Loopholes in Complex Contracts

Kenneth Ayotte and Adam B. Badawi[*]

In 2016, J. Crew exploited a loophole in a complex debt contract, allowing it to remove $250 million of collateral from its lenders' reach to refinance other debt. Why would an important and unanticipated flaw like this arise in a contract between sophisticated commercial actors? Answering this question requires a model where contracts can be imperfect and evolving. We create such a model in a principal-agent setting using genetic algorithms, a tool from the complexity science literature. As in real-world practice, contracts in a genetic algorithm evolve from prior contracts, combining terms from the "parent" generation based on their past performance.

Our main result is that loopholes arise as an inherent by-product of evolutionary learning. Through experience, contracting parties gradually allow the agent's activity level to increase as they learn how to permit more activity without increasing agency costs. But, as activity rises, contracts become more complex in two ways. First, more value-relevant states of the world arise. This increases the number of potential mistaken terms that the agent can exploit. Second, the interdependence of terms increases, making loopholes more severe. In particular, when a flaw arises in a term that identifies and blocks bad activity, the flaw does more damage when a greater activity level is permissible. As contracts evolve, then, they can achieve greater performance but also become less robust to mistakes.

[*] U.C. Berkeley School of Law. We would like to thank David Hirshleifer, Melanie Mitchell, Josh Teitelbaum, Spencer Williams, and audiences at George Mason, Georgetown, USC, the BYU Winter Deals Conference, the Corporate Restructuring & Insolvency Seminar, and the Annual Meeting of the American Law and Economics Association for helpful comments.

I. Introduction

In 2014, J. Crew renegotiated a $1.5 billion term loan from a syndicate of lenders. The loan provided the lenders with collateral, including the company's trademarks. The long and complex renegotiated agreement contained extensive restrictions on J. Crew's ability to transfer that collateral. When J. Crew encountered further financial distress, its legal and financial advisors examined all of its debt documents carefully. Buried within the term loan agreement, they found a "trap door." This loophole allowed the company, using a combination of other specific permissions in the agreement, to transfer $250 million of the trademark collateral into a subsidiary that was "unrestricted"—i.e., free from the covenants that normally constrain borrowing against that collateral.[1] This allowed J. Crew to exchange its unsecured bonds for new bonds secured by the intellectual property, effectively allowing junior bondholders to leapfrog the term lenders in the J. Crew capital structure. This kind of loophole exploitation has become a standard part of the playbook for distressed restructurings.

Why would a loophole of this kind—an unanticipated mistake that one party can exploit—arise in an agreement between sophisticated market actors? To find an answer, we must depart from a neoclassical model of optimal contracting, which assumes that parties are fully rational and forward-looking. One reason may be the underlying complexity of the term loan agreement. The trap door term was one of 21 separate "carve-outs" (i.e. permissions) in the Investments section of the negative covenants in the loan agreement. Further, the Investments section is one of several ways that value can be moved into an unrestricted subsidiary.[2] We might suspect that the greater the number of applicable terms, the greater the chance an agent can find a potential a flaw to exploit.

But this begs the question: why do agreements like this become so complex in the first place? Returning to the J. Crew example, the loophole provisions seem to spring from an intention to permit other activity the parties view as efficient. Specifically, the original intent of the trap door provision was to allow J. Crew to invest in overseas subsidiaries while keeping the profit free from U.S. taxation (Ayotte and Scully, 2021). Generally, the unrestricted subsidiary concept is used to allow companies to invest in joint ventures and other projects without having to worry about their effects on debt covenants.[3] Lenders could have insisted on a more airtight agreement by disallowing unrestricted subsidiaries entirely, as some borrowers do (Buccola and Nini 2022),

---

[1] The J. Crew transaction involved two steps involving interrelated carve-outs in its credit agreement that, according to J. Crew, created specific permission to transfer the IP and borrow against it. One carve-out allowed J. Crew to invest $150M from a loan party into a restricted subsidiary that is not a loan party. Such subsidiaries are not guarantors of the debt, but remain subject to the covenants in the loan agreement, which would restrict borrowing against the IP. A second carve-out permitted investment from a non-loan party restricted subsidiary in an unlimited amount.

[2] A similar "drop down" transaction occurred in the restructuring of Neiman Marcus. They used the ability of the company to "redesignate" a restricted subsidiary holding their Mytheresa business as an unrestricted subsidiary, and then transferred the equity using its restricted payments capacity to a parent-level entity.

[3] See Colin Chang, et al., Thinking Outside the Box: The Basics of Unrestricted Subsidiaries, White & Case Client Alert (2017), available at: https://www.whitecase.com/sites/whitecase/files/files/download/publications/thinking-outside-the-box-the-basics-of-unrestricted-subsidiaries.pdf.

but they might sacrifice efficiency gains by doing so. But this introduces a trade-off: once the door is open to unrestricted subsidiaries, they can be used in unintended ways.

In this paper, we introduce a modeling tool that is new to contracts analysis, genetic algorithms, to model the evolution of complex contracts. Our model allows us to capture both of these intuitions—the link between evolution and complexity, and the link between complexity and loopholes. We use the algorithm to model the evolution of principal-agent contracts, like J. Crew's term loan agreement. The agent in our algorithm searches a space for investment opportunities that can have positive or negative value. The agent has a preference to overinvest, so the algorithm experiments with packages of terms, akin to covenants, that try to restrict the agent from pursuing negative value investments while permitting positive value ones.

Our approach allows contracts to evolve in a way that matches two stylized facts about contract formation in commercial practice:

1. Lawyers do not write contracts from scratch: they start with existing contracts and borrow terms that are commonly used in the marketplace.[4] As a result, contracts are path dependent and evolve gradually, rather than immediately (Badawi, De Fontenay, Dyreng, and Hills, 2021, Ayotte and Ellias, 2021, Coates, 2016).
2. Innovations can have uncertain consequences because contract terms can interact with other terms in potentially unknown ways. As a result, observed past performance guides subsequent drafting.

The genetic algorithm is an intuitive method that captures these stylized facts about contract formation. Evolution in a genetic algorithm follows a process mimicking biological reproduction. "Children" (in our setting, the current generation of contracts) inherit their "genes" (the contract terms) by combining genes from parents, who are selected with probability increasing in their "fitness" (past performance). Terms from contracts that have performed better in the past are more likely to populate the next generation. The algorithm also introduces occasional random "mutations" (random changes to individual contract terms). Both mutations and unexpected interactions from combining contracts together have the potential to introduce loopholes that the agent can exploit.

We find that vulnerability to loopholes is an inherent by-product of evolutionary learning. Contract performance tends to improve over time as the algorithm learns strategies for controlling agency costs. In our setting, this means learning the right mix of contract terms that restrict the agent from taking bad investments while permitting the good ones. In the early generations, contracts have not yet learned these distinguishing strategies. As a result, the best performing contracts in early generations are ones that limit the agent's activity: they prevent the agent from exploring the investment space for opportunities. Early generation contracts are less complex contracts, in that fewer payoff-relevant states of the world arise due to the lower level

[4] As Richman (2011) puts it: "The paradigmatic question each attorney asks regarding a new legal product is, 'Has this worked before?' not 'How can we make this work best?'"

of activity. A real-world example of this is a contract that does not permit unrestricted subsidiaries at all. This prohibits all kinds of investment activity connected to these subsidiaries; it is also simpler to contract, because it does not attempt to distinguish permissible from impermissible uses of them. The simplicity of this term avoids any loopholes that might arise from attempts to distinguish restricted and unrestricted subsidiaries.

Over time, contracts learn strategies for allowing good investments and preventing bad. As they do, the contract gradually permits the agent to engage in more activity, so the agent finds more investment opportunities and invests more. Contracts evolve a greater number of payoff-relevant terms. But this complexity also increases the contract's vulnerability to loopholes: a greater number of payoff-relevant terms increases the chance for a potential exploitable mistake.

We also find that loopholes also have the potential to become more severe as activity rises. To spotlight the reason for this, we examine each contract term one-by-one, and rank the harm to contract performance if each term were changed to the least restrictive term. The harm to contract performance from the most "dangerous" term increases over time. When we examine the cause of this phenomenon, we find that this flows from a second type of complexity: greater interactivity between contract terms. The most dangerous term for a loophole is typically an agency cost-controlling term--one that identifies a bad investment and blocks it. This term interacts importantly with activity level terms--ones that affects the amount of investment the agent can make. Because permitted activity levels tend to grow as the contract evolves, the contract becomes less robust to mistakes in agency cost controlling terms.[5]

The link between the level of permitted activity and loophole severity is recognized in the practitioner literature. For example, Covenant Review, a service that analyzes debt contract terms, noted the greater danger of the same J. Crew trap door term when it is found in high yield bonds. J. Crew was only able to move $250 million in trademark value into the unrestricted subsidiary because of other covenants that capped the amounts it could invest. High yield bonds typically do not contain such caps, thus making the same loophole more dangerous.[6]

The intuition from our model also sheds light on the benefits and costs of contractual complexity. Our view of the benefits is most similar to Coates (2016), which finds that increasing complexity in M&A deal terms tends to serve efficiency goals. Other work is more skeptical about the benefits of complexity. In Hill (2001), complexity follows from lawyer risk-aversion and behavioral biases: because it is more dangerous to subtract rather than add, contracts tend to grow in complexity over time. In Anderson and Manns (2016), complexity flows from wasteful over-lawyering. Our model is different from these in highlighting a direct cost of complexity on contract performance: complexity increases vulnerability to loopholes.

A close paper to ours is Jennejohn, Nyarko, and Talley (2021). This paper also uses an algorithmic approach to gain intuition about patterns of contractual evolution. Our projects have

[5] One important aspect that is missing from our model is precognition—the ability of contracting parties to anticipate a potential loophole and block it in advance. Mutations occur randomly in our algorithm, and the probability of a mutation is independent of its potential harm. In future iterations, we plan to explore models that tie the probability of a mistake to its potential or realized harm.

some similarity, in that historical experience plays an important role in the evolution process, and also shows that contracts may not evolve to an optimum. Their project focuses more on the important role of law firms as conduits for past experience and learning. We do not tackle the important role of law firms here. Rather, our genetic algorithm places more focus on contractual complexity and its connection to loopholes.

Another closely related line of research is the work by Choi, Gulati, and Scott that examines how certain contractual clauses change, or do not change, over time. These authors argue that some provisions in commercial boilerplate can become encrusted through rote usage (Choi et al., 2017). The slow responses of transacting parties to unexpected interpretations of clauses such as *pari passu* provisions in sovereign debt contracts suggests that contractual change may sometimes be the product of random mutation rather than rational design (Choi et al., 2018). These authors also develop evidence that lawyers who operate in lower agency cost settings tend to be more responsive to legal change (Choi et al., 2021). We similarly examine changes to contracts that are sometimes the product of adaption and sometimes the product of mutation. But our project does not have the extensive focus on the evolution a single and important clause that has been subject to legal change. Rather we focus on how entire contracts become more complex over time and how that relates to susceptibility to loopholes.

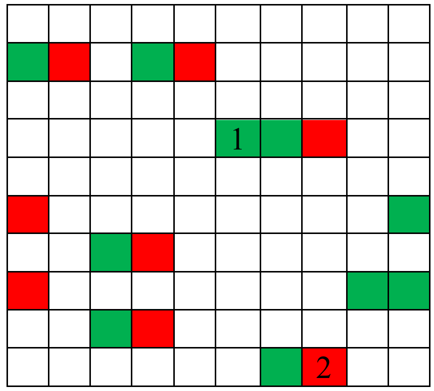
II. The Genetic Algorithm

Our model is adapted from the canonical genetic algorithms game of Robby, a can collecting robot (Mitchell, 2009). In Mitchell's game, the robot travels around a 10 x 10 board searching for randomly placed cans to pick up. The robot has a fixed number of moves to collect as many cans as possible, with moves directed by the algorithm that evolves over time. The robot's "fitness" increases with the number of cans it picks up. We start with Mitchell's game and modify it to suit our prinicipal-agent setting. A fuller description of the differences between our game and Mitchell's game can be found in the Appendix. We think the model applies best to debt contracting, but the intuition can apply more broadly to any principal-agent setting with complex terms.

A. The Investment Space

The classic principal-agent setup involves an agent with superior skill to the principal, but with differing objectives. For example, the agent may have a desire to overinvest because of private benefits (Aghion and Bolton 1992) or preferences for risky investments that shift value from creditors to shareholders (Jensen and Meckling 1976). We represent agency costs by populating our board with positive value (green) and negative value (red) investment opportunities. A representative investment space is presented in Figure 1 below:

Figure 1: Representative Investment Space



Our agent will begin in the upper left corner of the space and travel the board, as directed by the contract, to find investment opportunities. When he finds them, he may take them, or pass them by, depending on what the contract prescribes. The goal of the algorithm will be to evolve principal-agent contracts that allow the agent to take good investments while forcing the agent to pass by the bad. [7]

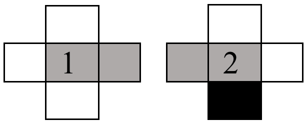
An important element of our setup is that bad investments are placed adjacent to good investments in a deterministic pattern that the contract can learn about. The real-world analogy might be that investment opportunities tend to have higher net present value when the firm's earnings are higher, so a covenant might block investment below an earnings threshold. We set up the board by randomly placing a good opportunity in a specified percentage of squares, and then place a costly opportunity to the east of the good opportunity whenever the space is empty (with some exceptions we address below).

B. Contracts and Agents

Each agent's movement in the investment space will be governed by a contract. The contract is a contingent action plan that maps the agent's current contractible state, or "scenario", into one of six possible actions. Figure 2 below is a visual illustration of the scenarios that correspond to spaces 1 and 2 in Figure 1 above. The middle square represents the agent's current position on the board. A scenario can describe only the agent's current square, and the squares to the immediate north, south, east and west; since it is not based on the entire board, our contracting space is inherently incomplete. Also, a scenario can distinguish the presence of an investment opportunity, but not whether an opportunity is good or bad; hence, we represent an opportunity with a gray color in Figure 2. It is also possible that the scenario lies near a border wall, shown in black. The three possibilities for any given square, then, are {opportunity, no opportunity, wall}.

[7] In our game, fitness (and thus evolution) is based on the principal's payoff. Thus, there are no agency costs in the drafting process (for example, the agent inserts pro-agent terms, etc). This would be an interesting issue for future work.

Figure 2: representative scenarios 1 and 2



The contract will prescribe an action for each of the 128 feasible scenarios. The possible actions that can be prescribed are:

{0: move north; 1: move east; 2: move west; 3: move south; 4: move random; 5: autonomy}.

Actions 0-3 direct the agent to move one square in the prescribed direction; move random picks randomly from choices 0-3. Move 5 is called "autonomy". This corresponds to a scenario where the contract gives the agent freedom to pursue and take investments. Specifically, when given autonomy, the agent will do the following:

a. If there is an opportunity in the current space, take it;
b. If not, scan the board in all directions (N,S,E,W) and move one space toward the nearest opportunity.

Thus, a contract seeking to maximize overall fitness faces a trade-off. Giving the agent autonomy is necessary to exploit the agent's ability to find and take opportunities that benefit the principal; but some restrictions on his autonomy may be necessary to limit the agent's preference for taking any possible investment opportunity, including those that are costly to the principal.

In this setting, there are 128 unique states that the agent can encounter. A contract, then, can be represented as a string of 128 digits from 0-5. An example is below:

{53235404414543014322411151545325123534344125430241430342054203551035105104021401530022135514251104415002421400540422233003353102}

C. The evolution process

Our population consists of 200 principal-agent pairs, each of whom starts with a contract populated with 128 randomly chosen terms. When play starts, the individual agents then navigate an investment space that has been randomly populated with good and bad investments. To determine what to do at each step, the agent cross-references his current scenario with the contract, and acts as instructed for a lifespan of 50 moves. At the end of each round, we calculate the contract's performance ("fitness") for each principal-agent pair. Fitness is calculated as the score assigned to good investments (call this g) times the number of good investments taken (N_g), less the score assigned to bad investments (b) times the number of bad investments taken (N_b):

$$F = gN_g - bN_b$$

One can think of "fitness" as the overall value produced by the contract: the joint payoff of the principal and the agent. The weights on good and bad investments, g and b, will allow us to vary the severity of agency costs, something we plan to expand upon in future drafts of the paper. For now, we use g=10 and b=40 in all our simulations. This represents an environment where agency costs are high, so the contract will focus heavily on avoiding bad investments. Fitness scores help to determine whether that principal-agent pair's contract terms will propagate into the next generation through the "mating" process. Each new generation of principal-agent pairs repeats this process until the simulation reaches 300 generations.

The key processes that generate evolution in the genetic algorithm are *crossover* and *mutation.* Crossover is the combining of terms from two "parent" contracts to create new "child" contract. To form a child contract, a random location in the 128-term contract is chosen (say, location 23). The first parent's contract would contribute the first 23 terms from its contract, and the second parent's contract would contribute the remaining 128 – 23 = 105 terms from its sequence. Parents are chosen from the population randomly, with probability that is increasing in their fitness. Crossover serves a dual purpose of preserving bundles of terms (called *schema*) that work well together, while experimenting with new schema by combining parent contract terms.

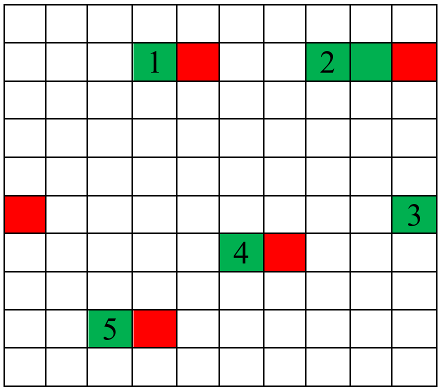
In contrast to crossover, *mutation* is the random replacement of individual genes. Each term in an agent's contract has an exogenous small probability (.001) of mutating from one generation to the next. If a term is selected for mutation, it is replaced through a random process.[8] Mutation plays an important role in generating diversity of genes in the population. But it can also introduce errors that undermine a contract.

D. Contracting Strategies: A Simple Example

Figure 3 shows a typical board that the agent may encounter and a common, but not exclusive, set of opportunity configurations. As discussed above, we place good opportunities next to bad opportunities in a deterministic way. Spaces 1, 4, and 5 in Figure 1 show this common arrangement of opportunities. They involve a good opportunity and a bad opportunity to its immediate east, with no other cans in the vicinity. Sometimes, however, two good opportunities will get randomly placed next to each other and, in this case, we add a costly opportunity to the east of the two good opportunities (Scenario 2). In other cases, the good opportunities will be at the edge of the board (Scenario 3). When that happens, we place the bad opportunity at the far edge of the same row. Although they do not appear in Figure 3, particularly complex arrangements occur when clusters of opportunities appear directly above and below one another.

[8] To allow the overall degree of autonomy in the contract to evolve endogenously, the probability that a mutation generates the autonomy action is the frequency of autonomy actions in the existing contract. This ensures that mutations will not force the contract toward a pre-defined degree of autonomy.

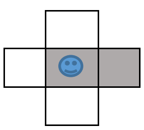
Figure 3



Fully exploiting these scenarios involves different amounts of complexity. For example, suppose an agent finds itself on square 1 above. Ideally, the contract will allow the agent to pick up the good opportunity, steer him away from the bad opportunity, and then point him toward other opportunities.

To do this, it might adopt the following sequence of contract terms:

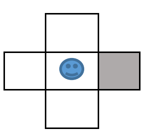
1)



Scenario (Y = opportunity, N = no opportunity)					Action
Current	North	South	East	West	
Y	N	N	Y	N	5 (autonomy)

Recall that giving the agent autonomy is the only way to get the agent to take an investment opportunity in his current space, so the agent will take the good opportunity. This removes the good opportunity from the board, leaving the agent in the following scenario:

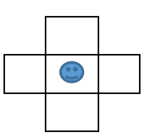
2)



Scenario (Y = opportunity, N = no opportunity)					Action
Current	North	South	East	West	
N	N	N	Y	N	3 (move south)

Moving south would direct him away from the bad opportunity. Having moved south, his scenario would now be no opportunities in any nearby space:

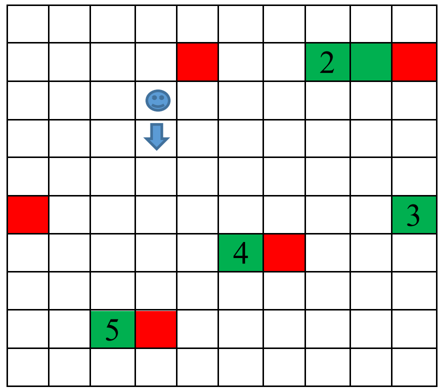
3)



Scenario (Y = opportunity, N = no opportunity)					Action
Current	North	South	East	West	
N	N	N	N	N	5 (autonomy)

Restoring autonomy would allow the agent to scan the board in all directions. Updating the board in Figure 3b below, the agent would scan the board in all four directions (N,S,E,W). Seeing that the nearest opportunity is to the south (the red square next to the space labeled 5), the agent would then move one space to the south, and the game would continue.

Figure 3b: Updated board after two moves



Observing that the only opportunity in his field of vision is a can at the bottom of the board (see cluster 5 in Figure 3 above), he would then move south.

There is an important point embedded in this simple example. Learning how to exploit one opportunity configuration can be applied to other similar configurations. Once the contract figures out how to move when reaching square 1, that same sequence of contract terms can be applied effectively on squares 4 and 5 as well. But this means that if a mistake arises in this sequence, it might also propagate over multiple configurations as well. This is the sense in which our model connects the activity level and the severity of loopholes. As we will see in the next section, the agent's activity level is endogenous to the contract: it depends on how much the contract allows the agent to travel the board and encounter more opportunities. As we will discuss in the next section, activity tends to rise over time as contracts evolve and learn.

II. Results

A. Evolutionary Learning: Activity Levels and Complexity

In Table 1, we show evidence from our genetic algorithms that demonstrate the link between evolution and activity levels over time. Data is based on averages over 100 runs. We record the average number of good and bad investments chosen, and the corresponding average fitness levels, using the weights g=10 and b=40. To measure the amount of activity, we measure the number of unique squares the agent touches on average in his lifespan. Greater movement around the board means that the agent will encounter more investment opportunities. To ascertain the complexity of the agent's behavior, we calculate the average number of scenarios the agent faces over a de minimus threshold.

We can see a clear pattern: over time, activity levels rise. At generation 50, the agent touches 12.9 unique squares, and takes only 1.32 good investments and 0.098 bad investments. By generation 300, the agent reaches 20.2 unique squares on average. Overall investment activity tends to rise over time, and the gap between good and bad investments rises as well, suggesting that contracts are learning how to distinguish good investments from bad. We also observe that the agent uses more of the contract. At generation 25, the agents face an average of 95.3 scenarios, while at Generation 300 the agents use an average of 108.7.

Table 1: Evolution of the Agent over Time					
Generation	Fitness	Good Investments	Bad Investments	Unique Squares	Scenarios Faced
25	3.5	0.55	0.050	7.7	95.3
50	9.3	1.32	0.098	12.9	103.0
100	14.8	2.02	0.136	16.6	105.5
150	17.4	2.35	0.153	18.1	106.7
200	19.3	2.60	0.167	19.0	107.4
250	20.7	2.80	0.181	19.7	108.1
300	21.9	2.95	0.190	20.2	108.7

Another way to see the activity level phenomenon more directly is to observe the behavior of a typical agent in an early and in a late generation.

Figure 4a: Generation 50, At Start

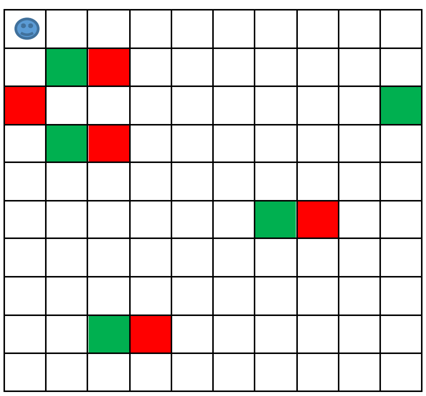
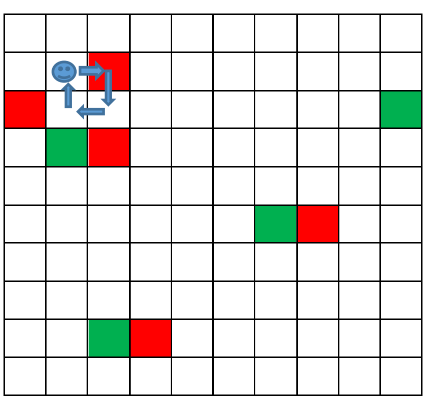


Figure 4b: Generation 50, At Finish



A typical generation 50 contract is still quite crude. The agent takes one good investment opportunity, and then moves around in a square pattern for the remainder of his lifespan. We call this early stage contracting strategy a "shelter in place" strategy. It prevents the agent from exploring for opportunities as a way to defend against bad investment opportunities. It is easy to see that very few scenarios arise for the agent under this simple strategy.

The figures above also help us understand the evolution of contractual complexity. By construction, all contracts are 128 scenario-action pairs, and this cannot change over time. But if we measure contract complexity by the number of states that actually *arise*, then this contract looks relatively simple: only seven unique scenarios actually arise in the agent's lifespan.

When an agent reaches generation 300, the agent's activity is greater and more complex:

Figure 5a: Generation 300, At Start

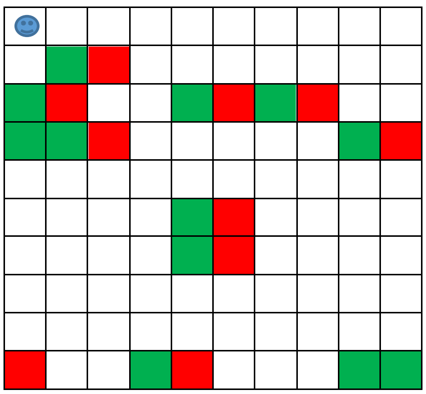
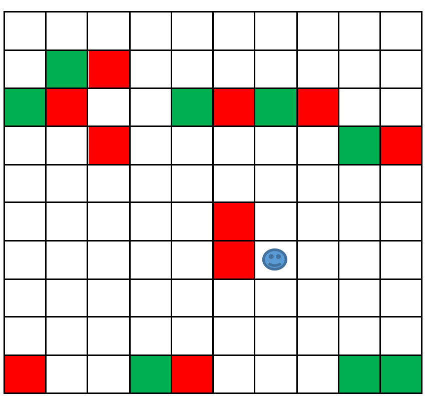


Figure 5b: Generation 300, At Finish



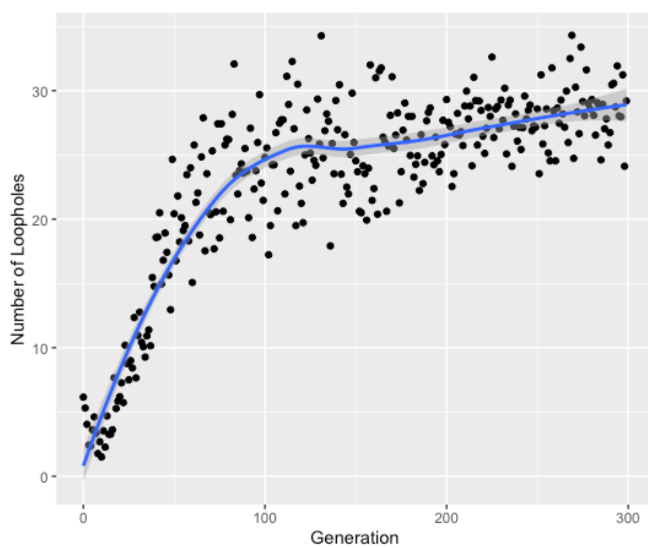
This agent covers more ground, and also invests more: in this game, he takes four good investments and zero bad investments. By allowing for more activity, the contract is substantially more complex. In this lifetime, the agent encounters 14 unique scenarios. This means that more terms in the contract become payoff-relevant.

B. Loopholes: Frequency and Severity

The previous section demonstrates a connection between evolution, activity, and complexity. This section shows evidence connecting complexity and loopholes. Specifically, we want to know how many terms in a contract would have an important effect on contract performance if a mistake arises. The more such terms exist, the more vulnerable is the contract to a loophole. We suspect that this will increase over time, for the reasons we discuss above: as learning occurs, the contract becomes more complex, as measured by the number of payoff-relevant terms. This increases its potential vulnerability.

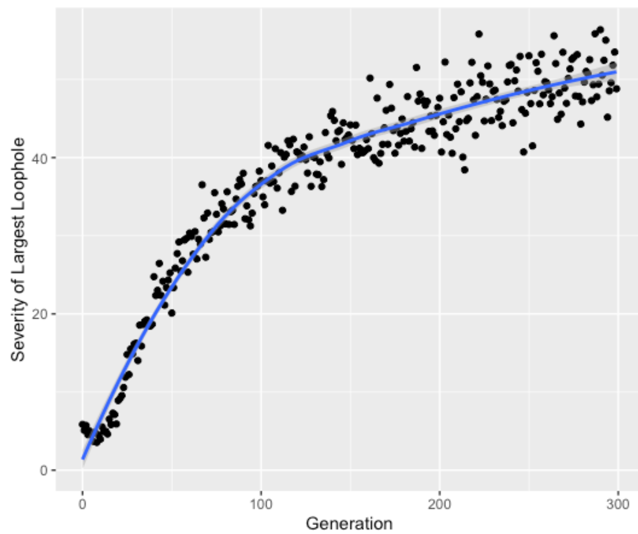
We test this hypothesis by taking a median-performing contract at each generation, and "shocking" its terms by changing them, 1-by-1, to the most pro-agent term (autonomy). We believe that shifting a restrictive term to one that permits agent autonomy captures how some loopholes arise, including the one in J. Crew. In that case, a term that was meant to restrict excessive investment was read to permit it. We then record the fitness decline from that change and count the number of times the change reduces fitness by more than a threshold value (here, 5 fitness points). Figure 6 below shows that the number of potential loopholes grows with time:

Figure 6: Number of Potential Loopholes



Next, we examine loophole severity, as measured by the most "dangerous" potential mistake to a term. As above, we take a median-performing contract and shock the terms 1-by-1, as before. Now, we record the greatest fitness decline among the 128 terms, and track it over time. We also find that loophole severity rises as the contract evolves:

Figure 7: Severity of Worst Loopholes



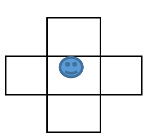
C. Mechanisms: Loopholes and Interactivity

The finding that a more evolved contract increases loophole severity poses the question of why this happens. We hypothesize that this result is the product of interactions between contract terms that become more important over time. Returning to our original motivating example, the J. Crew loophole case depended on many important interactions between terms. The company claimed that the following loan document term gave it permission to transfer $250 million of intellectual property to an unrestricted subsidiary:

(t) Investments made by any Restricted Subsidiary that is not a Loan Party to the extent such Investments are financed with the proceeds received by such Restricted Subsidiary from an Investment in such Restricted Subsidiary made pursuant to Sections 7.02(c)(iv), (i)(B) or (n); and

One can immediately see that the ability to use this permitted investment term depended importantly on the definitions of "Investment", "Restricted Subsidiary", and "Loan Party". These definitions are quite lengthy and complex themselves: the Investments definition is 225 words long. Moreover, investments under this section are limited to the proceeds of other investments made pursuant to other sections of the document in Sections 7.02. These sections contain permissible investment types and amounts—these sections limited J. Crew to transferring "only" $250 million in trademark value.

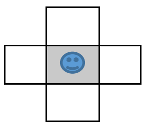
To understand how interactions can arise in our simulations, it helps to detail two terms in the contract. One we call the "travel" term. This term is the most important term affecting the agent's activity level: it occurs with high frequency, and affects the agent's movement through the investment space. It applies when the agent is on a blank square and is surrounded by blank squares on all sides.



Scenario (Y= opportunity, N = no opportunity)				
Current	North	South	East	West
N	N	N	N	N

If this term is set to “5” (autonomy), the agent will scan the board and move in a targeted way toward the nearest opportunity. Other terms that move the agent in a non-targeted direction will generate less overall activity[9].

The second term we call the “bad blocker” term. This is where the agent’s current space has an opportunity, with no other opportunities surrounding it:



Scenario (Y= opportunity, N = no opportunity)				
Current	North	South	East	West
Y	N	N	N	N

When this scenario arises, the agent is most likely facing a bad investment opportunity. The reason for this is that opportunities start in adjacent pairs (bad immediately to the east of good). But after taking a good opportunity, this often leaves a bad opportunity standing alone. If the agent moves to that space and is given autonomy, he will take the opportunity and lose fitness points. The greater activity level as contracts evolve means that this loophole will apply to more activity and become more costly to fitness. In the analysis that follows below, we assess the vulnerability of agents to loopholes by setting the bad blocker gene to autonomy. This will permit the agent to take a bad investment when this scenario arises.

In the early stages of learning, when the agent’s actions are strongly restricted, it is uncommon for the travel term to be set to autonomy.[10] With little knowledge of how to maneuver around bad investments to find the good ones, the algorithm does not maximally target opportunities. But later in the agent’s lifecycle, when the algorithm can better negotiate investment opportunities, the travel term is more often set to autonomy. This evolution suggests that the contract’s ability to navigate the investment space and the amount of activity the contract encourages interact with each other.

To develop evidence for this interactivity, we introduce changes to the contract at different points in its evolution. The goal of this exercise is to observe the effect of loopholes in the “bad blocker” term at two stages of the contract’s life cycle. Our hypothesis is that interactivity between the travel and bad blocker terms will increase over time. That is, we hypothesize that the severity of a bad blocker loophole should increase over time because the travel gene and the bad blocker gene will co-evolve to maximize fitness. As the contract gets better at blocking bad investments, the travel gene will encourage more activity to find more opportunities. But this interaction will cause more severe consequences when a mistake in a bad blocker arises.

To make this assessment, we calculate a “triple differences” in fitness. In words, we hypothesize that the algorithm evolves late generation travel terms that are more sensitive to a loophole in the

[9] In unreported results, we have verified that changing the travel term from 5 to any other term will, on average, reduce the number of unique squares the agent reaches and the number of good and bad opportunities he takes.
[10] In 100 runs of the simulation, autonomy is the modal travel gene in 17 runs at generation 25, 24 runs at generation 50, and 45 runs at generation 300.

"bad blocker" term than the early generation travel terms. Taking this step-by-step, we first calculate a difference-in-differences for late generation agents:

$$(F_{LG\_Active} - F_{LG\_Active\_Loophole}) - (F_{LG\_Inactive} - F_{LG\_Inactive\_Loophole}) \qquad (1)$$

The first term, F_{LG_Active}, is the average fitness of late generation agents without any modifications to the contract terms. The second term, $F_{LG\_Active\_Loophole}$, is the average fitness for those same agents who have their bad blocker terms mutated to autonomy. We expect that this difference will be the largest because late generation contracts will frequently have autonomy in the travel gene position. The third term, $F_{LG\_Inactive}$, is the average fitness for late generation agents that have had their travel terms mutated to the modal travel gene at generation 50 (this is the "move random" term). The fourth term, $F_{LG\_Inactive\_Loophole}$, is the average fitness for agents that have had their travel gene mutated to move random and their bad blocker genes mutated to autonomy. Due to the diminished activity that comes with the less active travel gene, we expect the second difference in (1) to be smaller than the first.

The next step is to do the same difference-in-differences at generation 50:

$$(F_{EG\_Inactive} - F_{EG\_Inactive\_Loophole}) - (F_{EG\_Active} - F_{EG\_Active\_Loophole}) \qquad (2)$$

The first term in (2), $F_{EG\_Inactive}$, is the average fitness of the early generation agents with no gene modification. The second term, $F_{EG\_Inactive\_Loophole}$, is the average fitness of those same agents who have their bad blocker genes mutated to the loophole, i.e. autonomy. The third term, F_{EG_Active}, is the average fitness for the early generation agent that has had its travel gene mutated to autonomy, the modal travel gene at generation 300. The fourth term, $F_{EG\_Active\_Loophole}$, is the average fitness for an early generation agent that has had both its travel gene mutated to autonomy and its bad blocker gene mutated to autonomy.

We expect the difference-in-differences in (2) to be lower than (1), i.e. a positive triple-difference, because the evolved travel gene at generation 50 tends to be a more cautious one—it creates less overall activity, resulting in fewer good and bad investments from being chosen. This should result in greater robustness to mistakes in the "bad blocker" gene.

Table 2: Analysis of Interactions Between Terms

Panel A							
Mean Values for Late Generation Agents							
(1)	(2)	(3)	(4)	(5)	(6)	(7)	
F_{LG_Active}	$F_{LG\_Active\_Loophole}$	Mean((1) - (2))	$F_{LG\_Inactive}$	$F_{LG\_Inactive\_Loophole}$	Mean((4) - (5))	Mean ((3) - (6))	N
21.9	-22.6	44.5	19.7	-18.4	38.1	6.3	100
Panel B							
Mean Values for Early Generation Agent Simulations							
(1)	(2)	(3)	(4)	(5)	(6)	(7)	
$F_{EG\_Inactive}$	$F_{EG\_Inactive\_Loophole}$	Mean((1) - (2))	F_{EG_Active}	$F_{EG\_Active\_Loophole}$	Mean((4) - (5))	Mean ((3) - (6))	N
9.3	-9.2	18.5	9.2	-11.1	20.3	-1.9	100
Panel C							
Wilcoxon Ranked Sum Test for Differences between Panel A(7) and Panel B(7)							
(1)	(2)						
Panel A (7)	Panel B (7)	Mean((1) - (2))	Wilcoxon Statistic	p-value	N		
6.3	-1.9	8.2	8676	0.00000	100		

The results from 100 runs of the simulation, which we summarize in Table 2, confirm our hypothesis. For the late generation contracts, we see a positive difference-in-differences, implying that switching to the modal travel gene from generation 50 weakens the contract's vulnerability to loopholes. Conversely, when we look at the generation 50 contracts, switching to the modal travel gene from generation 300 increases vulnerability to loopholes. In Panel C of Table 2, we confirm that the triple difference is statistically significant, with Wilcoxon statistic of 8676 and p-value of .00000.

We draw two conclusions from these results. The first is that they provide evidence that the contract terms interact with each other. Early in the agents' evolution, when they are unable to navigate investment opportunities, the genes that control activity tend to discourage exploration. But, once the agent learns how to maneuver through investment space, the contract encourages more activity. The second lesson is that removing restrictions from the agent and making it more active increases its vulnerability to loopholes. When the early generation agent is mutated to become more active, the consequences are not dire because the agent remains cautious. But making the late-generation agent less active provides a meaningful reduction in the harm caused by loopholes because that agent has fewer restrictions when it encounters investment opportunities. The statistically significant difference between the analysis in Panel A and Panel B provides substantial support for that inference.

III. Conclusion

In this paper we construct a genetic algorithm to analyze a setting that resembles financial contracting. Doing so allows us to avoid some of the assumptions that the neoclassical contracting paradigm makes. While the neoclassical approach recognizes that contracting is costly and faces problems verifying certain aspects of performance, this approach often assumes that parties have deep foresight and draft contracts from scratch. The genetic model departs from these assumptions by deriving contracts from the most successful contracts used in the past and by imposing ex ante uncertainty about the effects of changes to contracts that is only revealed by the ex post performance of these modified templates. These assumptions mirror the fact that deal lawyers rely extensively on past contracts to draft new agreements and that there is substantial uncertainty about the consequences of modifying terms in these contracts. Simulating contractual evolution in this environment allows us to observe what happens when we modify the underlying difficulty of the contracting environment, the frequency of investment opportunities, and the degree of conflict between principals and agents.

This approach allows us to model a contract's vulnerability to loopholes—unanticipated mistakes that the agent can exploit--at the expense of overall contract performance. Loopholes arise because contracts combine terms from previous generations that may generate unanticipated interactions. Mutations can also create random changes to contract terms. We find that complexity tends to increase over time for efficiency reasons: over time, the parties allow the agent to engage in more activity over time. But loopholes follow as a natural consequence: the more value-relevant terms are necessary to make a contract function, the greater is the chance of an exploitable mistake.

Although we believe the model generates useful intuition that can explain the presence of loopholes, we are also aware that there are many missing real-world features we would like to incorporate in future research. The most obvious is that our contracts evolve in a model that has no precognition at all. We think it is useful to create a stark contrast between our model and the neoclassical one where parties are omnisciently rational. Nevertheless, it is probably likely that lawyers drafting contracts can limit loophole severity and frequency by anticipating problems in advance and correcting them. We would like to incorporate this natural behavior into our algorithm. We also plan to explore how evolution varies with agency costs, and better understand why loopholes close, in addition to understanding why they open.

References

Aghion, Philippe and Bolton, Patrick, An incomplete contracts approach to financial contracting, 59 The Review of Economic Studies, 473-494 (1992).

Aghion, Philippe and Tirole, Jean, Formal and real authority in organizations, 105 Journal of Political Economy, 1-29 (1997).

Ash, Elliott and MacLeod, Bentley and Naidu, Suresh, The language of contract: Promises and power in union collective bargaining agreements, working paper (2019).

Ayotte, Kenneth and Scully, Christina, J. Crew, Nine West, and the Complexities of Financial Distress, Yale Law Journal Forum, 2021.

Ayotte, Kenneth and Ellias, Jared A., Bankruptcy process for sale. Yale Journal on Regulation, forthcoming 2021.

Badawi, Adam B. and Dyreng, Scott and de Fontenay, Elisabeth and Hills, Robert, Contractual complexity in debt agreements: The case of ebitda, working paper (2021).

Bratton, William W., Bond covenants and creditor protection: Economics and law, theory and practice, substance and process, 7 European Business Organization Law Review 39-87 (2006).

Chava, Sudheer and Roberts, Michael R., How does financing impact investment? The role of debt covenants, 63 The Journal of Finance 2085-21 (2008).

Choi, Stephen J. and Gulati, Mitu and Scott, Robert E., The black hole problem in commercial boilerplate}, 67 Duke Law Journal 1-77 (2017).

Choi, Stephen J. and Gulati, Mitu and Scott, Robert E., Variation in boilerplate: rational design or random mutation?, 20 American Law and Economics Review 1 (2018).

Choi, Stephen J. and Gulati, Mitu and Scott, Robert E., Are M&A Lawyers Really Better?, working paper (2021).

Coates, IV and John, C., Why Have M&A Contracts Grown? Evidence from Twenty Years of Deals, working paper (2016).

Grossman, Sanford J and Hart, Oliver D, The costs and benefits of ownership: A theory of vertical and lateral integration, 94 Journal of Political Economy 691-719 (1986).

Ivashina, Victoria and Vallee, Boris, Weak credit covenants, working paper (2020).

Jehiel, Philippe and Newman, Andrew F., Loopholes: Social Learning and the Evolution of Contract Form, working paper (2011).

Jennejohn, Matthew and Nyarko, Julian and Talley, Eric L, Contractual evolution, working paper (2021).

Jensen, Michael C and Meckling, William H, Theory of the firm: Managerial behavior, agency costs and ownership structure, 3 Journal of Financial Economics 305-360 (1976).

Kahan, Marcel and Klausner, Michael, Standardization and Innovation in Corporate Contracting (or "The Economics of Boilerplate"), 83 Virginia Law Review 713-70 (1997).

Mitchell, Melanie, Complexity: A guided tour, Oxford University Press (2009).

Murfin, Justin, The Supply-Side Determinants of Loan Contract Strictness, 67 The Journal of Finance 1565-1601 (2012).

Nini, Greg and Smith, David C and Sufi, Amir, Creditor control rights and firm investment policy, 92 Journal of Financial Economics 400-20 (2009).

Prat, Julien, Dynamic contracts and learning by doing, 9 Mathematics and financial economics 169-193 (2015).

Rauh, Joshua D. and Sufi, Amir, Capital structure and debt structure, 23 The Review of Financial Studies 4242-80 (2010).

Richman, Barak, Contracts Meets Henry Ford, 40 Hofstra Law Review 77-86 (2011).

Talley, Eric L., Discharging the Discharge for Value Defense, working paper (2021).

Simon, Herbert A., Theories of bounded rationality, 1 Decision and Organization 161-76 (1972).

Williamson, Oliver E., Markets and hierarchies: some elementary considerations, 63 The American Economic Review 316-25 (1973).

Appendix

How the Simulation Operates

Each run begins with a population of 200 agents that have randomly generated contract terms. Each term in the contract is a digit that ranges from 0 to 5. These six possibilities specify one of the six possible actions: move north, move east, move west, move in a random direction, or autonomy. The agent is "aware" of five spaces at any point on the board: the square it is on and spaces to its immediate north, east, south, and west. Those five spaces can take on one of three values: it can be a wall, it can have an investment opportunity, or it can be empty. This combination means that there are 5^3=243 situations that agent can find itself in, and, thus, the contract contain 243 genes. Of those, only 128 genes are situations that are possible (it is not possible, for example, to have walls on all four sides).

Each of the 200 agents begins on a randomly generated 10x10 board. The board is surrounded by a wall (i.e. if the agent tries to move east when there is a wall to the immediate east, he will remain in the same square). There are parameters that specify the chance that any given square will contain a good opportunity, which we set to ten percent. Bad opportunities appear systematically to the east of good opportunities with two exceptions. The first is when two or more good opportunities appear in adjacent horizontal squares. In that case, we place a bad opportunity to the east of the easternmost good opportunity. The second is when a good opportunity is placed at the eastern edge of the board. In that case, we place a bad opportunity in the westernmost square of the same row.

Each agent navigates the board for a specified number of moves (50) and does this for a specified number of tries (50). Each try uses a new randomly generated board. The fitness of each agent is averaged over all of its tries and this score is used to calculate the possibility that a given agent's genes will propagate to the next generation. That calculation is based on the normalized value of the each agent's fitness. The program randomly selects two parent agents based on those probabilities and then splits each parent's DNA at a random point before splicing them together. The program then mutates the genes of the "child" agents with a probability of .001. If selected for mutation, the gene gets replaced with a random gene with the probability that the gene will be autonomy dependent on the percentage of times autonomy appears in the child agent's contract. This new generation of agents repeats the process until the simulation reaches the specified number of generations (300).

Variable Definitions

Fitness – The primary measure of performance and the one that is used to determine how likely genes from an agent are to get passed to the next generation. For each individual agent, it is the score assigned to good opportunities (call this g) times the number of good opportunities taken (N_g), less the score assigned to bad opportunities times the number of bad opportunities taken (N_r): $F = gN_g - rN_r$.

Average Good Opportunities-The average number of good opportunities taken by the agent in a generation.

Average Bad Opportunities-The average number of bad opportunities taken by the agent in a generation.

Average Unique Squares- The average number of unique squares that each agent touches on the 10x10 board during a generation.

Average Scenarios Faced- The average number of scenarios above .01 that the agent faces in the 50 tries it makes in a generation.