithms
- spatial adaptivity

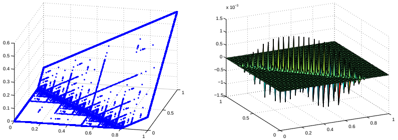




Why Spatial Adaptivity?



hierarchical coefficients



discretization error $\approx 10^{-4}$

- higher efficiency:
  only few surplusses
  contribute significantly

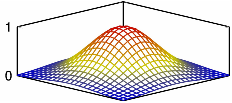
- higher accuracy:
  full-grid like refinement
  necessary to reduce
  discretization error
  at kinks, jumps, etc.

Results for a European Basket Option

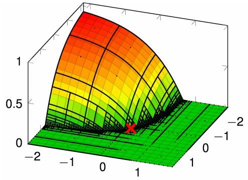
Comparison of degrees of freedom: non-adaptive vs. adaptive

dim.	accuracy (L^∞)	non-adaptive	adaptive		adaptive, weighted	
2	0.0002	98 305	13 402	(13.6%)	9 202	(9.4%)
3	0.0050	274 431	43 589	(15.9%)	28 488	(10.4%)
4	$\approx$ 0.0010	–	91 039	–	16 670	–
5	$\approx$ 0.0002	–	–	–	89 704	–

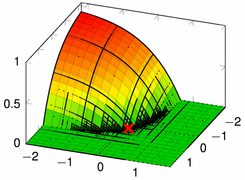
weighted adaptivity:



Gaussian function



non-weighted



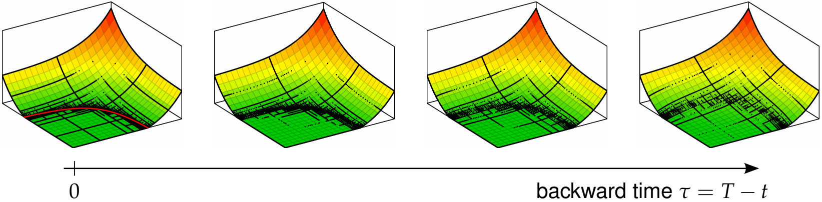
weighted

Results for a European Basket Option

Comparison of degrees of freedom: non-adaptive vs. adaptive

dim.	accuracy (L^∞)	non-adaptive	adaptive		adaptive, weighted	
2	0.0002	98 305	13 402	(13.6%)	9 202	(9.4%)
3	0.0050	274 431	43 589	(15.9%)	28 488	(10.4%)
4	$\approx$ 0.0010	–	91 039	–	16 670	–
5	$\approx$ 0.0002	–	–	–	89 704	–

dynamic realization:



Automated Evaluation Algorithm

1. parsing of the script
   - lexical, syntactic, semantic analysis

   → deriving the operator sequence
2. forward estimation of domain size
3. backward calculation

```
model asianOption
  import S "underlying"
  import K "strike"
  export V "option price"

  a=0
  loop j=1:n
    wait 1/n
    a = a+S
  end
  V=max(a/n-K,0)
end
```

Automated Evaluation Algorithm

1. parsing of the script
   - lexical, syntactic, semantic analysis

   → deriving the operator sequence
2. forward estimation of domain size
3. backward calculation

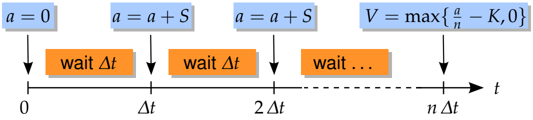
```
model asianOption
  import S "underlying"
  import K "strike"
  export V "option price"

  a=0
  loop j=1:n
    wait 1/n
    a = a+S
  end
  V=max(a/n-K,0)
end
```

```
model     →  MODEL head EOC body END
body      →  operator | operator body
operator  →  IF formula THEN formula
             ELSE formula END |
             WAIT formula |
             LOOP formula EOC body END |
             assignop EOC | ...
formula   →  NUMBER | LITERAL |
             formula PLUS formula | ...
assignop  →  LITERAL ASSIGN formula
...
```

Automated Evaluation Algorithm

1. parsing of the script
   - lexical, syntactic, semantic analysis

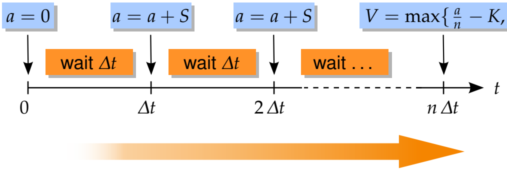
   $\rightarrow$ deriving the operator sequence
2. forward estimation of domain size
3. backward calculation

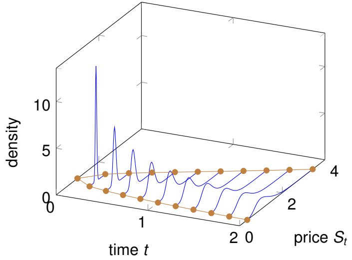


Automated Evaluation Algorithm

1. parsing of the script
   - lexical, syntactic, semantic analysis

   $\rightarrow$ deriving the operator sequence
2. forward estimation of domain size
3. backward calculation

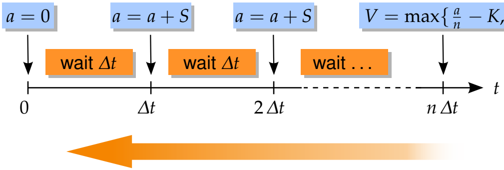


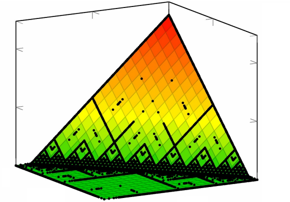


Automated Evaluation Algorithm

1. parsing of the script
   - lexical, syntactic, semantic analysis

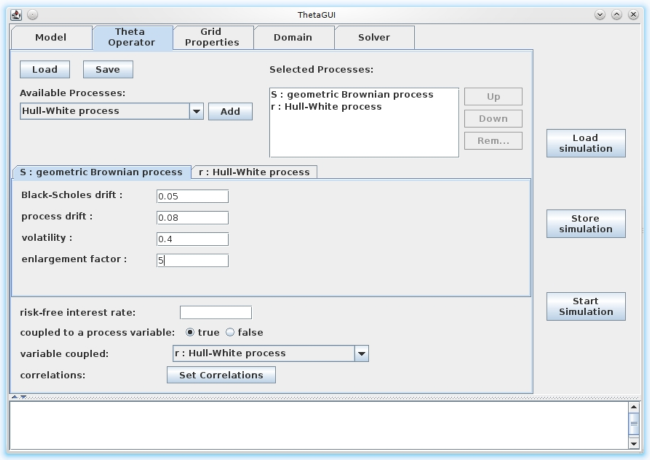
   $\rightarrow$ deriving the operator sequence
2. forward estimation of domain size
3. backward calculation





$$\frac{\partial V}{\partial t} + \frac{1}{2}\sum_{i,j=1}^{d} \sigma_{ij}(\mathbf{S},t)\frac{\partial^2 V}{\partial S_i \partial S_j} + \sum_{i=1}^{d} \mu_i(\mathbf{S},t)\frac{\partial V}{\partial S_i} - r(\mathbf{S},t)V = 0$$

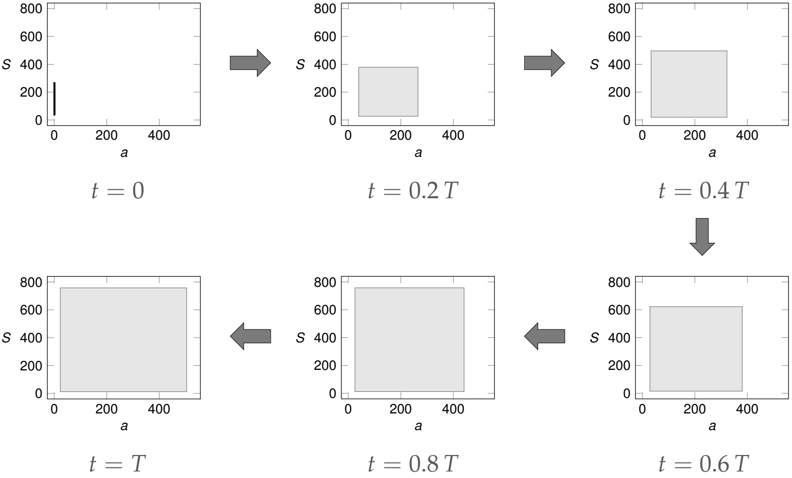
GUI: Model Parametrization



GUI $\Longleftrightarrow$ XML file with parametrization $\Longleftrightarrow$ framework

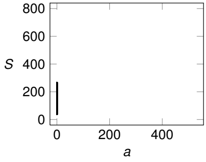
Calculation Cycle of an Asian Option

1. forward estimation of domain size

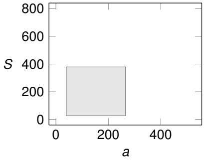


Calculation Cycle of an Asian Option

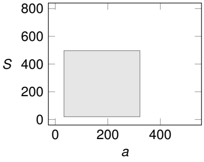
2. PDE backward calculation



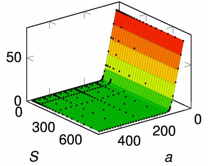
$t = 0$



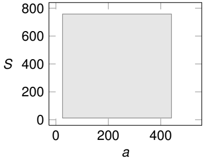
$t = 0.2\,T$



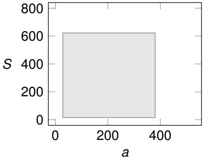
$t = 0.4\,T$



$t = T$



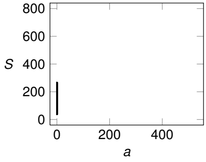
$t = 0.8\,T$



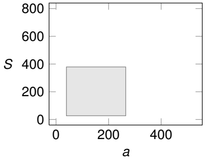
$t = 0.6\,T$

Calculation Cycle of an Asian Option

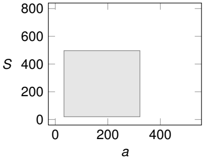
2. PDE backward calculation



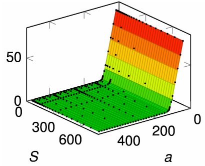
$t = 0$



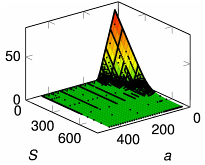
$t = 0.2\,T$



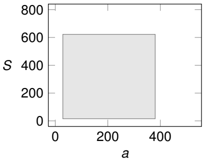
$t = 0.4\,T$



$t = T$



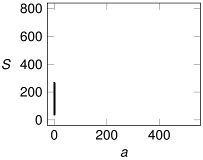
$t = 0.8\,T$



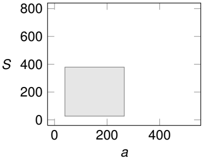
$t = 0.6\,T$

Calculation Cycle of an Asian Option

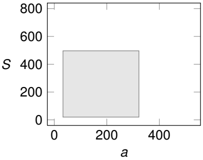
2. PDE backward calculation



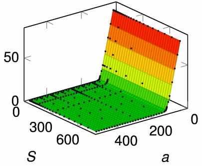
$t = 0$



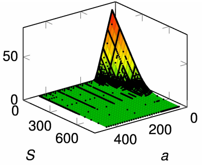
$t = 0.2\,T$



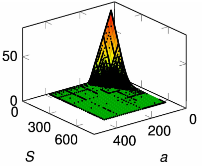
$t = 0.4\,T$



$t = T$



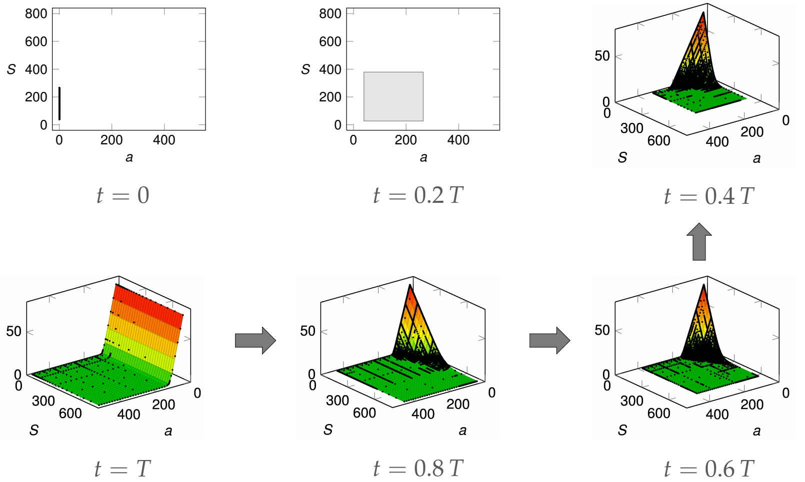
$t = 0.8\,T$



$t = 0.6\,T$

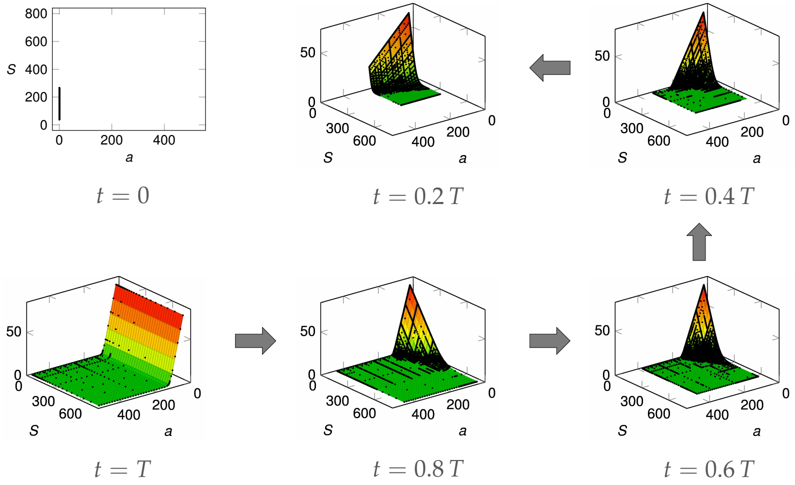
Calculation Cycle of an Asian Option

2. PDE backward calculation



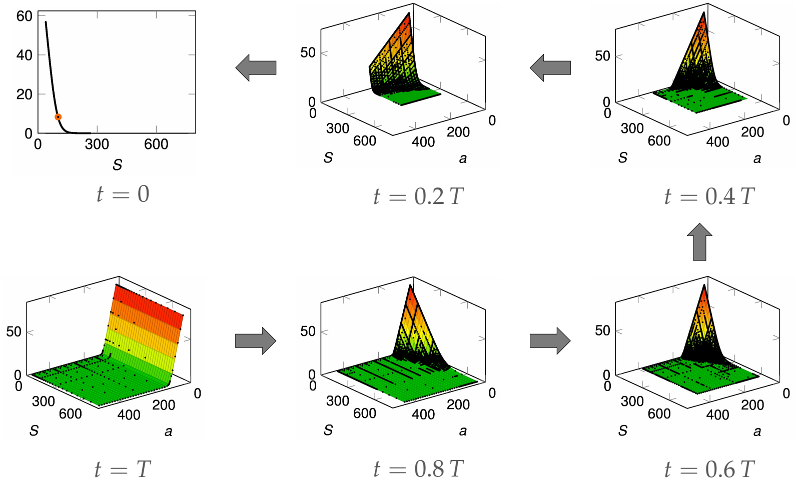
Calculation Cycle of an Asian Option

2. PDE backward calculation

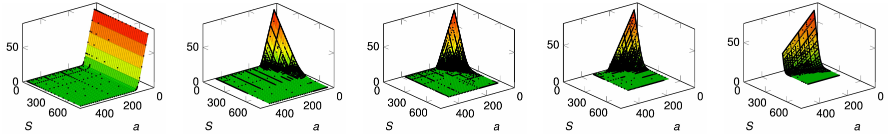


Calculation Cycle of an Asian Option

2. PDE backward calculation

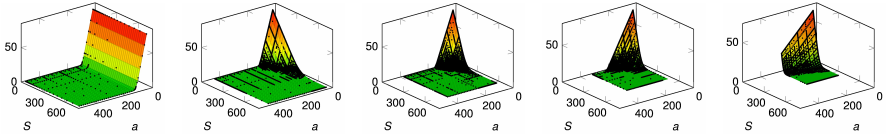


Asian Option – Results



max.	non-adaptive			adaptive		
level	∅ dofs	price	cpu time	∅ dofs	price	cpu time
7	1 281	8.13993	11	1 929	8.23363	19
8	2 727	8.11502	13	2 842	8.19200	28
9	5 357	8.17599	27	3 686	8.18145	37
10	10 703	8.17792	57	4 224	8.18027	43
11	21 543	8.17820	156	4 653	8.17958	50
12	43 758	8.17883	411	5 026	8.17951	55
13	89 284	8.17895	1 119	5 494	8.17956	64
14	181 117	8.18100	3 208	6 126	8.17943	69
15	359 173	8.18088	7 604	6 937	8.17948	84
MC reference value (10^6 paths): 8.18074						

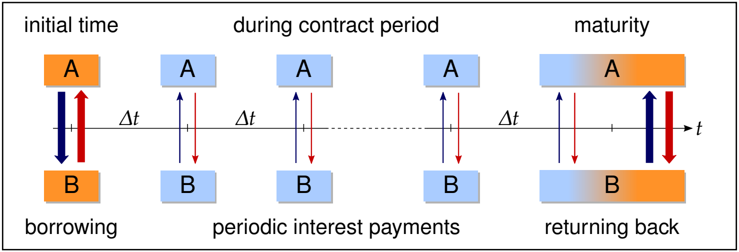
Asian Option – Results



max. level	non-adaptive ∅ dofs	non-adaptive price	non-adaptive cpu time	adaptive ∅ dofs	adaptive price	adaptive cpu time
7	1 281	8.13993	11	1 929	8.23363	19
8	2 727	8.11502	13	2 842	8.19200	28
9	5 357	8.17599	27	3 686	8.18145	37
10	10 703	8.17792	57	4 224	8.18027	43
11	21 543	8.17820	156	4 653	8.17958	50
12	43 758	8.17883	411	5 026	8.17951	55
13	89 284	8.17895	1 119	5 494	8.17956	64
14	181 117	8.18100	3 208	6 126	8.17943	69
15	359 173	8.18088	7 604	6 937	8.17948	84

MC reference value (10^6 paths): 8.18074 $\pm 2 \cdot 10^{-3}$

Cross Currency Swap



variable	model	stochastic process
domestic interest rate r_d	Ho-Lee	$dr_d(t) = (\mu_{r_d} + \sigma_{r_d}^2 t)\, dt + \sigma_{r_d}\, dW_1(t)$
foreign interest rate r_f	Ho-Lee	$dr_f(t) = (\mu_{r_f} + \sigma_{r_f}^2 t)\, dt + \sigma_{r_f}\, dW_2(t)$
FX rate s	FX process	$ds(t) = (r_d(t) - r_f(t)) \cdot s(t)\, dt + \sigma_s \cdot s(t)\, dW_3(t)$

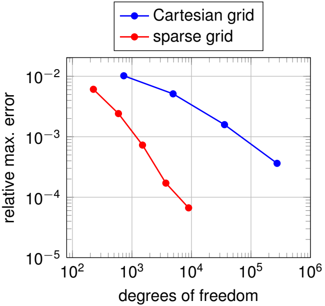
Cross Currency Swap

Steps for Evaluation:

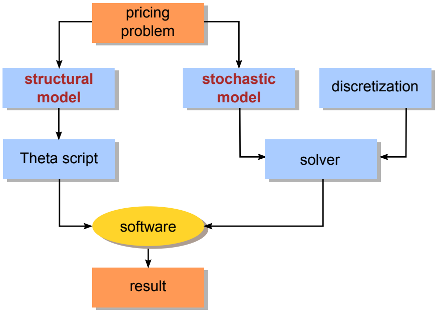
1. modeling the structure in form of a Theta script
2. specification of stochastic processes
3. specification of coupling of interest rate
4. start of automated evaluation

Numerical aspects:

- 3-dimensional problem
- smooth payoff function
- non-adaptive grid sufficient



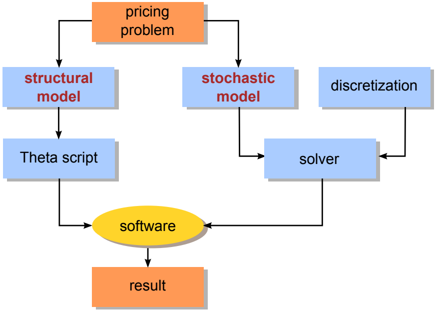
The Pricing Framework at a Glance



Financial products/strategies

- ✓ options with up to 6 underlyings
- ✓ swaps
- ✓ variable annuities
- ✓ hedging

The Pricing Framework at a Glance



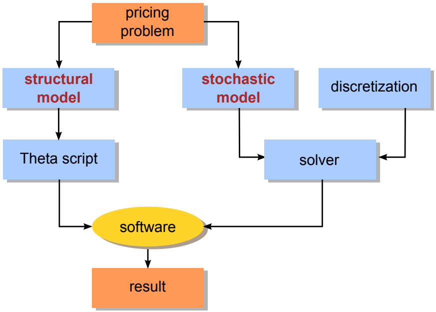
Financial products/strategies

- ✓ options with up to 6 underlyings
- ✓ swaps
- ✓ variable annuities
- ✓ hedging

numeric implementation

- ✓ higher-dimensional
- ✓ adaptive
- ✓ parallel & efficient

The Pricing Framework at a Glance



Financial products/strategies

- ✓ options with up to 6 underlyings
- ✓ swaps
- ✓ variable annuities
- ✓ hedging

numeric implementation

- ✓ higher-dimensional
- ✓ adaptive
- ✓ parallel & efficient

framework

- ✓ unique product specification
- ✓ automated evaluation
- ✓ modular structure

Thank you for your attention!

References

J. Benk, D. Pflüger and S. Schraufstetter
Pricing Multi-Dimensional Financial Derivatives with the Dimensional Adaptive Combination Technique under the Heston Process
Journal of Computational Finance, 2012, submitted.

A. Heinecke, S. Schraufstetter and H. Bungartz
A Highly Parallel Black-Scholes Solver Based on Adaptive Sparse Grids
International Journal of Computer Mathematics, 2012, accepted.

H. Bungartz, A. Heinecke, D. Pflüger and S. Schraufstetter
Parallelizing a Black-Scholes Solver based on Finite Elements and Sparse Grids
Concurrency and Computation: Practice and Experience, doi: 10.1002/cpe.2837, 2012.

H. Bungartz, A. Heinecke, D. Pflüger and S. Schraufstetter
Option Pricing with a Direct Adaptive Sparse Grid Approach
Journal of Computational and Applied Mathematics, 2011.

J. Benk, H. Bungartz, A. Nagy and S. Schraufstetter
Variants of the Combination Technique for Multi-Dimensional Option Pricing
Progress in Industrial Mathematics at ECMI 2010.

H. Bungartz, A. Heinecke, D. Pflüger and S. Schraufstetter
Parallelizing a Black-Scholes Solver based on Finite Elements and Sparse Grids
Proceedings of the IEEE International Parallel & Distributed Processing Symposium, 2010.

S. Schraufstetter and J. Benk
A General Pricing Technique Based on Theta-Calculus and Sparse Grids
Proceedings of the ENUMATH Conference, 2010.