New Dynamic Optimization Models for Tax Loss Valuation and Sourcing Problems

By Nilofar Varzgani
A dissertation submitted to the
Graduate School-Newark
Rutgers, The State University of New Jersey
in partial fulfillment of requirements
for the degree of
Doctor of Philosophy
Graduate Program in Management
Written under the direction of
Dr. S. Govindaraj
Dr. M.N. Katehakis
and approved by

Newark, New Jersey
October, 2017

ABSTRACT OF THE DISSERTATION

New Dynamic Optimization Models for Tax Loss Valuation and Sourcing Problems

By Nilofar Varzgani

Dissertation Directors:

S. Govindaraj

M.N.Katehakis

The dissertation comprises of three essays on multi-period dynamic optimization models. The first and second essays address the problem of valuing tax loss-carryforward(s) (TCLFs) and Carryback(s) (TLCBs) that arise frequently in theory and practice. While there have been a number of empirical papers that have shown TLCFs (TLCBs) are value relevant, there is little guidance on how to actually value these. The first essay introduces the valuation problem and provides a survey of literature in the area. The few analytical papers that have attempted to provide valuation formulas are highly stylized, do not capture the institutional complexity of the tax code, and are generally inadequate. In the second essay we develop a finite horizon discrete time stochastic dynamic programming framework for valuing TLCFs (TLCBs) that allows for piece-wise linear progressive taxation, and also incorporates many of the

institutional features of the tax code.

In the third essay we investigate a multi-product, multi-echelon contract manufacturer based business model. The decision problem faced by the manufacturing company is twofold: (a) how many contract manufacturers to get involved in business with (one supplier model vs a multi-supplier model, (b) how much volume should be allocated to each contract manufacturer, if the multiple supplier model is chosen. The objective is to maximize savings from the volume allocation process. Such problems arise often in many manufacturing industries such as electronics, pharmaceuticals, energy etc., for Original Equipment Manufacturing (OEM) firms. We provide a discrete time stochastic dynamic programming framework, and use numerical methods to study different volume allocation scenarios first in a one period setting, later extending the analysis to a multi-period model.

To Nana & Baba

Preface

This Ph.D. dissertation entitled "New Dynamic Optimization Models for Tax Loss Valuation and Sourcing Problems" has been prepared by Nilofar Varzgani during the period September 2010 to July 2017 at the department of Management Science and Information Systems at Rutgers University, Newark and New Brunswick.

The Ph.D. project has been completed under the supervision of my advisors Professor Suresh Govindaraj and Professor Michael N. Katehakis. The dissertation is submitted as a partial fulfillment of the requirement for obtaining the Ph.D. degree at the Rutgers University. The project was supported by a scholarship by Fulbright and an appointment as a Part-Time Lecturer at Rutgers Business School, Rutgers University.

First of all, I would like to express my most sincerest gratitude to my advisors Professor Michael Katehakis and Professor Suresh Govindaraj. and Michael Katehakis, for supporting me during these past few years. I truly appreciated the enormous amount of help and advise they gave me to get a Ph.D. degree. I will be forever very grateful to Professor Katehakis for being not only my academic father but also being a very supportive one at that. Michael is a master at simplifying complex problems, to approach them in the simplest way, thus helping me get over how overwhelming this process can be. He always says that the best solutions are the simple ones. Professor Suresh is someone you will never forget once you meet him. He is one of the smartest people I know. His vast knowledge base and intuition, which he shared with me over the past four years was of immense value. His strong work ethics have thought me to not only pursue my career with hard work and dedication but also with passion.

I would also want to specially thank the members of my PhD committee, Professors Lee Papayanopoulos (Late.), Spiros Papadimitriou and N.K. Chidambaran.

A special thanks to Luz Kosar, Goncalo Filipe and Monnique Desilva for all their support over these past few years. I am so very grateful to my family and friends (all across the globe), who advised me where necessary and supported me when times were tough. I would like to thank my parents and siblings (Mustay, Timtim, Humi) who have been my emotional backbone and supported me through out this degree. Without this support it would have been impossible to complete this dissertation. A big shout out to some very special friends for keeping me sane through out; Aiman, Rohit, Shalaka, Sanchit, Vishal, Richa, Nikita, Couz, Sneha, Mariam, Usman, Alisa, Moiz. I am also grateful to my RBS family comprising of Sitki, Emre, Laurens, Grace, Waj, Sevinch, Ming, Aparna, Karina and Emine, for the fun lunches, help and ideas. The long list above shows how blessed I am for having so many people to thank. I am truly grateful to Allah for giving me so much to be grateful for. Thank you Allah. I love you all dearly.

Nilofar Varzgani

Newark, NJ, October 2017

Contents

List of Figures

Introduction

We study the open problem of valuing Tax Loss Carryforwards (TLCFs) and Tax Loss Carrybacks (TLCBs), and propose a dynamic programming methodology, that permits both multiple tax brackets and time periods, which, we believe, is both flexible and practical. Given the pervasiveness, impact and size of TLCFs and TLCBs, it is somewhat surprising that the problem of valuing these still remains open. This thesis attempts to fill this gap in the literature[1].

By its very nature, valuation of TLCFs and TLCBs is a multi-period problem. Furthermore, taxes are usually paid at discrete intervals. In addition, there is also the added difficulty that TLCFs (TLCBs) have finite lives, and expire if unused in a said period of time. Laws limit how much of these tax benefits can be used in a given period. They also tend to regenerate if the firm incurs a loss. Given these complications, it is our view that the dynamic programming in a discrete time environment is the most natural framework to value TLCFs and TLCBs.

In the first and second chapters we propose a multi-period valuation model that is flexible enough to handle multiple tax brackets, which a key feature of progressive taxation.We apply our methodology to the specific case with 2 periods and 2 tax brackets. We further provide conditions under which it is always optimal to follow the first in first out (FIFO) rule for utilizing TLCFs. This was imposed as an assumption in earlier part of the analysis. Later, we generalize our model to accommodate both TLCFs and TLCBs. In the second chapter we study the effect of taxes and TLCFs on Merton's consumption investment model and analyse how the addition of another

[1] The work in this dissertation is normative, (in the spirit of standard dynamic asset pricing models).We provide an analytical framework to value TLCFs and TLCBS.

control variable in the form of the amount of TLCFs to use would cause the risk averse investor to become more inclined to invest in the risky asset increasing the demand for the risky asset.

In this paper we have shown how the dynamic programming approach can be used to solve a long outstanding problem of valuing TLCFs and TLCBs. Our approach is flexible enough to accommodate a variety of institutional restrictions and regulations. Prior work has suggested using the well-known option pricing methodology to address this problem. While there appears to be an option type feature to TLCFs and TLCB, there are marked differences and institutional constraints that would make it inappropriate to use the option pricing methodology to value these tax benefits. In contrast, the dynamic programming approach proposed by us is a more natural economic approach. In particular, our approach can be extended to models with risk averse utilities, and general asset pricing models. We also note that TLCBs and TLCFs may make it attractive to take on more risky assets due to the added tax savings, which, in turn will increase demand and prices for risky ventures.

In the third essay, we tackle the problem of strategic sourcing versus multiple supplier sourcing model using dynamic programming. In this essay we investigate a case study in which an electronics manufacturing firm outsources its assembly operations to contract manufacturers. Limited research has been done on the impact of supply chain risks on sourcing strategies. These risks include financial, political and environmental. We use first introduce the CM sourcing problem for a one period setting. This is identify the different aspects that need to be considered. The one period model does not involve any dynamic programming so it is close to what most OEMs follow now in order to chose suppliers. Hence the one period model serves as an existing base to compare our dynamic programming model to, identifying the benefits of formualting it using the later, hence our contribution to the literature. Each contract manufacturer is assumed to have a different level of improvement capability of inducing supply cost reduction, which will be beneficial for the maufacturing firm. Three types of CMs are considered based on their sizel (i) one which is a well-established, large contract manufacturer with many major accounts and hence the ability to offer lower overhead costs, (ii) one which is a medium sized player in the market with moderate to good capabilities, and (iii) one which is a new player in the market and relatively small compared to the above two types. We also present simulation results for different values of input parameters.

CHAPTER 1

Survey of Tax Environment

1.1 Definition of Tax Loss Carryforwards and Carrybacks

Net Operating Losses (NOL), defined as the amount by which the operating expenses exceed the revenues for the business, are of immense importance in the financial reporting of these businesses. These NOLs are not only important because they are a measure of the business's health but also because corporate tax codes allow for these NOLs to be used in some other tax reporting period to offset taxable income, which reduces the tax liability of the reporting business. Thus, they are a silver lining for businesses. These deductions can be used against previous or future tax returns. When they are used against future tax returns, they are termed as Tax Loss Carryforwards and when they are used against previous tax returns, they are termed as Tax Loss Carrybacks.These provisions are also called net operating loss (NOL) carrybacks and carryforwards. From here onwards, we refer to Tax Loss Carryforwards as TLCFs and to Tax Loss Carrybacks as TLCBs. These tax deductions are important because many businesses operate in industries that fluctuate greatly with the business cycle. They might have high profits one year, but end up making huge losses the next year. TLCFs and TCLBs help those businesses to smooth their fluctuating income, which results in the tax laws being more just with respect to time. The Internal Revenue

Service (IRS) also allows for inidividuals to report NOLs, just like businesses. The NOLs for inidividuals come from realized capital losses and can be used to offset realized capital gains.A capital gain or loss is realized when the asset is sold requiring a buy and a sell transaction. A realized capital gain generates a tax liability, and a capital loss can be used to offset your tax liability for gains. When an individual sells an asset such as a stock at a loss, for example, the tax law provides a TLCF to offset other capital gains and reduce the individual's tax liability, including capital gains realized in the future years.

1.2 Overview of Regulations Governing TLCFs and TLCBs

There are a number of restrictions on the use of TLCFs and TLCBs imposed by the tax code. In the United States, *US ASC 740-10-30-5* and *US IRC Title 26-Subtitle A-Section 172* governs these restrictions imposed. These restrictions control (i) the maximum amount of losses that can be used, (ii) the maximum number of years it can be carried forward or back, (iii)the acceptable categories of losses eligible for deductions, (iv) computations of deductions, and (v) special rules and limitations. These are discussed in detail below:

1.2.1 Determination of Deductions

- (i) the maximum amount of losses that can be used:

  This limit is given by the Section 1502 and 1503 of the Federal IRC, or to the extent allowed by the state if it is different from the one given by the federal code. For example, Illinois puts a limit of $100,000 on it's TLCFs, whereas

New Hampshire caps it at \$10 million. For carrybacks, Delaware limits it to only \$30,000 and Utah limits it to \$1 million. In 2015, New York eliminated it's cap of just \$10,000 on the use of TLCBs, after tax reforms were passed in 2014.

- (ii) the maximum number of years it can be carried forward or back:

  TLCFs and TLCBs have a limited life as mentioned earlier so they are subject to expiry. Different counrties impose different number of years on the permissible use of these losses against income. In the U.S., for instance, the federal code allows 20 years of NOL carryforwards and 2 years of NOL carryback. However, states vary widely on their net operating loss policies. Figure 1.1 and 1.2 below are two maps taken from the Tax Foundation's 2015 State Business Tax Climate Index, that show the number of NOL carryforward years and carryback years respectively, allowed by each state in the U.S. Some states such as Rhode Island only offer five years of carryforwards, where as majority of them conform with the federal standard of 20 years. As shown in Figure 1.2, several states do not allow NOL carrybacks at all. These rules are subject to change. For instance, Nebraska and New Mexico increased their NOL carryforward life from five years to twenty years in the year 2014, where as Illinois temporarily suspended it's NOL deductions entirely in 2011. So even these regulations are susceptible to frequent changes[1].

- (iii)the acceptable categories of losses eligible for deductions and (iv) computations of deductions:

  What can and can not be categorized as a deduction is also subject to strict laws (Section 1202). TLCFs (and TLCBs) broadly include net operating losses, capital losses, foreign tax and other general business credits. These rules limit what the individual or business can deduct when computing an NOL. In gen-

[1] The valuations model developed in Chapter 2 allows for these regulations to change.

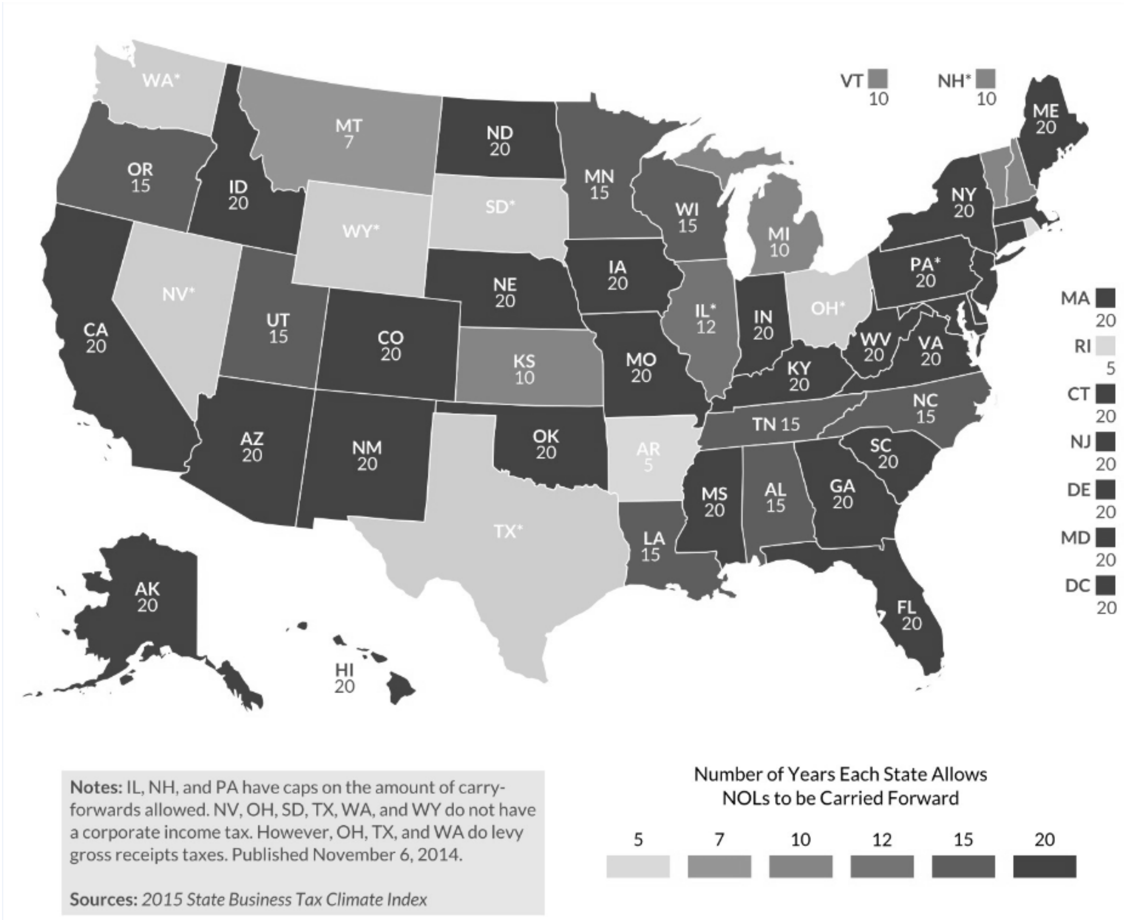

Figure 1.1: Corporate Net Operating Loss Carryforwards

eral, the some of the items not allowed when computing TLCF(or TLCBs) are:

- Any deduction for personal exemptions.
- Capital losses in excess of capital gains.
- Nonbusiness deductions in excess of nonbusiness income.
- The net operating loss deduction.
- The domestic production activities deduction.

- (v) special rules and limitations:

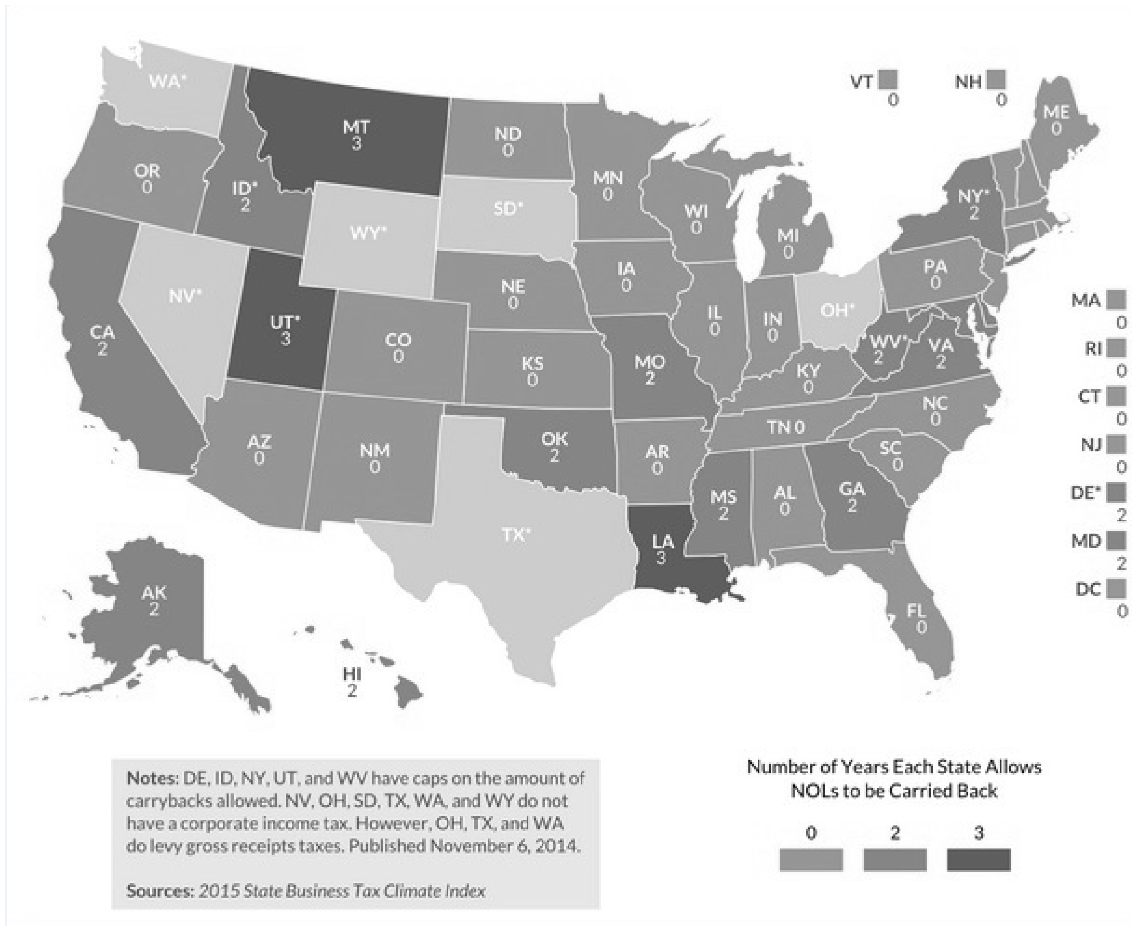

Figure 1.2: Corporate Net Operating Loss Carrybacks

These include the special rules for REITs[2] and specified liability losses.

1.2.2 The Section 382 Limitation

Since a net operating loss can be used to directly reduce the amount of taxable income, it can be considered a valuable asset. The Internal Revenue Service makes sure that if a business acquires an entity that has an NOL, the reason for doing so should not be the presence of the NOL. This restriction on the used of an acquired NOL is given

[2]Real Estate Investment Trusts: a company that owns or finances income-producing real estate. REITs provide investors of all types regular income streams, diversification and long-term capital appreciation.

by Section 382 and Section 269 of the IRC. Section 382 specifically states that:

> " If there is at least a 50% ownership change in a business that has an NOL, the acquirer can only use that portion of the NOL in each successive year that is based on the long-term tax-exempt bond rate multiplied by the stock of the acquired entity. "

Section 382 can be a significant problem if a business has large unused NOLs on its books. Despite this restriction, the presence of a large NOL can impact the price paid by an acquirer, since it effects the after tax cash flows that an acquirer will derive from the outcome of the deal.

1.3 Current Treatment of TLCFs (and TCLBs)

TLCFs (and TLCBs) generally come under the umbrella of deferred tax assets. In the United States they are subject to the financial reporting standard, Accounting Standards Codification (US ASC 740-10-55), which requires corporations to estimate and report the value of all deferred assets and liabilities on their balance sheets[3]. According to the guidelines of US ASC 740-10-30-5, the methodology used to arrive at the reported value of deferred taxes, any assumptions that are made, and other such details have to be reported. There is a requirement that deferred tax assets on the balance sheet of corporations must be offset and balanced by a valuation allowance account to reflect the management's estimated true value of the deferred tax asset[4]. This estimate has to be revisited and revised periodically. Any revisions

[3] Formerly known as SFAS109

[4] Specifically, if the management expects there is more than 50% chance they will not be able to realize some of deferred tax assets (for example, if they expect that future taxable income may not be large enough to fully avail of the deferred tax assets), the firm must report a valuation allowance as a contra account to deferred tax assets to account for this uncertainty as to how much of the deferred tax assets are like to be realized. The management has to periodically revise it estimation this allowance in the future, and the effects of change in allowance will be reflected in the income statement of the firm.

of this account will flow through the income statement and affect reported earnings[5]. Consequently, proper valuation reporting of TLCFs (and TLCBs) is a concern for regulatory bodies such as the Financial Accounting Standards Board (FASB) and the Securities and Exchange Commission (SEC) as well.

The basic rules for using TLCFs (and TLCBs) are:

1. Carry the amount back to the preceding allowable tax years and apply it against any taxable income, which can generate an immediate tax rebate. The business/individual has the option to waive this action and instead proceed directly to the next step. If they decide to do so, a statemen is attached to their tax return in the year in which the NOL was generated, documenting the waiver.

2. Carry the amount forward for the next "n " number of allowable years and apply it against any taxable income, which reduces the amount of taxable income in those years. After "n" years, any remaining NOL is cancelled (expired).

It makes financial sense to apply the NOL against the earliest periods possible, since the time value of money dictates that the tax savings in these periods is more valuable than for any tax savings in later periods.Obviously, the entity should carry an NOL forward if they paid no taxes in the prior years. The business or individual may also elect to carry the NOL forward if they expect their future income to greatly increase in coming years, potentially placing them in a higher tax bracket[6]. The higher the tax rate, the more an NOL will save in taxes. However, after this decision has been committed, it can not be reversed if the business doesn't do as well as expected[7].

[5]Literature that discusses the implication of this rule is covered in Section 1.4

[6]This will be considered as a decision variable in the model in Chapter 2 and Chapter 3 of this dissertation

[7]If NOLs are being generated in multiple years, the order in which they are used is also of importance. This will be discussed further in Chapter 2, where it is mathematically proved that the *First In First Out* sequence of usage is the most optimal sequence. This means that the earliest NOL should be completely drawn down before the next oldest NOL is accessed. This approach reduces the risk that an NOL will be terminated by the expiry rule noted earlier.

One prime example in recent times, of TLCFs being used in a similar manner to add value to the company is the case of General Motors (WSJ (2009)). When GM filed for bankruptcy, it chose a so-called "363 sale", which allows the company to take a fast track to the sale without the due process protections usually provided to creditors.The new GM will be allowed to claim a tax benefit from some $16 billion of net operating losses carried over from the old company, allowing it to avoid paying taxes on future profits, perhaps for years. According to The Wall Street Journal's article documenting this, nothing in the tax code permits the preservation of tax attributes like net operating losses in the context of an outright sale like GM's. Another such documentation of the use of TLCFs in popular media has been an article in Forbes that states how 70% of companies paid zero in corporate taxes (Mathur (2016)).The article explains that while "net income" generally refers to the net profit or loss after allowing for certain usual deductions (such as depreciation allowances, compensation payments and interest), "income subject to tax" allows companies with positive "net incomes" to claim an additional deduction as a result of prior-year operating losses, commonly refered to as a net operating loss deduction (NOLD). For 2012, the data show that approximately 20% of companies with positive "net incomes" (or profits) claimed a net operating loss deduction resulting in a zero tax liability.

1.4 Literature Review

The need for fair valuation of TLCFs (and TLCBs) arises frequently in the theory and practice of corporate finance, asset pricing, mergers and acquisitions, accounting and tax planning. Furthermore, it has also been shown that TLCFs and TLCBs may have important implications on corporate financing and investing decisions. (Auerbach and Poterba (1987)). In addition, there is also a regulatory concern regarding the fair valuation of TLCFs and TLCBs, as discussed in 1.2.2 and 1.3 .

In the absence of any valuation methodology presently for TLCFs and TLCBs, the value assigned to deferred tax assets, the allowances, and subsequent revisions, have been alleged to have become tools for management to manipulate earnings (for example Amir et al. (1997), Amir et al. (2001), Miller and Skinner (1998),Schrand and Wong (2003)). Consequently, proper valuation reporting of TLCFs (and TLCBs) is also of interest to the Financial Accounting Standards Board (FASB), and to the Securities and Exchange Commission (SEC) as well.

As noted earlier, these TLCFs and TLCBs are like the silver lining in the dark clouds of losses for a business. Indeed many empirical studies in the finance, tax, and accounting literature have shown that TLCFs and TLCBs are positively valued by investors (for example [Amir et al. (1997), Amir et al. (2001)], [Amir and Sougiannis (1999), [Ayers (1998))], [Chaney and Jeter (1994)], [Chang et al. (2009)], [Chen and Schoderbek (2000)], [Hasselback (1976)], [Hicks (1978)], [Miller and Skinner (1998)], [Wolk and Tearney (1973)], [Zeng (2003)])[8].

The literature so far has established the significance of TLCFs for reporting and accounting purposes but there still is a need for a model to value them. The demand for a theoretically sound and practical approach to value TLCFs (and TLCBs) is particularly pressing now given the recent financial crisis resulting in huge losses for firms. Even before the recent economic crisis, it has been pointed out in the literature that a large number of firms (almost three quarters of the firms in Compustat) have accumulated tax losses and TLCFs can range in value from 10 to 30 percent of total assets ([Bauman and Das (2004)]; [Graham (1996)]; [Miller and Skinner (1998)]). This number is much higher today. Figures 1.3 and 1.4 below, and Table A.1 (see Appendix), present evidence on the importance of firms with tax loss carryforwards in the years since 2002. Figure 1.3 shows the total population of firms which have

[8] Interestingly, after the recent financial crisis, there are anecdotes, where unprofitable firms have “proudly" advertised their losses to bait buyers.

reported tax loss carryforwards on their balance sheets in the North America Region only (US Companies), whereas Table A.1 (See Appendix), and Figure 1.4 report TLCFs by industry. It is quite clear from these figures and table that there has been a steep rise in reported TLCFs across a wide cross-section industries since 2002.

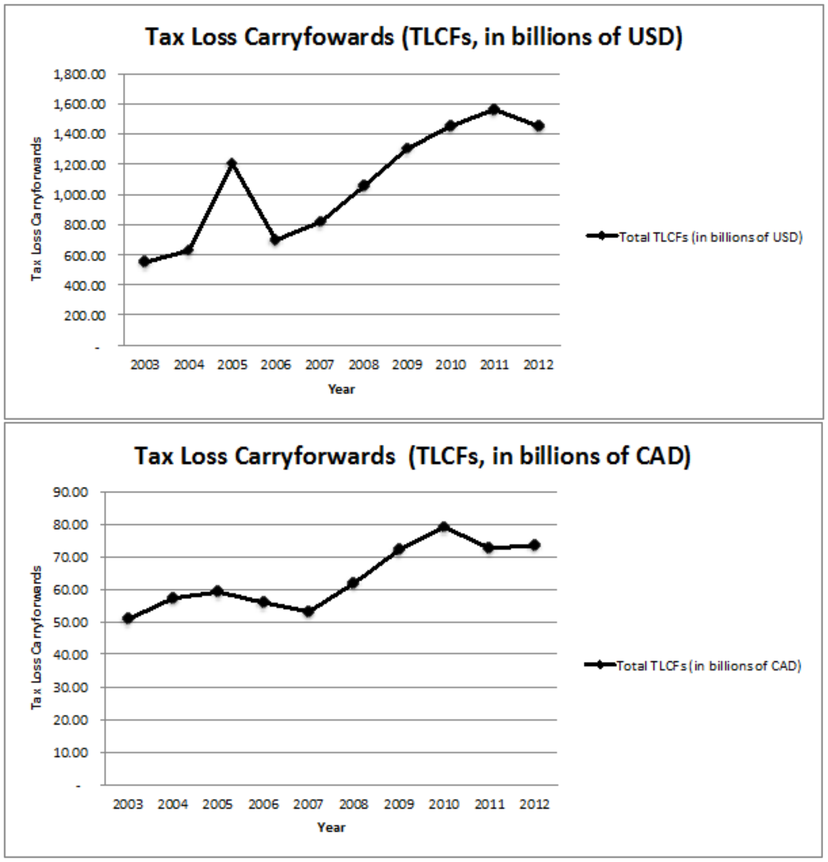

Figure 1.3: Tax Loss Carryforwards- North America Region (2003-2012)

The tax loss carryforwards are the ones reported on the balance sheet of the companies in USD in the respective years collected from COMPUSTAT Company Financial Statements Data Items for North America only.

Perhaps one reason for the lack of a unified valuation framework for TLCFs (and TLCBs) is the wide variety of corporate activities that generate these. Another reason, is the uncertainty about how much of the TLCFs (and TLCBs) can actually be used. Future earnings are uncertain, and, given the finite period over which the

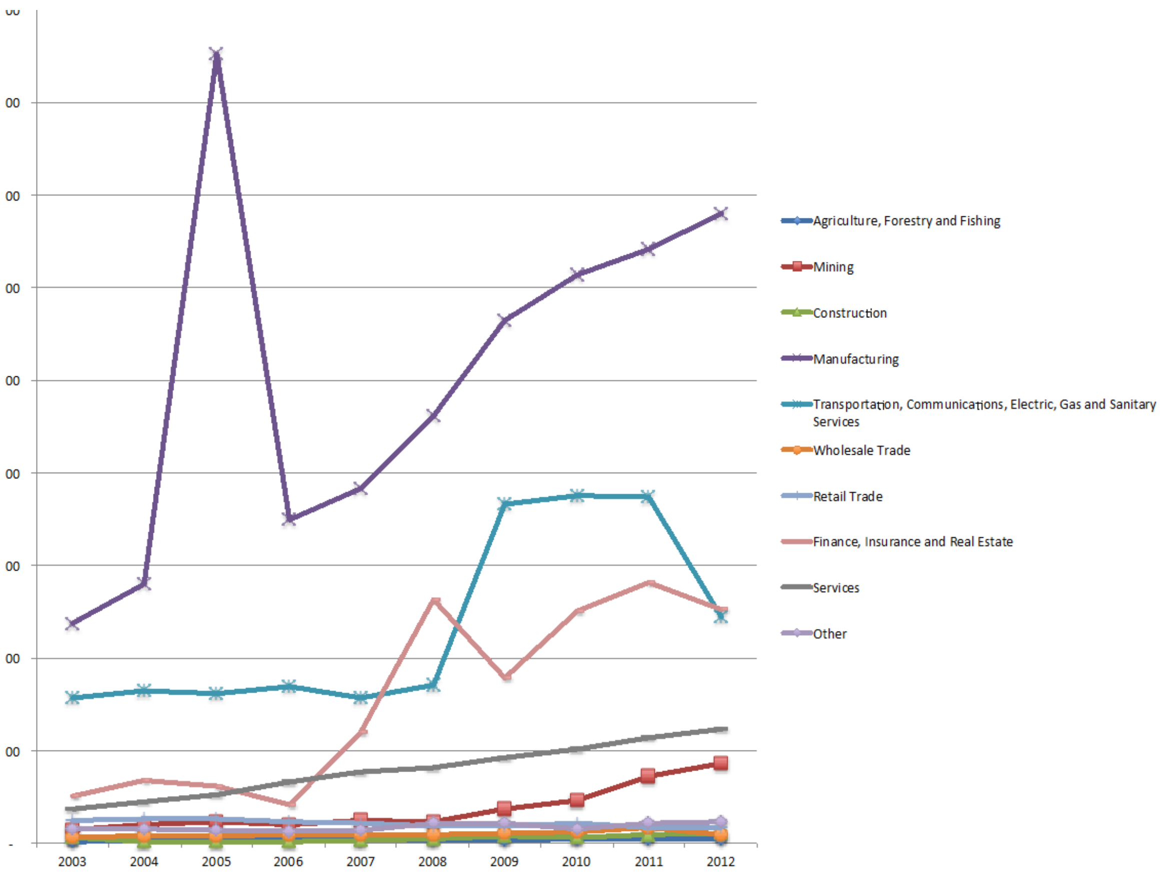

Figure 1.4: Tax Loss Carryforwards - North America Region (2003-2012) Industry Breakdown

Data Source: COMPUSTAT. The value of tax loss carryforwards are as reported (in USD) on the balance sheet for individual companies in North America Region for each year. The Standard Industry Classification (SIC) Code has been used here to group the companies in progressively broad industry classifications.

TLCFs (and TLCBs) can be used in the future, it is not clear how to estimate their tax benefits. This problem is further exacerbated by the fact that future losses give rise to new TLCFs (and TLCBs) with different expiry dates. Given the different expiry dates for the existing inventory of TLCFs (and TLCBs), and the regeneration of new ones going forward, the complications multiply.

Yet another important reason is the complexity of the tax code. Tax laws are formulated by lawmakers with vested interests representing varied economic and political interests, the result of which is a bewildering maze of rules and exceptions.As one common example, consider the case of mergers and acquisitions in the United States. The acquirer, after complying with the Internal Revenue Code (IRC) 269, also known as the valid business purpose doctrine, that voids acquisitions primarily aimed at avoiding taxes, may also have to conform to the provisions of IRC 382 that dictates how much of the TLCFs can actually be used annually in the future. To add to this, firms may also be subject to Section 1502 of the United States Treasury Department regulations (better known by its acronym, SRLY, or Separate Return Years Limitation Yearly), and other regulations of the department, such as Section 1503 (Dual Consolidation Losses). To model the complex tax rules, and their exceptions, often ends in futility.

The few authors that have attempted to obtain analytical solutions have largely ignored these institutional constraints, and simply ended up trying to fit the problem within existing paradigms, notably from the options literature (Sarkar (2012)]; [Streitferdt (2010)]). Specifically, Sarkar (2012) develops a real option valuation model where they compute the fair value of the firm's Tax Loss Carryforwards (TLCs) and shows that, for poorly performing firms with large tax loss carryforwards, (i) the realizable (or fair) value of the tax losses can be significantly smaller than the book value, and (ii) the tax losses can account for a significant fraction of the company's equity value. However, his work is limited by the fact that he does not consider the restrictions of the tax code. In addition, the real options methodology is not quite equipped to address the problem of valuing TLCFs.Streitferdt (2010) also uses the option methodology and is subject to similar limitations. While TLCFs do have features similar to options (they have a finite expiry time, and can only be used in states of nature where the firms have taxable incomes), the institutional structure of taxation does not permit the straightforward extensions of options theory and formulas to valuing TLCFs. [Waegenaere et al. (2003)] develops a model to examine the properties of the market-to-book ratio of deferred tax assets arising from TLCFs. However, the objective and structure of their paper is quite different from ours. Their work can be regarded as positive rather than normative as they seek to explain certain observed empirical characteristics of the market-to-book ratios rather than develop an analytical framework for valuing TLCFs and TLCBs[9].

[9] They assume that the valuation allowance used for calculating the value of TLCFs is determined using the median income without any room for managerial judgment or action. Our model, on the other hand, uses expected values and allows room for judgment in the form of the rule/policy used for taking action (we describe this in detail in Section 2). Also see Rhoades-Catanach (2003) for a discussion of their paper.

CHAPTER 2

A Dynamic Programming Model to Value TLCFs

2.1 Model Setup

A tax loss carryforward derives its value from the fact that it can be used to offset taxes against future taxable profits. We model a firm with initial positive taxable income of Y_0. Let $C_t = (c_t^0, c_t^1, c_t^2, \cdots, c_t^N)$ denote the vector of TLCFs that can be used at each time $t \geq 0$ respectively, where the superscripts represent the periods to expiry (useful life) for a given TLCF. For example, c_0^0, means this TLCF can be used at time $t = 0$, and will expire immediately if not used; and c_0^N, means that it can be used at time $t = 0$, but any unused portion can be carryforward for N periods ahead. The superscript N is a finite positive integer. In general, at any time, $t+1$, $\forall$ $t = 0, 1, 2, ., T-1$, the set of TLCFs available for use is pre-visible and known at time t (after taking actions described below), and is given by $C_{t+1} = (c_{t+1}^0, c_{t+1}^1, c_{t+1}^2, , c_{t+1}^N)$. We will assume at this stage that the TLCF that is expiring the earliest will be used first before using the one expiring next, and so on (the so called *first in first out* (FIFO) method)[1].

We use X_t to represent the after tax wealth in the time period t, for all $t = 0, 1, , T$. So the initial state at time, $t = 0$, is given by the triplet (X_0, Y_0, C_0); and Y_t to

[1] The FIFO assumption will be relaxed later in Chapter 2.XX

represent the return on wealth (or profits) on after-tax wealth X_t invested at time $t-1$. If Y_t is positive, it will be taxable. We assume that any point in time t, the firm re-invests all its wealth,X_t, back in the firm. We also assume a binary model for the return on invested wealth, that is, Y_t can be a profit (and taxable), w_t, or loss, u_t, with probability, p and $(1-p)$ respectively.

$$Y_t(X_t) = \begin{cases} w_t = r_t^p X_t & \text{with probability } p \\ u_t = r_t^l X_t & \text{with probability } (1-p) \end{cases}$$

where,$r_t^p > 0$, is the rate of return if profit state is reached and $r_t^l < 0$, is the rate of return if loss state is reached. We define an action at any time t, $t = 0, 1, ., T$, as the total TLCF elements from the set of TLCFs ,C_t, used to offset taxes. Let D_t , $\forall\ t = 0, 1, ., T$, be the set of all allowable actions vectors, A_t. This set of sets, D_t, allows for all sets of actions including FIFO and non-FIFO. Every action vector A_t comprises of elementary actions, a_{it}^n, $n = 0, 1, \cdots, N$ and $t = 0, 1, \cdots, T$, which has a one to one correspondence with the set of available TLCFs C_t. In addition, each elementary action is bounded below by zero and above by the corresponding available TLCF c_t^n. That is, it is permissible not to use any of the available TLCF, c_t^n, in which case $a_t^n = 0$, or use up everything, in which case $a_t^n = c_t^n$, or to use any part of it $(0 < a_t^n < c_t^n)$. We also allow for the possibility that there may be legal restrictions on how much of the available TLCFs can be used at a point in time.). Let L_t be this upper bound which is known at time, $t-1$. This L_t incorporates the cap on the maximum usuage of TLCFs described in Chapter 1.2.

$$D_t : A_t : 0 \leq a_t^n \leq c_t^n, \text{and } \bar{a}_{it} = \sum_{n=0}^{N} a_{it}^n \leq \min(w_t, L_t), \forall n \text{and } t.$$

Since TLCFs can only be used if there are taxable profits, the action depends on whether there is a profit or a loss. In the case of a loss, there is no action to take. Any action at time t regarding how much of the TLCFs to use will clearly depend on the realizations of Y_t, and the set of available TLCFs, and will be denoted by $A_t(Y_t, C_t)$. Consequently, any action depends on the taxable income, the available TLCFs, and the legal limitations, and will be represented by $A_t(Y_t, C_t, L_t)$.

Here on, to simplify the notation, let $\bar{c}_t = \sum_{n=0}^{N} c_t^n$, represent the sum of all available TLCFs for use at time t. Consistent with existing tax laws, any portion of TLCFs that remains unused at the end of its useful life will expire. The specifics of on how and how much of a TLCF can be utilized is governed by tax laws and regulations that we alluded to in the introduction. The firm is assumed to live until a terminal time T, at which time it is dissolved and a terminal dividend is paid. We note that the life of the TLCF with the longest life N can exceed the life of the firm T. To focus on the main objective of this section, that is the valuation of TLCFs, we assume that the firm does not pay interim dividends[2]. Additionally, $\bar{a}_{it}$ is also bound by $\bar{c}_t$ at the top but we did not include that in the set of constraints given above because the bounds imposed on each elemantry a_{it}^n take care of that condition so that $\bar{a}_{it}$ is never higher than $\bar{c}_t$.

For every time t, we allow for m piecewise linear progressive taxes with marginal taxes for each level of profits specified by the vector $T_t = (\tau_t^1, \tau_t^2, , \tau_t^m)$. The corresponding levels of profits separating any two marginal tax brackets is given by a vector $B_t = (b_t^1, b_t^2, , b_t^m)$, where $b_t^m > b_t^{(m-1)} > b_t^{(m-2)} > \cdots > b_t^1$, and without loss of generality, we set, $b_t^1 = 0$. The highest taxation is for profits above b_t^m, and these profits will be taxed at the highest marginal taxes of τ_t^m, and the profits between b_t^m and $b_t^{(m-1)}$ will be taxed at the next highest marginal rate of $\tau_t^{(m-1)}$, and so on.

[2]Note that intermediate dividends can be incorporated within our framework. So far we can just assume that there are no such dividends to avoid complicating the model.

2.2 Recursive Equations

The application of TLCFs to the multiple tax brackets has the tendency to complicate the notation quite a bit. To deal with this problem, we came up with recursive equations that satisfy the sequencial usage of the TLCFs according to their remaining useful lives and order of the tax brackets (highest to lowest). In order to show these recursive equations, we have to define some additional notation. The total taxable (positive) profit at any time w_t is divided into each piecewise linear taxable bracket, and it follows that $w_t = \sum_{(k=1)}^{m} w_t^k$, where w_t^k is the amount of profit taxed at the k^{th} tax bracket. The taxable profit at the highest tax bracket is given by $w_t^m = (w_t - b_t^m)^+$, while for all other tax brackets, that is for $k = 1, 2, \cdots, m-1$, it is given by $w_t^k = (w_t - (\sum_{(j=k+1)}^{m} w_t^j) - b_t^k)^+$. If the return on invested wealth is a profit, that is, $Y_t = w_t$, the action to decide on is a vector of elements corresponding to each element in the set C_t. The sum of all the elements in the action vector A_t at time t is applied to the highest tax bracket first (if possible),then to the next tax bracket, and so on, until the lowest tax bracket until the TLCFs or the profits are all exhausted (the so called residual method). The action sum $\bar{a}_t$ is then subtracted from the available set of TLCFs (using the FIFO rule). This is formally shown next.

Define,

$f_t^k(Y_t, A_t)$ as the taxable profits at time t after taking action A_t (at tax bracket k) , when the system is in state Y_t and;

$g_t^k(Y_t, A_t)$ as the remaining TLCFs at time t that can be used after taking action A_t (at tax bracket k),when the system is in state Y_t.

Consequently,

For $k = m : f_t^m(Y_t, A_t) = (w_t^m - \bar{a}_t)^+$; and $g_t^m(Y_t, A_t) = (\bar{a}_t - w_t^m)^+$

and

For $k = 1, 2, \cdots, m-1 : f_t^k(Y_t, A_t) = (w_t^k - g_t^{k+1})^+$; and $g_t^k(Y_t, A_t) = (g_t^{k+1} - w_t^k)^+$

2.3 Valuations of TLCFs

We value at initial time $t = 0$, the TLCFs that are available at the initial and future times (new ones that are generated by losses) until the terminal time T; that is, we value the discounted tax savings arising from the TLCFs, $C_t = (c_t^0, c_t^1, \cdots, c_t^N)$, $t = 0, 1, \cdots, T$, along the optimal path. The optimal path is a consequence of a firm acting rationally and optimally choosing an action vector A_t at each time t regarding how and how much of the TLCFs should be used to offset current taxes, so as to maximize the profits at terminal time, T. We assume $\alpha \in [0, 1)$ to be a given discount factor for the firm.

The idea is that the firm, given its history, H_t, up until time t, where history is to be defined as a record of the wealth, profits or losses, and actions taken until date t, that is to say, that history is given by the set $H_t = \{X_0, Y_0, A_0, X_1, Y_1, A_1, X_2, Y_2, A_2, \cdots X_t, Y_t, A_t\}$, will act rationally going forward. The firm will consider every possibility from current time t to terminal time T in choosing any action (to use TLCFs to offset taxes) from its set of available actions at current time t. We propose a backward recursive method all the way to time $t = 0$. What this means is that principles of Markovian dynamic programming can be applied, and we can value any set of TLCFs at initial time given that the firm acts optimally in the future. This would be the rational value of the TLCFs at time $t = 0$. Next we present this idea more formally.

Let, $\gamma_t(Y_t, A_t)$ be the taxes payable at time t after action A_t has been applied. For $m$tax brackets, the total taxes paid at time t will be given by:

$$\gamma_t(Y_t, A_t) = \sum_{k=1}^{m} f_t^k(Y_t, A_t)\tau_t^k \tag{2.1}$$

Let $s_t(Y_t, A_t)$, be the savings from TLCFs at time t, when the profit/loss state of the system is given by the value of Y_t, that is $Y_t = w_t$ or $Y_t = u_t$. The benchmark for computing the savings from TLCFs when there are no such TLCF deductions permitted, that is,

$$s_t(Y_t, A_t) = \gamma_t(Y_t, 0) - \gamma_t(Y_t, A_t)$$

The after tax wealth for any time period $t+1$ is given by X_{t+1}, which is calculated as:

$$X_{t+1}(X_t, Y_t, A_t) = X_t + Y_t - \gamma_t(Y_t, A_t), \forall t$$

In accounting for the set of available TLCFs, have to be cognizant of the fact that a new TLCF is created at any time t that the firm makes a loss. The life of this newly created TLCF can vary between 0 period ahead to N periods ahead, and has to be added appropriately to the set of TLCFs $C(t+1)(C_t, A_t) = (c_{t+1}^0, c_{t+1}^1, c_{t+1}^2, \cdots, c_{t+1}^N)$ available at the next time, $t+1$. That is, for all $t = 0, 1, \cdots, T-1$, if $Y_{t+1} = w_t$,then $C_{t+1}(C_t, A_t)$ such that $c_{t+1}^n = c_t^{n+1} - a_t^{n+1}$, for $0 \leq n \leq N-1$ and $c_{t+1}^N = 0$. On the other hand for a loss, that is if $Y_{t+1} = u_t$,then $C_{t+1}(C_t, A_t)$ such that $c_{t+1}^n = c_t^n$, for $0 \leq n \leq N-1$.

A loss at any time t will generate a new TLCF with a life ranging from 0 to N. Consequently this new loss will be added appropriately to the set of as a TLCF available for use in the next period. Without loss of generality, we assume that if there is a new loss of u_t, it will be added to a new TLCF in the the set of TLCFs

C_{t+1} that expires N periods from time $t+1$. More formally, we assume that:

$$\text{if } Y_{t+1} = u_t : c_{t+1}^N = u_t$$

Proposition 2.1. *Let at time $t = 0$, a firm be endowed with initial taxable income, Y_0, and, a set of TLCFs that can be used at time $t = 0$, $C_0 = (c_0^0, c_0^1, \cdots, c_0^N)$. Let A_t, $t = 0, 1, \cdots, T$, be the action vector, taken with respect to the usage of TLCFs given the realizations of Y_t at time t. Consequently,*

$$
\begin{aligned}
& a_t^n \in A_t; 0 \leq a_t^n \leq c_t^n, \\
& \sum_{n=0}^{N} a_t^n \leq \min(w_t, L_t), \\
& \textit{for } n = 0, 1, \cdots, N \textit{ and } t = 0, 1, \cdots, T, \\
& A_t \in D_t, \textit{ the set of all allowed action vectors,} \\
L_t &= \textit{Legal upper limit on the amount of TLCF that can be used at time } t
\end{aligned}
$$

Let R be a prescription for taking actions at each point in time. The dynamics of the after tax wealth X_{t+1} reinvested in the firm is given by:

$$X_{t+1} = \begin{cases} X_t + w_t - \gamma_t(Y_t, A_t), & \textit{if } Y_t = w_t \\ X_t - u_t, & \textit{if } Y_t = u_t \end{cases}$$

The probability that $Y_t = w_t$ is given by p, and $(1-p)$ is the probability that $Y_t = u_t$. Define $s_t(Y_t, A_t)$ as savings at time t when A_t is taken, that is,

$$s_t(Y_t, A_t) = \gamma_t(Y_t, 0) - \gamma_t(Y_t, A_t)$$

where $\gamma_t(Y_t, A_t)$ is defined in Eq. 2.1.

Let us denote by $V_t(X_t, Y_t, C_t; R, H_{t-1})$ *,*$0 \leq t \leq T$*, the conditional expected total savings of a process from time* $t = t$ *to* $t = T$ *given the history* H_{t-1}, X_t, Y_t, C_t *and policy* R*. Assuming a discount rate of* $\alpha \in [0, 1)$ *for the firm, and let* R^* *be the optimal prescription, then the maximum savings is given by the following:*

$$V_t^*(X_t, Y_t, C_t; R, H_{t-1}) = \max_{A_t \in D_t} \{W_t(X_t, Y_t, C_t; A_t)\} \text{ for } 0 \leq t \leq T \tag{2.2}$$

with

$$W_t(X_t, Y_t, C_t; A_t) = \begin{cases} s_t(Y_t, A_t) + \alpha \mathbb{E}_t^R[V_(t+1(X_{t+1}, Y_{t+1}, C_{t+1}(C_t, A_t); R, H_t)] \text{ if } Y_t = w_t \\ \alpha E_t^R[V_(t+1(X_{t+1}, Y_{t+1}, C_{t+1}(C_t, A_t); R, H_t)] \text{ if } Y_t = u_t \end{cases} \tag{2.3}$$

With terminal conditions $t = T$ *given by:*

$$W_t(X_t, Y_t, C_t; A_t) = \begin{cases} s_T(Y_T, A_T) \text{ if } Y_T = w_T \\ 0 \text{ if } Y_T = u_T \end{cases} \tag{2.4}$$

$$V_T(X_T, Y_T, C_T; R, H_{T-1}) = \begin{cases} \max_{A_T \in D_T} \{W_T(X_T, Y_T, C_T; A_T)\} \text{ if } Y_T = w_T \\ 0 \text{ if } Y_T = u_T \end{cases} \tag{2.5}$$

Proof. : The proof follows from the Principle of Optimality in the dynamic programming literature (see for example Derman and Derman (1970)). For our case, given that we are using a binomial tree model and a finite set of actions, a solution will always exist. □

2.4 Proof of Necessity of FIFO Condition

So far we have assumed the FIFO rule for actions while using TLCFs to offset taxes payable. In this section we identify conditions under which the FIFO rule is automatically the optimal action to take. Under these conditions, it would be unnecessary to impose the FIFO assumption. First we state and prove a Lemma.

Lemma 2.1. *Given taxable profits at any time $t = 0, 1 \cdots, T$, it is always optimal to first use the TLCF expiring immediately before using the other TLCFs.*

Proof. The proof is straightforward. If the first element in the TLCF set, that is, the one that is expiring in the current period, is not utilized fully before using the next available TLCFs that can be used in the next period and beyond, it will expire unused and cannot be carried forward to the following time period. Consequently, the related tax savings will be lost forever, and this cannot be an optimal action. Therefore, it follows that a rational decision maker will always find it optimal to use the TLCF expiring immediately before using any other available TLCF.

Next we turn to identifying the conditions under which it will be optimal to follow the FIFO rule in selecting from a set of available TLCFs. This means that a TLCF that is expiring earlier will always be used fully before going to the one that expires one period later or beyond. In other words, the elements of a set of available TLCFs are listed in an ascending order of their time to expiry, and it is always optimal to use each TLCF fully before using going to the next one on this ordered list. This is the FIFO rule.

We identify 2 possible costs of violating the FIFO rule and 1 benefit. The first cost will be an Explicit Cost, and the second one will be an Implicit Cost. The benefit will be an Implicit Benefit for reasons discussed below.

Explicit costs are defined as losses sustained when a FIFO rule is violated, and the TLCF from the available set of TLCFs that expires later is used before the one that expires earlier. So moving forward in time, there is the possibility that at the time the unused TLCF with the shorter expiry period is about to expire immediately (reached its expiry date), there may be insufficient taxable income, or even a loss, that will result in this TLCF being lost wholly or partly without proving any tax benefit. This is the Explicit Cost of violating the FIFO rule.

Implicit cost associated with a violation of the FIFO rule refers to the cost associated with a TLCF that had a longer expiry period was used earlier. So at the end of the original life of this TLCF, there is a possibility that there is a taxable profit but no other TLCF to use. So if this TLCF had not been used earlier, it would have provided tax benefits at this time. This is Implicit Cost of violating the FIFO rule.

However there is a mitigating benefit associated with violating the FIFO rule. It is possible that at the end of the useful life of the TLCF that was used earlier either partly or wholly by violating the FIFO rule, there is a loss instead of a profit. Therefore, any remaining portion of this TLCF could not have been used at this time, and would have just been wasted. It was just a luck of the draw that it was used earlier by violating the FIFO rule. We refer to this as an Implicit Benefit.

Every element of the set of TLCFs available at any time will have associated Explicit Costs, Implicit Costs, and Implicit Benefits. The FIFO rule will always be optimal if the total Expected Implicit Benefits are less the total expected Explicit and Implicit Costs. This stated in our next Proposition.

$\square$

Proposition 2.2. *Let the Explicit and Implicit Costs of violating the FIFO rule and the Implicit Benefits be defined as discussed above. Let Total Costs of violating the FIFO rule at any time t be the sum of the Explicit and Implicit Costs. The FIFO* *rule, that is,* $a_t^n > 0, \iff a_t^k = c_t^k, \forall k < n, n = 1, 2, 3, \cdots N, k = 0, \cdots, n-1$ *for using TLCFs will always be optimal if the Expected Total Costs are higher than the Expected Implicit Benefits.*

Proof. The proof of the above Proposition follows from Lemma ?? that states that the TLCF expiring immediately will always be used first, and the discussion above. □

Next we turn to showing that it is always optimal to follow the FIFO policy in using TLCFs provided the tax rates are assumed to be pre-visible. Before we prove this we need the following Lemma.

Lemma 2.2. *Let there are two permissible action vectors, at any point in time t, where t lies between 0 and T, with A_t violating the FIFO policy at time t and A_t consistent with the FIFO policy. More formally, let*

$$A_t = (a_t^0, a_t^1, \cdots, a_t^N) \qquad (2.6)$$

$$A_t = (a_t^0, a_t^1, , a_t^N) \qquad (2.7)$$

$$\sum_{n=0}^{N} \hat{a}_t^n = \sum_{n=0}^{N} a_t^n and a_t^{k_1} < a_t^{k_1}, a_t^{k_2} > a_t^{k_2}, where k_1 < k_2 \qquad (2.8)$$

Let $\hat{V}_t(X_t, Y_t, C_t; R, H_(t-1))$ represent the value function associated with action set A_t and $V_t(X_t, Y_t, C_t; R, H_(t-1))$ represent the value function with A_t. Then

$$\hat{V}_t(X_t, Y_t, C_t; R, H_(t-1)) V_t(X_t, Y_t, C_t; R, H_(t-1))$$

Proof. Without loss of generality assume that $t = T-1$, at which time $\hat{A}_{T-1}$ violates the FIFO rule and A_{T-1} does not. The sum of the action at time $T-1$ is the same for both the action vectors because they have both followed the same policy until then. At the next time period, $t = T$, the set of TLCF for associated with both action vectors would be given by $C_T(C_{T-1}, \hat{A}_{T-1})$ and $C_T(C_{T-1}, A_{T-1})$ respectively. If $k_1 = 0$ and

$k_2 = 1$, the only differencebetween $\hat{A}_{T-1}$ and A_{2T-1} is that $\hat{a}^0_{T-1} < a^1_{T-1}, \hat{a}^1_{T-1} > a^0_{T-1}$. The set $C_T(C_(T-1, A_{T-1})$ will contain TLCFs that have longer useful lives than $C_T(C_{T-1}, \hat{A}_{T-1})$ because of the possibility that the remaining amount of a^0_{T-1} that would have been used under the FIFO policy, may expire unused. Therefore, $C_T(C_{T-1}, \hat{A}_{T-1})$ to be lesser valuable (or less preferred) from a tax savings perspective than $C_T(C_{T-1}, A_{T-1})$, that is,

$$C_T(C_{T-1}, \hat{A}_{T-1}) \preceq C_T(C_{T-1}, A_{T-1})$$

This implies that

$$\sum_{n=0}^{N} \hat{a}^n_T \leq \sum_{n=0}^{N} a^n_T \tag{2.9}$$

Given that there some positive probability that some TLCFS may be lost unused if the action violating FIFO is followed, the expected savings associated with the action vector $\hat{A}_{T-1}$ will be lower than the savings associated with the action vector, A_{T-1}. It then follows that the value function with FIFO consistent actions will always be superior to value function that violates the FIFO rule,

$$\mathbb{E}_{T-1}\{V_T(X_T, Y_T, C_T(C_{T-1}, \hat{A}_{T-1}); R, H_{T-1})\} \leq \mathbb{E}_{T-1}\{V_T(X_T, Y_T, C_T(C_{T-1}, A_{T-1}); R, H_{T-1})\} \tag{2.10}$$

The above inequality holds because of the relationship in Eq. (2.9).

□

Theorem 2.1. *Assume tax rates are constant from time 0 to terminal time T. Then the policy consistent with the FIFO rule R_2 and associated with the action vector $A_(T-1)$ at time $t = T-1$, dominates any other policy R_1 that violates the FIFO rule and is associated with the action vector $\hat{A}_(T-1)$.*

Proof. At the time $t = T-1$, the value function for both action vectors is given by:

For $\hat{A}_{T-1} \in D_{T-1}$:

$$V^*_{T-1}(X_{T-1}, Y_{T-1}, C_{T-1}(C_{T-2}, A_{T-2})) =$$

$$\max_{\hat{A}_{T-1} \in D_{T-1}} W_{T-1}(X_{T-1}, Y_{T-1}, C_{T-1}(C_{T-2}, A_{T-2}); \hat{A}_{T-1})$$

$$W_{T-1}(X_{T-1}, Y_{T-1}, C_{T-1}(C_{T-2}, A_{T-2}); \hat{A}_{T-1}) =$$

$$s_{T-1}(Y_{T-1}, \hat{A}_{T-1}) + \alpha \mathbb{E}_{T-1}[V^*_T(X_T, Y_T, C_T(C_T, \hat{A}_T); R_1, H_{T-1})] \tag{2.11}$$

For $A_{T-1} \in D_{T-1}$:

$$V^*_{T-1}(X_{T-1}, Y_{T-1}, C_{T-1}(C_{T-2}, A_{T-2})) =$$

$$max_{A_{T-1} \in D_{T-1}} W_{T-1}(X_{T-1}, Y_{T-1}, C_{T-1}(C_{T-2}, A_{T-2}); A_{T-1})$$

$$W_{T-1}(X_{T-1}, Y_{T-1}, C_{T-1}(C_{T-2}, A_{T-2}); A_{T-1}) =$$

$$s_{T-1}(Y_{T-1}, A_{T-1}) + \alpha \mathbb{E}_{T-1}[V^*_T(X_T, Y_T, C_T(C_T, A_T); R_2, H_{T-1})] \tag{2.12}$$

Since the sum of the elements for each action vector is the same at $T-1$,

$$s_{T-1}(Y_{T-1}, \hat{A}_{T-1}) = s_{T-1}(Y_{T-1}, A_{T-1})$$

Now, from Lemma 2.2,

$$\mathbb{E}_{T-1}\{V_T(X_T, Y_T, C_T(C_{T-1}, \hat{A}_{T-1}); R, H_{T-1})\} \leq$$

$$\mathbb{E}_{T-1}\{V_T(X_T, Y_T, C_T(C_{T-1}, A_{T-1}); R, H_{T-1})\}$$

Therefore from Eq. 2.11 and Eq. 2.12 we have:

$$V^*_{T-1}(X_{T-1}, Y_{T-1}, C_{T-1}(C_{T-2}, A_{T-2}); R_1) \leq$$
$$V^*_{T-1}(X_{T-1}, Y_{T-1}, C_{T-1}(C_{T-2}, A_{T-2}); R_2)$$

□

2.5 Valuing Tax Loss Carryforwards with Tax Loss Carrybacks

In the United States, and in many other countries, it is quite common to allow firms to permit carry back current losses from operations and investments and be applied to earlier periods to reclaim taxes paid in those prior periods (though without interest). We refer to these as tax loss carrybacks (TLCBs). As is to be expected, the number of years for carryback and the amount of losses that can applied is limited by regulations. Furthermore, there is FIFO law that applies, which stipulates that the current losses must be applied against the oldest previous year allowed, before moving on to the next oldest, and all the way to the last year of after tax profit before the current year. Any current loss unused after this procedure turns into a TLCF.

Clearly this involves some computational complexity, but can still be accommodated within the model developed above. However, there is an important element here that makes the computations simpler. Current losses when carried back, save on taxes that had been paid in earlier periods. These taxes can be reclaimed at current time to increase current wealth, but they will not affect the past investments made before time current time t. However this increased wealth at current time will affect all future wealth going forward. The decision to carry back the losses and recover past taxes paid is solely based on whether added wealth right now is better than storing these losses and using them later as TLCFs. It is not difficult to see that unless taxes are expected to increase to very high levels in the future, it would be best to recover back taxes immediately.

To put this more formally, recall that when the outcome of an investment at time t is a loss, it is denoted as u_t. Let j be the number of periods it is permitted to carryback these losses[3]. Recall that L_t was used to define the legal upper limit on the amount of TLCFs that can be used at time t. Let us denote L_t^b as the legal upper limit the amount of TLCBs that can be used at time t. Let us define $a_t^b \in D_t^b$ as the action in dollars for the usage of the loss as a carryback. The following condition applies to a_t^b :

$$0 \leq a_t^b \leq \min\left(L_t^b, u_t\right)$$

Also recall that $f_t^k, \forall k = 1, 2, \cdots m$, is the taxable income at marginal tax rate $\tau_t^k, \forall k = 1, 2, \cdots m$ respectively. If it is decided that current losses will first be applied to different tax brackets at time $t-j$ first, then whatever is left is taken to reclaim taxes paid at time $t-j+1$, and so on until time $t-1$. Similar to the usage of TLCFs, we allow for laws governing how much of the losses can be carried back in one specific year.

To be more precise, the following would be the reclaimed taxes from time $t-j$:

$$(a_t^b \wedge f_{t-j}^m)\tau_{t-j}^m + ((a_t^b - f_{t-j}^m)^+ \wedge f_{t-j}^{m-1})\tau_{t-j}^{m-1} + ((a_t^b - f_{t-j}^m - f_{t-j}^{m-1})^+ \wedge f_{t-j}^{m-2})\tau_{t-j}^{m-2} + \cdots$$

[3]Net Operating Losses in the U.S. can be carried back 0-3 years, depending on the state. There are special categories of NOLs that can qualify for 3, 5 or 10 years of carry back time. In addition to that, corporations also have the option of waiving this carryback period and saving the losses for future use instead. If they make this choice, they can only use NOLs in the carryforward period. (U.S. Code 12, Subtitle A, Chapter 1, Subchapter P, Part II, Section 1212), this has been discussed in detail in Chapter 1

Any losses leftover after applying to the period $t-j$ will be applied to the next oldest period $t-j+1$ to obtain another sequence as in the above equation and so on till period $t-1$. Any losses unapplied after $t-1$ will be the new TLCF c_t^N in the vector C_t. So,

$$c_t^N = u_t - a_t^b$$

Let us denote $h_{t-j}(u_t, a_t^b)$ as the reclaimed taxes from the loss of u_t for time $t-j$. So;

$$h_q(u_t, a_t^b) = \sum_{z=1}^{m}((a_t^b - \sum_{i=z+1}^{m} f_q^i)^+ \wedge f_q^z)\tau_q^z$$

$$\forall q = t-j, t-j+1, \cdots, t-1$$

Hence, at time t (when the loss u_t was made), savings will be given by:

$$s_t(Y_t, A_t) = h_q(u_t, a_t^b) \forall q = t-j, t-j+1, \cdots, t-1$$

where, $Y_t = u_t$ and the next period wealth will be calculated as:

$$X_t = X_{t-1} - u_t + s_t(u_t, A_t)$$

The analysis proceeds from then on as in the section of TLCFs (Chapter 2.3).

2.6 Applications and Examples

To illustrate the application of the results we provide explicit representation and numerical solutions to some examples constructed to clarify the use of the algorithmic approach described in the above sections. We first show an example with the special case of two tax brackets to demonstrate Proposition 2.1 and how the model is applied. Next, we assume some numbers for the constants and the endowments to give approximate solutions using a simulations program constructed in *Matlab*[4]. There are examples of both FIFO and non-FIFO seqeunces of usage and an example including tax loss carrybacks to apply the formulae in Section 2.5.

2.6.1 Example: A Special Case with Two tax brackets and Two Time Periods

We first consider a special case with only two marginal tax brackets, high and low. The set of marginal tax rates is given by $_t = (\tau_t^l, \tau_t^h)$, where the superscript h refers to the high tax bracket and l refers to the low tax bracket. Similarly, $B_t = (b_t^1, b_t^2)$ where $b_t^1 = 0$. Here on for this example we use b_t instead of b_t^2 for the rest of this section to simply the notation. Then the savings for the two tax brackets is given by:

$$s_t(Y_t, A_t) = [(w_t - b_t)^+ \tau_t^h + \min[(b_t, w_t)\tau_t^l] - \{[(w_t - b_t)^+ \bar{a}_t]^+ \tau_t^h + [\min(b_t, w_t) - g_t^h]^+ \tau_t^l\}$$

To get rid of the minimum of b_t and w_t, we consider two separate cases of the example to simplefy it:

Case 1: The case where $w_t > b_t$

In this case, the savings will be given by:

$$s_t(Y_t, A_t) = [(w_t - b_t)\tau_t^h + b_t \tau_t^l] - \{[(w_t - b_t) - \bar{a}_t]^+ \tau_t^h + [b_t - (\bar{a}_t - (w_t - b_t))^+]^+ \tau_t^l\}$$

[4] The complete Matlab code is given in the Appendix

$$s_t(Y_t, A_t) = (w_t - b_t)\tau_t^h + b_t\tau_t^l - \begin{cases} ((w_t - b_t - \bar{a}_t)\tau_t^h + b_t\tau_t^l & \text{if } (w_t - b_t) > \bar{a}_t \\ 0 + (b_t - (\bar{a}_t - (w_t - b_t)))\tau_t^l & \text{if } (w_t - b_t) \leq \bar{a}_t \end{cases}$$

$$s_t(Y_t, A_t) = (w_t - b_t)\tau_t^h + b_t\tau_t^l - \begin{cases} (w_t\tau_t^h + (\tau_t^h - \tau_t^l)b_t - \bar{a}_t\tau_t^h & \text{if } (w_t - b_t) > \bar{a}_t \\ -\bar{a}_t\tau_t^l + w_t\tau_t^l & \text{if } (w_t - b_t) \leq \bar{a}_t \end{cases}$$

So $s_t(Y_t, A_t)$ can be simply written in terms of $\bar{a}_t$:

$$s_t(Y_t, A_t) = \begin{cases} \bar{a}_t\tau_t^h & \text{if } (w_t - b_t) > \bar{a}_t \\ (\tau_t^h - \tau_t^l)(w_t - b_t) + \bar{a}_t\tau_t^l & \text{if } (w_t - b_t) \leq \bar{a}_t \end{cases} \tag{2.13}$$

Case 2: The case where $w_t \leq b_t$

In this case, the savings will be given by:

$$s_t(Y_t, A_t) = [w_t\tau_t^l] - [(w_t - \bar{a}_t)\tau_t^l]$$

This is simplified to:

$$s_t(Y_t, A_t) = \bar{a}_{t t}^{\ l} \tag{2.14}$$

From Proposition 2.1, for the final time period $t = T$, the value function is given by:

$$V_T^*(.) = max_R V_T(R, .) = max_{A_T \in D_T}\{s_t(Y_t, A_t)\}$$

Let A_T^* be the action vector that gives the maximum in the above equation. Then:

$$V_T^*(.) = s_T(Y_T, A_T^*)$$

Depending on the state realized at terminal time T, the value of $s_T(Y_T, A_T^*)$ will be given by one of the equations given above (Eq.2.13 or Eq. 2.14) for $t = T$. Working recursively backwards in time to $t = T - 1$:

$$\begin{aligned} V^*_{T-1}(.) &= max_{A_{T-1} \in D_{T-1}} \{s_{T-1}(Y_{T-1}, A^*_{T-1}) + \alpha \mathbb{E}_{T-1}[V^*_T(X_T, Y_T, C_T; R, H_{T-1})]\} \\ &= max_{A_{T-1} \in D_{T-1}} \{s_{T-1}(Y_{T-1}, A^*_{T-1}) + \alpha p(s_T(Y_T, A^*_T))\} \end{aligned}$$

Repeating this process till we reach the initial time, $t = 0$, we get the value of the set of TLCF given by C_0 (assuming inputs for the probability distribution ofY_t, discount rate and tax laws) as:

$$V^*_0(.) = max_{A_0 \in D_0} \{s_0(Y_0, A_0) + \sum_{t=1}^{T} \alpha^t p^t (V^*_t(.))\}$$

where at each time $t = 0, 1, \cdots T$, the savings are given by either Eq. 2.13 or Eq. 2.14, depending on the state achieved for that time. Figure 3 gives a pictorial representation of the procedure for this example.

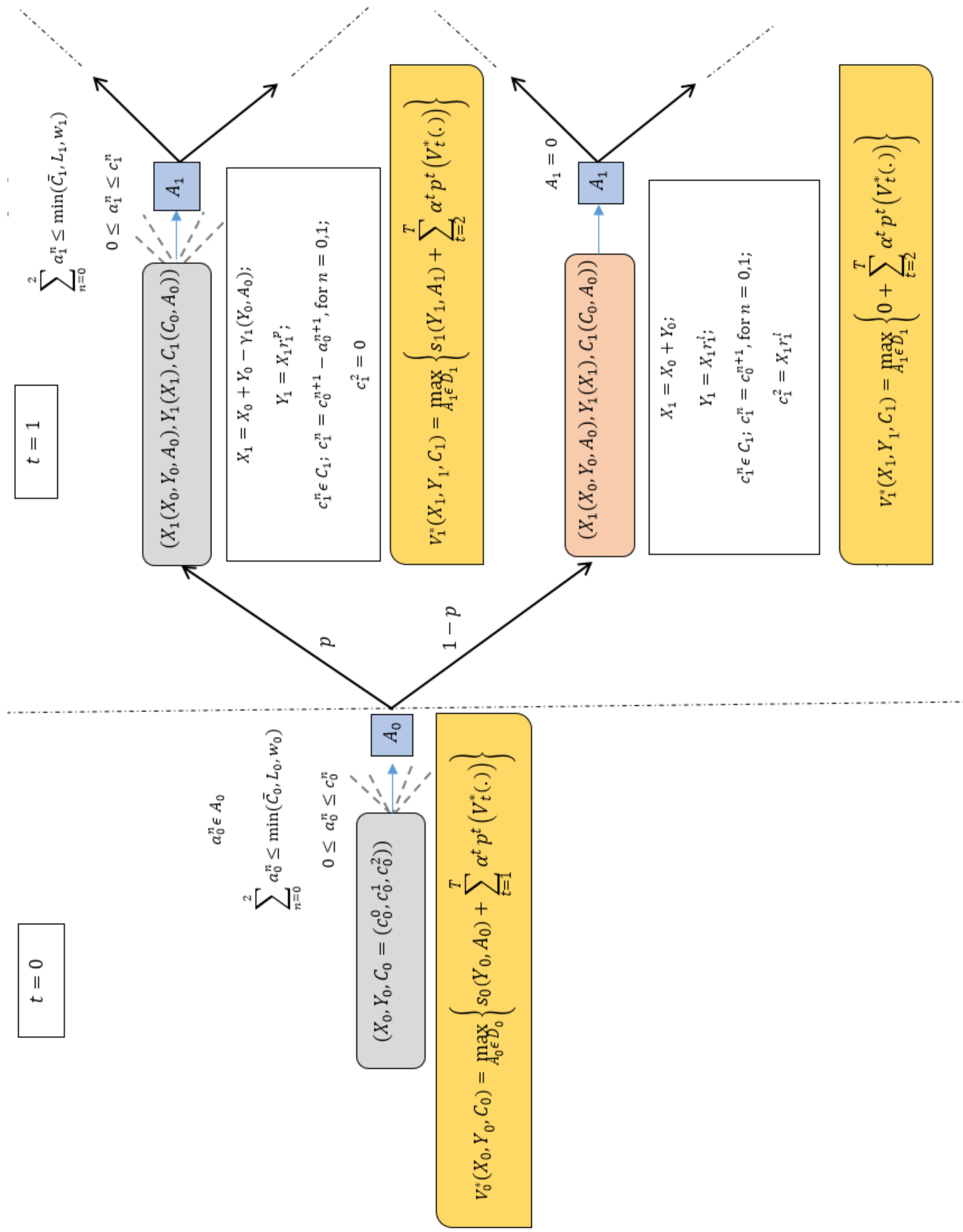

Figure 2.1: Model with Three Carryforwards and "m" Tax Brackets

2.6.2 Simulations Examples

The algorithm for the model has been programmed in *Matlab* to simulate results and compare the value of the TLCFs when different inputs to the model are changed. The complete *Matlab* code has been given in the Appendix. We present two examples using the *Matlab* code. The first example EX.A, has been generated using the following input values:

- Finite time horizon, $T = 2$.
- Initial wealth of $X_0 = \$500$.
- Initial taxable income, $Y_0 = \$150$.
- Initial set of TLCFs given, $C_0 = (c_0^0, c_0^1, c_0^2) = (\$100, \$75, \$150)$.
- Tax rates, ${}_t = (\tau_t^l, \tau_t^h) = (0.2, 0.35)$, for $t = 0, 1, 2$.
- $B_t = (0{,}200)$, for $t = 0, 1, 2$.
- Returns, $r_t = (r_t^p, r_t^l) = (0.2, -0.1)$, for $t = 0, 1, 2$.
- Probability $p = 0.5$.
- $L_t = \$500$, for $t = 0, 1, 2$.
- Discount rate, $\alpha = 0.95$.

Solution:

The results of the above input values are shown using a tree diagram as followed in Figure 2.2. The solution for the value of the given set of TLCFs,C_1, for the above given input values comes out as \$62.62. With an increment of 0.01 (down to the penny) for the action set, the number of actions per state is fairly large, so we replace the explicit enumeration of the entire tree with a heuristics policy to evaluate what might happen after we reach a certain state[5]. We select a random action sum $\bar{a}_t = \sum_{n=0}^{N} a_t^n$ for each positive state for all t from $t = 0, 1, \cdots, T-1$. At the final time $t = T$, all the available TLCFs are used if in the positive state. The value of the maximum savings of this trajectory is then used as an estimate of the value of being in that state. This process is repeated over$K$ iterations to cover most of the possible actions in each of the state and reach an approximate optimal solution for the value function. In the below example, the action at time $t = 1$ is chosen randomly and the followed trajectory is shown. This process is repeated for $K = 1000$ iterations to get the approximate maximum value of savings as \$62.62 with the optimal action vector of $A_0 = (\$100.00, 46.10, 0)$, to use most of the available TLCF at time $t = 1$ because tax rates remain the same in the future.

The below example EX.B is to demonstrate how the optimal action changes when tax rates are expected to increase in the future. All the inputs remain the same, except for $_2 = (\tau_2^l, \tau_2^h) = (0.4, 0.55)$ and $_3 = (\tau_3^l, \tau_3^h) = (0.6, 0.75)$ now.

Solution: Value of the Savings (V_0) from the given set $C_1 = (\$100, \$75, \$150)$ is \$128.36 with the optimal action being $A_0 = (a_0^0, a_0^1, a_0^2) = (\$100, \$4.21, \$0)$. Here the optimal action is save the TLCFs for use at time $t = 2$, when the tax rates are higher and the savings coming from them are also higher even after the discount rate is applied. Lemma 2.1 (rationality) holds.

[5] The tree diagram represents the Roll-out Heuristics Tree Search Approach.

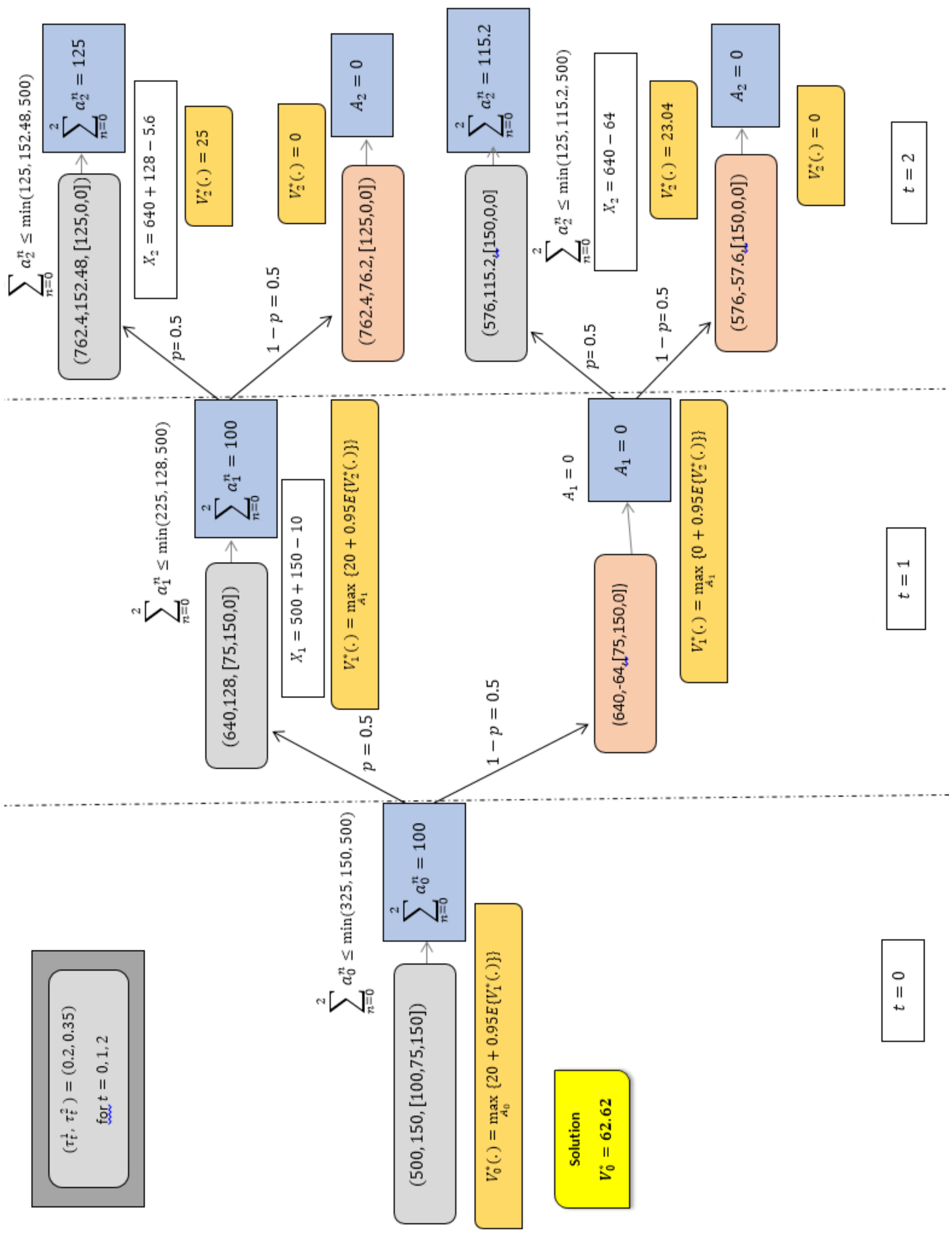

Figure 2.2: Three Period Model with Three Carryforwards and Two Tax Brackets (EX.A)

Example with Non-FIFO Seqeunce of Usage

Consider the same input values as in Example EX.A. We will use this example to show that a non-FIFO sequence of usage for the TLCFs in set C_1 will result in much lower savings in terms of expectations than the FIFO usage sequence even if the tax rates remain constant throughout time. Just as in Example EX.A, letâĂŹs assume that we start with the profit node of $Y_0 = \$150$ and we decide to use $\bar{a}_0 = \$150$ from the given set of TLCFs, C_0. The saving in time $t = 0$ would then be given by $s_0(Y_0 = w_0, A_0) = \$30.0$. However, we violate the FIFO sequence in determining the set of TLCFs to be carried forward to time $t = 1, C_1$. Given Lemma 1, we use the \$100 expiring now and the remaining \$50 is used from the TLCF expiring last, that is we use c_0^2, before we use c_0^1. Then $C_1 = (c_1^0, c_1^1, c_1^2) = (\$75, \$100, \$0)$. If in the next time period, $t = 1$, we end up in the profit state, we can still get savings of $s_1(Y_1 = w_1, A_1) = \$26.0$. But if the node is a loss node instead (with a probability of 0.5), the savings would be $s_1(Y_1 = u_1, A_1 = 0) = \$0$ and $C_2 = (c_2^0, c_2^1, c_2^2) = (\$100, \$0, \$0)$. So even if we end up on a positive node at time $t = 2$, we can only get savings of $s_2(Y_2 = w_2, A_2) = \$20$. So the expected discounted value of these savings at time $t = 0$, will be \$49.34.

On the other hand if the FIFO sequence was used, the expected discounted savings would be \$62.20. This is because the explicit loss of $c_1^0 = \$75$ expiring unused is eliminated. In addition to that the implicit loss of not being able to use the portion c_0^2 which was used at $t = 1$, \$50, could have been used at $t = 2$ instead. This difference between the savings from FIFO and non-FIFO will be smaller if the probability of loss is higher than the probability of profit, but even then the FIFO results in higher savings in terms of expectations.

However, if in case we incur a loss at time $t = 2$, such that the saved portion of c_0^2 due to FIFO cannot be used anymore, a non-FIFO sequence gave us an implicit benefit of $\$50\tau_1^h$ because if we had used FIFO, this portion of c_0^2 would have expired unused in time $t = 2$.

2.6.3 Example with TCLBs:

We use this example EX.C to show how Section 2.5 can be used to add TLCBs to the model. The input values here are the same as Example EX.A. The example calculations here will be done taking into consideration that now when a new loss is made, it can also be carried back and not only forward. At time $t = 1$, if we make a loss, u_1, we can not carry it back to $t = 0$ because we made no profit then to apply it against that. So we can only carry u_1 forward just like the model where there are no TLCBs. At time $t = 2$, if we generate a loss, we can carry it back if we had made a profit at $t = 1$. So the loss $u_2 = \$76.2$ can be carried back as well as carried forward. Applying the formula in Section 2.5, this loss of $u_2 = \$76.2$ will give us reclaimed taxes of $h_q(u_2)$. Here $q = 1$ because it can only be carried back one period. Recall from Example EX.A that $\sum_{n=0}^{2} a_1^n = \$100$, $f_1^1 = \$28$ and $f_1^2 = \$0$.

$$\begin{aligned} h_1(u_2) &= ((76.2 - 0) \wedge 28)\tau_1^l + (0 \wedge 76.2)\tau_1^h \\ &= s_2(u_2, A_2) = \$5.6 \end{aligned}$$

where, $Y_2 = u_2$. So, the new after tax wealth at time $t = 3$ is given by:

$$X_3 = X_2 - u_2 + s_2(u_2, A_2)$$

$$X_2 = 762.4 - 76.2 + 5.6 = 691.8$$

The remaining loss from $u_(2)$ of $\$48.2$, can now be added to c_3^2 to be used in time $t = 3$. The new savings from the TLCBs of $s_2(u_2, A_2)$ will now replace the savings of $\$0$ at the loss node in the calculations for V_0^*. Therefore the new value generated for the combined set of TLCFs and TLCBs would include the discounted $h_1(u_2)$, to bump up the value to \$67.94.

2.7 Valuing TLCFs for a Risk Averse Investor

2.7.1 Introduction to the Risk Averse Investor

The valuation model in the above sections is specific to a risk neutral investor, therefore we only maximize the value of the savings, which is equivalent to maximizing the linear utility of savings. In this section we consider the case of a risk averse investor. We make the conjecture that introducing a concave utility function in the above model, will only change the value of the TLCF set given at time $t = 0$ and have no effect on the action set. Specifically we are interested to see how the actions change if instead of maximizing savings, we maximize the utility of these savings from TLCFs in order to incorporate the different risk preferences of an investor (individual or corporate)[6]. We assume the same model set up as earlier, except for a small change, which is the addition of a utility function in the equation for the value of savings.

Consider Eq. (2.2)-Eq(2.5) from Proposition 2.1 reproduced below for ease of reading;

$$V_t^*(X_t, Y_t, C_t; R, H_{t-1}) = \max_{A_t \in D_t} \{W_t(X_t, Y_t, C_t; A_t)\} \text{ for } 0 \leq t \leq T$$

[6]More assumptions and details about the utility preferences of the risk averse investor is given in the following chapter when the consumption-investment model of a risk averse investor is re-visited with taxes and TLCFs included

with

$$W_t(X_t, Y_t, C_t; A_t) = \begin{cases} s_t(Y_t, A_t) + \alpha \mathbb{E}_t^R[V_(t+1(X_{t+1}, Y_{t+1}, C_{t+1}(C_t, A_t); R, H_t)] \text{ if } Y_t = w_t \\ \alpha E_t^R[V_(t+1(X_{t+1}, Y_{t+1}, C_{t+1}(C_t, A_t); R, H_t)] \text{ if } Y_t = u_t \end{cases}$$

With terminal conditions $t = T$ given by:

$$W_t(X_t, Y_t, C_t; A_t) = \begin{cases} s_T(Y_T, A_T) \text{ if } Y_T = w_T \\ 0 \text{ if } Y_T = u_T \end{cases}$$

$$V_T(X_T, Y_T, C_T; R, H_{T-1}) = \begin{cases} \max_{A_T \in D_T}\{W_T(X_T, Y_T, C_T; A_T)\} \text{ if } Y_T = w_T \\ 0 \text{ if } Y_T = u_T \end{cases}$$

We modify the above equations to incorporate the different risk preferences of the investor here, so Eq (2.3) and (2.4) to get R*, the optimal prescription, given by the following:

$$W_t(X_t, Y_t, C_t; A_t) = \begin{cases} U(s_t(Y_t, A_t) + \alpha \mathbb{E}_t^R[V_(t+1(X_{t+1}, Y_{t+1}, C_{t+1}(C_t, A_t); R, H_t)] \text{ if } Y_t = w_t \\ \alpha E_t^R[V_(t+1(X_{t+1}, Y_{t+1}, C_{t+1}(C_t, A_t); R, H_t)] \text{ if } Y_t = u_t \end{cases} \tag{2.15}$$

With terminal conditions $t = T$ given by:

$$W_t(X_t, Y_t, C_t; A_t) = \begin{cases} U(s_T(Y_T, A_T) \text{ if } Y_T = w_T \\ 0 \text{ if } Y_T = u_T \end{cases} \tag{2.16}$$

So Eq (2.2) and Eq. (2.5) are still the same but W_t is now defined as in Eq (2.15) and Eq (2.16). We apply the same method of backward induction to get the maximum value of the utility of savings as in Chapter 2.3.

2.7.2 Example: Simulations with Risk Averse Investor in the Model

In this example, we assume that the utility of the savings is given by the following isoelastic function:

$$U(s_t(Y_t, A_t), t) = \frac{s_t(Y_t, A_t)^{1-\eta} - 1}{1-\eta} \forall t \tag{2.17}$$

This is the form of power utility is commonly used in the finance literature because it appears plausible that the usage of CRRA utility function could be a reasonable approximation of the real-world behavior. Unfortunately, it turns out to be somewhat complicated to find an appropriate risk aversion parameter, η. Some papers like Mehra and Prescott (1985), apriori impose an upper bound of 10 for the risk aversion parameter. Janecek (2004) argues that for the purposes of taking investment risks, it is reasonable to assume $\eta > 10$ even for investors experienced in risk taking, and significantly larger η in order of 30 or more for standard households. An attempt to measure a real risk aversion was performed by Binswanger (1981) in rural India, by offering a set of positive games to randomly chosen farmers in India. The farmers could decide between accepting certain monetary sum and participating in a gamble with better expected return but with risk. Binswanger obtains a reasonable size of relative risk aversion under the assumption of a utility function with constant partial risk aversion in the neighborhood of the payoff levels. Taking the suggested wealth level of 10,000 as given, the experiment suggests that the relative risk aversion of individual farmers under he assumption of power utility is in the range of 10-30. Several other authors argue that the risk aversion parameter could be around 2. In our examples, we use the utility function given in Eq. 2.17, with 2 values of the parameter, η. Though the values derived for the set of TLCFs would depend on the selection of a value for η, it does not affect the optimal action sequence, which remains the same as the one solved for without utility functions. In short, the model gives the same action policy for a risk neutral as for a risk averse investor. We modify the Matlab code given in Appendix A.3 to Appendix A.4, to maximize the utility of savings, Eq. 2.17 at each time period with action as the control variable. In order to compare how the addition of a risk averse utility function changes the value of the TLCF set given and the action set. Example EX. D, has been generated using the following input values as Example EX.A:

- Finite time horizon, $T = 2$
- Initial wealth of $X_0 = \$500$
- Initial taxable income, $Y_0 = \$150$
- Initial set of TLCFs given, $C_0 = (c_0^0, c_0^1, c_0^2) = (\$100, \$75, \$150)$
- Tax rates, ${}_t = (\tau_t^l, \tau_t^h) = (0.2, 0.35)$, for $t = 0, 1, 2$.
- $B_t = (\$0, \$200)$, for $t = 0, 1, 2$.
- Returns, $r_t = (r_t^p, r_t^l) = (0.2, -0.1)$, for $t = 0, 1, 2$.
- Probability $p = 0.5$
- $L_t = \$500$, for $t = 0, 1, 2$.
- Discount rate, $\alpha = 0.95$
- $\eta = 2$

Solution:

The value of the above given set of TLCFs comes out to be \$54.694 when solved without the utility function as in Example EX.A. The optimal actions at time $t = 0$

and $t = 1$ are given by:

$$\bar{a}_0 = 149.97 \text{ and } \bar{a}_1 = 129.99$$

$$A_0 = (\$100, \$49.97, \$0) \text{ and } A_1 = (\$25.03, \$104.969, \$0)$$

The same input, but this time with the addition of utility function given in Eq. (2.17) and $\eta = 2$, gives the value of the above given set of TLCFs as \$1.88. The optimal actions at time $t = 0$ and $t = 1$ remain the same:

$$\bar{a}_0 = 149.88 \text{ and } \bar{a}_1 = 129.99$$

$$A_0 = (\$100, \$49.88, \$0) \text{ and } A_1 = (\$25.12, \$104.88, \$0)$$

This value reduces to \$0.217 when $\eta = 10$. But the optimal actions still remain the same. The optimal actions at time$t = 0$ and $t = 1$ remain the same:

$$\bar{a}_0 = 149.95 \text{ and } \bar{a}_1 = 129.99$$

$$A_0 = (\$100, \$49.95, \$0) \text{ and } A_1 = (\$25.05, \$104.95, \$0)$$

CHAPTER 3

Multi-Period Dynamic Volume Allocation: An Electronics Sourcing Case Study

3.1 Introduction to Electronics Component Sourcing

Outsourcing, especially in the manufacturing sector, has become a critical strategic decision that can allow organisations to develop and leverage their capabilities. There are different processes that can be entrusted to other companies from an offshore location, allowing the business to focus it's resources on core business instead. The most obvious advantage of outsourcing stems from the cost savings that it brings. Difference in wages between Western and Asian countries allow for significant labor cost reductions, keeping quality constant or potentially higher in some cases. In addition to the reduced cost offering, outsourcing also brings in the benefit of focuing on business competencies. Each party brings in their expertise in the pool, thus doing a better job with their knowledge and understanding of the domain. It also frees up resources to focus on research and development to move to providing higher value added services.

One of the forms of outsourcing is Contract Manufacturing. Many industries including aerospace, defense, computer,energy, electronics are currently adopting this business model of using a contract manufacturer (hereon refered to as the "CM"). These are Original Equipment Manufacturers, typically refered to as OEMs that use contract manufacturers the most. Such companies usually don't buy components and parts based on value but based on cost. These componnets (plus manufacturing processes) tend to be ones that they can do themselves otheriwse. Because they can do it themselves, they are likely to have some idea of what it costs. Therefore they want to compare the cost of doing it themselves to the price of having someone else do it. The value is in not having to do it on thier own, which can only be understood by comparing the cost of doing it inhouse with the higher price of having someone else do it. For instance, electronic contract manufacturers (CMs) build products designed by their OEM customers, who know the approximate cost to manufacture their product. The OEM must choose between manufacturing themselves or outsourcing. Because the OEM knows the approximate cost of manufacturing and could choose to build it themselves (at least theoretically), CMs use cost plus to determine price. For the CM, the basic components of cost are materials, direct labor, and overhead. Typically, the direct labor,overhead costs and profit are combined into the "Manufacturing Value Add" (MVA).

The way the supplychain works for such a business model is summarized in Figure 3.1. The OEM firm owns the design/formula for the product with which they approach the CM. The CM will quote the parts on material, processes, labor and tooling costs. Typically the hiring OEM firm will request quotes from multiple CMs. After the bidding process is complete, the firm will select a source, and then, for the agreed-upon price, the CM acts as the hiring firm's factory, producing units of the product on behalf of the firm. This can be the sub-assembled product shipped to the firm for complete assembly or sometimes shipping units of the product on behalf of the firm to customers (direct fulfilment), as shown in Figure 3.1.

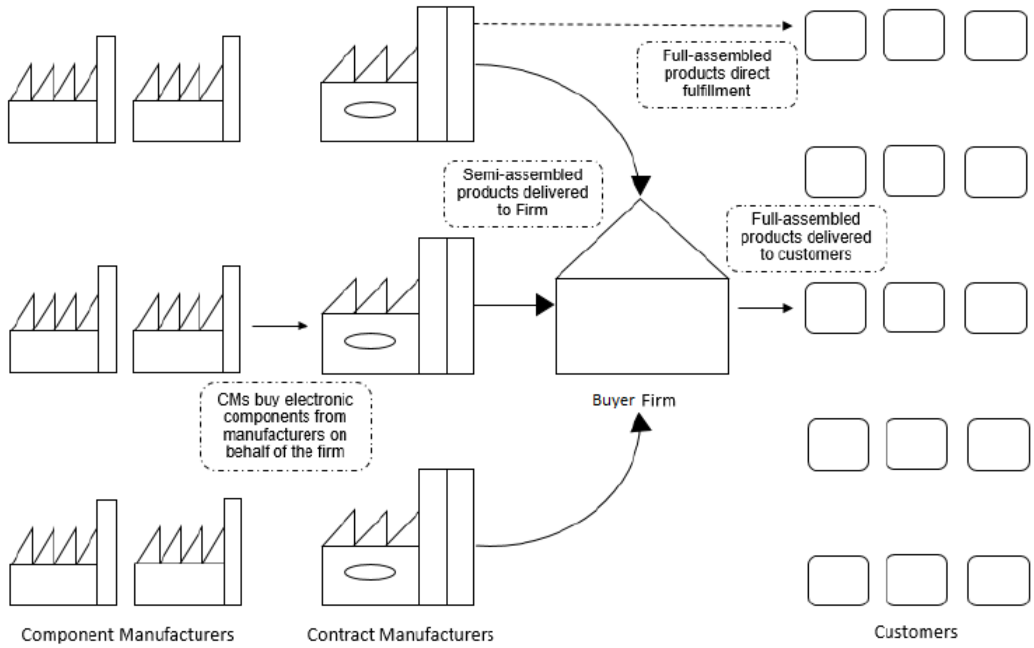

Figure 3.1: Overview of Supplychain Disaggregation Model Using Contract Manufacturers

1. The figure shows there are multiple components that could go in the manufacturing of the full product. Some of these components could be custom-made parts which are specific to the design of the product, in which case the firm may chose to negotiate directly with the component manufacturer and provide agreed pricing schedules to the CM for purchase of the components.

As a part of the decision making process to select source CM(s), the firm comes across the dilemma of deciding between having one supplier and using the volume as leverage for concessions to get the best possible cost advantage, or considering the argument of business risk which basically translates into not putting all your eggs into one basket. Having only one supplier qualified for the business puts forth a huge risk of failure to deliver and eventually shut down of production. This is why most manufacturers opt for the multi-supplier model where they have dual or more procurement requirements. Having more than one supplier qualified at the same time mitigates the risk of failing to meet customer demands due to unavailability of supplies. However this later option takes away the volume leverage causing cost of components to go up or even worse, not being able to meet the MOQs in the market. This problem is particularly noticeable for small to medium scale firms who do not have plenty of business to offer to suppliers anyway. In this essay we investigate such a situation in which an electronics manufacturing firm outsources its assembly operations to contract manufacturers. We use numerical methods to study different volume allocation scenarios first in a one period setting, later extending the analysis to a multiperiod model. Each contract manufacturer is assumed to have a different level of improvement capability of inducing supply cost reduction, which will be beneficial for the maufacturing firm. Three types of CMs are considered based on their size: (i) one which is a well-established, large contract manufacturer with many major accounts and hence the ability to offer lower overhead costs, (ii) one which is a medium sized player in the market with moderate to good capabilities, and (iii) one which is a new player in the market and relatively small compared to the above two types. They way these differences in the CMs is incorporated in the model is through thier bid inputs. A large well established CM, will normally be able to provide better overhead costs compared to a small one, by definition. On the other hand, they may bump up thier manufacturing value add piece of the total cost by charging a higher profit margin due to their brand power. A smaller CM which is a new player in the market is more likely to be very hungry for the business and may bid aggressively on the component cost piece of the total cost plus charge a smaller profit margin just to get their foot in the door. The decision problem faced by the manufacturing company is twofold: (a) how many contract manufacturers to get involved in business with (one supplier model vs n supplier model, where $n > 1$), (b) how much volume should be allocated to each contract manufacturer, if the multiple supplier model is chosen. A simulation model helps us develop a set of results, which can numerically solve the decision problem. Numerical examples are employed to demonstrate how the analysis can be utilized in a real-world setting.

Section 3.2 of this essay presents a crirtical review of past work on the subject of volume allocation and supplychain analysis of models with similar structures. It also addresses the question of why this study is important, by giving the areas where applications in a real life setting are possible and beneficial. In Section 3.3, the problem of optimal volume allocations is formulated as a linear programming model. The latter considers features that the earlier models have failed to consider. Section 3.4 extends this formulation to take into account the uncertainity in customer demands. In Section 3.5, a case study is presented to illustrate the applicability of the model.

3.2 Literature and Applications

Many big multi-national companies today benefit from the services of CMs. For instance, Apple Inc. uses Taiwan-based giants Hon Hai Precision Industry (Foxconn) and Pegatron as it's CMs to manufacture iPads and iPhones in China [Chan et al. (2013)]. Bigger companies often have specialist component engineers in charge of selecting, sourcing, testing and qualifying parts, as well as maintaining approved parts lists and finding substitutes for obsolete components. But design engineers at smaller firms, who cannot afford the overheads of a dedicated component engineer, can often find themselves involved in or in charge of the component sourcing process, and liaising with production over parts sign-offs [Leung (2013)]. Our study aims at mathematically modeling the steps in the sourcing process, automating this time consuming process so that the firm trying to make the decision on the cost benefit analysis of which suppliers to use and at what volume allocations in order to derive the maximum possible savings, given the inputs of the model and the business constraints. The model has been simplified on assumptions but it still is general enough to adapt to business requirements, thus providing room for customization and further extensions to facilitate application in real life settings. The size and capabilities of the hiring firm using this model will automatically be incorporated in the bids that the CMs will provide based on the size of the prize, hence it is organically taken care of in the set up of the model.

Existingt literature on the topic is limited. There are only a few papers that try to study this specific contract manufacturer based model in a mathematical franework. [Kim (2003)] seems to be the only study that actually looks into the supplychain model for CMs more closely in an analytical way. In his paper, Kim investigates a situation in which a manufacturing company outsources its assembly operations to two contract manufacturers, taking into account time (as a dynamic factor) and processing level (in terms of assembling) simultaneously. Each contract manufacturer is assumed to have a different level of improvement capability of inducing supply cost reduction that, in turn, benefits the manufacturing company. He tried to answer the question, how much should be outsourced to each contract manufacturer (i.e., less capable or more capable); and (b) how processed (in terms of assembling) should the semi-finished units be when returned from the contract manufacturers getting to a set of mathematical results, which can solve the decision problem. The rest of the literature on the subject is conceptual which either focuses on the advantages/disadvantages of the outsourcing of the process, make Vs buy decision or the framework that explores the linkage between the evolution of global production networks or the formation of capabilities by local suppliers ([Ernst and Kim (2002)], [Chan and Chung (2002)], [Van den Bossche et al. (2014)]) .

The need for a multi-period dynamic sourcing model still remains. It involves a major business decision that OEM firms usually have to make, should they have one strategic partner in the form of a contract manufacturer, both growing together, merging technology roadmaps. This strategic decision not only provides maximum savings potential due to economies of scale and volume discounts but also creates a true strategic alliance commited to long-term collaboration. The high savings potential comes from scale benefits and volume dsicounts. However, this decision of having a sole supplier leads to supply disruption risk. Even when there are no quality issues and defects and quality standards are constantly met, inevitable risk of natural disasters leading to factories being subject to shut down still exists. Incidents of earthquakes and floods on factory sites are not unheard of and cannot be planned for completely. This needs strong governance structures to sustain value and share business risk. That is why most OEM firms want multiple suppliers qualified for different business segments, minimizing the supply disruption risk. On one hand, this tackles the risk mitigation but results in less savings than the single supplier model due to loss of volume discounts. Rebates end up being lower and harder to negotiate to cover Non-Recurring Expenses (NRE). The savings leakage due to volume creep and loss of leveraged buy and incremental overheads to manage multiple suppliers is particularly higher for medium to smaller OEM firms.

The question that rises is, as a manufacturer if you are to have two or more suppliers, what should be the optimal volume allocations amongst the n (where $n \geq 2$) suppliers with the objective that the best possible price is achieved for the business? Our model that will spit out the answer to the above question will depend not only on the bids put forth by the suppliers (CMs) but also factors such as non-recurring facility expenses, currency fluctuations, lead times, delivery methods, contract terms and conditions (including payment terms) and other qualitative aspects which could include perks like designated account managers, employees on sight, quality of engineers and testers etc. Hence when you consider all these aspects that are a part of the final decision making process to award business to suppliers, it soon becomes too convoluted and not objective anymore. The applications for such a model are tremendous in many industries, but not limited to, especially the aerospace, defense, computer, semiconductor, energy, medical, food manufacturing, personal care, and automotive fields. Some types of contract manufacturing include complex assembly, aluminum die casting, grinding, broaching, gears, and forging. Even the pharmaceutical industry uses this process with CMs called contract manufacturing organizations. A contract manufacturing organization (CMO), sometimes called a contract development and manufacturing organization (CDMO), is a company that serves other companies in the pharmaceutical industry on a contract basis to provide comprehensive services from drug development through drug manufacturing. This allows major pharmaceutical companies to outsource those aspects of the business, which can help with scalability or can allow the major company to focus on drug discovery and drug marketing instead. Although our model focuses on an electronics component case study, the inputs have been designed in a way that components could be customized to whichever industry the OEM firm operates in. The basic structure of the process and the problem at hand remain the same even in other industries such as the pharmaceutical industry. Business constraints may differ but can be incorporated easily.

3.3 Model Setup

This work considers the design of multiproduct and multi-echelon production across more than one contract manufacturers. We consider the supplychain framework to start from the component manufacturers, although in reality it is the raw material suppliers that are at step zero. In order to simplify and keep the focus of the model on the processes between the OEM firm and the CM, we assume that all component manufacturers have the same capabilities to acquire raw materials from the market and there are no issues from the component manufacturer's end. Hence for us the supplychain process starts at the component manufacturers providing bids to CMs for the materials. To explain this better with the help of an example, consider an electronic product like the iPhone. Apple Inc. will here be the OEM firm looking for a source contract manufacturer. The contract manufacturers that could be invited to the bid process will be companies like Foxconn, Flextronics, Jabil, Sanmina (who are the bigger players in the market) or smaller players like Celestica and Benchmark Electronics. The invited CMs will then try to bid to produce the iPhone but they will need to get the quotes of the different components that go into manufacturing the iPhone from the individual component manufacturers. One such component is the lithium ion battery that goes into the iPhone. All CMs will then approach the battery manufacturers to get the price of the component. The cost for all components combined will make up the material cost for the iPhone for each CM, upon which they will decide how much Manufacturing Value Add (MVA) cost to add. This MVA portion of the cost is usually made up of frieght in, labor, overhead, SG&A and profit margin costs. The total cost for each product will be a quote provided by each CM to Apple. In addition to the cost provided, CMs also give volume discount options to secure a higher share of the business which are contingent on certain level of order quantity being met. This will be the main focus of our work which helps decide what level of volume discounts make an OEM firm reconsider it's multi-supplier model and trade it in for a sole-supplier model to take advantage of the huge savings in terms of volume discounts.

As shown in Figure 3.2, the network consists of a number of existing multiple products. Let p represent the numbers of products our OEM firm manufactures at the current date, where $p = 1, 2, \cdots, P$. A finite number of CMs are invited to bid on all the p products that the firm sells. Let j be the number of CMs that are invited to bid on the OEM firm's business, where $j = 1, 2, \cdots J$. Usually a firm will not invite more than 8-10 CMs to bid since the process is already complicated and adding more CMs does not really add further value to the process. Only a few select CMs are requested to provide bids based on past working relationships and market reputation. Let the

u_p be the number of unique components that go in the making of each product p, where $u_p = 1, 2, \cdots U_p$ for each $p = 1, 2, \cdots P$. Since the products sold by the OEM firms are bound to have similar components going in them, as they are all assumed to be different SKUs of the similar product, we provide the possibility of having components be similar across products. For instance, the same battery may be used in an iPhone and an iPad. Let s be the number of components which are similar in atleast two products. Then the total number of unique components for all products will be given by $\hat{u} = \sum_{p=1}^{P} U_p - s$.

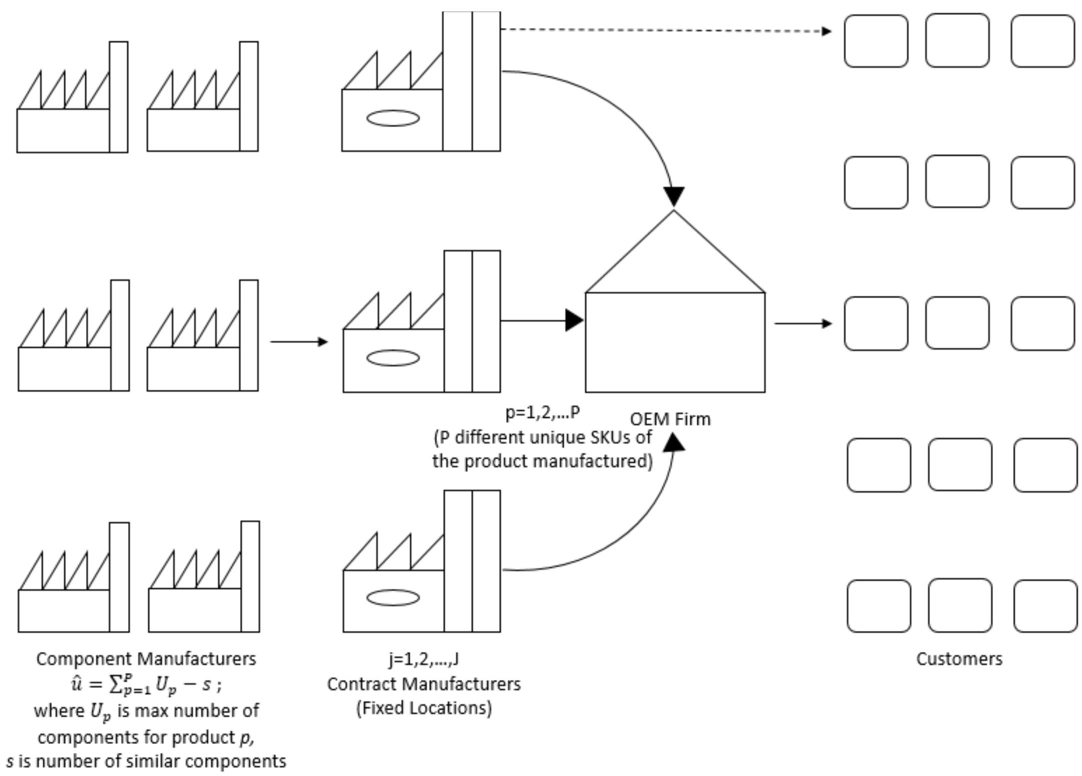

Figure 3.2: Supplychain Disaggregation Model Using Contract Manufacturers

We assume that all the J contract manufacturers that are a part of the bidding process are producing at one fixed location, so that the freight cost is the same for all of them. This is not an unrealistic assumption because the factory sites in Mexico, China or Vietnam are usually located in the same vicinity for all these major CMs in reality too. We also assume that the production capacity for all the CMs is always higher than demand, which means they can always fullfill orders on time. In the CM based business model, inventory is usually stored at the CM's warehouse and they retain ownership untill the consignment leaves their facility. This is why the CM may or may not also charge an inventory holding cost as a part of the total cost. To keep the model focused on the volume allocation problem, we will assume that inventory holding is not a problem for the CM, i.e. there is enough space and resources available to hold inventory for a few months based on lead times. Although we have included the extra charge of inventory in some cases as a part of the bid that the CM places. But the assumption is that if one CM is agreeing to hold inventory for the OEM, all CMs will match that offer to be competitive.

The OEM company places all its P products for bid to the J CMs shortlisted for the process. All the J CMs are assumed to have completed a qualitative assesment to be selected for the bidding round. The different qualitative factors considered in shortlisting these CMs include:

- Financial Stability: assesment of financial statements for the past three years
- Industry Participation: main business sector that the CM operates in is the same as the OEM's. assesment of product handling experiences
- Customer Support: dedicated account managers
- Human Resources: number of quality engineers
- Information technology support system: data transfer and eletronic billing capabilities
- Geographic coverage of shipments: coevrage of direct fulfilment orders to customers

3.3.1 Variables

Variable	Description
C	$c_i \in C$; baseline component cost (in dollars) for each component, $i = 1, 2, \cdots \hat{u}$
M	$m_p \in M$; baseline manufacturing value add (MVA) cost for each product p, where $p = 1, 2, \cdots P$
Q	$q_{pi} \in Q$; q_{pi} is the quantity of component i (in units) that goes in product p, where $i = 1, 2, \cdots \hat{u}$ $p = 1, 2, \cdots P$
D	$d_p \in D$; demand (in unit)s for the each product p, where $p = 1, 2, \cdots P$
K^j	$k^j_{pi} \in K^j$; k^j_{pi} is the component cost (in dollars) for component i bid by CM j for product p, where $i = 1, 2, \cdots \hat{u}$ $p = 1, 2, \cdots P$ $j = 1, 2, \cdots J$
L^j	$l^j_p \in L^j$; l^j_p is the MVA cost (in dollars) bid by CM j for product p, where $p = 1, 2, \cdots P$ $j = 1, 2, \cdots J$

The OEM firm under consideration is assumed to be currently paying a cost for the manufacturing of its P different products. This cost is a combination of the material cost (component cost) and the manufacturer value add (MVA) cost. We call this current cost the baseline cost because the decision to chose a contract manufacturer from the bidding process will depend on the maximization of the savings as compared to this baseline cost. The assumption here is that if the bidding process to invite new and old contract manufacturers to bid on the OEM's business was not initiated, the OEM would be paying this baseline cost indefinitely. So the bidding process will allow the OEM to consider and chose suppliers that provide a lower cost compared to the baseline, resulting in savings.

The total baseline cost (in dollars) for the OEM isthus given by:

$$B = C.Q^T + M \tag{3.1}$$

$$\text{where } b_p \in B; p = 1, 2, \cdots P$$

As discussed earlier, we are assuming that the preliminary round of shortlisting CMs was conducted based on the qualitative factors described above, as a result of which J CMs were invited to bid on the business. The purpose of this bidding process is to identify and partner with suppliers (contract manufacturers) who can provide electronic manufacturing services to the OEM firm. One important assumption that needs to be highlighted here is that it has been communicated to all the invited CMs that while submitting their quotes for specific products, they have to keep in mind that the OEM firm will consider awarding all products volumes based on the cost estimates which means the lowest cost CM will be awarded the business but not neccessarily a 100% of the volume will be given to one single supplier. This means they need to make an assumption while providing their cost estimates that they will be awarded only the minimum order quantity (MOQ) of each product. Any additional volume on top of that MOQ will be awarded based on volume disounts offered. This is further explained through an example below (numbers are illustrative only for explanation purpose).

Let's say that there is a product named "P1" and let's assume that it has an annual demand of 100 units.Let's further assume that the MOQ for that product is 20% of the total demand. Here MOQ is defined as the minimum order that the OEM firm is willing to give to any one partcular CM because awarded volume below that MOQ will not be worth the cost of resources to manage that CM.The CM's quote for "P1" should assume that they will be awarded 20 units in volume. In other words, the CM should not submit their quotes for product "P1" assuming that they will be a contract manufacturer for all the 100 units in demand. This assumption ensures that the OEM firm can breakdown the volume amongst multiple CMs based on cost and mitigate supply diruption risk.

Based on the above assumption then, and using Kronecker delta from linear algebra, we calculate the total product cost estimate for CM "j" from the bids for component cost and MVA cost as:

$$\bar{K}^j = \text{Total component cost by product} = \sum_i e_i^T (K^j . Q^T) \mathbf{X_i} \tag{3.2}$$

$$L^j = \text{Total MVA cost by product}$$

$$T^j = \text{Total product cost} = \bar{K}^j + L^j \tag{3.3}$$

where e_i is the i-th canonical base vector and $\mathbf{X_i}$ is the projection on the i-th coordinate; $i = 1, 2, \cdots, P$.

References

Eli Amir and Theodore Sougiannis. Analysts' interpretation and investors' valuation of tax carryforwards*. *Contemporary Accounting Research*, 16(1):1–33, 1999.

Eli Amir, Michael Kirschenheiter, and Kristen Willard. The valuation of deferred taxes*. *Contemporary Accounting Research*, 14(4):597–622, 1997.

Eli Amir, Michael Kirschenheiter, and Kristen Willard. The aggregation and valuation of deferred taxes. *Review of Accounting Studies*, 6(2-3):275–297, 2001.

Alan J Auerbach and James M Poterba. Tax loss carryforwards and corporate tax incentives. In *The effects of taxation on capital accumulation*, pages 305–342. University of Chicago Press, 1987.

Benjamin C Ayers. Deferred tax accounting under sfas no. 109: an empirical investigation of its incremental value-relevance relative to apb no. 11. *Accounting Review*, pages 195–212, 1998.

Rajiv D Banker and Inder S Khosla. Economics of operations management: A research perspective. *Journal of Operations Management*, 12(3-4):423–435, 1995.

Mark P Bauman and Somnath Das. Stock market valuation of deferred tax assets: evidence from internet firms. *Journal of Business Finance & Accounting*, 31(9-10): 1223–1260, 2004.

Hans P Binswanger. Attitudes toward risk: Theoretical implications of an experiment in rural india. *The Economic Journal*, 91(364):867–890, 1981.

David N Burt. *Proactive Procurement: The Key to Increased Profits, Productivity, and Quality;[a Step-by-step Procedural Guide to Move from a Reactive Purchasing Activity to a Proactive Profit-making Procurement System]*. Prentice-Hall, 1984.

Terence T Burton. Jit/repetitive sourcing strategies:'tying the knot'with y. *Production and Inventory Management Journal*, 29(4):38, 1988.

Jenny Chan, Ngai Pun, and Mark Selden. The politics of global production: Apple, foxconn and china's new working class. *New Technology, Work and Employment*, 28(2):100–115, 2013.

Michael FS Chan and Walter WC Chung. A framework to develop an enterprise information portal for contract manufacturing. *International Journal of Production Economics*, 75(1):113–126, 2002.

Paul K Chaney and Debra C Jeter. The effect of deferred taxes on security prices. *Journal of accounting auditing and finance*, 9(1):91–116, 1994.

Cheryl Chang, Kathleen Herbohn, and Irene Tutticci. Market's perception of deferred tax accruals. *Accounting & Finance*, 49(4):645–673, 2009.

Kevin CW Chen and Michael P Schoderbek. The 1993 tax rate increase and deferred tax adjustments: A test of functional fixation. *Journal of Accounting Research*, pages 23–44, 2000.

Cyrus Derman and Cyrus Derman. Finite state markovian decision processes. Technical report, 1970.

Dieter Ernst and Linsu Kim. Global production networks, knowledge diffusion, and local capability formation. *Research policy*, 31(8):1417–1429, 2002.

John R Graham. Proxies for the corporate marginal tax rate. *Journal of Financial Economics*, 42(2):187–221, 1996.

James R Hasselback. An empirical examination of annual report presentation of the corporate income tax expense. *The Accounting Review*, 51(2):269–276, 1976.

Sam A Hicks. Choosing the form for business tax incentives. *Accounting Review*, pages 708–716, 1978.

Karl Janecek. What is a realistic aversion to risk for real-world individual investors. *Available online a t: http://www. sba21. com/personal/RiskAversion. pdf*, 2004.

Bowon Kim. Dynamic outsourcing to contract manufacturers with different capabilities of reducing the supply cost. *International Journal of Production Economics*, 86(1):63–80, 2003.

Isaac Leung. Electronic component sourcing: evolution and strategies. 2013. URL https://electronicsnews.com.au/electronic-component-sourcing-evolution-and-strategies/.

Aparna Mathur. Why 70% of companies paid zero in corporate taxes: They had zero profits, April 2016.

Rajnish Mehra and Edward C Prescott. The equity premium: A puzzle. *Journal of monetary Economics*, 15(2):145–161, 1985.

Gregory S Miller and Douglas J Skinner. Determinants of the valuation allowance for deferred tax assets under sfas no. 109. *Accounting Review*, pages 213–233, 1998.

Sebastian Moritz and Richard Pibernik. The optimal number of suppliers in the presence of volume discounts and different compensation potentials-an analytical and numerical analysis. 2008.

Charles L Munson and Meir J Rosenblatt. Theories and realities of quantity discounts: An exploratory study. *Production and operations management*, 7(4):352–369, 1998.

Renesas Press Release. Impact of the december 28 earthquake in north part of ibaraki on renesas electronics operation, December 2016.

Michael Quayle. Industrial procurement: factors affecting sourcing decisions. *European Journal of Purchasing & Supply Management*, 4(4):199–205, 1998.

Shelley C Rhoades-Catanach. Discussion of valuation of a firm with a tax loss carryover. *Journal of The American Taxation Association*, 25(s-1):83–86, 2003.

Sudipto Sarkar. Valuation of tax loss carryforwards. In *CAAA Annual Conference*, 2012.

Catherine M Schrand and MH Wong. Earnings management using the valuation allowance for deferred tax assets under sfas no. 109*. *Contemporary Accounting Research*, 20(3):579–611, 2003.

Felix Streitferdt. The valuation of tax loss carryforwards. 2010.

Srinivas Talluri and Ram Narasimhan. A methodology for strategic sourcing. *European journal of operational research*, 154(1):236–250, 2004.

Patrick Van den Bossche, Rajeev Prabhakar, Kevin Phillippi, and Joe Blount. The road to contract manufacturing success. 2014. URL https://www.atkearney.com/documents/10192/4531776/SCMR.Contract.Mfging.pdf/daa21120-1a9f-4b5a-a27a-0867a4071444.

Anja De Waegenaere, Richard C Sansing, and Jacco L Wielhouwer. Valuation of a firm with a tax loss carryover. *Journal of the American Taxation Association*, 25 (s-1):65–82, 2003.

James A Welch and P Ranganath Nayak. Strategic sourcing: a progressive approach to the make-or-buy decision. *The Executive*, 6(1):23–31, 1992.

Harry I Wolk and Michael G Tearney. Income tax allocation and loss carry-forwards: Exploring uncharted ground. *The Accounting Review*, 48(2):292–299, 1973.

WSJ. Gm's tax shelter, July 2009.

Tao Zeng. The valuation of loss carryforwards. *Canadian Journal of Administrative Sciences/Revue Canadienne des Sciences de l'Administration*, 20(2):166–176, 2003.

APPENDIX A

Appendix-Simulations and Figures

Tax Loss Carryforwards, Billions of USD											
Industry	SIC Code	**2003**	**2004**	**2005**	**2006**	**2007**	**2008**	**2009**	**2010**	**2011**	**2012**
Agriculture, Forestry and Fishing	0100-0999	2.47	2.71	4.44	4.38	4.09	3.72	4.05	4.21	4.57	4.71
Mining	1000-1499	22.96	28.59	32.00	29.30	34.41	39.65	64.27	80.15	100.91	118.35
Construction	1500-1799	7.29	1.89	2.37	1.97	3.28	5.67	7.60	6.70	10.16	11.15
Manufacturing	2000-3999	339.63	384.40	967.71	453.93	477.06	542.89	633.38	664.95	677.24	700.65
Transportation, Communications, Electric, Gas and Sanitary Services	4000-4999	222.19	241.10	224.43	210.38	205.06	209.73	399.56	402.26	389.06	264.09
Wholesale Trade	5000-5199	9.69	9.16	11.89	14.89	14.04	11.05	11.21	13.80	18.28	9.34
Retail Trade	5200-5999	30.62	30.83	29.84	26.91	24.13	23.28	21.89	25.61	19.71	17.51
Finance, Insurance and Real Estate	6000-6799	71.58	86.39	81.55	59.46	148.19	289.51	200.79	277.10	300.81	270.06
Services	7000-8999	87.36	93.16	90.23	101.05	101.98	103.45	105.94	112.88	123.94	127.96
Public Administration	9100-onwards	17.89	19.01	20.01	19.67	15.62	22.01	21.93	15.94	22.39	22.87

Table A.1: Table showing the TLCFs (in $) for North America between 2003-2012

```
function[V0,Action,Act]=TLCFTaxes(T,x0,y0,C0,tax,b,r1,prob,L,alpha)

%Modified: June 27, 2016
%Created by: Nilofar Varzgani
%PhD Candidate, MS
%Rutgers Business School
%
%Valuation of Tax Loss Carryforwards (TLCFs)
%This function returns the value of a given set of TLCF (called C1) and
%action to be taken regarding utilization of that set assuming uncertain
%income and multiple tax brackets
%
%Assume probabilities of profit and loss are constant.
%Assume tax rates and rate of return remain constant.
%
%Inputs
%T       finite time horizon
%x0      initial wealth
%y0      initial profit or loss. If loss, this number has to be negative.
%C0      initial set of TLCFs given at time 0
%tax     array of tax rates, dim m X T (m being the highest tax bracket in rows)
%b       array of upper income level in each tax bracket
%r1      array of rate of return (profit or loss).
%prob    probability of profit
%L       upper limit on the use of TLCFs given by section 382
%alpha   discount factor
%
%Outputs
%
%V0          Optimal value of the set of TLCFs end of period of T cycles
%A           amount of TLCF to be used.
%---------------------------------------------------------%
%Generate arrays to store values in
N=size(C0,2);             %maximum number of TLCFs.
m=size(tax,1);            %maximum tax brackets.
state=zeros(1,T);         %array to store profit/loss.
wealth=zeros(1,T+1);      %array to store aftertax wealth; col 1 is time 0.
prop=ones(N,T);           %array to store proportion of each TLCF used.
                          %rows belong to the life of TLCF, N being the
                          %newest. Cols are the time period
C=zeros(N,T+1);           %array to store TLCFs to carryforward;col 1 is time 0.
Cbar=zeros(1,T+1);        %array to store the sum of TLCF available for use.
taxpayb=zeros(1,T);       %benchmark tax payable.
%taxpay=zeros(1,T);       %taxpayable after action.(uncomment if required)
Saving=zeros(1,T);        %savings at each time period.
Action=zeros(1,T);        %actions for each time period (selected by random
                          %function).
Act=zeros(N,T);           %action vector elements for each element of C.
V_a=zeros(1,T);           %Array to store value at every time period.
```

Figure A.1: Simulations for Risk Neutral Investor (Examples in Chapter 2.6)

```
%Store input values into the grids

wealth(1)=x0;
C(:,1)=C0(:);
cc=0;
    for i=1:N
        cc=cc+C0(i);
        Cbar(1)=cc;
    end

%Action at time t=0;

%%Case 1: Starting at Profit state
    if (y0>=0)
        %%%%Divide profit into tax brackets
        profitbreak=zeros(1,m);

        bcounter=m;
        while bcounter > 0
            if (y0>b(bcounter))
                profitbreak(bcounter)=y0-b(bcounter);
                for bb=1:bcounter-1
                    profitbreak(bb)=b(bb+1)-b(bb);
                end
                break
            else profitbreak(bcounter)=0;
                bcounter=bcounter-1;
            end
        end

%%Action set
        amin0=0;
        %Generate array to store max values for each time period.
        atemp_0=[L,Cbar(1),y0];
        amax_0=max(min(atemp_0),0);

        %Generate the action grid
        inc=0.01; %set a value for the increment for A_0
                  %0.01 is chosen to represent it down to the penny.
        nr0=((amax_0-amin0)/inc)+1;
        nn0=round(nr0);  %to make sure nn0 is an integar.
        A_0=linspace(amin0,amax_0,nn0);

%%Application of TLCFs
%%Grids to store f and g for each action
    f_0temp=zeros(nn0,m); %rows for each action and col for each tax bracket
    g_0temp=zeros(nn0,m);

    for k=1:m
        f_0temp(1,k)=profitbreak(1,k);
```

```
    end

    g_0temp(1,m)=amax_0;

%%Find the taxable wealth after action in the highest tax bracket.
        for a=2:nn0
            f_0temp(a,m)=max(profitbreak(m)-A_0(a),0);
            g_0temp(a,m)=max(A_0(a)-profitbreak(m),0);
        end

        for k=m-1:-1:1
            for a=2:nn0
                f_0temp(a,k)=max(profitbreak(k)-g_0temp(a,k+1),0);
                g_0temp(a,k)=max(g_0temp(a,k+1)-profitbreak(k),0);
            end
        end

%%Calculate benchmark tax payables
    tax0=tax(:,1);
    taxpayb_0=profitbreak*tax0;
    taxpayb(1)=taxpayb_0;

    taxpay_0=f_0temp*tax0;

%%Calculate savings for each action value
    save_0=zeros(nn0,1);
    for a=1:nn0
        save_0(a,1)=taxpayb_0-taxpay_0(a,1);
    end

%%Generate a random number to pick an index value for action array.

    joe_0=randi([1,nn0],1,1);  %pick an a0ction randomly at each iteration
    Action(1)=A_0(joe_0);
    Saving(1,1)=save_0(joe_0);

%%Calculate new wealth for each action value
    wealth(2)=wealth(1)+y0-taxpay_0(joe_0);

%Calculate set of TLCF to be carried forward to the next time period

%%generate an array for proportions. Assuming FIFO here.
    prop_0=prop(:,1);
    a0=Action(1);
    for i=1:N
        y=prop_0(i)*C(i,1);
        if (y==a0)
            prop_0(i)=1;
            for ii=i+1:N
                prop_0(ii)=0;
```

```
            end
            break
        elseif (y>a0);
            prop_0(i)=a0/C(i,1);
            for j=i+1:N
                prop_0(j)=0;
            end
            break
        else
            yy=a0-y;
            prop_0(i)=1;
            a0=yy;
        end
    end

    %update proportions array.
    prop(:,1)=prop_0;

        for i=1:N-1
            C(i,2)=C(i+1,1)*(1-prop_0(i+1));
        end
            C(N,2)=0;  %no new TLCF is generated when state=profit

    %Update Cbar
    tcf=0;
    for i=1:N
        tcf=tcf+C(i,2);
        Cbar(2)=tcf;
    end

    %Action taken at time t=0, is stored in Act(:,1)
    for n=1:N
        Act(n,1)=C(n,1)*prop_0(n);
    end

%%Case 2: Starting at Loss state
    else

        %Calculate new wealth since there is no action to take
        wealth(2)=wealth(1)+y0;

        %Calculate set of TLCF to be carried forward to the next time period
        for i=2:N-1
            C(i,2)=C(i+1,2);
        end
        C(N,2)=y0; %new TLCF generated from loss.

        Action(1)=0;
        Saving(1)=0;
        Act(:,1)=0;
```

```
        %Update Cbar
        tcf=0;
        for i=1:N
            tcf=tcf+C(i,2);
            Cbar(2)=tcf;
        end

        %Update proportions array
        prop(:,1)=0;
    end

%%%Now going forward in time after action at t=0 has been decided.

for t=2:T

%Calculate profit/loss (Here t=2 represents time 1 and so on)

profit_1=wealth(t)*r1(1); %change this when changing the structure of r
loss_1=wealth(t)*r1(2);

%simulate random outcome of investment
    sim=rand();
    if (sim<=prob)
        state(t)=profit_1;
    else state(t)=loss_1;
    end

%%%%Profit state
if (sim<=prob)
    %%%%Divide profit into tax brackets
    profitbreak_1=zeros(1,m);

    bcounter_1=m;
    while bcounter_1 > 0
        if (profit_1>b(bcounter_1))
            profitbreak_1(bcounter_1)=profit_1-b(bcounter_1);
            for bbb=1:bcounter_1-1
                profitbreak(bbb)=b(bbb+1)-b(bbb);
            end
            break
        else profitbreak_1(bcounter_1)=0;
            bcounter_1=bcounter_1-1;
        end
    end

    %Action set
    amin=0;
    %Generate array to store max values for each time period.
    atemp_1=[L,Cbar(t),profit_1];
```

```
amax_1=max(min(atemp_1),0);

%Generate the action grid
inc=0.01; %set a value for the increment for A_1
          %0,01 is chosen to represent it down to the penny.
nr=((amax_1-amin)/inc)+1;
nn=round(nr);  %to make sure nn is an integar.
A_1=linspace(amin,amax_1,nn);

%Application of TLCFs
%%Grids to store f and g for each action
f_1temp=zeros(nn,m); %rows for each action and col for each tax bracket
g_1temp=zeros(nn,m);

for k=1:m
    f_1temp(1,k)=profitbreak_1(1,k);
end

g_1temp(1,m)=amax_1;

%%Find the taxable wealth after action in the highest tax bracket.
for a=2:nn
    f_1temp(a,m)=max(profitbreak_1(m)-A_1(a),0);
    g_1temp(a,m)=max(A_1(a)-profitbreak_1(m),0);
end


for k=m-1:-1:1
    for a=2:nn
        f_1temp(a,k)=max(profitbreak_1(k)-g_1temp(a,k+1),0);
        g_1temp(a,k)=max(g_1temp(a,k+1)-profitbreak_1(k),0);
    end
end

%%Calculate benchmark tax payables
tax1=tax(:,t);
taxpayb_1=profitbreak_1*tax1;
taxpayb(t)=taxpayb_1;

taxpay_1=f_1temp*tax1;

%%Calculate savings for each action value
save_1=zeros(nn,1);
for a=1:nn
    save_1(a,1)=taxpayb_1-taxpay_1(a,1);
end

%Generate a random number to pick an index value for action array.
if (t==T)
    Saving(1,t)=max(save_1); %to find max savings and action
```

```
        tempmax=max(save_1);
        for l=1:nn
            if save_1(l)==tempmax
                ag=l;
            end
        end
        Action(t)=A_1(ag);
        wealth(t+1)=wealth(t)+profit_1-taxpay_1(ag);

    else
    joe=randi([1,nn],1,1);  %pick an action randomly at each iteration
    Action(t)=A_1(joe);
    Saving(1,t)=save_1(joe);

    %Calculate new wealth for each action value
    wealth(t+1)=wealth(t)+profit_1-taxpay_1(joe);
    end

    %Calculate set of TLCF to be carried forward to the next time period
        %%generate an array for proportions. Assuming FIFO here.
        prop_1=prop(:,t);
        a=Action(t);
        for i=1:N
            y=prop_1(i)*C(i,t);
            if (y==a)
                prop_1(i)=1;
                for ii=i+1:N
                    prop_1(ii)=0;
                end
                break
            elseif (y>a);
                prop_1(i)=a/C(i,t);
                for j=i+1:N
                    prop_1(j)=0;
                end
                break
            else
                yy=a-y;
                prop_1(i)=1;
                a=yy;
            end
        end
        prop(:,t)=prop_1;


    for i=1:N-1
        C(i,t+1)=C(i+1,t)*(1-prop_1(i+1));
    end
    C(N,t+1)=0;  %no new TLCF is generated when state=profit
```

```
    %Update Cbar
    tc=0;
    for i=1:N
        tc=tc+C(i,t+1);
        Cbar(t+1)=tc;
    end

    %Store the action at time t into the Act array for each element of C
    for n=1:N
        Act(n,t)=prop_1(n)*C(n,t);
    end

    %%%%Loss state
    else

        %Calculate new wealth since there is no action to take
        wealth(t+1)=wealth(t)+loss_1;

        %Calculate set of TLCF to be carried forward to the next time
        %period
        for i=2:N-1
            C(i,t+1)=C(i+1,t);
        end
        C(N,t+1)=loss_1; %new TLCF generated from loss.

        Action(t)=0;
        Saving(t)=0;
        Act(:,t)=0;

        %Update Cbar
        tc=0;
        for i=1:N
            tc=tc+C(i,t+1);
            Cbar(t+1)=tc;
        end

end

end

%Calculate value function
V_a(1,T)=Saving(1,T);

for t=T-1:-1:1
    V_a(1,t)=Saving(1,t)+(alpha*V_a(1,t+1)); %going backwards in time
end

V0=V_a(1);
```

Published with MATLAB® R2016a

```
%This code will give input values to the function TLCFTaxes.m previously generated
%and produce iterations for optimal results.
%--------------------------------------------------------%
%%Input values (can be changed)
T=3;
x0=500;
C0=[100,75,150];
tax=[0.2,0.2,0.2;0.35,0.35,0.35];
y0=150;
b=[0,200];
r1=[0.2,-.10];
prob=0.5;
L=500;
alpha=0.95;
N=size(C0,2);

K=1000;                          %maximum iterations
A= zeros(1,T);                   %action vector
V1=0;
A1=zeros(1,T);
AA=zeros(N,T);
%Simulate this function J times to compare results.
%J=100;
Vstore=zeros(1,1);
Astore=zeros(1,T);
AA_store=zeros(N,T);
%for j=1:J

    for i=1:K
        [V0, A, Act]=TLCFTaxes(T,x0,y0,C0,tax,b,r1,prob,L,alpha);
        if(V1>V0)
```

Figure A.2: Simulations for Risk Neutral Investor-Code to Call the Function

```
            V0=V1;
            A=A1;
            Act=AA;
        else
            V1=V0;
            A1=A;
            AA=Act;
        end
        V1=V0;
        A1=A;
        AA=Act;
    end
    Vstore(1,1)= V0;
    Astore(1,:)= A1;
    AA_store=AA;
%end
disp(Vstore)
disp(Astore)
disp(AA_store)
```

```
   62.6572

  149.7600  113.0903   62.1497



  100.0000   25.2400   62.1497

   49.7600   87.8503         0

         0         0         0
```

Published with MATLAB® R2016a

```
function[V0,Action,Act]=TLCFTaxes_power_ut(T,x0,y0,C0,tax,b,r1,prob,L,alpha,nu)

%Modified: June 27, 2016
%Created by: Nilofar Varzgani
%PhD Candidate, MS
%Rutgers Business School
%
%Valuation of Tax Loss Carryforwards (TLCFs)
%This function returns the value of a given set of TLCF (called C1) and
%action to be taken regarding utilization of that set assuming uncertain
%income and multiple tax brackets using utility functions.
%
%The max number of tax brackets that can be used here is 3.
%
%Assume probabilities of profit and loss are constant.
%Assume tax rates and rate of return remain constant.
%
%Inputs
%T      finite time horizon
%x0     initial wealth
%y0     initial profit or loss. If loss, this number has to be negative.
%C0     initial set of TLCFs given at time 0
%tax    array of tax rates, dim m X T (m being the highest tax bracket in rows)
%b      array of upper income level in each tax bracket
%r1     array of rate of return (profit or loss).
%prob   probability of profit
%L      upper limit on the use of TLCFs given by section 382
%alpha  discount factor
%nu     the constant for the power utility function.(include if changing
%power utility function)
%
%Outputs
%V0         Optimal value of the set of TLCFs end of period of T cycles
%A          amount of TLCF to be used.
%-------------------------------------------------------%
%Generate arrays to store values in
N=size(C0,2);           %maximum number of TLCFs.
m=size(tax,1);          %maximum tax brackets.
state=zeros(1,T);       %array to store profit/loss.
wealth=zeros(1,T+1);    %array to store aftertax wealth; col 1 is time 0.
prop=ones(N,T);         %array to store proportion of each TLCF used.
                        %rows belong to the life of TLCF, N being the
                        %newest. Cols are the time period
C=zeros(N,T+1);         %array to store TLCFs to caryforward; col1 is time 0.
Cbar=zeros(1,T+1);      %array to store the sum of TLCF available for use.
taxpayb=zeros(1,T);     %benchmark tax payable.
%taxpay=zeros(1,T);     %tax payable after action.(uncomment if required)
Saving=zeros(1,T);      %savings at each time period.
Utility=zeros(1,T);     %utility of savings in each time period.
Action=zeros(1,T);      %actions for each time period (selected by random
```

Figure A.3: Simulations for Risk Aversel Investor (Examples in Chapter 2.7)

```
                        %function).
Act=zeros(N,T);         %action vector elements for each element of C.
V_a=zeros(1,T);         %Array to store value at every time period.

%Store input values into the grids

wealth(1)=x0;
C(:,1)=C0(:);
cc=0;
    for i=1:N
        cc=cc+C0(i);
        Cbar(1)=cc;
    end

%Action at time t=0;
%%%%Case 1: Starting at Profit state
    if (y0>=0)
        %%%%Divide profit into tax brackets
        profitbreak=zeros(1,m);

        bcounter=m;
        while bcounter > 0
            if (y0>b(bcounter))
                profitbreak(bcounter)=y0-b(bcounter);
                for bb=1:bcounter-1
                    profitbreak(bb)=b(bb+1)-b(bb);
                end
                break
            else profitbreak(bcounter)=0;
                bcounter=bcounter-1;
            end
        end

%Action set
        amin0=0;
        %generate array to store max values for each time period.
        atemp_0=[L,Cbar(1),y0];
        amax_0=max(min(atemp_0),0);

        %generate the action grid
        inc=0.01; %set a value for the increment for A_0
                  %0.01 is chosen to represent it down to the penny.
        nr0=((amax_0-amin0)/inc)+1;
        nn0=round(nr0);  %to make sure nn0 is an integer.
        A_0=linspace(amin0,amax_0,nn0);

%%Application of TLCFs
%%Grids to store f and g for each action
    f_0temp=zeros(nn0,m); %rows for each action and col for each tax bracket
    g_0temp=zeros(nn0,m);
```

```
    for k=1:m
        f_0temp(1,k)=profitbreak(1,k);
    end
    g_0temp(1,m)=amax_0;

%%Find the taxable wealth after action in the highest tax bracket.
    for a=2:nn0
        f_0temp(a,m)=max(profitbreak(m)-A_0(a),0);
        g_0temp(a,m)=max(A_0(a)-profitbreak(m),0);
    end

    for k=m-1:-1:1
        for a=2:nn0
            f_0temp(a,k)=max(profitbreak(k)-g_0temp(a,k+1),0);
            g_0temp(a,k)=max(g_0temp(a,k+1)-profitbreak(k),0);
        end
    end

%%Calculate benchmark tax payables
    tax0=tax(:,1);
    taxpayb_0=profitbreak*tax0;
    taxpayb(1)=taxpayb_0;
    taxpay_0=f_0temp*tax0;

%%Calculate savings and utility of savings for each action value
    save_0=zeros(nn0,1);
    util_save=zeros(nn0,1);
    for a=1:nn0
        save_0(a,1)=taxpayb_0-taxpay_0(a,1);
        util_save(a,1)=((save_0(a,1)^(1-nu))-1)/(1-nu); %change this to change the
utility function
    end

%Generate a random number to pick an index value for action array.

    joe_0=randi([1,nn0],1,1);  %pick an a0ction randomly at each iteration
    Action(1)=A_0(joe_0);
    Saving(1,1)=save_0(joe_0);
    Utility(1,1)=util_save(joe_0);

%%Calculate new wealth for each action value
    wealth(2)=wealth(1)+y0-taxpay_0(joe_0);

%%Calculate set of TLCF to be carried forward to the next time period
    %%Generate an array for proportions. Assuming FIFO here.
    prop_0=prop(:,1);
    a0=Action(1);
    for i=1:N
        y=prop_0(i)*C(i,1);
```

```
        if (y==a0)
            prop_0(i)=1;
            for ii=i+1:N
                prop_0(ii)=0;
            end
            break
        elseif (y>a0);
            prop_0(i)=a0/C(i,1);
            for j=i+1:N
                prop_0(j)=0;
            end
            break
            else
                yy=a0-y;
                prop_0(i)=1;
                a0=yy;
        end
    end
%Update proportions array.
    prop(:,1)=prop_0;
```

```
    for i=1:N-1
        C(i,2)=C(i+1,1)*(1-prop_0(i+1));
    end
    C(N,2)=0;  %no new TLCF is generated when state=profit

%Update Cbar
    tcf=0;
    for i=1:N
    tcf=tcf+C(i,2);
    Cbar(2)=tcf;
    end

%Action taken at time t=0, is stored in Act(:,1)
    for n=1:N
        Act(n,1)=C(n,1)*prop_0(n);
    end

%%%%Case 2: Starting at Loss state
    else
```

```
        %Calculate new wealth since there is no action to take
        wealth(2)=wealth(1)+y0;
        %Calculate set of TLCF to be carried forward to the next time period
```

```
        for i=2:N-1
            C(i,2)=C(i+1,2);
        end
        C(N,2)=y0; %new TLCF generated from loss.

        Action(1)=0;
```

```
        Saving(1)=0;
        Utility(1)=0;
        Act(:,1)=0;

    %Update Cbar
    tcf=0;
    for i=1:N
        tcf=tcf+C(i,2);
        Cbar(2)=tcf;
    end

    %Update proportions array
    prop(:,1)=0;
    end

%%%Now going forward in time after action at t=0 has been decided.

for t=2:T

    %Calculate profit/loss (Here t=2 represents time 1 and so on)

    profit_1=wealth(t)*r1(1); %change this when changing the structure of r
    loss_1=wealth(t)*r1(2);

    %simulate random outcome of investment
    sim=rand();
    if (sim<=prob)
        state(t)=profit_1;
    else state(t)=loss_1;
    end

    %%%%Profit state
    if (sim<=prob)
    %%%%Divide profit into tax brackets
    profitbreak_1=zeros(1,m);

    bcounter_1=m;
    while bcounter_1 > 0
        if (profit_1>b(bcounter_1))
            profitbreak_1(bcounter_1)=profit_1-b(bcounter_1);
            for bbb=1:bcounter_1-1
                profitbreak(bbb)=b(bbb+1)-b(bbb);
            end
            break
        else profitbreak_1(bcounter_1)=0;
            bcounter_1=bcounter_1-1;
        end
    end

    %Action set
```

```
    amin=0;
    %generate array to store max values for each time period.
    atemp_1=[L,Cbar(t),profit_1];
    amax_1=max(min(atemp_1),0);

    %generate the action grid
    inc=0.01; %set a value for the increment for A_1
              %0,01 is chosen to represent it down to the penny.
    nr=((amax_1-amin)/inc)+1;
    nn=round(nr);  %to make sure nn is an integar.
    A_1=linspace(amin,amax_1,nn);

    %Application of TLCFs
    %%grids to store f and g for each action
        f_1temp=zeros(nn,m); %rows for each action and col for each tax bracket
        g_1temp=zeros(nn,m);

        for k=1:m
            f_1temp(1,k)=profitbreak_1(1,k);
        end

        g_1temp(1,m)=amax_1;

    %%find the taxable wealth after action in the highest tax bracket.
    for a=2:nn
        f_1temp(a,m)=max(profitbreak_1(m)-A_1(a),0);
        g_1temp(a,m)=max(A_1(a)-profitbreak_1(m),0);
    end

    for k=m-1:-1:1
        for a=2:nn
            f_1temp(a,k)=max(profitbreak_1(k)-g_1temp(a,k+1),0);
            g_1temp(a,k)=max(g_1temp(a,k+1)-profitbreak_1(k),0);
        end
    end

    %%calculate benchmark tax payables
    tax1=tax(:,t);
    taxpayb_1=profitbreak_1*tax1;
    taxpayb(t)=taxpayb_1;
    taxpay_1=f_1temp*tax1;

    %%calculate savings for each action value
    save_1=zeros(nn,1);
    util_1=zeros(nn,1);
    for a=1:nn
        save_1(a,1)=taxpayb_1-taxpay_1(a,1);
        util_1(a,1)=((save_1(a,1)^(1-nu))-1)/(1-nu); %change this to change the
utility function
    end
```

```
%generate a random number to pick an index value for action array.
if (t==T)
    Saving(1,t)=max(save_1); %to find max savings and action
    Utility(1,t)=max(util_1); %to find max utility of savings and action
    tempmax=max(util_1);
    for l=1:nn
        if util_1(l)==tempmax
            ag=l;
        end
    end
    Action(t)=A_1(ag);
    wealth(t+1)=wealth(t)+profit_1-taxpay_1(ag);
else
joe=randi([1,nn],1,1);  %pick an action randomly at each iteration
Action(t)=A_1(joe);
Saving(1,t)=save_1(joe);
Utility(1,t)=util_1(joe);

%Calculate new wealth for each action value
wealth(t+1)=wealth(t)+profit_1-taxpay_1(joe);
end

%Calculate set of TLCF to be carried forward to the next time period
    %%generate an array for proportions. Assuming FIFO here.
    prop_1=prop(:,t);
    a=Action(t);
    for i=1:N
        y=prop_1(i)*C(i,t);
        if (y==a)
            prop_1(i)=1;
            for ii=i+1:N
                prop_1(ii)=0;
            end
                break
        elseif (y>a);
            prop_1(i)=a/C(i,t);
            for j=i+1:N
                prop_1(j)=0;
            end
            break
        else
            yy=a-y;
            prop_1(i)=1;
            a=yy;
        end
    end
    prop(:,t)=prop_1;

    for i=1:N-1
```

```
            C(i,t+1)=C(i+1,t)*(1-prop_1(i+1));
        end
        C(N,t+1)=0;  %no new TLCF is generated when state=profit

        %Update Cbar
        tc=0;
        for i=1:N
            tc=tc+C(i,t+1);
            Cbar(t+1)=tc;
        end

        %Store the action at time t into the Act array for each element of C
        for n=1:N
            Act(n,t)=prop_1(n)*C(n,t);
        end

    %%%%Loss state
    else

        %Calculate new wealth since there is no action to take
        wealth(t+1)=wealth(t)+loss_1;

        %Calculate set of TLCF to be carried forward to the next time period
        for i=2:N-1
            C(i,t+1)=C(i+1,t);
        end
        C(N,t+1)=loss_1; %new TLCF generated from loss.
        Action(t)=0;
        Saving(t)=0;
        Utility(t)=0;
        Act(:,t)=0;

        %Update Cbar
        tc=0;
        for i=1:N
            tc=tc+C(i,t+1);
            Cbar(t+1)=tc;
        end
    end
end

%Calculate value function
V_a(1,T)=Utility(1,T);

for t=T-1:-1:1
    V_a(1,t)=Utility(1,t)+(alpha*V_a(1,t+1)); %going backwards in time
end

V0=V_a(1);
```

Published with MATLAB® R2016a

```
%Created by Nilofar Varzgani
%PhD Candidate, MS
%Rutgers Business School
%Modified: June 27, 2016
%
%This code will give input values to the function TLCF previously generated
%and produce iterations for optimal results with utility functions.
%----------------------------------------------------------%
%%Input values (can be changed)

T=2;
x0=500;
C0=[100,75,150];
tax=[0.2,0.2;0.35,0.35];
y0=150;
b=[0,200];
r1=[0.2,-.10];
prob=1;
L=500;
alpha=0.95;
N=size(C0,2);
nu=2;

K=1000;                              %maximum iterations
A= zeros(1,T);                       %action vector
V1=0;
A1=zeros(1,T);
AA=zeros(N,T);

%Simulate this function J times to compare results.
%J=100;
Vstore=zeros(1,1);
Astore=zeros(1,T);
AA_store=zeros(N,T);

%for j=1:J

    for i=1:K
        [V0, A, Act]=TLCFTaxes_power_ut(T,x0,y0,C0,tax,b,r1,prob,L,alpha, nu);
        if(V1>V0)
            V0=V1;
            A=A1;
            Act=AA;

        else
            V1=V0;
            A1=A;
            AA=Act;
        end
        V1=V0;
```

Figure A.4: Simulations for Risk Averse Investor-Code to Call the Function

```
        A1=A;
        AA=Act;

    end

    Vstore(1,1)= V0;
    Astore(1,:)= A1;
    AA_store=AA;
%end
disp(Vstore)
disp(Astore)
disp(AA_store)
```

```
    1.8801



  149.9600  129.9984



  100.0000   25.0400

   49.9600  104.9584

         0         0
```

Published with MATLAB® R2016a

```
"""
author 1:- Gaurav Gawade
author 2:- Nilofar Varzgani
"""
from xlrd import open_workbook
import pandas as panda


def read(file_name, numbers_products, number_of_customers):
    """
    This function reads the file data.xls using pandas library which is stored
in the dataframe named excel_data. Since the data.xls contains sheet which
have specific names the names are generated using for loop based on number of
products input. Also, a call is made to convert_to_dictionary function which
then further processes the data.

    :param file_name: Name of the excel file to be processed.
    :param numbers_products: Number of products given as a input.
    :param number_of_customers: Number of customers given as a input.
    :return: None.
    """
    excel_data = panda.ExcelFile(file_name)

    sheet_names = excel_data.sheet_names
    list_names = []
    string_p = "P"
    string_cm = "CM"
    for index in range(1, numbers_products + 1):
        result = ""
        result = string_p + str(index)
        list_names.append(result)
        result = ""
    convert_to_dictionary(sheet_names, excel_data, list_names,
number_of_customers)


def convert_to_dictionary(sheet_names, excel_data, list_names,
number_of_customers):
    """
    The data obtained in the excel_data dataframe is read using loop wherein
each sheet is
    converted to dictionary and stored in another dictionary named
'global_dictionary'. Also,
    a call is made to calculate_price which performs calculation for each
products.
    :param sheet_names: List of names of sheets.
    :param excel_data: The dataframe which contains the data extracted from
data.xls file.
    :param list_names: Contains the name of the sheets which is generted in
'read' function from
                        line number 23 to 27.
    :param number_of_customers: Number of customers given as a input.
    :return: None.
```

Figure A.5: Simulations for Electronic Component Sourcing Case Study

```
    """

    global_dictionary = {}
    cm = "CM"
    customer = {}
    product = {}
    for sheet in sheet_names:
        dataframe = excel_data.parse(sheet)
        global_dictionary[sheet] = dataframe.to_dict()

    calculate_price(global_dictionary, list_names, number_of_customers)


def calculate_price(global_dictionary, list_names, number_of_customers):
    """
    Based on the number of customers the price for each product is calculated
in this function also, the call
    is made to discounts function.
    Note: Significant lines are explained for better understanding the
functionality of this function.
    :param global_dictionary: The dictionary which contains the entire data
wherein the key is sheet name and
    value is another dictiornary which contains actual data.
    :param list_names: Contains the name of the sheets which is generted in
'read' function from
                        line number 23 to 27.
    :param number_of_customers: Number of customers given as a input.
    :return: None.
    """
    cm = "CM"
    mva = "MVA"
    product_index = ""
    customer = {}
    sum_cost = 0
    for index in range(1, number_of_customers+1):
        for name in list_names:
            #The below line calculates the total cost bid by each customer for
each product.
            #for eg- data from CM1-P1, CM1-P2, CM2-P1 and so on...
            # and the data is stored in customer dictionary with key being
customer and value being total cost.
            sum_cost = sum(global_dictionary.get(cm + str(index) + "-"
+str(name)).get("Total Price ").values())

            #The below for loop calculates the total MVA for all the products
by each customer also
            #the data is stored in customer dictionary with key being customer
and value being total cost.
            for key, val  in global_dictionary.get(cm + str(index) + "-"
+mva).get("Product").items():
                if val == name:
                    sum_cost = sum_cost + global_dictionary.get(cm +
str(index) + "-" +mva).get("Price").get(key)
                if customer.get(cm + str(index)) is not None:
```

```
                    customer[cm + str(index)] = customer.get(cm + str(index))
+ sum_cost
                else:
                    customer[cm + str(index)] = sum_cost
                sum_cost = 0
    discounts(customer, global_dictionary, list_names, number_of_customers)

def discounts(customer, global_dictionary, list_names, number_of_customers):
    """
    This function calculates the discounts for each product performing some
basic calculations. Also, a call
    is made to calculate_baseline function.
     Note: Significant lines are explained for better understanding the
functionality of this function.
    :param customer: This is the dicitionary which contains the total cost for
all products bid by each customer.
    :param global_dictionary: The dictionary which contains the entire data
wherein the key is sheet name and
    value is another dictornary which contains actual data.
    :param list_names:  Contains the name of the sheets which is generted in
'read' function from
                        line number 23 to 27.
    :param number_of_customers: Number of customers given as a input.
    :return:
    """

    cm_bids = {}
    baseline = {}
    discounts = global_dictionary.get("Discount")
    final_cost = 0
    for key, value in
global_dictionary.get("Discount").get("Manufacturer").items():
        if customer.get(value) is not None:
            #This loop is used to calculate discount based on the volume for
each product and are stored in
            #cm_bids dictionary where key is CM1_V1(Volume by each customer)
and value is volume calculated.
            cm_bids[value + "V1"] = customer.get(value) -
(global_dictionary.get("Discount").get("V1").get(0) * customer.get(value))
            cm_bids[value + "V2"] = customer.get(value) - (
            global_dictionary.get("Discount").get("V2").get(0) *
customer.get(value))
            cm_bids[value + "V3"] = customer.get(value) - (
            global_dictionary.get("Discount").get("V3").get(0) *
customer.get(value))

    calculate_baseline(cm_bids, global_dictionary, list_names,
number_of_customers)


def calculate_baseline(cm_bids, global_dictionary, list_names,
number_of_customers):
    """
    This function calculates the baseline for each product performing some
```

```
basic calculations and the
    customer which has the lowest bid is dsipalyed.
    is made to calculate_baseline function.
    Note: Significant lines are explained for better understanding the
functionality of this function.
    :param cm_bids:
    :param global_dictionary:
    :param list_names:
    :param number_of_customers:
    :return:
    """
    baseline_list = []
    bl = "BL"
    mva = "MVA"
    sum_cost = 0
    index = 1
    for index in range(1, len(list_names) + 2):
        string = bl + "-"+"P"+str(index)
        if index == len(list_names) + 1:
            string = bl + "-" + mva
            #The total cost for each product is calculated below from baseline
data.
            sum_cost +=
(sum(global_dictionary.get(string).get("Price").values()))
        else:
            sum_cost += sum(global_dictionary.get(string).get("Total Price
").values())

    for key, value in cm_bids.items():
        cm_bids[key] = sum_cost - cm_bids.get(key)

    print(cm_bids)
    print("Preferable CM would be ", min(cm_bids, key=cm_bids.get))


def main():
    """
    This is the main function which makes call to several other functions.
    :return:
    """
    file_name = 'data.xlsx'
    numbers_products = int(input("Enter number of products"))
    number_of_customers = int(input("Enter number of customers"))
    read(file_name,numbers_products,number_of_customers)


if __name__ == '__main__':
    main()
```