

# Pseudorandom Financial Derivatives

[Extended Abstract]

David Zuckerman  
Department of Computer Science  
University of Texas at Austin  
1616 Guadalupe, Suite 2.408  
Austin, TX 78701  
diz@cs.utexas.edu

## ABSTRACT

Arora, Barak, Brunnermeier, and Ge [1] showed that taking computational complexity into account, a dishonest seller could dramatically increase the lemon costs of a family of financial derivatives. We show that if the seller is required to construct derivatives of a certain form, then this phenomenon disappears. In particular, we define and construct *pseudorandom derivative families*, for which lemon placement only slightly affects the values of the derivatives. Our constructions use expander graphs.

We study our derivatives in a more general setting than Arora et al. In particular, we analyze arbitrary tranches of the common collateralized debt obligations (CDOs) when the underlying assets can have significant dependencies.

## Categories and Subject Descriptors

J.4 [Social and Behavioral Sciences]: Economics; F.m [Theory of Computation]: Miscellaneous

## General Terms

Economics, Theory

## Keywords

pseudorandom, finance, derivative, expander

## 1. INTRODUCTION

Financial derivatives play a major role in our financial system, as became all too apparent in the recent financial crisis. A derivative is a financial product whose value is a function of one or more underlying assets. They can be used to hedge risk, provide leverage, or simply to speculate. The major benefit of derivatives is that they facilitate the buying and selling of risk.

While a derivative may depend on only one asset, in this paper we study derivatives that depend on many assets. One supposed benefit of such derivatives is that they can mitigate the effects of asymmetric information. That is, a seller may be aware that certain underlying assets are lemons, and try to strategically place

the lemons among the derivatives in order to minimize the derivatives' value. Nevertheless, an unknowledgeable buyer can buy less information-sensitive derivatives based on these assets without incurring significant risk. In other words, the *lemon cost* of these derivatives should be small. The lemon cost is the value without any lemons minus the value with lemons.

Arora, Barak, Brunnermeier, and Ge [1] introduced computational complexity into this discussion. They showed that contrary to the conventional wisdom, once computational complexity is accounted for, the lemon costs of derivatives could increase dramatically, at least under a plausible computational assumption. Indeed, in the recent financial crisis, it appears that sellers did pack lemons into their CDOs without buyers' knowledge; see for example Michael Lewis's *The Big Short* [7].

Before describing how we get around this problem, we first describe the setting of Arora et al. There are  $n$  assets and  $m$  derivatives, where each derivative is a function of  $r$  underlying assets. We will have  $n \ll mr$ , so that each asset underlies several derivatives. This is typically not the case if the underlying assets are, say, mortgages; however, in the common case that the underlying assets are credit default swaps, it is often the case that an asset can underlie several derivatives.

Arora et al. model the relationship between derivatives and underlying assets as a bipartite graph. The nodes are the derivatives and assets, and there is an edge between a derivative and an asset if the derivative depends on that asset. If the derivatives are for sale to the public, then the seller must make this graph public.

Now consider a seller who knows that certain underlying assets are lemons. For certain derivatives, it is advantageous for the seller to concentrate many of these lemons into a small number of derivatives. Thus, these lemons and the lemon-loaded derivatives will correspond to a dense subgraph of the original graph. Arora et al. observed that if it is computationally intractable to check whether an arbitrary graph (or even a somewhat random graph) contains a dense subgraph, then it is computationally intractable to catch such a dishonest seller. Therefore, the lemon cost could be quite high.

We circumvent this problem. Instead of allowing the seller to use an arbitrary bipartite graph to construct the CDO family, we mandate that the seller use a specific bipartite graph. Of course, the buyer can easily check that the seller did use the specified graph. While the seller will still be able to assign assets to nodes arbitrarily, we can choose the graph judiciously to avoid the dense subgraph problem, for the following reason. Although it may be computationally intractable to test whether an arbitrary graph, or even a somewhat-random graph, contains a dense subgraph, it is nevertheless possible to explicitly construct a graph with no dense subgraphs. We choose such a graph for the seller.

Graphs with no dense subgraphs are related to certain fundamen-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

EC'11, June 5–9, 2011, San Jose, California, USA.

Copyright 2011 ACM 978-1-4503-0261-6/11/06 ...\$10.00.

tal objects in the theory of pseudorandomness: randomness extractors and expander graphs. For a survey of these objects and other aspects of pseudorandomness, see [10].

Our constructions motivate the notion of a *pseudorandom derivative family*. This is a set of derivatives such that no matter how the lemons are placed by an adversary, the sum of the value changes of the derivatives will be small. In other words, adversarial placement of lemons behaves similarly to random placement.

Alternatively, we may decompose the lemon cost into the unavoidable lemon cost plus the cost of dishonesty. The unavoidable lemon cost is the lemon cost for an honest seller who randomly places the lemons. The cost of dishonesty is the additional cost from a dishonest seller who strategically places the lemons. A pseudorandom derivative family is one where the cost of dishonesty is small.

In our main result, we show how to construct good pseudorandom derivative families, using expander graphs with expansion close to the degree.

Of course, in order to analyze values we need a model for the underlying assets. Arora et al. assume the underlying assets are independent fair coin flips, taking the value 1 with probability 1/2, whereas the lemons always take value 0.

We analyze a more realistic model with dependencies. In particular, we only require that the probability distribution on any  $r$  assets depends only on how many of the assets are lemons. We also don't need to assume lemons have value 0. We do obtain our strongest results when we assume good assets "dominate" lemons, as explained below.

One model satisfying our requirements allows dependencies among assets to occur through some global random variable  $Z$ . This  $Z$  could represent the state of the economy and housing market, and other relevant information. We make no assumptions about  $Z$ . For each fixing of  $Z$ , to say  $z$ , there are two probability distributions  $D_g = D_g(z)$  and  $D_\ell = D_\ell(z)$ . Conditioned on  $Z = z$ , our model assumes all assets are independent, with good assets chosen according to  $D_g$ , and lemons chosen according to  $D_\ell$ . Moreover, we say good assets dominate lemons if for any  $z$ , if  $X$  is chosen according to  $D_g(z)$  and  $Y$  is chosen according to  $D_\ell(z)$ , then for all  $a$ ,  $\Pr[X \geq a] \geq \Pr[Y \geq a]$ .

We also need a model for the derivatives. One of the most common derivatives is the collateralized debt obligation, or CDO. The size of the CDO market was recently over a trillion dollars. Arora et al. analyze these CDOs, but only safe (senior) tranches, and they focus on an unrealistic binary variant of them. We focus on the realistic, tranching CDOs, and study arbitrary tranches. We obtain our strongest results for the entire CDO.

A CDO has a natural structure, packaging many underlying assets into *tranches*. For example, a CDO could have 100 underlying mortgages, each of which is supposed to pay \$1,000. The "senior" tranche, for instance, could collect the first \$85,000. Thus, if more than \$85,000 is paid from these 100 mortgages, this tranche receives \$85,000; if some amount  $x \leq \$85,000$  is paid, the tranche receives  $x$ . The next tranche could range from \$85,000 to \$95,000. If more than \$95,000 is paid, this tranche receives the full \$10,000; if less than \$85,000 is paid, this tranche receives nothing. If the amount  $x$  paid is between \$85,000 and \$95,000, then the tranche receives  $x - \$85,000$ . In general, the  $[a, b]$  tranche receives  $\min(x, b) - \min(x, a)$ .

We begin by explaining our model and defining key terms in Section 2. We then describe how expander graphs give pseudorandom CDOs in Section 3. We modify existing expander constructions to obtain our CDOs in Section 4. Finally, we analyze the case when good assets don't necessarily dominate lemons in Section 5.

## 2. THE MODEL AND KEY DEFINITIONS

First we give some notation. For a positive integer  $n$ , we let  $[n]$  denote the set  $\{1, 2, \dots, n\}$ . For a vector  $v = (v_1, \dots, v_s)$ , we let  $\|v\|_1 = \sum_i |v_i|$ , the  $L_1$  norm.

Our CDOs will be functions of underlying assets. We first describe our assumptions about the underlying assets, and then define pseudorandom CDOs.

### 2.1 Model for Underlying Assets

In our model, there are two types of assets, *lemons* and *good assets*. Good assets must dominate lemons in a sense below. This requirement will be satisfied if lemons always take value zero, but it allows more general distributions on lemons. Each CDO will depend on  $r$  assets. Our results hold as long as the probability distribution on any  $r$  assets depends only on how many of the assets are lemons.

We now elaborate on one natural model which satisfies the two requirements above. We model dependencies among assets as occurring through some global random variable  $Z$ . This  $Z$  could represent the state of the economy and housing market, and other relevant information. We make no assumptions about  $Z$ . For each fixing of  $Z$ , to say  $z$ , there are two probability distributions  $D_g = D_g(z)$  and  $D_\ell = D_\ell(z)$ . Conditioned on  $Z = z$ , our model assumes all assets are independent, with good assets chosen according to  $D_g$ , and lemons chosen according to  $D_\ell$ . Moreover, we say good assets dominate lemons if for any  $z$  and  $a$ ,

$$\Pr_{X \sim D_g(z)}[X \geq a] \geq \Pr_{Y \sim D_\ell(z)}[Y \geq a].$$

We can relax the requirement that assets are conditionally independent. It suffices that the conditional distribution on assets is  $r$ -wise independent, i.e., any  $r$  of them are independent. (This does not imply that they are mutually independent.)

We normalize asset values so that each asset's maximum value is 1. We let  $\mu$  and  $\lambda$  be the expected values of each good asset and lemon, respectively. The dominance requirement implies  $\mu \geq \lambda$ , and let  $\delta = \mu - \lambda$  be the additional expected value of a good asset.

### 2.2 Pseudorandom CDOs

**DEFINITION 2.1.** *A collateralized debt obligation (CDO) is a derivative on an underlying portfolio of assets. For  $0 = a_0 < a_1 < \dots < a_s$  (called attachment points), the  $i$ th tranche is given by the interval  $[a_{i-1}, a_i]$ . If the underlying portfolio pays off  $x$ , then the value of the  $i$ th tranche is  $\text{value}_{[a_{i-1}, a_i]}(x) = \min(x, a_i) - \min(x, a_{i-1})$ . If the tranche is understood, we often omit it as a subscript in value.*

Since assets are normalized to have maximum value 1, if the CDO depends on  $r$  assets, the last attachment point  $a_s = r$ .

We will be interested in families of CDOs.

**DEFINITION 2.2.** *An  $(n, m, r)$ -CDO family is a set of  $m$  CDOs on  $n$  assets identified with the set  $[n]$ , where each CDO depends on  $r$  assets.*

We will have  $n \ll mr$ , so that each asset underlies several derivatives.

The seller (creator of the CDOs) knows that some  $\ell$  assets are lemons, and may identify the lemons with any subset  $L \subseteq [n]$  of size  $\ell$ . We will be interested in the total value of tranches in our CDO family.

DEFINITION 2.3. For  $L \subseteq [n]$ , let  $\text{tv}_{[a,b]}(L)$  denote the total expected value of all  $[a, b]$  tranches in the CDO family, if the assets corresponding to assets  $L$  are lemons. If the tranche is understood, we often omit it as a subscript. We define the vector  $\vec{\text{tv}}(L) = (\text{tv}_{[a_0, a_1]}(L), \text{tv}_{[a_1, a_2]}(L), \dots, \text{tv}_{[a_{s-1}, a_s]}(L))$ .

A dishonest seller will try to choose the subset  $L$  to minimize  $\text{tv}(L)$ . A CDO family is pseudorandom if the seller cannot gain significantly by this choice.

DEFINITION 2.4. An  $(n, m, r)$ -CDO family is pseudorandom for  $\ell$  lemons for  $[a, b]$  tranches with error  $\epsilon$  if for any two subsets  $L, L' \subseteq [n]$  of size  $\ell$ ,

$$|\text{tv}_{[a,b]}(L') - \text{tv}_{[a,b]}(L)| \leq \epsilon m(b - a).$$

Note that  $m(b - a)$  is the maximum possible value of the  $[a, b]$  tranches with no lemons. Thus, for any CDO family the error  $\epsilon \leq 1$ .

We further define pseudorandomness for the entire CDO family. We can't generalize the above definition naively, to say that the total value of the CDO doesn't change significantly if the lemons are moved. This is because the total value of the CDO equals the total value of the underlying assets; therefore moving lemons won't change the value at all. Instead, we strengthen the definition to ensure that not much value can be transferred among the different tranches. That is, we add up the value changes of each tranche; this gives the  $L_1$ -norm.

DEFINITION 2.5. An  $(n, m, r)$ -CDO family is pseudorandom for  $\ell$  lemons with error  $\epsilon$  if for any two subsets  $L, L' \subseteq [n]$  of size  $\ell$ ,

$$\|\vec{\text{tv}}(L') - \vec{\text{tv}}(L)\|_1 \leq \epsilon mr.$$

Note that  $mr$  is the maximum possible value of the entire CDO family with no lemons. The error  $\epsilon$  for the CDO family is at most the maximum error for a tranche, and hence at most 1.

We can compare our notion of pseudorandom error to the traditional notion of lemon cost. The lemon cost is the value without any lemons minus the value with lemons. In a sense, we are dividing the lemon cost into two components: the unavoidable lemon cost plus the cost of dishonesty. The unavoidable lemon cost is the lemon cost for an honest seller who randomly places the lemons. The cost of dishonesty is the additional cost from a dishonest seller who strategically places the lemons. Thus, the normalized cost of dishonesty is upper bounded by the pseudorandom error.

On the other hand, the pseudorandom error above is at most the normalized lemon cost, but it could be significantly less. For example, if all underlying assets are lemons, the lemon cost will be high, but the error in our definition will be 0, since the value doesn't change depending on the lemon placement. Indeed, the pseudorandom error is small if the lemon cost doesn't depend significantly on the lemon placement.

### 2.3 Bipartite Expander Graphs

Following Arora et al., we view the relationship between derivatives and underlying assets as a bipartite graph. We review the basic definitions.

DEFINITION 2.6. A bipartite graph is a triple  $(A, B, E)$ , with left vertices  $A$ , right vertices  $B$ , and edges  $E \subseteq A \times B$ . We usually view  $E$  as unordered pairs of vertices. Sometimes we refer to a bipartite graph on  $A \cup B$  to mean some bipartite graph  $(A, B, E)$  with suitable choice of edges  $E$ . For a subset of vertices  $S \subseteq A \cup B$ , let  $\Gamma(S) = \{v | (\exists w \in S) \{v, w\} \in E\}$  denote the set of neighbors

of  $S$ . We often write  $\Gamma(v)$  for  $\Gamma(\{v\})$ . The degree of a vertex  $v$  is  $|\Gamma(v)|$ . The graph is  $d$ -left-regular if all left vertices have degree  $d$ , and similarly for right-regular. The graph is  $(d, r)$ -biregular if it is  $d$ -left-regular and  $r$ -right-regular.

The vertices  $A$  and  $B$  correspond to the assets and derivatives, respectively, with an edge between a derivative vertex and asset vertex if the derivative depends on the asset.

Since Arora et al. showed how dense subgraphs can be problematic, it is natural to try to choose a graph with no dense subgraphs. It is natural to use known constructions of suitable "randomness extractors," which can be shown to lack dense subgraphs. Indeed, this was our original approach. However, we obtain stronger results in a simpler manner by considering the related *expander graphs*, where we require expansion of asset vertices.

DEFINITION 2.7. A bipartite graph on  $[n] \cup [m]$  is an  $(\ell_{max}, \gamma)$ -expander if for every subset  $S \subseteq [n]$  of size at most  $\ell_{max}$ ,  $|\Gamma(S)| \geq \gamma|S|$ .

Note that we only need expansion of left vertices; expansion of right vertices is not required. We will need a strong form of an expander, called a *unique-neighbor expander*.

DEFINITION 2.8. Let  $\Gamma_i(S)$  denote the set of vertices  $v$  with  $|\Gamma(v) \cap S| = i$ .  $\Gamma_1(S)$  are called the unique neighbors of  $S$ .

DEFINITION 2.9. A bipartite graph on  $[n] \cup [m]$  is an  $(\ell_{max}, \gamma)$ -unique-neighbor expander if for every subset  $S \subseteq [n]$  of size at most  $\ell_{max}$ ,  $|\Gamma_1(S)| \geq \gamma|S|$ .

The following simple lemma is well known.

LEMMA 2.10. A  $d$ -left-regular  $(\ell_{max}, d - \Delta)$ -expander is an  $(\ell_{max}, d - 2\Delta)$ -unique neighbor expander.

PROOF. Consider any subset  $S$  on the left of size  $\ell \leq \ell_{max}$ . It has at least  $(d - \Delta)\ell$  neighbors, which leaves at most  $\Delta\ell$  edges unaccounted for. Thus  $|\Gamma_1(S)| \geq |\Gamma(S)| - \Delta\ell$ , as required.  $\square$

It is not hard to show that random graphs are excellent expanders, using the probabilistic method. However, we need to be able to certify that a graph is an expander; otherwise Arora et al. showed how the seller can cheat. We therefore seek explicit constructions of expanders.

Explicit expander constructions are highly nontrivial. The classic constructions of Gabber and Galil [3] and Lubotzky-Phillips-Sarnak [8] do not give unique-neighbor expanders. Ta-Shma, Umans, and Zuckerman constructed the first unique-neighbor expanders of polylogarithmic left degree [9], and Capalbo et al. were the first to achieve constant left degree [2]. For our purposes, the best expanders were constructed by Guruswami, Umans, and Vadhan [5], although these have polylogarithmic degree. For more on expanders we refer the reader to the excellent survey [6].

### 3. EXPANDERS GIVE PSEUDORANDOM CDOs

Before discussing expander constructions, we first show how unique-neighbor expanders give pseudorandom CDOs. It is helpful to compare our bounds to a natural trivial bound. To this end, observe that any biregular  $(n, m, r)$ -CDO family is pseudorandom against  $\ell$  lemons for  $[a, b]$  tranches with error at most  $d\ell\delta/(m(b - a))$ . This is because converting  $\ell$  good assets to lemons decreases the value of the entire CDO family by  $d\ell\delta$ , since each lemon is in  $d$  CDOs.

We show that a CDO family built from a  $(d, r)$ -biregular  $(\ell, d - \Delta)$ -unique neighbor expander has error at most  $2\Delta\ell\delta/(m(b - a))$ .

That is, we replace  $d$  from the trivial bound by  $2\Delta$ . Moreover, the naive bound on the error for the entire CDO is the maximum of the errors for each tranche. We are instead able to improve the error to  $3\Delta\ell\delta/(mr)$ .

The intuition for the proof is natural. We consider some placement of lemons. By the unique-neighbor expansion, we have fairly tight bounds on both the number of derivatives containing no lemons, and the number containing exactly one lemon. Thus, when we subtract values for two different lemon placements, there is a lot of cancellation.

**THEOREM 3.1.** *A CDO built from a  $(d, r)$ -biregular  $(\ell, d - \Delta)$ -unique neighbor expander is pseudorandom for  $\ell$  lemons. For the tranche  $[a, b]$ , the error is at most  $2\Delta\ell\delta/(m(b - a))$ , and for the entire CDO the error is at most  $3\Delta\ell\delta/(mr)$ .*

Before beginning the proof, we define the following.

**DEFINITION 3.2.** *Let  $\text{val}_{[a,b]}(g) = \mathbb{E}[\text{value}_{[a,b]}(X)]$ , where the random variable  $X$  is the payoff of an underlying portfolio on  $r$  assets,  $g$  of which are good. If the tranche is understood, we often omit it as a subscript.*

Since good assets dominate lemons, we deduce that  $\text{val}$  is a non-decreasing function of  $g$ . This is obvious if lemons always take value zero, but requires a short proof in general.

**LEMMA 3.3.** *For any tranche  $[a, b]$  and  $g' \geq g$ ,  $\text{val}_{[a,b]}(g') \geq \text{val}_{[a,b]}(g)$ .*

**PROOF.** First fix  $Z = z$ . Now let  $F_D$  denote the cumulative distribution function of distribution  $D$ . We can choose random variables  $X$  and  $Y$  according to  $D_g = D_g(z)$  and  $D_\ell = D_\ell(z)$ , respectively, by choosing  $W \in [0, 1]$  uniformly and outputting  $X = F_{D_g}^{-1}(W)$  and  $Y = F_{D_\ell}^{-1}(W)$ . This ‘‘coupling’’ and the domination condition imply that for every point in the probability space,  $X \geq Y$ . Thus, we may substitute good assets for lemons in such a way that for any point in the probability space, the value of every asset either increases or remains the same. The lemma follows.  $\square$

Since a CDO simply restructures payoffs, the sum of the expected payoffs of the CDO equals the sum of the payoffs of the underlying assets. This gives the following observation.

**OBSERVATION 3.4.** *For any  $g$ ,  $\sum_{i=1}^s \text{val}_{[a_{i-1}, a_i]}(g) = g\mu + (r - g)\lambda = r\lambda + g\delta$ .*

Lemma 3.3 and Observation 3.4 imply the following corollary.

**COROLLARY 3.5.** *For any  $g, i$ , and tranche  $[a, b]$ ,  $0 \leq \text{val}_{[a,b]}(g + i) - \text{val}_{[a,b]}(g) \leq i\delta$ .*

Let  $t_i(L) = |\Gamma_i(L)|$ , for  $0 \leq i \leq r$ . The following lemma is key to proving Theorem 3.1.

**LEMMA 3.6.** *For any tranche  $[a, b]$  and any  $L, L' \subseteq [n]$  with  $|L| = |L'| = \ell$ , we have:*

$$\begin{aligned} \text{tv}_{[a,b]}(L') - \text{tv}_{[a,b]}(L) &\leq \Delta\ell(\text{val}_{[a,b]}(r) - \text{val}_{[a,b]}(r - 1)) \\ &\quad + \sum_{i=2}^r t_i(L)(\text{val}_{[a,b]}(r) - \text{val}_{[a,b]}(r - i)). \end{aligned}$$

**PROOF.** Fix the tranche  $[a, b]$ . Since  $\cup_{i=0}^r \Gamma_i(L) = [m]$ , we have  $\sum_{i=0}^r t_i(L) = m$ .

We must study the quantity

$$\text{tv}(L) = \sum_{i=0}^r t_i(L) \text{val}(r - i).$$

By the unique neighbor expansion property,  $t_1(L) \geq (d - \Delta)\ell$ . Therefore,  $|t_1(L) - t_1(L')| \leq \Delta\ell$ . Assume without loss of generality that  $\text{tv}(L) \leq \text{tv}(L')$ . Using  $\sum_{i=0}^r t_i(L) = \sum_{i=0}^r t_i(L')$ , we can now bound:

$$\begin{aligned} &\text{tv}(L') - \text{tv}(L) \\ &= \sum_{i=0}^r (t_i(L) - t_i(L'))(-\text{val}(r - i)) \\ &= \sum_{i=0}^r (t_i(L) - t_i(L'))(\text{val}(r) - \text{val}(r - i)) \\ &\leq |t_1(L) - t_1(L')|(\text{val}(r) - \text{val}(r - 1)) \\ &\quad + \sum_{i=2}^r t_i(L)(\text{val}(r) - \text{val}(r - i)) \\ &\leq \Delta\ell(\text{val}(r) - \text{val}(r - 1)) + \sum_{i=2}^r t_i(L)(\text{val}(r) - \text{val}(r - i)). \end{aligned}$$

$\square$

We can now prove the theorem.

**PROOF OF THEOREM 3.1.** First note that

$$\sum_{i=1}^r it_i(L) = d\ell,$$

since both sides count the number of edges incident to  $L$ . Since  $t_1(L) \geq (d - \Delta)\ell$ , we have

$$\sum_{i=2}^r it_i(L) \leq \Delta\ell.$$

Now fix the tranche  $[a, b]$ , and we now bound its error. By Lemma 3.6, Corollary 3.5, and the above,

$$\begin{aligned} &\text{tv}(L') - \text{tv}(L) \\ &\leq \Delta\ell(\text{val}(r) - \text{val}(r - 1)) + \sum_{i=2}^r t_i(L)(\text{val}(r) - \text{val}(r - i)) \\ &\leq \Delta\ell\delta + \sum_{i=2}^r t_i(L)i\delta \\ &\leq 2\Delta\ell\delta. \end{aligned}$$

Dividing by  $m(b - a)$  gives the result for the  $[a, b]$  tranche.

Now we analyze the error for the entire CDO. In a similar manner, we get:

$$\begin{aligned} &\|\vec{\text{tv}}(L') - \vec{\text{tv}}(L)\|_1 \\ &\leq \Delta\ell \sum_{i=1}^s (\text{val}_{[a_{i-1}, a_i]}(r) - \text{val}_{[a_{i-1}, a_i]}(r - 1)) + \\ &\quad \sum_{i=1}^s \sum_{j=2}^r (t_j(L) + t_j(L'))(\text{val}_{[a_{i-1}, a_i]}(r) - \text{val}_{[a_{i-1}, a_i]}(r - j)) \\ &= \Delta\ell\delta + \sum_{j=2}^r (t_j(L) + t_j(L'))j\delta \\ &\leq 3\Delta\ell\delta. \end{aligned}$$

Dividing by  $mr = dn$  gives the required result.  $\square$

## 4. CONSTRUCTIVE EXPANDERS AND CDOS

In this section, we show how suitable explicit expanders yield pseudorandom CDOS. Roughly, if the number of lemons is small compared to the number of derivatives, we get a big gain over the trivial bound.

**THEOREM 4.1.** *For any  $\alpha \in (0, 1]$  and positive integers  $n, m, d, r$  such that  $nd = mr$ , the following holds for  $\Delta = 2(2d)^\alpha (\log_d n) \log_d m$  and any positive integer  $\ell_{max} \leq (\Delta m / (8d^3))^\alpha$ . There is an explicit pseudorandom  $(n, m, r)$ -CDO family against  $\ell$  lemons, for all  $\ell \leq \ell_{max}$ . For the tranche  $[a, b]$ , the error is at most  $4\Delta\ell\delta / (m(b-a))$ , and for the entire CDO the error is at most  $6\Delta\ell\delta / (mr)$ .*

To prove this, we use the strong and elegant expander construction of Guruswami, Umans, and Vadhan [5]. We will set parameters in a different order, so we use their Theorem 3.3, obtained before they set parameters.

**THEOREM 4.2.** [5] *For any positive integer  $h$ , a prime power  $q$ , and  $n$  and  $m$  powers of  $q$ , there is an explicit construction of a  $(\ell_{max}, q - \Delta)$  expander on  $[n] \cup [m]$  with left degree  $q$ ,  $\ell_{max} = h^{\log_q m - 1}$ , and  $\Delta = (h - 1)(\log_q n - 1)(\log_q m - 1)$ .*

Before setting parameters, we need the following simple observation.

**OBSERVATION 4.3.** *Suppose we are given a  $(\ell_{max}, d - \Delta)$  expander with left-degree  $d$ . If we remove any left vertices, and add any right vertices, the graph remains a  $(\ell_{max}, d - \Delta)$  expander. If for each left vertex, we remove an arbitrary  $d - d'$  edges, then the graph becomes a  $(\ell_{max}, d' - \Delta)$  expander with left degree  $d'$ .*

We now set parameters from Theorem 4.2 as follows.

**COROLLARY 4.4.** *For any  $\alpha \in (0, 1]$  and positive integers  $n, m$ , and  $d$ , there is an explicit construction of a  $(\ell_{max}, d - \Delta)$  expander on  $[n] \cup [m]$  with left degree  $d$  for  $\ell_{max} = (m / (4d^2))^\alpha$  and  $\Delta = (2d)^\alpha (\log_d n) \log_d m$ .*

**PROOF.** Let  $q$  be the smallest power of 2 that is at least  $d$ . Let  $n'$  be the smallest power of  $q$  at least  $n$ , and let  $m'$  be the largest power of  $q$  at most  $m$ . By Observation 4.3, it suffices to construct a  $(\ell_{max}, q - \Delta)$  expander on  $[n'] \cup [m']$  with left-degree  $q$ . Set  $h = \lceil q^\alpha \rceil$  and  $\ell = \log_q m' = \lfloor \log_q m \rfloor$ , so  $q^\ell \leq m < q^{\ell+1}$ . We use the expander constructed in Theorem 4.2. It suffices to lower bound  $\ell_{max}$  and upper bound  $\Delta$ . We get:

$$\ell_{max} \geq h^{\ell-1} \geq q^{\alpha(\ell-1)} > (m/q^2)^\alpha \geq (m/(4d^2))^\alpha,$$

and

$$\begin{aligned} \Delta &\leq (h-1)(\log_q n' - 1)(\log_q m' - 1) \\ &< q^\alpha (\log_q n) \log_q m < (2d)^\alpha (\log_d n) \log_d m. \end{aligned}$$

This completes the proof.  $\square$

This and other known unique-neighbor expander constructions give left-regular graphs. However, we need the graph to be biregular. We show how to convert a left-regular graph to biregular while increasing the left-degree only slightly, at the expense of increasing the number of right vertices. The following extends a lemma from [4].

**LEMMA 4.5.** *Suppose we are given a  $d_0$ -left-regular  $(\ell_{max}, \gamma)$  expander on  $[n] \cup [m_0]$ , and parameters  $m, d, r$  such that  $nd = mr$ ,  $d_0 < d \leq m_0$ , and  $m \geq m_0 d / (d - d_0)$ . We can efficiently construct a  $(d, r)$ -biregular  $(\ell_{max}, \gamma)$  expander on  $[n] \cup [m]$ .*

**PROOF.** Let  $r_0 = nd_0/m_0$  denote the original average right degree. For any right node  $v \in [m_0]$  of degree  $r_v > r$ , divide it into  $\lceil r_v/r \rceil$  vertices, where  $\lfloor r_v/r \rfloor$  have degree  $r$  and at most one has degree less than  $r$ . (Partition neighbors arbitrarily.)

The number of new nodes added is at most

$$\sum_{v \in [m_0]} \left( \left\lceil \frac{r_v}{r} \right\rceil - 1 \right) < \sum_{v \in [m_0]} \frac{r_v}{r} = \frac{m_0 r_0}{r} = \frac{nd_0 m}{nd} = \frac{d_0 m}{d}.$$

Thus, the total number of right nodes is less than

$$m_0 + \frac{d_0}{d} m \leq \frac{d - d_0}{d} m + \frac{d_0}{d} m = m.$$

Add isolated nodes to the right to make the total number of right nodes exactly  $m$ . Now add edges arbitrarily to the right and left to make all left degree  $d$  and right degrees  $r$ , which is possible because  $nd = mr$ . Naively, this may allow multiple edges, but we can avoid this by filling edge slots in the following order. For left nodes, cycle over all nodes  $d - d_0$  times, filling one edge slot each time. For right nodes, cycle over all nodes once, filling all edge slots for a node before proceeding to the next node.  $\square$

**COROLLARY 4.6.** *For any  $\alpha \in (0, 1]$  and positive integers  $n, m, d$ , and  $r$  such that  $nd = mr$ , there is an explicit construction of a  $(d, r)$ -biregular  $(\ell_{max}, d - \Delta)$  expander on  $[n] \cup [m]$  for  $\Delta = 2(2d)^\alpha (\log_d n) \log_d m$  and  $\ell_{max} = (\Delta m / (8d^3))^\alpha$ .*

**PROOF.** Set  $\Delta_0 = \Delta/2$ ,  $d_0 = d - \Delta_0$ , and  $m_0 = \Delta_0 m / d$ . By Corollary 4.4, there is an explicit construction of a  $(\ell_{max}, d_0 - \Delta'_0)$  expander on  $[n] \cup [m_0]$  with left degree  $d_0$  for  $\ell'_{max} = (m_0 / (4d_0^2))^\alpha \geq \ell_{max}$  and  $\Delta'_0 = (2d_0)^\alpha (\log_d n) \log_d m_0 \leq \Delta_0$ . Now apply Lemma 4.5.  $\square$

Combining Corollary 4.6 and Lemma 2.10 with Theorem 3.1 yields Theorem 4.1.

If  $d$  is smaller, we could use the expanders of [2], but the degree is not as good a function in the error and our results are not as strong.

## 5. TWO GENERAL ASSETS

In this section we obtain bounds even if the probability distribution of lemons is not dominated by the probability distribution of good assets. We only assume that  $\mu \geq \lambda$ , where  $\mu$  and  $\lambda$  are the expected values of each good asset and lemon, respectively. We don't need to think of the second asset as a lemon; instead consider two general assets with expected values  $\mu \geq \lambda$ . Now, even the "trivial" bounds change; such bounds can be deduced from our bounds below. We show that in the general case, the  $\delta$  in Theorem 3.1 must be replaced by  $\mu$  for the  $[a, b]$  tranche. Moreover, we no longer get better bounds for the entire CDO than can be deduced from the bounds in the tranches.

**THEOREM 5.1.** *A CDO built from a  $(d, r)$ -biregular  $(\ell, d - \Delta)$ -unique neighbor expander is pseudorandom for  $\ell$  lemons. For the tranche  $[a, b]$ , the error is at most  $2\Delta\ell\mu / (m(b - a))$ .*

In this case, Lemma 3.3 may no longer hold, and as a result, neither may Corollary 3.5. We instead obtain an analog of Corollary 3.5 with  $\delta$  replaced by  $\mu$ .

**LEMMA 5.2.** *For any  $g, i$ , and tranche  $[a, b]$ ,  $|\text{val}_{[a,b]}(g+i) - \text{val}_{[a,b]}(g)| \leq i\mu$ .*

To prove this, it is helpful to use the following expression for the value of the  $[a, b]$  tranche.

**OBSERVATION 5.3.** *Let  $X$  denote a random variable representing the payoff of a portfolio underlying a CDO. Then the value of the corresponding  $[a, b]$  tranche is*

$$\int_a^b \Pr[X > x] dx.$$

**PROOF OF LEMMA 5.2.** It suffices to prove the lemma for  $i = 1$ . Let  $X$  be the payoff of a portfolio on  $r - 1$  assets,  $g$  of which are good. Let  $Y$  be the payoff of a good asset, and  $Z$  the payoff of a lemon. We wish to show that

$$\left| \int_a^b \Pr[X + Y > w] dw - \int_a^b \Pr[X + Z > w] dw \right| \leq \mu.$$

To this end, first note that

$$\int_a^b \Pr[X + Y > w] dw \geq \int_a^b \Pr[X > w] dw,$$

and similarly for  $X + Z$ , since both  $Y$  and  $Z$  are nonnegative. It therefore suffices to show that

$$\int_a^b \Pr[X + Y > w] dw - \int_a^b \Pr[X > w] dw \leq \mu,$$

and hence the corresponding inequality for  $X + Z$ .

Condition on  $X = x$ ; we show this inequality for any  $x$ . Observe that

$$\Pr[x + Y > w] - \Pr[x > w] = \begin{cases} \Pr[Y > w - x] & \text{if } x \leq w \\ 0 & \text{otherwise} \end{cases}$$

Letting  $y = w - x$  gives:

$$\int_a^b (\Pr[x + Y > w] - \Pr[x > w]) dw \leq \int_0^\infty \Pr[Y > y] dy = \mu.$$

This completes the proof.  $\square$

**PROOF OF THEOREM 5.1.** We now proceed as in our earlier proof, replacing Corollary 3.5 with Lemma 5.2. Everything else goes through as before.  $\square$

## Acknowledgements

I thank Sanjeev Arora, Rafa Mendoza-Arriaga, Kumar Muthuraman, Ryan O'Donnell, and Stathis Tompaidis for useful comments and discussions. Thanks also to the anonymous referees for helpful comments.

## 6. REFERENCES

- [1] S. Arora, B. Barak, M. Brunnermeier, and R. Ge. Computational complexity and information asymmetry in financial products. In *Innovations in Computer Science (ICS) conference*, 2010.
- [2] M. Capalbo, O. Reingold, S. Vadhan, and A. Wigderson. Randomness conductors and constant-degree expansion beyond the degree/2 barrier. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, pages 659–668, 2002.
- [3] O. Gabber and Z. Galil. Explicit construction of linear sized superconcentrators. *Journal of Computer and System Sciences*, 22:407–420, 1981.
- [4] V. Guruswami, J. Lee, and A. Wigderson. Almost Euclidean subspaces of  $\ell_1^n$  via expander codes. *Combinatorica*, 30:47–68, 2010.
- [5] V. Guruswami, C. Umans, and S. Vadhan. Unbalanced expanders and randomness extractors from Parvaresh-Vardy codes. *Journal of the ACM*, 56:1–34, 2009.
- [6] S. Hoory, N. Linial, and A. Wigderson. Expander graphs and their applications. *Bulletin of the American Mathematical Society*, 43:439–561, 2006.
- [7] M. Lewis. *The Big Short*. W.W. Norton & Co., 2010.
- [8] A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. *Combinatorica*, 8:261–277, 1988.
- [9] A. Ta-Shma, C. Umans, and D. Zuckerman. Lossless condensers, unbalanced expanders, and extractors. *Combinatorica*, 27:213–240, 2007.
- [10] S. Vadhan. The unified theory of pseudorandomness. *SIGACT News*, 38, 2007.