Computer Aided Tax Evasion Policy Analysis: Directed Search using Autonomous Agents

Jacob Rosen
MIT
32 Vassar St
jbrosen@mit.edu

Erik Hemberg
MIT
32 Vassar St
hembergerik@csail.mit.edu

Geoff Warner
MITRE CORPORATION
McLean, VA 22102
gwarner@mitre.org

Sanith Wijesinghe
MITRE CORPORATION
McLean, VA 22102
sanith@mitre.org

Una-May O'Reilly
MIT
32 Vassar St
unamay@csail.mit.edu

ABSTRACT

Abusive tax shelters implemented through partnerships and S corporations have become increasingly popular amongst tax planners, helping high-income taxpayers to underreport an estimated $91 billion of income annually in the US alone [4]. The most challenging problems for tax collection agencies in this respect are *a*) the recent upswing in large, tiered partnership structures and *b*) the evolving nature of tax evasion schemes in response to auditing policy.

By representing tax evasion schemes as sequences of financial transactions, we are able to conduct a directed combinatoric search that can find effective abusive tax shelters, given an initial ecosystem of taxable entities and their respective portfolios. Assigning auditing likelihoods to certain types of transactions allows us to consider policies that would result in increased compliance. We accomplish this by considering each tax plan and auditing policy as individual agents and conducting a search over them with a genetic algorithm.

We demonstrate the ability of the system to accurately model tax liability in financial scenarios through experiments run on a tax shelter known as Distressed Asset Debt (DAD).

Categories and Subject Descriptors

M.2.5.1 [**Social and professional topics**]: Taxation

General Terms

Algorithms, Economics

Keywords

tax evasion, grammatical evolution, genetic algorithms, auditing policy, agent-based modeling

1. INTRODUCTION

The Government Accountability Office (GAO) estimates that roughly $91 billion of income is misreported by partnerships and S corporations annually [4]. These types of businesses are particularly attractive to tax planners because they are characterized as "pass-through" entities, meaning that the shareholders, not the corporation itself, are responsible for any tax liability that it takes on. Thus, auditing these corporations can become extremely difficult for the IRS because it involves information regarding both the corporation and each individual shareholder. This logistical reality results in a large set of convoluted tax rules, definitions and exceptions. With some of the largest partnerships in the country containing upwards of $20,000$ partners [5], obfuscating taxable income can become commonplace. Additionally, many of the shareholders in these complex partnership structures are themselves partnerships or other pass-through entities,

We focus primarily on tax evasion schemes that attempt to offset real gains in a taxpayer's portfolio by acquiring assets with a large built-in loss, or artificially stepping up the basis in previously owned assets. When the financial documents are filed, it appears as though the taxpayer incurred substantial losses, which can cancel out the income generating gains elsewhere in their portfolio. Generally, tax shelters that require the utilization of multiple partnerships are planned and implemented by professional *tax shelter promoters.*

Furthermore, whenever the IRS finds a strategy to successfully audit or disallow tax benefits from abusive tax shelters, a new tax shelter emerges that, while similar to the previous iteration, is undetectable by the IRS [20]. For example, when an IRS notice was issued that disallowed tax benefits gained from the Distressed Asset Debt (DAD) scheme, explained further in section 4, a new tax shelter quickly arose that was nearly identical, except made use of trusts rather than partnerships to disguise taxable gain. The sheer number of clauses within the Internal Revenue Code seem to allow tax shelter promoters to subtlety permute citations or justifications to avoid IRS scrutiny.

Prior analytic models of tax evasion focus on macroeconomic parameters such as GDP growth or the tax rate that incentivize taxpayers to turn to tax shelter promoters [15]. While these statistical models provide valuable insight into measures that Congress can take to mitigate abusive tax shelters, they provide no information that the IRS could use to improve their ability to detect abuses of the tax code and subsequently alter their policy directives.

Conversely, we take a microeconomic approach that focuses on the mechanics underlying the ability to evade tax. By treating tax evasion schemes as agents and calculating the taxable income that they generate as described further in section 3, we can determine the structure of the most effective schemes.

Furthermore, tax evasion schemes lend themselves well to computational representation because they are generally composed of multiple accounting rules that, while simple individually, can generate complex results [14], discussed further in section 5. Thus, a process that can quickly become overcomplicated for even highly experienced tax professionals is a simple task for a properly configured computer model. Here we extend a previous attempt to model the human process of inventing tax evasion schemes and determining audit observables[19].

Complementing the generation of effective tax evasion schemes is our treatment of IRS policies as agents. That is, we assume that within the tax ecosystem, there exist a list of *observables* that policy-makers use to determine whether an audit should be conducted. Each agent is then a list of numerical weights, each associated with a different observable, that represents the relative likelihood that the observable is indicative of abusive behavior, discussed further in section 3.

Our representation of auditing policy mirrors "IRS notices", that are the Internal Revenue Service's primary form of creating new policy. For example, IRS notice 2005-32 required a mandatory basis adjustment in scenarios that were commonly found in the DAD schemes, which strongly contributed to its disappearance [9]. These notices usually describe a scenario that will result either in *a*) a disallowance of tax benefits or *b*) legal action. Typically, many aspects of an abusive tax shelter can be characterized by a list of events that compose such a scenario.

This method, which we refer to as **SCOTE**, allows us to construct policy suggestions by determining which combinations of indicators are highly correlated with large losses, as explained further in section 3.2. The goal is to characterize classes of tax evasion schemes by the presence of a discrete set of observable features, which auditing agencies can use to make more effective policy.

2. BACKGROUND

Before describing our methodology, we will discuss 1. previous quantitative studies and models of tax evasion behavior and 2. basic mechanics regarding taxation of assets.

2.1 Previous Work

Many previous economic models have confronted the issue of individuals engaging in tax evasion schemes, focusing primarily on the effects that tax policy has on the incentive to evade. For example, one study casts agents as either honest, imitative or free riders, and a GA is used to update the population's utility function [15]. Yet another uses an agent based model to construct a game theoretic approach of tax evasion that results in cyclical compliance behavior [12], but does not contain any policy suggestions to increase compliance. Several other attempts have been made to investigate how psychological inclinations towards tax evasion are affected by various auditing policies [3, 7, 8, 17]. Recent agent based models have further analyzed the effect of social network structures on the occurrence of tax evasion [1, 10].

SCOTE is distinguished from previous studies due to our focus on the effect that auditing policy has on the *composition of tax evasion schemes*, rather than their occurrence. While studying the implications of federal tax policy on noncompliance is a worthwhile endeavor, little work has been done on how auditing agencies can change their policy in the short term to increase the efficiency of their auditing procedure.

2.2 Asset Taxation

Crucial to the implementation of our model is the treatment of asset transactions within the tax law. While our computation follows specifically the US tax code, many of the tax concepts can be easily extended to other countries' protocols.

We focus on tax liability incurred during the sale and trade of investment property, which we assume is all taxed at the same rate. An asset has two main fields that must be defined: *a)* **Fair Market Value** (FMV) is the value of an asset at a given time and *b)* **Basis** is meant to represent the price at which the asset was originally acquired by the current owner. When an asset is sold or exchanged for another asset, the seller must recognize either a *gain* or a *loss* on the sale, which is the difference between the assets FMV and its basis. If the FMV exceeds the basis, then the taxpayer recognizes a *gain* on the transaction, and is thus added to their taxable income. Conversely, the taxpayer recognizes a *loss* on the transaction when the basis exceeds the FMV. In the cases that we study, that loss can then be *deducted* from the taxpayers taxable income. For example, if a taxpayer sold asset A for a gain of \$100 and asset B for a loss of \$80, their total taxable income would come to \$20.

2.2.1 *Partnerships and Carryover Bases*

Upon being sold, the new basis of an asset will generally become the price that was paid for it. But there are many cases in which an asset is transferred from one entity to another and the basis is *carried over* to the acquiring entity, particularly with transactions involving partnerships. For example, if Taxpayer A sells its share in Partnership P1, that has a basis of $\$40,000$ as shown in figure 2, for $\$80,000$ to Taxpayer B, then A recognizes $\$40,000$ as gain. The new basis in the share of P1 owned by B becomes $\$80,000$. Alternatively, if Taxpayer A were to have contributed the share in P1 to another partnership in exchange for a share, then A would *not* have recognized any gain and the basis in the partnership share would remain $\$40,000$.

The primary focus of our model is how tax shelter promoters can structure partnerships in order to manipulate the bases of certain assets that leave their clients in a more favorable tax position. An important tool in the construction of these structure is the choice that partnerships have to make a §754 election. Making this election allows partnerships to adjust the bases of their assets when certain events occur, mainly *a)* the distribution of assets to a partner or *b)* the transfer of a partnership interest. The decision to make a §754 election can result in drastically different tax consequences and whether or not the IRS disallows a transaction can depend on that decision. The pass-through nature of partnerships, combined with their ability to manipulate the bases in assets, makes them very useful to tax shelter promoters.

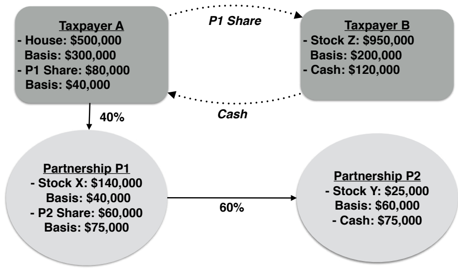

Figure 1: Pre-Transaction

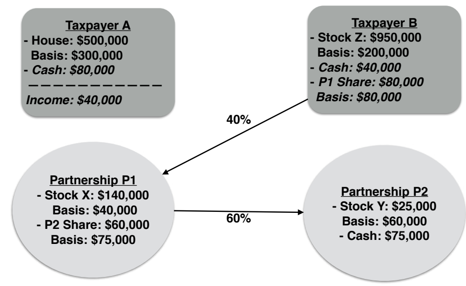

Figure 2: Post-Transaction

3. METHOD

The overarching model, which we will refer to as SCOTE, has three levels that must be abstracted before we can conduct a directed search *1* the representation of tax evasion schemes as *transaction sequences 2* the representation of auditing policy as *audit score sheets 3* the representation of the tax law as a series of *decision rule trees*

Each tax evasion scheme must be evaluated against the accounting logic built into the model. Auditing policy depends on certain events that occur within a given tax evasion scheme. The tax law determines which events the auditing policy is able to observe. While all levels are highly interconnected with each other, their abstract representations must be considered separately. Each level is described in more detail in this section.

3.1 Transaction Representation

We represent the tax ecosystem as a *graph*. The *nodes* in the graph represent tax entities while the *edges* represent ownership relations between those entities. The state of the model is changed by a transaction, a pair of actions in opposite directions. An action transfers an asset from one entity to another entity, which subsequently updates the state of the graph. Moreover, each entity owns a portfolio of assets. An asset can only reside in a portfolio and it is transferred from the portfolio of one entity to the portfolio of another entity. The graph makes our design modular. We can add different kinds of entities by introducing more nodes in the graph and similarly we can introduce more diversity within nodes by having different kinds of assets.

Figure 1 shows an example graph with four nodes and their asset portfolios'.

Taxpayer A owns 40% of partnership P1 that has an FMV $\$80,000$ and a basis of $\$40,000$ and wants to sell his interest to Taxpayer B, who is willing to pay him in cash. Taxpayer A is connected to P1 by a link, and P1 is subsequently connected to P2 by a link.After the transaction occurs, figure 2 shows that Taxpayer A has $\$80,000$ of cash in his portfolio, as well as $\$40,000$ in taxable income to reflect the difference between the purchase price of his interest in P1 and its basis. Additionally, Taxpayer B has $\$80,000$ less cash in her portfolio, is now connected to P1 through a link and her basis in the share of P1 is $\$80,000$ to reflect the price she paid for it.

3.1.1 Transaction Formalism

Mathematically, the state of the model can be described as some $\gamma \in \Gamma$, such that

$$\gamma = \{\mathbf{e}, \mathbf{a}, d\}$$

where $\mathbf{e} = \{e_i\}_{i=0}^{k_1}$ is the set of k_1 entities , $\mathbf{a} = \{a_i\}_{i=0}^{k_2}$ is the set of k_2 assets, and $e_i \in E$ and $a_i \in A$. The operator d determines ownership relations between entities and assets, i.e. $d : A \mapsto E$, where A is the space of assets and E is the space of entities.

We then define a list of transactions as a vector $\mathbf{t} = \{t_i\}_{i=0}^{k}$ for some $k \in \mathbb{Z}_+$, such that $t_i \in \bar{T}$ is the space of all transactions. A transaction here is defined as

$$t = \{e_f, e_t, a_f, a_t\}$$

where $e_f, e_t \in E$ and $a_f, a_t \in A$ are two entities and two assets.

3.2 Auditing Policy Representation

The IRS issues tax guidance on matters related to regulations, revenue rulings and revenue procedures using a number of announcements and notices. These collective communications can be used to clarify the intent of the tax code and determine specific transactions and/or transaction types deemed in violation of certain regulatory statutes. In SCOTE, the IRS audit priorities are modeled based on this public information.

For example, a recent amendment to the IRC altered §743(a) to read

> The basis of partnership property shall not be adjusted as the result of (1) a transfer of an interest in a partnership by sale or exchange or on the death of a partner unless (2) the election provided by section 754 (relating to optional adjustment to basis of partnership property) is in effect with respect to such partnership or (3) unless the partnership has a substantial built-in loss immediately after such transfer.

where we have added numbers in parenthesis to signify *observable* events. This amendment is captured across as *1* The sale of a partnership interest in exchange for a *taxable* asset *2* The partnership whose shares are being transferred has not made a 754 election *3* The seller's basis in respect to the non-cash assets owned by the partnership exceeds their FMV by more than $\$250,000$

Our approach to represent audits in SCOTE is to use a list of audit points (scalar weights), corresponding to all observable events[1] that can occur when a list of transactions is executed. An audit score sheet is a collection of audit points, each corresponding to a different type of observable that may be present in a transaction. The higher the audit

[1]These events may be captured on a tax return or require supporting documentation to substantiate during a formal audit.

points associated with a certain type of event, the more suspicious that type of event is. The audit points aim to model the work of an actual auditor. In order to mirror the limited resources available for auditing, we constrain the sum of audit points to equal one. We define the sum of all of the audit points present in a list of transactions, multiplied by their respective frequencies, as the audit score associated with *a)* a list of transactions *b)* an audit score sheet. Visually, audit score sheets can be represented by a spreadsheet, with n rows corresponding to a different type of audit priorities as shown in Table 1. One can imagine a hypothetical auditor going through a list of transactions and incrementing the frequency in the far right column whenever each type of event is observed.

Observable	Points	Frequency
1	$Point_1$	$Frequency_1$
2	$Point_2$	$Frequency_2$
3	$Point_3$	$Frequency_3$
$1 \cup 2$	$Point_{1\cup 2}$	$Frequency_{1\cup 2}$
$1 \cup 3$	$Point_{1\cup 3}$	$Frequency_{1\cup 3}$
$2 \cup 3$	$Point_{2\cup 3}$	$Frequency_{2\cup 3}$
$1 \cup 2 \cup 3$	$Point_{1\cup 2\cup 3}$	$Frequency_{1\cup 2\cup 3}$

Table 1: Each row has three columns with 1) the type of observable corresponding to the three characterized observables from the IRS notice, 2) the associated audit point and 3) the number of times it occurs in a list of transactions

Using this formulation, we define that an audit score is interpreted as the *likelihood* that a list of transactions will be audited. That is, the more types of events associated with high levels of suspicion there are in a list of transactions, the higher the audit score will be.

The observables on the audit score sheet can range from basic facts about a transaction, such as whether a material asset is being exchanged, to more complex aspects of the model state, such as ownership linkages between multiple entities.

3.2.1 Auditing Policy Formalism

Suppose that there are n types of events that are detectable by an auditor, represented by $\{b_i\}_{i=0}^n$. Associated with each type of event b_i is an audit point $\alpha_i \in \mathbb{R}_+$ and the frequency that the event occurs within a list of transactions $f_i \in \mathbb{Z}_+$. We can then write the audit score s, corresponding to a list of transactions and list of audit points $\{\alpha_i\}_{i=0}^n$ as

$$s = \sum_{i=0}^{n} \alpha_i * f_i \text{ where } \sum_{i=0}^{n} \alpha_i = 1$$

3.3 Tax Law Representation

The tax law is broken down into three parts for a transaction. We need to check if *a)* each transaction is legal *b)* how the transaction alters the bookkeeping (state of the graph) and *c)* the taxation of the transaction,

Feasibility of a transaction Check the rules regarding the feasibility of a transaction. A transaction consists of two actions, one action transfers an asset from one entity to another entity, and the other action transfers an asset in the opposite direction. The transaction is infeasible if there is a logical error in the transaction, e.g an entity cannot give an asset it does not own. Conversely, a transaction is illegal if it is in theory possible, but illegal under the tax law, such as obtaining a cash distribution from a partnership immediately after contributing assets to it.

Transfer of assets Check the rules regarding the transfer of assets. E.g. when a partnership asset is transferred, how should the basis of the underlying assets be adjusted?

Calculate the taxable gain/loss from the transaction Check the rules regarding the taxable gain/loss calculation of the transaction. E.g. when an entity exchanges an annuity no taxable gain or loss can be incurred

3.3.1 Tax Law Formalism

In order to formalize the way in which tax law is applied to the model, we make the observation that laws governing a transaction depend on the "type" of assets and entities being exchanged. For example, the laws governing the exchange of a hotel for cash between two taxpayers are different from those governing the contribution of an annuity to a partnership in exchange for a share. Thus, we can determine the laws governing a given transaction by the combination of both asset and entity types.

Consider the abstract transaction $t = (e_f, e_t, a_f, a_t)$, which states that entity e_f gives e_t the asset a_f in exchange for a_t. Given our previous observation, the laws governing the legality, asset transfer and tax calculation of the transaction are determined by the combination of asset and entity "types".

Define $\hat{E}$ to be the finite set of entity *types*, and $\hat{A}$ to be the finite set of asset *types* and let $\overline{T}$ be the set of all transactions. We can then write the set of all transactions as a union of disjoint subsets $\overline{T} = \cup_{i=0}^n T_i$, where each subset contains all transactions of a certain combination of asset and entity types. The steps that follow can be described as below.

1. a transaction type t is checked to see if it is within the bounds of the legal/feasible region by first determining to which subset $\overline{T}$ it belongs. We define $\mu : T_i \mapsto \Phi$ as a map from a subset $T_i \in \overline{T}$ to Φ that determines the laws ϕ that govern the transaction, given it's combination of asset/entity types.

2. if the transaction is deemed to be legal/feasible under the applicable tax laws, then the model state transitions from γ_t to γ_{t+1}.

3. taxable gain/loss calculation takes a transaction t and a model state γ_t and maps it to a recognizable loss value r_l for each taxable entity and an updated model state.

3.4 Directed Search Representation

Given an initial collection of entities and assets, we search over one of two solutions: *1*) the transaction sequence that yields the lowest taxable income, given a specific audit score sheet, or *2*) the audit score sheet that results in the highest likelihood of auditing an abusive tax evasion scheme, given a specific list of transactions.

In order to generate both a final taxable gain/loss value and an audit score, the tax simulator in SCOTE must take both a transaction sequence agent *and* an auditing policy agent as inputs. As each transaction in the list is executed, the tax simulator calculates the audit score for the types of financial events indicated on the audit score sheet.

Given that the goal is to show that optimal transaction sequences and auditing policies can be found, different search heuristics can be used. In situations with a small search space, a brute force method may be more appropriate. Conversely, our numerical representation of transaction sequence and auditing policy agents allows us to utilize a wide array of alternative search algorithms such as hill climbing or other population based searches. But we find that the size of our search space, along with the antagonistic relationship between tax evasion schemes and auditing policies, lend themselves well to a genetic algorithm approach.

3.4.1 Evolutionary Search using a Grammar

A tax scheme or audit score sheet generated by the GA is represented by a list of integers. A parser is used to read these integers and generate a list of transactions or audit point distribution with the help of a *grammar*. The output consist of a list of Java interpretable objects that are input to the tax simulator to calculate the resulting taxable income and audit likelihood. The parser in this case bridges the gap between the GA and the tax simulator.

In order to implement our evolutionary search across tax evasion schemes, it is necessary to create a mapping between the evasion schemes and a corresponding numerical representation. This is accomplished using a BNF-grammar that maps a numerical input (genotype) to the representative output (phenotype), as seen in figure 3. The method of implementing a GA with a variable length representation and a grammar is known as Grammatical Evolution (GE) [16].

In GE, a grammar defines the language that describes the output sentences that can be produced. A grammar has terminal symbols, non-terminal symbols, a start symbol and rules for rewriting non-terminal symbols to terminal symbols and non-terminal symbols. The grammar is used in a generative approach and the production rules are applied at each stage of a derivation process, starting from the start symbol, until a complete program is formed. The mapping (derivation) is complete when the sentence is one that is comprised of only terminal symbols.

3.4.2 Agent Application to Graph State Formalism

We can now begin to fully describe the progression of the model within the search heuristic as a function $\mathbf{F} : \mathbf{T} \times \Gamma \times \Psi \mapsto \mathbb{R}_+^2$ that takes as input a list of transactions, an initial graph state and an auditing policy, and generates a recognizable loss value and audit score. Contained within the model state is the *recognizable loss* r_L. In other words, for any $\mathbf{t} \in \mathbf{T}$ and $\gamma_0 \in \Gamma$ generated from the same vector of integers $\mathbf{x}$ and an accompanying auditing behavior $\psi \in \Psi$,

$$\mathbf{F}\left(\mathbf{t}, \gamma_0, \psi\right) = \left(r_L, s\right)$$

where s is the audit score defined in 3.2.1. The function $\mathbf{F}$ can be broken up into a list of transition functions that has the same length as the number of transactions in the transaction set contained within the function call (k). Each transition function generates a new model state and an audit score. So for all $i \in [0, k]$,

$$F_i\left(t_i, \gamma_i, \psi\right) = \left(\gamma_{i+1}, s_i\right) \quad \text{where} \quad s = s_k$$

3.4.3 Grammar Formalism

The grammar that generates lists of transactions and initial model states is defined as

$$\Xi_t : \mathbb{Z}_+^n \mapsto \mathbf{T} \times \Gamma$$

that maps a list of n integers to an element in the set of lists of transaction ($\mathbf{T}$) and an element in the set of all model states (Γ). Thus, for any $\mathbf{x} \in \mathbb{Z}_+^n$,

$$\Xi_t(\mathbf{x}) = (\mathbf{t}, \gamma_0)$$

where $\mathbf{t} \in \mathbf{T}$ is a list of transactions and $\gamma_0 \in \Gamma$ is an initial model state.

We can now define the space of auditing behavior as Ψ, where for some $m \in \mathbb{Z}_+$,

$$\Psi = \{\{b_i\}_{i=0}^m : b_i \in [0,1] \text{ and } \sum_{i=0}^m b_i = 1\} \subset \mathbb{R}_+^m$$

The grammar $\Xi_a : \mathbb{Z}_+^m \mapsto \Psi$ maps a vector $\mathbf{y} \in \mathbb{Z}_+^m$ to an element in the set of auditing behavior.

3.4.4 Objective Function

The goal of the tax evader is to minimize audit likelihood and maximize recognizable loss. Thus, we can represent the objective function, h for a tax evasion scheme, given a specific audit score sheet, as

$$h = r_l(1 - s)$$

Note that the objective is positively correlated with the *recognizable loss*, which is to be expected. The second term in the function represents the likelihood of the audit disallowing the tax benefits gained from the scheme, which takes into account not only the likelihood of an audit (*audit score*), but also the amount of tax that is evaded. In this way, we are able to take into account both the effectiveness of a tax evasion scheme from a purely tax perspective, as well as from a risk perspective.

The goal of the auditor is to maximize the likelihood of an audit of a list of transactions with high recognizable loss. The objective function for an audit score sheet given a specific tax evasion scheme is the same as that shown above, but with the opposite sign

$$h = -r_l * (1 - s)$$

An audit score sheet is fit for a specific evasion scheme if either *1)* there is a high level of taxable gain *2)* if there is a high likelihood that if not much tax is collected, then the scheme will be audited

3.4.5 Directed Search Formalism

Define the objective function $h : \mathbb{R}_+^2 \mapsto \mathbb{R}$ as such

$$h\left(r_l, s\right) = r_l(1 - s)$$

It is now possible to fully define the maximizing objectives of transaction sequence agents as

$$\underset{\mathbf{t}^* \in \mathbf{T}, \gamma_0^* \in \Gamma}{\arg\max} \left[h\left(\mathbf{F}\left(\hat{\mathbf{t}^*}, \gamma_0^*, \psi\right)\right)\right]$$

over all $\mathbf{y} \in B(\hat{\mathbf{y}}, r_1)$ for some $\hat{\mathbf{y}} \in \mathbb{Z}_+^m$, where $B(\hat{\mathbf{y}}, r_1)$ is a *ball* of radius $r_1 \in \mathbb{R}_+$ around $\hat{\mathbf{y}}$. This represents the fact

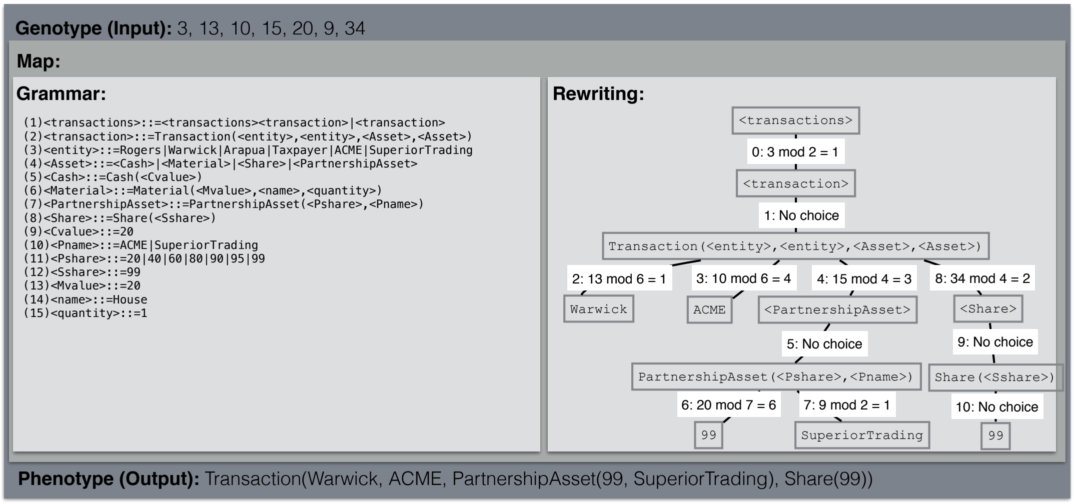

Figure 3: Example of mapping a list of integers (Genotype) into a list of transactions (Phenotype) by using GE

that the goal of the GA is to find the locally optimal auditing behavior agent around some subset of auditing behavior agents, rather than attempting to search the entire Φ space.

The objective of the auditing behavior agents is to maximize the *positive* h function, the opposite of the objective for the transaction sequence agents, i.e. the goal is

$$\arg\max_{\psi^* \in \Psi} \left[-\ h \left(\left(\mathbf{F} \left(\hat{\mathbf{t}}, \gamma_0, \psi^* \right) \right) \right] \right.$$

over all $\mathbf{x} \in B(\hat{\mathbf{x}}, r_2)$ for some $\hat{\mathbf{x}} \in \hat{X}$, where $B(\hat{\mathbf{x}}, r_2)$ is a *ball* of radius $r_2 \in \mathbb{R}_+$ around $\hat{\mathbf{x}}$. Similar to the previous objective function, this represents the fact that the GA only searches for local maxima around a subset of all transaction sets and initial model states.

4. EXPERIMENTS

Here we demonstrate with a GA that we can find optimal transaction sequences and auditing policies, given an initial graph state and the respective objective functions described in 3.4.4. While the efficiency of the search is an important concern, we are more interested in exploring the search space of SCOTE rather than any particular result. Thus, we chose to explore a partnership structure involved in a canonical tax evasion scheme to evaluate SCOTE's ability to generate transaction sequences and auditing policy distributions.

The Distressed Asset Debt transaction, or DAD, is an abusive tax evasion scheme that was disallowed in 2004 by the US government, primarily by issuing the IRS notice discussed in 3.2. We hypothesize that given reasonable constraints, the directed search will find both the DAD scheme and the audit point distribution that represents the IRS notice. A simplified version of the scheme operates as follows:

A Brazilian retailer named Parua had overextended credit to many of its customers and is forced to go through a bankruptcy reorganization. The overextended credit in the form of trade receivables are called *distressed assets*, which are assets with a negligible FMV but a very high basis ($30 million in this case). Parua then contributes the distressed assets to a partnership Samarth, which is directly controlled by a tax shelter promoter Smith and has *not* made a §754 election. Samarth subsequently contributes the same trade receivables to another partnership Superior Trading in exchange for a 99% interest, as seen in figure 4. Because

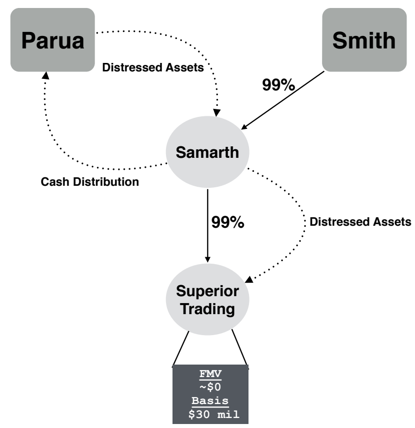

Figure 4: DAD Setup

both exchanges were contributions, which are non-taxable events, the basis in the distressed assets *carries over* [13].

In the next transaction, Samarth contributes its 99% interest in Superior Trading to another partnership ACME Co. which had also not made a §754 election, in exchange for a 99% share in ACME Co, shown in figure 5. Again, this is a non-taxable event so there are no bases or tax liabilities are affected.

Finally, as shown in figure 6, an outside taxpayer who has previously arranged with Smith to implement a tax minimizing scheme purchases Samarth's share in ACME for $1 million. Because ACME had not made a §754 election, the basis in its share of Superior Trading, as well as the basis of the distressed assets, were not affected. Thus, when Parua ultimately claims that the debts are not collectible, it appears as though the taxpayer incurred a $29 million loss and is able to deduct that from their taxable income.

4.1 Results

The search heuristic in SCOTE was implemented by extending an existing GA library (EVOGPJ). The GA performs a search on lists of transactions to find the specific sequence of transactions or distribution of audit weights that maximize a fitness score. For both experiments we ran the model 100 times, each for 100 generations with a population

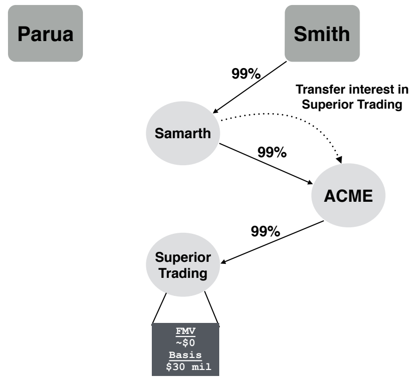

Figure 5: DAD Step 1

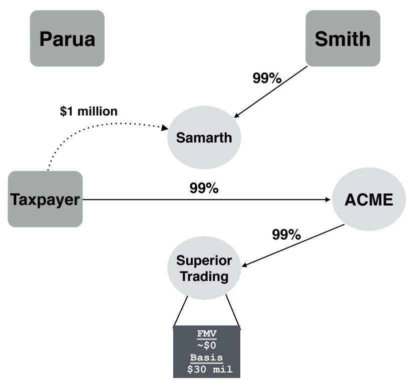

Figure 6: DAD Step 2

size of 500 and implemented with the grammar shown in figure 3. The crossover rate was 0.7, the mutation rate 0.1, and both tournament selection and elite replacement were set to 2.

First, we search for the DAD scheme, which consists of three transactions: *1* Parua's contribution of the distressed assets to Samarth *2* Samarth's contribution of the distressed assets to Superior Trading in exchange for a partnership interest *3* Samarth's contribution of its interest in Superior Trading to ACME in exchange for an interest in ACME followed by the realization that the distressed assets are non-collectable, which from a tax perspective is equivalent to the sale of the distressed assets to a third party.

We were able to generate the transactions in a portion of the runs, which serves as a strong proof of concept.

The next step is to test if the proper audit point distribution can be generated. We include the three separate observables from the IRS notice discussed in section 3.2.1, as well as their joint probabilities, which are illustrated in table 1.

When we evolve a population of audit score sheets with the 7 specified events, we find that all of the audit points converge to zero, except for the point associated with all three events occurring simultaneously, which converges to one, as shown in figure 7. This indicates that we are able to evolve the IRS notice because a transaction only arouses suspicion when all three events are present. Thus, we are effectively able to find both the DAD scheme and the distribution of audit points necessary to detect it.

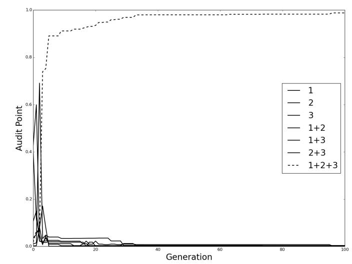

Figure 7: Example run of evolved audits

5. DISCUSSION

A common anecdote regarding manipulation of the tax code involves the childhood "no backsies" rule.[2] The rule stipulates that if there is a line of children, then one can allow their friend to enter the line in front of, but not behind them. The moral justification for the rule is that if everyone in the line suffers from the extra wait time, then the child that let their friend cut in line should suffer as well. But this rule is easily evaded if, immediately upon letting their friend cut in front of them, the child exits the line. In turn, the child's friend allows them to legally cut in line in front of them, effectively engineering a "backsie" from two legal actions.

Essentially, the goal of professional tax shelter promoters is to find analogous engineering techniques within their jurisdiction's tax law. By separately representing multiple aspects of the tax law, we can construct tax plans that are composed specifically to generate favorable tax treatment for the involved parties without regard to the intent behind any of the individual statutes.

In this way, the model serves as a useful tool for policy-makers to understand how taxable income flows through complex partnership structures. There is substantial prior work on how computer modeling aids the learning process [11, 2, 18], all of which indicate that abstractly representing a complex system can be the most effective way to learn about it. Furthermore, calculating taxable income through complex partnership structures falls into the category of *conceptual problems*, which lend themselves particularly well to learning through computer modeling [6].

Determining potential tax evasion schemes given a complex partnership structure poses highly non-linear behavior. The calculation of taxable income given a list of transactions is a complicated process on its own, especially when considering tiered partnership structures. Thus, deducing transaction sequences that yield the lowest taxable income can become extraordinarily difficult without the use of computational techniques. By treating each transaction sequence as an agent and formulating sensible objective functions, we have created a methodology for conducting a directed search over a theoretically infinite search space.

Policy-makers, as well as tax professionals in private practice, could greatly benefit from the use of these computational techniques. Many implications of complex partnership structures are unknown, given the computational complexity involved in tax calculations. An agent-based mod-

[2]Taken from a discussion with Ameek Ponda J.D., LL.M. of Sullivan & Worchester on October 28, 2014

eling approach will allow policy-makers to determine what types of abusive behavior are possible within such structures. Additionally, the inclusion of audit likelihood in the tax plans' objective functions let policy-makers evaluate potential responses to changes in auditing policy.

6. CONCLUSIONS AND FUTURE WORK

We present a methodology and functional computer model to both *a*) calculate taxable income and *b*) find potentially abusive tax-minimizing techniques within complex partnership structures. Given the recent surge in tax abuse associated with such structures, as well as the computational difficulty in an intuitive approach, an agent-based modeling approach yield many benefits.

While many have used quantitative methodologies to explore effective tax evasion policy, prior attempts have focused primarily on the tendency that taxpayer have to engage in tax evasion. Conversely, we are interested not in the incentives that individuals have to evade tax, but how tax law and accounting rules lend themselves to the generation and implementation of the evasion schemes.

Our experiments showed that our representation provides a means to explore the result of various transaction sequences and auditing policies, as well as being able to search for the optimal evasion scheme or audit score sheet agent. This is accomplished by modeling separate aspects of the tax law and iterating over various combinations to determine the resulting taxable income, as well as the financial observables that result in significant tax savings.

In the future, we plan on representing additional tax evasion schemes such as Custom Adjustable Rate Debt Structure (CARDS). Adding new schemes to our representation will require us to increase the scope of our taxable gain calculations to include additional accounting artifacts. This process will add new artifacts of the law that will subsequently increase the space of potential tax evasion schemes.

7. REFERENCES

[1] A. L. Andrei, K. Comer, and M. Koehler. An agent-based model of network effects on tax compliance and evasion. *Journal of Economic Psychology*, 2013.

[2] P. Bilkstein and U. Wilensky. Less is more: Agent-based simulation as a powerful learning tool in materials science. In *IV International Joint Conference on Autonomous Agents and Multiagent Systems*. AAMAS, 2005.

[3] J. S. Davis, G. Hecht, and J. D. Perkins. Social behaviors, enforcement, and tax compliance dynamics. *The Accounting Review*, 78(1):39–69, 2003.

[4] GAO. http://www.gao.gov/assets/670/663185.pdf, 2014.

[5] GAO. Large partnerships: Growing population and complexity hinder effective irs audits. http://www.gao.gov/products/GAO-14-746T, 2014.

[6] S. Gulwani. Example-based learning in computer-aided stem education. *Communications of the ACM*, 57(8):70–80, 2014.

[7] S. Hokamp and M. Pickhardt. Income tax evasion in a society of heterogeneous agents–evidence from an agent-based model. *International Economic Journal*, 24(4):541–553, 2010.

[8] S. Hokamp and G. Seibold. How much rationality tolerates the shadow economy?–an agent-based econophysics approach. In *Advances in Social Simulation*, pages 119–128. Springer, 2014.

[9] IRS. Interim rules and procedures for partnerships under section 833 of the american jobs creation act of 2004.

[10] A. Korobow, C. Johnson, and R. Axtell. An agent–based model of tax compliance with social networks. *National Tax Journal*, pages 589–610, 2007.

[11] C. Landstrom, S. Whatmore, and S. Lane. Learning through computer model improvisations. *Science, Technology & Human Values*, 38:678–700, 2013.

[12] V. Lipatov. Evolution of tax evasion. 2003.

[13] R. M. Lipton. Bad dad – seventh circuit rejects distressed asset/debt shelter. *Journal of Taxation*, 119(5):236–240, 2013.

[14] D. McBarnet. Legitimate rackets: Tax evasion, tax avoidance, and the boundaries of legality. *The Journal of Human Justice*, 3(2):56–74, 1992.

[15] L. Mittone and P. Patelli. Imitative behaviour in tax evasion. In *Economic simulations in swarm: Agent-based modelling and object oriented programming*, pages 133–158. Springer, 2000.

[16] M. O'Neill and C. Ryan. *Grammatical evolution: evolutionary automatic programming in an arbitrary language*, volume 4. Springer, 2003.

[17] M. Pickhardt and A. Prinz. Behavioral dynamics of tax evasion–a survey. *Journal of Economic Psychology*, 40:1–19, 2014.

[18] F. Stonedahl, M. Wilkerson-Jerde, and U. Wilensky. Re-conveiving introductory computer science curricula through agent-based modeling. In *Proceedings of the EduMAS workshop at AAMAS 2009, May 12*, pages 63–70. AAMAS, 2009.

[19] G. Warner, S. Wijesinghe, U. Marques, O. Badar, J. Rosen, E. Hemberg, and U.-M. O'Reilly. Modeling tax evasion with genetic algorithms. *The Economics of Governance*, 15(5), 2014.

[20] D. Wright Jr. Financial alchemy: How tax shelter promoters use financial products to bedevil the irs (and how the irs helps them), 2013.