

Optimizing the Compression Cycle: Algorithms for Multilateral Netting in OTC Derivatives Markets

Dominic O’Kane
EDHEC Risk Institute,
393-400 Promenade des Anglais,
Nice Cedex 3,
France
dominic.okane@edhec-risk.com

First Version: 6 February 2013

May 29, 2014

Abstract

Concerns about systemic credit risk in the financial system due to the OTC derivatives market has encouraged the use of counterparty credit risk mitigation techniques, including the use of compression. In compression, market participants share position information via a third party company, TriOptima, which then proposes a set of trades which will ‘compress’ their multilateral exposures. The algorithm used by TriOptima is not public. In this paper we propose and analyse a set of optimal compression algorithms for fungible derivatives. We find that they all perform extremely well across a range of criteria and we discuss their relative attributes. Our focus is on the CDS market, although the methods analyzed here can be applied to other OTC derivative markets where fungibility is high. We argue that compression is an effective counterparty risk mitigation technique if done optimally that should be encouraged by regulators, especially as we show that the benefits increase dramatically with the number of participants.

1 Introduction

The Global Financial Crisis (GFC) of 2008-9 raised fears of a systemic crash in the financial system due to the bankruptcy of one or more large dealers, and the subsequent impact this could have on the rest of the dealer community, all connected via over \$692¹ trillion of over-the-counter (OTC) derivative contracts. Although this fear was not realised, as a precaution legislators have brought in reforms designed to enhance transparency and reduce counterparty exposure in the OTC derivatives market.

To reduce counterparty risk, the Dodd-Frank law in the US has required that all standardised OTC derivatives be cleared via a central counterparty (CCP) by the end of 2012. The impact of this on systemic counterparty risk has been explored in Duffie and Zhu [2011] and Cont and Moussa [2010]. According to the June 2012 BIS report, over \$2.6 trillion of single-name CDS and CDS indices have now been assigned to CCPs. However this is still less than 10% of a credit derivatives market with a total outstanding notional of \$26.9 trillion.

Another mechanism for reducing the size of the OTC derivatives market is the compression cycle. This is a practice which began in 2006, and is run by the Swedish company TriOptima. Although it is applied to both the interest rate swaps market and the credit derivatives market, our focus here is on the latter. The compression cycle begins with a set of market participants each voluntarily submitting position information and market value information to TriOptima on a confidential basis. Typically, the number of participants is around 20 and these consist mostly of the large dealers and some of the larger buy side institutions. Some dealers centralise their participation and submit as a single entity, while others do so only at a desk level. TriOptima collect this data and, using an algorithm² propose to the participants a package of trades which, if executed, will reduce the outstanding amount of market risk positions. The new trades will also preserve the initial market risk exposure (measured in notional or other terms) of each participant. Participants may also impose additional constraints such as excluding certain counterparties. They may also limit their final position sizes. The whole compression process is usually completed within 3-5 days. There will be some fee associated with the compression which will be paid to TriOptima by each participant and will be linked to the amount of compression achieved.

The frequency of the compression cycle depends on the reference credit of the CDS contract. It occurs monthly for European corporate-linked single-name CDS. The main Emerging Market corporate-linked CDS are also compressed monthly, although those linked to EMEA and LATAM credits are only compressed every two months. CDS tranches linked to the main CDX IG and iTraxx Main indices are compressed quarterly, with European sovereign and Japanese and Asian

¹This is the amount outstanding of over-the-counter (OTC) derivatives in June 2013. See BIS [2013] for details.

²This algorithm is proprietary. Although there is an algorithm described in Brouwer [2012], TriOptima inform us in Sjöberg [2013] that they have never used this algorithm. We do not know what they do.

index and single-names compressed semi-annually.

The compression cycle has become another stage in the settlement process of credit events. It takes place just prior to the CDS auction process and its purpose is to reduce the number of participants in the CDS auction, the number of contracts which need to be settled, and the quantity of physical assets which are to be delivered. The aim is to make the CDS auction easier to manage by reducing the number of trades to be settled.

According to TriOptima, it has over the past several years eliminated over \$77 trillion in CDS notional, of which \$30.2 trillion was eliminated in 2008, \$14.5 trillion was eliminated in 2009 and \$8.5 trillion was eliminated in 2010. This has fallen to \$3.5 trillion in 2012 suggesting that many of the legacy positions have now been compressed. Compression has also been applied to the interest rate swap market where the process has been adapted to deal with the much lower fungibility of interest rate swaps compare to CDS.

Indeed, one of the main obstacles to the success of any market compression process is the fungibility of the derivative product. Within the CDS index market, fungibility is high because CDS indices mature on one of the four ‘CDS dates’ which fall on the 20th of March, June, September and December, and because each series has the same coupon. This is now also the case in the single-name CDS market - in 2003 the market switched from daily moving maturity dates to these CDS dates, and following the 2009 ‘CDS Big-Bang’, the CDS market re-couponed all CDS contracts so that all contracts of a given issuer have the same coupon. This means that the exact same 5Y CDS contract continues to be traded for a period of 3 months at which point liquidity flows to a new 5Y contract.

Year	Fraction of Outstanding Notional (%)
2013	24.53
2014	18.98
2015	14.42
2016	16.25
2017	18.11
2018+	7.71

Table 1: Market breakdown by maturity date. Source: DTCC

In terms of maturity dates, a snapshot of all CDS contracts settled at the DTCC in early 2013, and shown in Table 1, shows that only 7.71% of contracts have a maturity in excess of 5 years. This tells us that 92.3% of the CDS market is made up by 20 different contracts per issuer. According to DTCC data, across the top 1000 issuers (reference entities) there are over 1.94 million CDS contracts. Their total gross notional is \$13.5 trillion. If we look at just the top 100 issuers (this includes sovereigns and corporates) only, we see that the average gross exposure per reference entity (issuer) is \$53 billion while the average net exposure is \$3.56 bn. The average

number of contracts per reference entity is 5,134. This means that across the top 100 issuers, the average contract size is just over \$10.3 million. If we assume that contracts are equally weighted across 20 different maturity dates, then the average number of fungible contracts per issuer is about 250, with an average total gross notional of \$2.65bn and an average net notional of \$178m.

1.1 CDS Market Structure

The CDS market can be split into two classes of participant: financials and non-financials. Financials are mainly dealers while non-financials are their customers such as investment managers, insurance companies, bank loan books and hedge funds. Dealers tend to hedge their risks while customers assume risk for some level of return.

Dealers trade frequently and with each other as they hedge their credit risk exposures. This is done in order to dynamically hedge existing trades which include exotic structures such as synthetic CDOs. It also occurs when a trade is novated by a non-dealer with a dealer. The dealer will try to find another customer to take the other side of the trade and to so remove the underlying risk. If this is not possible, the dealer will hedge with another dealer and so the risk may have to be passed along a chain of dealers before it can be placed in the non-dealer segment of the market. This chain of counterparty exposures is one reason why the inter-dealer market is so much larger than the dealer-customer market. Table 2 confirms that there are generally many more trades between dealers than between dealers and non-financial counterparties. It shows that the interdealer CDS market consists of \$9.3 trillion of trade notionals at the end of January 2013. This compares to just \$2.1 trillion between dealers and non-financials.

Buyer Type	Seller Type	
	Dealer	Non-Dealer
	Dealer	Non-Dealer
	\$9,311,633m (1,471,997)	\$2,102,972m (236,178)
	\$2,087,350m (237,014)	\$22,332m (2,582)

Table 2: The structure of the single-name CDS credit market in January 2013. The dollar number is the total gross notional. The number in brackets is the number of contracts. Source:DTCC

Since interdealer trades account for most of the CDS positions at the DTCC, our analysis here will focus here on modelling this specific section of the OTC derivatives market. We will assume that the market consists of 5 to 20 main dealers who are all able to trade with each other. Indeed, according to a Fitch 2009 survey Ratings [2009], the top 10 banks make up an estimated 67% of CDS trading.

While the compression cycle has been designed to enable the netting of multilateral trades, it can also net bilateral trades. It may seem unnecessary for counterparties to involve a third-party

in a purely bilateral netting process, such bilateral positions can build up over time for various reasons including (i) the banks may disagree about the tear-up value of the position, (ii) the bank which has the negative mark to market side of the trade may not wish to fund the cash needed to effect the unwind payment, and (iii) the bank which initiates the tear-up may expect a worse price than if it were to enter into an offsetting position with another dealer. Since many of these issues are solved by the TriOptima process, it makes sense for bilateral trades to be included too. However, in the following algorithms we will assume that all bilateral netting has already been achieved and so we only consider the benefits gained from multilateral netting.

1.2 The Credit Default Swap contract

The methodology for derivative trade compression described here is fairly generic and so applies to most derivative contracts. However we will focus here on the credit default swap (CDS) because it is perceived to have a greater counterparty risk than many other contracts. This is because the nature of credit is one of sudden, unexpected, large losses. In a complex network of such contracts, these sudden large losses can in theory percolate through the system triggering a potential cascade of counterparty failures.

A CDS is a bilateral contract in which one party, a protection seller, provides default protection to the other party, the protection buyer, in the event of the default of a reference credit (a company or sovereign). They do this in return for a regular coupon fee. The notional amount of protection bought is F . This protection lasts for some agreed time period and the coupon is fixed at the initiation of the contract. If there is a credit event (a bankruptcy, failure to pay or restructuring) of the reference entity, the protection seller must pay the protection buyer an amount $F(1 - R)$ where R is the recovery rate of the reference credit³.

During the life of the CDS, a long protection contract bought by party A from B has some value V_{AB} known as its mark to market. The symmetry of a derivative position means that the value of the contract to the protection seller B is $V_{BA} = -V_{AB}$. The valuation of CDS has been described in detail in O’Kane [2008] and is not relevant to this work except to state that V_{AB} is directly proportional to the notional of the contract F .

2 The Modelling Framework

We suppose that we have a CDS market consisting of N counterparties. We consider just one fungible contract, a CDS on a specific reference entity and with a specific CDS maturity date. We assume that each counterparty $i = 1, \dots, N$ has bought protection with a notional of C_{ij}

³This is determined via an auction procedure following the protocols designed by the ISDA (see www.isda.com).

	CP1	CP2	CP3	CP4	CP5	CP6	CP7	CP8	CP9	CP10	h_i
CP1		-70	115	30	65	55	20	-5	-15	-45	150
CP2	70		-60	70	85	65	-55		40	-10	205
CP3	-115	60		50	110	-80	35	80	20	100	260
CP4	-30	-70	-50		-155	5	30	-95	-50	130	-285
CP5	-65	-85	-110	155		-45	30	-65	30	-30	-185
CP6	-55	-65	80	-5	45		-15	-20	70	30	65
CP7	-20	55	-35	-30	-30	15		-25	-35	-115	-220
CP8	5		-80	95	65	20	25		-45	40	125
CP9	15	-40	-20	50	-30	-70	35	45		65	50
CP10	45	10	-100	-130	30	-30	115	-40	-65		-165

Table 3: Example initial configuration of $N=10$ counterparty exposures. The L_1 exposure is \$53.4m and the L_2 exposure is \$64.0m. 2% of the off-diagonal exposures equal zero and the GCD is \$5m.

on this reference entity to this maturity date from every other counterparty $j = 1, \dots, N$. Note that in practice a counterparty i may have several contracts with counterparty j on the same contract in which case C_{ij} is the net position. We can also view this net position as counterparty j buying $-C_{ij}$ of protection from counterparty i . The matrix C is therefore anti-symmetric, i.e. $C_{ij} = -C_{ji}$. Counterparties do not trade with themselves so $C_{ii} = 0$. The net long protection of each counterparty i is given by $h_i = \sum_{j=1}^N C_{ij}$.

Our aim is to determine a set of optimal counterparty exposures \hat{C}_{ij} which reduce the overall counterparty risk of the market while ensuring that the risk exposure for each counterparty is preserved, i.e. $h_i = \sum_{j=1}^N \hat{C}_{ij}$. The initial set of counterparty exposures is given by C_{ij} and the post-compression set of exposures is \hat{C}_{ij} . Hence after any compression cycle, each pair of counterparties need to execute the following trade: party i must buy a notional of $(\hat{C}_{ij} - C_{ij})$ of protection from counterparty j . If the current mid-market upfront cost of protection is V_{Mid} , then each trade for counterparty i will have initial cost $(\hat{C}_{ij} - C_{ij}) \cdot V_{Mid}$. Overall however, the counterparty should not see any change in its net exposure since the linear constraints mean that $\sum_{j=1}^N (\hat{C}_{ij} - C_{ij}) \cdot V_{Mid} = V_{Mid} \cdot \sum_{j=1}^N (\hat{C}_{ij} - C_{ij}) = 0$ for $i = 1, \dots, N$. If all of the trades (both purchases and sales of protection) are executed at mid-market levels, there should be no cost involved in the compression as all of these values will net out to zero. Implicit in this assumption is that counterparties do not adjust prices depending on who they trade with, e.g. CVA adjustments are not included or are small due to mitigation effects such as collateralisation.

It is important to take into account the fact that most CDS trades are done in units of \$5m to \$10m. Therefore a compression algorithm which leaves a counterparty with small positions, or positions which are not round multiples of \$5m, may make this position less liquid and harder to unwind subsequently. Also, completely removing a position eliminates any need for managing collateral posting, regulatory capital charges, risk monitoring and payment processing with its

associated costs. For this reason, we favour algorithms which remove positions completely rather than those which tend to simply reduce position sizes. In the following, we will set out three different algorithmic approaches to compression. In each approach, we use the counterparty exposures in Table 3 as a test case. These were generated randomly and have a greatest common divisor (GCD) of \$5m.

2.1 Compression Metrics

To assess the quality of the compression, we need measures for the change in the counterparty exposure across the market caused by the compression. We therefore define the entry-wise exposure matrix norm

$$\|L(C, N)\|_p = \left(\frac{2}{N(N-1)} \sum_{i=1}^N \sum_{j=i+1}^N |C_{ij}|^p \right)^{1/p}.$$

Based on this we have the mean absolute exposure

$$\|L(C, N)\|_1 = \frac{2}{N(N-1)} \sum_{i=1}^N \sum_{j=i+1}^N |C_{ij}|$$

which we call L_1 for short, and

$$\|L(C, N)\|_2 = \sqrt{\frac{2}{N(N-1)} \sum_{i=1}^N \sum_{j=i+1}^N |C_{ij}|^2}.$$

which is the root mean square exposure or L_2 . We also find it useful to define

$$F_x(C, N) = \frac{2}{N(N-1)} \sum_{i=1}^N \sum_{j=i+1}^N \mathbb{1}_{C_{ij}=n_{ij}x}$$

where n_{ij} is an integer. The fraction of exposures which equal zero is given by $F_0(C, N)$ while the fraction of exposures which are integer multiples of some amount is given by $F_u(C, N)$. With these quantities, we can then calculate the compression ratio. For example the compression ratio of $\|L(C, N)\|_p$ with N counterparties is given by

$$CR_{p,N} = \frac{2}{N(N-1)} \frac{\|L(\hat{C}, N)\|_p}{\|L(C, N)\|_p}.$$

A compression ratio less than one is a sign of a successful compression.

3 Compression Algorithms

3.1 A Loop Compressing Algorithm

In their US patent application, Brouwer [2012], TriOptima present an algorithm for netting counterparty exposures. TriOptima have recently informed us that this is not the algorithm used for compression, and that they have never used this algorithm. Nevertheless we use this algorithm as our starting point, especially as it is an example of a class of compression algorithms which work well even if we will show that they are not optimal.

Stage one of the algorithm is the enumeration all of the closed loops of claims. We define a claim by party i on party j as a positive value for the counterparty exposure, $C_{ij} > 0$. A closed loop of claims is a sequence of claims which starts and ends at the same counterparty and does not visit any counterparty more than once. Cyclic permutations of a loop are considered to be the same loop. Once all of the closed loops of claims have been detected, stage two of the algorithm is to net each loop which is done by subtracting the smallest exposure a (all the exposures are positive claims so $a > 0$ by definition) in the loop from each exposure in the loop.

As an example, we consider the initial configuration in Table 3 and choose CP2 as our starting counterparty. By scanning horizontally we see that CP2 has a claim on CP4 of 70. We then descend to row CP4 and see that CP4 has a claim on CP6 of 5. We then descend to row CP6 to see that CP6 has a claim on CP5 of 45. We then go to row CP5 to see that CP5 has a claim on CP7 of 30. And we see on row CP7, that CP7 has a claim on CP2 of 55. We therefore have a loop of 5 distinct parties with the sequence of counterparties $\{\text{CP2}, \text{CP4}, \text{CP6}, \text{CP5}, \text{CP7}, \text{CP2}\}$. To compress this loop, the algorithm determines the smallest claim, which in this case is 5, and then subtracts this from each of the exposures in the loop.

Making this adjustment preserves the linear exposure constraints. To show this we can write any loop of claims starting and ending at counterparty i as a sequence of exposures $\{C_{ij}, C_{jk}, \dots, C_{wi}\}$. The index of each counterparty in the loop appears twice, once as a first index and once as a second index. If we subtract any amount a from all of the exposures in the loop, we will have new exposures $\{C_{ij} - a, C_{jk} - a, \dots, C_{wi} - a\}$. What effect does this have on the net exposures? First we consider $h_n = \sum_{q=1}^N C_{nq}$ where $n \notin \{i, j, \dots, w\}$. In this case none of the exposures that experience compression ever enter into the summation since to do so they would need one index to be in $\{i, j, \dots, w\}$. As a result, the compression cannot change the net exposure h_n . Second, we consider the case of $n \in \{i, j, \dots, w\}$. We will find exposures to the pair of counterparties o and m to which

	CP1	CP2	CP3	CP4	CP5	CP6	CP7	CP8	CP9	CP10	h_i
CP1				10	65	55	20				150
CP2				40	60	65			40		205
CP3					85	-50	35	70	20	100	260
CP4	-10	-40			-125			-95	-50	35	-285
CP5	-65	-60	-85	125		-40	5	-65			-185
CP6	-55	-65	50		40				70	25	65
CP7	-20		-35		-5			-25	-35	-100	-220
CP8			-70	95	65		25		-30	40	125
CP9		-40	-20	50		-70	35	30		65	50
CP10			-100	-35		-25	100	-40	-65		-165

Table 4: Results of loop compression algorithm applied to the initial configuration in Table 3. The L_1 notional is \$33.8m and the L_2 notional is \$47.92m. The percentage of off-diagonal zeros is 36% and the GCD is \$5m.

counterparty n connects in the triplet $\{m, n, o\}$. This gives

$$h_n = \sum_{q=1}^N \hat{C}_{nq} = \dots + \hat{C}_{nm} + \hat{C}_{no} + \dots = \dots - (C_{mn} - a) + (C_{no} - a) + \dots = \sum_{q=1}^N C_{nq}.$$

We conclude that the net exposure h_n is not changed by the compression. Note also that the value of a is general, a fact that we will exploit later when we consider locally optimal compression algorithms. In this loop algorithm, $a = \min[C_{ij}, C_{jk}, \dots, C_{wi}]$ where all $C_{ij}, C_{jk}, \dots, C_{wi} > 0$.

The problem of finding all of the closed loops in a network of claims is a well-studied problem in graph theory where it is described as the problem of finding all of the elementary circuits of a simple directed graph as examined by Johnson [1975]. The approach described in TriOptima's patent is a variation on the Depth-First Search (DFS) algorithm described and developed in Tarjan [1972].

In Table 4 we show the results of the loop compression algorithm applied to the initial configuration in Table 3. The algorithm has reduced the L_1 notional from \$53.4m to \$33.3m while the L_2 notional has fallen from \$64.0m to \$48.9m. We note that the number of zeros in the matrix has increased from 2 to 32 (we exclude the diagonals) showing that a large number of counterparty exposures have been completely eliminated. The GCD of \$5m has been preserved. The algorithm appears to work well.

One limitation of the loop algorithm is that it only changes non-zero exposures. This is because the netting only applies to closed loops and a closed loop cannot by definition have a claim of zero in it. This means that netting cannot create additional closed loops. As a result the compression can be broken down into two stages, with stage one being the enumeration of the entire list of closed loops, and stage two being the netting of each of these loops. From an algorithm design perspective this is attractive as it permits greater modularity. But there is a downside. The

number of closed loops grows more than exponentially with the number of counterparties. We conducted numerical experiments⁴ to count the number of closed loops in a network in which each counterparty is connected to every other counterparty. We found that for $N = 10$, the maximum number of closed loops is over 7,400, with an average number around 3,400. We must therefore be mindful of the very large number of loops that may be present for higher values of N . This number of loops can become prohibitive in terms of memory and computational time. Indeed, it may actually be more efficient to simply generate and compress closed loops one at a time. Each loop that is compressed should quickly reduce the number of closed loops which remain. However even if we speed up the loop detection algorithm, this will not improve the degree of compression, it will just decrease the time taken to perform it.

We conclude that the compression algorithm described in Brouwer [2012] does reduce counterparty exposures. However, depending on the degree of connectivity between counterparties and the value of N , it may become a very expensive algorithm from a computational time perspective. It is also not clear how to incorporate additional equality or inequality constraints into the compression. It is also not an optimisation approach in any sense, just a heuristic algorithm which does compress. We believe that it should be possible to improve on the loop compression algorithm if we apply optimisation techniques to determine the trades which result in the greatest overall decrease in counterparty exposures.

3.2 Optimal Method I: Minimising the L_1 Exposure

We formulate the minimization of the counterparty exposures as a linear programming (LP) problem. The objective function which is to be minimised is $\sum_{ij} |C_{ij}|$ which is the sum over the absolute values of the counterparty exposures. This is equivalent to minimising $\|L(C)\|_1$. We have the constraints $\sum_{j=i}^N \hat{C}_{ij} = h_i$ for $i = 1, \dots, N$. This can be written as an LP problem by introducing two variables C_{ij}^+ and C_{ij}^- where $C_{ij} = C_{ij}^+ - C_{ij}^-$. We also consider just C_{ij}^+ and C_{ij}^- with $i < j$ and set $C_{ij}^+ = -C_{ji}^+$ and $C_{ij}^- = -C_{ji}^-$. The problem of minimising the sum of the absolute values then becomes that of finding the minimum of the linear sum

$$\Omega(C) = \sum_{i=1}^N \sum_{j=1}^N (C_{ij}^+ + C_{ij}^-)$$

⁴We used Monte-Carlo methods to generate random matrices \mathbf{C} and used the algorithm in Johnson [1975] to enumerate all of the closed loops.

	CP1	CP2	CP3	CP4	CP5	CP6	CP7	CP8	CP9	CP10	h_i
CP1										150	150
CP2				45	145					15	205
CP3					40		220				260
CP4		-45				-65		-125	-50		-285
CP5		-145	-40								-185
CP6				65							65
CP7			-220								-220
CP8				125							125
CP9				50							50
CP10	-150	-15									-165

Table 5: Results of global L_1 exposure minimisation applied to the initial configuration in Table 3. Mean absolute notional: \$19.0m RMS notional: \$51.23m. The percentage of off-diagonal zeros is 80% and the GCD is \$5m.

subject to

$$\sum_{j=1}^N (C_{ij}^+ - C_{ij}^-) = h_i \text{ for } i = 1, \dots, N, \quad C_{ij}^+ \geq 0, \quad C_{ij}^- \geq 0.$$

We were able to solve this LP problem using a numerical solver package⁵. Using the same initial configuration as Table 3, we find the configuration in Table 5 to be optimal. We note that this method minimises the L_1 exposure which falls from \$53.4m to \$19.0m. It also preserves the GCD of \$5m. It also reduces the number of non-zero exposures with exactly 80% of the exposures now equal to zero compared to 2% initially.

3.3 Optimal Method II: Minimising the L_2 Exposure

A second approach to minimising the market outstanding notional is to find the configuration \hat{C}_{ij} which minimises the objective function

$$\Omega(C) = \sum_{i=1}^N \sum_{j=i+1}^N C_{ij}^2 \quad (1)$$

This consists of $N(N-1)/2$ distinct counterparty exposures subject to the N constraints $h_i = \sum_{j=1}^N \hat{C}_{ij}$, with the requirements that $\hat{C}_{ij} = -\hat{C}_{ji}$ and $\hat{C}_{ii} = 0$. This is equivalent to minimising L_2 . Note that since $\sum_{j=1}^N h_i = 0$ there are actually just $N-1$ linearly independent constraints. By using a quadratic objective function rather than a linear one, we effectively apply a greater cost penalty to large exposures and so the optimal solution should prefer lower absolute exposures

⁵We used LPSolve. This open source project can be found at <http://lpsolve.sourceforge.net>

sizes compared to the L_1 exposure minimisation.

The structure of the problem means that we have $N(N-1)/2$ variables with $(N-1)$ linearly independent constraints. Mathematically, this optimisation becomes a problem with $(N-1)(N-2)/2$ independent variables. It can be solved exactly by introducing N Lagrange multipliers λ_i to enforce the N risk conservation constraints. We have

$$\Omega(C) = \sum_{i=1}^N \sum_{j=i+1}^N C_{ij}^2 + \sum_{i=1}^N \lambda_i \left(h_i - \sum_{j=1}^N C_{ij} \right). \quad (2)$$

We differentiate this objective function with respect to C_{mn} to give

$$\frac{\partial \Omega(C)}{\partial C_{mn}} = 2C_{mn} + \lambda_m - \lambda_n.$$

Setting this equal to zero, we have

$$\hat{C}_{mn} = \frac{\lambda_n - \lambda_m}{2}.$$

We can then show that

$$\begin{aligned} h_m &= \sum_{j=1}^N \hat{C}_{mj} = \frac{1}{2} \sum_{j=1}^N \lambda_j - \frac{N}{2} \lambda_m \\ h_n &= \sum_{j=1}^N \hat{C}_{nj} = \frac{1}{2} \sum_{j=1}^N \lambda_j - \frac{N}{2} \lambda_n. \end{aligned}$$

Subtracting we have

$$h_m - h_n = \frac{N(\lambda_m - \lambda_n)}{2}.$$

The optimal exposure is therefore given by

$$\hat{C}_{mn} = \frac{h_m - h_n}{N}. \quad (3)$$

This solution is remarkably simple and possesses all of the required properties including $\hat{C}_{mn} = -\hat{C}_{nm}$ and $C_{mm} = 0$. It also satisfies the linear constraints $\sum_{n=1}^N \hat{C}_{mn} = (1/N) \sum_{n=1}^N (h_m - h_n) = h_m$. We also see that the resulting matrix will not be sparse unless we have many pairs (m, n) for which $h_m = h_n$. The division by the number of counterparties also tells us that the optimisation

	CP1	CP2	CP3	CP4	CP5	CP6	CP7	CP8	CP9	CP10	h_i
CP1		-5.5	-11.0	43.5	33.5	8.5	37.0	2.5	10.0	31.5	150
CP2	5.5		-5.5	49.0	39.0	14.0	42.5	8.0	15.5	37.0	205
CP3	11.0	5.5		54.5	44.5	19.5	48.0	13.5	21.0	42.5	260
CP4	-43.5	-49.0	-54.5		-10.0	-35.0	-6.5	-41.0	-33.5	-12.0	-285
CP5	-33.5	-39.0	-44.5	10.0		-25.0	3.5	-31.0	-23.5	-2.0	-185
CP6	-8.5	-14.0	-19.5	35.0	25.0		28.5	-6.0	1.5	23.0	65
CP7	-37.0	-42.5	-48.0	6.5	-3.5	-28.5		-34.5	-27.0	-5.5	-220
CP8	-2.5	-8.0	-13.5	41.0	31.0	6.0	34.5		7.5	29.0	125
CP9	-10.0	-15.5	-21.0	33.5	23.5	-1.5	27.0	-7.5		21.5	50
CP10	-31.5	-37.0	-42.5	12.0	2.0	-23.0	5.5	-29.0	-21.5		-165

Table 6: The L_2 exposure minimisation given the initial configuration in Table 3. The L_1 exposure is \$23.2m, the L_2 exposure is \$27.7m, there are no off-diagonal zero exposures and the GCD is \$0.50m.

will tend to have few zero exposures but that size of most exposures will be low. This solution also makes it clear that this algorithm produces optimal configurations with counterparty exposures that will, in the worst case, have a GCD which will be initial GCD divided by the number of counterparties.

Table 6 shows the optimal solution obtained by applying this optimisation approach to the initial configuration presented in Table 3. The L_1 exposure has been reduced to \$23.2m while the L_2 exposure has been reduced from \$64m to \$27.7m. As expected, the solution consists of many small exposures with no zeros. This compares to the L_1 exposure minimisation solution in Table 5 which has many zeros, but a number of large absolute exposures. Initially, the GCD of the exposures was \$5.0m. After the L_2 exposure compression the resulting counterparty exposures have a GCD of \$0.50m which agrees with our observation above that in the worst case the new GCD will be the initial one divided by the number of counterparties.

As we have a closed form solution for \hat{C}_{ij} we can write the value of the objective function in closed form also. We have

$$\Omega(\hat{C}) = \sum_{i=1}^N \sum_{j=i+1}^N \hat{C}_{ij}^2 = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=i+1}^N (h_i - h_j)^2 = \frac{1}{N} \sum_{i=1}^N h_i^2 \quad (4)$$

where we used the fact that $\sum_{i=1}^N h_i = 0$.

3.4 Optimal Method III: Semi-Local Minimisation

All of the methods described so far have been global optimisations which rely upon the algorithm having a knowledge of all of the counterparty exposures. We now examine a more local algorithm which could be applied successively to small groups of counterparties, thereby avoiding

the organisation challenges of coordinating the activities and obtaining the agreement of many counterparties over a compression cycle⁶. We call it ‘semi-local’ as we refer to the next step of compression beyond simple bilateral netting which involves having information about three counterparties thereby enabling a trilateral compression.

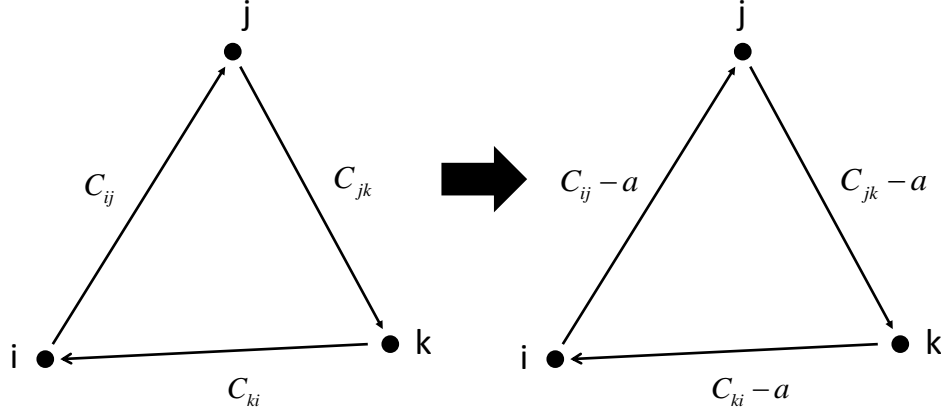


Figure 1: Compression of a 3-counterparty loop by subtracting the amount a from each exposure. The direction of the arrows signifies the order of the indices on loop $\{C_{ij}, C_{jk}, C_{ki}\}$.

The idea is that we iterate over all three-counterparty loops, or ‘triplets’, which we denote by the directed sequence of exposures $\{C_{ij}, C_{jk}, C_{ki}\}$, and then apply a simple rule to compress the exposures. Figure 1 shows the counterparty weights before and after the adjustment. These do not change the value of the net exposure of any of the counterparties. After compression the exposure of counterparty i is still $(C_{ij} - C_{ki})$, that of counterparty j is still $(C_{jk} - C_{ij})$ and that of counterparty k is still $(C_{ki} - C_{jk})$.

We wish to use an optimisation approach, but this time it is semi-local rather than global. We try two approaches. In the first we search for the value of a that minimises L_2 , the summed quadratic exposures of the triplet. Specifically we want the value of a that minimises $(C_{ij} - a)^2 + (C_{jk} - a)^2 + (C_{ki} - a)^2$. The solution is easily found to be

$$a = \frac{C_{ij} + C_{jk} + C_{ki}}{3}. \quad (5)$$

In the second case, we choose to minimise L_1 , the summed absolute exposures of the triplet. In

⁶The compression cycle is *fragile* in the sense that if one party rejects the proposal made by TriOptima, the whole process must fail and be restarted. This event becomes more likely the more counterparties there are involved.

this case we need the value of a that minimises $|C_{ij} - a| + |C_{jk} - a| + |C_{ki} - a|$. This is given by

$$a = \text{median}(C_{ij}, C_{jk}, C_{ki}). \quad (6)$$

This will clearly zero the connection that is the median. We note in passing that the approach of locating only closed loops, and then subtracting the value of the exposure with the smallest absolute size is not optimal in either a quadratic or absolute value sense. Also, we do not require a closed loop of positive claims for these algorithms to work. The signs do not matter. Furthermore, we allow any exposure to change its value, including zero exposures.

Implementation of both algorithms is very simple. We iterate through each of the unique triplets (i, j, k) . For each triplet combination (i, j, k) we apply the compression rule and then pass on to the next triplet. We find that if we allow both algorithms to sweep over all of the triplets between 2 and 5 times, they will converge to a fixed solution. It is possible to understand this by viewing each algorithm as a sequence of minimisations in a 3-dimensional subspace of the full $N(N - 1)/2$ parameter space.

In the L_2 exposure minimisation the objective function is convex so we would expect that the iterations in the subspace converge towards the unique global minimum of the global L_2 objective function. And this is what we find. While converging to the optimal solution the intermediate solution always satisfies all of the linear constraints. This happens by default since this loop netting procedure has no effect on net exposures per counterparty as we proved in section 3.1. However the GCD of the initial and intermediate solutions are not preserved and the intermediate GCD can even be much lower than u/N . This is because the local L_2 minimisation rule in equation 5 involves subtracting a quantity in which we have divided exposure notionals by 3. It is only in the limit of convergence that the u/N lower bound on the GCD is recovered.

The L_1 objective function is not convex and so we would not expect it to converge to the same solution as the global L_1 exposure minimisation. We find that the solution found may be very different in terms of the counterparty exposures. What is interesting is that we find that the solution to which the algorithm converges has an objective function value equal to the value given by the solution found by the global L_1 exposure minimisation. This suggests that there are many local minima with the same value of the objective function. The solution found by this locally optimal method is therefore just as valid as the one found using the global L_1 exposure optimiser. What is also interesting about the solution found is that although it finds a solution with the same value of the L_1 exposure, this solution generally has a lower L_2 exposure. Note also that the GCD is also preserved by the local L_1 minimisation as the updating rule in equation 6 only involves subtracting an existing notional exposures.

We do not show the solution found by the local L_2 exposure minimisation algorithm since it

	CP1	CP2	CP3	CP4	CP5	CP6	CP7	CP8	CP9	CP10	h_i
CP1					10		140				150
CP2				150	45					10	205
CP3				85	30		80			65	260
CP4		-150	-85			-20			-30		-285
CP5	-10	-45	-30					-100			-185
CP6				20						45	65
CP7	-140		-80								-220
CP8					100					25	125
CP9				30						20	50
CP10		-10	-65			-45		-25	-20		-165

Table 7: The local L_1 exposure minimisation given the initial configuration in Table 3. The L_1 exposure is \$19.0m, the L_2 exposure is \$41.5m, 67% of off-diagonal exposures equal zero and the GCD is \$5.0m.

is exactly the same as that found by the global L_2 exposure minimisation shown in Table 6. We show the results of the local L_1 exposure minimisation in Table 7. We see that the L_1 exposure has fallen to \$19.0m. This a considerable improvement on the initial value of \$53.4m. The L_2 exposure has decreased from \$64m to \$41.5m. We also note that 67% of the exposures are zero and the rest are all integer multiples of \$5m. The algorithm is also very fast - a run with $N = 20$ and 5 sweeps takes less than 0.001 seconds⁷.

3.5 Constraints

To be used in a realistic setting, all of the algorithms need to be able to permit the imposition of additional linear equality and inequality constraints. For example, a counterparty i may impose a constraint that it cannot trade with counterparty j . In the linear approach we can enforce this by setting $\hat{C}_{ij} = 0$ and adding it to the list of linear constraints that can be handled by the LP solver. Linear constraints are also straightforward to incorporate into the analytic quadratic approach too. In the local approach, as we loop over the triplets we can simply test whether the constraint is violated by the compression and if it is, not allow it.

Inequality constraints may also be required. For example, counterparty i may also put a lower and upper bound on its exposure with counterparty k , e.g. $-10 < C_{ik} < 50$. In the case of the global L_1 minimisation this can be handled easily by most LP algorithms and is not a problem. For the global L_2 minimisation it would require us to use a quadratic programming routine. Once again, the local approaches can handle inequality constraints by testing to see if the constraint is violated by the compression as we loop over each triplet, and if so to reject it.

⁷This is on an Intel Core i5-2320 CPU running at 3.00GHz with 8GB of RAM.

Compression Method	L_1 Exposure (\$m)	L_2 Exposure (\$m)	Percentage of Zeros F_0 (%)	GCD u (\$m)
None: Pre-compression	53.4	64.0	2	5.0
Loop Compression	33.8	47.9	36	5.0
I - L_1 Minimisation	19.0	51.3	80	5.0
II - L_2 Minimisation	23.2	27.7	0	0.5
III(a) - Local L_1	19.0	41.5	67	5.0
III(b) - Local L_2	23.2	27.7	0	0.5

Table 8: Summary of results for each compression method and one initial configuration of exposures.

4 Results

4.1 Single Scenario Results

In Table 8 we summarise the results of the compression of the initial scenario shown in Table 3. In terms of minimising the L_2 exposure, the quadratic minimisation works best with a minimum value of \$27.7m. This is to be expected given that the objective function is L_2 . However, this benefit is offset by fact that the resulting exposures are not multiples of \$5m. In terms of minimising L_1 , the LP minimisation works best also as expected since the objective function is L_2 . We also find that the local versions of these minimisations do just as well. We note that the local L_1 minimisation finds a solution with the same L_1 exposure as the global optimiser but with a lower L_2 exposure. This is explained by the fact that the fraction of zeros is 67% versus 80% - fewer zeros means that the exposures can be spread over more counterparties so that individual exposures can remain smaller. However, it is not possible to draw firm conclusions about the relative merits of these different compression algorithms based on one scenario. We therefore do a Monte Carlo analysis across a broad range of randomly generated initial configurations in order to compare the different algorithms.

4.2 Statistical Results

We generate $p = 1, \dots, P$ initial counterparty configurations C_{ij}^p by assuming that $C_{ij}^p \sim u \cdot \text{round}(\sigma \cdot g_{ij})$ where $g_{ij} \sim N(0, 1)$ is an independent Gaussian random draw and u is an integer representing the typical trade notional multiple. We set u to \$5m. Parameter σ is not an integer and is used to scale the trade sizes and we set $\sigma = 10$. We only draw C_{ij} for $j > i$ since $C_{ij}^p = -C_{ji}^p$. We examine the L_1 and L_2 compression ratios $CR_1(N)$ and $CR_2(N)$ where

$$CR_p(N) = \mathbb{E} \left[\frac{\|L(\hat{C}, N)\|_p}{\|L(C, N)\|_p} \right].$$

N	$CR_{2,N}$ (%)	$CR_{1,N}$ (%)	$F_0(N)$ (%)	$F_u(N)$ (%)
3	92.54	90.26	12.00	100
5	82.53	73.49	25.80	100
10	70.10	53.27	44.68	100
12	67.03	48.05	50.13	100
15	63.16	42.83	55.22	100

Table 9: Statistics for the loop compression algorithm based on 1,000 scenarios.

N	$CR_{2,N}$ (%)	$CR_{1,N}$ (%)	$F_0(N)$ (%)	$F_u(N)$ (%)
3	85.07	72.06	36.20	100
5	79.01	50.90	61.41	100
10	76.64	33.43	80.62	100
15	75.97	26.94	86.98	100
20	75.78	23.06	90.24	100

Table 10: Statistics for the global L_1 exposure minimisation algorithm based on 1,000 scenarios.

We also examine the fraction of zeros $F_0(N)$ and the fraction of exposures with an initial GCD of u , given by $F_u(N)$ where

$$F_x(N) = \mathbb{E} [F_x(C, N)].$$

The expectation is over realisations of the initial counterparty exposures.

We present first the results for the loop compression algorithm in Table 9. The compression ratios $CR_{1,N}$ and $CR_{2,N}$ fall monotonically with increasing N showing that the compression algorithm is effective and that compression increases with the number of counterparties. For $N = 10$ the L_1 notional is on average 53.27% of its initial value and the L_2 exposure is on average 70.10% of its initial value. The loop compression algorithm also succeeds in zeroing a growing fraction of counterparty exposures, which is what we would expect given that it always zeroes the lowest claim on a loop. We see that more than 50% of the exposures are on average equal to zero for $N \geq 12$. The algorithm also preserves the initial GCD of the exposures which is \$5m.

The results of the global L_1 minimisation are shown in Table 10. We see that the minimisation produces an L_1 exposure which falls quickly with increasing N and more quickly than the loop compression algorithm. The $CR_{2,N}$ which measures the quadratic L_2 exposure does not fall so quickly reflecting the fact that this algorithm prefers a small numbers of large exposures and a large number of zero exposures. This is supported by the observation that the average fraction of zero connections is already above 80% for $N = 10$, and even reaches 90% for $N = 20$. The GCD of the initial exposures is preserved in all cases.

The results of the global L_2 minimisation are shown in Table 11. As with all the algorithms,

N	$\sqrt{2/N}$	$CR_{2,N}$ (%)	$CR_{1,N}$ (%)	$F_0(N)$ (%)	$F_u(N)$ (%)
3	81.65	77.98	82.45	1.90	32.30
5	63.25	61.80	64.54	1.39	20.62
10	44.72	43.70	44.77	0.94	10.09
15	36.51	35.94	36.45	0.74	6.56
20	31.62	31.65	32.03	0.64	4.98

Table 11: Statistics for the global L_2 exposure minimisation based on 1,000 scenarios.

N	$CR_{2,N}$ (%)	$CR_{1,N}$ (%)	$F_0(N)$ (%)	$F_u(N)$ (%)
3	86.12	75.43	31.10	100
5	75.16	51.83	52.70	100
10	62.30	33.72	67.06	100
15	55.36	26.53	72.28	100
20	51.80	23.25	74.78	100

Table 12: Statistics for the locally optimal L_1 exposure algorithm with 1,000 scenarios and 10 sweeps.

these results were averaged over 1,000 random initial configurations of exposures C . The $CR_{2,N}$ which measures the quadratic L_2 exposure falls quickly with increasing N and more quickly than any of the other algorithms. By $N = 10$, the $CR_{2,N}$ is at 43.70%. The number of zero exposures is very low and does not improve with increasing N . This is the nature of the quadratic minimisation which prefers lots of small exposures. The GCD of the initial exposures is reduced below the initial number in all cases.

We can obtain an analytical estimate of the dependence of the compression ratio of the L_2 exposure on the number of counterparties N . In Appendix B we show that the expected compression ratio for the L_2 exposure measure $CR_{2,N} \simeq \sqrt{2/N}$. This formula, shown in the second column of Table 11 agrees well with the Monte Carlo L_2 compression ratio shown in the third column of the table, especially as N increases.

The results of the local L_1 minimisation algorithm are shown in Table 12. These are a sort of hybrid between the global L_1 minimisation and the global L_2 minimisation in the sense that while the L_1 measure is similar, the local algorithm reduces the L_2 exposure by more than the global algorithm. The cost of this is that the local algorithm has fewer zero exposures. This method is fast and we run 1,000 scenarios up to $N = 20$. Given the additive nature of the approach, it is no surprise to see that fraction of the new exposures which are either zero or multiples of \$5m is fixed at 100%. This feature of the starting configuration is always preserved. It performs very well in compressing the L_2 notional and does even better for L_1 notionals. We do not show the results for the L_2 local algorithm since they are the same as those in Table 11.

We can compare the performance of the different algorithms. First, we consider how well they reduce the L_2 notional. Figure 2 plots the L_2 compression ratio as a function of the number of

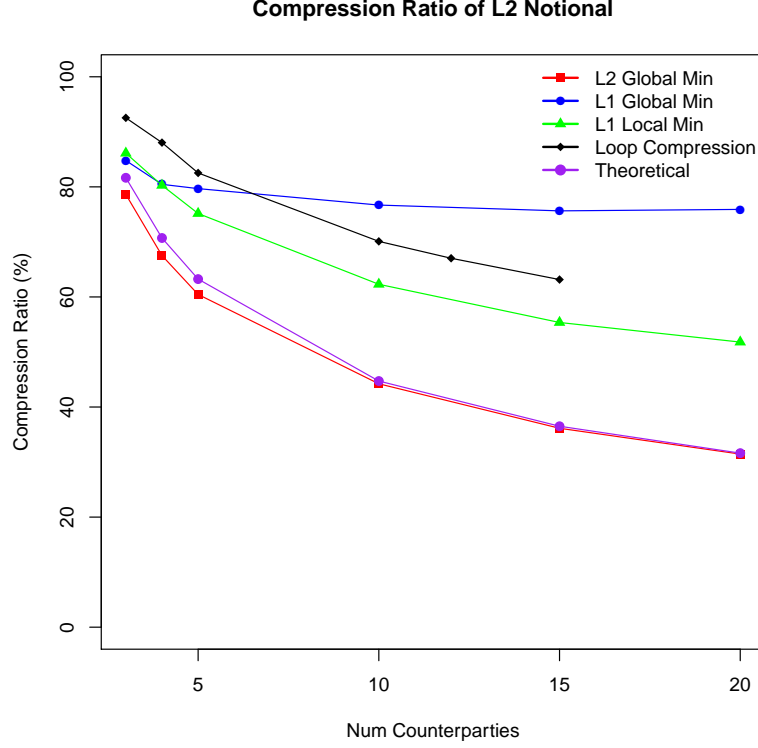


Figure 2: Comparison of the different algorithms in terms of their L_2 compression ratio.

counterparties N . We see that the global L_2 minimisation does best - in theory no other algorithm can do better. The Semi-Local L_1 comes next followed by the Loop Algorithm and then the Global L_1 minimisation. We have also plotted the theoretical estimate of the optimal L_2 compression ratio given by $\sqrt{2/N}$ and see that it agrees well with the global L_2 minimisation result.

Figure 3 shows the L_1 notional compression ratio as a function of the number of counterparties. This time we see that it the global L_1 minimisation algorithm does best and has essentially the same performance as the local L_1 minimisation algorithm. The global L_2 minimisation comes next followed by the loop compression algorithm. In Figure 4 we see the fraction of zeros F_0 after compression as a function of the number of counterparties N . The global L_1 minimisation algorithm does best, setting on average 80% of exposures to be zero when the number of counterparties equals 20. The local L_1 minimisation does not do as well. The loop compression algorithm comes next and is followed by the global L_1 minimisation which sets almost none of the exposures equal to zero after compression.

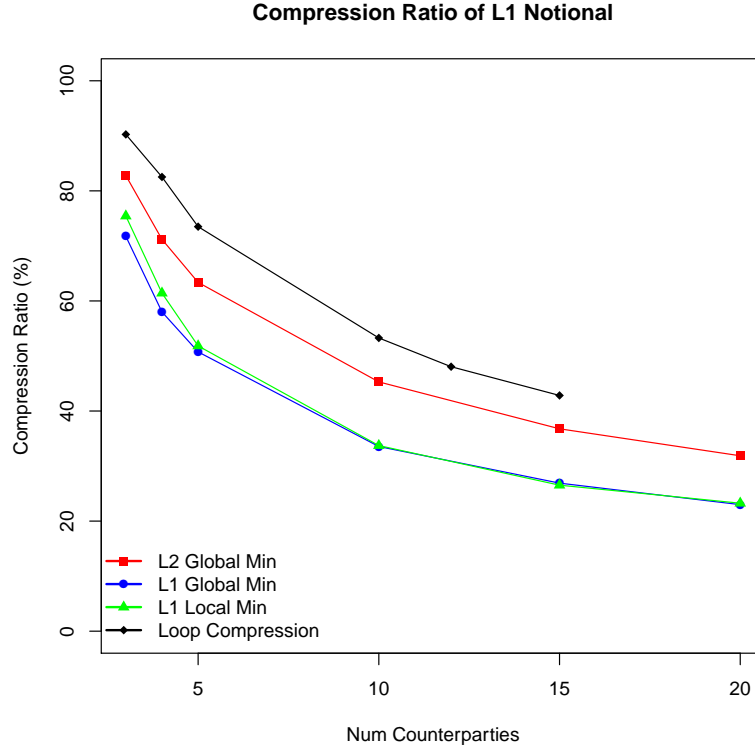


Figure 3: Comparison of the different algorithms in terms of their L_1 compression ratio.

5 Conclusions

We have examined a number of different approaches to the challenge of compressing the counterparty exposure in the OTC derivatives market for CDS. Our aim has been to determine how to maximise the amount of multilateral netting that can be done. All of the algorithms proposed here compress the counterparty exposures but with varying degrees of success.

In terms of pure compression, we favour the global L_1 minimisation algorithm. It is fairly straightforward to implement, fast and permits the incorporation of additional linear equality and inequality constraints. It also preserves the GCD and zeroes many of the exposures. Our only criticism is that it has a tendency to assign small numbers of large exposures which may be deemed to be a concentration of risk. It therefore contrasts with the global L_2 minimisation which has a solution which dislikes large exposures. The disadvantage of this approach is it cuts very few exposures to zero and does not preserve the initial GCD. The analytical solution is interesting and helps us to build insight about the compression process. The local approaches are also interesting. Of these the local L_1 minimisation seems to combine the best of both worlds in the sense that it has an almost identical compression ratio to the global L_1 minimisation with the advantage of

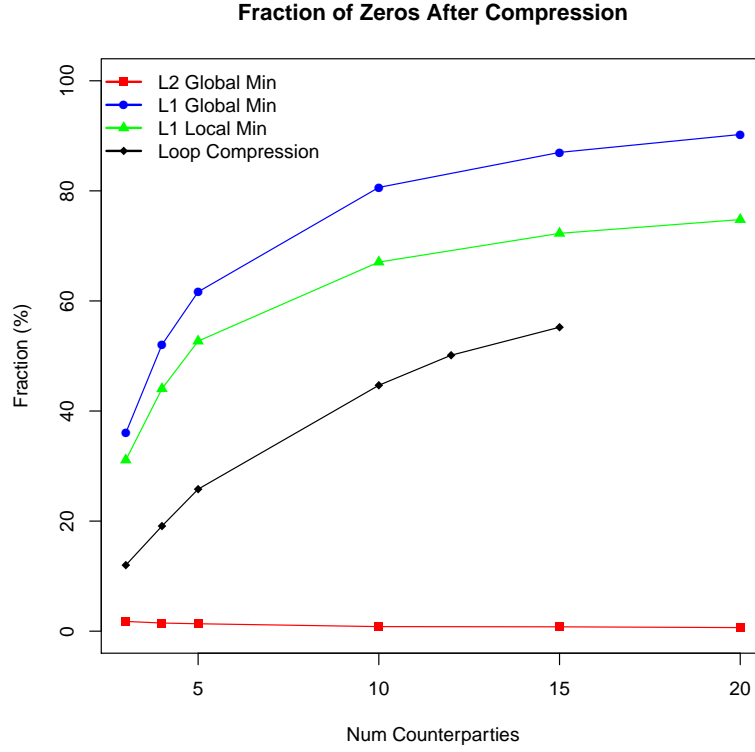


Figure 4: Comparison of different algorithms in terms of the percentage of final zeros.

having a much lower L_2 . This effect was seen across a wide range of initial scenarios. We consider this local algorithm to be the most interesting of all.

What all of these algorithms make very clear is that the success of a compression cycle is very dependent on the number of participating counterparties. Indeed we found that if we minimise the L_2 notional exposure, the average compression ratio is approximately equal to $\sqrt{2/N}$. It is therefore important for the dealer community, who represent the vast bulk of CDS trades, and regulators to ensure as high a level of participation as possible in the compression cycle.

There is an additional benefit in encouraging use of the compression cycle as an alternative counterparty risk mitigation approach to the use of CCPs. This is that by using compression, counterparties are still able to take advantage of the diversification of their exposures across different derivative asset classes which occurs within the close-out netting of the ISDA Master Agreement and the CSA. Combining compression and close-out netting further reduces the size of their mutual net exposures and the need for collateral.

In further work we will address the application of compression to the interest rate swap market. Although similar to CDS, there are additional complications in this market due to the fact that

interest rate swaps are less fungible than CDS⁸. We will also explore the effect of transaction costs and counterparty exposure constraints on compression.

Acknowledgements

The author would like to thank Paul Glasserman for useful suggestions. He would also like to thank Sam Morgan, Matthew Livesey, Christopher Randall, and TriOptima for their comments. This research benefited from the support of the French Banking Federation (FBF) Chair on *Banking regulation and innovation* under the aegis of the Louis Bachelier laboratory in collaboration with the Fondation Institut Europlace de Finance (IEF) and EDHEC.

A Estimating the Expected L_2 Compression Ratio

The expected L_2 compression ratio is defined as

$$CR_{2,N} = \mathbb{E} \left[\sqrt{\frac{\Omega(\hat{C})}{\Omega(C)}} \right]$$

where $\Omega(C)$ is given in Equation 1 and $\Omega(\hat{C})$ is given by Equation 4. However we will begin by calculating

$$\mathbb{E} \left[\frac{\Omega(\hat{C})}{\Omega(C)} \right]$$

where we have moved the expectation inside the square root. We will quantify the difference at the end of this section. From Appendix A, we know that $\hat{C}_{ij} = (h_i - h_j)/N$ which leads to Equation 4. This means that we can write the compression ratio for a specific initial scenario of exposures \mathbf{C} as a function of the initial randomly generated \mathbf{C} matrix.

$$\frac{\Omega(\hat{C})}{\Omega(C)} = \frac{1/N \sum_{i=1}^N h_i^2}{\sum_{i=1}^N \sum_{j=i+1}^N C_{ij}^2} = \frac{1}{N} \left(\frac{\sum_{i,j=1}^N C_{ij}^2 + \sum_{i=1}^N \sum_{j,k,j \neq k}^N C_{ji} C_{ki}}{\sum_{i=1}^N \sum_{j=i+1}^N C_{ij}^2} \right).$$

⁸This is because of their non-fixed coupons and the fact that their maturity date can fall on any business day of the year.

This can be rewritten as

$$\frac{\Omega(\hat{C})}{\Omega(C)} = \frac{2}{N} \left(1 + \frac{\sum_{i=1}^N \sum_{j < k}^N C_{ji} C_{ki}}{\sum_{i=1}^N \sum_{j=i+1}^N C_{ij}^2} \right).$$

The second term inside the bracket is an odd function of any C_{ij} . As the C_{ij} are independently drawn from a Gaussian distribution with zero mean, it will therefore be zero on the expectation. Hence we can conclude that

$$\mathbb{E} \left[\frac{\Omega(\hat{C})}{\Omega(C)} \right] = \frac{2}{N}.$$

This result allows us to use Jensen's inequality for a concave function to conclude that

$$CR_{2,N} \leq \sqrt{2/N}. \tag{7}$$

Using numerical experiments (a Monte Carlo simulation of 5,000 random draws of matrix \mathbf{C}), we confirmed this result and found that the relative difference between the compression ratio and $\sqrt{2/N}$ is typically less than 2%.

References

- BIS (November 2013). Derivative statistics. *Statistics Report from the Bank for International Settlements*.
- Brouwer, D. (2012). System and method of implementing massive early terminations of long term financial contracts. *TriOptima US Patent*, 8,306,905.
- Cont, R. and Moussa, A. (2010). Too interconnected to fail: default contagion and systemic risk in financial networks. *Working Paper*.
- Duffie, D. and Zhu, H. (2011). Does a central clearing counterparty reduce counterparty risk. *The Review of Asset Pricing Studies*, 1(1):74–95.
- Johnson, D. (1975). Finding all of the elementary circuits of a directed graph. *SIAM Computing*, 4(1):77–84.
- O’Kane, D. (2008). Modelling single-name and multiname credit derivatives. *Wiley Finance*.
- Ratings, F. (2009). Global credit derivatives survey: Surprises, challenges and future. *Credit Policy*.
- Sjoberg, P. (2013). Email communication. *TriOptima AB*.
- Tarjan, R. (1972). Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2):146–160.