UBS

Credit Portfolio Simulation with MATLAB®

MATLAB® Conference 2015 Switzerland

Dr. Marcus Wunsch
Associate Director

Statistical Risk Aggregation Methodology

Risk Methodology, UBS AG

Disclaimer: The opinions expressed here are purely those of the speaker, and may not be taken to represent the official views of UBS.

June 9, 2015

Key Takeaways

- Credit risk can be captured with the structural Merton-type model
- This model can be implemented using the MC (Monte Carlo) method
- Parallelization led to a remarkable 25x speedup of simulation time
- This was done using the MathWorks Parallel Computing Toolbox

UBS

SRAM and UBS

- About SRAM:
  - Statistical Risk Aggregation Methodology (SRAM) team
  - I am mainly responsible for credit risk
  - We are a team of 9 people (backgrounds in physics, applied math, statistics)
  - SRAM aggregates all risks of UBS for Economic Capital (Basel Pillar 2)
  - We collaborate closely with reporting, IT, and other methodology teams
- About UBS:
  - Swiss global financial services company
  - Serving private, institutional, and corporate clients worldwide
  - Serving retail clients in Switzerland
  - Business strategy is centered on its global WM business and its universal bank in Switzerland, complemented by its GIAM business and its IB
  - UBS is present in all major financial centers worldwide (NY, London, CH, HK, Tokyo etc.)
  - It has offices in more than 50 countries and employs roughly 60k people (~22k in CH)

Innovations, Challenges, and Achievements (1)

- Speed-up of simulation

	1st version (on desktop)	2nd version	Current version
Simulation time	3 days	18 hours	1 hour

- The simulation of 500'000 default scenarios is parallelized along the MC dimension:

Scenario	Default
1	FALSE
2	TRUE
3	FALSE
...	...
500'000	FALSE

Scenario	Default
1	FALSE
2	TRUE
3	FALSE
...	...
1'000	FALSE

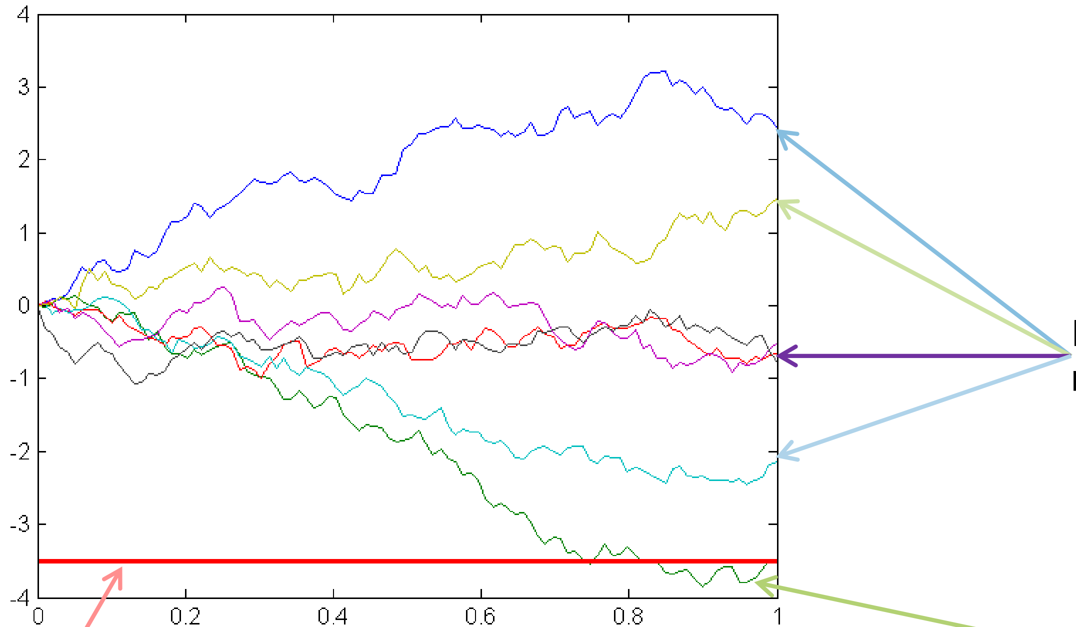
Innovations, Challenges, and Achievements (2)

- Credit portfolios can be quite large: # counterparties > 100'000
- MATLAB workers only have limited memory
- $\xrightarrow{\text{memory constraints}}$ There is a limit on MC simulations one can run on each MATLAB worker
- In our case, one worker can handle about 1'000 MC simulations

UBS

Structural Merton model

- Company A's asset returns are governed by a Brownian motion $d\rho_t = \left(r - \frac{\sigma^2}{2}\right) * dt + \sigma * dW_t$
- We perform Monte Carlo simulations to obtain 500'000 scenarios
- Default occurs if asset (returns) fall below a threshold implied by the liability level



In these scenarios, Company A would not have defaulted

Default threshold implied

by probability of default/rating

In this scenario, Company A would have defaulted

UBS

A Merton-type Bernoulli mixture model

- A firm's asset returns depend on common factors and specific factors
- Common factors drive the correlation between different firms' asset returns
- Structural Merton model $\xrightarrow{becomes}$ Merton-type Bernoulli mixture model

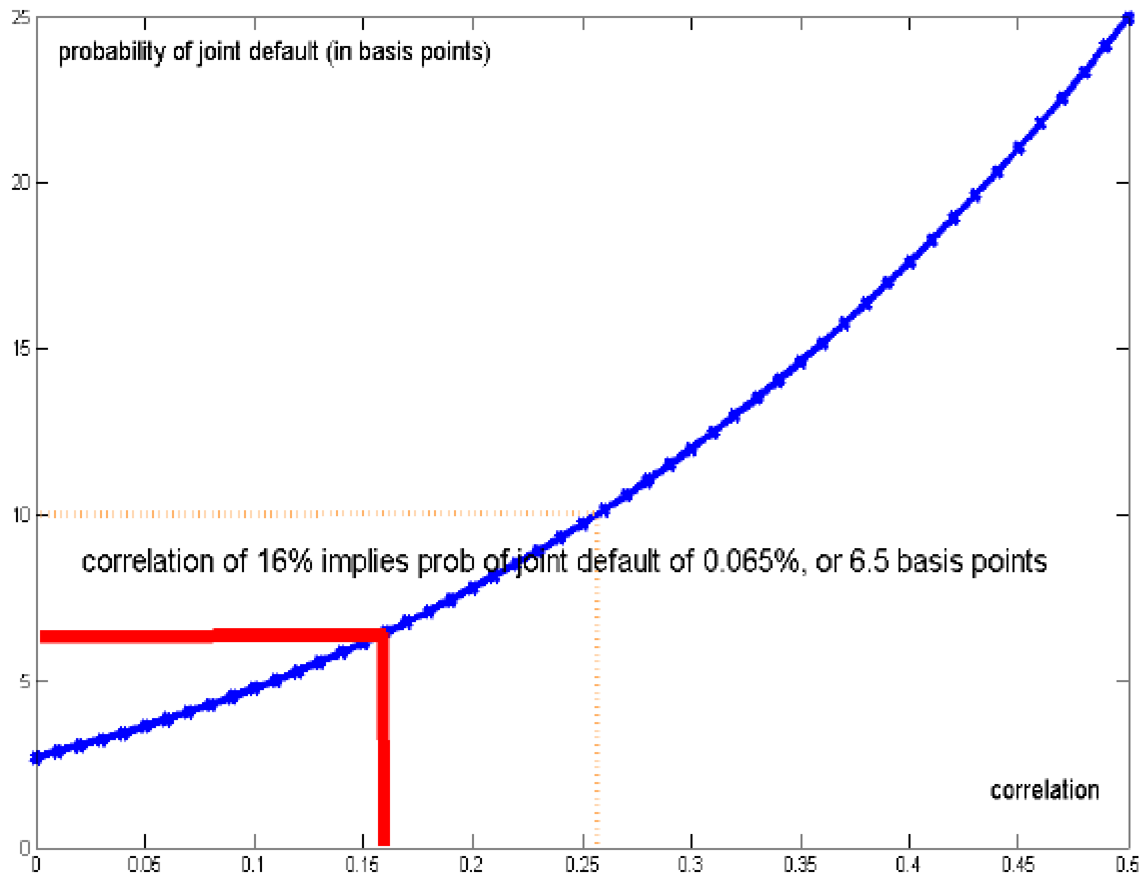
UBS

Probability of Joint Default

- In the one-factor portfolio model with uniform correlation ρ, the probability that two counterparties i, j default *jointly is given by*

$$\mathbf{JPD}_{i,j} = \mathbf{P}\left[l_i = 1, l_j = 1\right] = \boldsymbol{\Phi}_2\left[\Phi^{-1}[p_i], \Phi^{-1}\left[p_j\right]; \rho\right]$$

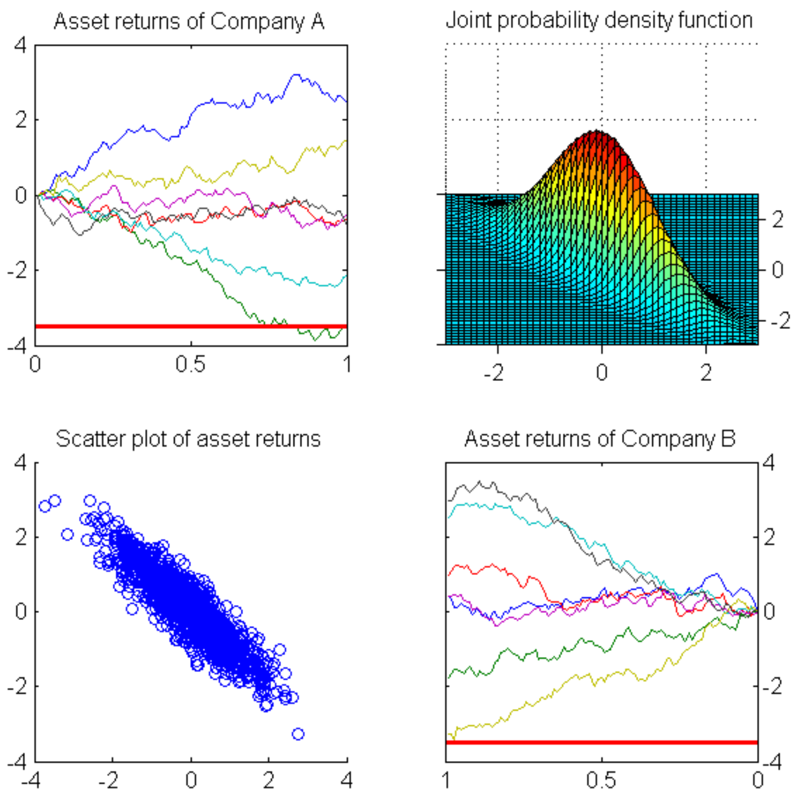
$\mathbf{JPD}_{i,j}$



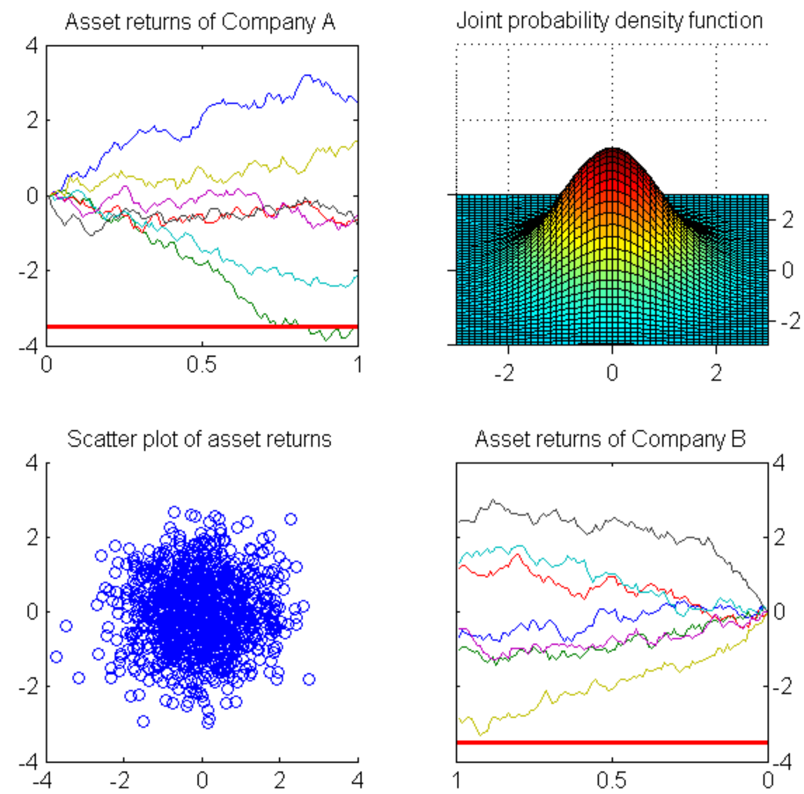
ρ

Correlated defaults (1)

Correlation $\rho = -90\%$



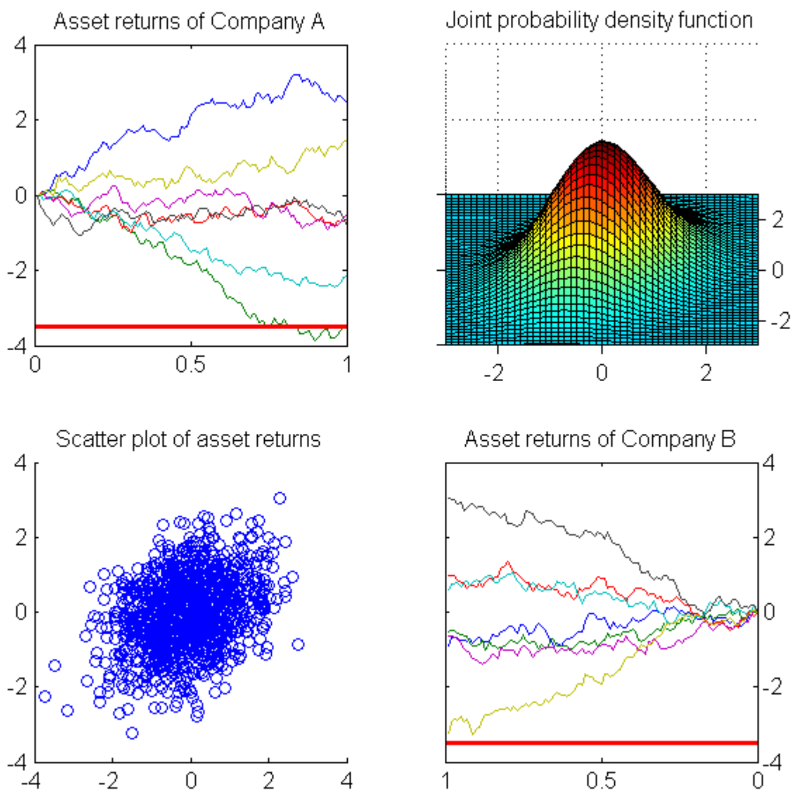
Correlation $\rho = 0\%$



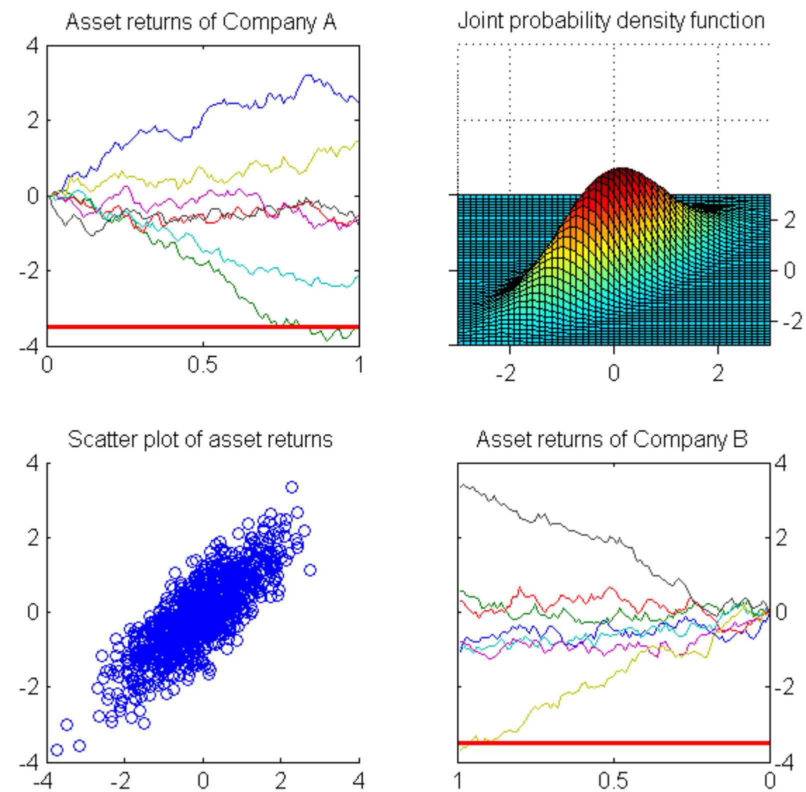
UBS

Correlated defaults (2)

Correlation $\rho = 30\%$



Correlation $\rho = 80\%$



Outline of Simulation

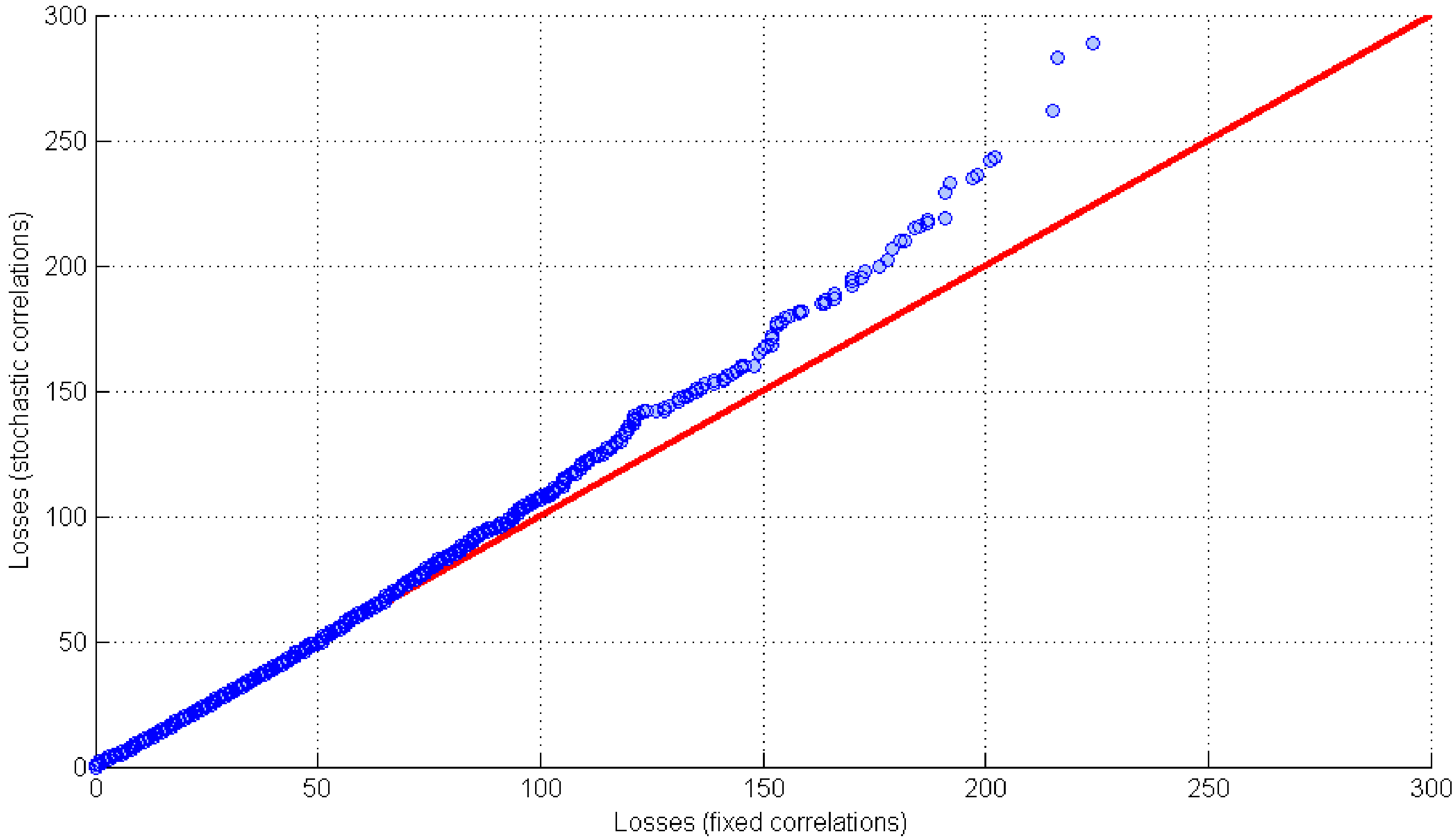
- Returns are simulated jointly using a multi-factor model

$$\boldsymbol{r}_t = \boldsymbol{B} * \boldsymbol{F}_t + \boldsymbol{\varepsilon}_t, \qquad \mathrm{Cov}\left(\boldsymbol{r}_t, {\boldsymbol{r}_t}^{\mathrm{T}}\right) = \boldsymbol{B} * \boldsymbol{B}^{\mathrm{T}} + \boldsymbol{D}$$

1. Draw idiosyncratic returns $\boldsymbol{\varepsilon}_t \sim N(\mathbf{0}, \mathrm{diag}(\boldsymbol{D})\ )$
2. Draw a covariance matrix $(\boldsymbol{B} * \boldsymbol{B}^{\mathrm{T}}) \sim SW_n\left(\frac{1}{P}\boldsymbol{B} * \boldsymbol{B}^{\mathrm{T}}; P\right)$
3. Draw systematic returns $(\boldsymbol{B} * \boldsymbol{F}_t) \sim N\left(\mathbf{0}, (\boldsymbol{B} * \boldsymbol{B}^{\mathrm{T}})\right)$
4. Create full returns $\boldsymbol{r}_t = (\boldsymbol{B} * \boldsymbol{F}_t) + \boldsymbol{\varepsilon}_t$
5. Standardize returns $\widetilde{\boldsymbol{r}_t} = \boldsymbol{r}_t .* \left(\mathrm{diag}\left(\boldsymbol{B} * \boldsymbol{B}^{\mathrm{T}}\right) + \mathrm{diag}(\boldsymbol{D})\right)^{-\frac{1}{2}}$
6. Compute loss indicator $\boldsymbol{l} = \mathbf{1}_{\{\widetilde{\boldsymbol{r}_t} < \Phi^{-1}(\boldsymbol{PD})\}}$
7. Compute loss distribution $\boldsymbol{L} = \boldsymbol{EAD} .* \boldsymbol{LGD} .* \boldsymbol{l}$

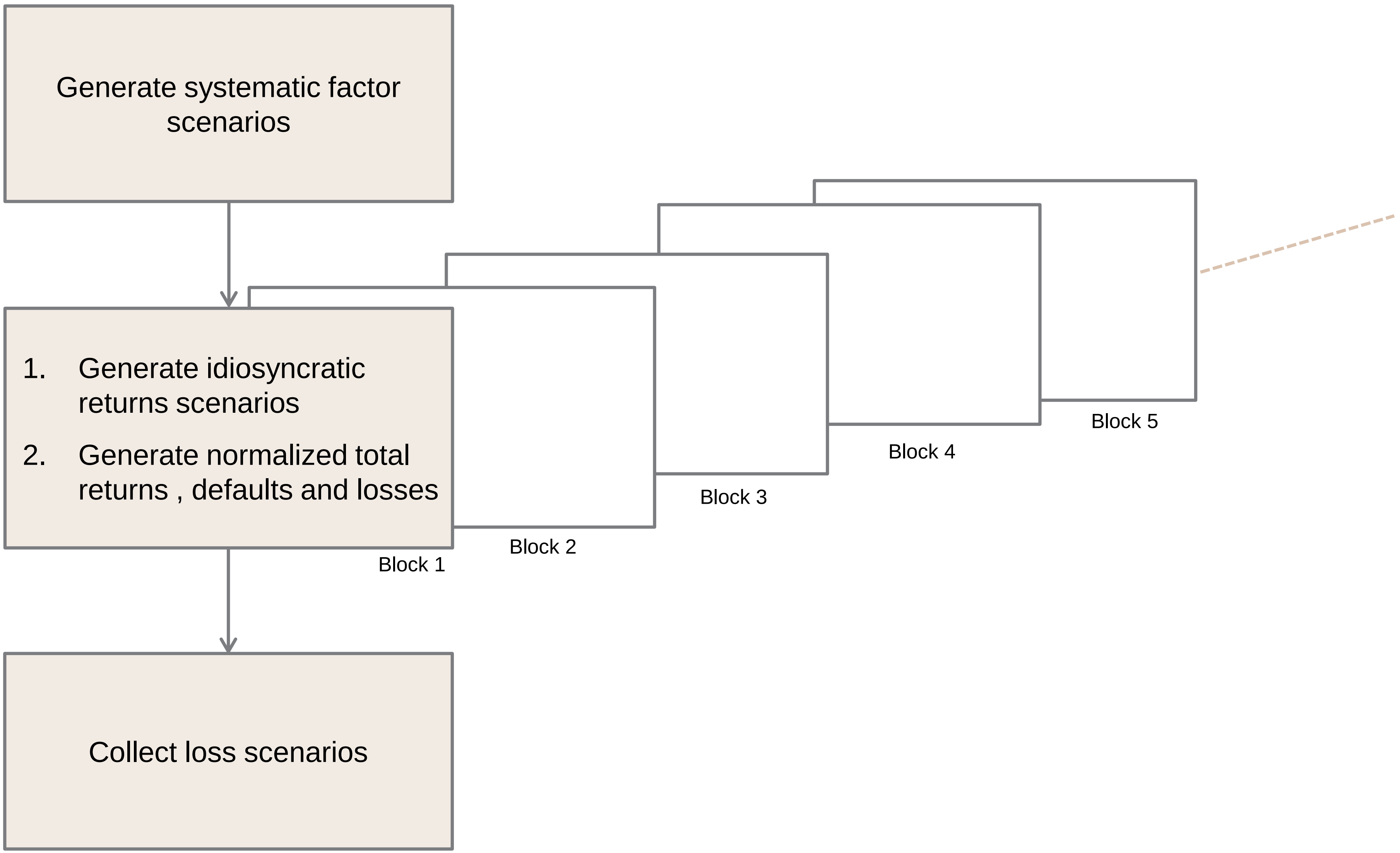
UBS

Random versus fixed correlations: impact on loss distribution



- Each blue circle depicts a loss scenario. The x-value shows the realized loss based on fixed correlations, while the y-value indicates the corresponding realized loss arising from random correlations. While the maximum loss in the fixed correlations regime is only CHF 225m, it is CHF 290m with random correlations. – If both loss distributions were identical, all the loss scenarios would lie on the red line.

Code Architecture: Illustration



Conclusion and Outlook

- Parallelization led to a remarkable 25x speedup of the simulations
- Challenges ahead:
  - Further reducing run time by simulating more efficiently
  - Finding a scheduler that does not self-destruct when offloading too big jobs
  - Handling huge data outputs (TB)