Algorithmics Incorporated

Valuation of Synthetic CDOs

Ben De Prisco, Alexander Kreinin

Algorithmics Inc.

RiskLab International Conference

Madrid, October 2005

Algorithmics Incorporated

Valuation of Synthetic CDOs

Ben De Prisco, Alexander Kreinin

Algorithmics Inc.

RiskLab International Conference

Madrid, October 2005

HR and other resources

Algorithmics

1. Ian Iscoe
2. Asif Lakhany
3. Jennifer Lai
4. Ahmed Nagi

RiskLab UofT

1. Luis Seco
2. Marcos Escobar

CS UofT

1. Ken Jackson
2. X. Ma

Publications:

Ian Iscoe, Alex Kreinin: "Pricing and Valuation of Synthetic CDOs " (Journal of

Ian Iscoe, Alex Kreinin: "Pricing and Valuation of Basket Default Swaps"

(Journal of Computational Finance)

Ben De Prisco, Ian Iscoe, Alex Kreinin: "Loss in Translation " (Risk Magazine

Outline:

Algorithmics Incorporated

- Financial introduction into CDOs.
- Monte Carlo Pricing.
- Balance Equations.
- Compound Poisson approximation.
- Recursive Methods
- CDO Valuation Methods: Performance & Accuracy.
- Future Directions

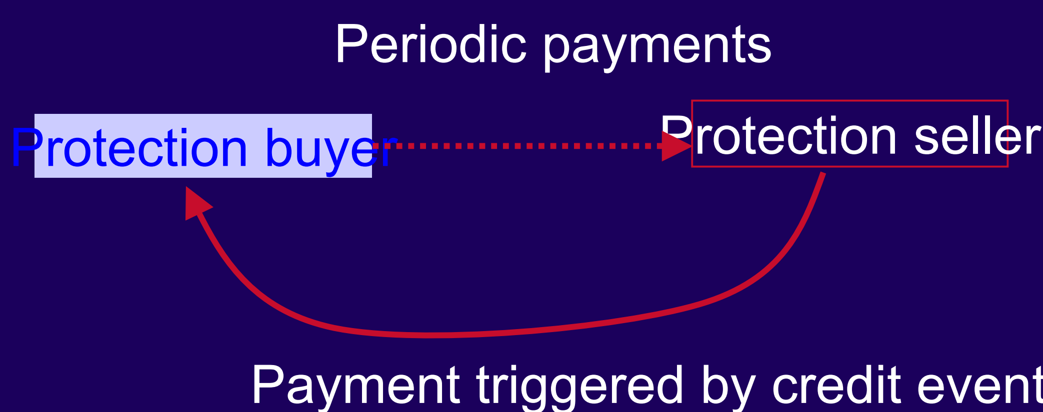
Basket Credit Derivatives

1. Application of Limit Theorems of Probability Theory if the number of names is large.
2. Efficient recursion for basket default event distribution (when the number of names is small).
3. Optimized quadratures designed for CD payoff functions

Introduction: CDO and BDS

Algorithmics Incorporated

Simplest form of synthetic credit derivative: single name credit defa



Basket default swap

The default model is more complicated

Credit event triggering payment may be 1st, 2nd, 3rd, default => two elements of randomness

Two things are important: who and when

True CDO

Algorithmics Incorporated

Collateral Pool of Credit Risky Instruments:

- Bonds
- Loans
- CDS
- ABS

Tranched Securities:

- Equity
- Mezzanines
- Senior

Cash flow distribution policy: “waterfall” diagram: Super-Senior, Senior, Mezzanine tranches get first money flow and only in the end Equity tranche holders get leftovers.

<u>Collateral Pool is actively managed.</u>

Pool needs to satisfy various quality tests. (Overcollateralization, interest coverage, diversity etc…)

Synthetic CDO

Algorithmics Incorporated

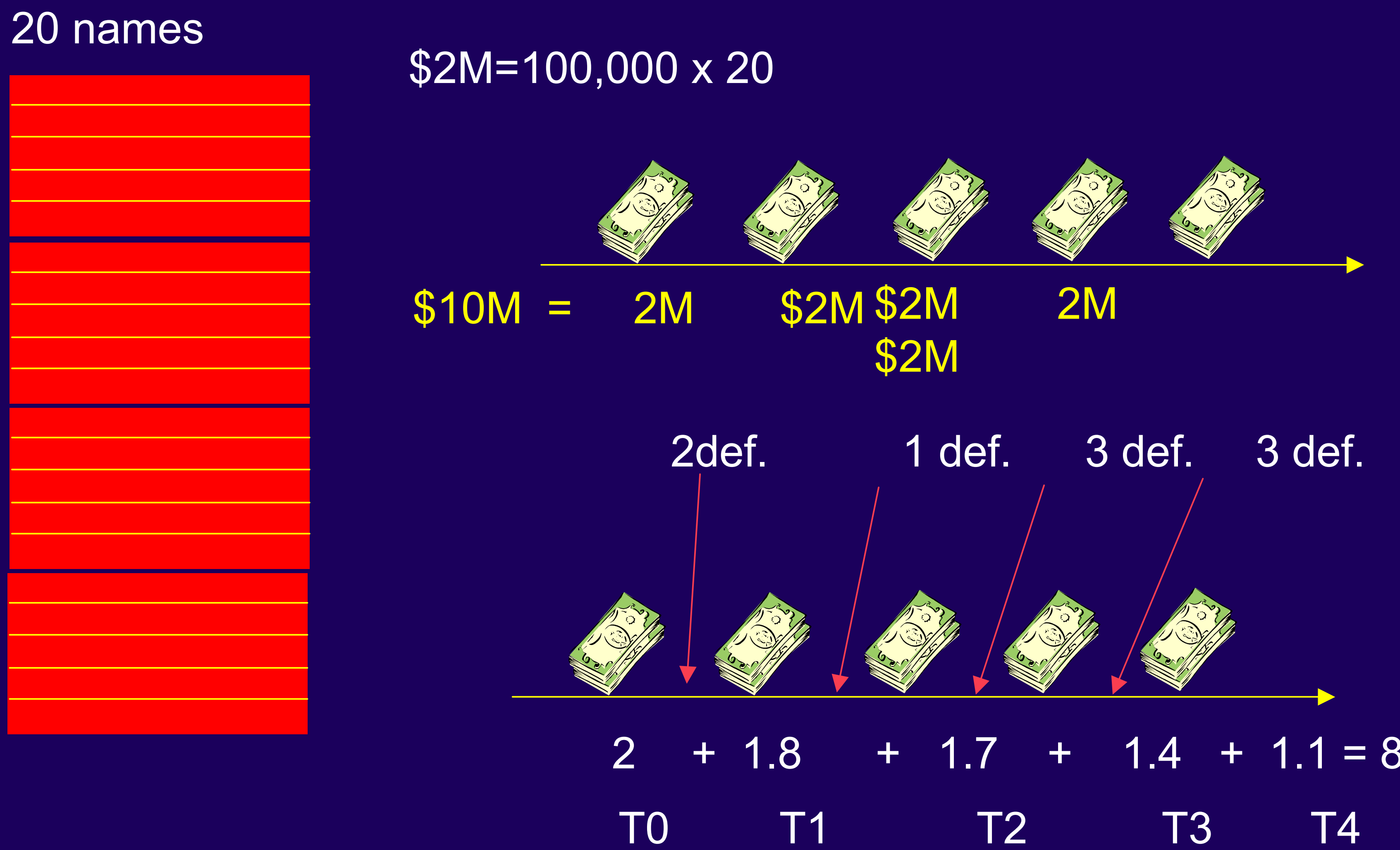
Premiums and Default payout cashflows accrue to various tranches in pre-specified allocations.

Synthetic Pool is NOT actively managed (i.e. underlying names do not change).

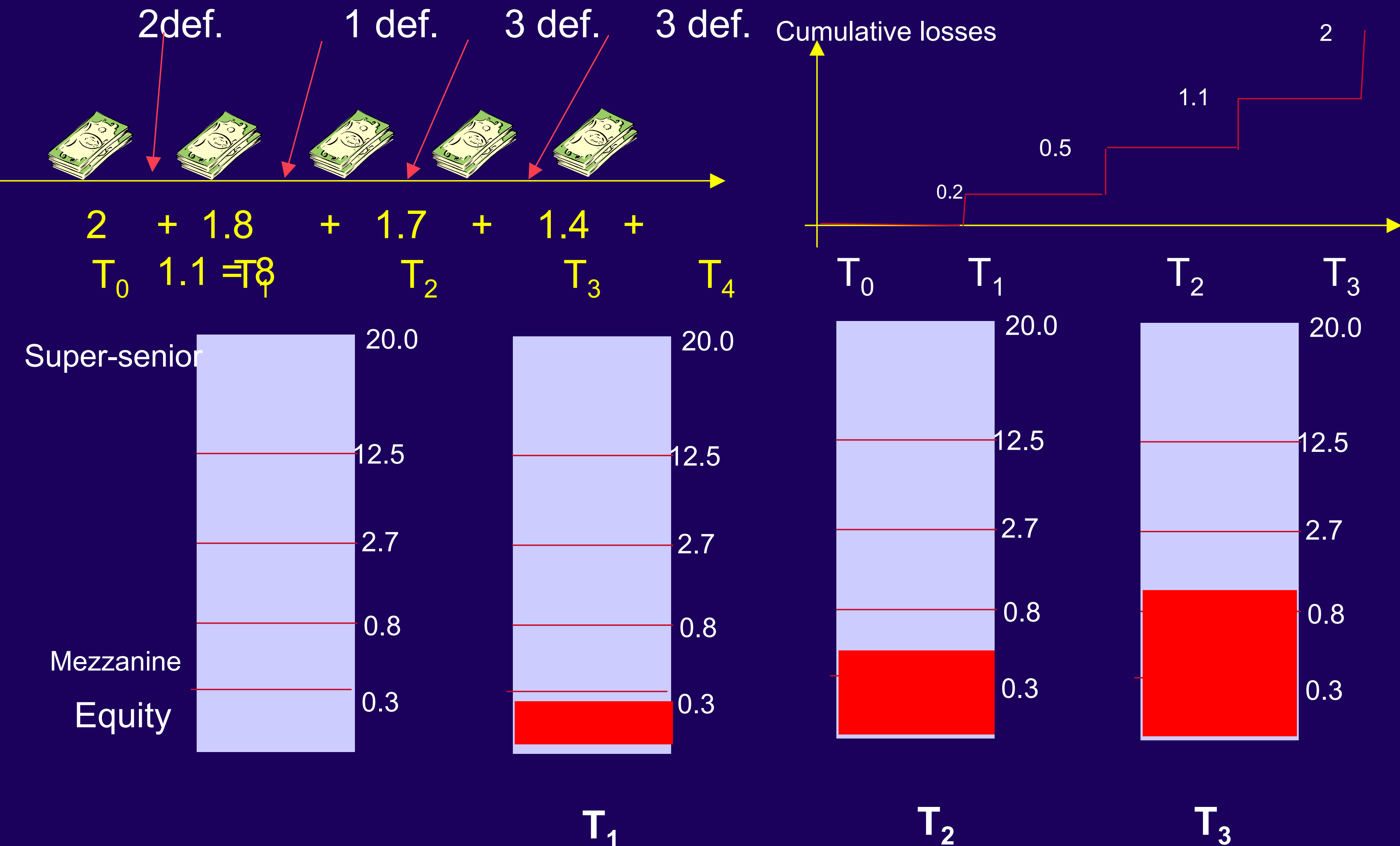
Synthetic CDOs often referred to as Tranched Portfolio Default Swaps

CDO Pool Cash Flow

Algorithmics Incorporated



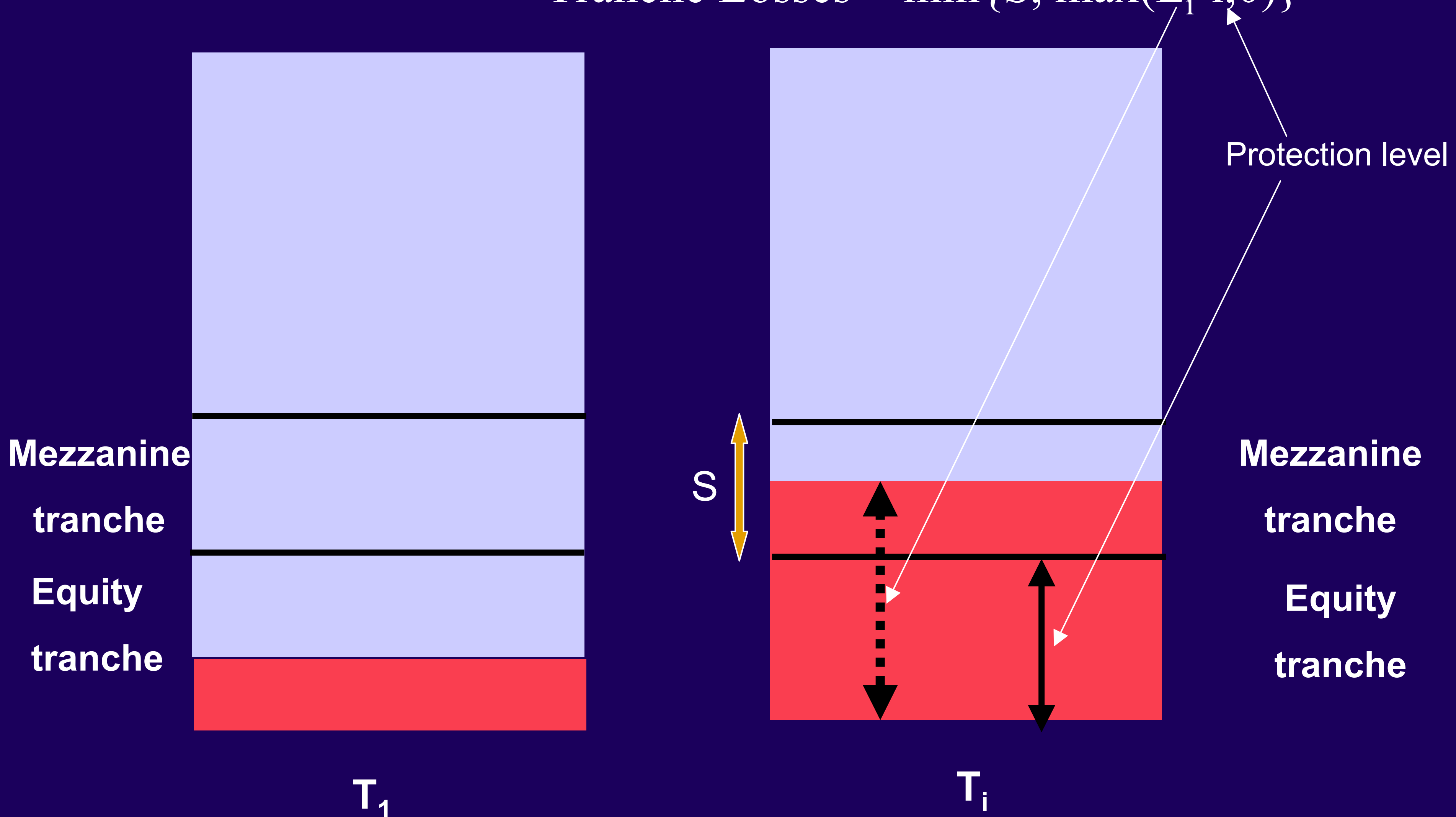
Synthetic CDO Tranches



Synthetic CDO Tranches

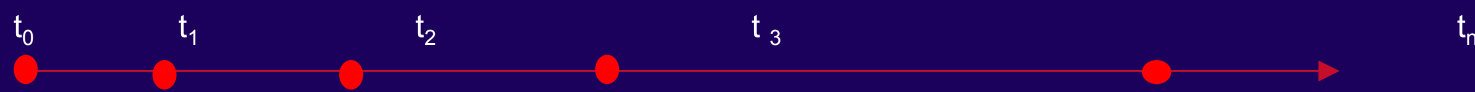
Algorithmics Incorporated

Tranche Losses = $\min\{S, \max(L_i - l, 0)\}$



Balance Equation for CDO

Algorithmics Incorporated



$$\mathrm{E}\left[\sum_{i=1}^{n} s_i (t_i - t_{i-1}) \cdot (S^{(j)} - L_i^{(j)}) \cdot d_i\right] = \mathrm{E}\left[\sum_{i=1}^{n} (L_i^{(j)} - L_{i-1}^{(j)}) \cdot d_i\right]$$

$$s = \frac{\sum_{i=1}^{n} (\mathrm{E}L_i - \mathrm{E}L_{i-1}) \cdot d_i}{\sum_{i=1}^{n} (t_i - t_{i-1}) \cdot (S - \mathrm{E}L_i) \cdot d_i}$$

The problem is reduced to computation of $\mathrm{E}L_i$

MC Pricing: General Idea

Algorithmics Incorporated

Expectation of premium payments = Expected Losses

1. Simulate correlated default events through time (correlated default times) for names in underlying pool
2. Under each path, apply the pay-offs of either N^{th} to Default basket or CDO tranche
3. Value represented by average value across all MC paths

Conditional Independence

Algorithmics Incorporated

(X, μ) – generic space of (latent) variables with a measure μ;

$$X = x: \quad P(x) = \begin{Vmatrix} p_1^{(1)}(x) & p_2^{(1)}(x) & \ldots & p_n^{(1)}(x) \\ p_1^{(2)}(x) & p_2^{(2)}(x) & \ldots & p_n^{(2)}(x) \\ \ldots & \ldots & \ldots & \ldots \\ p_1^{(K)}(x) & p_2^{(K)}(x) & \ldots & p_n^{(K)}(x) \end{Vmatrix}$$

Let $N = (N^{(1)}, \ldots, N^{(K)})$ be vector of recovery-adjusted notionals

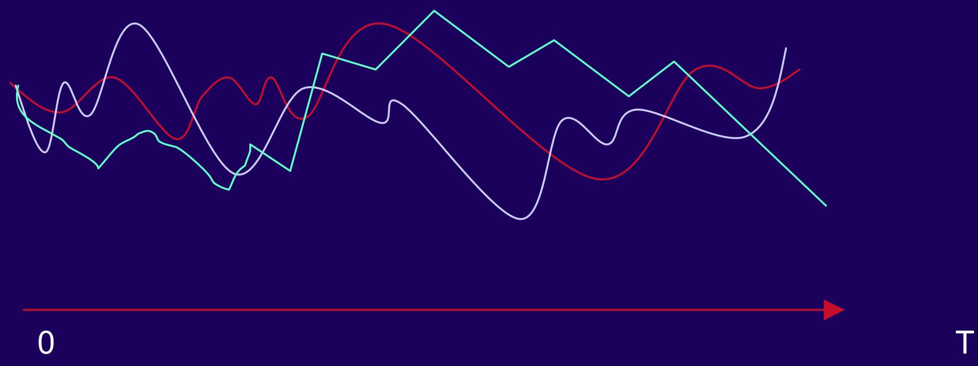
$$EL_i(x) = F(P(x), N, l, u)$$

$$EL_i = \int EL_i(x)\, d\mu(x) = \int F(P(x), N, l, u)\, d\mu(x)$$

The Space X and Measure μ

Algorithmics Incorporated

X- C([0, T]), μ – Gaussian :

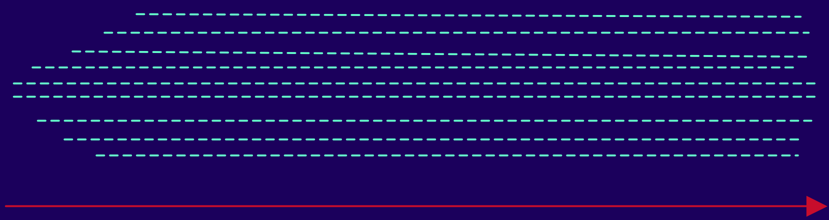


Iscoe, Kreinin, Rosen – ARQ (1999)

Hull, White (2001) – JD (2001)

m=1

X = $\mathbf{R}^m$, μ – Gaussian, T-distribution:



Andersen, Sidenius, Basu (2003) Risk : Recursive valuation of pool loss distriution

Brasch (2004) – Working paper: Recursive valuation of mean tranche losses

Hull, White (2004) JD : Recursive valuation of pool loss distribution using convolution of hom. groups

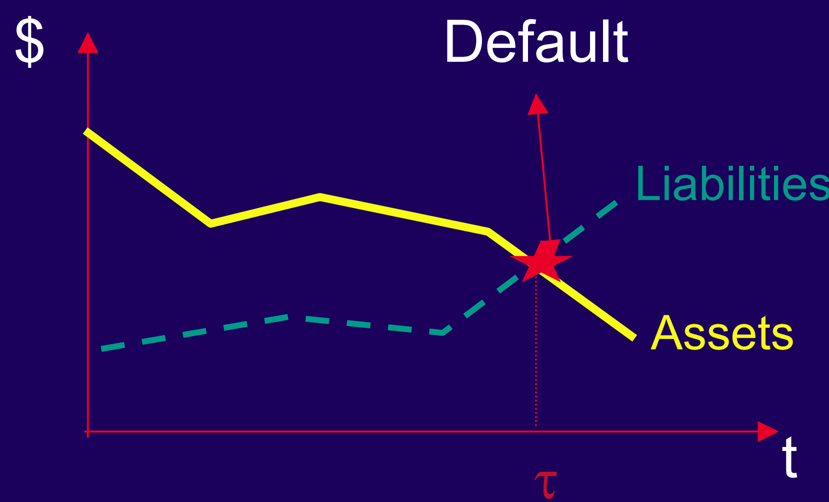
De Prisco, Iscoe, Kreinin (2005)) Risk : Compound Poisson Approximation of the pool loss distribution

Correlation Model

Algorithmics Incorporated

Correlated Credit Events for Estimation of Losses

Merton's Model



How can we measure creditworthiness ?

$$Y_i^{(k)} \equiv Y^{(k)} = \sum_{m=1}^{M} \alpha_m^{(k)} X_m + \beta^{(k)} \varepsilon^{(k)},$$

Systemic Component

Specific Component

Conditional independence framework

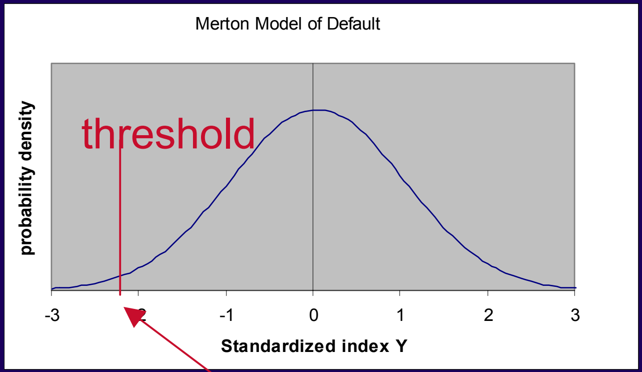
$$Y_i^{(k)} \equiv Y^{(k)} = \sum_{c=1}^{C} \alpha_c^{(k)} X_c + \beta^{(k)} \varepsilon^{(k)}, \quad i = 1,2,\ldots,n, \quad k = 1,2,\ldots,K,$$

CWI

Credit drivers

Idiosyncratic component

$$\sum_{c=1}^{C} \sum_{c'=1}^{C} \alpha_c^{(k)} R_{cc'} \alpha_{c'}^{(k)} + \left(\beta^{(k)}\right)^2 = 1.$$



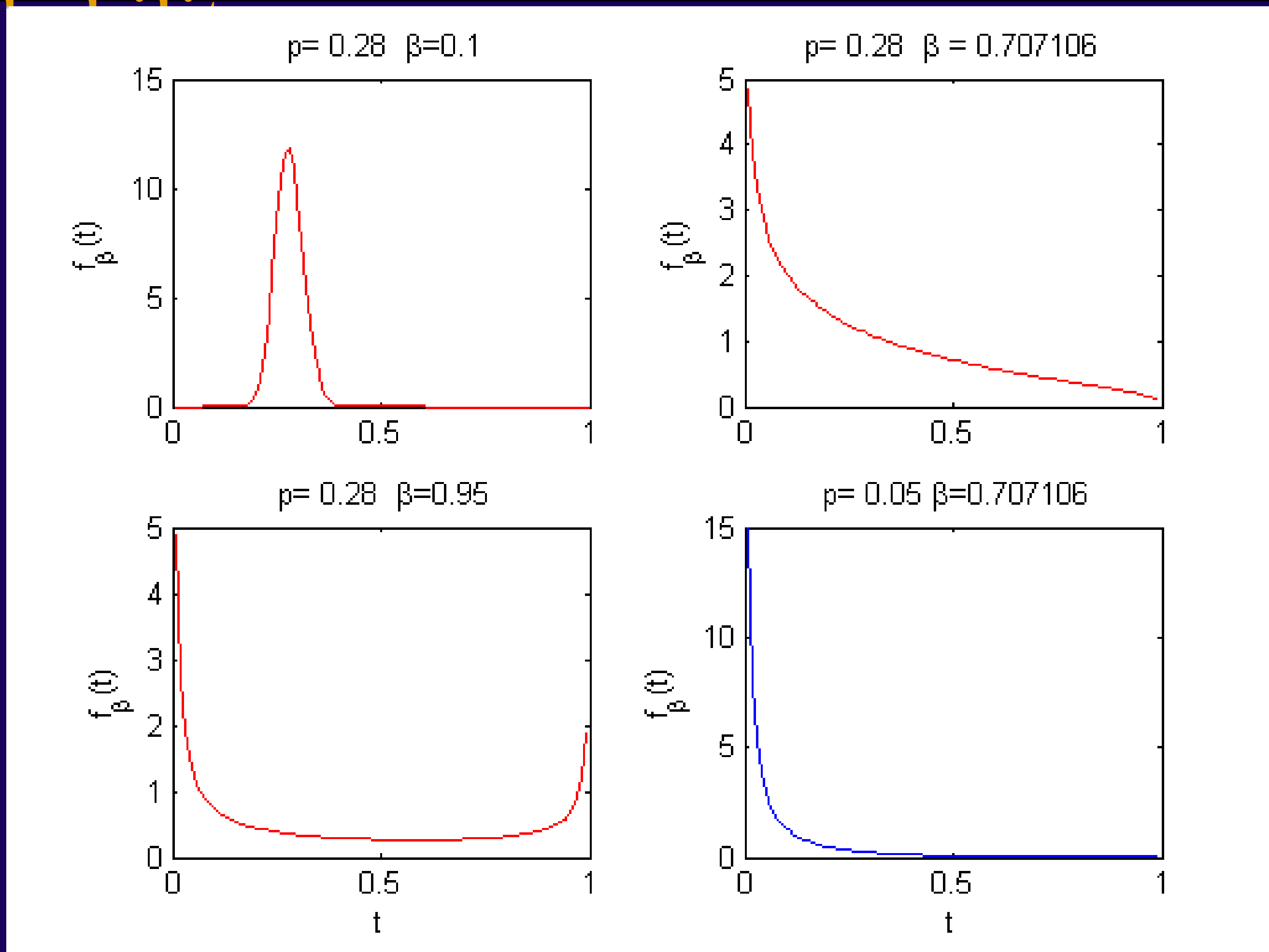
Conditional default probabilities

$$\hat{\pi}_i^{(k)}(x) = \Phi\left(\left[H_i^{(k)} - \sum_{c=1}^{C} \alpha_c^{(k)} x_c\right] / \beta^{(k)}\right).$$

$$p = \Phi(H)$$

Unconditional default probability

Conditional default pro



$$\hat{\pi}_i^{(k)}(x) = \Phi\left(\left[H_i^{(k)} - \sum_{c=1}^{C} \alpha_c^{(k)} x_c\right] / \beta^{(k)}\right).$$

Unconditional probabilities

$$\Pi_i^{(k)} = \int_{\mathbb{R}^C} \Pi_i^{(k)}(x)\, d\varphi(x), \quad k = 1, 2, \ldots, K, \quad i = 1, 2, \ldots, n,$$

Unconditional mean discounted losses

$$\mathbb{E}\left(L_\tau d_\tau\right) = \int_{\mathbb{R}^C} \mathbb{E}_x\left(L_\tau d_\tau\right)\, d\varphi(x),$$

$$(m-1)\cdot \Pr\left(\tau_m = \tau^{(k)}; \tau_m \in (t_{i-1}, t_i]\right) = \sum_{j \neq k} \Pr\left(\tau_{m-1}(B^j) = \tau^{(k)}; \tau_{m-1} \in (t_{i-1}, t_i]\right)$$
$$-(K+1-m)\cdot \Pr\left(\tau_{m-1} = \tau^{(k)}; \tau_{m-1} \in (t_{i-1}, t_i]\right)$$

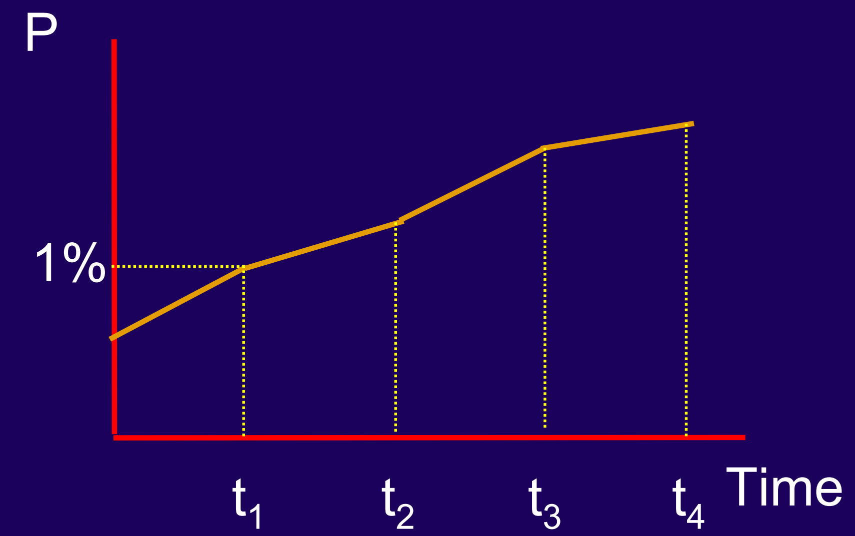
$$\Pr\left(\tau_m = \tau^{(k)}; \tau_m \in (t_{i-1}, t_i]\right) = \Pi_i^{(k)}(m)$$
$$\Pr\left(\tau_{m-1} = \tau^{(k)}; \tau_{m-1} \in (t_{i-1}, t_i]\right) = \Pi_i^{(k)}(m-1)$$

Calibration of Default Boundary

Algorithmics Incorporated

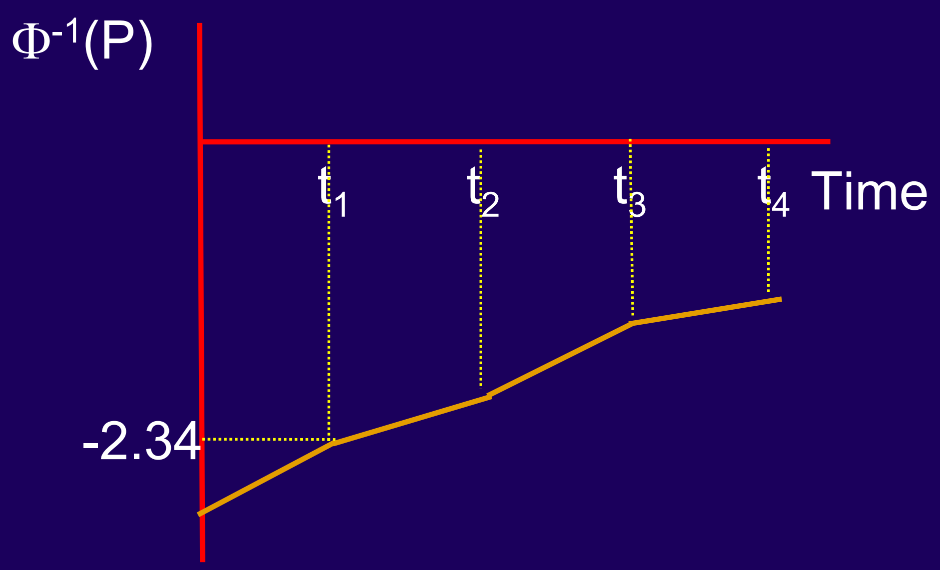
Risk Neutral Probability of Default Term Structure for each name (calibrated from Par CDS quoted spreads –> hazard rates)

$$\mathbb{P}\left(\tau^{(k)} \leq t_i\right) = \hat{\pi}_i^{(k)}, \qquad i = 1, 2, \ldots, n.$$



Translated to Default Boundary Curve for each name

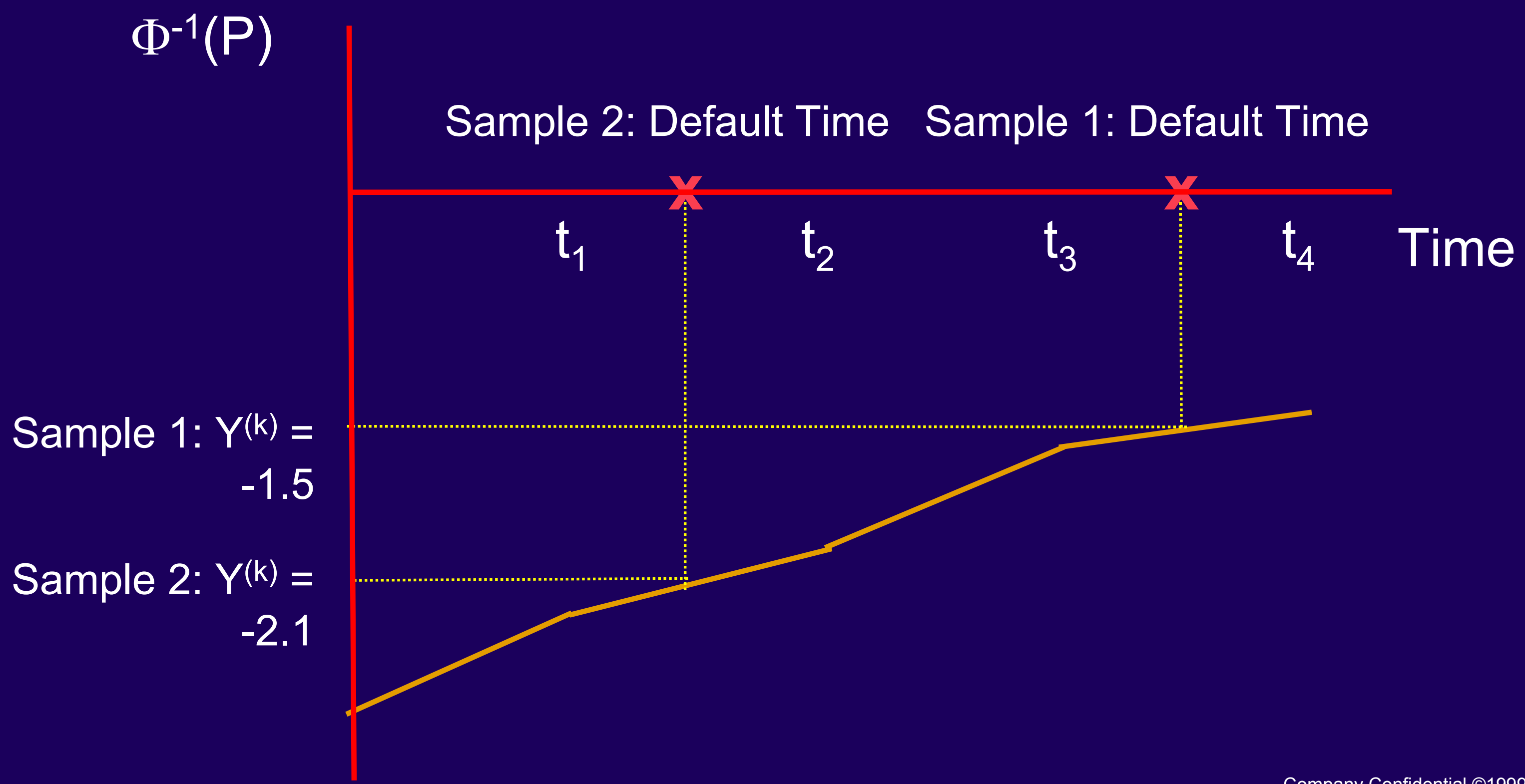
$$H_i^{(k)} = \Phi^{-1}(\hat{\pi}_i^{(k)}),$$



MC Pricing: Default Times

Name creditworthiness $Y^{(k)}$ sampled under each MC run

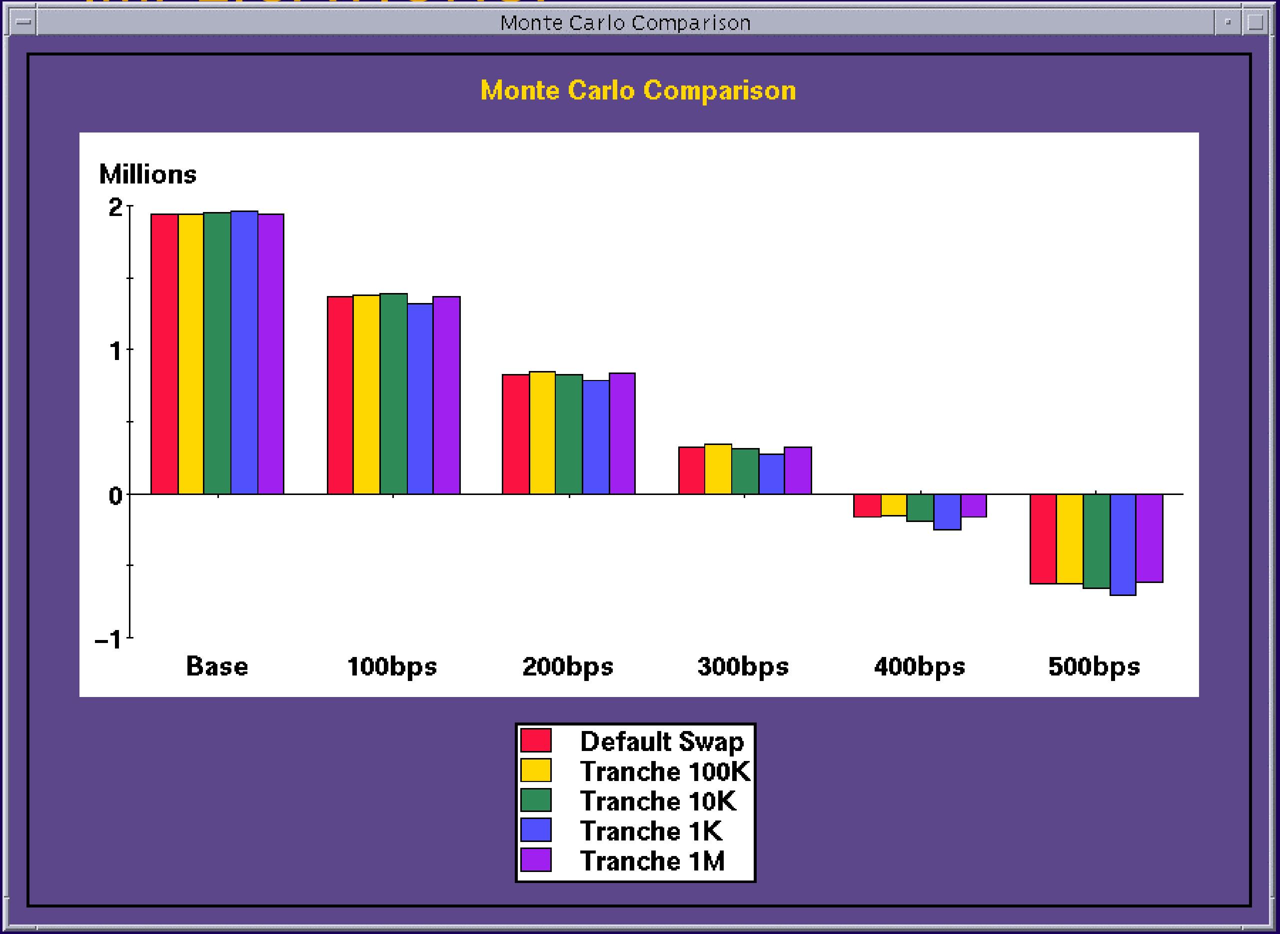
$$\tau^{(k)} = \min\{t_i : Y^{(k)} < H_i^{(k)}\} \qquad (\tau^{(k)} = +\infty \text{ if } Y^{(k)} \geq H_i^{(k)}, \text{ for all } 1 \leq i \leq n).$$



MC Pricing: Number of Paths

Algorithmics Incorporated

PERFORMANCE IMPLICATIONS!



Requires 1000s of pricing paths under single scenario to estimate value appropriately!

Need fast analytic approximations.

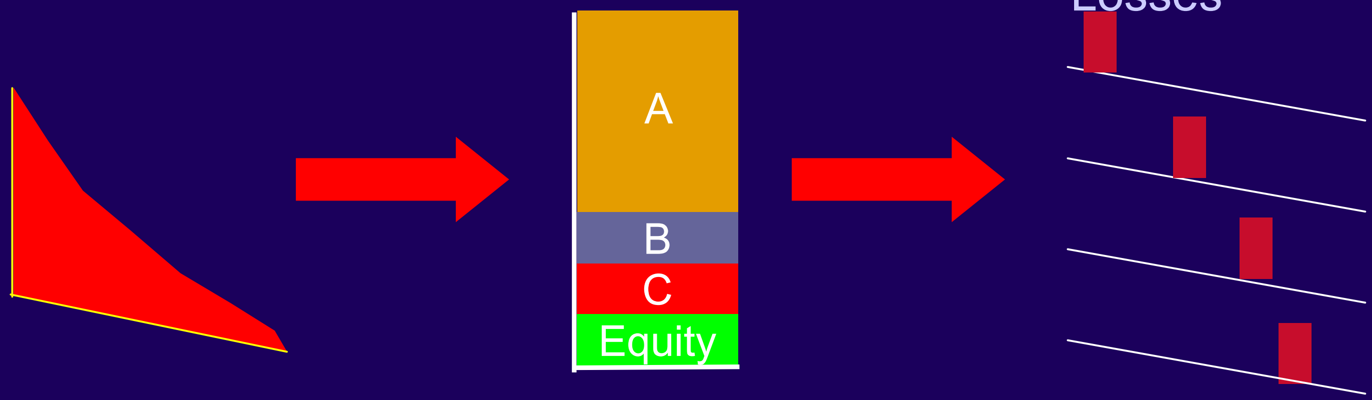
Alternative Approach

Review

5. Unconditional Pool Losses

5a. Tranche Loss Profiles

5b. Tranche Expected Losses



H-W approach, as well as various other analytic flavours generally require the derivation of entire pool loss distribution before tranche profiles are overlaid.

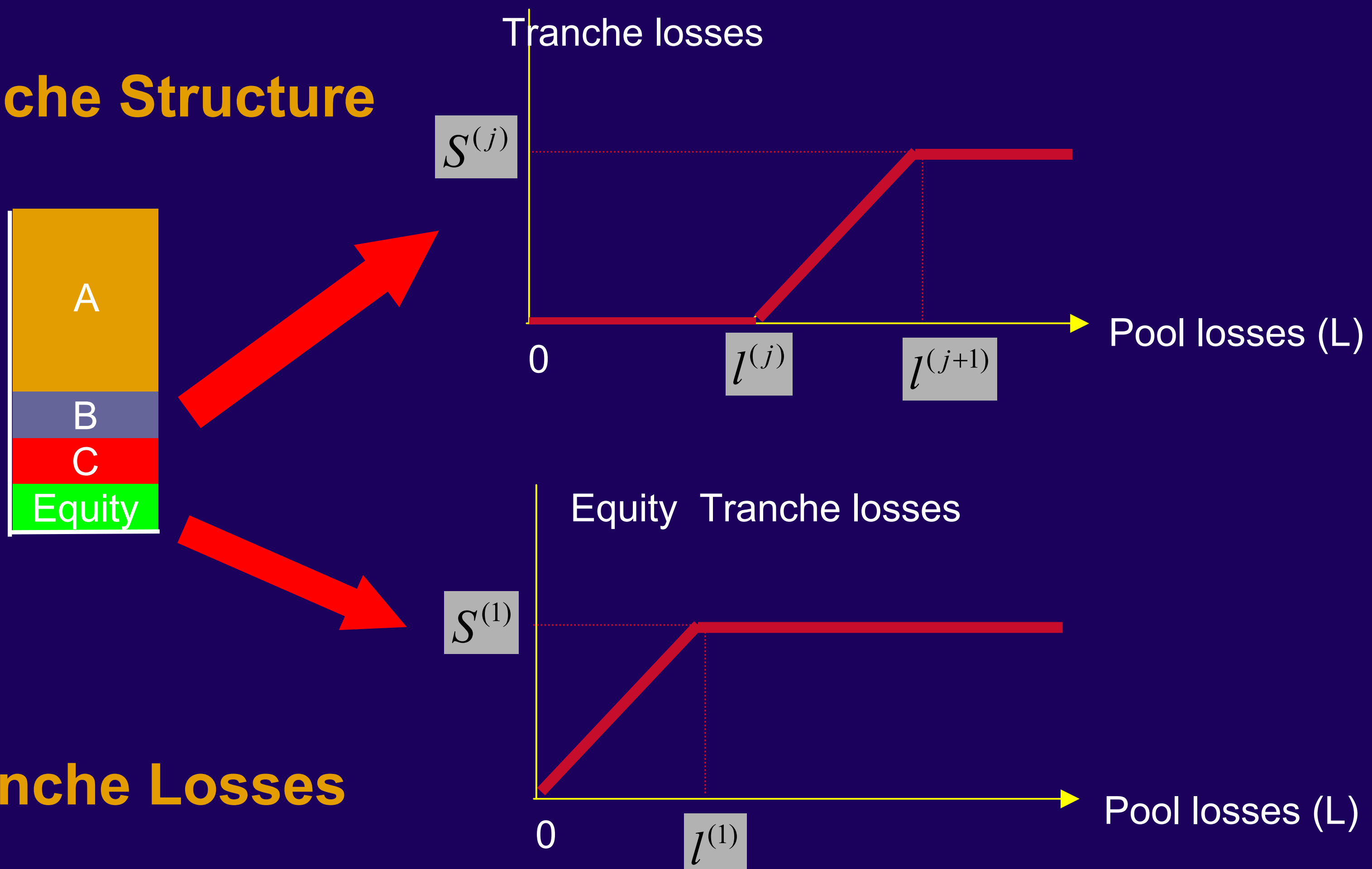
This is a time consuming process and ignores the fact that investors may often own only certain tranches.

To speed up computation time, pool loss distribution should focus on relevant bands for the tranches held.

Tranche Losses

Tranche Structure

Tranche Losses



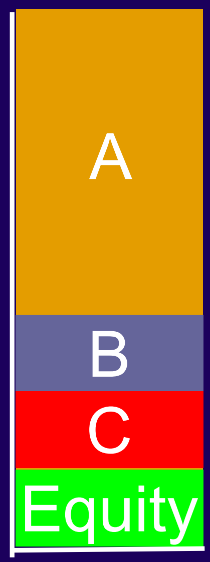
$$L^{(j)} = (L - l(j))^{+} - (L - l(j) - S(j))^{+}$$

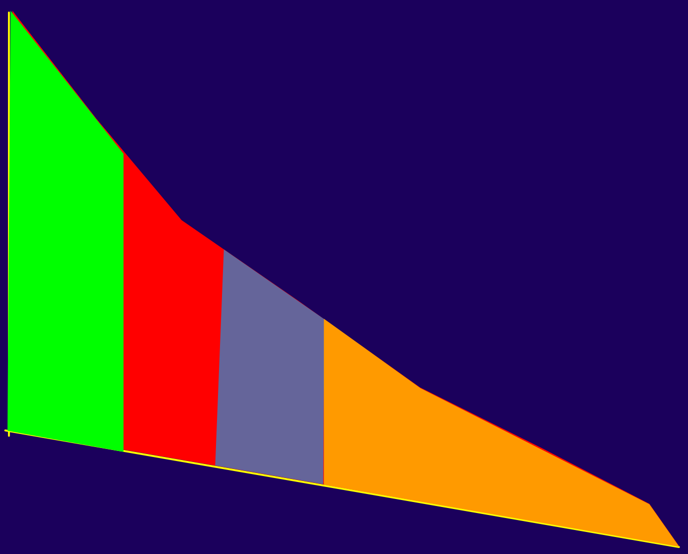
Tranche loss distribution

Algorithmics Incorporated

Tranche Structure

Pool Loss Distribution





Therefore, we only need to integrate the distribution for the bands we care about!!!

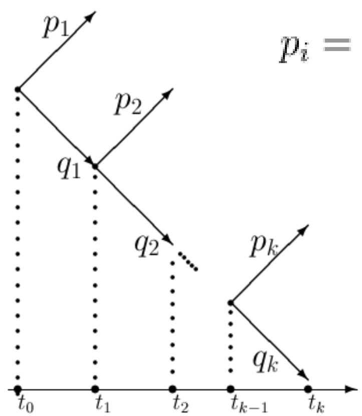
Homogeneous Pool

Algorithmics Incorporated

Consider the simplest case: instruments in pool have common notional and common term structure of default probabilities:

$$\pi_i = \mathbb{P}\left(\tau = t_i\right), \quad i = 1, 2, \ldots, n$$

Conditional default probabilities:



$$p_i = \mathbb{P}\left(\tau = t_i \,|\, \tau \geq t_i\right)$$

$$p_i = \frac{\pi_i}{1 - \sum_{m=1}^{i-1} \pi_m}$$

$$\pi_i = p_i \cdot \prod_{m=1}^{i-1} q_m$$

Therefore, the number of instruments in default at time t_i, has Binomial distribution:

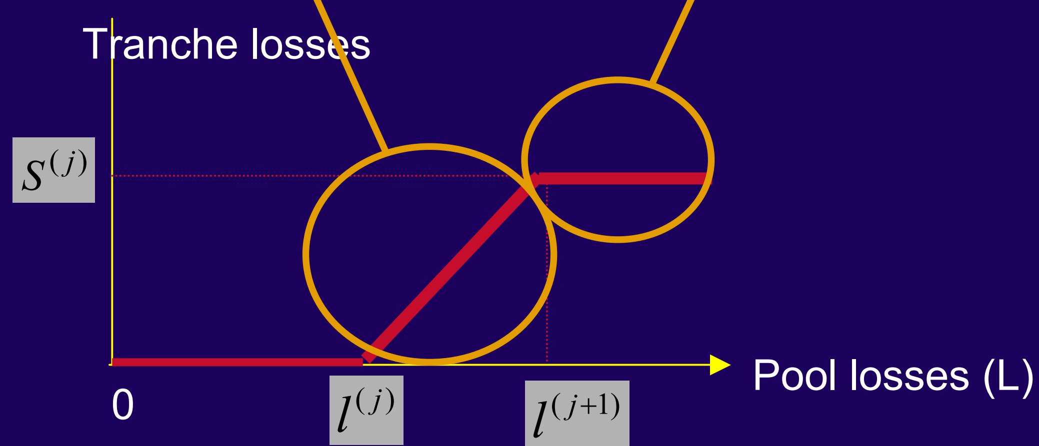
$$\mathbb{P}\left(\nu_1 = k_1\right) = \binom{K}{k_1} p_1^{k_1} q_1^{K-k_1}, \quad q_1 = 1 - p_1,$$

Homogeneous Pool

Therefore, for a given tranche, expected losses can be derived directly as:

$$\mathbb{E}\left[L_i^{(j)}\right] = \sum_{1 \vee \frac{l^{(j)}}{N_1} \leq k < \frac{l^{(j+1)}}{N_1}} (N_1 k - l^{(j)}) \mathrm{Bin}(k; K, \hat{\pi}_i) \quad + \quad S^{(j)} \sum_{\frac{l^{(j+1)}}{N_1} \leq k \leq K} \mathrm{Bin}(k; K, \hat{\pi}_i). \quad (3.15)$$

where $\hat{\pi}_i = \sum_{m=1}^{i} \pi_m, 1 \leq i \leq n$ represents cumulative default probabilities



Convergence to Poisson process

Independent default events

$L_i = N^{(1)} \cdot \nu_i$, where ν_i is the number of default events by the time t_i.

$$\nu_i = \sum_{k=1}^{K} I(\tau^{(k)} \le t_i); \quad \nu_i \Rightarrow \mathrm{Pois}(\hat{\pi}_i)$$

$$\hat{\pi}_i = \sum_{m=1}^{i} \pi_m, \quad 1 \le i \le n$$

$$\nu_i(x,K) \Rightarrow \mathrm{Pois}(\hat{\pi}_i(x)) \qquad (\text{as K} \to \infty)$$

$\underset{\mu}{\mathrm{Mix}}$ ↓ $\qquad \underset{\mu}{\mathrm{Mix}}$ ↓

$$\upsilon(K) \Rightarrow \upsilon^*$$

?

$$\nu_t(x,K) \Rightarrow \mathrm{Pois}(\hat{\pi}_t(x)) \qquad (\text{as K} \to \infty)$$

$\underset{\mu}{\mathrm{Mix}}$ ↓ $\qquad \underset{\mu}{\mathrm{Mix}}$ ↓

$$\upsilon_t(K) \Rightarrow \upsilon_t^*$$

Poisson approximation for homogeneous pool

Algorithmics Incorporated

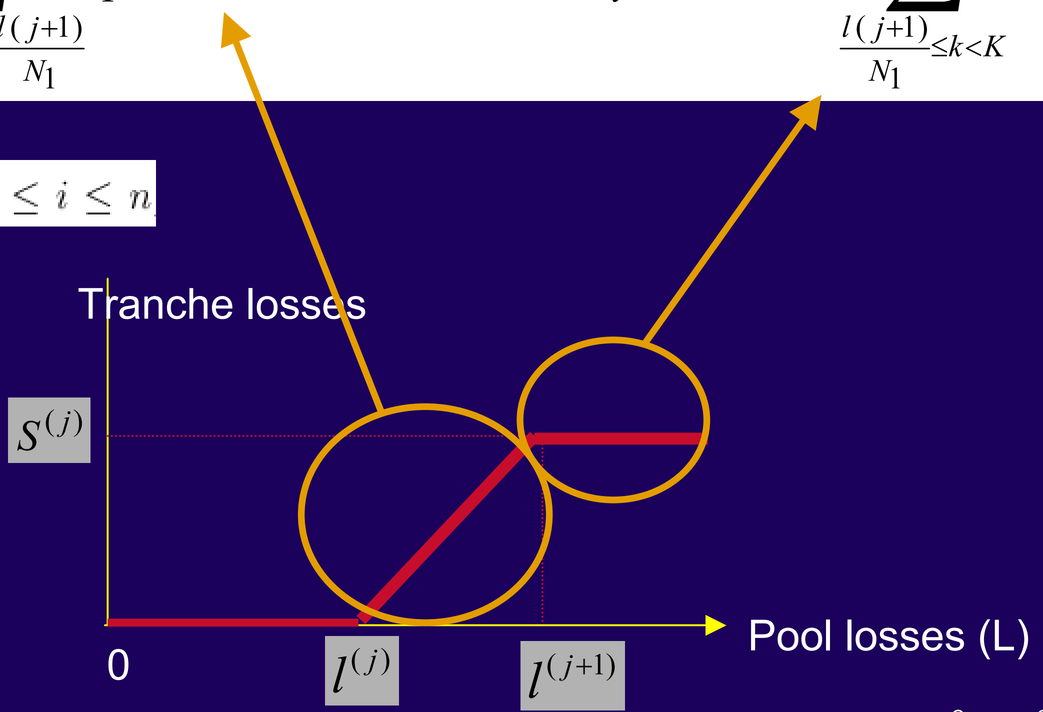
If the pool is large enough, then v_i is approxim $Poisson(\hat{\lambda}_i)$

$$\mathbb{P}(\nu_i = k) \doteq e^{-\hat{\lambda}_i}(\hat{\lambda}_i)^k/k!, \quad k = 0, 1, \ldots.$$

Therefore expected tranche losses can be calculated as:

$$E[L_i^{(j)}] \cong \sum_{1 \vee \frac{l(j)}{N_1} \le k < \frac{l(j+1)}{N_1}} (N_1 k - l^{(j)}) Pois(k; \hat{\lambda}_i) + S^{(j)} \sum_{\frac{l(j+1)}{N_1} \le k < K} Pois(k; \hat{\lambda}_i)$$

where $\hat{\lambda}_i = K\hat{\pi}_i, 1 \le i \le n$



Compound Poisson Process

Algorithmics Incorporated

A **compound Poisson process** can be represented as:

$$X(t) = \sum_{i=1}^{N(t)} Y_i, \qquad t \geq 0$$

Where {N(t), t>=0} is a Poisson process, and {Y_n, n>=0} is a family of independent and identically distributed random variables which are also independent of {N(t), t>=0}.

Heterogeneous Pool

Algorithmics Incorporated

Assume all underlying instruments in the pool have different notionals and individual term structures of default probabilities:

$$\pi_i^{(k)} = \mathbb{P}\left(\tau^{(k)} = t_i\right), \quad i = 1, 2, \ldots, n; \quad k = 1, 2, \ldots, K;$$

Generally speaking:

$$\begin{aligned}
\hat{\pi}_i^{(k)} &= \sum_{m=1}^{i} \pi_m^{(k)}, \quad 1 \leq i \leq n \\
\hat{\lambda}_i &= \sum_{k=1}^{K} \hat{\pi}_i^{(k)}, \quad 1 \leq i \leq n \\
f_i(N) &= \sum_{k:N_k=N} \hat{\pi}_i^{(k)} / \hat{\lambda}_i, \quad 1 \leq i \leq n, \quad N = 1, 2, \ldots
\end{aligned}$$

Note: f_i is simply the relative frequency of notional values

Heterogeneous Pool

Algorithmics Incorporated

Consider a random variable for Notionals in the pool denoted by $\mathcal{N}_i$ as having the probability mass function f_i described previously, then the distribution of pool losses can be represented approximately by the following compound Poisson process:

$$L_i \overset{\mathcal{D}}{\approx} \sum_{m=1}^{\mathcal{M}_i} \mathcal{N}_i^{(m)}$$

where $(N_i^{(m)})_{m=1}^{K}$ is an i.i.d. sequence of copies of $\mathcal{N}_i$ and indepe $M_i \sim Pois(\hat{\lambda}_i)$

Interpretation:

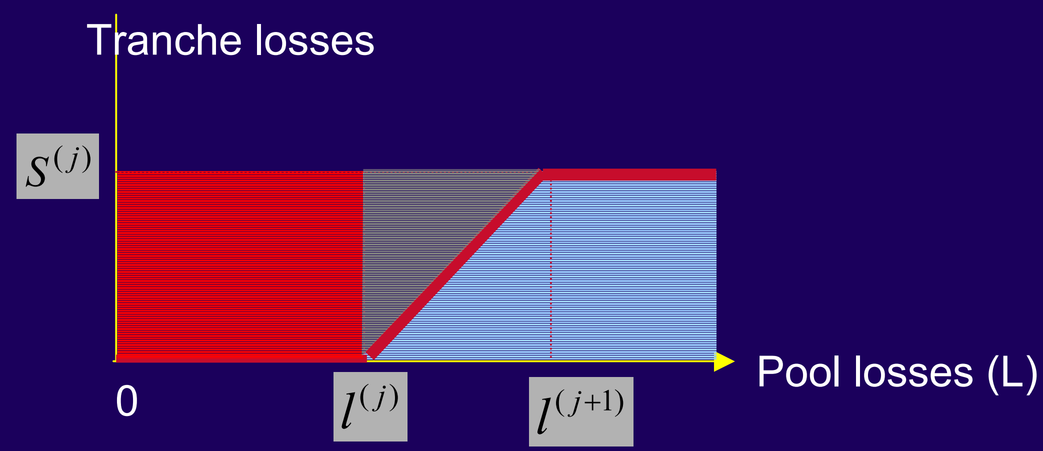
Pool default events arrive in accordance with a Poisson process. The loss suffered with each default event is assumed to be independent and identically distributed. Then $\{L_i, t>=0\}$ is a **compound Poisson process** where L_i denotes the cumulative default losses incurred by t and $\mathcal{N}_i^{(m)}$ represents the loss of the m^{th} default.

Expected tranche loss of heterogeneous pool

Algorithmics *Incorporated*

Therefore expected tranche losses can be calculated as:

$$\mathbb{E}L_i^{(j)} \doteq S^{(j)}(1-e^{-\hat{\lambda}_i}) - e^{-\hat{\lambda}_i}\left\{ S^{(j)} \sum_{m=1}^{l^{(j)}} \frac{\hat{\lambda}_i^m}{m!} \sum_{m\le N\le l^{(j)}} f_i^{\star m}(N) + \sum_{m=1}^{l^{(j+1)}-1} \frac{\hat{\lambda}_i^m}{m!} \sum_{l^{(j)}<N<l^{(j+1)}} \left[l^{(j+1)}-N\right] f_i^{\star m}(N)\right\}$$



Heterogeneous Pool

Algorithmics Incorporated

1. Scenarios (integration pts)

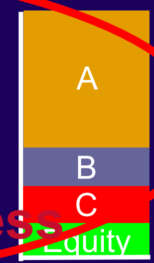
2. Conditional probabilities $p(X)$

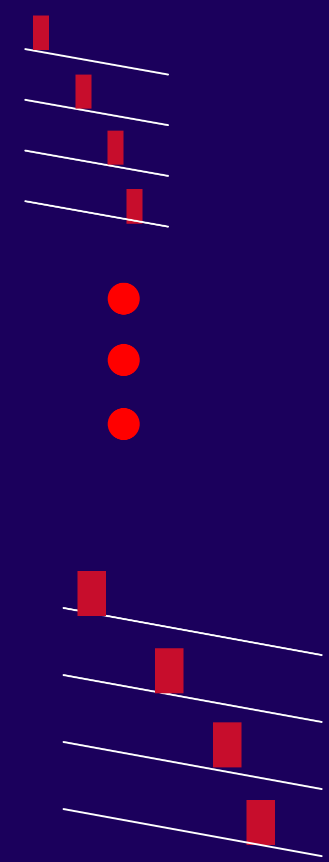
3. Obligor losses l

4. Tranche Loss Profiles

4a. Tranche Expected Losses

$X = x_1$ $\quad p_j(X = x_1)$, $j = 1, \ldots, n$ $\quad l_j$, $j = 1, \ldots, n$





Compound Poisson Process

$X = x_2$ $\quad p_j(X = x_2)$, $j = 1, \ldots, n$

$X = x_3$ $\quad p_j(X = x_3)$, $j = 1, \ldots, n$

5. Unconditional Expected Tranche Losses

CDO Performance/Accuracy

Algorithmics Incorporated

How many MC pricing paths are required?

Assume: Homogeneous pool where K=100 and initial credit ratings same

T=5yrs, Equity 3%, Mezz. Jr. 1%, Mezz. 2.1%, Senior 6%, S-Senior 87.9%

Different initial credit ratings

Credit rating	Credit Spreads (bp)				
	Equity	*Mezz. Jr.*	*Mezz.*	*Senior*	*S-Senior*
Baa3 (Sim.)	2418	466	102	2	0
Baa3 (Anal.)	2419	467	102	2	0
Baa2 (Sim.)	956	38	3	0	0
Baa2 (Anal.)	959	41	4	0	0
Baa1 (Sim.)	492	3	1	0	0
Baa1 (Anal.)	498	4	1	0	0
A3 (Sim.)	245	0	0	0	0
A3 (Anal.)	246	0	0	0	0
Aa2 (Sim.)	50	0	0	0	0
Aa2 (Anal.)	53	0	0	0	0

Table 2: CDO tranche spreads: MC vs. analytics

MC Error Estimates

95% conf intervals based on 10,000 p $c_0(Baa2) = [950,\ 967], \quad c_1(Baa2) = [36,\ 48], \quad c_2(Baa2) = [2,\ 5].$

95% conf intervals based on 100,000 $c_0(Baa2) = [953,\ 962], \quad c_1(Baa2) = [40,\ 43], \quad c_2(Baa2) = [3,\ 4].$

Conclusion: Reasonable error bounds with 100,000 paths

CDO Performance/Accuracy

Algorithmics Incorporated

Accuracy of Poisson Approximation

For homogeneous portfolio, binomial represents “true” value

Different number of underlying names in pool

K	Tranche	Baa2 Spreads (bp)		Baa1 Spreads (bp)	
		Poisson	Binomial	Poisson	Binomial
200	Equity	978	978	246	246
	Mezzanine Jr.	7	6	0	0
	Mezzanine	0	0	0	0
	Senior	0	0	0	0
150	Equity	977	977	246	246
	Mezzanine Jr.	7	6	0	0
	Mezzanine	0	0	0	0
	Senior	0	0	0	0
100	Equity	958	958	246	246
	Mezzanine Jr.	43	41	0	0
	Mezzanine	4	3	0	0
	Senior	0	0	0	0
75	Equity	938	941	246	246
	Mezzanine Jr.	72	69	1	1
	Mezzanine	9	9	0	0
	Senior	0	0	0	0
50	Equity	898	898	244	244
	Mezzanine Jr.	118	115	5	5
	Mezzanine	29	27	0	0
	Senior	1	1	0	0

Table 3: CDO spreads: Poisson approximation vs. Binomial analytics

Conclusion:
Poisson is reasonable approximation for pools even as small as 50 names

CDO Performance/Accuracy

Algorithmics Incorporated

Accuracy of Poisson Approximation

Heterogeneous pool

Poisson comparison vs 100,000 MC paths

Tranche	Poisson	MC	95% CI
Equity	736	740	[730 750]
Mezzanine Jr.	392	394	[387 401]
Mezzanine	285	285	[281 290]
Senior	110	110	[106 112]
Super-Senior	3	3	[3 3]

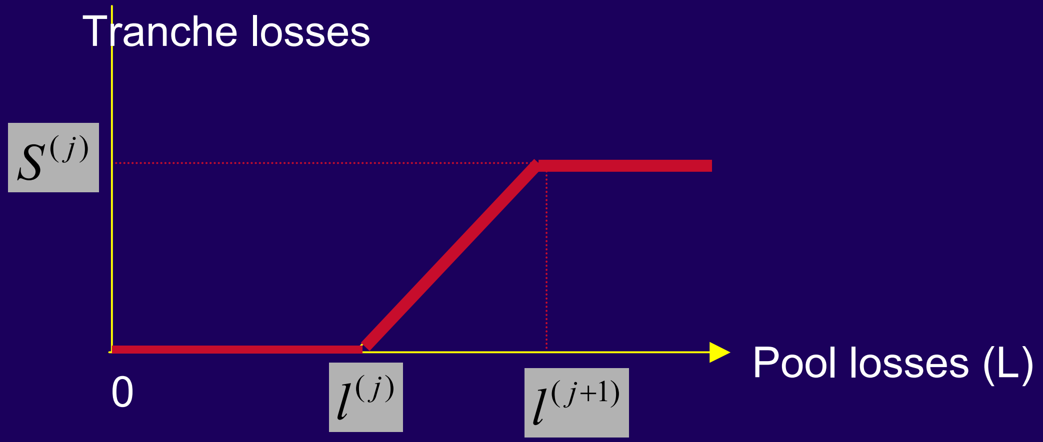
Table 8: Tranche spreads (bp): MC vs. Poisson approximation.

Poisson results always contained within 95% MC conf intervals based on 1

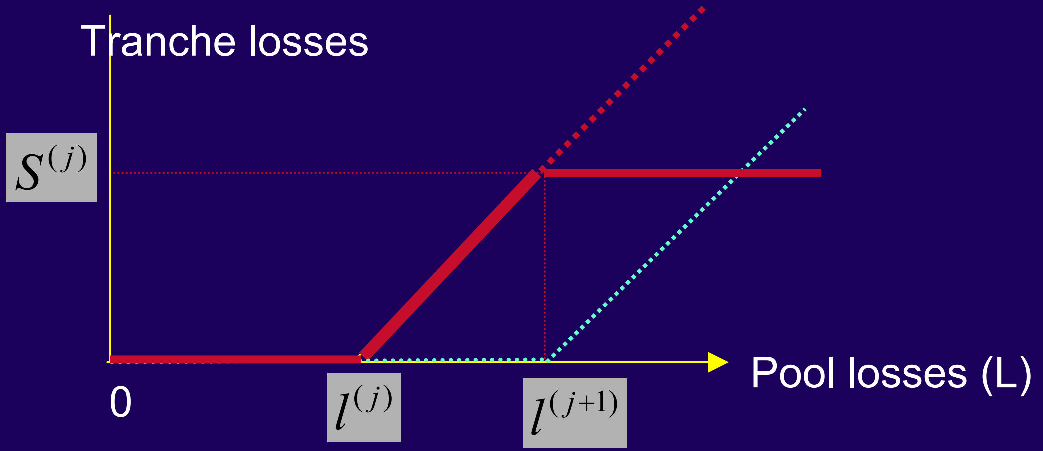
Conclusion: Poisson approximation is as reliable as MC pricing with 100,000 paths.

Algorithmics Incorporated

Tranche Losses



$$L^{(j)} = (L - l(j))^{+} - (L - l(j) - S(j))^{+}$$



CDO Performance – Algo

Algorithmics Incorporated

1) HW recursive approach is faster than HW bucketed convolution approach. This approach was implemented along with the following **“enhancements”:**

- Modified recursion formula to remove instability problems with the suggested approach
- Gauss-Hermite integration points (for credit driver) replaced with Willow points thus only requiring 14 points for integration (vs upwards of 30)
- Pool loss distributions only calculated once and re-used for several tranches using same pool

2) Compound Poisson approximation. This approach will be faster than approach above in cases where:

- Only 1 or 2 tranches are owned (accounting for small % of entire pool)
- The dispersion of assumed recovery rates in the pool are low

3) Standard Monte Carlo pricing – for those who have time to kill (and does nottrust analytics)

CDO Performance

Assuming the number of names in the pool is fixed, performance will depend on 3 dimensions:

- •Number of homogeneous pools
- •Dispersion of recovery rates
- •[illegible]

Performance Numbers for CDOs

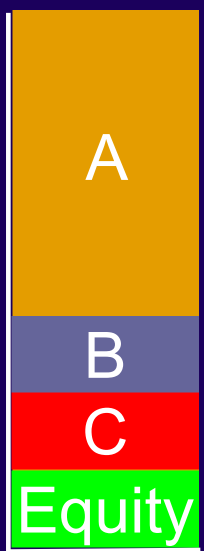
100 Names in the Pool.
100 single step scenarios.
Monte Carlo pricing (100k paths) takes **585s**
All times measured in seconds.
Times will scale linearly by # of scenarios and time steps.

	1 Tranche (6%)				4 Tranches (12.125%)			
	High Dispersion		Low Dispersion		High Dispersion		Low Dispersion	
	EHW	Algo	EHW	Algo	EHW	Algo	EHW	Algo
1 Pool (Homogeneous)	NA	NA	9.4	5.1	NA	NA	9.5	14.1
2 Pools	10.0	23.9	16.2	6.9	10.1	41.8	16.3	17.6
3 Pools	10.8	23.1	18.2	9.0	10.9	41.1	19.3	19.6
4 Pools	20.3	46.2	25.2	11.3	20.4	75.6	25.4	23.1
5 Pools	23.7	46.7	27.0	16.7	23.8	75.9	27.3	31.2
6 Pools	25.9	45.4	30.3	25.8	26.0	71.2	30.4	43.5
7 Pools	28.6	45.5	NA	NA	28.7	71.0	NA	NA

Summary:

- • Compound Poisson process will be faster in cases of single tranches with low recovery rate dispersion
- •EHW/Algo algorithms will be anywhere between 20x-100x faster than MC pricing!!!!!!

Heterogeneous Pool



Query:

What if someone does own all the tranches in the pool. Is there still a benefit to the Algo approach?

Response:

YES! Due to the fact that:

$$E(PoolLosses)_i = \sum_{j=1}^{m} E(L_i^{(j)})$$

we never need to integrate for the entire pool distribution. We can calculate the expected tranche losses for Equity, B and C and back out A's expected tranche losses through the relationship above.

In fact, if we ever face a tranche that covers more than half the pool, it makes sense to apply the Algo Approach to the complementary tranche and back out the desired expected tranche losses since the complement will be faster to

Future Directions

Algorithmics Incorporated

1. Dynamic correlation model
2. Robust calibration algorithm
3. CDO of CDOs (valuation, calibration, sensitivitie

Part I

Loss in Translation

The End