Energy-economic model for transition to renewables

by

John S. Mizanski

A thesis submitted in conformity with the requirements
for the degree of Masters of Modelling and Computational Science
Faculty of Graduate Studies (Science)
University of Ontario Institute of Technology

Supervisor(s): Dr. P. Berg

Abstract

Energy-economic model for transition to renewables

John S. Mizanski

Masters of Modelling and Computational Science

Faculty of Graduate Studies

University of Ontario Institute of Technology

2013

Presented in this thesis is a macro-economic model based on a three-factor production function, utilizing energy, capital, and labour as the three production factors. With a primary focus on energy production we include two sources of energy, the first derived from a fixed amount of natural resources and the other generated from renewable sources. The production of energy from non-renewable sources is based on a Hubbert-type model of extraction. The production of energy from renewables is dependent on investment, innovation, and a natural limit to energy production. We also seek to include an aspect of investment in technology through the inclusion of efficiency and innovation factors. Two models of growth are examined for efficiency and innovation. The model, a set of differential algebraic equations, has been implemented in a C++ program which is provided at the end of this thesis. We provide two solution methods, the first based on a classic Runge-Kutta fourth order solution combined with Newton's method, and the second an implementation of the DASSL implicit DAE solver package. The model shows a promising incremental improvement over the model proposed by Berg *et al.* [2], however the sheer volume of parameters and the extensive sensitivity of the model to certain parameters introduces new difficulties in estimating values and hence the confidence in long term predictions made.

Acknowledgements

It is with immense gratitude that I acknowledge the support and help of my supervisor Dr. Peter Berg, without whose guidance this thesis would not have been possible. I would also like to extend thanks to all faculty members at UOIT for the exceptional experience and inspiration they provide in fostering creative applications of the tools taught. Thank you to my friends, parents, and colleagues as well for their much needed encouragement and support.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

For thousands of years prior to the utilization of fossil fuels as the primary fuel for energy production humans relied on energy derived from renewable sources, whether it was burning wood or charcoal for heat, harnessing wind for sailing, or consuming food which ultimately derives its energy content from the sun. With the development of fossil fuel based energy and the industrial revolution, the rate of production and consumption of energy has continued to climb ever higher and at an ever increasing rate. Given the finite nature of fossil fuel supplies it seems imperative to ask how long growth can be based on increasing production. Also, to what extent can innovation and renewable sources of energy relieve the demands for energy? There have been a few shocks to petroleum production within the past half century which have spurred research into understanding the relations between resource development and economic output (Dasgupta 1974 [7], Stiglitz 1974 [17], Solow 1978 [16], Weitzman 1999 [20], and more recently in relation to carbon emissions such as Tahvonen and Salo in 2001 [18]).

The work of Tahvonen and Salo in particular has provided a strong motivation for the work done in this thesis. In 2001, spurred on by the recently signed Kyoto accord they published a paper studying transitions between non-renewable and renewable energy within a welfare maximizing macroeconomic model. In particular they look at the path followed for non-renewable resource consumption both with and without technological advancement and conclude that CO2 emissions may follow an inverted-U relation with wealth even in the absence of environmental policy. Their exploration of technological advancement was also of interest as we explored the same areas in our model. This thesis explores many of the same areas, however it differs in approach. We also take additional inspiration from the work of Hubbert (1956) [10] on fossil fuel production and Kemfert (1998) [11] on economic constant elasticity of substitution (CES) production functions in creating the functional relations within our model. A key element in many of the models listed above is technological advancement or innovation, which as mentioned, are also examined within our model. More recently climate change and CO_2 emissions have driven research in energy production, along with the Tahvonen and Salo paper see also Nordhaus 1996 [14], Krabs and Pickl 2004 [12], and Yamaji *et al.* 2000 [1].

This thesis presents an energy-economic model for the output generated by an economy based on the supply of energy, labour, and investment capital. The model expands on and reworks the model developed by Berg *et al.* (2011) [2], however it should be noted that the introduction of non-renewable energy has changed the model dynamics and as such our model cannot replicate the results of the single non-renewable energy source model. It is built around a production function inspired by the work of Kemfert (1998) [11], utilizing the factors listed above. Kemfert analysed data from West German industry using several variations of a 2-level nested CES production function in order to estimate the values for the substitution elasticities. For our model, energy is supplied to the economy by two possible sources; the first comes from the model developed by Berg *et al.* and is a function for the production cycle of a non-renewable energy source. The function generates a production curve which resembles Gaussian curves from the work of M. K. Hubbert (1956) [10], and is itself a function of extraction efficiency, capital investment, and remaining reserves. To this we add a second source based on a renewable energy production. For this source the production function is dependent on technological innovation, capital investment, and a geological limit to production capacity. This geological limit has been used to capture the finite nature of our planet in size an resources, such as the land available for dedication to renewable energy production (damable rivers, suitable land for wind turbines, and solar arrays, amongst others). The capital investment in the economy is found by maximizing the profit within the economy, and for the available labour a simplifying assumption is carried over from the work of Berg *et al.*, that its growth is exponential. In this chapter the motivation for the model development is presented. In the following chapters, the derivation of the model, the numerical methods implemented in determining the model solutions, and the results obtained from testing the model are also presented. Finally, the conclusion gives an overview of the model's performance and suggestions for further improvements which could be explored.

1.1 Fossil fuel energy production

The function chosen for fossil fuel based energy production scales with remaining reserves, and capital investment. The function displays diminishing returns on investment capital and, due to the fixed initial reserves, also produces a Hubbert-style production peak. The paper written by M. King Hubbert in 1956 [10] was one of the first to explore the inevitable long term outcome of utilizing energy derived from non-renewable resources. In his paper Hubbert first examines the historical production of coal and oil. Observing the graphs, a similar pattern of slow initial growth emerges which rapidly increased exponentially until an inflection point is reached. Here, the rate of production increase slows, resulting in a peak and subsequent decline. The ultimate production cycle produces a plot resembling a Gaussian function. After a certain point the rate at which new reserves are discovered is not able to match the rate at which existing reserves are depleted and the overall production of the region must begin a decline. In his paper Hubbert (1956) [10] provides plots for the oil production for the world, the US, and for several US-states.

Two of those US states were Ohio, which was an early oil producer in the US whose production had a clear peak between 1890 and 1900 (Hubbert 1956, pg 10), and Illinois, whose production plot shows an early peak around 1910. This is followed by declining production until the late 1930s when production rapidly increases to a much higher peak in 1940, ending in another period of decline. The explanation provided by Hubbert for the second peak is in regards to new technology which enabled previously undiscovered reserves to be found and extracted. One astonishing prediction in the Hubbert paper is the peak of US oil production in 1970, which occurred 14 years after publication, as is shown in the plot of US oil production data gathered from British Petroleum (Figure 1.1).

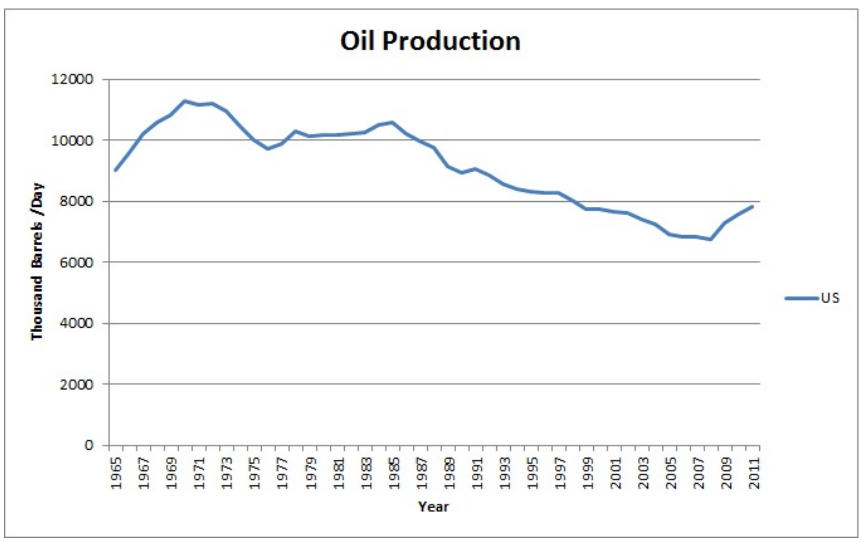

Figure 1.1: United States oil production (1965-2011): data from BP 2011

It should be noted that his paper also predicted a much quicker decline in production following the peak, which has not been observed. One country which provides a very clear example of the production curve predicted by Hubbert is Norway (see Figure 1.2). Norway has had a fairly stable political and regulatory environment, and production increases began at a relatively late period which meant that the level of technology required to find and develop the reserves was readily available.

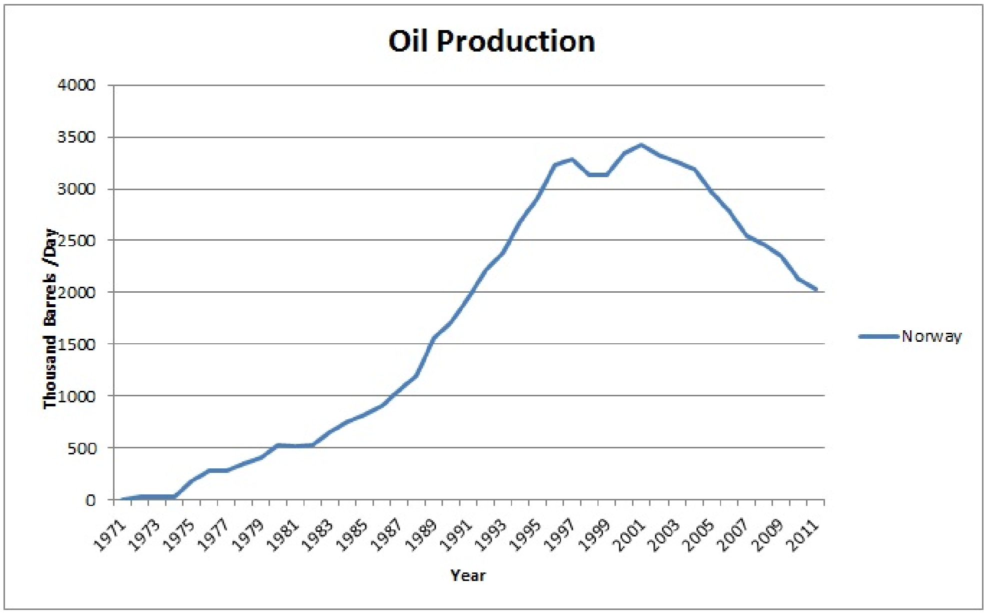

Figure 1.2: Norway oil production (1971-2011): data from BP 2011

However, looking at a region with wars and political events affecting it, the production becomes far less predictable. Figure 1.3 shows the oil production rates for the major producers from the Middle East (Iran, Iraq, Kuwait, and Saudi Arabia). The production curves do not resemble the Gaussian curves depicted by Hubbert and there are clear signs that conflicts during the 1970s and 1980s had long lasting impacts on the production rates.

Hubbert does highlight the possibility of technological improvements that can cause previously underdeveloped resources to be viable for development. We can see this type of production curve in Figure 1.4 for Canada where an earlier peak in production occurred for conventional oil but a more recent surge in production is due to development of the oil sands. The increased prices for petroleum, along with improvements in technology among other factors, make the more difficult task of extracting oil from oil sands economically viable.

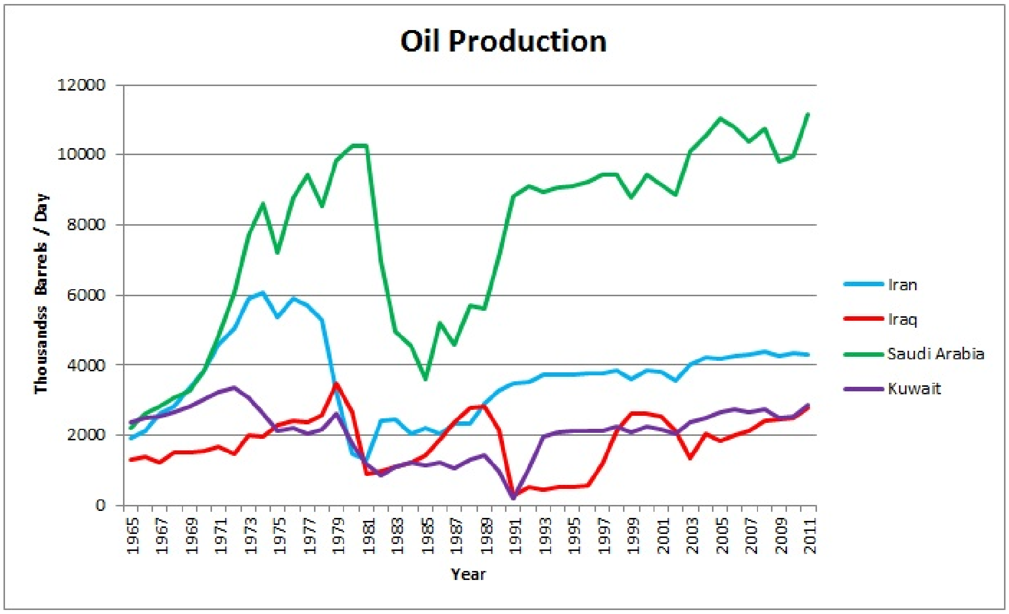

Figure 1.3: Oil production for Iran, Iraq, Kuwait, and Saudi Arabia (1965-2011): data from BP 2011

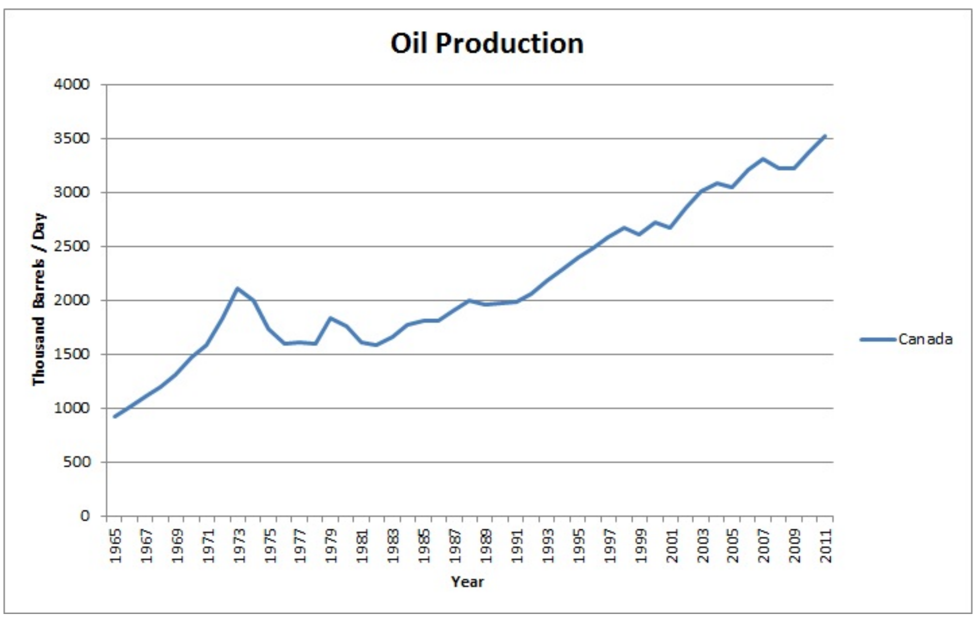

Figure 1.4: Canada oil production (1965-2011): data from BP 2011

1.2 Renewable energy production

For renewable energy the chosen production function depends on investment capital which allows the building and maintaining of infrastructure, and on innovation which improves the effectiveness in converting the investment capital into energy production capacity. The function also has a natural geological limit to the production capacity which sets a cap on the maximum production capacity that can be derived from renewable sources and reflects the finite nature of the planet. It seems possible that future inventions, discoveries or events might alter this limit; however for the purposes of this thesis the limit has been fixed. The resulting curve will resemble that depicted by Hubbert (1956) for nuclear energy, a monotonically increasing function in time which asymptotically approaches a maximum, set as a parameter for the model. This shape for a renewable energy production curve is also seen in the work of Tahvonen and Salo (2001) [18]. One would rationally expect a shape similar to this for a non-exhaustible source of energy. While production capacity is initially developed, it is placed in the most efficient areas and the most easily accessible. As the economies of scale begin to kick in, the rate of production capacity increase will grow rapidly. This additional production capacity will eventually become harder to install as locations to harness natural resources such as wind, solar, rivers, etc. will become increasingly more difficult to find. This will result in a leveling off of production capacity.

1.3 Simplifying assumptions

Due to the inherent complexity of the model a number of simplifying assumptions have been made. The first we examine is the growth rate of the available labour. This has been set to the growth rate of population which signifies that the economy experiences full employment, where productivity is derived from each person. This is obviously not the case in reality, and should be considered as a subject of further model development. The population has also been assumed to grow exponentially. Using data from the World Bank, the world population has been plotted along with exponential growth and a linear best-fit curve, as shown below (Figure 1.5).

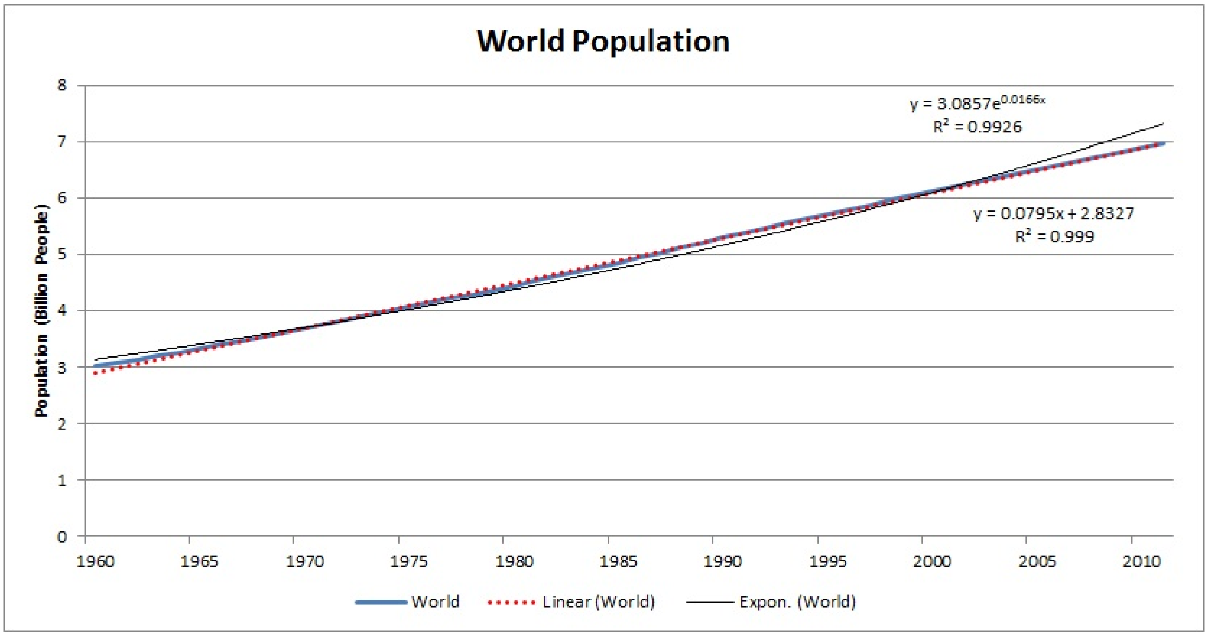

Figure 1.5: World population (1960-2011): data from World Bank 2011

From this we can see that although exponential growth is a relatively decent approximation with a growth rate of 1.66 percent, the linear approximation produces a better result when the R^2 values are compared. This may suggest that we are at an inflection point in the growth rate of world population. Although not considered in this thesis, changing the population growth rate to a linear function may produce an improvement in the model accuracy for short term predictions. For the case of long term predictions (100+ years) it seems plausible that the global population growth will continue to decrease and may even turn negative. In order to capture all these patterns a more sophisticated model for labour would be needed. This task is left to further expansions of the model, and we continue to carry forward the simplifying assumption of exponential growth.

In addition a number of other parameters have been simplified by the assumption of exponential growth, ignoring any functional relations they may possess. They include labour productivity which provides the contribution to economic output made by each person within the economy, and energy efficiency providing the conversion between energy consumption and the resulting contribution to economic production. The model is also presented initially with the assumption that technological change in the form of innovation for renewables and extraction efficiency for fossil fuels grow exponentially. As a modification to the model, this thesis examines removing this last assumption and providing differential equations for the growth rate of both. In order to simplify the introduction of a second energy source, the price of energy has been set to be the same no matter which source it is derived from, and therefore differences in production costs between energy sources is not reflected in their relative demand. This too should be considered as an area in which to make further improvements in the model.

1.4 Solution methods

The models developed in this thesis result in systems comprised of ordinary differential equations (ODE) and algebraic equations. These are known in a more general form as differential algebraic equations (DAE). The general implicit form is given by:

$$0 = F(t, y, y', x), \tag{1.1}$$

where y' is the derivative of y with respect to time. In this example y is the differential variable and x is the algebraic variable. The models we develop contain one (or two) ODEs and a single algebraic constraint. The system given above (equation 1.1) can be split into the differential and the algebraic equations:

$$0 = f(t, y, y', x), \tag{1.2}$$

$$0 = g(t, y, x). \tag{1.3}$$

In certain cases it is possible to take the derivative of the constraint g for the algebraic variable x and solve for x' resulting in an ODE which can be solved numerically using one of the many methods available for solving ODEs. However this is not always possible or practical, which necessitates the constraint being solved simultaneously with the ODEs. For the models presented in this thesis, two solution methods are implemented. The first has been built specifically for the models in this work, while the second is a software package, DASSL, which was developed by Linda Petzold (1996). A more detailed introduction to DAEs may be found in (Petzold 1996) [3].

This thesis is organized as follows; in the next chapter the model is developed, providing the mathematical framework. In chapter 3, the implemented solution methods are detailed. Chapter 4 contains the results from the many tests performed and chapter 5 gives the conclusion including the suggestions for future development. The appendix contains the C++ code developed for the models.

Chapter 2

The Model

Presented below is a model for an idealized economy which simulates economic growth, price of energy, and energy investment among other quantities. The output generated by this economy is a function of three input variables: labour, energy, and investment capital. The labour pool is assumed to grow exponentially and the model always exhibits full employment. The energy consumed by the economy is derived from two sources. Firstly, energy may be produced from the extraction of fossil fuels. We choose to model this by use of a Hubbert-style oil production function whereby the rate of production is proportional to the unused natural reserves. As was pointed out by Hubbert in his 1956 publication [10], the natural rate of transformation from organic matter to the hydrocarbons that make up fossil fuels is slow enough that from a modeling standpoint, the amount available may be regarded as fixed. The consequence of these two relations is that the production of energy from fossil fuels must eventually peak and go into decline as the quantity of untapped reserves dwindles and inevitably becomes depleted. This is often referred to as the Hubbert Peak. The second source of energy available in our model is derived from renewable sources. This energy source does not suffer from a fixed initial stock and as such does not present a physical limit to the total energy that may be produced. It is assumed however that a natural limit exists to the rate at which renewable energy may be produced. This limit can be interpreted as the theoretical maximum amount of power that may be produced within a finite space. For each form of renewable energy one can imagine limits to the availability of the natural resource from which the energy is derived. This parameter can also be thought to capture in a limited way certain aspects associated with land use within the economy. Further expansions to the model could examine the possibility of introducing variability of the limit in time. The ultimate goal in building this model is to begin examining the transition the economy must make as the reserves of fossil fuels decline.

Although our model seeks to simulate a substantial set of variables, listed in Table 2.1 below, there are three primary equations from which most other variables are derived: *i)* the production function which determines the output and consequently the growth of our economy; *ii)* the fossil fuel energy production function which sets the rate of production proportional to the untapped reserves, and *iii)* the renewable energy production function which seeks to model energy production as a function of innovation and capital investment. We begin this chapter by presenting these three equations and the motivation for their chosen form.

Variable	Name	Units
$L(t)$	Labour	people
$\gamma(t)$	Innovation in renewables industry	
$C(t)$	Fossil fuel extraction efficiency	$\$^{1-\epsilon}$ mega barrels$^{\epsilon-1}$
$A(t)$	Labour productivity	\$ / person
$B(t)$	Energy efficiency	\$ / mega barrel (equivalent)
$Y(t)$	Production function	$GDP(\$)$ / year
$K(t)$	Capital investment	\$ / year
$p_E(t)$	Price of energy	\$ / mega barrels of oil equivalent
$E_f(t)$	Energy from fossil fuels	mega barrels of oil equivalent
$E_r(t)$	Energy from renewables	mega barrels of oil equivalent
$K_f(t)$	Capital investment in fossil fuels	\$ / year
$K_r(t)$	Capital investment in renewables	\$ / year
ρ	Elasicity coefficient between capital and energy	
α	Elasicity coefficient between labour and capital/energy	
β	Parameter between capital and energy	
ϵ	Coefficient for diminishing returns of fossil fuels	
ϕ	Renewable energy investment factor	
r_K	Rent on capital	
r_E	Rent on energy investment capital	
R_f	Fossil fuel reserves	mega barrels of oil equivalent
R_r	Limit to energy production from renewables	mega barrels of oil equivalent

Table 2.1: Model variables and parameters with units.

2.1 Production function

Production functions have long been a topic of study in macroeconomics. The production function (PF) seeks to relate the output of an entity to the inputs utilized in the production of those outputs. For our model the production function provides the gross domestic product for the economy as a function of the labour, capital, and energy supplied. The CES (or Constant Elasticity of Substitution) production function is a common form whereby the ease with which one production input may be substituted by another is fixed. The general CES production function for a two-factor model utilizing production factors capital (K) and labour (L) is given by:

$$Y = F\left(\alpha K^r + (1-\alpha)L^r\right)^{1/r}.$$

Here, Y is output or GDP of the economy, F is the productivity factor which scales the production and can work as a parameter for efficiency, α is the share factor, and r is related to the elasticity of substitution s by,

$$s = 1/(1-r).$$

The elasticity of substitution is defined on the interval $0 < s < \infty$. Therefore, we have $-\infty < r < 1$. One limiting case which has been studied many times, is that of imperfect compliments, where $s = 1$ and $r = 0$. In order to derive this case we first transform the general CES production function into the following form,

$$lnY = lnF + \frac{ln\left(\alpha e^{rlnK} + (1-\alpha)e^{rlnL}\right)}{r}.$$

Now in order to take $\lim\limits_{r\to 0}$, we must apply L'Hopital's rule,

$$\lim_{r\to 0} lnY = \lim_{r\to 0}\left(lnF + \frac{\alpha ln(K)e^{rlnZ} + (1-\alpha)ln(L)e^{rlnL}}{\alpha e^{rlnZ} + (1-\alpha)e^{rlnL}}\right),$$

$$\Rightarrow \lim_{r\to 0} lnY = lnF + \alpha lnK + (1-\alpha)lnL.$$

This leads to what is referred to as a Cobb-Douglas form production function, named after Paul Douglas and Charles Cobb who developed this form in the 1920s [6],

$$Y = FK^{\alpha}L^{1-\alpha}.$$

In the Cobb-Douglas PF both production inputs are essential to the entity in producing output. Should any input fall to zero the output falls as well to zero. This production function along with the CES production function also allow for the property where by doubling both the capital and labour simultaneously will result in the output Y also doubling. Therefore if there were two identical worlds they would then have double the output of one world.

Major work on production functions in the first half of the 20th century focused primarily on functions of two inputs. Two common choices for those inputs were capital and labour (Douglas, 1928) [6] (Solow, 1956) [15]. The 1970s brought increased use of production functions, incorporating resources in various forms (Stiglitz, 1974) [17] (Solow, 1978) [16] (Dasgupta, 1974) [7].

We now present the production function chosen for our model. The production function gives the GDP in dollars Y from the inputs; labour L, capital K, and energy E. In this model, we choose to simplify the variety of energy sources currently available and represent the energy with two variables, one representing the energy produced from renewable sources E_r, and the other representing energy produced from fossil fuels E_f. Certainly for particular sectors of the economy such as agriculture land may seem a more important factor of production than energy. The global GDP however is primarily provided by the industrial and service sectors which combined account for roughly 94% of the total [5]. For this reason we have not included land explicitly but rather implicitly as a source of motivation for placing a limit on the production capacity in renewable energy.

$$Y = Y\left(L, K, E_f, E_r\right).$$

The production function is a key equation in the model. It provides the structural relations between the inputs and the final output, and within this it also carries the relations between the various inputs themselves. One of the more popular choices for relating the inputs is through use of the above mentioned substitution elasticities. These elasiticities of substitution are meant to convey the ease with which a decrease in one input may be countered with an increase in another input while keeping production output constant. It seems reasonable to expect that this could be a highly variable parameter over time and across regions, dependent on the available technology and the process under examination. However for simplicity and due to the wide usage of CES production functions we will use fixed elasiticities of substitution.

For our production function we begin with a two-level nested constant elasticity of substitution form inspired by the work of Kemfert on West German industry (Kempfert, 1998) [11]. Kemfert examined nested 3-factor CES production functions with production factors labour, energy and capital against industrial data. She found that the form with capital and energy as the nested pair produced good results from which the elasticities of substitution were estimated. For this form the inputs for energy and capital are linked with a substitution parameter, and the other parameter is used to model the substitution elasticity between labour and capital/energy:

$$Y = F\left[\alpha(\beta E^{\rho} + (1-\beta)K^{\rho})^{\frac{r}{\rho}} + (1-\alpha)L^{r}\right]^{\frac{1}{r}}.$$

The primary focus of this model is the relation between capital and energy. For this reason, we choose to set the elasticity of substitution for labour equal to one or $r = 0$. This results in a Cobb-Douglas style relation between labour and the combination of capital/energy, and as a result our economy will produce nothing without an initial population.

For clarity, let $Z = (\beta E^{\rho} + (1-\beta)K/^{\rho})^{1/\rho}$ and now write the production function as:

$$Y = F\left(\alpha Z^r + (1-\alpha)L^r\right)^{1/r}.$$

Following the same procedure as for the derivation of the Cobb-Douglas production function we end at:

$$Y = FZ^\alpha L^{1-\alpha}.$$

Now substituting the function for Z back in and setting $F = 1$, we arrive at the production function used by our model,

$$Y = \left[\beta(B(E_r + E_f))^\rho + (1-\beta)K^\rho\right]^{\frac{\alpha}{\rho}} (AL)^{(1-\alpha)}. \quad (2.1)$$

Both A and B are functions of time: A representing the labour productivity, and B representing energy efficiency. These replace the need for the parameter F and this is why we have chosen to set $F = 1$.

2.2 Profit

We now look to the equation for the profit generated by our economy. The profit may be found by subtracting the cost of inputs used from the total output produced. The inputs for our economy are energy, capital and labour. The associated costs are the price of energy p_E, the rent (or interest) paid on investment capital r_K, and the wage paid for labour w, respectively:

$$P_y = Y - p_{E_r}E_r - p_{E_f}E_f - wL - r_kK. \quad (2.2)$$

For simplicity, we impose equal pricing of energy regardless of the source, $p_{E_r} = p_{E_f}$. By using this assumption the economy exhibits no preference for any one form of energy over another. Realistically the price of energy should vary with the cost of production from source to source. However the prices would be expected to be of the same order of magnitude. Given the volatility of oil prices in reality that will not be captured by the model, we can only expect to simulate prices in the range of those viewed in the market and hence the assumption that the prices would be roughly equivalent seems justifiable as a starting point. Here, we write

$$P_y = Y - p_E\left(E_r + E_f\right) - wL - r_k K. \tag{2.3}$$

By maximizing the profit with respect to the energy, we obtain the marginal product of energy which is determined by the price of energy. And by maximizing the profit with respect to the investment capital, we obtain the marginal product of capital which is determined by the interest rate:

$$\begin{aligned}
\frac{\partial P_y}{\partial E_f} &= 0 = \frac{\partial Y}{\partial E_f} - p_E, \\
\frac{\partial P_y}{\partial E_r} &= 0 = \frac{\partial Y}{\partial E_r} - p_E, \\
\frac{\partial P_y}{\partial K} &= 0 = \frac{\partial Y}{\partial K} - r_k,
\end{aligned}$$

or simply

$$\frac{\partial Y}{\partial E_f} = \frac{\partial Y}{\partial E_r} = p_E$$

and

$$\frac{\partial Y}{\partial K} = r_k.$$

Next we find $\partial Y/\partial E_r$, $\partial Y/\partial E_f$, and $\partial Y/\partial K$ from the production function. First, we get

$$\frac{\partial Y}{\partial K} = \alpha(1-\beta)K^{\rho-1}(AL)^{1-\alpha}\left[\beta\left(B(E_r+E_f)\right)^{\rho} + (1-\beta)K^{\rho}\right]^{\alpha/\rho-1} = r_k.$$

Now for energy, we obtain

$$\frac{\partial Y}{\partial E_r} = \alpha\beta B^\rho (E_r + E_f)^{\rho-1} (AL)^{1-\alpha} \left[\beta \left(B(E_r + E_f)\right)^\rho + (1-\beta)K^\rho\right]^{\alpha/\rho - 1} = p_E. \qquad (2.4)$$

Finally dividing the price of energy by the rent on capital, we arrive at

$$\frac{p_E}{r_k} = \frac{\beta B^\rho (E_r + E_f)^{\rho-1}}{(1-\beta)K^{\rho-1}} \Rightarrow \frac{K}{E_r + E_f} = \left(\frac{\beta B^\rho r_k}{(1-\beta)p_E}\right)^{\frac{1}{\rho-1}},$$

and by bringing the energy terms over to the right-hand side, we arrive at our equation for capital investment

$$K = (E_r + E_f)\left(\frac{(1-\beta)p_E}{\beta B^\rho r_k}\right)^{\frac{1}{1-\rho}}. \qquad (2.5)$$

2.3 Energy

One of the primary motivations for the work on this model has been to examine the impact of energy availability on production. The functions we choose to implement as our energy supply from both fossil fuels and renewable sources will therefore have a defining role in how each of these sources is exploited.

2.3.1 Energy from fossil fuels

There are two sources for energy in our economy, and the first is dependent on the consumption of some fixed reserves of fossil fuels E_f. Motivated by the work of Hubbert (Hubbert, 1950) [10], we choose to scale the energy produced by fossil fuels with the remaining reserves, given by $R_f - Q$, where R_f is the initial global reserves of fossil fuels and Q is the cumulative production to date. The energy produced is set equal to the rate of extraction. The energy produced also scales with investment in the petroleum industry K_f and we make use of a diminishing returns parameter $0 < \epsilon < 1$,

$$E_f = \frac{dQ}{dt} = C{K_f}^{\epsilon} \left(R_f - Q\right)^{1-\epsilon}. \tag{2.6}$$

Here, C represents the efficiency with which the fossil fuels may be extracted and is time dependent. This form for the fossil fuel energy production function scales to multiple worlds as when the initial reserves R_f, the cumulative production Q, and the investment capital K_f are all doubled, the production of energy also doubles.

2.3.2 Energy from renewables

The energy from renewables can be modeled as a fraction of a theoretical maximum R_r. The energy production level is driven by capital investment K_r, and innovation γ determines the effectiveness of the capital invested. The parameter K_0 influences the rate at which the maximum production level is reached and $\phi \geq 0$ helps control the influence between capital investment and innovation. In order to maintain R_r as a maximum we must limit $K_0 \geq 0$. Although not necessary we have limited our testing on ϕ to values less than 1, beyond this level we found capital to be too effective, resulting in very low investment levels over time. Additionally, an innovation factor has been modeled with the parameter $\gamma(t)$. In time as the level of innovation increases the requirement for capital investment in renewable energy K_r is diminished in order to achieve the same level of production.

$$E_r = R_r \frac{\gamma\left(t\right) {K_r}^{\phi}}{\gamma\left(t\right) {K_r}^{\phi} + K_0^{\phi}}. \tag{2.7}$$

From this relation, we can deduce that the energy produced from renewables will follow a monotonically increasing function of capital and innovation that scales with the invested capital K_r up to a natural limit R_r. Figure 2.1 below shows plots for equation (2.7), where ϕ is successively increased for each plot from left to right. As ϕ is increased, the role of capital investment in reaching maximum production is increased.

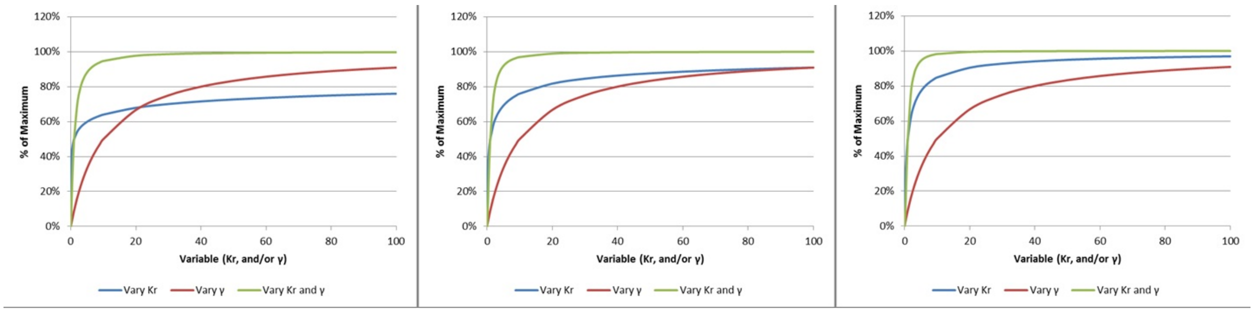

Figure 2.1: Renewable energy production as a percentage of its maximum, for $\phi = 0.25$, $\phi = 0.5$ and $\phi = 0.75$. For each case the energy produced is plotted as a function of K_r while $\gamma = 1$ is fixed (Blue), as a function of γ while $K_r = 1$ is fixed (Red), and as a function of both (Green). $K_0 = 0.1$ for all plots

It should also be noted that this form for the renewable energy production scales to multiple worlds, for two identical worlds the maximum level of production R_r, and the capital investment parameters K_r and K_0 would be doubled and consequently the energy produced will also double.

2.3.3 Price of energy

If we take our previous equation for the price of energy (equation 2.4), and substitute our capital investment (equation 2.5) in place of K, we arrive at

$$p_E = \left(\frac{(E_r + E_f)^{\rho-1}}{(AL)^{\alpha-1}}\right) \alpha\beta B^\rho \left[\beta\left(B(E_r + E_f)\right)^\rho + (1-\beta)\left((E_r + E_f)\left(\frac{\beta B^\rho r_k}{(1-\beta)p_E}\right)^{\frac{1}{\rho-1}}\right)^\rho\right]^{\frac{\alpha}{\rho}-1},$$

or

$$p_E = \frac{\alpha\beta B^\rho}{(1-\beta)}\left(\frac{E_r + E_f}{AL}\right)^{\alpha-1}\left[\frac{\beta B^\rho}{(1-\beta)} + \left(\frac{(1-\beta)p_E}{\beta B^\rho r_k}\right)^{\frac{\rho}{1-\rho}}\right]^{\frac{\alpha}{\rho}-1}. \tag{2.8}$$

As we can see, p_E is found on the right-hand side of this equation. Here, we have an implicit non-linear function for the price of energy.

2.3.4 Capital investment in energy sector

The profit of generating energy from fossil fuels P_f or renewables P_r is found by subtracting the cost of capital invested in producing the energy from the revenue generated by selling the energy,

$$\begin{aligned} P_f &= p_E E_f - r_E K_f, \\ P_r &= p_E E_r - r_E K_r. \end{aligned}$$

Here it is assumed that the energy industries are capital intensive but not labour intensive. The cost of capital or interest is represented here by r_E and is fixed for the entire energy industry. These profit equations provide an opportunity to include certain policy effects, such as subsidies or taxes. Although this is not something examined in this thesis it could provide an interesting point for future work. Maximizing the profit with respect to capital invested gives the marginal product of investment capital for the fossil fuel industry,

$$\frac{\partial E_f}{\partial K_f} = \frac{\partial E_r}{\partial K_r} = \frac{r_E}{p_E}.$$

Next we look to find $\partial E_f / \partial K_f$ and $\partial E_r / \partial K_r$ from the supply equations (2.6) and (2.7). First, we have

$$\frac{\partial E_f}{\partial K_f} = \epsilon {K_f}^{\epsilon - 1} C (R - Q)^{1-\epsilon}. \tag{2.9}$$

Substituting $\partial E_f / \partial K_f = r_E / p_E$ into equation (2.9) and solving for K_f gives

$$K_f = (R - Q) \left(\frac{p_E \epsilon C}{r_E} \right)^{\frac{1}{1-\epsilon}}. \tag{2.10}$$

At this point, we will also substitute this equation for fossil fuel capital investment back into the supply equation (2.6) to give

$$E_f = C^{\frac{1}{1-\epsilon}} \left(\frac{p_E \epsilon}{r_E} \right)^{\frac{\epsilon}{1-\epsilon}} (R_f - Q). \tag{2.11}$$

Following the same procedure for our renewables supply equation (2.7) yields

$$\frac{\partial E_r}{\partial K_r} = R_r \frac{\gamma^2 \phi {K_r}^{2\phi-1} + \gamma \phi {K_0}^{\phi} {K_r}^{\phi-1} - \gamma^2 \phi {K_r}^{2\phi-1}}{\left(\gamma {K_r}^{\phi} + {K_0}^{\phi}\right)^2} = \frac{R_r \gamma \phi {K_0}^{\phi} {K_r}^{\phi-1}}{\left(\gamma {K_r}^{\phi} + {K_0}^{\phi}\right)^2}.$$

Substituting $\partial E_r / \partial K_r = r_E / p_E$ and rearranging, we get

$$\frac{\gamma}{{K_0}^{\phi}} {K_r}^{\phi+1} + 2K_r + \frac{{K_0}^{\phi}}{\gamma} {K_r}^{1-\phi} - \frac{\phi R_r p_E}{r_E} = 0$$

or

$$p_E = \frac{r_E}{R_r \phi} \left[\frac{\gamma}{{K_0}^{\phi}} {K_r}^{\phi+1} + 2K_r + \frac{{K_0}^{\phi}}{\gamma} {K_r}^{1-\phi} \right]. \tag{2.12}$$

If we then recall equation (2.8)

$$p_E = \frac{\alpha \beta B^{\rho}}{(1-\beta)} \left(\frac{E_r + E_f}{AL} \right)^{\alpha-1} \left[\frac{\beta B^{\rho}}{(1-\beta)} + \left(\frac{(1-\beta) p_E}{\beta B^{\rho} r_k} \right)^{\frac{\rho}{1-\rho}} \right]^{\frac{\alpha}{\rho}-1},$$

and substitute equations (2.7) for E_r, and (2.11) for E_f and then equation (2.12) for p_E, we arrive at a very intricate, highly non-linear equation for K_r:

$$\begin{aligned} 0 = & \frac{r_E}{\phi R_r} \left[2K_r + \frac{\gamma(t)}{{K_0}^{\phi}} {K_r}^{\phi+1} + \frac{{K_0}^{\phi}}{\gamma(t)} {K_r}^{1-\phi} \right] - \frac{\alpha \beta B(t)^{\rho}}{(1-\beta)} (A(t)L(t))^{1-\alpha} \\ & \cdot \left[\frac{\beta B(t)^{\rho}}{(1-\beta)} + \left(\frac{(1-\beta) r_E}{\phi R_r \beta B(t)^{\rho} r_k} \left[K_r + \frac{\gamma(t)}{{K_0}^{\phi}} {K_r}^{\phi+1} + \frac{{K_0}^{\phi}}{\gamma(t)} {K_r}^{1-\phi} \right] \right)^{\frac{\rho}{1-\rho}} \right]^{\frac{\alpha}{\rho}-1} \\ & \cdot \left(\left[\frac{R_r \gamma(t) {K_r}^{\phi}}{\gamma(t) {K_r}^{\phi} + K_0^{\phi}} \right] + \left[C(t)^{\frac{1}{1-\epsilon}} \left(\frac{\epsilon}{\phi R_r} \left[2K_r + \frac{\gamma(t)}{{K_0}^{\phi}} {K_r}^{\phi+1} + \frac{{K_0}^{\phi}}{\gamma(t)} {K_r}^{1-\phi} \right] \right)^{\frac{\epsilon}{1-\epsilon}} (R_f - Q) \right] \right)^{\alpha-1} \end{aligned}$$

Despite the intricate nature of the constraint equation in developing and testing our numerical methods we were able to see that the roots are quite well defined. This constraint needs to be solved in our numerical simulation.

2.4 Labour and population

The labour is a measure of the population which contributes to the production Y. In reality there would be a rate of unemployment and the labour pool would be a function of the population and the unemployment rate. The unemployment rate could then be dependent on a number of factors such as wages. It is also plausible that in the long term (100+ years) the global population may plateau or even go to a period of decline. Since the focus of the work in this thesis has been on energy it has been decided to carry forward the simplified exponential growth for population from Berg *et al.* [2]. This could be interpreted as exponential population growth with the economy representing full employment, or more simply that the pool of available labour grows exponentially in time,

$$L(t) = L_0 e^{g_L t}. \tag{2.13}$$

Although this is certainly not a realistic function for very large time periods it will work as a simple approximation for short time scales of a few decades. Along similar lines, we set the functions for labour efficiency $A(t)$ and energy efficiency $B(t)$ to be exponential in time as well.

2.5 DAE system

Taking all this together we have a system with 4 remaining variables: the amount of fossil fuels extracted Q, the capital investment in renewables K_r, the fossil fuel extraction efficiency C, and the innovation in renewables γ. For this model, we have chosen two methods of representing C and γ. The first is to treat them as exponential functions in time as was done for labour, energy efficiency and labour productivity. The second is to provide differential equations for both, where the growth rate is dependent on profit and investment.

2.5.1 Exponential technological growth

For the first case, we have a differential algebraic equation (DAE) system with an ordinary differential equation (ODE) for Q and a non-linear constraint equation to solve for K_r:

$$\frac{dQ}{dt} = C(t)^{\frac{1}{1-\epsilon}} \left(\frac{\epsilon}{\phi R_r} \left[2K_r + \frac{\gamma(t)}{{K_0}^{\phi}} {K_r}^{\phi+1} + \frac{{K_0}^{\phi}}{\gamma(t)} {K_r}^{1-\phi} \right] \right)^{\frac{\epsilon}{1-\epsilon}} (R_f - Q) \tag{2.14}$$

with the constraint

$$0 = p_E(K_r) - \alpha X \left(\frac{AL}{E_r(K_r) + E_f(K_r, Q)} \right)^{1-\alpha} \left[X + \left(\frac{p_E(K_r)}{X r_k} \right)^{\frac{\rho}{1-\rho}} \right]^{\frac{\alpha}{\rho}-1} . \tag{2.15}$$

Here, we used

$$X(t) = \frac{\beta (B(t))^{\rho}}{(1-\beta)}$$

and $p_E(K_r)$, $E_r(K_r)$ and $E_f(K_r, Q)$ are given by equations (2.12), (2.7) and (2.11) respectively.

2.5.2 Technological change determined dynamically

As an improvement to the original model, we decided to replace the assumption that technological advancements in renewables production γ and fossil fuel extraction C grow exponentially in time. Although there are many relations and variables that would seem to present themselves as potentially having an effect on technological change, we choose a fairly simple relation at least in part motivated by the work of Hartwick [9] who examined the policy of utilizing the profits derived from the production of exhaustible resources for investment in renewable ones. For our relation the rate of technological change is driven by the profits generated in the energy sector. It certainly seems plausible that the technology improvements would be developed internally by the energy sector and during years when profits are plentiful; an increase in research spending would result in an increase in the rate of technological change. This method also seems to present a market driven approach to the investments whereby investment in an energy sector follows demand. The ODEs we have chosen for our model are as follows:

$$\frac{dC}{dt} = \tau_f \frac{P_f}{K_f}, \tag{2.16}$$

$$\frac{d\gamma}{dt} = \tau_r \frac{P_r}{K_r} + \tau_{rf} \frac{P_f}{K_r}. \tag{2.17}$$

In these equations we can see that technological innovation increases during periods when the energy sectors achieve higher returns on their capital investment. A coefficient (τ_r) is used to determine the conversion between the returns and the rate of innovation growth. We have also added a second term to the innovation in the renewables sector. It allows for profits from the fossil fuels sector to drive innovation in the renewables sector, and another coefficient (τ_{rf}) is used as a scaling factor. Expressing these two ODEs by the variables from our reduced system, we have the following equations:

$$\frac{d\gamma}{dt} = \tau_r \frac{P_r}{K_r} + \tau_{rf} \frac{P_f}{K_r}$$

$$\Rightarrow \frac{d\gamma}{dt} = \tau_r \left(\frac{p_E E_r}{K_r} - r_E \right) + \tau_{rf} \left(\frac{p_E E_f}{Kr} (1 - \epsilon) \right) \tag{2.18}$$

and

$$\frac{dC}{dt} = \tau_f \frac{P_f}{K_f} = \tau_f r_E \left(\frac{1}{\epsilon} - 1 \right). \tag{2.19}$$

The latter can be solved to produce linear growth in extraction efficiency,

$$C = \tau_f r_E \left(\frac{1}{\epsilon} - 1 \right) t + C_0. \tag{2.20}$$

So for this case, our resulting DAE system is given by two ODEs (one for Q and one for γ), and a non-linear constraint for K_r.

Chapter 3

Model Implementation

As was shown in the previous chapter our models each result in a DAE system. For our model using exponential growth for extraction efficiency and innovation our resultant system is comprised of one ODE for cumulative fossil fuel production $Q(t)$ (equation 2.14) and a non-linear constraint which we must solve for the capital investment in renewables $K_r(t)$ (equation 2.15). For the model where innovation and fossil fuel extraction efficiency are modeled by ODE functions the resultant DAE system includes the ODE for innovation (equation 2.18) in addition to equation (2.14) and equation (2.15). Two numerical methods have been implemented within the code provided in appendix 1.

3.1 DASSL

The first method uses the freely available software package DASSL. The DASSL software was developed in the FORTRAN programing language by Linda Petzold *et al.* [3]. The software utilizes backward differentiation formulas (BDF) of variable order (from 1 to 5) with a variable step size. A fixed leading coefficient implementation of the BDF method is their chosen way to expand the BDF method to variable step sizes. Further details on the DASSL software and the solution methods implemented are found in [3].

Since the DASSL software has been developed to solve implicit DAE systems we must state our model in this form. In the introduction, equation (1.1) gives the general implicit form for a DAE system. Transforming our model equations, equation (2.14), equation (2.15), and equation (2.18) into this form we arrive at the following system which may be provided to DASSL.

$$\begin{aligned} 0 &= F1(t, K_r, Q, \gamma) \\ &= p_E(K_r) - \alpha X \left(\frac{AL}{E_r(K_r) + E_f(K_r, Q)} \right)^{1-\alpha} \left[X + \left(\frac{p_E(K_r)}{X r_k} \right)^{\frac{\rho}{1-\rho}} \right]^{\frac{\alpha}{\rho}-1}, \end{aligned} \tag{3.1}$$

where,

$$X(t) = \frac{\beta(B(t))^{\rho}}{(1-\beta)},$$

$$\begin{aligned} 0 &= F2(t, K_r, Q, Q', \gamma) \\ &= \frac{dQ}{dt} - C(t)^{\frac{1}{1-\epsilon}} \left(\frac{\epsilon}{\phi R_r} \left[2K_r + \frac{\gamma(t)}{{K_0}^{\phi}} {K_r}^{\phi+1} + \frac{{K_0}^{\phi}}{\gamma(t)} {K_r}^{1-\phi} \right] \right)^{\frac{\epsilon}{1-\epsilon}} (R_f - Q), \end{aligned} \tag{3.2}$$

and

$$\begin{aligned} 0 &= F3(t, K_r, Q, \gamma, \gamma') \\ &= \frac{d\gamma}{dt} - \tau_r \left(\frac{p_E E_r}{K_r} - r_E \right) + \tau_{rf} \left(\frac{p_E E_f}{Kr} (1-\epsilon) \right). \end{aligned} \tag{3.3}$$

Therefore the implict form for the system is given by,

$$0 = F(t, K_r, Q, Q', \gamma, \gamma') = [F1, F2, F3].$$

Although DASSL provides the option of supplying an external Jacobian function for the system, we have chosen to utilize the default setting in which the partial derivatives are approximated numerically by finite difference equations.

3.2 Runge-Kutta with Newton's method

The second solution method implemented is built around the use of an explicit Runge-Kutta method to solve the ODEs. This method has been enhanced by ensuring the non-linear constraint is fulfilled at each stage of the Runge-Kutta method. The classic fourth-order Runge-Kutta (RK4) method is given by [8]

$$
\begin{aligned}
w_0 &= \alpha \\
k_1 &= hf(t_i, w_i) \\
k_2 &= hf\left(t_i + \frac{h}{2}, w_i + \frac{k_1}{2}\right) \\
k_3 &= hf\left(t_i + \frac{h}{2}, w_i + \frac{k_2}{2}\right) \\
k_4 &= hf(t_i + h, w_i + k_3) \\
w_{i+1} &= w_i + \frac{1}{6}(k_1 + 2k_2, 2k_3, k_4) \\
For : i &= 0, 1, 2, ..., N-1
\end{aligned}
$$

To apply this to our models the function $f(t, w)$ represents the ODEs for $Q(t)$ (equation 2.14) and $\gamma(t)$ (equation 2.18). And to ensure that the investment in renewables is able to shift in reaction to changes in innovation and fossil fuel reserves, a Newton's method is used to solve the non-linear constraint (equation 2.15) at each stage. This methodology was also used by Berg *et al.* [2] for their DAE system and the results displayed an acceptable level of relative error compared to their analytical solution. Since we do not have an analytical solution with which to compare our RK4 solution we will examine the relative error between our RK4 solution and the solution from the implicit DAE solver DASSL.

In order to solve the constraint using a Newton's method we must first take the partial derivative of equation (2.15) with respect to K_r.

Let us define

$$0 = g(K_r) = p_E(K_r) - \alpha X(AL)^{1-\alpha} h(K_r) j(K_r) \tag{3.4}$$

where

$$X(t) = \frac{\beta (B(t))^{\rho}}{(1-\beta)},$$

$$h(K_r) = (E_r + E_f)^{\alpha-1} = \left[R_r \frac{\gamma(t) {K_r}^{\phi}}{\gamma(t) {K_r}^{\phi} + K_0^{\phi}} + C^{\frac{1}{1-\epsilon}} \left(\frac{p_E \epsilon}{r_E} \right)^{\frac{\epsilon}{1-\epsilon}} (R_f - Q) \right]^{\alpha-1} \tag{3.5}$$

and

$$j(K_r) = \left[X + \left(\frac{1}{X \cdot r_E} \right)^{\frac{\rho}{1-\rho}} \cdot p_E^{\frac{\rho}{1-\rho}} \right]^{\frac{\alpha}{\rho}-1}. \tag{3.6}$$

We know that

$$p_E = \frac{r_E}{R_r \phi} \left[\frac{\gamma}{{K_0}^{\phi}} {K_r}^{\phi+1} + 2K_r + \frac{{K_0}^{\phi}}{\gamma} {K_r}^{1-\phi} \right].$$

This yields

$$\frac{\partial p_E}{\partial K_r} = \frac{r_E}{R_r \phi} \left[\frac{(1+\phi)\gamma}{{K_0}^{\phi}} {K_r}^{\phi} + 2 + \frac{(1-\phi){K_0}^{\phi}}{\gamma} {K_r}^{-\phi} \right]. \tag{3.7}$$

The partial derivatives of $h(K_r)$ and $j(K_r)$ w.r.t. K_r are:

$$\frac{\partial h}{\partial K_r} = (\alpha-1)(E_r + E_f)^{\alpha-2} \left[R_r \frac{\phi \gamma {K_0}^{\phi} {K_r}^{\phi-1}}{\left(\gamma {K_r}^{\phi} + K_0^{\phi} \right)^2} + (R_f - Q) C^{\frac{1}{1-\epsilon}} \left(\frac{\epsilon}{r_E} \right)^{\frac{\epsilon}{1-\epsilon}} \left(\frac{\epsilon}{1-\epsilon} \right) p_E^{\frac{\epsilon}{1-\epsilon}-1} \frac{\partial p_E}{\partial K_r} \right] \tag{3.8}$$

and

$$\frac{\partial j}{\partial K_r} = \left(\frac{\alpha}{\rho} - 1 \right) \left[X + \left(\frac{1}{X \cdot r_E} \right)^{\frac{\rho}{1-\rho}} \cdot p_E^{\frac{\rho}{1-\rho}} \right]^{\frac{\alpha}{\rho}-2} \left[\left(\frac{1}{X \cdot r_E} \right)^{\frac{\rho}{1-\rho}} \left(\frac{\rho}{1-\rho} \right) p_E^{\frac{\rho}{1-\rho}-1} \frac{\partial p_E}{\partial K_r} \right]. \tag{3.9}$$

Now by aggregating equations (3.5) through (3.9), the partial derivative of $g(K_r)$ can be found,

$$\frac{\partial g}{\partial K_r} = \frac{\partial p_E}{\partial K_r} - \alpha \cdot X(AL)^{1-\alpha}\left[h \cdot \frac{\partial j}{\partial K_r} + j \cdot \frac{\partial h}{\partial K_r}\right], \tag{3.10}$$

and our Newton's method approximation to K_r may be found by iterating [8]

$$K_{ri+1} = K_{ri} - \frac{g(K_r)}{\partial g/\partial K_r}. \tag{3.11}$$

Chapter 4

Results

This chapter is divided into three primary sections. In the first, the behaviour of the model and the sensitivity to various model parameters are presented. In the second section, an attempt to best fit the model to real world data is made and in the third, the model solutions generated by use of DASSL and our RK4 method are examined. In this thesis, the focus for testing has been primarily on the parameters most closely related to energy production, whether from fossil fuels or renewable sources. For all tests, the model has been set to simulate the entire lifetime for the fossil fuel reserves. $Q(0)$ has therefore been set to zero for all test runs and the stopping criteria has been chosen to be when $Q = R_f$, or in other words, when all fossil fuel reserves have been depleted. The selected step size for all tests was 0.08, equivalent to roughly one month[1]. The sensitivity of the model to step size is explored briefly in section 2. As well, aside from section 3 the fossil fuel reserves are set at 1,000,000 units. In table 4.1, the fixed model parameters for all tests are provided. All test results presented in this chapter, unless otherwise stated, were produced using the DASSL solution method for the ODE model with the tolerance set to 10^{-5}.

The initial value for the investment in renewables $K_r(0)$ is found by using our New-

[1] This excludes the tests run to examine the sensitivity of the solutions to step size.

Name	Parameter	Value
Renewable energy production constant	K_0	0.1
Initial population	$L(0)$	100,000
Initial innovation	$\gamma(0)$	0.001
Initial extraction efficiency	$C(0)$	0.001
Initial labour productivity	$A(0)$	0.01
Initial energy efficiency	$B(0)$	0.01
Growth rates	$g_{L,A,B}$	0.01
Fossil fuel reserves	R_f	1,000,000
Interest rates	r_E, r_K	0.05
Labour to capital-energy production parameter	α	0.25
Capital to energy share parameter	β	0.5
Capital to energy production parameter	ρ	0.05
Extraction efficiency growth factor	τ_f	0.001
DASSL tolerance	*tolerance*	10^{-5}

Table 4.1: Model settings used for parameter testing.

ton's method solver with the initial conditions for all other variables inserted. This does however require that we provide an initial guess to the Newton's solver, and this leads to the question of whether our supplied initial guess will result in the correct solution being found. By simply examining the behaviour of the condition (equation 2.15) within a reasonable range around the root showed that for the variables used in our tests, only one root was present. Our initial guess for the parameter testing was $K_r(0) = 1$, and for our real world test the initial guess supplied was $K_r(0) = 1000$. However, the root at $t = 0$ is considerably higher on the order of 10^{11} dollars. The remaining variables not listed in table 4.1 will be examined in this chapter. In section 1, we study the reaction of the model to changes in the following constants; ϵ (diminishing returns factor for capital investment in fossil fuels), ϕ (renewable energy capital investment relative influence factor), τ_{rf} (innovation growth factor for fossil fuel profits), τ_r (innovation growth factor for renewable energy profits), R_r (maximum energy production possible from renewable energy), and r_E (interest rate for capital in the energy sector). In section 2, using data retrieved from British Petroleum (2011) and the World Bank (2011), the model parameters are adjusted so as to fit the model solution to real world data. In section 3, the discrepancies between our solution methods are examined and the sensitivity to step size for each model is provided.

4.1 Model behaviour and sensitivity

In this section, a general overview of the model behaviour and reaction to the variation of selected parameters is presented. The primary advances in this model beyond the work by Berg *et al.* in 2011 have been related to energy functions including innovation in renewable energy, and extraction efficiency for fossil fuels. For this reason, the parameters chosen and presented below are related most closely to these functions. Beginning with fossil fuel energy generation, the model sensitivity to changes in the diminishing returns factor, ϵ, is examined. Subsequently, the renewable energy generation is examined and the sensitivity to variations in the factor ϕ, which controls the relative importance of innovation and capital investment, are presented. Beyond these, the model sensitivity is also examined for changes in the renewable energy production cap, R_r. Also, the innovation function is examined and variations in both τ_r and τ_{rf} are presented. The final portion of this section is set aside to present the model sensitivity to interest rates.

4.1.1 Fossil fuel diminishing returns factor

Let us recall our equation for fossil fuel energy production,

$$E_f = C^{\frac{1}{1-\epsilon}} \left(\frac{p_E \epsilon}{r_E} \right)^{\frac{\epsilon}{1-\epsilon}} (R_f - Q) = C {K_f}^{\epsilon} (R_f - Q)^{1-\epsilon}. \tag{4.1}$$

The parameter ϵ represents the diminishing returns factor for capital investment and remaining reserves. In reality this parameter could reflect the nature of accessibility of the reserves. Progressively the act of extracting oil becomes more difficult and finding new reserves requires looking in ever more difficult regions to access. In this section, we examine the effects of varying ϵ on energy production, investment, and the price of energy. As we can see from the equation above, as ϵ is increased the exponents of extraction efficiency C and of the fraction $\frac{p_E \epsilon}{r_E}$ approach one another, although as long as ϵ is below 1 the exponent on extraction efficiency will always be larger. Or more simply, as ϵ is increased from 0 to 1, the importance of capital investment grows and the level of remaining reserves has diminishing importance in determining production. For our test, the parameter ϕ was set to 0.5 as a mid-point of the range, R_r caps renewable production at 4,000 units representing a level slightly below the peak production of fossil fuels on a path of tempered resource depletion, the innovation parameter τ_r is 0.001, and τ_{rf} is set to 0. The interest rates have been fixed at 5%. Below we present the various fossil fuel production curves as ϵ is varied from 0.075 to 0.75.

As can be seen in Figure 4.1, an adjustment in the value of ϵ results in a shift of the

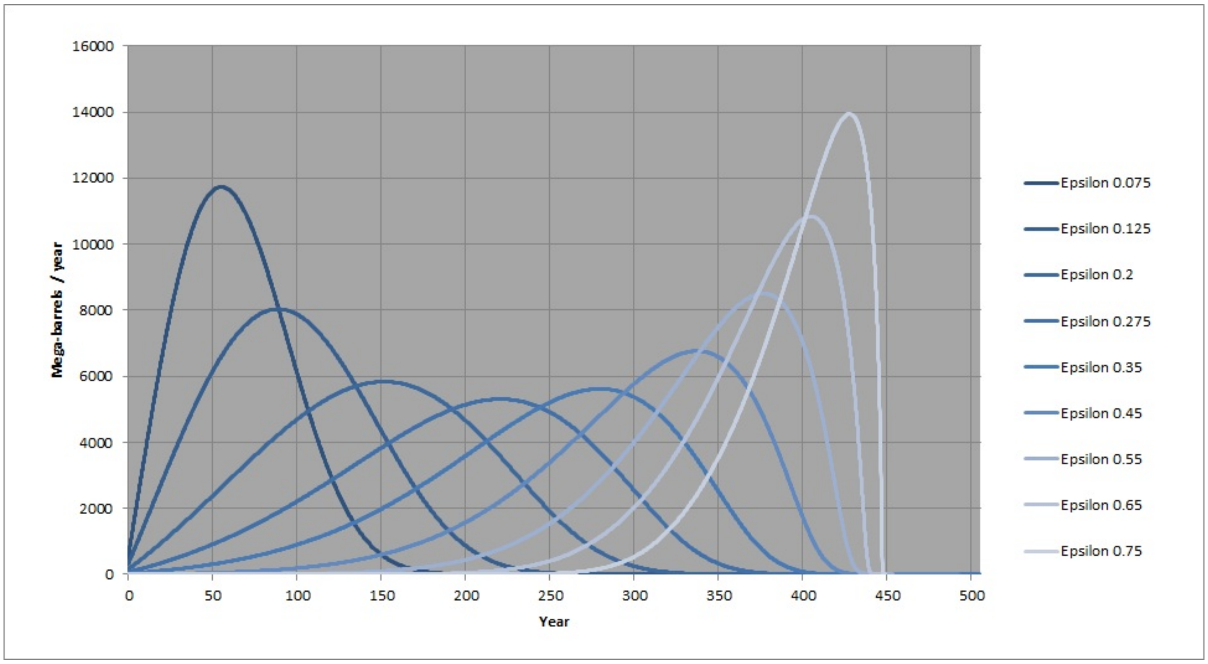

Figure 4.1: ϵ test: fossil fuel energy production (E_f)

production peak, earlier in time for smaller values and later in time for larger values. Since the available resources are finite, the area under each curve is the same and thus curves with higher peaks also contain steeper increases and decreases surrounding the peak. The most gradual peaks occur for values of epsilon in the range of 0.2 to 0.35. As the value of ϵ is moved further away from this range, we see that the production peak becomes much steeper and reaches much higher maximum production rates. It is important to note that the qualitative result of a bell shaped production curve is unaffected by changes in ϵ. Below we show the corresponding plots for cumulative production (Figure 4.2).

Again from this plot, for the mid-range values of ϵ the consumption of fossil fuels is most gradual and for values outside this range, the majority of consumption occurs over a shorter period of time, either earlier for lower values of ϵ or later for higher values.

With the variation in available fossil fuel energy we also see changes in the production of renewable energy as ϵ is varied. In Figures 4.3 and 4.4, we present two plots exhibiting the various paths for renewable energy production. The first plot (Figure 4.3) gives the

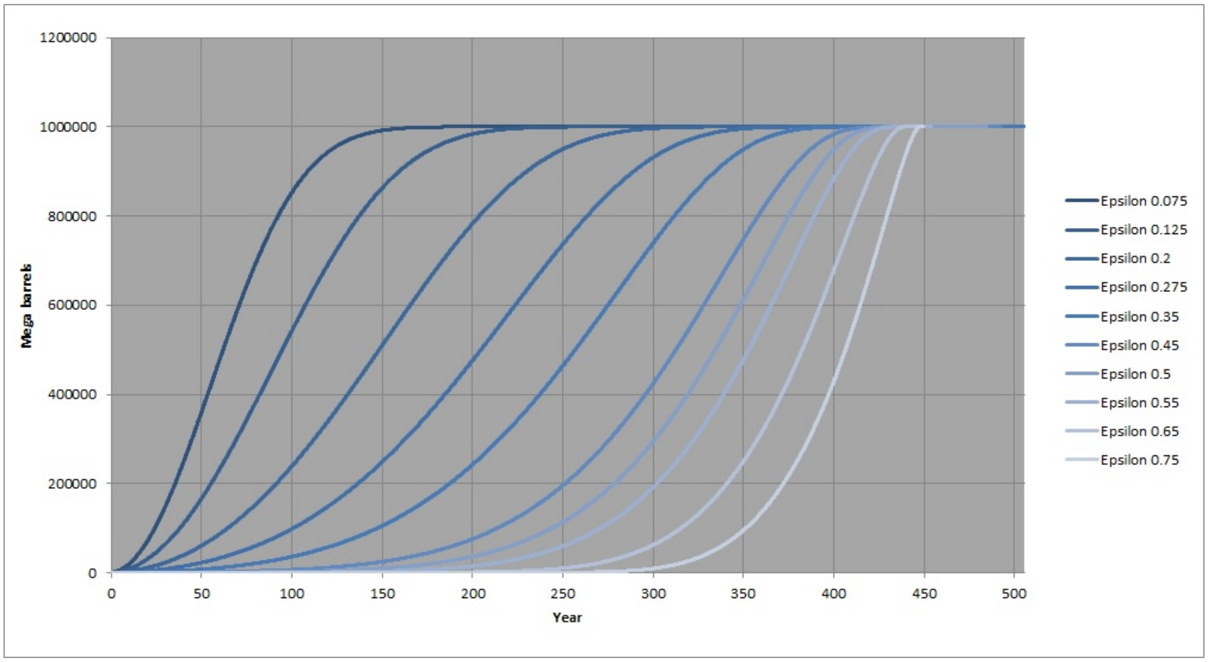

Figure 4.2: ϵ test: fossil fuel cumulative production (Q)

entire range, and the second plot (Figure 4.4) zooms in and shows a detailed view of the early variations.

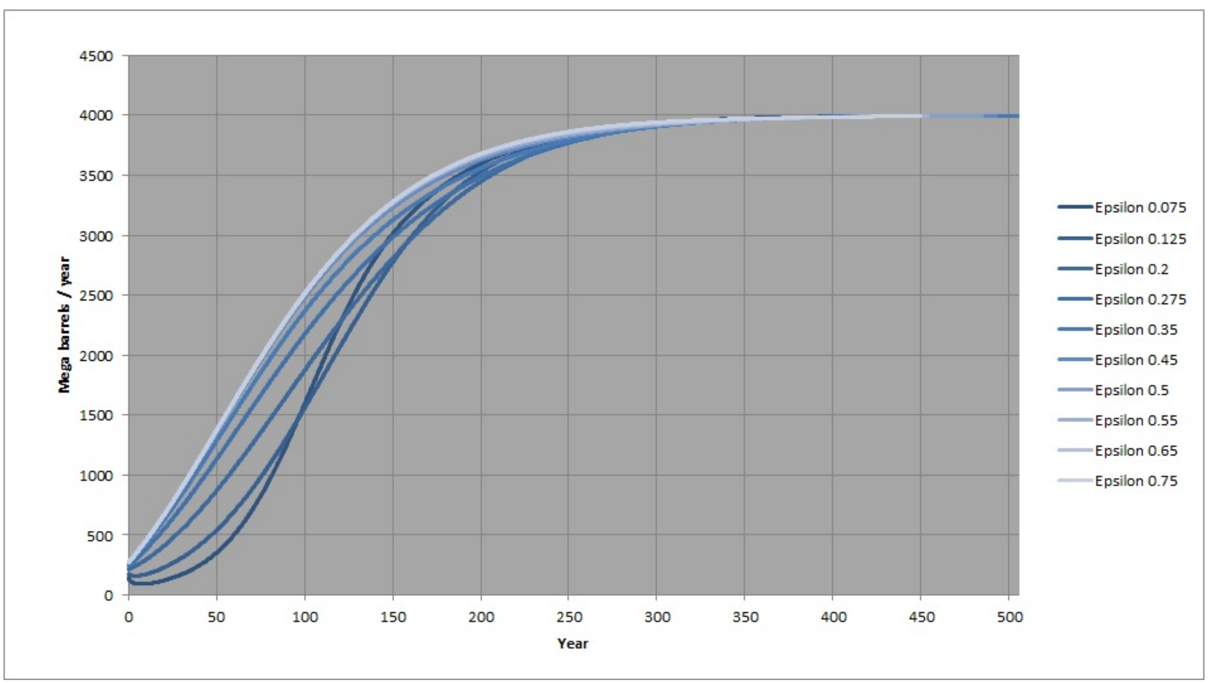

Figure 4.3: ϵ test: renewable energy production (E_r)

In the Figure 4.4, the path for $\epsilon = 0.075$ is marked in red to highlight that it crosses the other paths. Although the initial demand for renewable energy is lower due to the abundant availability of fossil fuel energy, since the fossil fuel reserves are depleted much sooner now, the demand for renewable energy ramps up quickly. This is opposed to the case for $\epsilon = 0.75$ where the fossil fuel energy production is quite small early on, adding demand for renewable energy. This earlier demand for renewable energy is dampened as time goes on and fossil fuel energy production increases. Since there is a cap for renewable energy production in our model, all curves for renewable energy production converge to the same long-term value.

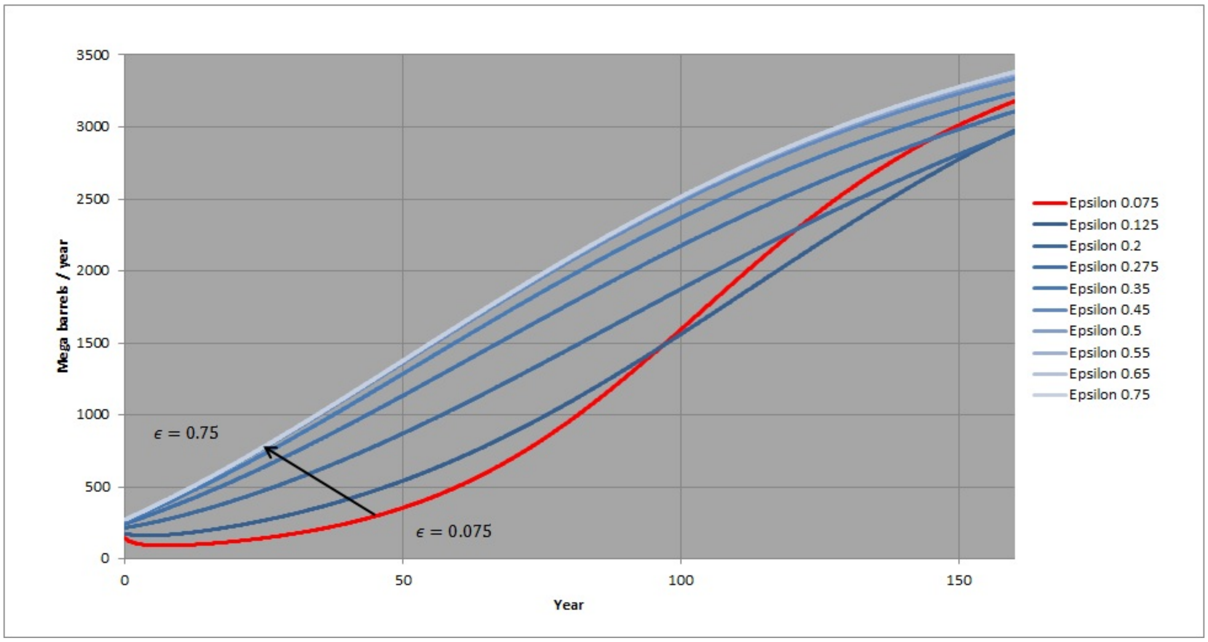

Figure 4.4: ϵ test: renewable energy production (detail)

Another way to see the effects on renewable energy demand is to examine the changes in renewable energy investment as we vary ϵ. This plot is shown in Figure 4.5.

For the mid-range values of ϵ, the increase in investment is relatively steady growth whereas for values in the lower range, we observe lower initial investment and a very rapid increase in investment as fossil fuel reserves are depleted. Also for higher values of ϵ, a double peak in renewables investment occurs. The investment initially grows but as

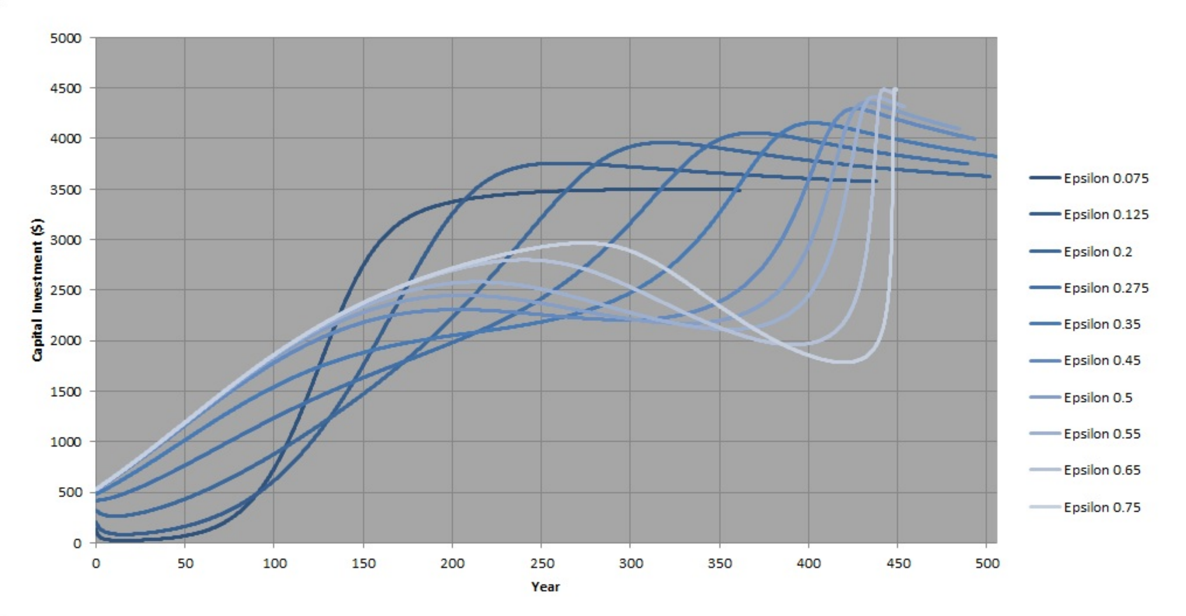

Figure 4.5: ϵ test: investment in renewable energy (K_r)

fossil fuel production ramps up, investment in renewables becomes tempered and enters a stage of decline until the reserves have been depleted. At this point, investment returns to very rapid growth in order to catch up with demand for renewable energy.

Finally, we look at how all of this affects the resulting price of energy. Due to the large growth in the price of energy over the 500 year time-frame, this plot is presented on a log scale.

In Figure 4.6 it is clear that the mid-range values are able to keep the price of energy slightly below exponential growth for some time compared to the cases where ϵ is higher or lower than the mid-range (0.2 to 0.35). As well the depletion of fossil fuel reserves corresponds to a rapid growth in the price of energy as it realigns with the exponential growth. The long term exponential growth of energy price is somewhat intuitive due to the fixed rate of production and an exponential increase in population. In Figure 4.6, the solutions eventually all return to an exponential growth set by the maximum renewable energy production.

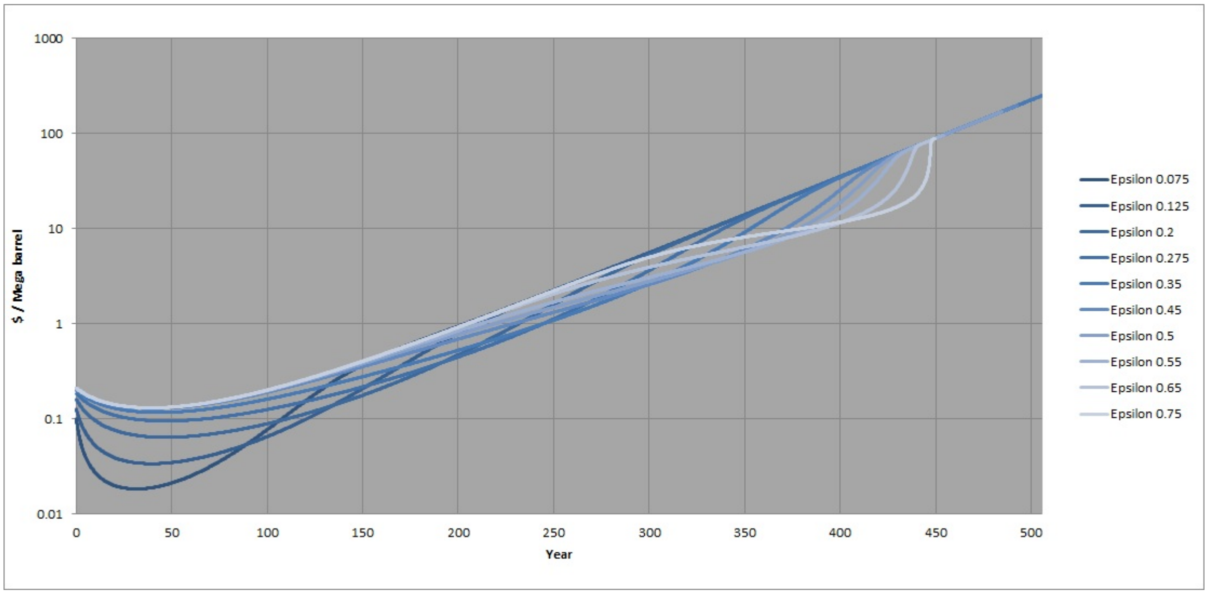

Figure 4.6: ϵ test: price of energy (p_E)

4.1.2 Renewable energy capital investment factor

For this test, all parameters have been kept the same as for the previous test (see Table 4.1), except that ϵ has now been fixed at 0.275, corresponding to the case of tempered growth and decline of fossil fuel production. Let us recall our equation for renewable energy production,

$$E_r = R_r \frac{\gamma\left(t\right) {K_r}^{\phi}}{\gamma\left(t\right) {K_r}^{\phi} + K_0^{\phi}}. \tag{4.2}$$

In this relation, ϕ represents a method for controlling the relative importance of capital investment in renewable energy production. Adjustments in the value of ϕ can produce dramatic effects in the production and investment in renewable energy, as well as the growth rate for innovation and the price of energy.

We begin by presenting the renewable energy production (Figure 4.7) which shows some interesting behaviour. For values of ϕ near zero the energy production is very low initially but experiences a dramatic rise within the first hundred years. As ϕ is increased the production curve flattens out resulting in much lower production during the simulation period. This trend continues until ϕ reaches the range around 0.1. Beyond this level we see that production follows an S-shaped path where the rate of production growth is lessened in the early years and following an inflection point begins to rise at a much faster rate towards the maximum production level. As ϕ is increased further towards 1 the S-shaped curve displayed by the production profile shifts to the left causing the initial level of renewable energy production to be much closer to the maximum. To get an understanding of why this behaviour is observed we show the plots for capital investment and innovation in (Figure 4.8) and (Figure 4.9). Both have been shown on a log scale.

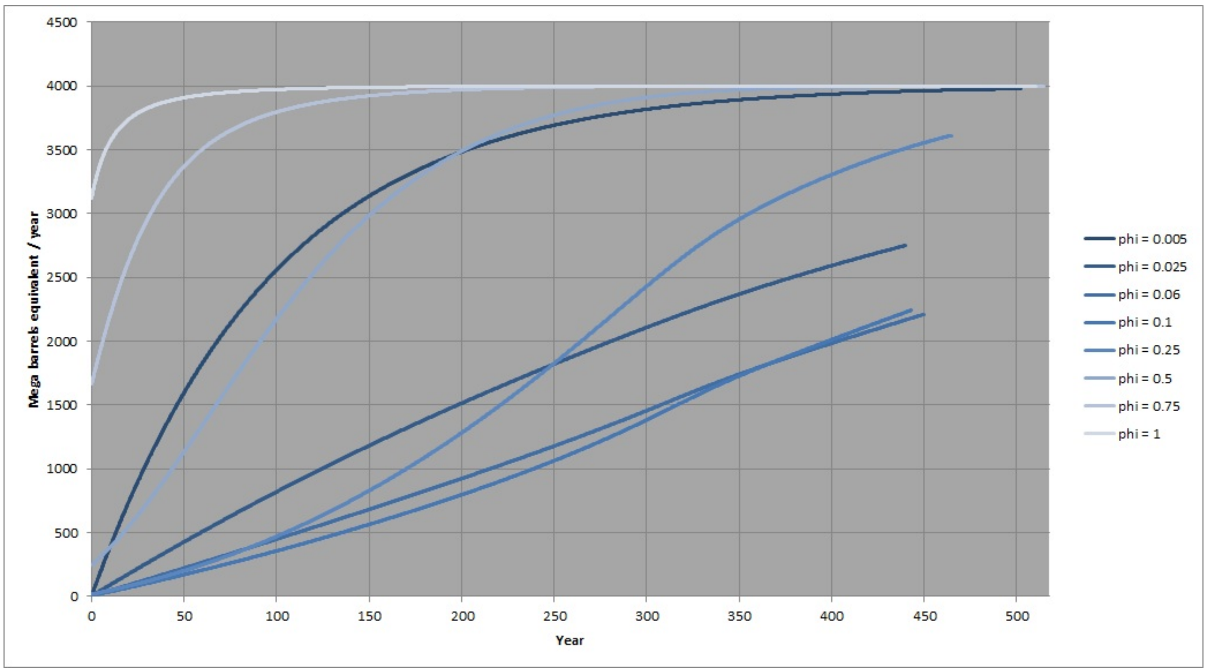

Figure 4.7: ϕ test: renewable energy production (E_r)

By observing the paths given for capital investment we can see that as ϕ is increased the initial level of capital investment is much higher. Initially as ϕ moves away from zero the growth rate for investment capital is positive and increases with ϕ. However as ϕ is increased further and the initial level of investment is raised, the required investment in future years is decreased such that for $\phi = 0.75$ and $\phi = 1$ the growth rate for capital

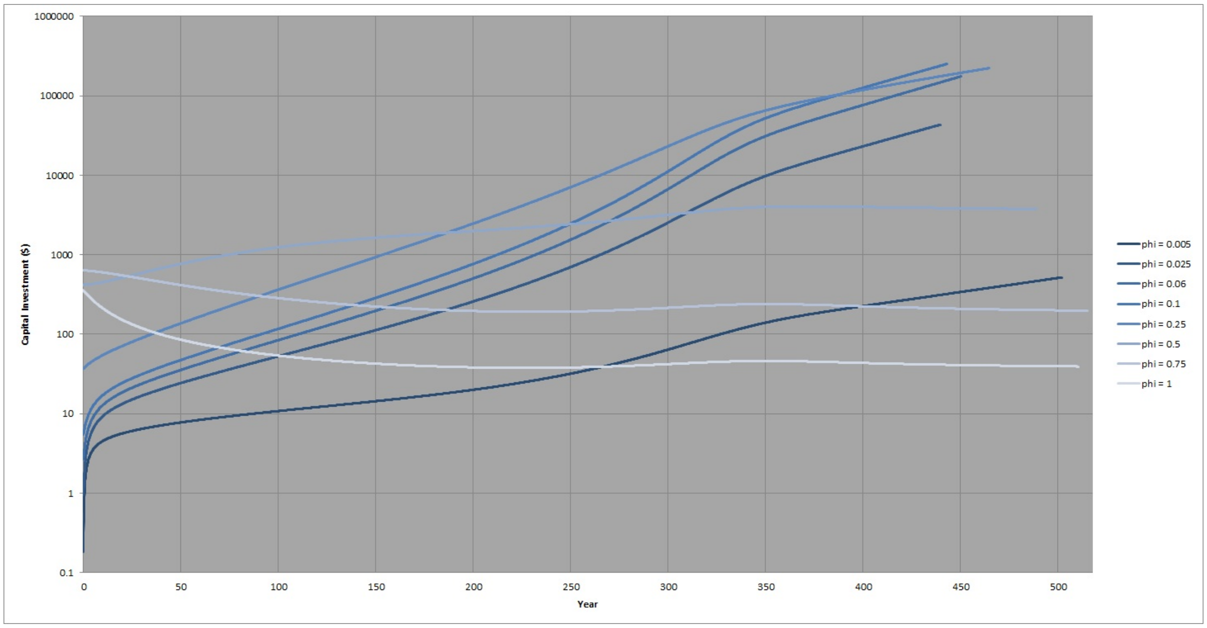

Figure 4.8: ϕ test: investment in renewable energy production (K_r)

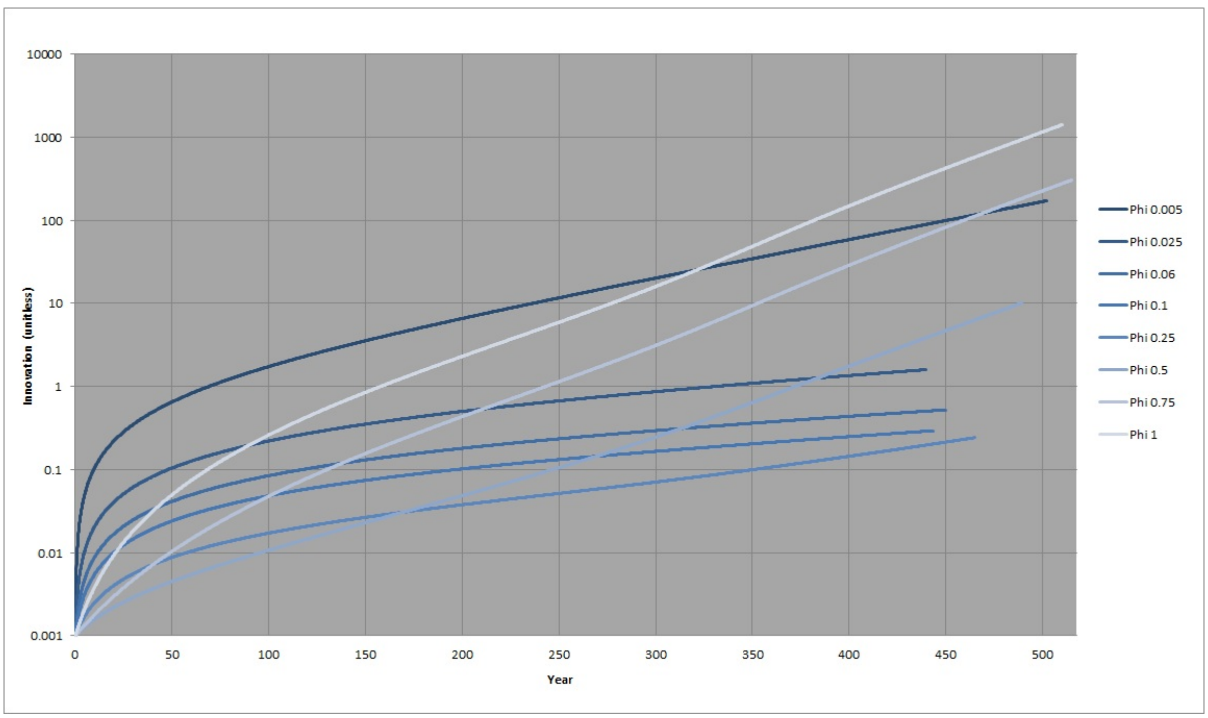

Figure 4.9: ϕ test: innovation for renewable energy production (γ)

investment is negative. The opposite of this pattern is observed for innovation (Figure 4.9). In this case we see that for low values of ϕ the innovation growth rate is initially very large but quickly tempers into a state of slow steady growth. As ϕ is increased the growth rate initially is decreased but the steady growth state corresponds to a much faster rate of innovation. This helps to explain the production paths shown in (Figure 4.7). For small values of ϕ the rapid rise in innovation spurs production increases in renewable energy. The relatively low levels of capital investment coupled with the slower long term growth in innovation help to explain the ever decreasing growth rate for renewable energy production. For high values of ϕ the initially large volume of investment capital boost the initial rate of production, and the decreasing capital over time is compensated by the much faster growth rate for innovation. For moderate values of ϕ we see the effects of lower initial capital and a slower rate of innovation growth working together and keeping the production level for renewables relatively low.

We can also see (Figure 4.10) that despite the dramatic changes to renewable energy production as ϕ is varied, the production of energy from fossil fuels remains quite stable.

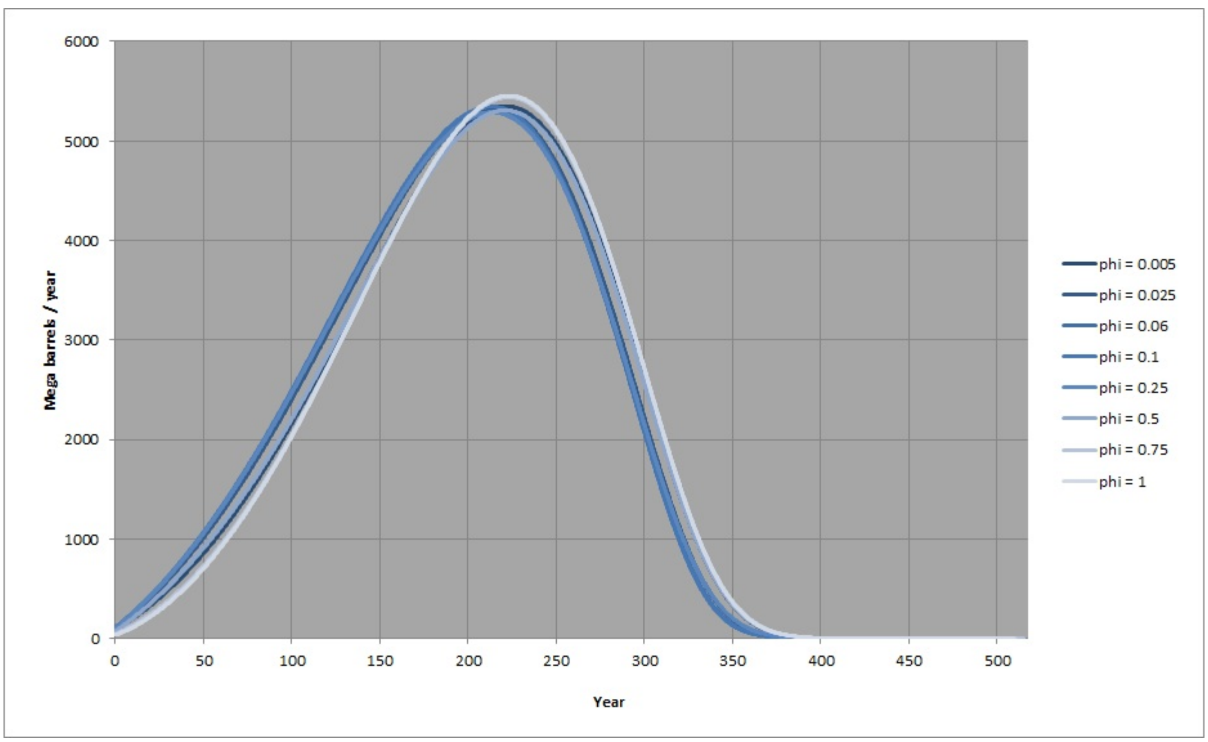

Figure 4.10: ϕ test: fossil fuel energy production (E_f)

As a result the long term trends for the price of energy are very similar, however the dramatic differences in the level of renewable energy production during the early years does cause a much lower relative price for high values of ϕ. The price of energy plots are provided in (Figure 4.11)

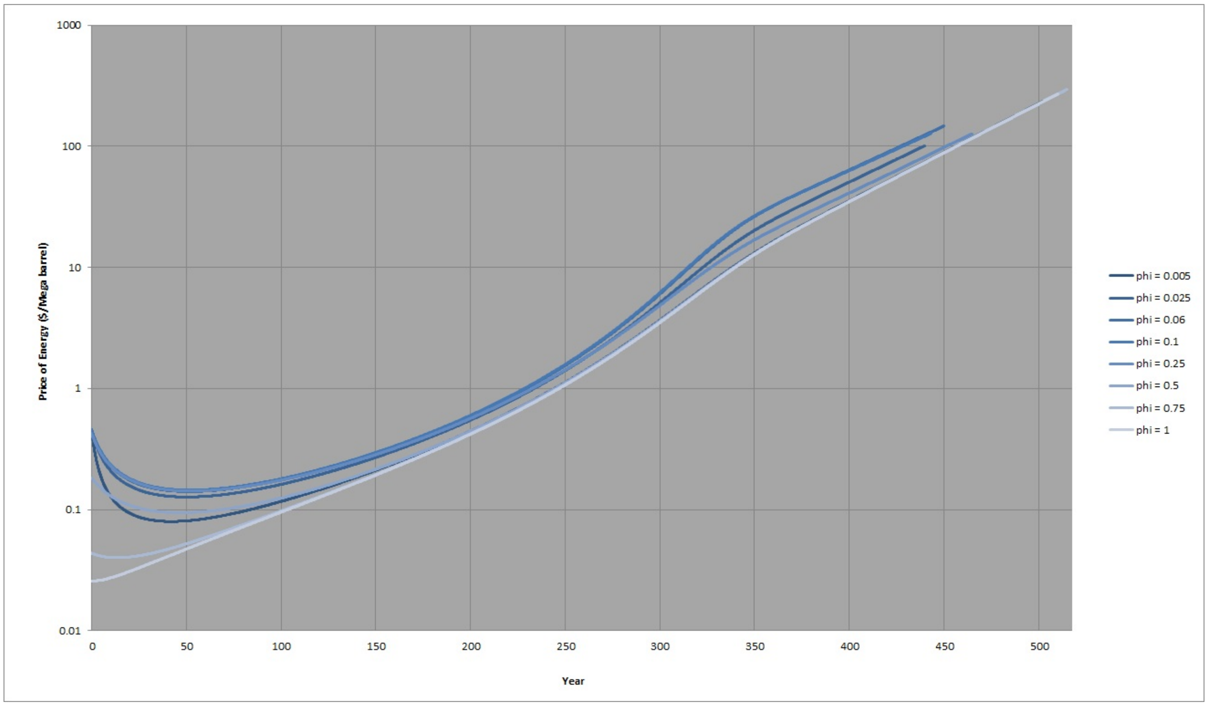

Figure 4.11: ϕ test: price of energy (p_E)

4.1.3 Innovation in renewable energy production

In this section, we present a few tests that were performed to investigate how innovation is affected by the variation in growth rate parameters τ_r and τ_{rf}. For these tests, we use the same parameters as in the above two tests (Table 4.1), however ϵ is set to 0.275 and ϕ is set to 0.5. From the previous section the case of $\phi = 0.5$ corresponds to the case of fairly flat investment growth and nearly constant exponential growth of innovation. Let us recall our ODE for innovation growth,

$$\frac{d\gamma}{dt} = \tau_r \frac{P_r}{K_r} + \tau_{rf} \frac{P_f}{K_r}. \tag{4.3}$$

We first examine the effects of including the second term which works to increase renewables innovation in relation to fossil fuel production profits. We compare three cases: *i*) the second term set to zero, *ii*) the second term is proportional to the first component, and finally, *iii*) the second term has a larger proportion than the first term. Our first plot shows the effects on the level of innovation (Figure 4.12). It is plotted on a log scale. As is expected, the larger the value of τ_{rf}, the faster the rate of innovation growth. The case of equal growth parameters results in quite steady exponential growth, for the case of a stronger effect from the fossil fuel contribution the rate of innovation growth is boosted during the years of strong fossil fuel production, the slowing of the growth rate at around 400 years can be attributed to the depletion of fossil fuel reserves, and thus the contribution to innovation.

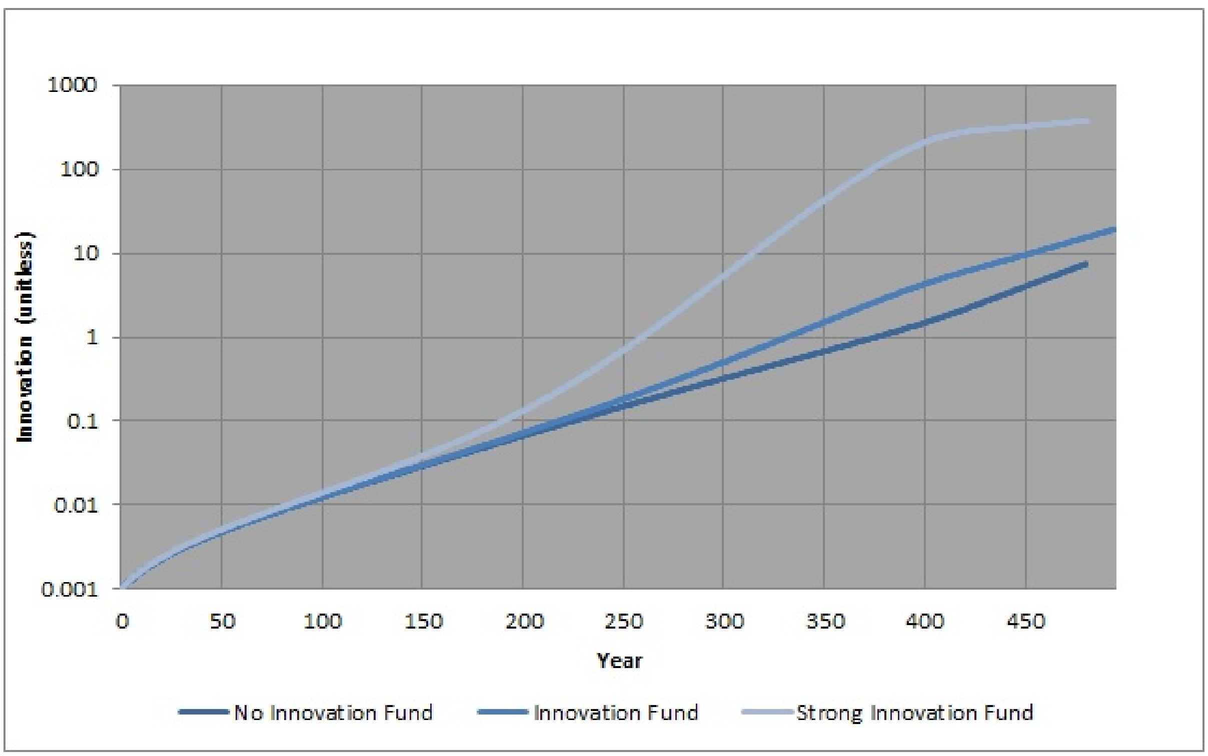

Figure 4.12: τ_{rf} test: innovation for renewable energy production (γ)

If we now turn our focus to the effects on investment shown in (Figure 4.13) we can see that the higher levels of innovation correspond to decreased need for capital investment, particularly in the later years beyond 100. This shows a balancing between the level of capital investment and the current level of innovation. This balancing can be seen quite clearly if we look at the renewable energy production paths in (Figure 4.14). Here we see that there is nearly no change to the production as τ_{rf} is varied, and thus the changes in investment must offset the changes in innovation.

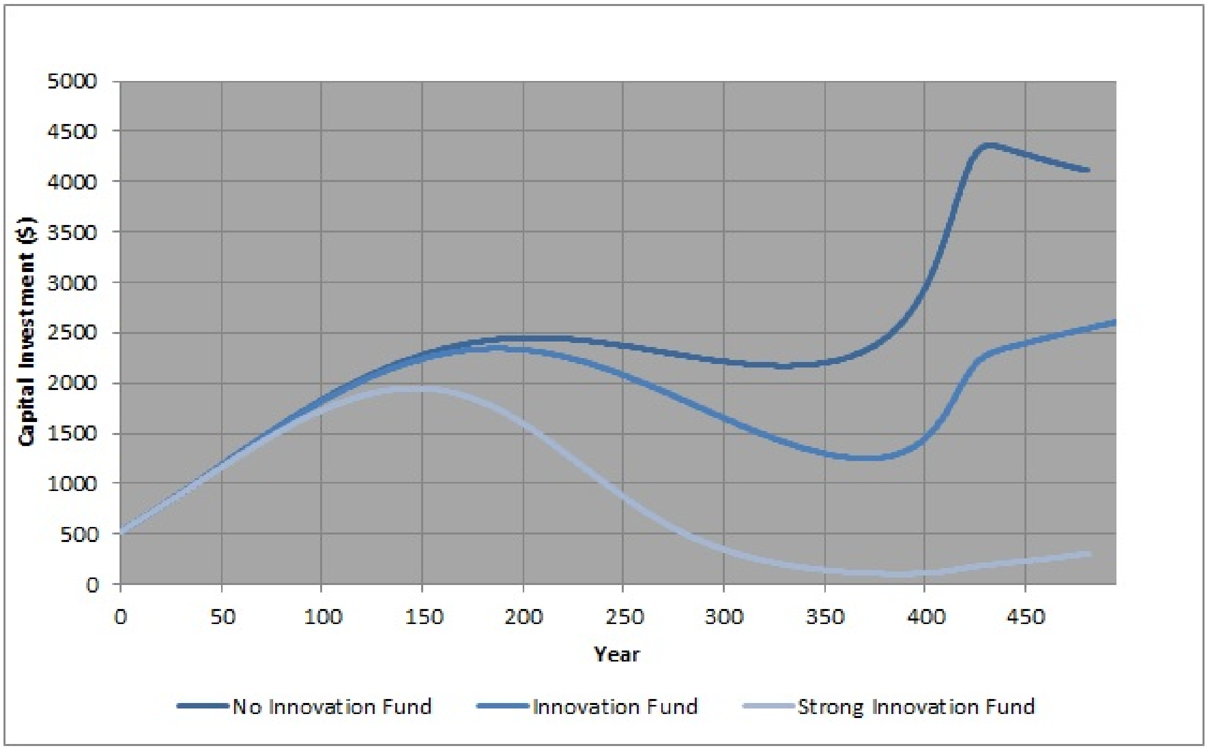

Figure 4.13: τ_{rf} test: investment in renewable energy production (K_r)

Since the change in energy production is quite minimal the resulting change in the price of energy is also negligible (Figure 4.15). This is an interesting result as it means that at least in our model the economy cannot use fossil fuel profits to innovate its way out of a downturn in total energy production and that increases in innovation generated in this way primarily work to decrease the need for investment capital.

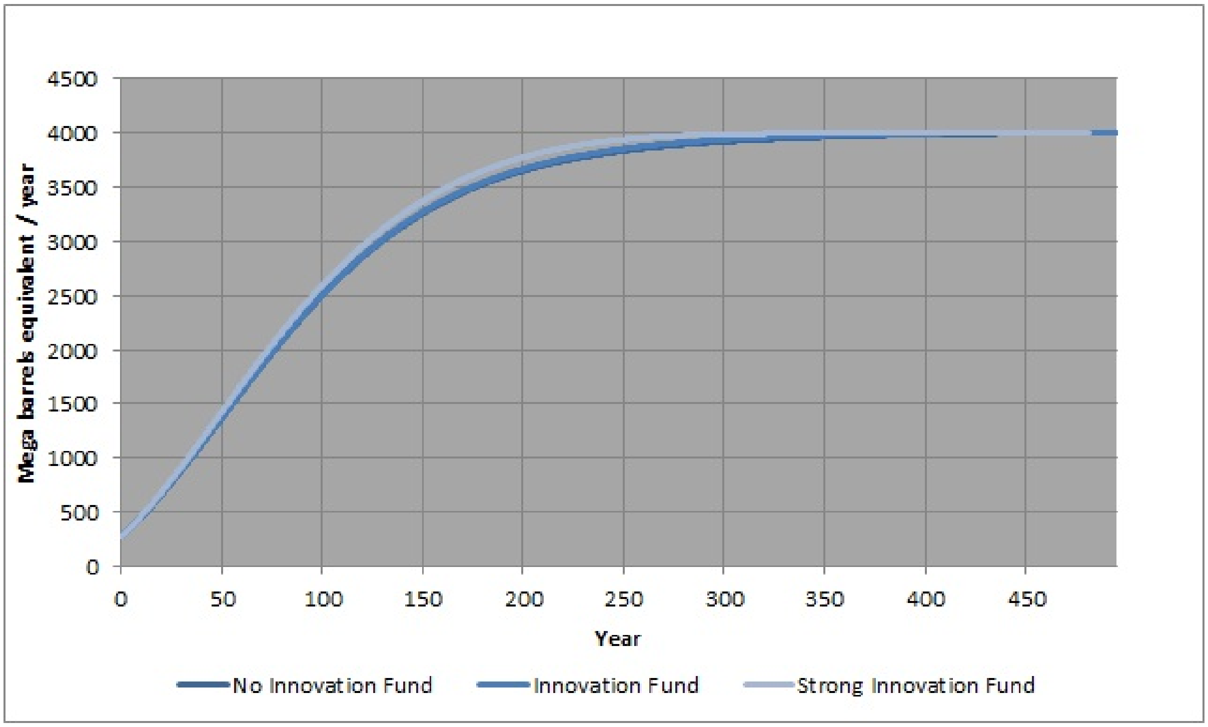

Figure 4.14: τ_{rf} test: renewable energy production (E_r)

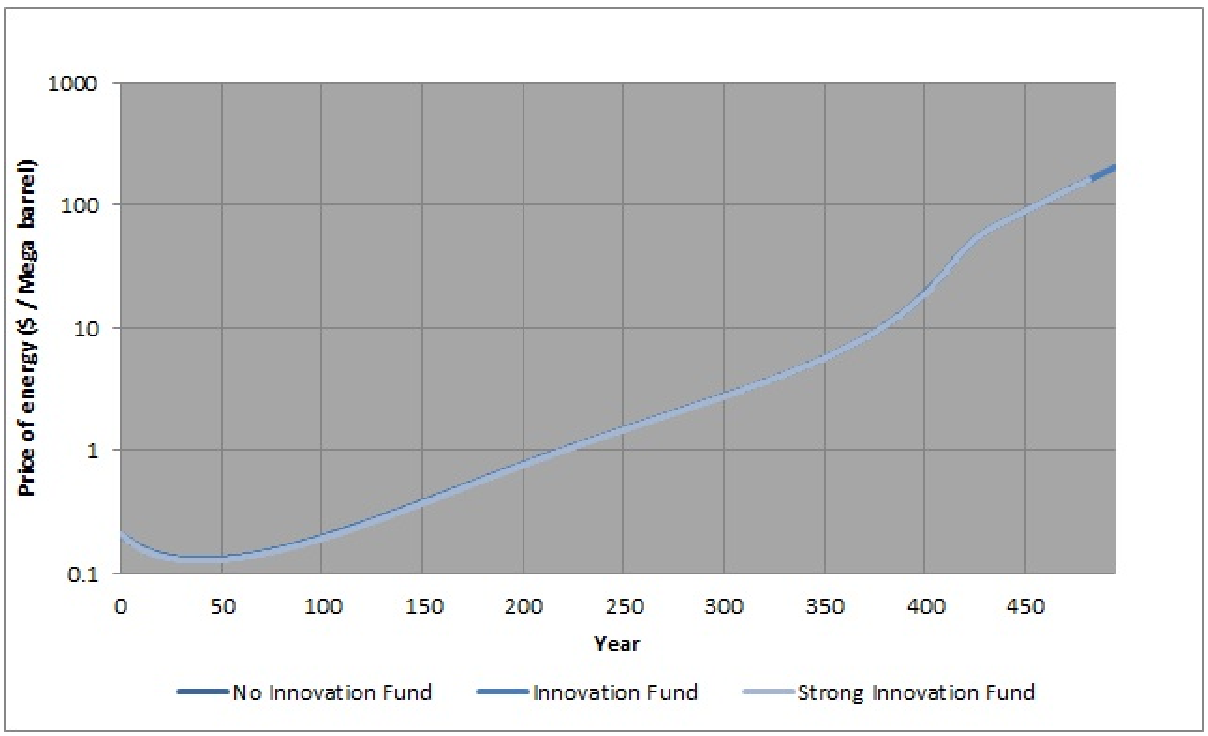

Figure 4.15: τ_{rf} test: price of energy (p_E)

In this section, we also examine the effect of changing τ_r. This parameter controls the growth rate for innovation in renewable energy. For larger values of τ_r the production limit for renewable energy is reached much sooner in the simulation. For smaller values the growth rate of renewable energy production is slower and hence the production limit is reached at a later date. To show the effects, we created a plot of total energy production combining the energy from renewables with that produced from fossil fuels. The goal of this test was to examine the possibility of creating a monotone transition from one source of energy production to the other. That is, without a drop in total production at any point.

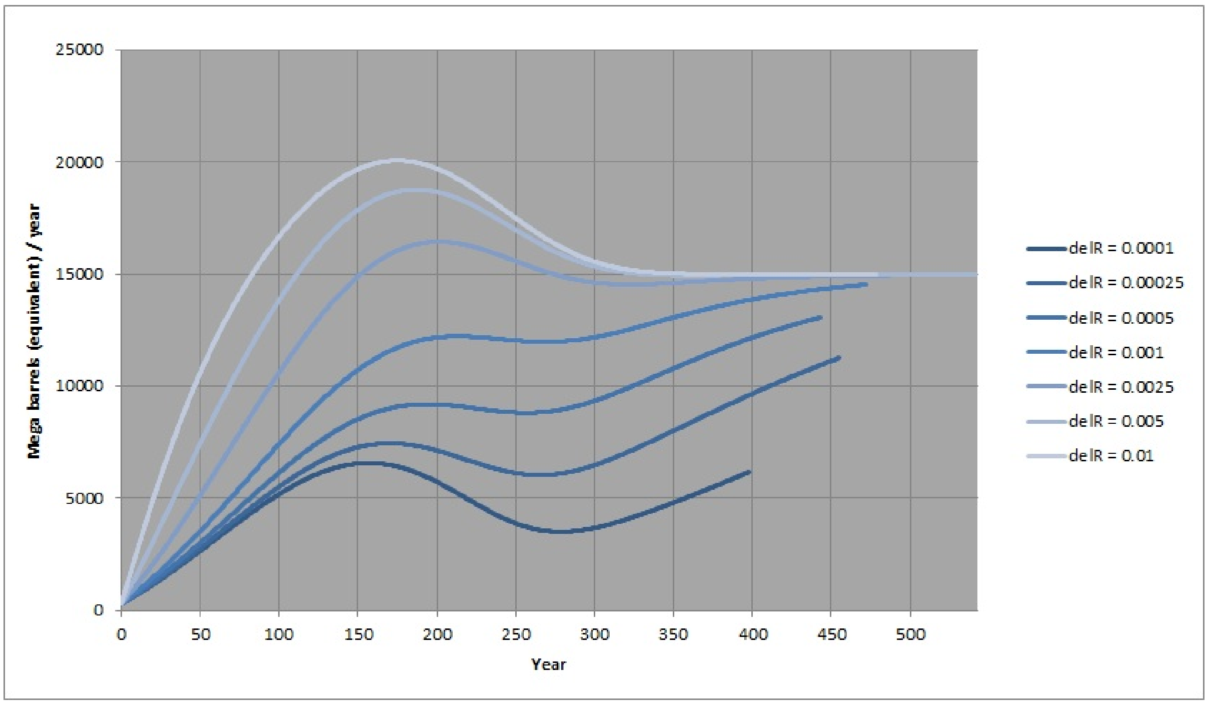

Figure 4.16: τ_r test: total energy production (E)

From Figure 4.16, it does seem possible to eliminate the downturn in energy production during the transition if the correct level of innovation growth is used. A growth rate too rapid results in an intermediate peak as renewable energy production reaches the maximum, while total energy rises well above the long-term production level, then declines as fossil fuel reserves are depleted and total energy production returns to the production limit from renewable sources. In contrast a growth rate too slow produces a case where the growth rate from renewable energy production is insufficient to counteract the decrease in fossil fuels production. In both cases the economy experiences a prolonged period of negative energy production growth. This is an interesting result when compared with the effects of adjusting the contribution by fossil fuel profits where the renewable energy production path was unchanged, while adjusting the contribution of renewable energy profits does result in a change in the production curve.

4.1.4 Renewable energy production cap

Below, we present the resulting effects on energy production by the variation of the maximum cap on renewable energy production.

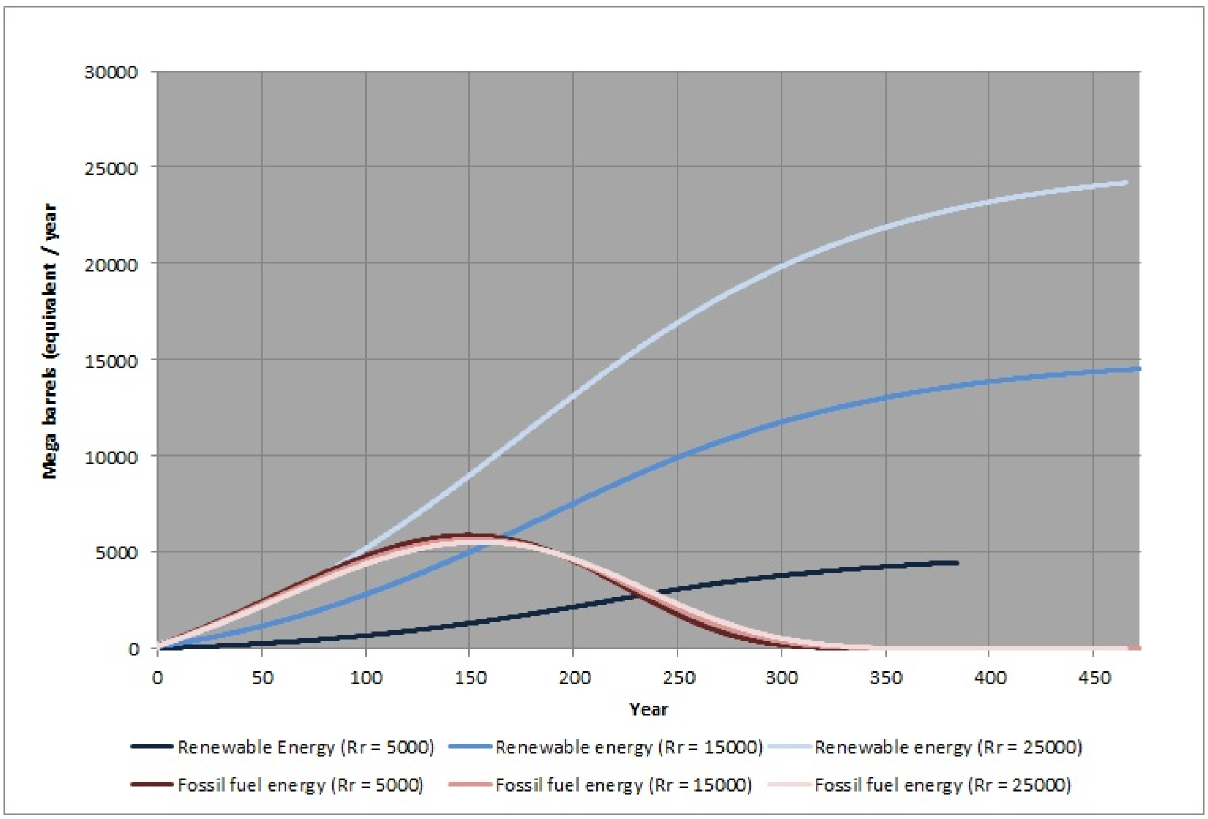

Figure 4.17: R_r test: fossil fuel and renewable energy production (E_r and E_f)

It is obvious that increasing the limit placed on renewable energy production capacity lifts the curve to match the new, higher level of production. The rate of increase in energy production capacity also increases as the limit is raised. It is of note that the qualitative features of the renewable energy production paths are unchanged by the increased cap on maximum production. In Figure 4.18, the resulting total energy curve is also provided. We can see from this that for an economy with a higher production cap on renewable energy it is possible that the transition between fossil fuel and renewable energy as the primary source of energy will not generate a downturn in total energy supply.

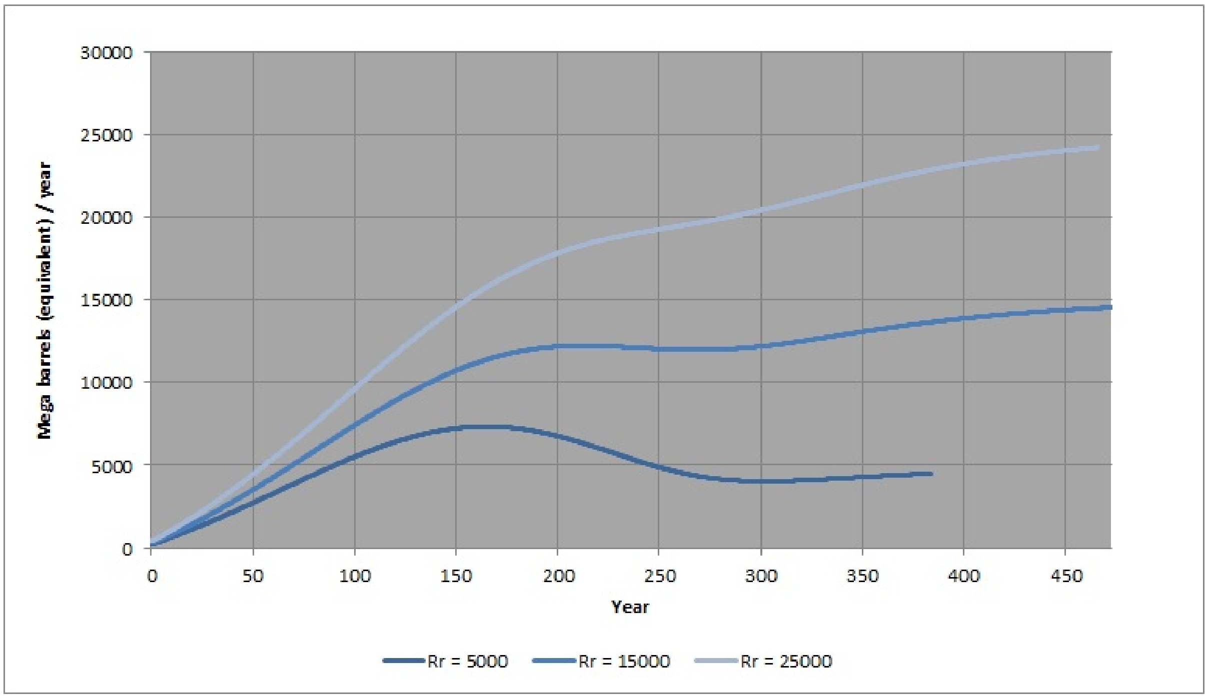

Figure 4.18: R_r test: total energy production (E)

4.1.5 Interest rate sensitivity

In this section, the effect observed from varying the rent on capital for energy producers is examined briefly. First, the price of energy is plotted on a log scale as r_E is varied from 0.1% to 10% (Figure 4.19).

Note that for interest rates above 1%, the price of energy begins in a state of decline which eventually gives way to rising energy prices later on. The long run trend is towards

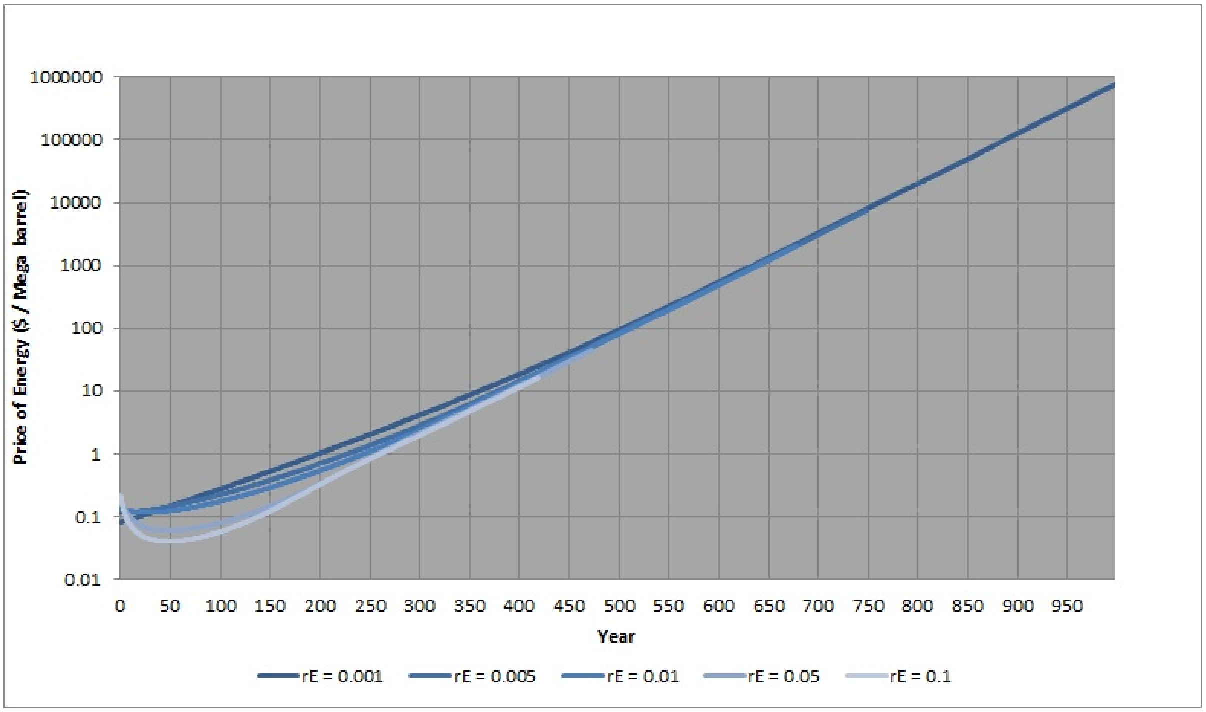

Figure 4.19: r_E test: price of energy (p_E)

exponential growth of energy prices, as we have seen in the previous test, and all curves eventually return to the same level. The plots for total energy are given in Figure 4.20, innovation in Figure 4.21 and investment in renewable energy in Figure 4.22.

As the interest rate is lowered, the total energy production curve experiences slower growth and also exhibits monotone growth in the transition from fossil fuel energy to renewable energy production. From the plots for investment and innovation we can see that high interest rates on borrowed capital reduce the investment capital and thus put growth pressure on innovation. Innovation is thus highest and grows the fastest in high interest environment, and is lower for low interest. Under the high interest rate environment as innovation decreases the capital investment required the price of energy also decreases and thus the price of energy experiences a period of declining energy prices over the first 50 years.

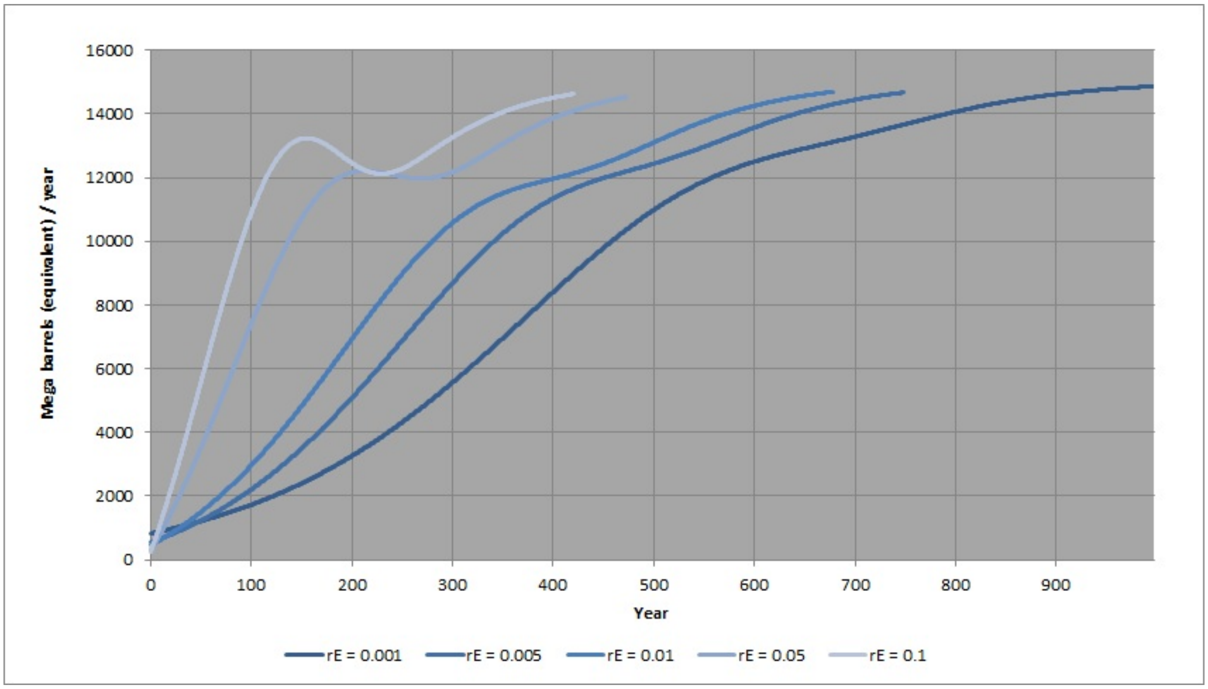

Figure 4.20: r_E test: total energy production (E)

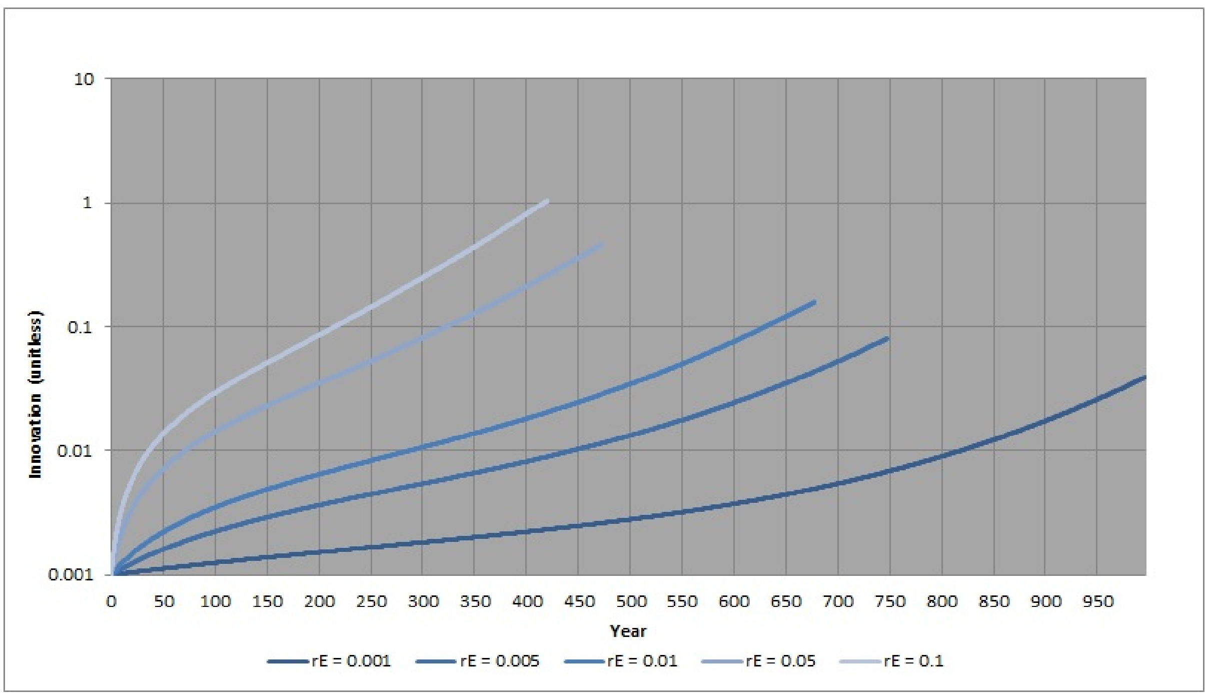

Figure 4.21: r_E test: innovation in renewable energy (γ)

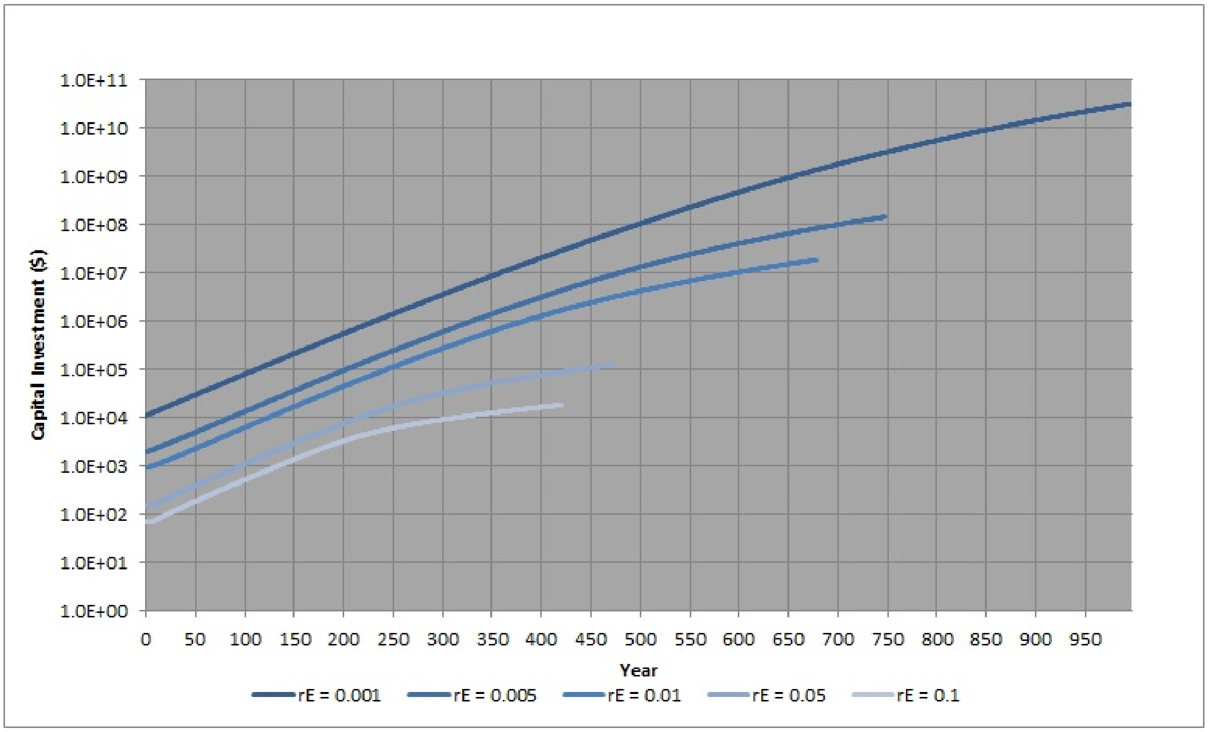

Figure 4.22: r_E test: investment in renewable energy (K_r)

4.2 Modelling real world data

In this section, the results obtained from attempting to fit the model to real world data are presented. The parameter selection is detailed first, followed by plots providing the results obtained in comparison to the real world data. The data used to fit the model has been gathered from two main sources. The first is the annual review of world energy produced by British Petroleum (BP 2011). It contains the energy production data for fossil fuels (including oil, natural gas, and coal) as well as for renewables which for the purposes of this thesis include hydro, nuclear, and other forms of renewable energy such as wind, solar, and geothermal. Although the production of nuclear energy is dependent on non-renewable resources, the analysis done by Hubbert (1956) [10] on the potential energy supply which can be produced from available nuclear fuels, gives us confidence in including this under our renewables. The second primary source of real world data comes from the World Bank. This data provides estimates for the world population and the global GDP values. Many of the model parameters used by Berg *et al.* in 2011 [2] are carried forward as approximate values. This includes labour productivity initial value and growth rate, energy efficiency initial value and growth rate, and the production function factors α and β. The remaining parameters, mostly new, were used to modify the model output so as to provide a simple fit to the real world data. The *fit* has not been performed using any standard mathematical method such as least squares, but has been done mearly by eye. The parameter testing done earlier provided a means to select the appropriate values for ϕ and ϵ and by incrementally adjusting the growth factors τ_r, τ_f, g_γ, g_C, along with renewable energy parameter K_0, the production capacity R_r, and the initial value for innovation γ until the model output tracked the real world data reasonably well (relative error $\leq 15\%$). Some of the parameters used by the model are particularly difficult if not impossible to measure from available real world data. This could be seen as a detriment to the model. At this point the estimates for model parameters are quite crude due to the data available. Increased data over a longer period of time and covering more areas, such as investment in the various forms of energy production could go a long way to improving confidence in the results. Due to the limited availability of data for renewable energy production, the real world data used for this test covers the time period 1980 to 2011. To determine an estimate for the fossil fuel reserves, the reserves of oil ($\sim$ 1.650 trillion barrels in 2011), natural gas ($\sim$ 1.350 trillion barrels equivalent in 2011), and coal ($\sim$ 4 trillion barrels equivalent in 2011) have been combined with the cumulative production up to 2011 to give the total fossil fuel reserves of approximately 9.7 trillion barrels of oil equivalent. From the available data, the cumulative fossil fuel consumption by 1980 is approximated as 700 billion barrels of oil equivalent. A list of parameters used for this test is provided in Table 4.2.

Since the bulk of the work performed on the model relates to energy production, the fit of our model with world production data for fossil fuels and renewables is of primary concern. The plots presented below in Figure 4.23 and Figure 4.24, contain the model

Name	Parameter	Value
Renewable energy production constant	K_0	25,000
Initial population	$L(0)$	4.5 billion people
Initial innovation	$\gamma(0)$	0.0095
Initial extraction efficiency	$C(0)$	0.002625
Initial labour productivity	$A(0)$	298,000
Initial energy efficiency	$B(0)$	273,000,000
Population growth rate	g_L	0.017
Labour productivity growth rate	g_A	0.03
Innovation growth rate	g_γ	0.025
Extraction efficiency growth rate	g_C	0.0325
Energy efficiency growth rate	g_B	0.015
Maximum renewable energy production	R_r	100,000 mega barrels/ year
Total fossil fuel reserves	R_f	9,700,000 mega barrels
Interest rates	r_E, r_K	0.05
Labour to capital-energy production parameter	α	0.3
Capital to energy share parameter	β	0.066
Capital to energy production parameter	ρ	0.55
Renewable energy production parameter	ϕ	0.1
Fossil fuel energy production parameter	ϵ	0.05
Innovation growth parameter	τ_r	0.001
Extraction efficiency growth parameter	τ_f	0.00011
DASSL tolerance	*tolerance*	10^{-5}

Table 4.2: Model settings for test runs

estimates and real world data for the energy production from each source (renewable or non-renewable energy). The second plot (Figure 4.24) provides a detailed view for the period up to 2040. The plots contain the two proposed solutions for innovation in renewables and extraction efficiency.

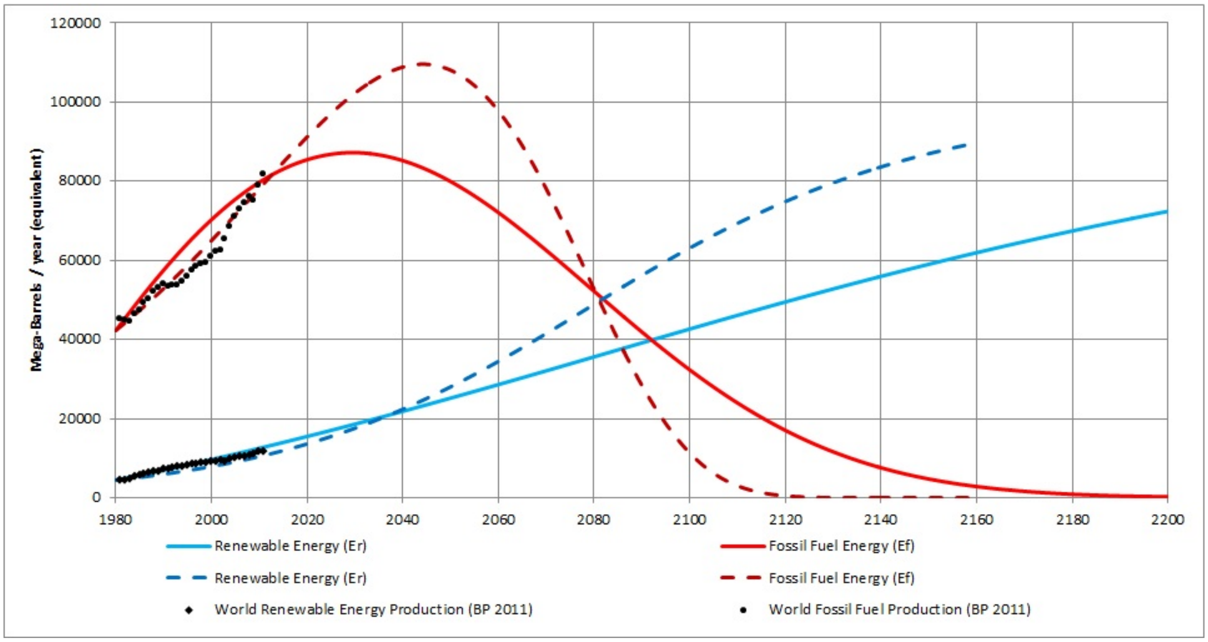

Figure 4.23: Real world test: fossil fuel and renewable energy production - ODE model (solid lines) and exponential model (dashed lines)

The model results fall nicely into the same order of magnitude and also match the growth rates reasonably well. The production curves for exponential growth of innovation and extraction efficiency (the dashed curves in Figure 4.23 and 4.24) show marginally slower initial growth in energy for the first 40 years, but overtake the more linear growth produced by the ODE solutions. By 2060, the exponential solutions are much higher. The higher peak production in fossil fuels corresponds to a much steeper decline and complete depletion of fossil fuels occurs over 40 years earlier than in the other scenarios.

One of the results noted by Berg *et al.* (2011) was that the price of energy for their model was over estimated by a factor of 10-50, the primary reason given, having only one source of energy production. With the addition of a renewable source of energy to

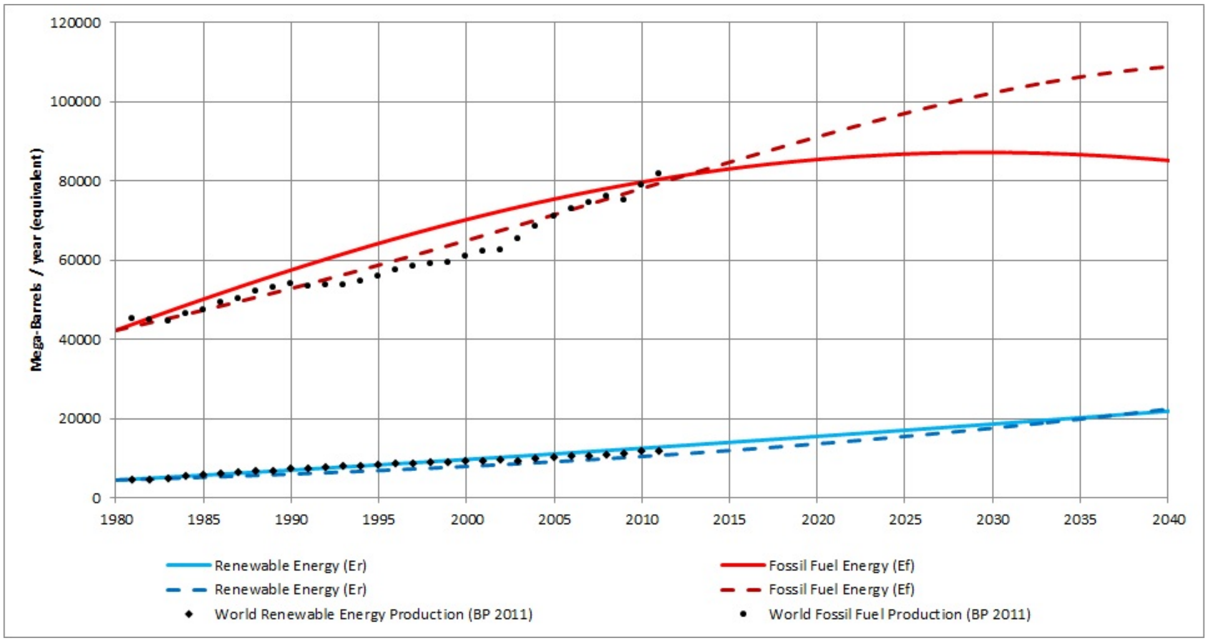

Figure 4.24: Real world test: fossil fuel and renewable energy production (detail)

the model presented in this thesis, it is expected that this result can be improved. In the plot below are the prices of energy predicted by the models as well as the real world cost of oil. The real world data gives an estimate of the spot price for crude oil (BP 2011).

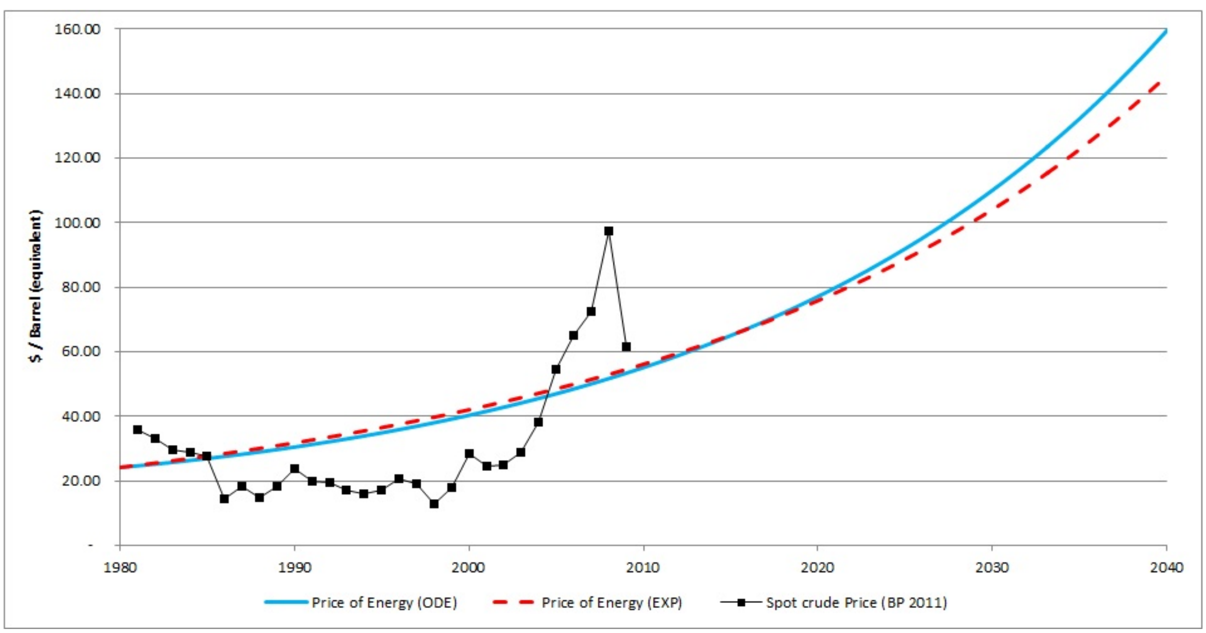

Figure 4.25: Real world test: price of energy

As the graph shows (see Figure 4.25), the model now produces energy prices within the right order of magnitude. This is a notable improvement and validates that the changes made to the model are producing more realistic results. The price of energy increases rapidly following the peak in fossil fuel production (shown earlier in the testing section). We can observe from (Figure 4.25) that the curve representing exponential growth generates a lower price of energy beyond 2020, corresponding to the higher fossil fuel production shown in (Figure 4.23). One obvious omission in the model is volatility in the price of energy. This is discussed further in the next section.

The model also predicts the economic output, or GDP, which we can compare to the real world data obtained from the World Bank (2011). The results from this comparison are given below, plotted on a log scale (Figure 4.26).

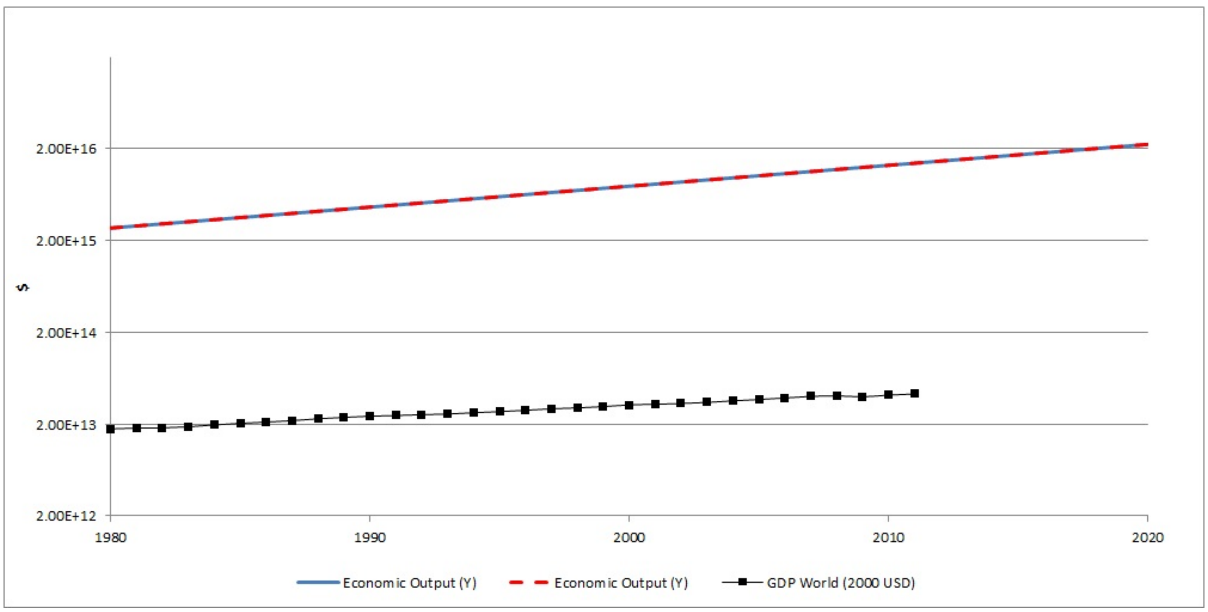

Figure 4.26: Real world test: economic output (GDP)

It is at this point that the results begin to deviate from the real world data. The GDP figures predicted by the model are too high by approximately two orders of magnitude. Certainly there are many elements which could be contributing to this overestimation. The CES production function work done by Kemfert [11] which provided us with the layout for our production function and also estimations of the substitution elasticities was performed on data restricted to the industrials sector in West Germany. One would expect capital and energy to be more freely substituted in an advanced industrialized economy than for the world as a whole and thus this could contribute to increased GDP. It is also possible that the substitutability of production factors may vary across different sectors of the economy and given that industrials only accounts for about 30% of global GDP [5]. The services sector, which accounts for over 60% of global GDP, may not provide the same substitutability of capital and energy as for the industrials. As well the model still has several functions affecting the economic output which are being approximated by simple exponential growth; labour, productivity, and energy efficiency. If the population figures used by the model are compared to the real world data, the discrepancy mentioned in the introduction become apparent. The plot below shows the exponential population growth produced by the model and the real world population growth which has more linear growth (Figure 4.27).

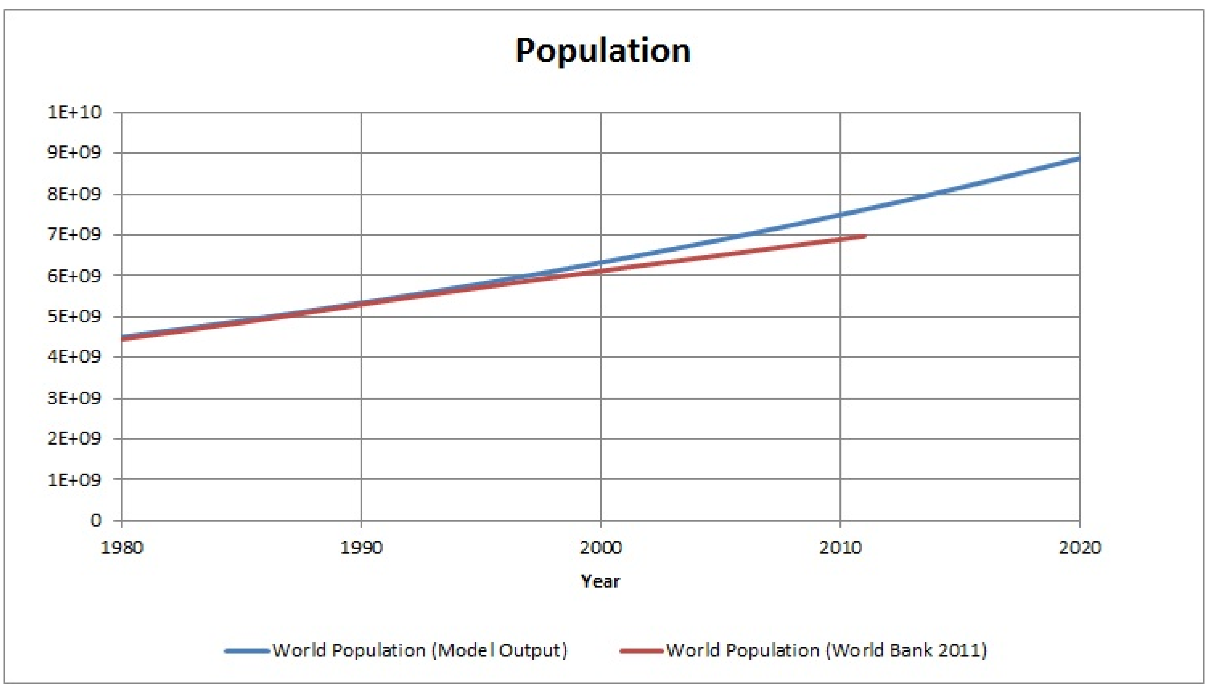

Figure 4.27: Real world test: population (available labour)

This discrepancy in the population is further exacerbated by the assumption of full employment, which takes the approximation farther away from the real population available for labour.

4.3 Solution stability and error

In this section, the model is tested to investigate the stability of the solution when longer time steps are taken. As well, the relative error between the solution produced by DASSL and the RK4 solution developed for this thesis is provided.

4.3.1 Time step sensitivity

For this section, the solutions are computed using time-steps of 0.08, 1, and 10 years, representing data points monthly, yearly, and for each decade, respectively. The first plot (Figure 4.28) given below is for the DASSL solution to the exponential model. The solutions provide an excellent match even for very long 10 year time steps. The peak is still located at the same year. The year of complete depletion does vary as the relative differences in fossil fuel production in the final few decades can become large due to the very small magnitude of production.

The plots produced by using the RK4 method developed earlier in this thesis performed fairly well (Figure 4.29). However, the 10 year time-step shows a noticeable difference in the peak production height for fossil fuels.

In the following two plots (Figures 4.30 and 4.31), the solutions from DASSL and the RK4 solvers are provided for the ODE model where the same monthly, yearly, and decade time-steps are used. Note that this model produces a lower peak production ($\sim$ 90,000 mega barrels / year) and a much longer depletion horizon ($\sim$ 400 years). The model is extremely stable for all time steps tested. The solution for the RK4 model is provided below (Figure 4.31), and it too is remarkably stable even up to the 10-year time-step.

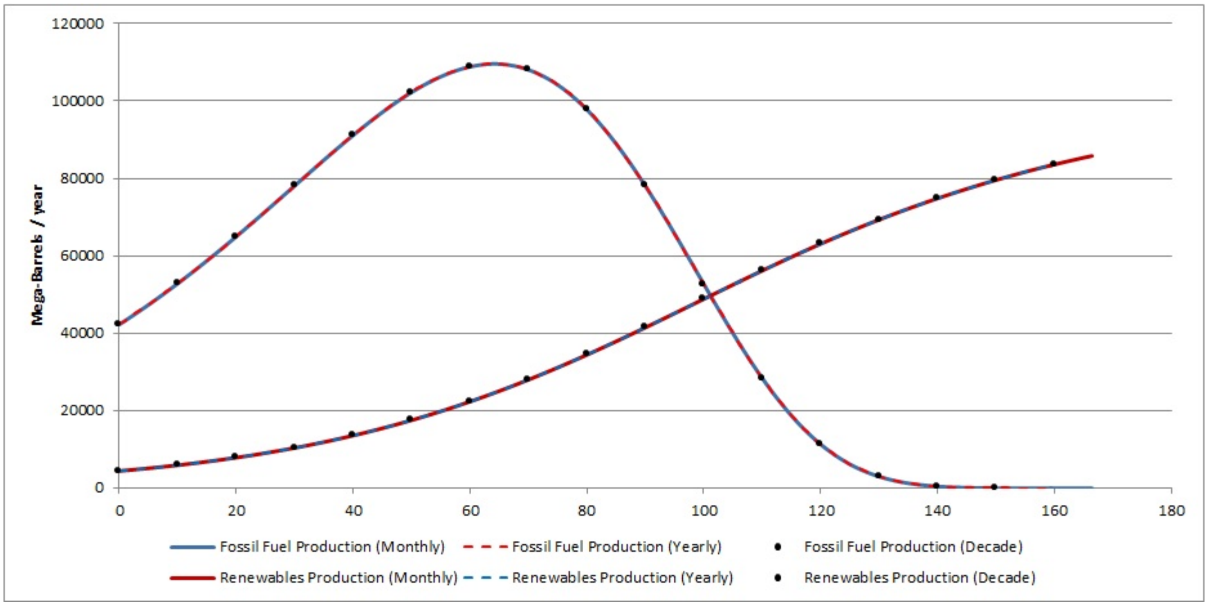

Figure 4.28: Time-step test: DASSL solution for energy production, exponential model

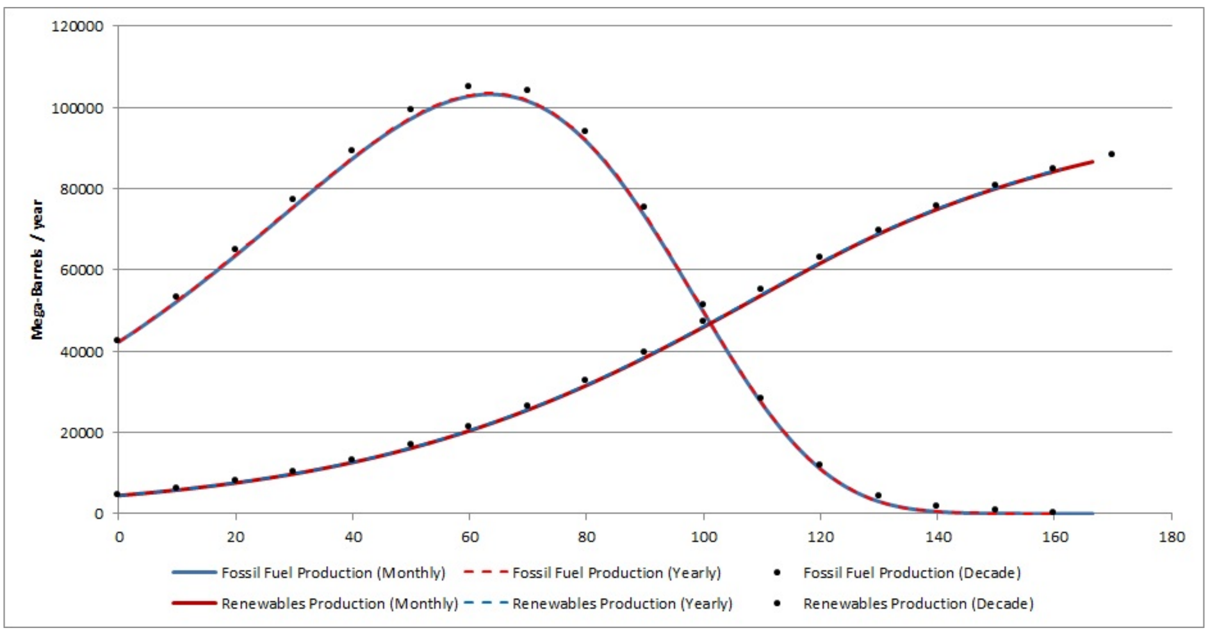

Figure 4.29: Time-step test: RK4 solution for energy production, exponential model

Although both solution methods were fairly stable for all step sizes considered, the implicit DASSL solution proved much better for very large time steps of the order of decades, than that provided by the explicit RK4-Newton's method which clearly over estimates production when the time step is extended to decades.

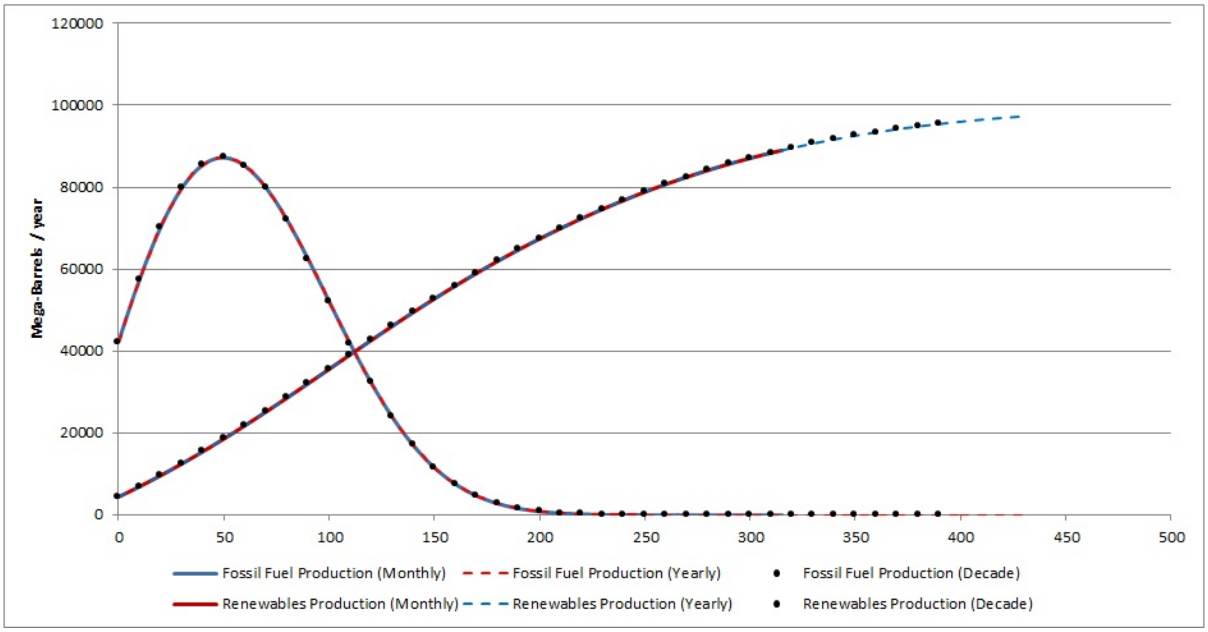

Figure 4.30: Time-step test: DASSL solution for energy production, ODE model

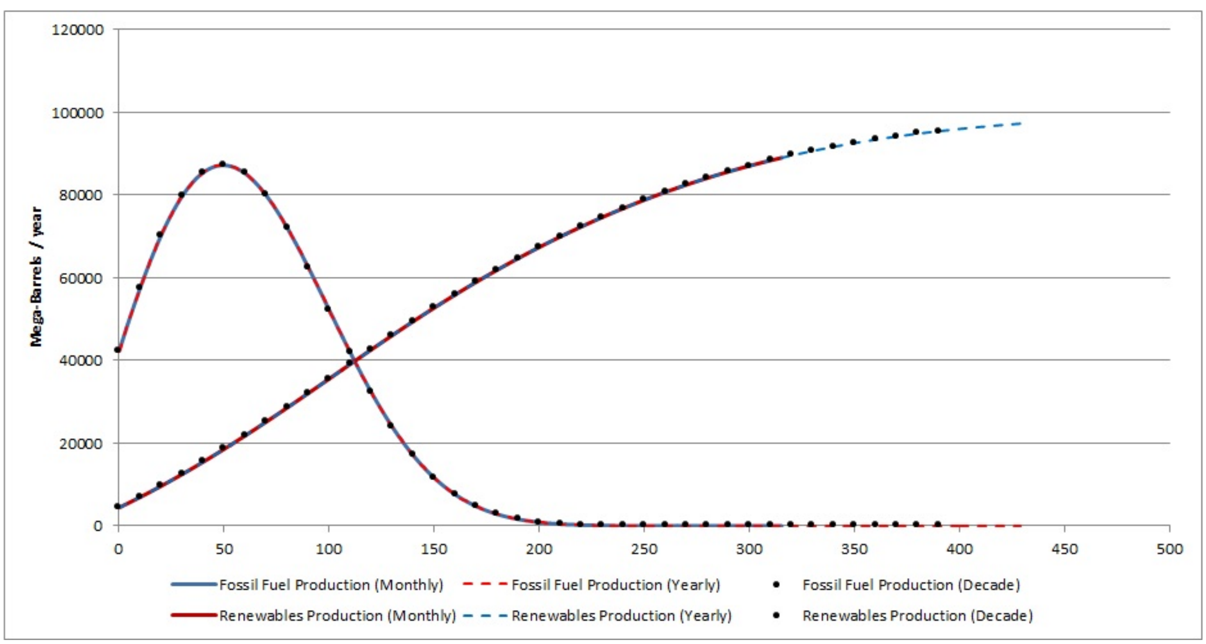

Figure 4.31: Time-step test: RK4 solution for energy production, ODE mode

4.3.2 DASSL vs RK4

In this section, the two solution methods are compared for each model. The energy plots including production and price are shown and the relative error of our developed RK4 solution is also plotted. First we show the energy production plots for the exponential model, the dashed lines for the RK4 solution and the solid lines for the DASSL solution.

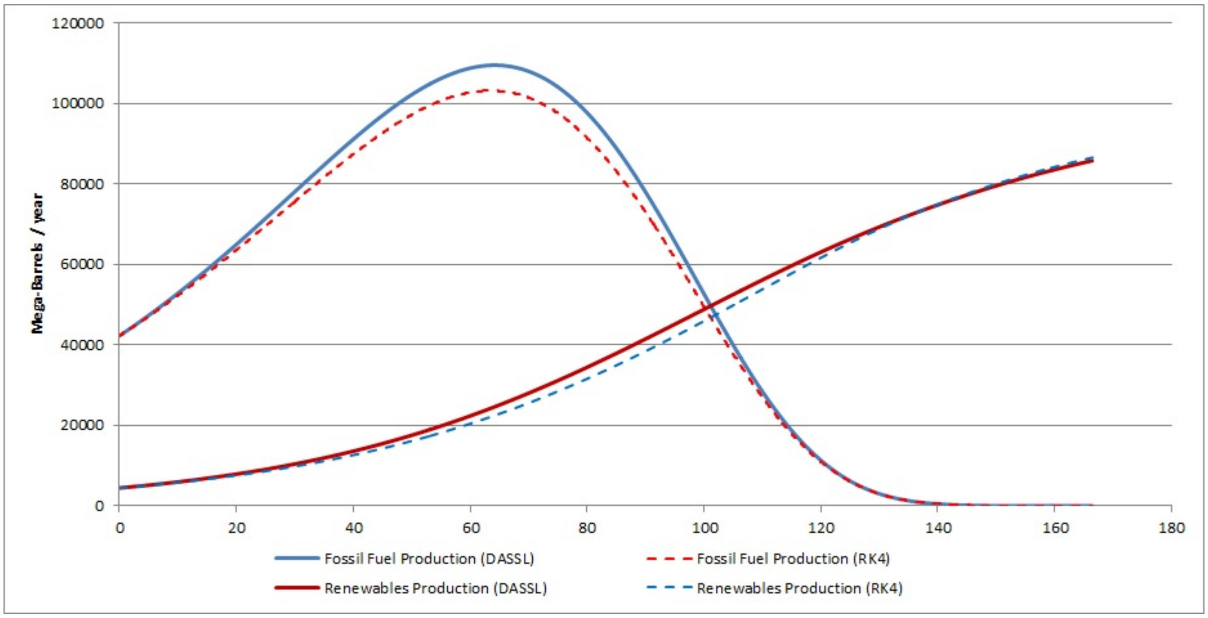

Figure 4.32: Error test: energy production

There is a noticeable difference at peak production. The error plots below show the extent of the differences. In particular, the error in the price of energy reaches approximately −70% about 60 years in and then swings to +90% in the following 100 years.

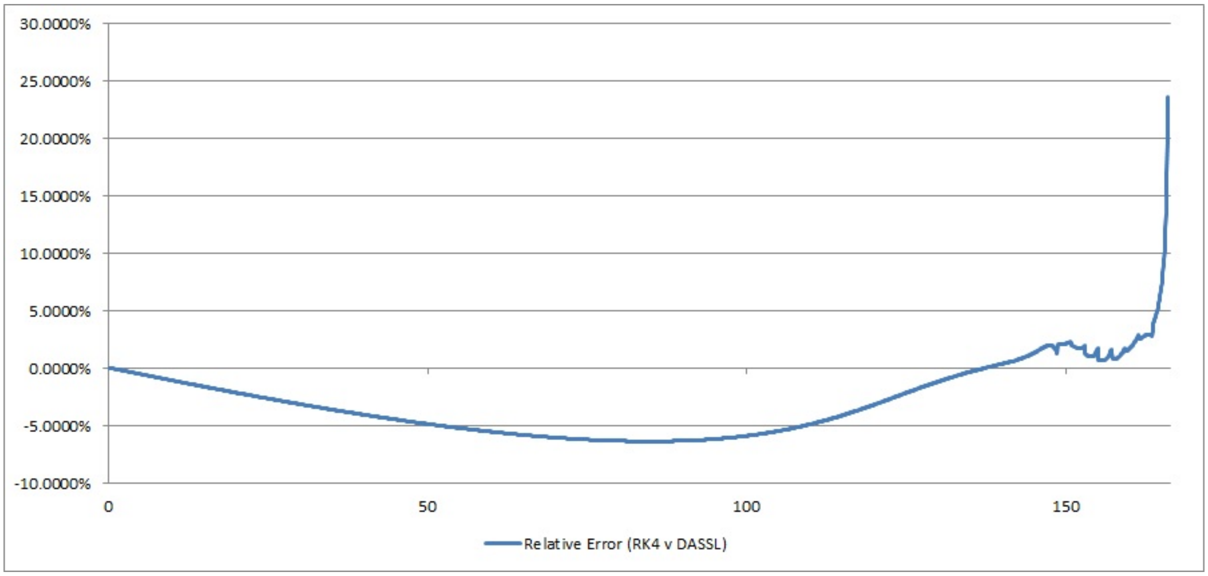

Figure 4.33: Error test: fossil fuel energy production - relative error

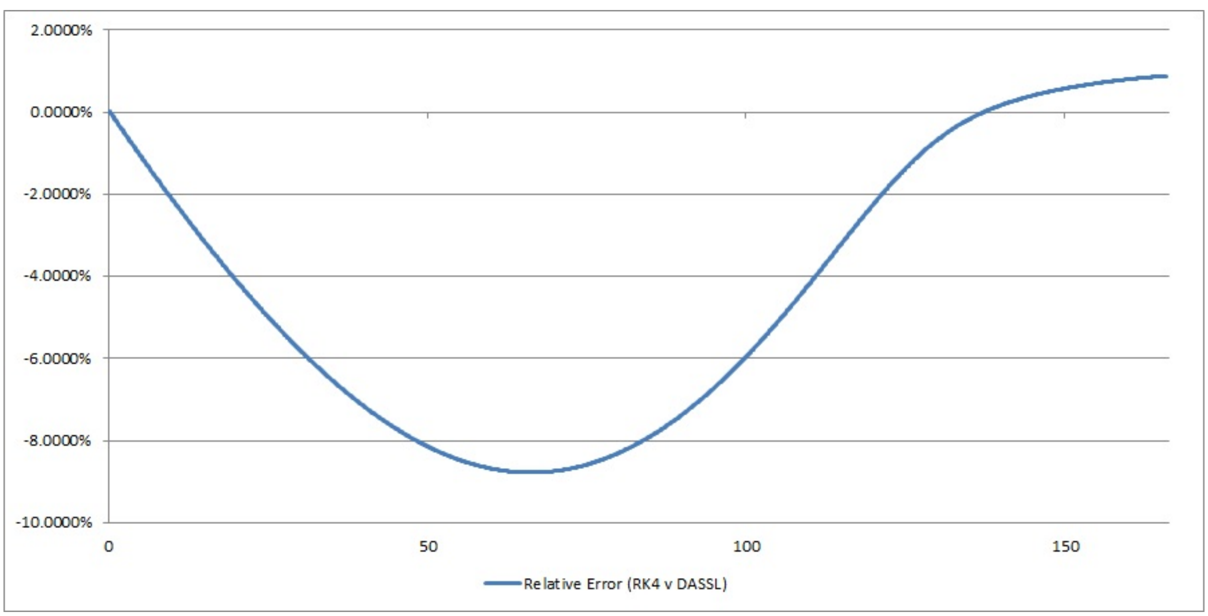

Figure 4.34: Error test: renewable energy production - relative error

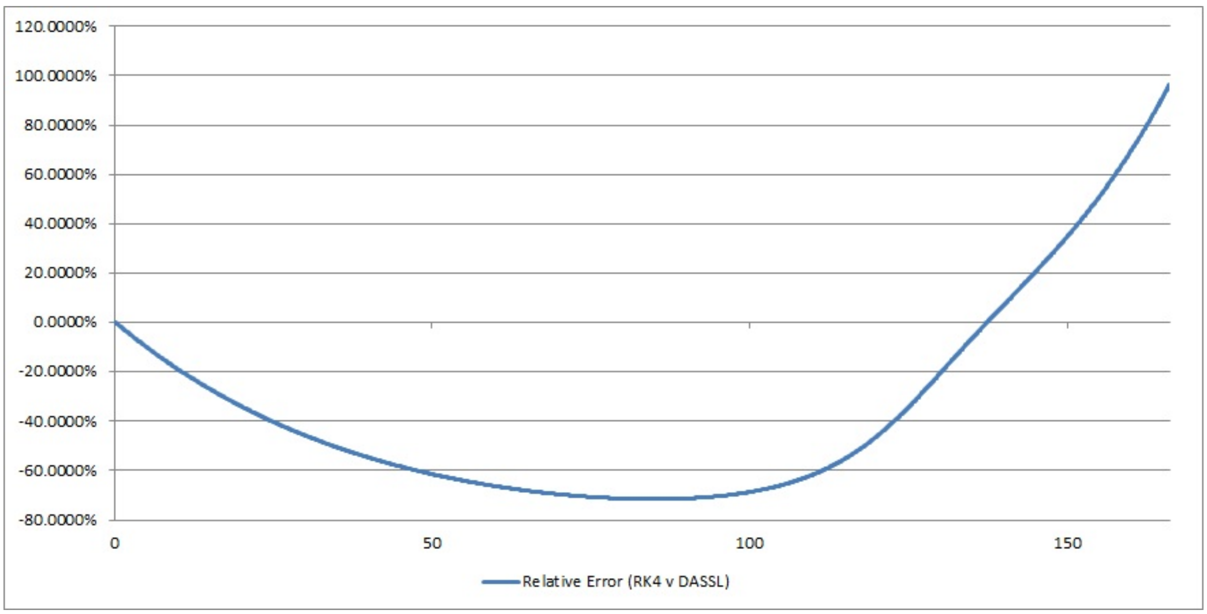

Figure 4.35: Error test: price of energy - relative error

The following plots display the same information but this time using the ODE model (Figures 4.38 to 4.42). In this case, the solutions are nearly identical. The error plots show that excluding the final years of fossil fuel production makes the relative error negligible. For this model, the error for innovation has also been provided. This result as with the previous test of step size shows that there are some limitations of the RK4-Newton's method solution when dealing with large exponential growth terms or over long time steps.

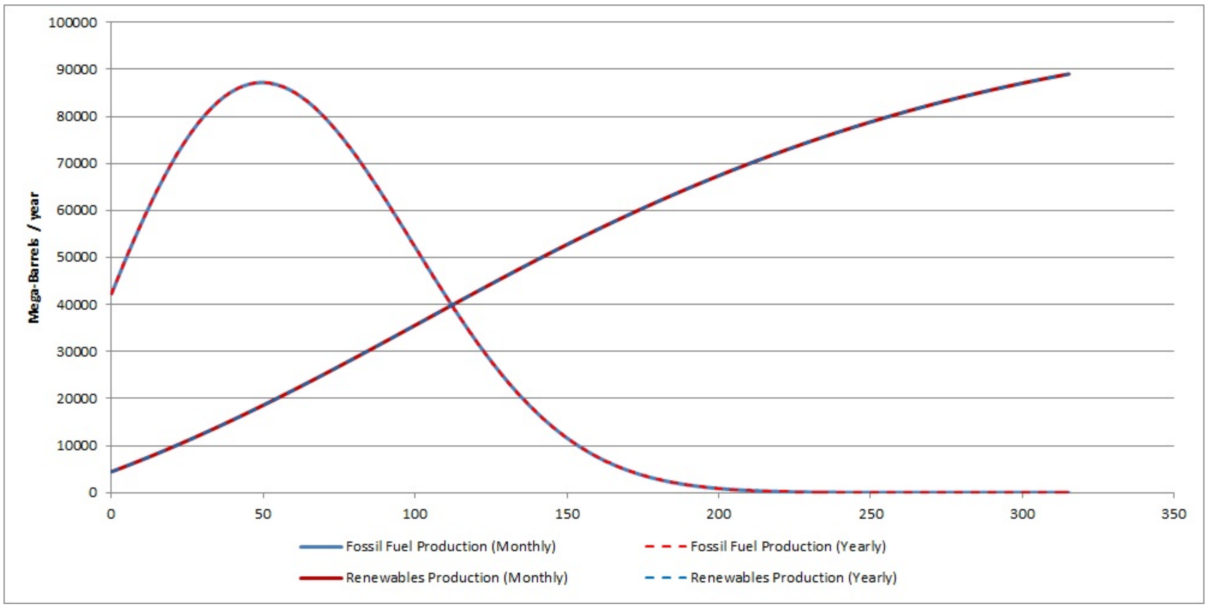

Figure 4.36: Error test: energy production

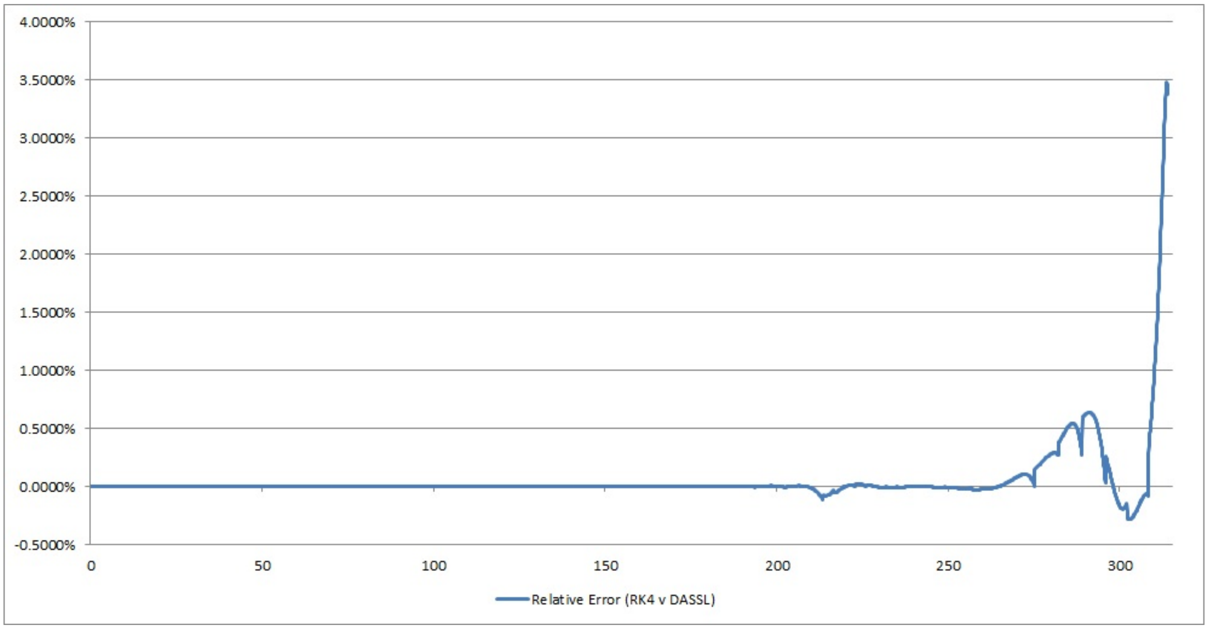

Figure 4.37: Error test: fossil fuel energy production - relative error

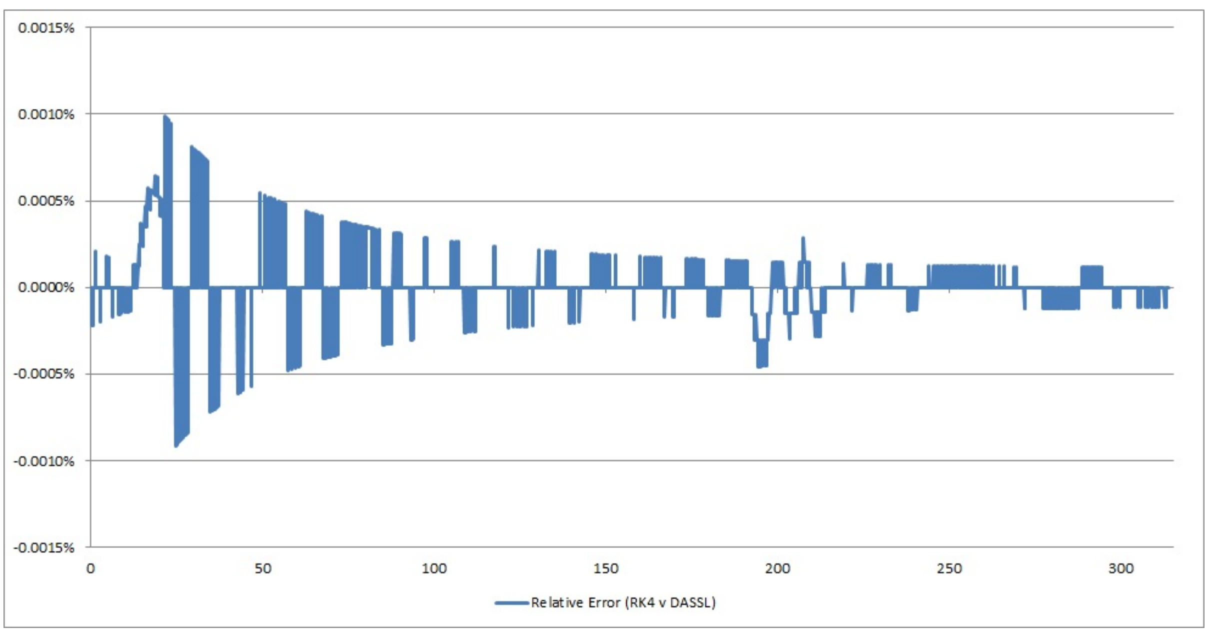

Figure 4.38: Error test: renewable energy production - relative error

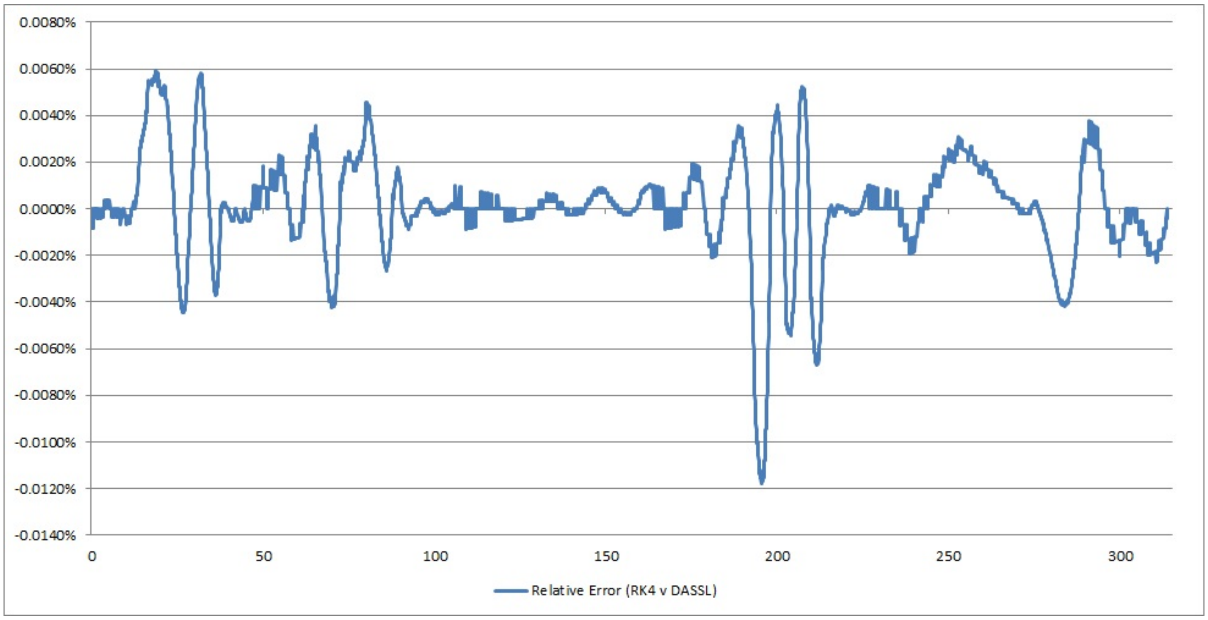

Figure 4.39: Error test: price of energy - relative error

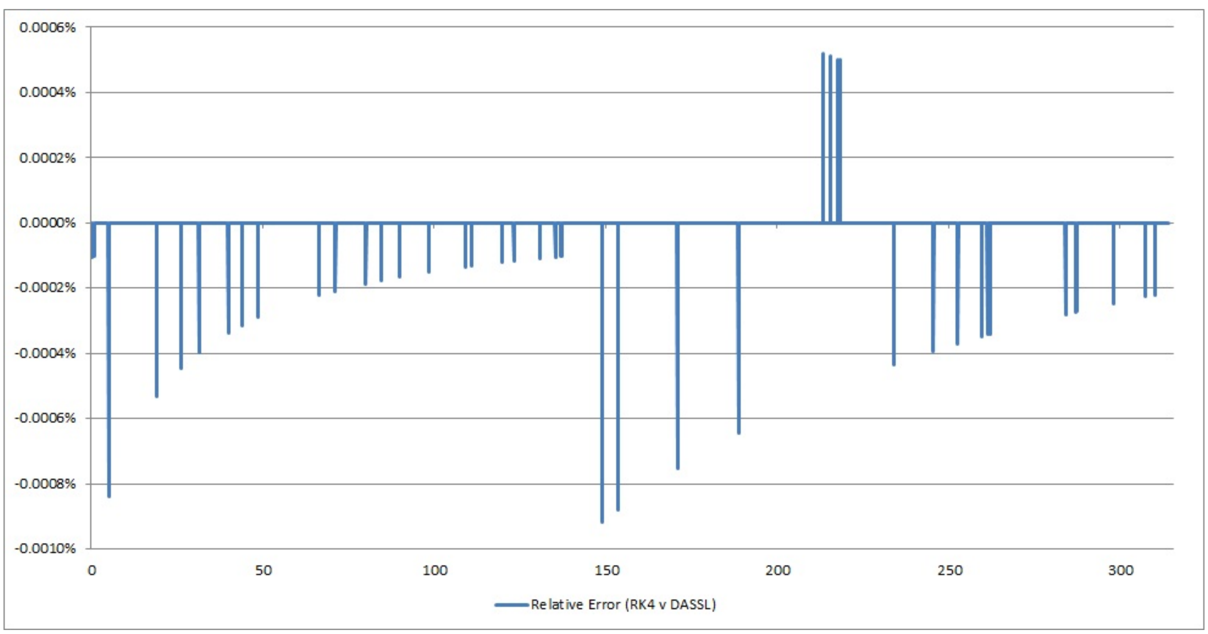

Figure 4.40: Error test: innovation in renewables - relative error

Chapter 5

Conclusion

In this chapter, the results presented in the last chapter are reviewed and analysed. The weak points found in the model are identified and some areas on which to focus further effort and improvements are suggested. First, the variability and adaptability of the model are assessed, followed by an assessment of the errors found between the solutions. Finally we conclude with a critique of the model when it is compared against real world data of the past three decades.

5.1 Model variability

In the first section of the results chapter the model is extensively tested to find the response and sensitivity to various model parameters. As has been stated many times, the primary focus of the model at this stage is on energy production. As such, the model parameters which are considered and tested are primarily related to the energy production functions. The first parameter under consideration was the diminishing returns factor ϵ used in the fossil fuel energy production equation. As the parameter is varied, the resulting peak in fossil fuel production is shifted. For lower values, the curve shifts so that the peak occurs at earlier times. Naturally, when the parameter is increased, the peak is shifted later into the future. As the peak is shifted away from the median values, it becomes narrower and as such also higher. This is due to the fixed initial reserves from which the energy is derived. The effect on renewable energy production is also provided; however the resulting effects are much more subtle than for fossil fuel production. It is notable that for values which correspond to earlier peaks, the growth in renewable energy production capacity is delayed but eventually rises more rapidly as the fossil fuel reserves are depleted. Also shown are the effects on energy price and investment in renewable energy. The price of energy displays the most rapid increase for values corresponding to an earlier peak in fossil fuel production. Once the fossil fuels are depleted the price of energy returns to a state of exponential growth. For the case when the peak is pushed to later periods the price of energy is held low until the fossil fuels are depleted. At this point, a very rapid rise occurs, seeing a return to the exponential growth state. The plot of investment in renewables displays interesting behaviour; a dip in investment is seen to coincide with the development of fossil fuel energy. The period before and after the development of the fossil fuels, the growth rate for investment in renewables is positive.

The second parameter under investigation, ϕ, relates to the renewable energy production and it controls the influence of capital on renewable energy production. This parameter was shown to have virtually no effect on the production cycle for fossil fuels but it had a major impact on the production curve for renewable energy. The plots were provided for three phases. For the first phase, there was steadily declining growth in renewable energy production capacity as the natural production limit was approached. For the second phase, the solutions became fairly stable, showing nearly linear growth in production capacity. The third and final phase displays an S-shaped curve where the growth rate for production capacity rises until an inflection point is reached and the growth rate begins to slow until the limit to production capacity is reached. Also shown were the plots for innovation and investment in renewables which displayed an opposite reaction to the parameter adjustments. The second phase corresponding to slow linear growth in production capacity coincides with the lowest levels of innovation and the highest levels of capital investment.

The third test depicts the effects of the variation of the parameters included in the ODE function for innovation in renewables. As expected, the inclusion of the second term in our innovation ODE gives a significant boost to innovation which in turn decreases the requirement for capital investment. An interesting point to note is that the overall production of renewables is relatively unchanged, and so these parameters seem to control the relative need for capital investment much more than the production capacity involved in renewable energy production.

In the final test of the model variation, the effect of raising the production limit for renewable energy is shown. The results indicate that the fossil fuel production is nearly identical for all cases but the production from renewables experiences more rapid growth for cases where the limit to production capacity is greatest.

5.2 Error testing and solution methods

In the final section, the error of our Runge-Kutta solution was provided for both the exponential and ODE model versions. For the exponential case the Runge-Kutta solution deviates from the DASSL solution significantly, particularly for the price of energy. Testing with a variety of step sizes showed that the solutions were quite stable, even for remarkably long time steps of 10 years.

For the model that utilizes ODE functions for innovation and extraction efficiency the results of the DASSL and the Runge-Kutta solution match extremely well. The errors are negligible, and a large step size may be used by either model. For nearly all other testing done, a step size of approximately one month was used.

5.3 Comparison with real world data

By far the most interesting investigation performed was an attempt to best fit the model results to real world data. For this test, energy production data was gathered from BP (2011) and aggregated into two forms, fossil fuel energy constituting oil, natural gas, and coal, and renewable energy which included all other forms of energy (primarily hydro, nuclear, wind, solar, and geothermal). This data was crucial in forming our estimates. However, it is limited in the timeframe of data provided and with only a single source, the estimates are assumed to be fairly crude. The required data for world population and global GDP were gathered from the World Bank (2011). The model shows promise when we examine the results. As in Berg *et al.* (2011), the production rate for fossil fuels was matched to the right order of magnitude. As well, the production figures for renewable energy were also able to match the real world data to the right order of magnitude. The growth rates for the two forms of energy were also relatively accurate for the short period examined. The exponential model seems to be a slightly better fit to the shape of the fossil fuel production data. A major improvement over the results presented by Berg (2011) is the accuracy of the price of energy produced by the model. The model produces energy prices in the right order of magnitude as those observed for crude oil during the same period. The model prediction for global GDP was also given. This comparison highlights the need for further improvements in the model as GDP was over estimated by nearly two orders of magnitude. Some of this discrepancy should be the result of the model parameters which were taken from the earlier work by Berg (2011) who in turn had used estimates obtained from a study on the German chemical industry (Kemfert 1998), this is obviously not an accurate representation of the global values. Another contributing factor stems from the assumptions made regarding population growth and using the assumption of full employment in the model. Since the real world will always exhibit less than full employment, our model has a surplus of available labour in comparison to the real world. Moreover, it was observed in the introduction and again in the results chapters that the assumption of exponential population growth is not representative of real population growth, which for the past three decades has exhibited linear growth rather than exponential.

5.4 Future areas of improvement

The most immediate modifications should be aimed at improving the accuracy of the GDP predictions. This could involve simple adjustments to the population function. Accuracy could also be improved by a deeper analysis of the parameters used in the production function. Determining a relation for the available labour which does not include full employment may also yield improved results in GDP prediction. Along with these modifications, the replacement of the remaining exponential functions with more representative relations for labour productivity and energy efficiency would seemingly aid the accuracy of GDP predictions. One hardship in developing this sort of model is the determination of parameters for which there is very little data to fit.

Another consideration which becomes apparent when the real world energy production and price data is examined is the inclusion of randomness and noise in certain parameters. There are many cases in the real world data where idiosyncratic shocks to production occur, which, as a result, change the production curves at later periods. Whether it is related to political events, regulation changes, wars or conflicts, there are many outside factors which have had noticeable and significant impact on global energy production, and consequently energy price in the past. The randomness could also be tied in with the innovation functions. This could add noise to the production increases and predicted energy prices.

A final consideration for improvements comes from the recognition that world energy production is actually the aggregation of energy production amongst many countries, each of which may display a different supply of renewable energy, rate of innovation growth, rent on investment capital, or limits to renewable energy production. If the model is expanded to include a network of countries, each developing their own economies, this may better reflect this reality. The work by Yamaji (2000) [1], Krabs (2004) [12] and Sumalia (2008) [19] provide an interesting application of game theory principals to emissions reduction/trading models and natural resource problems, and inspiration could be drawn for improvements in our model.

Chapter 6

Appendix: Program Code

This appendix contains the C++ code for the model. The code is broken down into three parts: *i)* the first is the main program which controls the solution method implemented and the model outputs, *ii)* the second contains the functions relevant to reading the configuration files, and *iii)* the functions used by the main program. First a sample configuration file is given.

6.1 Sample configuration file

```
# This is a configuration file for "Transition2Renewables.exe"
# Use "#" as leading character for comment lines
# Place all input settings inside {}

#################################################
# %%%%%    Numerical Method Parameters    %%%%
#################################################

# Prefix for output filenames
```

```
Output_Prefix = {RWT_exp_002_}

Size_of_Timestep = {1}

Threshold = {0.00001}

# Enable Debug mode to output additional data, intermediate steps (Select "ON" or "OFF")

Debug_Mode = {OFF}

# Solution Method (Select 0 for DASSL, or 1 for RK4)

Solution_Switch = {3}

# Innovation/Extraction Efficiency (Select 0 for exponentials, or 1 for ODEs)

ODE_Switch = {0}


#############################################

# %%%%%%%%%% Initial values   %%%%%%%%%%

#############################################


# Q(0)

Fossil_Fuels_Extracted_To_Date = {700000}

# Kr(0), initial guess for initalization

Capital_Investment_in_Renewables = {1000}

# K0

Capital_Investment_Factor = {25000}

# L(0)

Initial_Population = {4500000000}

# I(0)

Innovation_Coefficient = {0.0095}

# C(0)

Extraction_Efficiency_Coefficient = {0.002625}

# A(0)

Labour_Productivity_Coefficient = {298000}

# B(0)
```

```
Energy_Efficiency_Coefficient = {273000000}


##############################################
# %%%%%   Exponential Growth Factors %%%%%
##############################################


# gL
Population_Growth = {0.017}
# gA
Labour_Productivity_Growth = {0.03}
# gI
Innovation_Growth = {0.025}
# gC
Extraction_Efficiency_Growth = {0.025}
# gB
Energy_Efficiency_Growth = {0.015}


#######################################
# %%%%%     Model Parameters %%%%%%
#######################################


# Rr
Maximum_Energy_Extractable_From_Renewable_Sources = {100000}
# Rf
Fossil_Fuel_Reserves = {9700000}
# rK
Rent_on_Capital_Economy = {0.05}
# rE
Rent_on_Capital_Energy = {0.05}
```

```
# Labour to Capital/Energy Substitution Parameter

Alpha = {0.3}

# Capital/Energy Proportion Contribution Factor

Beta = {0.066}

# Capital to Energy Substitution Parameter

Rho = {0.55}

# Renewable Investment Power

Phi = {0.1}

# Fossil Fuels Extraction Diminishing Returns Factor

Epsilon = {0.050}

# Fraction of Renewables Profit Invested in Innovation"<<endl;

delR = {0.001}

# Fraction of Fossil Fuel Profit Invested in Extraction Efficiency"<<endl;

delF = {0.00011}

# Fraction of Fossil Fuel Profit Invested in Innovation"<<endl;

delrf = {0.0}
```

6.2 Transition2Renewables - Main

```
/* Definitions.h */

/////////////////////////

#include <iostream>

#include <fstream>

#include <cmath>

#include <math.h>

#include <vector>

#include <string>
```

```
#include <sstream>
#include <map>

using namespace std;
//////////////////////////

/*
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Transition2Renewables.exe

written by: John Mizanski
for Masters Thesis in the Modeling and Computation Science Program at UOIT
Summer/Winter 2012

Graduate Supervisor: Dr. P. Berg

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
*/

#include<Definitions.h>
#include<Functions_v2.h>
#include<ConfigReader.h>

// exponential growth factors
double gL, gI, gEQ, gEL, gEE;
// misc. factors
```

```
double alpha, beta, rho, phi, epsilon;
// initial values
double Qz, Kz, Lzero, Iz, ELz, EEz, EQz, Krz;
// energy limiting factors and interest rate factors
double Rr, Rf, rE, rK;
// engine variables (tolerance, step-size, number of timesteps)
double thresh, h, N;
// growth rate factors for ODE representation of innovation and extraction efficiency
double delR, delF, delRF;
string term_cond;
string prefix;
string DEBUG;
int SOL_Switch, Inn_Switch;
int IN1, IN2, IN3, IN4, IN5, IN6, IN7, IN8, IN9, IN10; // DASSL parameters
double relTOL, absTOL; // DASSL Tolerances

vector<double> Q; // Oil Consumed
vector<double> Kr; // Investment in Renewable Energy
vector<double> I; // Innovation
vector<double> EQ; // Fossil Fuel Extraction Efficiency

string ConfigFile;

double Q_new, pE_new, Er_new, Ef_new, Kr_new, Kf_new,
K_new, L_new, Y_new, I_new, EQ_new, EE_new, EL_new;

int main(int argc, char *argv[])
{
cout.precision(12);
```

```
/* //////////////////////////////////////////////////////////
Read Configuration file
////////////////////////////////////////////////////////// */
if(argc > 2)
{
cout << "Ensure configuration filename has no spaces"<<endl;
return 1;
}
else if(argc == 2)
ConfigFile = argv[1];
else {}
if(ConfigFile == "")
ConfigFile = "T2R_Settings.txt"; // defauflt configuration file filename
cout << "Using config: " << ConfigFile << endl;
if(ConfReader(ConfigFile.c_str()) == 1)
{
cout<<"ERROR - failed to read configuration file: "<<ConfigFile<<endl;
system("pause");
return 2;
}
else
cout<<"Finished Reading Configuration file"<<endl;
cout<<endl;
/* //////////////////////////////////////////////////////////
End Read Configuration file
////////////////////////////////////////////////////////// */
double DBG;
if(DEBUG == "ON")
```

```
DBG = 1.0;
else
DBG = 2.0;
int DASSL_Switch = SOL_Switch;
int EQ_Switch = Inn_Switch;
double Q_Stop = 1;
int time_zero = 1990; // 0;
/////////////////////// Save model constants to a vector ////////////////////////
double rPar[24];
rPar[0] = Rf; rPar[1] = Rr; rPar[2] = delF; rPar[3] = delRF; rPar[4] = delR;
rPar[5] = rE; rPar[6] = rK; rPar[7] = Kz; rPar[8] = alpha; rPar[9] = beta;
rPar[10]= epsilon; rPar[11]= phi; rPar[12]= rho; rPar[13]= ELz; rPar[14]= gEL;
rPar[15]= EEz; rPar[16]= gEL; rPar[17]= Lzero; rPar[18]= gL; rPar[19]= EQz;
rPar[20]= gEQ; rPar[21]= Iz; rPar[22]= gI; rPar[23]= DBG;

int iPar[1];
iPar[0] = EQ_Switch;
double Kr_Initialized = NewtonsMethod(Krz, Qz, Iz, EQz, 0.0, thresh, rPar);
cout<<"Initialized Value for Kr = "<<Kr_Initialized<<endl;
Kr.push_back(Kr_Initialized);
Q.push_back(Qz);
/////////////////////// Open Output File ///////////////////////////
ofstream oDATA_RK, oDATA_D;
/////////////////////// Save DASSL Parameters ///////////////////////////
if(DASSL_Switch == 0 || DASSL_Switch == 3)
{
string filename = prefix + "Transition2Renewables_DataTable_DASSL.tsv";
oDATA_D.open(filename.c_str(), ios::out);
oDATA_D<<"Year\tInvestment in Renewables (Kr)\tTotal Fossil Fuels Extracted (Q)
```

```
\tInnovation for Renewables (I)\tFossil Fuel Extraction Efficiency (C)\t";
oDATA_D<<"Renewable Energy (Er)\tFossil Fuel Energy (Ef)\tPrice of Energy (pE)
\tEconomic Output (Y)\n";
cout<<"DASSL Solution Method Selected"<<endl;
DASSL_Reader("DASSL_Settings.txt");
int info[15];
info[0] = 0; // Set to 0 for 1st call
info[1] = IN1; // 0) Scalar tol, 1) Array tol
info[2] = IN2; // 0) no intermediate steps, 1) output intermediate steps
info[3] = IN3; // 0) no TSTOP, 1) yes TSTOP
info[4] = IN4; // 0) solve PDs with numerical differences, 1) supply Jacobian function
info[5] = IN5; // 0) dense matrix, 1) bandded matrix
info[6] = IN6; // 0) no MAX stepsize, 1) supply MAX stepsize
info[7] = IN7; // 0) no initial stepsize-, 1) supply initial stepsize
info[8] = IN8; // 0) default MAX order (5), 1) set MAX order
info[9] = IN9; // 0) no restrictions on negative solutions, 1) restrict negative solutions
info[10] = IN10; // 0) initial conditions are consistent, 1) ICs are a guess

double Rtol = relTOL;
double Atol = absTOL;

double dWorkArray[100];
//dWorkArray[2] = 0.05; // initial stepsize
int lengthOfDWork = 100;
int iWorkArray[24];
int lengthOfIWork = 24;
int IDID;
if(EQ_Switch == 0) // Innovation and Extraction Efficiency are exponentials
{
```

```
double y0 = Kr_Initialized;
double y1 = Qz;
double yP0 = 0.0;
double yP1 = -F2(y0, y1, 0.0, Iz, 0.0, EQz, rPar);

int noOfEquations = 2;
double y[2];
double yPrime[2];

int i = time_zero; // time-step counter
Q_new = Qz;

pE_new = EnergyPrice(Kr_Initialized, Iz, Kz, rE, Rr, phi);
Er_new = Renew_Energy(Kr_Initialized, Iz, Kz, Rr, phi);
Ef_new = FF_Energy(Qz, EQz, pE_new, Rf, epsilon, rE);
K_new = Capital_Invest(Er_new, Ef_new, pE_new, EEz, rK, beta, rho);
Y_new = Production(Er_new, Ef_new, K_new, Lzero, ELz, EEz, beta, alpha, rho);
// output inital time-step
oDATA_D<<0<<"\t"<<Kr_Initialized<<"\t"<<Qz<<"\t"<<Iz<<"\t"<<EQz<<"\t"<<
Er_new<<"\t"<<Ef_new<<"\t"<<pE_new<<"\t"<<Y_new<<"\n";

while(Q_new < Q_Stop * Rf)
{
i++;
double time = i * h; // cout<<"Time = "<<time<<endl; // advance time by one time-step
if(!((i-1) % 100)) // on screen up-date on engine progress (every 100 time-steps)
{
// displays Year, Kr, and Q
cout<<"\n Number of Time-Steps Completed: "<<i-1<<endl;
```

```
cout<<(i-1)*h<<"\tKr = "<<Kr.at(i-1)<<"\tQ = "<<Q.at(i-1)<<endl;
cout<<endl;
}
y[0] = y0; //cout<<"Kr = "<<y0<<endl;
y[1] = y1; //cout<<"Q = "<<y1<<endl;

yPrime[0] = yP0;
yPrime[1] = yP1;

double Stime = time - h;
//cout<<"Begin DASSL"<<endl;
//system("pause");
ddassl_(RES, noOfEquations, Stime, y, yPrime, time, info, Rtol, Atol, IDID,
dWorkArray, lengthOfDWork, iWorkArray, lengthOfIWork, rPar, iPar, jac);
//cout<<"DASSL run -> "<<IDID<<endl;

y0 = y[0]; y1 = y[1];
yP0 = yPrime[0]; yP1 = yPrime[1];

Kr_new = y0;
Q_new = y1;
Kr.push_back(Kr_new);
Q.push_back(Q_new);

if(!(i % 1/*0*/)) // modify for less frequent outputs
{
I_new = exponential(Iz, gI, time);
L_new = exponential(Lzero, gL, time);
EQ_new = exponential(EQz, gEQ, time);
```

```
EE_new = exponential(EEz, gEE, time);
EL_new = exponential(ELz, gEL, time);
pE_new = EnergyPrice(Kr_new, I_new, Kz, rE, Rr, phi);
Er_new = Renew_Energy(Kr_new, I_new, Kz, Rr, phi);
Ef_new = FF_Energy(Q_new, EQ_new, pE_new, Rf, epsilon, rE);
K_new = Capital_Invest(Er_new, Ef_new, pE_new, EE_new, rK, beta, rho);
Y_new = Production(Er_new, Ef_new, K_new, L_new, EL_new, EE_new, beta, alpha, rho);

// output time-step
oDATA_D<<time<<"\t"<<Kr_new<<"\t"<<Q_new<<"\t"<<I_new<<"\t"<<EQ_new<<"\t"<<
Er_new<<"\t"<<Ef_new<<"\t"<<pE_new<<"\t"<<Y_new<<"\n";
}
}
}
else if(EQ_Switch == 1) // Innovation and Extraction are ODEs
{
I.push_back(Iz);
EQ.push_back(EQz);

pE_new = EnergyPrice(Kr_Initialized, Iz, Kz, rE, Rr, phi);
Er_new = Renew_Energy(Kr_Initialized, Iz, Kz, Rr, phi);
Ef_new = FF_Energy(Qz, EQz, pE_new, Rf, epsilon, rE);
K_new = Capital_Invest(Er_new, Ef_new, pE_new, EEz, rK, beta, rho);
Y_new = Production(Er_new, Ef_new, K_new, Lzero, ELz, EEz, beta, alpha, rho);
// output inital time-step
oDATA_D<<0<<"\t"<<Kr_Initialized<<"\t"<<Qz<<"\t"<<Iz<<"\t"<<EQz<<"\t"<<
Er_new<<"\t"<<Ef_new<<"\t"<<pE_new<<"\t"<<Y_new<<"\n";

double y0 = Kr.at(0);
```

```
double y1 = Q.at(0);

double y2 = I.at(0);

double yP0 = 0.0;

double yP1 = -F2(y0, y1, 0.0, y2, 0.0, EQz, rPar);

double yP2 = -F3(y0, y1, y2, 0.0, 0.0, EQz, rPar);


int noOfEquations = 3;

double y[3];

double yPrime[3];


// time-step counter

int i = time_zero;

Q_new = Qz;


while(Q_new < Q_Stop * Rf)

{

i++;

double time = i * h; // cout<<"Time = "<<time<<endl; // advance time by one time-step

if(!((i-1) % 100)) // on screen up-date on engine progress (every 100 time-steps)

{

// displays Year, Kr, and Q

cout<<"\n Number of Time-Steps Completed: "<<i-1<<endl;

cout<<(i-1)*h<<"\tKr = "<<Kr.at(i-1)<<"\tQ = "<<Q.at(i-1)<<endl;

cout<<endl;

}

y[0] = y0; //cout<<"Kr = "<<y0<<endl;

y[1] = y1; //cout<<"Q = "<<y1<<endl;

y[2] = y2; //cout<<"I = "<<y2<<endl;
```

```
yPrime[0] = yP0;
yPrime[1] = yP1;
yPrime[2] = yP2;

double Stime = time - h;
//cout<<"Begin DASSL"<<endl;
//system("pause");
ddassl_(RES, noOfEquations, Stime, y, yPrime, time, info, Rtol, Atol, IDID,
dWorkArray, lengthOfDWork, iWorkArray, lengthOfIWork, rPar, iPar, jac);

//cout<<"DASSL run -> "<<IDID<<endl;

y0 = y[0]; y1 = y[1]; y2 = y[2];
yP0 = yPrime[0]; yP1 = yPrime[1]; yP2 = yPrime[2];

Kr_new = y0;
Q_new = y1;
I_new = y2;
// ODE for Extraction Efficiency can be solved and produces a linear growth rate
EQ_new = ExtractionEfficiency(EQz, delF, rE, epsilon, time);
Kr.push_back(Kr_new);
Q.push_back(Q_new);
I.push_back(I_new);
EQ.push_back(EQ_new);

if(!(i % 1/*0*/)) // for less frequent outputs
{
//I_new = exponential(Iz, gI, time);
L_new = exponential(Lzero, gL, time);
```

```
//EQ_new = exponential(EQz, gEQ, time);
EE_new = exponential(EEz, gEE, time);
EL_new = exponential(ELz, gEL, time);
pE_new = EnergyPrice(Kr_new, I_new, Kz, rE, Rr, phi);
Er_new = Renew_Energy(Kr_new, I_new, Kz, Rr, phi);
Ef_new = FF_Energy(Q_new, EQ_new, pE_new, Rf, epsilon, rE);
K_new = Capital_Invest(Er_new, Ef_new, pE_new, EE_new, rK, beta, rho);
Y_new = Production(Er_new, Ef_new, K_new, L_new, EL_new, EE_new, beta, alpha, rho);

// output time-step
oDATA_D<<time<<"\t"<<Kr_new<<"\t"<<Q_new<<"\t"<<I_new<<"\t"<<EQ_new<<"\t"<<
Er_new<<"\t"<<Ef_new<<"\t"<<pE_new<<"\t"<<Y_new<<"\n";
//cout<<time<<"\t"<<Kr_new<<"\t"<<Q_new<<"\t"<<I_new<<"\t"<<EQ_new<<"\t"<<
//Er_new<<"\t"<<Ef_new<<"\t"<<pE_new<<"\t"<<Y_new<<"\n";
}
}
}
}
if(DASSL_Switch == 1 || DASSL_Switch == 3)
{
string filename = prefix + "Transition2Renewables_DataTable_RK4.tsv";
oDATA_RK.open(filename.c_str(), ios::out);
oDATA_RK<<"Year\tInvestment in Renewables (Kr)\tTotal Fossil Fuels Extracted (Q)
\tInnovation for Renewables (I)\tFossil Fuel Extraction Efficiency (C)\t";
oDATA_RK<<"Renewable Energy (Er)\tFossil Fuel Energy (Ef)\tPrice of Energy (pE)
\tEconomic Output (Y)\n";
if(EQ_Switch == 0) // Innovation and Extraction Efficiency are exponentials
{
pE_new = EnergyPrice(Kr_Initialized, Iz, Kz, rE, Rr, phi);
```

```
Er_new = Renew_Energy(Kr_Initialized, Iz, Kz, Rr, phi);
Ef_new = FF_Energy(Qz, EQz, pE_new, Rf, epsilon, rE);
K_new = Capital_Invest(Er_new, Ef_new, pE_new, EEz, rK, beta, rho);
Y_new = Production(Er_new, Ef_new, K_new, Lzero, ELz, EEz, beta, alpha, rho);
// output inital time-step
oDATA_RK<<0<<"\t"<<Kr_Initialized<<"\t"<<Qz<<"\t"<<Iz<<"\t"<<EQz<<"\t"<<
Er_new<<"\t"<<Ef_new<<"\t"<<pE_new<<"\t"<<Y_new<<"\n";

int i = time_zero; // time-step counter
Q_new = Q.at(i);

while(Q_new < Q_Stop * Rf)
{
i++;
double time = i * h; // advance time by one time-step

if(!((i-1) % 100)) // on screen up-date on engine progress (every 100 time-steps)
{
// displays Year, Kr, and Q
cout<<"\n Number of Time-Steps Completed: "<<i-1<<endl;
cout<<(i-1)*h<<"\tKr = "<<Kr.at(i-1)<<"\tQ = "<<Q.at(i-1)<<endl;
cout<<endl;
}
// Solve for Kr and Q using combination of Runge-Kutta 4, and Newton's method
vector<double> NXT = T2R_RK4(Kr.at(i - 1), Q.at(i - 1), Iz, time, rPar,
h, thresh, EQ_Switch);

Kr_new = NXT.at(0);
ISNAN("Investment in Renewables",Kr_new);
```

```
Kr.push_back(Kr_new);
Q_new = NXT.at(1);
ISNAN("Oil Consumed",Q_new);
Q.push_back(Q_new);

if(!(i % 1/*0*/)) // for less frequent outputs
{
I_new = exponential(Iz, gI, time);
L_new = exponential(Lzero, gL, time);
EQ_new = exponential(EQz, gEQ, time);
EE_new = exponential(EEz, gEE, time);
EL_new = exponential(ELz, gEL, time);
pE_new = EnergyPrice(Kr_new, I_new, Kz, rE, Rr, phi);
Er_new = Renew_Energy(Kr_new, I_new, Kz, Rr, phi);
Ef_new = FF_Energy(Q_new, EQ_new, pE_new, Rf, epsilon, rE);
K_new = Capital_Invest(Er_new, Ef_new, pE_new, EE_new, rK, beta, rho);
Y_new = Production(Er_new, Ef_new, K_new, L_new, EL_new, EE_new, beta, alpha, rho);

// output time-step
oDATA_RK<<time<<"\t"<<Kr_new<<"\t"<<Q_new<<"\t"<<I_new<<"\t"<<EQ_new<<"\t"<<
Er_new<<"\t"<<Ef_new<<"\t"<<pE_new<<"\t"<<Y_new<<"\n";
}
}
}
else if(EQ_Switch == 1) // Innovation is moddeled with an ODE and Extraction Efficiency has
{
I.push_back(Iz);
EQ.push_back(EQz);
pE_new = EnergyPrice(Kr_Initialized, Iz, Kz, rE, Rr, phi);
```

```
Er_new = Renew_Energy(Kr_Initialized, Iz, Kz, Rr, phi);
Ef_new = FF_Energy(Qz, EQz, pE_new, Rf, epsilon, rE);
K_new = Capital_Invest(Er_new, Ef_new, pE_new, EEz, rK, beta, rho);
Y_new = Production(Er_new, Ef_new, K_new, Lzero, ELz, EEz, beta, alpha, rho);
// output inital time-step
oDATA_RK<<0<<"\t"<<Kr_Initialized<<"\t"<<Qz<<"\t"<<Iz<<"\t"<<EQz<<"\t"<<
Er_new<<"\t"<<Ef_new<<"\t"<<pE_new<<"\t"<<Y_new<<"\n";

int i = time_zero;
Q_new = Qz;
Kr_new = Kr_Initialized;
I_new = Iz;
EQ_new = EQz;

while(Q_new < Q_Stop * Rf)
{
i++;
double time = i * h;
if(!((i-1) % 100))
{
cout<<"\n Number of Time-Steps Completed: "<<i-1<<endl;
cout<<"Year = "<<(i-1)*h<<"\tKr = "<<Kr.at(i-1)<<"\tQ = "<<Q.at(i-1)<<
"\tI = "<<I.at(i-1)<<"\tEQ = "<<EQ.at(i-1)<<endl;
cout<<endl;
}
vector<double> NXT = T2R_RK4(Kr.at(i - 1), Q.at(i - 1), I.at(i - 1), time, rPar,
h, thresh, EQ_Switch);

Kr_new = NXT.at(0);
```

```
Q_new = NXT.at(1);
I_new = NXT.at(2);
// ODE for Extraction Efficiency can be solved and produces a linear growth rate
EQ_new = ExtractionEfficiency(EQz, delF, rE, epsilon, time);
Kr.push_back(Kr_new);
Q.push_back(Q_new);
I.push_back(I_new);
EQ.push_back(EQ_new);

if(!(i % 1/*0*/)) // for less frequent outputs
{
//I_new = exponential(Iz, gI, time);
L_new = exponential(Lzero, gL, time);
//EQ_new = exponential(EQz, gEQ, time);
EE_new = exponential(EEz, gEE, time);
EL_new = exponential(ELz, gEL, time);
pE_new = EnergyPrice(Kr_new, I_new, Kz, rE, Rr, phi);
Er_new = Renew_Energy(Kr_new, I_new, Kz, Rr, phi);
Ef_new = FF_Energy(Q_new, EQ_new, pE_new, Rf, epsilon, rE);
K_new = Capital_Invest(Er_new, Ef_new, pE_new, EE_new, rK, beta, rho);
Y_new = Production(Er_new, Ef_new, K_new, L_new, EL_new, EE_new, beta, alpha, rho);

// output time-step
oDATA_RK<<time<<"\t"<<Kr_new<<"\t"<<Q_new<<"\t"<<I_new<<"\t"<<EQ_new<<"\t"<<
Er_new<<"\t"<<Ef_new<<"\t"<<pE_new<<"\t"<<Y_new<<"\n";
}
}
}
else
```

```
{
cout<<"EQ_Switch should be set to 0 or 1"<<endl;
system("pause");
exit(1);
}
}
else
{
cout<<"DASSL_Switch should be set to 0 or 1"<<endl;
system("pause");
exit(1);
}
oDATA_D.close();
oDATA_D.clear();
oDATA_RK.close();
oDATA_RK.clear();

//system("pause");
return 0;
}
```

6.3 Transition2Renewables - ConfigReader

```
/* ConfigReader.h */
///////////////////////////////////
#include<Definitions.h>
```

```
// elasticities, diminishing returns parameters
extern double alpha, beta, rho, phi, epsilon;
// Exponential Growth parameters
extern double gL, gI, gEQ, gEL, gEE;
// Initial Values
extern double Qz, Krz, Kz, Lzero, Iz, ELz, EEz, EQz;
// Energy Reserves, Rent on Capital
extern double Rr, Rf, rE, rK;
// Threshold, stepsize, number of time steps
extern double thresh, h, N;
extern string term_cond;
extern string prefix;
extern double delR, delF, delRF;
extern string DEBUG;
extern int SOL_Switch, Inn_Switch;
extern int IN1, IN2, IN3, IN4, IN5, IN6, IN7, IN8, IN9, IN10;
extern double relTOL, absTOL;

double ReadDouble(string textline);
int ReadInteger(string textline);
string ReadString(string textline);
int ConfReader(string filename);
int DASSL_Reader(string filename);
////////////////////////////////////////


#include<Definitions.h>
#include<ConfigReader.h>
```

```
double ReadDouble(string textline)
{
size_t start = textline.find_first_of("{");
size_t end = textline.find_first_of("}");

double temp;
stringstream ss;
ss << textline.substr(start + 1, end - start);
ss >> temp;

//cout<<temp<<endl;

return temp;
}
int ReadInteger(string textline)
{
size_t start = textline.find_first_of("{");
size_t end = textline.find_first_of("}");

int temp;
stringstream ss;
ss << textline.substr(start + 1, end - start);
ss >> temp;

//cout<<temp<<endl;

return temp;
}
```

```
string ReadString(string textline)
{
size_t start = textline.find_first_of("{");
size_t end = textline.find_first_of("}");

string temp;
temp = textline.substr(start + 1, end - start - 1);

//cout<<temp<<endl;

return temp;
}
int ConfReader(string filename)
{
ifstream confIN(filename.c_str(), ios::in);

if(!confIN.is_open())
{
cout<<"Error opening Configuration file: "<<filename<<endl;
system("pause");
return 1;
}
cout << "Reading program settings from configuration file ..."<<endl;
cout<<endl;

string textline;

while(getline(confIN, textline, '\n'))
{
```

```
//cout<<textline<<endl;
if(textline == "")
continue;
if(textline.substr(0, 1).find("#") != string::npos)
continue;
if(textline.find("Debug_Mode")!=string::npos)
DEBUG = ReadString(textline);
if(textline.find("Solution_Switch")!=string::npos)
SOL_Switch = ReadInteger(textline);
if(textline.find("ODE_Switch")!=string::npos)
Inn_Switch = ReadInteger(textline);
if(textline.find("Fossil_Fuels_Extracted_To_Date")!=string::npos)
Qz = ReadDouble(textline);
//else if(textline.find("Price_of_Energy")!=string::npos)
// pEz = ReadDouble(textline);
else if(textline.find("Capital_Investment_in_Renewables")!=string::npos)
Krz = ReadDouble(textline);
else if(textline.find("Capital_Investment_Factor")!=string::npos)
Kz = ReadDouble(textline);
else if(textline.find("Initial_Population")!=string::npos)
Lzero = ReadDouble(textline);
else if(textline.find("Innovation_Coefficient")!=string::npos)
Iz = ReadDouble(textline);
else if(textline.find("Extraction_Efficiency_Coefficient")!=string::npos)
EQz = ReadDouble(textline);
else if(textline.find("Labour_Productivity_Coefficient")!=string::npos)
ELz = ReadDouble(textline);
else if(textline.find("Energy_Efficiency_Coefficient")!=string::npos)
EEz = ReadDouble(textline);
```

```
else if(textline.find("Population_Growth")!=string::npos)
gL = ReadDouble(textline);
else if(textline.find("Labour_Productivity_Growth")!=string::npos)
gEL = ReadDouble(textline);
else if(textline.find("Innovation_Growth")!=string::npos)
gI = ReadDouble(textline);
else if(textline.find("Extraction_Efficiency_Growth")!=string::npos)
gEQ = ReadDouble(textline);
else if(textline.find("Energy_Efficiency_Growth")!=string::npos)
gEE = ReadDouble(textline);
else if(textline.find("Maximum_Energy_Extractable_From_Renewable_Sources")!=string::npos)
Rr = ReadDouble(textline);
else if(textline.find("Fossil_Fuel_Reserves")!=string::npos)
Rf = ReadDouble(textline);
else if(textline.find("Rent_on_Capital_Economy")!=string::npos)
rK = ReadDouble(textline);
else if(textline.find("Rent_on_Capital_Energy")!=string::npos)
rE = ReadDouble(textline);
else if(textline.find("Alpha")!=string::npos)
alpha = ReadDouble(textline);
else if(textline.find("Beta")!=string::npos)
beta = ReadDouble(textline);
else if(textline.find("Rho")!=string::npos)
rho = ReadDouble(textline);
else if(textline.find("Phi")!=string::npos)
phi = ReadDouble(textline);
else if(textline.find("Epsilon")!=string::npos)
epsilon = ReadDouble(textline);
//else if(textline.find("Number_of_Timesteps")!=string::npos)
```

```
// N = ReadInteger(textline);
else if(textline.find("Size_of_Timestep")!=string::npos)
h = ReadDouble(textline);
else if(textline.find("Threshold")!=string::npos)
thresh = ReadDouble(textline);
else if(textline.find("Use_Max_Timesteps")!=string::npos)
term_cond = ReadString(textline);
else if(textline.find("Output_Prefix")!=string::npos)
prefix = ReadString(textline);
else if(textline.find("delR")!=string::npos)
delR = ReadDouble(textline);
else if(textline.find("delF")!=string::npos)
delF = ReadDouble(textline);
else if(textline.find("delrf")!=string::npos)
delRF = ReadDouble(textline);
else{}
}
cout<<Qz<<"\t- Q(0) - Fossil fuels extracted to date"<<endl;
//cout<<pEz<<"\t- Price of energy"<<endl;
cout<<Krz<<"\t- Kr(0) - Capital_Investment_in_Renewables"<<endl;
cout<<Kz<<"\t- Ko - Capital_Investment_Factor"<<endl;
cout<<Lzero<<"\t- L(0) - Population"<<endl;
cout<<Iz<<"\t- I(0) - Current level of innovation"<<endl;
cout<<EQz<<"\t- C(0) - Current fossil fuel extracton efficiency"<<endl;
cout<<ELz<<"\t- A(0) - Current labour productivity"<<endl;
cout<<EEz<<"\t- B(0) - Current energy efficiency"<<endl;
cout<<gL<<"\t- gL - Population growth factor"<<endl;
cout<<gEL<<"\t- gA - Labour productivity growth factor"<<endl;
cout<<gI<<"\t- gI - Innovation growth factor"<<endl;
```

```
cout<<gEQ<<"\t- gC - Extraction efficiency growth factor"<<endl;
cout<<gEE<<"\t- gB - Energy efficiency growth factor"<<endl;
cout<<Rr<<"\t- Rr - Limit to energy production from renewables"<<endl;
cout<<Rf<<"\t- Rf - Fossil fuel reserves"<<endl;
cout<<rK<<"\t- rK - Rent paid on investment capital"<<endl;
cout<<rE<<"\t- rE - Rent on capital paid by energy industry"<<endl;
cout<<alpha<<"\t- Alpha - Elasiticity parameter"<<endl;
cout<<beta<<"\t- Beta - Share parameter"<<endl;
cout<<rho<<"\t- Rho - Elasticity parameter"<<endl;
cout<<phi<<"\t- Phi - Renewables capital power factor"<<endl;
cout<<epsilon<<"\t- Epsilon - Diminishing returns factor"<<endl;
//cout<<N<<"\t- Number of timesteps"<<endl;
cout<<delR<<"\t- delR - Fraction of Renewables Profit Invested in Innovation"<<endl;
cout<<delF<<"\t- delF - Fraction of Fossil Fuel Profit Invested in Extraction Efficiency"
<<endl;
cout<<delRF<<"\t- delRF - Fraction of Fossil Fuel Profit Invested in Innovation"<<endl;
cout<<h<<"\t- Stepsize"<<endl;
cout<<thresh<<"\t- Threshold"<<endl;
cout<<prefix<<"\t- Output prefix"<<endl;

confIN.close();
confIN.clear();
return 0;
}
int DASSL_Reader(string filename)
{
ifstream confIN(filename.c_str(), ios::in);

if(!confIN.is_open())
```

```
{
cout<<"Error opening Configuration file: "<<filename<<endl;
system("pause");
return 1;
}
cout << "Reading program settings from configuration file ..."<<endl;
cout<<endl;

string textline;

while(getline(confIN, textline, '\n'))
{
//cout<<textline<<endl;
if(textline == "")
continue;
if(textline.substr(0, 1).find("#") != string::npos)
continue;
if(textline.find("INFO_01")!=string::npos)
IN1 = ReadInteger(textline);
else if(textline.find("INFO_02")!=string::npos)
IN2 = ReadInteger(textline);
else if(textline.find("INFO_03")!=string::npos)
IN3 = ReadInteger(textline);
else if(textline.find("INFO_04")!=string::npos)
IN4 = ReadInteger(textline);
else if(textline.find("INFO_05")!=string::npos)
IN5 = ReadInteger(textline);
else if(textline.find("INFO_06")!=string::npos)
IN6 = ReadInteger(textline);
```

```
else if(textline.find("INFO_07")!=string::npos)
IN7 = ReadInteger(textline);
else if(textline.find("INFO_08")!=string::npos)
IN8 = ReadInteger(textline);
else if(textline.find("INFO_09")!=string::npos)
IN9 = ReadInteger(textline);
else if(textline.find("INFO_10")!=string::npos)
IN10 = ReadInteger(textline);
else if(textline.find("Relative_TOL")!=string::npos)
relTOL = ReadDouble(textline);
else if(textline.find("Abs_TOL")!=string::npos)
absTOL = ReadDouble(textline);
else{}
}
confIN.close();
confIN.clear();

return 0;
}
```

6.4 Transition2Renewables - Functions v2

```
/* Functions_v2.h */
///////////////////////////////////
#include<Definitions.h>

void ISNAN(string VarName, double Var);
```

```
double exponential(double initial, double rate, double time);
double Profit(double Ex, double Kx, double pE, double rE);
double EnergyPrice(double Kr, double I, double Ko, double rE, double Rr, double phi);
double Renew_Energy(double Kr, double I, double Ko, double Rr, double phi);
double FF_Energy(double Q, double EQ, double pE, double Rf, double epsilon, double rE);
double FF_Invest(double Ef, double pE, double rE, double epsilon);
double Capital_Invest(double Er, double Ef, double pE, double EE, double rK,
double beta, double rho);
double Production(double Er, double Ef, double K, double L, double EL, double EE,
double beta, double alpha, double rho);
double ExtractionEfficiency(double EQo, double delF, double rE, double epsilon, double t);

double F1(double Kr, double Q, double I, double t, double EQ, const double rPar[]);
double F2(double Kr, double Q, double QPrime, double I, double t, double EQ,
const double rPar[]);
double F3(double Kr, double Q, double I, double IPrime, double t, double EQ,
const double rPar[]);
double F1_dKr(double Kr, double Q, double I, double t, double EQ, const double rPar[]);
double F1_dQ(double Kr,double Q ,double I, double t, double EQ, const double rPar[]);
double F1_dI(double Kr, double Q, double I, double t, double EQ, const double rPar[]);
double F2_dKr(double Kr, double Q, double I, double t, double EQ, const double rPar[]);
double F2_dQ(double Kr, double I, double EQ, const double rPar[]);
double F2_dI(double Kr, double Q, double I, double t, double EQ, const double rPar[]);
double F3_dKr(double Kr, double Q, double I, double t, double EQ, const double rPar[]);
double F3_dQ(double Kr, double I, double t, double EQ, const double rPar[]);
double F3_dI(double Kr, double Q, double I, double t, double EQ, const double rPar[]);
double NewtonsMethod(double Kr, double Q, double I, double EQ, double t, double TOL,
const double rPar[]);
```

```
vector<double> T2R_RK4(double Kr, double Q, double I, double time, const double rPar[],
double h, double TOL, int EQ_Switch);

/////////////////////////////////////////////////////////////
////////////   FUNCTIONS FOR DASSL    ///////////////////
/////////////////////////////////////////////////////////////

void jac(const double& time, const double y[], const double yPrime[], double** PD,
double& CJ, double rPar[], int iPar[]);
void RES(const double& time, const double y[], const double yPrime[], double residue[],
int& iRes, const double rPar[], const int iPar[]);

extern "C" void ddassl_(
void (*funcptr)(const double& time, const double y[], const double yPrime[],
double residue[], int& iRes, const double rPar[], const int iPar[]),
const int& noOfEquations,
const double& currentTime,
const double initialY[],
const double initialYPrime[],
const double& finalTime,
const int info[15],
const double& relativeTolerance,
const double& absoluteTolerance,
int& outputStatusFlag,
const double dWorkArray[],
const int& lengthOfDWork,
const int iWorkArray[],
const int& lengthOfIWork,
const double rParArray[],
```

```
const int iParArray[],
void (*jacobian)(const double& time, const double y[], const double yPrime[],
double** PD, double& CJ, double rPar[], int iPar[])
);
////////////////////////////////////////////////////////////////////////
#include<Definitions.h>
#include<Functions_v2.h>

void ISNAN(string VarName, double Var)
{
if(_isnan(Var))
{
cout<<"[ERROR] - "<<VarName<<" = NaN"<<endl;
system("pause");
exit(1);
}
}
double exponential(double initial, double rate, double time)
{
double temp = initial * exp(rate*time);
return temp;
}
double Profit(double Ex, double Kx, double pE, double rE)
{
double tProfit = pE * Ex - rE * Kx;
return tProfit;
}
double EnergyPrice(double Kr, double I, double Ko, double rE, double Rr, double phi)
{
```

```
double Kr_phiP = pow(Kr, 1.0 + phi);
double Kr_phiN = pow(Kr, 1.0 - phi);
double Ko_phi = pow(Ko, phi);
double pE = (rE/(phi * Rr)) * (2.0 * Kr + (Ko_phi/I) * Kr_phiN + (I/Ko_phi) * Kr_phiP);
//cout<<"pE = "<<pE<<endl;

return pE;
}
double Renew_Energy(double Kr, double I, double Ko, double Rr, double phi)
{
double Kr_phi = pow(Kr, phi);
double Ko_phi = pow(Ko, phi);
double Er = Rr * I * Kr_phi/(I * Kr_phi + Ko_phi);
//cout<<"Er = "<<Er<<endl;

return Er;
}
double FF_Energy(double Q, double EQ, double pE, double Rf, double epsilon, double rE)
{
double tp = pow(epsilon * pE / rE, epsilon / (1.0 - epsilon) );
double Ef = pow(EQ, 1.0 / (1.0 - epsilon)) * (Rf - Q) * tp;
//cout<<"Ef = "<<Ef<<endl;

return Ef;
}
double FF_Invest(double Ef, double pE, double rE, double epsilon)
{
double Kf = Ef * pE * epsilon / rE;
//cout<<"Kf = "<<Kf<<endl;
```

```
return Kf;
}
double Capital_Invest(double Er, double Ef, double pE, double EE, double rK,
double beta, double rho)
{
double temp = beta * pow(EE, rho) * rK / ((1.0 - beta) * pE);
double K = (Er + Ef) * pow(temp, 1.0/(rho - 1.0));
//cout<<"K = "<<K<<endl;

return K;
}
double Production(double Er, double Ef, double K, double L, double EL, double EE,
double beta, double alpha, double rho)
{
double temp = (beta * pow(EE*(Er+Ef), rho) + (1.0 - beta) * pow(K, rho));
double Y = pow(temp, alpha/rho) * pow(EL * L, 1.0 - alpha);
//cout<<"Production (Y) = "<<Y<<endl;

return Y;
}
double ExtractionEfficiency(double EQo, double delF, double rE,
double epsilon, double t)
{
double C = delF * rE * (1.0 / epsilon - 1 ) * t + EQo;
//cout<<"EQ = "<<C<<endl;

return C;
}
```

```
double F1(double Kr, double Q, double I, double t, double EQ, const double rPar[])
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;
double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;
double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;
double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;
double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;
double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;
double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;
double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;
double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;
double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;
double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;
double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;
double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;
double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;
double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;
double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;
double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;
double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;
double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;
double Io = rPar[21]; //cout<<"I = "<<I<<endl;
double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;

double pE = EnergyPrice(Kr, I, Ko, rE, Rr, phi);
double A = exponential(ELo, gEL, t);
double L = exponential(Lo, gL, t);
```

```
double B = exponential(EEo, gEE, t);
double BRHO = pow(B, rho);
double Er = Renew_Energy(Kr, I, Ko, Rr, phi);
double Ef = FF_Energy(Q, EQ, pE, Rf, epsilon, rE);
double tmp = beta * rK * BRHO / (pE * (1.0 - beta));
double tmp2 = pow(tmp, (rho / (rho - 1.0)));

//double Ans = pE - alpha * beta * BRHO * pow((A*L)/(Er + QPrime), (1.0 - alpha))
* pow( beta * BRHO + (1.0 - beta) * tmp2, (alpha / rho - 1.0) );
double Ans = pE - alpha * beta * BRHO * pow((A*L)/(Er + Ef), (1.0 - alpha))
* pow( beta * BRHO + (1.0 - beta) * tmp2, (alpha / rho - 1.0) );
//cout<<"Function 1 = "<<Ans<<endl;

return Ans;
}
double F2(double Kr, double Q, double QPrime, double I, double t, double EQ,
const double rPar[])
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;
double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;
double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;
double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;
double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;
double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;
double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;
double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;
double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;
```

```
double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;
double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;
double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;
double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;
double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;
double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;
double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;
double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;
double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;
double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;
double Io = rPar[21]; //cout<<"I = "<<I<<endl;
double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;

double pE = EnergyPrice(Kr, I, Ko, rE, Rr, phi);
double Cpow = pow(EQ, (1.0 / (1.0 - epsilon) ) );
double tmp = pow(epsilon * pE / rE, (epsilon / (1.0 - epsilon) ) );

double Ans = QPrime - (Rf - Q) * Cpow * tmp;
//cout<<"Function 2 = "<<Ans<<endl;

return Ans;
}
double F3(double Kr, double Q, double I, double IPrime, double t, double EQ,
const double rPar[])
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;
double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;
```

```
double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;
double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;
double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;
double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;
double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;
double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;
double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;
double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;
double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;
double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;
double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;
double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;
double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;
double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;
double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;
double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;
double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;
double Io = rPar[21]; //cout<<"I = "<<I<<endl;
double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;

double pE = EnergyPrice(Kr, I, Ko, rE, Rr, phi);
double Er = Renew_Energy(Kr, I, Ko, Rr, phi);
double Ef = FF_Energy(Q, EQ, pE, Rf, epsilon, rE);
double tmp = pE * Er / Kr - rE;
double tmp2 = (1.0 * epsilon) * pE * Ef / Kr;

double Ans = IPrime - delR*tmp - delRF*tmp2;
//cout<<"Function 3 = "<<Ans<<endl;
```

```
return Ans;

}

double F1_dKr(double Kr, double Q, double I, double t, double EQ, const double rPar[])

{

double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;

double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;

double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;

double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;

double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;

double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;

double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;

double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;

double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;

double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;

double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;

double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;

double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;

double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;

double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;

double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;

double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;

double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;

double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;

double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;

double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;

double Io = rPar[21]; //cout<<"I = "<<I<<endl;

double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;


double A = exponential(ELo, gEL, t);
```

```
double B = exponential(EEo, gEE, t);

double L = exponential(Lo, gL, t);

double pE = EnergyPrice(Kr, I, Ko, rE, Rr, phi); //cout<<"pE = "<<pE<<endl;

double Er = Renew_Energy(Kr, I, Ko, Rr, phi); //cout<<"Er = "<<Er<<endl;

double Ef = FF_Energy(Q, EQ, pE, Rf, epsilon, rE); //cout<<"Ef = "<<Ef<<endl;

double KrPHI = pow(Kr, phi);

double KoPHI = pow(Ko, phi);

double BRHO = pow(B, rho);

double tmp = pow( beta * BRHO * rK / ( (1.0 - beta) * pE) , rho/(rho - 1.0) );

// Same as tmp but no pE.

double tmp2 = pow( beta * BRHO * rK / ( (1.0 - beta)) , rho/(rho - 1.0) );


double dpE_dKr = rE / (Rr * phi) * ( (1.0 + phi) * I * KrPHI / KoPHI + (1.0 - phi)

* KoPHI / (I * KrPHI) + 2.0 );

double dEr_dKr = Rr * phi * I * KoPHI * KrPHI / (Kr * (I * KrPHI + KoPHI)

* (I * KrPHI + KoPHI) );

double dEf_dKr = (Rf - Q) * pow(EQ, 1.0/(1.0 - epsilon))

* pow(epsilon/rE , epsilon/(1.0 - epsilon))

* (epsilon/(1.0 - epsilon)) * pow(pE, epsilon/(1.0 - epsilon) - 1.0) * dpE_dKr;


double h = pow( (Er + Ef), alpha - 1.0);

double dh_dKr = (alpha - 1.0) * pow( (Er + Ef), alpha - 2.0) * (dEr_dKr + dEf_dKr);

double j = pow(beta * BRHO + (1.0 - beta) * tmp, (alpha / rho) - 1.0);

double dj_dKr = ( alpha/rho - 1.0 ) * j / (beta * BRHO + (1.0 - beta) * tmp)

* ( (1.0 - beta) * tmp2 * (rho / (1.0 - rho)) * pow(pE, rho/(1.0 - rho) - 1.0)

* dpE_dKr );

double Ans = dpE_dKr - alpha * beta * BRHO * pow(A*L, 1.0 - alpha)

* ( h * dj_dKr + j * dh_dKr );
```

```
return Ans;
}
double F1_dQ(double Kr,double Q ,double I, double t, double EQ, const double rPar[])
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;
double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;
double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;
double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;
double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;
double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;
double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;
double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;
double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;
double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;
double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;
double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;
double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;
double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;
double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;
double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;
double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;
double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;
double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;
double Io = rPar[21]; //cout<<"I = "<<I<<endl;
double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;

double A = exponential(ELo, gEL, t);
```

```
double B = exponential(EEo, gEE, t);
double L = exponential(Lo, gL, t);
double pE = EnergyPrice(Kr, I, Ko, rE, Rr, phi);
double Er = Renew_Energy(Kr, I, Ko, Rr, phi);
double Ef = FF_Energy(Q, EQ, pE, Rf, epsilon, rE);
double BRHO = pow(B, rho);
double tmp = pow( beta * BRHO * rK / ( (1.0 - beta) * pE) , rho/(rho - 1.0) );

double Ans = (alpha - 1.0) * alpha * beta * BRHO * pow(EQ, 1.0/(1.0 - epsilon) )
* pow(A*L, 1.0 - alpha) * pow(Er + Ef, alpha - 2.0)
* pow(epsilon * pE / rE, epsilon / (1.0 - epsilon))
* pow( beta * BRHO + (1.0 - beta) * tmp, alpha/rho - 1.0);

return Ans;
}
double F1_dI(double Kr, double Q, double I, double t, double EQ, const double rPar[])
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;
double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;
double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;
double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;
double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;
double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;
double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;
double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;
double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;
double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;
```

```
double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;
double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;
double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;
double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;
double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;
double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;
double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;
double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;
double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;
double Io = rPar[21]; //cout<<"I = "<<I<<endl;
double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;

double A = exponential(ELo, gEL, t);
double B = exponential(EEo, gEE, t);
double L = exponential(Lo, gL, t);
double pE = EnergyPrice(Kr, I, Ko, rE, Rr, phi);
double Er = Renew_Energy(Kr, I, Ko, Rr, phi);
double Ef = FF_Energy(Q, EQ, pE, Rf, epsilon, rE);
double KrPHI = pow(Kr, phi);
double KoPHI = pow(Ko, phi);
double BRHO = pow(B, rho);
double tmp = pow( beta * BRHO * rK / ( (1.0 - beta) * pE) , rho/(rho - 1.0) );
// Same as tmp but no pE.
double tmp2 = pow( beta * BRHO * rK / ( (1.0 - beta)) , rho/(rho - 1.0) );

double dpE_dI = (rE / (Rr * phi)) * ( Kr * KrPHI / KoPHI - KoPHI
* Kr / (I * I * KrPHI) );
double dEr_dI = Rr * ( KoPHI * KrPHI / ( (I * KrPHI + KoPHI) * (I * KrPHI + KoPHI) ) );
double dEf_dI = (Rf - Q) * pow(EQ, 1.0/(1.0 - epsilon)) * pow(epsilon/rE ,
```

```
epsilon/(1.0 - epsilon))
* (epsilon/(1.0 - epsilon)) * pow(pE, epsilon/(1.0 - epsilon) - 1.0) * dpE_dI;

double h = pow( (Er + Ef), alpha - 1.0);
double dh_dI = (alpha - 1.0) * pow( (Er + Ef), alpha - 2.0) * (dEr_dI + dEf_dI);
double j = pow(beta * BRHO + (1.0 - beta) * tmp, (alpha / rho) - 1.0);
double dj_dI = ( alpha/rho - 1.0 ) * j / (beta * BRHO + (1.0 - beta) * tmp)
* ( (1.0 - beta) * tmp2 * (rho / (1.0 - rho)) * pow(pE, rho/(1.0 - rho) - 1.0) * dpE_dI );

double Ans = dpE_dI - alpha * beta * BRHO * pow(A*L, 1.0 - alpha)
* ( h * dj_dI + j * dh_dI );

return Ans;
}
double F2_dKr(double Kr, double Q, double I, double t, double EQ, const double rPar[])
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;
double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;
double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;
double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;
double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;
double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;
double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;
double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;
double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;
double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;
double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;
```

```
double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;
double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;
double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;
double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;
double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;
double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;
double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;
double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;
double Io = rPar[21]; //cout<<"I = "<<I<<endl;
double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;
double pE = EnergyPrice(Kr, I, Ko, rE, Rr, phi);
double KrPHI = pow(Kr, phi);
double KoPHI = pow(Ko, phi);

double dpE_dKr = rE / (Rr * phi) * ( (1.0 + phi) * I * KrPHI / KoPHI + (1.0 - phi)
* KoPHI / (I * KrPHI) + 2.0 );

double Ans = - (Rf - Q) * pow(EQ, 1.0 / (1.0 - epsilon) )
* pow( epsilon/rE, epsilon / (1.0 - epsilon) )
* (epsilon / (1.0 - epsilon) ) * pow(pE, epsilon/(1.0 - epsilon) - 1.0) * dpE_dKr;
return Ans;
}
double F2_dQ(double Kr, double I, double EQ, const double rPar[])
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;
double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;
double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;
```

```
double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;
double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;
double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;
double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;
double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;
double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;
double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;
double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;
double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;
double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;
double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;
double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;
double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;
double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;
double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;
double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;
double Io = rPar[21]; //cout<<"I = "<<I<<endl;
double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;

double pE = EnergyPrice(Kr, I, Ko, rE, Rr, phi);
double tmp = 1.0 / (1.0 - epsilon);

double Ans = pow(EQ, tmp) * pow(epsilon * pE / rE, epsilon * tmp);
return Ans;
}
double F2_dI(double Kr, double Q, double I, double t, double EQ, const double rPar[])
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
```

```
double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;

double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;

double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;

double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;

double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;

double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;

double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;

double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;

double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;

double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;

double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;

double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;

double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;

double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;

double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;

double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;

double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;

double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;

double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;

double Io = rPar[21]; //cout<<"I = "<<I<<endl;

double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;


double pE = EnergyPrice(Kr, I, Ko, rE, Rr, phi);

double KrPHI = pow(Kr, phi);

double KoPHI = pow(Ko, phi);


double dpE_dI = (rE / (Rr * phi)) * ( Kr * KrPHI / KoPHI - KoPHI * Kr / (I * I * KrPHI) );


double Ans = - (Rf - Q) * pow(EQ, 1.0 / (1.0 - epsilon) )
```

```
* pow( epsilon/rE, epsilon / (1.0 - epsilon) )
* (epsilon / (1.0 - epsilon) ) * pow(pE, epsilon/(1.0 - epsilon) - 1.0) * dpE_dI;
return Ans;
}
double F3_dKr(double Kr, double Q, double I, double t, double EQ, const double rPar[])
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;
double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;
double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;
double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;
double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;
double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;
double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;
double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;
double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;
double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;
double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;
double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;
double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;
double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;
double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;
double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;
double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;
double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;
double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;
double Io = rPar[21]; //cout<<"I = "<<I<<endl;
double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;
```

```
double pE = EnergyPrice(Kr, I, Ko, rE, Rr, phi);
double Er = Renew_Energy(Kr, I, Ko, Rr, phi);
double Ef = FF_Energy(Q, EQ, pE, Rf, epsilon, rE);
double KrPHI = pow(Kr, phi);
double KoPHI = pow(Ko, phi);

double dpE_dKr = rE / (Rr * phi) * ( (1.0 + phi) * I * KrPHI / KoPHI + (1.0 - phi)
* KoPHI / (I * KrPHI) + 2.0 );
double dErKr_dKr = I * KrPHI * ( (phi - 1.0) * KoPHI - I * KrPHI ) / ( Kr * Kr
* ( I * KrPHI + KoPHI) );
double dEfKr_dKr = (Rf - Q) * pow(EQ, 1.0 / (1.0 - epsilon))
* pow(epsilon/rE, epsilon/(1.0 - epsilon)) * ( epsilon/(1.0 - epsilon)
* pow(pE, epsilon/(1.0 - epsilon) - 1)
 * dpE_dKr / Kr - pow(pE, epsilon/(1.0 - epsilon)) / (Kr * Kr) );

double Ans = - delR * (pE * dErKr_dKr + Er * dpE_dKr / Kr) - delRF
* (1.0 - epsilon) * (pE * dEfKr_dKr + Ef * dpE_dKr / Kr);
return Ans;
}
double F3_dQ(double Kr, double I, double t, double EQ, const double rPar[])
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;
double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;
double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;
double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;
double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;
```

```
double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;
double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;
double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;
double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;
double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;
double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;
double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;
double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;
double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;
double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;
double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;
double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;
double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;
double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;
double Io = rPar[21]; //cout<<"I = "<<I<<endl;
double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;

double pE = EnergyPrice(Kr, I, Ko, rE, Rr, phi);
double tmp = 1.0 / (1.0 - epsilon);

double dEf_dQ = - pow(EQ, tmp) * pow(epsilon * pE / rE, epsilon * tmp);

double Ans = delRF * (1.0 - epsilon) * pE * dEf_dQ / Kr;
return Ans;
}
double F3_dI(double Kr, double Q, double I, double t, double EQ, const double rPar[])
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
```

```
double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;

double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;

double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;

double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;

double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;

double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;

double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;

double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;

double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;

double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;

double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;

double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;

double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;

double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;

double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;

double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;

double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;

double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;

double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;

double Io = rPar[21]; //cout<<"I = "<<I<<endl;

double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;


double pE = EnergyPrice(Kr, I, Ko, rE, Rr, phi);

double Er = Renew_Energy(Kr, I, Ko, Rr, phi);

double Ef = FF_Energy(Q, EQ, pE, Rf, epsilon, rE);

double KrPHI = pow(Kr, phi);

double KoPHI = pow(Ko, phi);

double tmp = 1.0/(1.0 - epsilon);
```

```
double dpE_dI = (rE / (Rr * phi)) * ( Kr * KrPHI / KoPHI - KoPHI * Kr / (I * I * KrPHI) );
double dEr_dI = Rr * ( KoPHI * KrPHI / ( (I * KrPHI + KoPHI) * (I * KrPHI + KoPHI) ) );
double dEf_dI = (Rf - Q) * pow(EQ, tmp) * pow(epsilon/rE , epsilon * tmp)
* (epsilon * tmp) * pow(pE, epsilon * tmp - 1.0) * dpE_dI;

double Ans = - (delR / Kr) * (pE * dEr_dI + Er * dpE_dI) - (delRF * (1.0 - epsilon) / Kr)
* (pE * dEf_dI + Ef * dpE_dI);
return Ans;
}
double NewtonsMethod(double Kr, double Q, double I, double EQ, double t,
double TOL, const double rPar[])
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
double delf = rPar[2]; //cout<<"delf = "<<delf<<endl;
double delrf = rPar[3]; //cout<<"delrf = "<<delrf<<endl;
double delr = rPar[4]; //cout<<"delr = "<<delr<<endl;
double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;
double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;
double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;
double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;
double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;
double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;
double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;
double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;
double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;
double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;
double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;
double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;
```

```
double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;

double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;

double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;

double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;

double Io = rPar[21]; //cout<<"I = "<<I<<endl;

double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;

double DBG = rPar[23]; //cout<<"Debug mode = "<<DBG<<endl;


ofstream oNM;

if(DBG == 1.0)

{

oNM.open("NewtonsMethod_Intermediate_Steps.txt", ios::out);

oNM<<"Q\t"<<Q<<"\tI\t"<<I<<endl;

oNM<<"Kr\tF1\tdF1/dKr\tAdjustment\n";

}

int cnt = 0;

int MAX = 500;

double K_Next = Kr;

double K_Prev = 0.0;


double dist = 1.0;


while(dist > TOL && cnt <= MAX)

{

cnt++;

K_Prev = K_Next;


double f = F1(K_Next, Q, I, t, EQ, rPar);

double fPrime = F1_dKr(K_Next, Q, I, t, EQ, rPar);
```

```
K_Next = K_Prev - f / fPrime;
dist = abs(K_Next / K_Prev - 1.0);

if(DBG == 1.0)
{
oNM<<K_Prev<<"\t"<<f<<"\t"<<fPrime<<"\t"<< -f/fPrime <<endl;
}
}
oNM.close();
oNM.clear();
double Ans = K_Next;
return Ans;
}
vector<double> T2R_RK4(double Kr, double Q, double I, double time,
const double rPar[], double h, double TOL, int EQ_Switch)
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;
double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;
double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;
double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;
double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;
double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;
double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;
double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;
double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;
double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;
```

```
double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;
double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;
double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;
double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;
double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;
double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;
double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;
double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;
double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;
double Io = rPar[21]; //cout<<"I = "<<I<<endl;
double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;
double DBG = rPar[23]; //cout<<"Debug mode = "<<DBG<<endl;

ofstream oRK4;
if(DBG == 1.0)
{
string filename = "Runge-Kutta_Intermediate_Steps.txt";
oRK4.open(filename.c_str(), ios::out);
oRK4<<"Q\tQP\tI\tIP\tKr\tEQ\tpE\n";
}
double EQ1, EQ2, EQ3, EQ4;
double I1, I2, I3, I4;
double IP1, IP2, IP3, IP4;

double t = time - h;
double t2 =  (t + h/2);
double t4 = (t + h);

if(EQ_Switch == 0)
```

```
{
EQ1 = exponential(EQo, gEQ, t);
I1 = exponential(Io, gI, t);
}
else if(EQ_Switch == 1)
{
EQ1 = delF * rE * ( (1.0 / epsilon) - 1.0) * t + EQo;
I1 = I;
}
else{}
double pE1 = EnergyPrice(Kr, I1, Ko, rE, Rr, phi);
double QP1 = FF_Energy(Q, EQ1, pE1, Rf, epsilon, rE);
IP1 = - F3(Kr, Q, I1, 0.0, t, EQ1, rPar);


if(DBG == 1.0)
{
oRK4<<Q;
oRK4<<"\t"<<QP1;
oRK4<<"\t"<<I1;
oRK4<<"\t"<<IP1;
oRK4<<"\t"<<Kr;
oRK4<<"\t"<<EQ1;
oRK4<<"\t"<<pE1<<endl;
}
ISNAN("RK4 has failed as QPrime(t)",QP1);
/////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////
if(EQ_Switch == 0)
{
```

```
EQ2 = exponential(EQo, gEQ, t2);
I2 = exponential(Io, gI, t2);
}
else if(EQ_Switch == 1)
{
EQ2 = delF * rE * ( (1.0 / epsilon) - 1.0) * t2 + EQo;
I2 = (I + h*IP1/2);
}
else{}
double Q2 = (Q + h*QP1/2);
// newton's method at K2
double Kr2 = NewtonsMethod(Kr, Q2, I2, EQ2, t, TOL, rPar);
double pE2 = EnergyPrice(Kr2, I2, Ko, rE, Rr, phi);
double QP2 = FF_Energy(Q2, EQ2, pE2, Rf, epsilon, rE);
IP2 = - F3(Kr2, Q2, I2, 0.0, t2, EQ2, rPar);

if(DBG == 1.0)
{
oRK4<<Q2;
oRK4<<"\t"<<QP2;
oRK4<<"\t"<<I2;
oRK4<<"\t"<<IP2;
oRK4<<"\t"<<Kr2;
oRK4<<"\t"<<EQ2;
oRK4<<"\t"<<pE2<<endl;
}
ISNAN("RK4 has failed as QPrime(t+1/2)",QP2);
/////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////
```

```
if(EQ_Switch == 0)
{
EQ3 = exponential(EQo, gEQ, t2);
I3 = exponential(Io, gI, t2);
}
else if(EQ_Switch == 1)
{
EQ3 = delF * rE * ( (1.0 / epsilon) - 1.0) * t2 + EQo;
I3 = (I + h*IP2/2);
}
else{}
double Q3 = (Q + h*QP2/2);
double Kr3 = NewtonsMethod(Kr2, Q3, I3, EQ3, t2, TOL, rPar);
double pE3 = EnergyPrice(Kr3, I3, Ko, rE, Rr, phi);
double QP3 = FF_Energy(Q3, EQ3, pE3, Rf, epsilon, rE);
IP3 = -F3(Kr3, Q3, I3, 0.0, t2, EQ3, rPar);

if(DBG == 1.0)
{
oRK4<<Q3;
oRK4<<"\t"<<QP3;
oRK4<<"\t"<<I3;
oRK4<<"\t"<<IP3;
oRK4<<"\t"<<Kr3;
oRK4<<"\t"<<EQ3;
oRK4<<"\t"<<pE3<<endl;
}
ISNAN("RK4 has failed as QPrime(t+1/2)",QP3);
/////////////////////////////////////////////////////////
```

```
/////////////////////////////////////////////////////////
if(EQ_Switch == 0)
{
EQ4 = exponential(EQo, gEQ, t4);
I4 = exponential(Io, gI, t4);
}
else if(EQ_Switch == 1)
{
EQ4 = delF * rE * ( (1.0 / epsilon) - 1.0) * t4 + EQo;
I4 = (I + h*IP3);
}
else{}
double Q4 = (Q + h*QP3);
double Kr4 = NewtonsMethod(Kr3, Q4, I4, EQ4, t4, TOL, rPar);
double pE4 = EnergyPrice(Kr4, I4, Ko, rE, Rr, phi);
double QP4 = FF_Energy(Q4, EQ4, pE4, Rf, epsilon, rE);
IP4 = -F3(Kr4, Q4, I4, 0.0, t4, EQ4, rPar);

if(DBG == 1.0)
{
oRK4<<Q4;
oRK4<<"\t"<<QP4;
oRK4<<"\t"<<I4;
oRK4<<"\t"<<IP4;
oRK4<<"\t"<<Kr4;
oRK4<<"\t"<<EQ4;
oRK4<<"\t"<<pE4<<endl;
}
ISNAN("RK4 has failed as QPrime(t+1)",QP4);
```

```
oRK4.close();
oRK4.clear();

// Time-step Q
double Q_new = Q + (h/6) * (QP1 + 2 * QP2 + 2 * QP3 + QP4);
double I_new = I + (h/6) * (IP1 + 2 * IP2 + 2 * IP3 + IP4);
double Kr_new = NewtonsMethod(Kr4, Q_new, I_new, EQ4, t4, TOL, rPar);

vector<double> SOL;
SOL.push_back(Kr_new);
SOL.push_back(Q_new);

if(EQ_Switch == 1)
SOL.push_back(I_new);

return SOL;
}
//////////////////////////////////////////////////////////////
////////////   FUNCTIONS FOR DASSL    ///////////////////
//////////////////////////////////////////////////////////////
void jac(const double& time, const double y[], const double yPrime[],
double** PD, double& CJ, double rPar[], int iPar[])
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;
double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;
double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;
```

```
double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;

double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;

double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;

double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;

double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;

double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;

double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;

double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;

double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;

double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;

double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;

double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;

double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;

double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;

double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;

double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;

double Io = rPar[21]; //cout<<"I = "<<I<<endl;

double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;


int EQ_Switch = iPar[0]; //cout<<"EQ_Switch = "<<EQ_Switch<<endl;

double t = time;


if(EQ_Switch == 0) // exponentials

{

double Kr = y[0];

double Q = y[1];

double QP = yPrime[1];

double I = exponential(Io, gI, t);

double EQ = exponential(EQo, gEQ, t);
```

```
PD[0][0] = F1_dKr(Kr, Q, I, t, EQ, rPar);
PD[0][1] = F1_dQ(Kr, Q, I, t, EQ, rPar);
PD[1][0] = F2_dKr(Kr, Q, I, t, EQ, rPar);
PD[1][1] = F2_dQ(Kr, I, EQ, rPar) + CJ; // dF2/dQ' = 1
}
else if(EQ_Switch == 1) // ODEs
{
double Kr = y[0];
double Q = y[1];
double I = y[2];
double QP = yPrime[1];
double IP = yPrime[2];
double EQ = ExtractionEfficiency(EQo, delF, rE, epsilon, t);
PD[0][0] = F1_dKr(Kr, Q, I, t, EQ, rPar);
PD[0][1] = F1_dQ(Kr, Q, I, t, EQ, rPar);
PD[0][2] = F1_dI(Kr, Q, I, t, EQ, rPar);
PD[1][0] = F2_dKr(Kr, Q, I, t, EQ, rPar);
PD[1][1] = F2_dQ(Kr, I, EQ, rPar) + CJ; // dF2/dQ' = 1
PD[1][2] = F2_dI(Kr, Q, I, t, EQ, rPar);
PD[2][0] = F3_dKr(Kr, Q, I, t, EQ, rPar);
PD[2][1] = F3_dQ(Kr, I, t, EQ, rPar);
PD[2][2] = F3_dI(Kr, Q, I, t, EQ, rPar) + CJ; // dF3/dI' = 1
}
else{}
};
void RES(const double& time, const double y[], const double yPrime[],
double residue[], int& iRes, const double rPar[], const int iPar[])
{
double Rf = rPar[0]; //cout<<"Rf = "<<Rf<<endl;
```

```
double Rr = rPar[1]; //cout<<"Rr = "<<Rr<<endl;
double delF = rPar[2]; //cout<<"delf = "<<delf<<endl;
double delRF = rPar[3]; //cout<<"delrf = "<<delrf<<endl;
double delR = rPar[4]; //cout<<"delr = "<<delr<<endl;
double rE = rPar[5]; //cout<<"rE = "<<rE<<endl;
double rK = rPar[6]; //cout<<"rK = "<<rK<<endl;
double Ko = rPar[7]; //cout<<"Ko = "<<Ko<<endl;
double alpha = rPar[8]; //cout<<"alpha = "<<alpha<<endl;
double beta = rPar[9]; //cout<<"beta = "<<beta<<endl;
double epsilon = rPar[10]; //cout<<"epsilon = "<<epsilon<<endl;
double phi = rPar[11]; //cout<<"phi = "<<phi<<endl;
double rho = rPar[12]; //cout<<"rho = "<<rho<<endl;
double ELo = rPar[13]; //cout<<"ELo = "<<ELo<<endl;
double gEL = rPar[14]; //cout<<"gEL = "<<gEL<<endl;
double EEo = rPar[15]; //cout<<"EEo = "<<EEo<<endl;
double gEE = rPar[16]; //cout<<"gEE = "<<gEE<<endl;
double Lo = rPar[17]; //cout<<"Lo = "<<Lo<<endl;
double gL = rPar[18]; //cout<<"gL = "<<gL<<endl;
double EQo = rPar[19]; //cout<<"EQo = "<<EQo<<endl;
double gEQ = rPar[20]; //cout<<"gEQ = "<<gEQ<<endl;
double Io = rPar[21]; //cout<<"I = "<<I<<endl;
double gI = rPar[22]; //cout<<"gI = "<<gI<<endl;

int EQ_Switch = iPar[0]; //cout<<"EQ_Switch = "<<EQ_Switch<<endl;

double Kr = y[0]; //cout<<"Kr = "<<Kr<<endl;
double Q = y[1]; //cout<<"Q = "<<Q<<endl;

double Q_prime = yPrime[1]; //cout<<"Q' = "<<Q_prime<<endl;
```

```
double t = time; //cout<<"time = "<<t<<endl;


double I, I_prime, EQ, DELTA_C;
if(EQ_Switch == 0)
{
I = exponential(Io, gI, time);
EQ = exponential(EQo, gEQ, time);
}
else if(EQ_Switch == 1)
{
I = y[2]; //cout<<"I = "<<I<<endl;
I_prime = yPrime[2]; //cout<<"I' = "<<I_prime<<endl;
EQ = ExtractionEfficiency(EQo, delF, rE, epsilon, time);
DELTA_C = F3(Kr, Q, I, I_prime, time, EQ, rPar);
residue[2] = DELTA_C;
}
double DELTA_A = F1(Kr, Q, I, time, EQ, rPar);
double DELTA_B = F2(Kr, Q, Q_prime, I, time, EQ, rPar);


residue[0] = DELTA_A;
residue[1] = DELTA_B;
};
```

Bibliography

[1] Akimoto, K., Matsunaga, A., Fujii, Y., and Yamaji, K. *Game theoretic analysis for carbon emission permits trading among multiple world regions with an optimization global energy model*. Electrical Engineering in Japan, Vol. 131, No. 2 (2000).

[2] Berg, P., Hanz, P., and Milton, I. *An energy-economic oil production model*. IMA Journal of Applied Mathematics, (2011) 1-26. doi:10.1093/imamat/hxr049

[3] Brenan, K. E., Campbell, S. L., and Petzold, L. E. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. SIAM, 1996.

[4] British Petroleum. *Statistical review of world energy*. http://www.BP.com

[5] Central Intelligence Agency. *CIA world fact book: GDP - composition by sector* https://www.cia.gov/library/publications/the-world-factbook/index.html

[6] Cobb, C. W., and Douglas, P. H. *A theory of production*. The American Economic Review, Vol. 18.1 (1928) 139-165.

[7] Dasgupta, P., and Heal, G. *The optimal depletion of exhaustable resources*. Review of Economic Studies, Vol. 41 (1974) 3-28.

[8] Douglas Faires, J., and Burden, R. *Numerical Methods*. Brooks/Cole Thompson Learning Inc., 3rd ed. (2003) 193-200.

[9] Hartwick, J. *Intergenerational equity and the investing of rents from exhaustible resources*. The American Economic Review, Vol. 67 (1977) 972-974.

[10] Hubbert, M. K. *Nuclear energy and the fossil fuels.* Drilling and Production Practice, American Petroleum Institute, (1956) 7-25.

[11] Kemfert, C. *Estimated substitution elasticities of a nested CES production function approach for Germany.* Energy Economics 20, (1998) 249-264.

[12] Krabs, W., and Pickl, S. *A game-theoretic treatment of a time discrete emission reduction model.* International Game Theory Review, Vol. 6, No. 1, (2004) 21-34.

[13] Milton, D. I. M. *Applications and nummerical investigations of differential algebraic equations.* Master of Science thesis, UOIT, (2010).

[14] Nordhaus, W. D., and Yang, Z. *A regional dynamic general-equilibrium model of alternative climate change strategies.* The American Economic Review, Vol. 86, No. 4, (1996) 741-765.

[15] Solow, R. M. *A contribution to the theory of economic growth.* The Quarterly Journal of Economics, Vol. 70, No. 1, (1956) 65-94.

[16] Solow, R. M. *Resources and economic growth.* The American Economist, Vol. 22, No. 2, (1978) 5-11.

[17] Stiglitz, J. E. *Growth with exhaustable natural resources: the competitive economy.* The Review of Economic Studies, Vol. 41, (1974) 139-152.

[18] Tahvonen, O., and Salo, S. *Economic growth and transitions between renewable and nonrenewable energy resources.* European Economic Review 45, (2001) 1379-1398.

[19] Sumalia, U. R., Dinar, A., and Albiac, J. *Game theoretic applications to environmental and natural resource problems.* Environment and Development Economics 14, (2008) 1-5.

[20] Weitzman, M. L., *Pricing the limits to growth from minerals depletion.* The Quarterly Journal of Economics, Vol. 114, No. 2, (1999) 691-706.

[21] World Bank. *World bank development indicators.* http://www.worldbank.org