The Fertilizer Use Optimizer

Solver and Macros Programming Manual

Jim Jansen and Charles Wortmann

University of Nebraska-Lincoln
Table of Contents

CHAPTER 1: Introduction and Background ........................................................................... 3
  1.1 Background .................................................................................................................... 3
  1.2 The Research ............................................................................................................... 3
  1.3 The Uganda Fertilizer Optimization Tool ................................................................. 4
  1.4 Wider Applications ....................................................................................................... 4
  1.5 Layout of Solver and Macros Programming Manual .................................................. 4

CHAPTER 2: Solver Basics and Applications ...................................................................... 5
  2.1 Optimization Basics ...................................................................................................... 5
    2.1.1 Process Overview .................................................................................................... 5
    2.1.2 Optimization Mechanics ....................................................................................... 5
  2.2 Microsoft Solver .......................................................................................................... 7
    2.2.1 Adding Solver Add-in to Excel ............................................................................. 7
    2.2.2 Terminology of Solver .......................................................................................... 9
  2.3 Application of Solver .................................................................................................... 13
    2.3.1 Example Application ............................................................................................. 13
    2.3.2 Fertilizer Optimizer ............................................................................................... 22

CHAPTER 3: Macro Program Basics .................................................................................... 38
  3.1 Accessing Macro Code ................................................................................................ 38
    3.1.1 Adding Developer Tab to Excel ............................................................................ 38
    3.1.2 Unlocking Fertilizer Optimizer Macros ................................................................. 39
  3.2 Structure of Macro Programs ...................................................................................... 43
    3.2.1 Visual Basic for Application Editor ..................................................................... 43
    3.2.2 Layout of Code ...................................................................................................... 46
    3.2.3 Command Statements ......................................................................................... 48

CHAPTER 4: Fertilizer Optimizer Macro .............................................................................. 54
  4.1 Fix_Land_Area_Req_Checkbox Macro ........................................................................ 54
  4.2 Help Macro ................................................................................................................... 56
  4.3 Hide_Protect Macro ..................................................................................................... 57
  4.4 Optimize_Solver Macro ............................................................................................... 58
  4.5 Print_Output Macro ...................................................................................................... 62
  4.6 Reset_Form Macro ..................................................................................................... 63
CHAPTER 1: Introduction and Background

The Ugandan Fertilizer Optimization Tool determines the optimal crop-nutrient-rate combination for maximizing net returns on investment in fertilizer use, especially for finance constrained situations. Crop specific nutrient response functions are optimized in a Microsoft Excel spreadsheet using the Solver add-in for each smallholder to maximize returns on their limited investment.

1.1 Background

Low commercial fertilizer use by smallholder farmers in developing regions of the world commonly constrains productivity. Many of these farmers do not have the financial capacity to purchase enough fertilizer to maximize net returns on their limited investment per hectare. High fertilizer costs and low commodity prices often reduce profit potential. Competing needs for money often take priority. Such farmers need high net returns on their investments to justify the application of fertilizers.

Recommendations for non-finance constrained fertilizer use commonly strive to maximize mean net returns across all planted acres. These recommendations are infeasible for smallholders with limited financial capacities. Maximizing net returns requires the fertilizer investments focus on crop-nutrient with the highest marginal returns until the budgeted financial resources are exhausted.

![Nutrient Response Functions](image)

**Figure 1.** Nutrient Response Functions

1.2 The Research

A collaborative research team in Uganda led by Dr. Crammer Kayuki Kaizzi of the Ugandan National Agriculture Research Organization, financially supported by the Alliance of a Green Revolution in Africa (AGRA), and conducted 80 field trials to determine nutrient response functions for 15 crop-nutrient combinations. These functions were for corn, sorghum, upland rice, dry bean, soybean and peanut (Kaizzi et al., 2012 a,b,c). Some crop-nutrient combinations were
more profitable than others (Figure 1). The response functions were curvilinear and the figure also illustrates the effect of application rate on profitability and implies a need to determine combinations of crop-nutrient-rate that will give the best net return on the amount of fertilizer that the farmer can afford to use.

The fertilizer response relationships displayed in Figure 1 can be used to prioritize crop-nutrient-rate options. Depending on which crops the farmer wishes to plant, application of a low rate of N to upland rice and bean may be of highest priority if the financial constraint is severe. With a less severe financial constraint, the priority options include additional N applied to rice and bean, some N applied to maize and sorghum, and some P applied soybean and groundnut. With no financial constraint, fertilizer should be applied for each crop-nutrient combination that maximizes net return per hectare for the given fertilizer cost to commodity value ratios.

1.3 The Uganda Fertilizer Optimization Tool

To enable full optimization across the 15 crop-nutrient response functions, the Excel-Solver based Uganda Fertilizer Optimization Tool was developed by the Agricultural Economics and Agronomy and Horticulture Departments at UNL (http://cropwatch.unl.edu/web/soils/home/). The Tool considers the land area that the farmer wishes to plant to each crop, expected commodity values at harvest, the costs of fertilizer use, and the budget constraint. The output includes the recommended fertilizer rate for each crop and the expected effects on crop yields and net returns.

Using the Tool when the financial constraint is moderate to high, the estimated net returns to fertilizer use are typically greater than twice as much as when fertilizer is applied to maximize net returns per hectare. The greater potential for profitability with the Tool is expected to enable finance constrained farmers to gradually break out of poverty and increase fertilizer use to the point of maximizing net returns per hectare. This fertilizer use optimization approach was introduced to 60 government and non-government extension staff in Uganda with training for the remaining extension staff planned.

1.4 Wider Applications

This approach has the ability to increase the profitability of fertilizer use for finance-constrained crop production throughout Sub-Saharan Africa and on other continents. The crop-nutrient response functions will need to be determined for the appropriate crops of the agro-ecological zone in which the Tool would be applied. This manual explains the process necessary to update and adapt the Tool to a region different than Uganda.

1.5 Layout of Solver and Macros Programming Manual

The following four sections of this manual explain: Chapter 2 – installation and use of the Solver add-in with application to the Fertilizer Optimization Tool, Chapter 3 – installation, development, and use of Macro Programming to automate features in Excel, Chapter 4 – editing Macro Programs used in the Tool, and Chapter 5 – trouble shooting Macro problems and additional references.
CHAPTER 2: Solver Basics and Applications

The Ugandan Fertilizer Optimization Tool runs on the Microsoft Excel 2007 or 2010 platform with the Solver add-in to determine optimal crop-nutrient combinations. Chapter 2 explains the basic process of an optimization, the installation steps of Solver, functions of the add-in, and applications of the Tool.

2.1 Optimization Basics

A reiterative process is performed each time the Tool runs an optimization for a user. The following two sections give an overview of this process and explain the mechanics of an optimization.

2.1.1 Process Overview

The process stage of the Ugandan Fertilizer Optimization Tool considers the farmer input data or farmer specified constraints, pre-determined model constraints, and the model’s optimization mode (Figure 2). The farmer imposed constraints include: i) the expected land area to be planted and predicted value at harvest for each crop; ii) fertilizers available; iii) the cost of using each; and iv) the farmer’s budget constraint. The model is constrained to avoid exceeding the range of inference for the underlying equations with maximum and minimum fertilizer amount limits imposed by the model for the crop-nutrient response functions.

![Fertilizer Optimization Diagram](image)

**Figure 2.** Operational flow model of the fertilizer optimization tool developed for Uganda.

Maximums prevent the amount of a specific nutrient recommended for a crop-nutrient function from exceeding the nutrient rate required for the yield response to plateau. Minimum nutrient application rates of zero kg ha\(^{-1}\) for all crop-nutrient response functions prevent a non-negativity constraint of the objective function. Also, the Tool requires some N application before P can be applied to cereals and bean, and some P application before K can be applied to soybean and groundnut.

2.1.2 Optimization Mechanics

The reiterative process performed by the Tool using Microsoft Solver add-in to reiteratively search for a solution that optimizes a specified mathematical function, often referred to as an objective
function, subject to specified constraints. The objective function in this case is to maximize net returns to fertilizer use as the difference of added crop revenue and added fertilizer costs, subject to farmer input imposed constraints and internal constraints of the tool. The 15 crop-nutrient response functions are combined with fertilizer use, costs, and expected crop values to estimate expected net income given investment limitations until the financial resource is exhausted. The optimizer selects the crop-nutrient-rate combinations that deliver the highest net return on investment. The selection of the crop-nutrient-rate combinations relate to a circular reference where each combination must satisfy all constraints imposed by the user and tool.

The tool achieves the objective function of maximizing total expected net returns to fertilizer use by determining the optimal combination of crop-nutrient-rates subject to the budget and response function constraints. The costs for the total amount of fertilizer recommended cannot exceed the financial resources available for investment. Once the optimal crop-nutrient-rate combinations have been determined, the results are displayed including the optimized crop-fertilizer application rates for the 15 possible crop-nutrient combinations, expected effects on yield and net returns to fertilizer use, and total expected net returns to investment in fertilizer use. Each set of constraints imposed by the user delivers a unique, but optimized solution based upon attributes pertinent to the farmer’s operation.
2.2 Microsoft Solver

The Solver add-in must be enabled for the Fertilizer Optimization spreadsheet to function appropriately. When enabled the Solver add-in appears under the Data tab on the Quick Access Toolbar.

![Solver Add-in and Data Tab](image)

2.2.1 Adding Solver Add-in to Excel

The following series of steps show how to add the Solver Add-in to the Data tab on the Quick Access Toolbar in Excel:

1) Select the File tab on the Quick Access Toolbar
2) Select Options on File drop down menu

![Step 1 - File Tab](image)

![Step 2 - Options](image)
3) Select Add-Ins on the left hand side of the Excel Options window
4) In the Add-Ins drop down list select the Solver Add-in options
5) Select Go

After selecting Go the Add-Ins options window appears
6) Select Solver Add-in
7) Press OK

The Solver Add-in appears on the Data tab of the Quick Access Tool Bar.

2.2.2 Terminology of Solver

Clicking the Data tab and Solver Add-in button brings up the Solver Parameters window.

![Solver Parameters Window](image)

The Solver Parameters window shows the major portions of the optimization engine. Major sections in the Tool include:

**Set Objective** – identifies the target cell containing the object functions for the model.

**To: Max/Min/Value Of** – determines the goal solution value, whether a Max (maximizing profits), Min (Minimizing Costs), or a Value Of a certain amount.

**By Changing Variable Cells** – dictates the cells which may be reiteratively changed until the object function achieves the Max/Min/Value Of specified by the user.
Subject to the Constraints – mathematical constraints placed upon the objective function that restricts values the changing variable cells may assume.

Select a Solving Method – specifies the solving algorithm used to perform an optimization.

Solve – runs the optimization algorithm to reiteratively change the variable cells subject to constraints set by the user until the search for the max/min/value of solution has been optimally determined.

These sections are the major areas of the Solver Parameters window. To add constraints to the Solver Parameters window, the user must click Add.

Clicking Add on the Solver Parameters window opens the Add Constraint window.

To enter in constraints to the Solver Parameters window, the Cell Reference(s) must establish a relationship (<=, =, >=, int, bin, or dif ) to the Constraint(s) by selecting an option from the drop down constraint box list. Once the relationship has been established, selecting Add on the Add Constraint window uploads this constraint to the Solver Parameters window. If any additional constraints need to be added, the same processes is repeated. Selecting Cancel returns the user back to the Solver Parameters window.

If an error was made while inputting a constraint, selecting Change on the Solver Parameters window displays the Change Constraint window.

The Change Constraint window allows for Cell References, relationships, or constraints to be readjusted. Once the changes have been made, selecting Add brings the new relationships to the Solver Parameters window. Selecting Cancel returns the user back to the Solver Parameters window.

Other options under the Subject to the Constraints section include Delete and Reset All. While having a constraint selected, pressing Delete removes the selected item from the Solver
Parameters window. Also, selecting Reset All removes all cells specified in the Set Objective, By Changing Variables Cells, and Subject to the Constraints.

Depending upon the type of optimization problem, the Select a Solving Method may need to be changed. GRG Nonlinear is Solver’s default solving method and the process used in the Fertilizer Optimizer. When a user selects Solve on the Solver Parameters window, the optimization algorithm runs and searches for a solution that satisfies the object function’s goal subject to the set of constraints for the problem.

After running the optimization, the Solver Results window appears.

Summary information displayed in the Solver Results window indicates whether the optimization algorithm found an appropriate solution satisfying all constraints or if a feasible solution could not be found. In either case the user can choose to Keep Solver Solution or Restore Original Values. Selecting Keep Solver Solution and OK returns the user to the spreadsheet with the optimized values. Choosing Restore Original Values and OK removes any optimization solution and restores the initial objective function values prior to running Solver.

After running the optimization and Solver fails to find an optimized solution the following message is displayed.
In the event that Solver fails to find an appropriate solution, the Solver Results dialog box displays the message that Solver could not find a feasible solution satisfying all constraints. Solver usually fails to find an appropriate solution due to either the number or type of constraints being applied to the optimization. Reducing the number and type of constraints may allow Solver to find an appropriate solution. Also, lessening the severity of constraints present in the optimization may also allow Solver to find an optimized solution. In the event Solver cannot find an appropriate solution, pressing Cancel allows the user to return to the Solver Parameters window to review and possibly make changes to the constraints present in the optimization.

Solver is a powerful tool which allows a user to optimize an objective function given a set of constraints confining the set of appropriate solutions. To further explain how Solver works with Excel based optimization problems Section 2.3 evaluates how to perform optimizations and evaluate the results using this tool.
2.3 Application of Solver

The Solver add-in allows for the maximization of an objective function given a set of constraints limiting the range of solutions. After completing Section 2.3, the Solver add-in should be linked with an example optimization problem and the Fertilizer Optimizer. Also, results from these optimizations are evaluated to determine the validity of the solutions.

2.3.1 Example Application

To explain how Solver functions in the Fertilizer Optimizer the following example provides a simplified version of the Tool involving maize with a nitrogen response function. The Excel file titled Example Optimization contains the simplified maize model having only the crop’s nitrogen response function used in the optimization with Solver. Opening this file should display an image similar to the one below.

![Example Optimization Excel File](image)

Major sections of this Excel file include the crop input parameters, response function calculations, and the optimized results section.
The crop input parameters section includes the Crop Selection and Prices, Fertilizer Selection and Prices, and Budget Constraint Tables.

<table>
<thead>
<tr>
<th>Crop Selection and Prices</th>
<th>Crop</th>
<th>Area Planted (Ha)</th>
<th>Expected Grain Value/kg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maize</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Fertilizer Selection and Prices</th>
<th>Fertilizer Product</th>
<th>N</th>
<th>Price/Kg 50 bag</th>
</tr>
</thead>
<tbody>
<tr>
<td>Urea</td>
<td>46%</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Budget Constraint</th>
<th>Amount available to invest in fertilizer</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>

Similar to the input section of the Fertilizer Optimizer both of these areas have headings with a yellow background color.

Input data from the crop input parameters section links to the response function calculations on the right hand side of the Excel file. The response function calculations include the Fertilizer Nutrient and Price, Fertilizer Constraints, Fertilizer Application Rates, Crop and Yield Responses, and Return and Expenses Tables.

<table>
<thead>
<tr>
<th>Fertilizer Nutrient and Price</th>
<th>Fertilizer</th>
<th>Nitrogen</th>
<th>Price/kg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Urea</td>
<td></td>
<td>46%</td>
<td>3.5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Fertilizer Constraints</th>
<th>Crop</th>
<th>Min Rate</th>
<th>Max Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maize N</td>
<td></td>
<td>0</td>
<td>100</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Fertilizer Application Rates</th>
<th>Crop</th>
<th>Ha Planted</th>
<th>Application Rate</th>
<th>N App Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maize</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Crop and Yield Responses</th>
<th>Function</th>
<th>Constant</th>
<th>Coefficient 1</th>
<th>Coefficient 2</th>
<th>Base Production</th>
<th>Exp Change</th>
<th>Exp Yld/kg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maize N</td>
<td>3.9287</td>
<td>2.1367</td>
<td>0.9476</td>
<td>1.79</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Returns and Expenses</th>
<th>Function</th>
<th>Net Value</th>
<th>Fert Cost</th>
<th>Net Return</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maize N</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
</tr>
</tbody>
</table>

Calculations preformed in the Fertilizer Optimizer have a similar series of tables to determine the optimal application rates.
Finally, the optimized results section includes the Fertilizer Optimization and Expected Average Effects per Ha tables.

<table>
<thead>
<tr>
<th>Fertilizer Optimization</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Crop</strong></td>
</tr>
<tr>
<td>Maize</td>
</tr>
<tr>
<td>Total Cost</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Expected Average Effects per Ha</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Crop</strong></td>
</tr>
<tr>
<td>Maize</td>
</tr>
</tbody>
</table>

Also similar to the optimized results section of the Fertilizer Optimizer, both of these spreadsheets have tables with a dark red heading background color.

For this particular example, the assumption will be made that 1 ha of maize will be planted with an expected grain value of $0.20 per kg and the cost of urea is $75.00 per 50 kg bag ($1.50 per kg). Also, the producer has a budget constraint for this example of $200.00. After entering all of these values the crop input parameters have all been entered appropriately for this particular example.

To determine the effects applying urea has on the maize a value in-between 0 to 100 may be entered in the Application Rates (cell I15) of the Fertilizer Application Rates. These values are the minimum and maximum values for the nitrogen fertilizer response function of maize.

<table>
<thead>
<tr>
<th>Fertilizer Application Rates</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Crop</strong></td>
</tr>
<tr>
<td>Maize</td>
</tr>
</tbody>
</table>

Entering application rates of 25, 50, 75, and 100 kg of urea per ha has corresponding yield increases of 978, 1,519, 1,805, and 1,959 kg per ha. Net returns to the fertilizer investments for these particular application rates include $160, 229, 248, and 242.

<table>
<thead>
<tr>
<th>Expected Average Effects per Ha</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Crop</strong></td>
</tr>
<tr>
<td>Maize</td>
</tr>
</tbody>
</table>

From this application analysis one may see that the maximum application rate does not necessarily guarantee the maximum net returns to the fertilizer investment. The optimal fertilizer application rate for this particular problem most likely exists between 75 to 100 kg of urea since this is the range where the net returns reach a maximum and then decline. The goal of this example is to identify the urea application rate that maximizes net returns on the fertilizer investment. By using the reiterative process of Solver the application rate may be identified which maximizes net returns subject to the problem’s constraints.
The following information must be entered into the Solver Parameters input window to run the optimization which determines the optimal urea application rate on maize.

Steps to enter this information into the Solver Parameters input window include:

1) Set the Objective Function: J25 in the Returns and Expenses Table.

Double clicking on cell J25 when the Solver Parameter window is not open shows the formula for the objective function. The objective function is the difference between the Net Value (anticipated yield from applying urea multiplied by the expected selling price) minus the Fert Cost (fertilizer cost from applying urea). Changing the fertilizer application rate causes both the Net Value and Fert Cost to change. Solver will reiteratively change the application rate in cell I25 until a maximum value has been identified.
2) Select To: Max.

Solver’s algorithm searches the range of applicable urea application rates that maximizes net returns on the investment. The range of applicable rates is subject to the set of constraints imposed upon them.

3) Set the By Changing Variable Cells: I15 in the Fertilizer Application Rates Table.

```plaintext
<table>
<thead>
<tr>
<th>Crop</th>
<th>Ha Planted</th>
<th>Application Rate</th>
<th>N App Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maize</td>
<td>1</td>
<td>0</td>
<td>=I15*C12</td>
</tr>
</tbody>
</table>
```

Cell I15 links to cell J15 or the N App Amount (nitrogen application amount) which ties to the Exp Change (expected change) in cell L20 for the maize nitrogen response function in the Crop Yield and Responses Table.

```
<table>
<thead>
<tr>
<th>Function</th>
<th>Constant</th>
<th>Coefficient 1</th>
<th>Coefficient 2</th>
<th>Base Production</th>
<th>Exp Change</th>
<th>Exp Yld/kg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maize N</td>
<td>3.9287</td>
<td>2.1387</td>
<td>0.9476</td>
<td>1.79</td>
<td>0.0000</td>
<td>0.0</td>
</tr>
</tbody>
</table>
```

The Exp Change in cell L20 then links to the Exp Yld/kg (expected yield per kg) in cell M20. After subtracting the Fert Cost (fertilizer costs) in cell I25 the Net Return in cell J25 is calculated for Solver to maximize.

```
<table>
<thead>
<tr>
<th>Function</th>
<th>Net Value</th>
<th>Fert Cost</th>
<th>Net Return</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maize N</td>
<td>0.00</td>
<td>0.00</td>
<td>0.0</td>
</tr>
</tbody>
</table>
```

The Exp Yld/kg also links to the Yield Increases output cell C29 in the Expected Average Effects per Ha Table in the optimized results section.

4) Subject to the Constraints: I15 <= I10, I15>H10, and I25 <=C16.
The I15 >= H10 constraint requires the Application Rate in the Fertilizer Application Rates Table be greater than or equal to the Min Rate in the Fertilizer Constraints Table. Setting the Min Rate in cell H10 equal to zero imposes a non-negativity constraint on the Application Rate. Also, the I15 <= I10 constraint requires the Application Rate be less than or equal to the Max Rate in the Fertilizer Constraints Table. Setting the Max Rate in cell I10 equal to 100 establishes the maximum which the Application Rate may assume.

Finally, the I25 <= C16 constraint requires the Fert Cost in the Returns and Expenses Table to be less than or equal to the Budget Constraint (amount available to invest in fertilizer). Solver will not select an application rate that has a cost exceeding the budget constraint imposed by the user.
After checking to see if the objective function is set to a maximum value by changing the variable cell subject to the set of constraints, the Solver Parameter input window has all of the necessary values and the optimization may be run by pressing Solve.

After pressing Solve on the Solver Parameters input window the Solver Results output window appears. Steps to check the validity of the solution include:

1) Check the Solver Results window to determine the validity of the optimization’s solution.

When Solver has found an optimal solution, the message “Solver has converged to the current solution. All constraints are satisfied.” appears. Additional information is provided below the prior statement indicating the number of iteration necessary for completing the optimization.

2) Press OK to keep the optimal solution found by Solver.

If Solver does not find the appropriate optimized solution, the Solver Results output window will display an error message. The error message states “Solver could not find a feasible solution.” due to the types of constraints being imposed upon the objective function or other cells in the problem.
Double checking the types of constraints imposed on the objective function along with values set in these constraints usually corrects the problems found by Solver. Also, when designing an optimization problem all constraints should not be imposed on the problem all at once. Gradually testing Solver with each additional constraint serves as a good way to verify the validity of the solution. Sometimes the severity or number of constraints imposed on Solver may go beyond the ability of the optimizer to find an optimal solution.

The Application Rate cell I15 of the Fertilizer Application Rates Table indicates the optimal kg per Ha of Urea which should be applied.

<table>
<thead>
<tr>
<th>Fertilizer Application Rates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crop</td>
</tr>
<tr>
<td>Maize</td>
</tr>
</tbody>
</table>

The amount of 78.9 kg of Urea per Ha of Maize translates to 36.3 kg of actual nitrogen applied per Ha.

The Exp Yld (expected yield) per kg with the application of the Urea is 1,835.8 kg as shown by cell M20 in the Crop and Yield and Responses Table. Multiplying the yield increase by the expected selling price equals the Net Value for the fertilizer application in cell H25 of the Returns and Expenses Table.
Multiplying the kg of Urea applied times the cost per kg of Urea equals the Fert Cost in I25 of the Returns and Expenses Table. The difference between the Net Value and Fert Cost equals the Net Return in cell J25.

The Application Rate, Exp Yld/kg, Fert Cost, and Net Return are linked with the output cells in the optimized results section.

As previously estimated, the optimal fertilizer application rate given the anticipated selling price and constraints on the problem is 79 kg of urea for the Ha of Maize planted. The total cost for the urea is $118 dollars. The anticipated Yield Increase with the addition of fertilizer equates to 1,836 kg of Maize per Ha. After accounting for the additional fertilizer expense the Net Returns on the limited fertilizer investment equates to $249.
2.3.2 Fertilizer Optimizer

Similar to the Example Optimization spreadsheet, the Fertilizer Optimizer spreadsheet contains three major sections including the crop input parameters, response function calculations, and the optimized results section. Opening the Fertilizer Optimizer spreadsheet should produce an image similar to the below.

![Fertilizer Optimizer Spreadsheet Image]

Pressing CTRL + SHIFT + U on the keyboard at the same time should display the response function calculations on the right-hand side of the spreadsheet.
Major sections of the crop input parameters include the Crop Selection and Prices, Land Area Requirements, Fertilizer Selection and Prices, and Budget Constraint Tables.

<table>
<thead>
<tr>
<th>Crop Selection and Prices</th>
<th>Land Area Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Crop</strong></td>
<td><strong>Area Planted (Ha)</strong></td>
</tr>
<tr>
<td>Maize</td>
<td>0.00</td>
</tr>
<tr>
<td>Sorghum</td>
<td>0.00</td>
</tr>
<tr>
<td>Upland rice, paddy</td>
<td>0.00</td>
</tr>
<tr>
<td>Beans</td>
<td>0.00</td>
</tr>
<tr>
<td>Finger millet</td>
<td>0.00</td>
</tr>
<tr>
<td>Soybeans</td>
<td>0.00</td>
</tr>
<tr>
<td>Groundnuts, unshelled</td>
<td>0.00</td>
</tr>
<tr>
<td><strong>Total hectares</strong></td>
<td>0</td>
</tr>
</tbody>
</table>

Input sections of the Fertilizer Optimizer have yellow shading in the headings of the tables. Editable input ranges for the tables are in white.

The Crop Selection and Prices Table in cell range B14:D23 in the spreadsheet contains the Crop, Area Planted (Ha), and Expected Grain Value/kg for the optimization. If the Fix land area requirement option is not selected in the Land Area Requirements Table, the Area Planted (Ha) may receive a different acreage allocation depending upon the returns associated with applying fertilizer to a particular crop.

<table>
<thead>
<tr>
<th>Fertilizer Selection and Prices</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Fertilizer Product</strong></td>
</tr>
<tr>
<td>Urea</td>
</tr>
<tr>
<td>Triple super phosphate, TSP</td>
</tr>
<tr>
<td>Diammonium phosphate, DAP</td>
</tr>
<tr>
<td>Muriate of potash, KCL</td>
</tr>
<tr>
<td>xxx</td>
</tr>
</tbody>
</table>

Constraints for the Area Planted (Ha) or land allocation are given in the Land Area Requirements Table.
The Land Area Requirements Table located in cell range F14:H23 of the spreadsheet includes the Min Planted (Ha), Max Planted (Ha), and Max Total (Ha). If the user wants to decide how many Ha of each crop to plant the Fix land area requirement checkbox may be selected to lock in the area of each crop.

<table>
<thead>
<tr>
<th>Land Area Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min Planted (Ha)</td>
</tr>
<tr>
<td>0.00</td>
</tr>
<tr>
<td>0.00</td>
</tr>
<tr>
<td>0.00</td>
</tr>
<tr>
<td>0.00</td>
</tr>
<tr>
<td>0.00</td>
</tr>
<tr>
<td>0.00</td>
</tr>
<tr>
<td>0.00</td>
</tr>
</tbody>
</table>

By not selecting the Fix land area requirement checkbox and entering a value for the Max Total (Ha) greater than zero in cell H16 of the spreadsheet allows Solver to allocate land to the crops being planted with the greatest response to fertilizer nutrient application.

The Fertilizer Selection and Prices Table located in cell range B25:F31 displays the name of the Fertilizer Product, the N, P2O5, and K2O nutrient contents of each, and Price/50 kg bag.

<table>
<thead>
<tr>
<th>Fertilizer Selection and Prices</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fertilizer Product</td>
</tr>
<tr>
<td>---------------------</td>
</tr>
<tr>
<td>Urea</td>
</tr>
<tr>
<td>Triple super phosphate, TSP</td>
</tr>
<tr>
<td>Diammonium phosphate, DAP</td>
</tr>
<tr>
<td>Murate of potash, KCL</td>
</tr>
</tbody>
</table>

Entering a value greater than zero for the Price/50 kg bag of a particular Fertilizer Product allows Solver to consider the nutrients in the optimization.

The final user crop input parameter is located in cell range B33:C34 for the Budget Constraint Table.

<table>
<thead>
<tr>
<th>Budget Constraint</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amount available to invest in fertilizer</td>
</tr>
</tbody>
</table>

The amount the user has available for investing in fertilizer is entered in the Budget Constraint Table.
By pressing CTRL + SHIFT + U on the keyboard, the right hand side of the Fertilizer Optimizer spreadsheet with the response function calculations section appears from being hidden. The response function calculations includes: the fertilizer nutrient properties section, Area Planted (Ha) Check, Fertilizer Constraints, Fertilizer Application Amount, Fertilizer Costs, Crop and Yield Responses, and Accumulated Sums Tables.

Values in the response function calculation section with red shading or red font are sections not currently active due to response functions not being significant for a particular nutrient and crop.

The first table on the right hand section in cell range N8:T13 of the response function calculation includes the nitrogen, phosphorous, and potassium percentages by product along with the price/kg of the fertilizer.
The constants of 0.437 for phosphorous and 0.83 for potassium are multiplied by the content levels of these nutrients for each fertilizer product to determine the amount available to the plant. These constants are predetermined values and remain fixed in the Fertilizer Optimizer spreadsheet.

The Area Planted (Ha) Check Table in cell range L15:L22 and Fix land area req check Table in cell range L24:L25 checks to see what the minimum and maximum planted Ha are for each crop.

<table>
<thead>
<tr>
<th>Fertilizer</th>
<th>Abbreviation</th>
<th>Nitrogen</th>
<th>Phosphorous</th>
<th>Potassium</th>
<th>Price/kg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Urea</td>
<td>Urea</td>
<td>46.0%</td>
<td>0.0%</td>
<td>0.0%</td>
<td>0</td>
</tr>
<tr>
<td>Triple Super Phosphate</td>
<td>TSP</td>
<td>0.0%</td>
<td>20.1%</td>
<td>0.0%</td>
<td>0</td>
</tr>
<tr>
<td>Diammonium Phosphate</td>
<td>DAP</td>
<td>18.0%</td>
<td>20.1%</td>
<td>0.0%</td>
<td>0</td>
</tr>
<tr>
<td>Muriate of Potash</td>
<td>KCl</td>
<td>0.0%</td>
<td>0.0%</td>
<td>49.8%</td>
<td>0</td>
</tr>
</tbody>
</table>

If statements in L16:L22 check the minimum and maximum planted Ha for each crop. If the minimum and maximum planted Ha equal the same values, then the If statement assumes that particular acreage allocation, otherwise a default value of 0.0001 is assigned. A value greater than zero must be entered for Solver to consider a crop when optimizing the acreage allocation. When the Fix land area req check assumes a value of true, the Fix land area req checkbox is selected causing the acreage allocation for all crops to be fixed to a specific value.

Constraints for the fertilizer application are given in the Fertilizer Constraints Table in cell range N16:AA24.

<table>
<thead>
<tr>
<th>Crop</th>
<th>Urea Min</th>
<th>TSP Min</th>
<th>DAP Min</th>
<th>KCl Min</th>
<th>xxx Min</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maize</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Sorghum</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Upland rice, paddy</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Beans</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Finger millet</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Soybeans</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Ground nuts, unshelled</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Values in cell range O18:S24 establish the minimum fertilizer application rate of zero which implies a non-negativity constraint on the optimization.
In cell range T17:X24 the maximum fertilizer application rates by product are established to create an upper limit.

<table>
<thead>
<tr>
<th></th>
<th>Urea Max</th>
<th>TSP Max</th>
<th>DAP Max</th>
<th>KCl Max</th>
<th>xxx Max</th>
<th>N Sum</th>
<th>P Sum</th>
<th>K Sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>46</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>46</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>46</td>
<td>25</td>
<td>15</td>
</tr>
<tr>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>46</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>15</td>
<td>30</td>
<td>50</td>
</tr>
<tr>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>15</td>
<td>35</td>
<td>50</td>
</tr>
</tbody>
</table>

Besides individual constraints on the amount of fertilizer that may be applied by product, the total sum for the element form of N, P, and K must also be constrained for each crop. In cell range Y17:AA24 the Fertilizer Total Maxes for the N Sum, P Sum, and K Sum are added together for each crop.

The Area Planted and Price Table in cell range L31:N40 along with the Fertilizer Application Amount Table in cell range O31:S42 defines the Ha Planted, Value/kg of the crop, and the application rate of each fertilizer product for each crop.

The Ha Planted in cell range M33:M38 of the Area Planted and Prices Table are directly linked with the Area Planted (Ha) of the Crop Selection Prices crop input parameters section. Also, the Value/kg of the Area Planted and Price Table links back to the Expected Grain Value/kg of the Crop Selection and Prices Table in the same crop input parameters section. The Fertilizer Application Amount in cell range O33:S39 links directly to the Fertilizer Optimization Table in the optimized results section. The Fertilizer Application Amount is the variable cells that Solver changes to find the optimal solution.
The Fertilizer Units Applied/Ha Table in cell range T33:AA40 along with the Fertilizer Totals Table in cell range AB33:AD40 determine the elemental forms of N, P, or K applied to each particular crop.

<table>
<thead>
<tr>
<th>Fertilizer Units Applied/Ha</th>
<th>Fertilizer Totals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Urea kg (N)</td>
<td>TSP kg (P)</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

To determine the elemental form of N, P, or K allocated by each fertilizer product in the Fertilizer Units Applied/Ha Table, the fertilizer concentration rates from the first table in the right hand section in cell range N8:T13 is multiplied by the Fertilizer Application Amount in cell range O33:S39. Row T40:AA40 in the Fertilizer Units Applied per Ha Table sums the amount of N, P, or K applied by each product. The Fertilizer Totals Table sums the amount of N, P, and K applied to each crop from the Fertilizer Units Applied/Ha Table.

Expenses for the various fertilizers are presented in the Fertilizer Cost Table of cell range O48:U55.

<table>
<thead>
<tr>
<th>Fertilizer Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crop</td>
</tr>
<tr>
<td>-------</td>
</tr>
<tr>
<td>Maize</td>
</tr>
<tr>
<td>Sorghum</td>
</tr>
<tr>
<td>Upland rice, paddi</td>
</tr>
<tr>
<td>Beans</td>
</tr>
<tr>
<td>Finger millet</td>
</tr>
<tr>
<td>Soybeans</td>
</tr>
<tr>
<td>Groundnuts, enshelled</td>
</tr>
<tr>
<td>Sum</td>
</tr>
</tbody>
</table>

The cost of each individual fertilizer product in the Fertilizer Cost Table is calculated by multiplying the Price/kg from the first table in cell range N8:T13 by the Fertilizer Application Amount in cell range O33:S39. The cost is summed by each fertilizer product in row O55:S55 of the Fertilizer Cost Table. The Fertilizer Cost/Crop in column T48:T55 is the sum of each individual fertilizer cost per crop from cell range O48:S54 multiplied by the Ha Planted of each crop from the Area Planted and Prices Table of cell range M33:M39. Cell T55 gives the sum of the Fertilizer Cost/Crop column in T48:T55 which is the total cost of all fertilizer applied in the optimization. The budget constraint in cell U55 linked from Budget Constraint Table of the crop input parameters section gives the total amount of funds available for the constrained investment. Solver uses the total cost of the fertilizer and the budget constraint in the optimization to determine the maximum amount of fertilizer which may be applied until the funds are exhausted.
The Crop and Yield Responses Table in cell range L61:R81 contains the fertilizer response functions for each crop along with the expected change in crop yield.

<table>
<thead>
<tr>
<th>Functions</th>
<th>Constant</th>
<th>Coefficient 1</th>
<th>Coefficient 2</th>
<th>Base Production</th>
<th>Expected Change</th>
<th>Expected Yield Kg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maize N</td>
<td>3.9297</td>
<td>2.1387</td>
<td>0.9476</td>
<td>1.79</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Maize P</td>
<td>3.98</td>
<td>0.377</td>
<td>0.809</td>
<td>3.603</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Maize K</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>#NUM!</td>
<td>0.00</td>
</tr>
<tr>
<td>Sorghum N</td>
<td>2.27</td>
<td>1.68</td>
<td>0.532</td>
<td>0.69</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Sorghum P</td>
<td>2.305</td>
<td>0.362</td>
<td>0.839</td>
<td>1.943</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Sorghum K</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>#NUM!</td>
<td>0.00</td>
</tr>
<tr>
<td>Upland Rice N</td>
<td>3.675</td>
<td>2.396</td>
<td>0.958</td>
<td>1.279</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Upland Rice P</td>
<td>3.79</td>
<td>0.566</td>
<td>0.947</td>
<td>3.234</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Upland Rice K</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>#NUM!</td>
<td>0.00</td>
</tr>
<tr>
<td>Beans N</td>
<td>1.79</td>
<td>0.989</td>
<td>0.892</td>
<td>0.001</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Beans P</td>
<td>1.81</td>
<td>0.286</td>
<td>0.926</td>
<td>1.524</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Beans K</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>#NUM!</td>
<td>0.00</td>
</tr>
<tr>
<td>Finger millet P</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>#NUM!</td>
<td>0.00</td>
</tr>
<tr>
<td>Finger millet K</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>#NUM!</td>
<td>0.00</td>
</tr>
<tr>
<td>Millet P</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>#NUM!</td>
<td>0.00</td>
</tr>
<tr>
<td>Millet K</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>#NUM!</td>
<td>0.00</td>
</tr>
<tr>
<td>Soybeans K</td>
<td>1.972</td>
<td>0.285</td>
<td>0.974</td>
<td>1.687</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Groundnuts, shelled P</td>
<td>1.792</td>
<td>0.937</td>
<td>0.893</td>
<td>0.856</td>
<td>0</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Expected change in crop yield from the application each fertilizer nutrient is calculated by using the fertilizer response function for each crop. The Expected Change in cell range Q63:Q81 of the Crop and Yield Responses Table pulls the total elemental form of N, P, or K applied to each particular crop from the Fertilizer Totals Table in cell range AB31:AD40. Multiplying the Expected Change by 1,000 provides the Expected Yield Kg in cell range R63:R81.

The Fertilizer Application Rates Table in cell range W53:X55 has the constraints on the amount of fertilizer that must be applied before the P or K response functions provide a significant change in yield from their application.

<table>
<thead>
<tr>
<th>N Amt Req P</th>
<th>P Amt Req K</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>10</td>
</tr>
</tbody>
</table>

The N Amt Req P (the N amount required for P) in cell W55 of the Fertilizer Application Rates Table is the amount of N that must be applied to a crop before the expected yield increase from the application of P provides a significant yield response. Also, the P Amt Req K (the P amount required for K) in cell X55 is the amount of P that must be applied to a crop before the expected yield increase from the application of K may be counted as significant. If statements in the Fertilizer Requirements Table checks values in the Fertilizer Application Rates Table to determine the minimum fertilizer application rates necessary for the P or K yield response functions on whether the yield response from their application is significant.
Financial calculations from the Fertilizer Response functions are calculated in the Returns and Expenses Table in cell range S61:V81. Checks for the nutrient requirements are provided in the Fertilizer Requirements Table in cell range W61:X81. The Output Variable – kg/Ha Table in cell range Y61:Z81 summarizes the optimization results for each crop.

The Returns and Expenses Table includes the Individual Net Value from each individual crop response function in column S63:S81, Net Value per Crop summed from each Individual Net Value in column T63:T81, Fertilizer Costs in column U63:U81 linked with the Fertilizer Costs/Crop in the Fertilizer Cost Table, and the Net Return being the difference between the Net Value per Crop and the Fertilizer Costs in column V63:V81.

If statements in the Fertilizer Requirements Table gives the N Amt Req P in column W63:W81 and P Amt Req K in column X63:X81. If statements in these two columns either assume a value of zero or one depending upon whether the minimum nutrient amount for a particular crop in the Fertilizer Application Rates Table is greater than the nutrient’s sum in Fertilizer Totals Table. The values of either zero or one in the Fertilizer Requirements Table are multiplied by Expected Change Column in the Crop and Yield Responses Table. If a value of zero has been assumed the minimum application rate has not been met and the Expected Change assumes a value of zero. However, if the minimum application rate has been met then the Expected Change may assume a value greater than the minimum application rate.

Finally, the Output Variables – kg/Ha Table summarizes the Yield Increases in column Y63:Y81 and the Net Returns in column Z63:Z81. The Yield Increases is the sum of individual response functions from the Expected Yield Kg column by crop in the Crop and Yield Responses Table. The Net Returns column provides the anticipated net returns from the Returns and Expenses Table and adjusts this value to a per Ha basis. Both the Yield Increases and Net Returns information link back to the Expected Average Effects per Ha Table in the optimized results section.
Data from the Returns and Expenses Table is summed together in the Accumulated Sums Table in cell range T87:V89.

<table>
<thead>
<tr>
<th>Accumulated Sums</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Net Value</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fertilizer Cost</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Net Returns</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

The Net Value in cell T89 sums together the Net Value per Crop from the Returns and Expenses Table. Fertilizer Cost in cell U89 adds all of the individual Fertilizer Costs together from the same Table. Finally, the Net Returns in cell V89 sums all of the Net Returns together by crop from the Net Return column in the Returns and Expenses Table. Solver maximizes the total Net Returns in this Table when maximizing the optimization problem.

Major sections of the optimized results section include the Fertilizer Optimization, Expected Average Effects per Ha, and Total Expected Net Returns to Fertilizer Tables.
The first section in the optimized results section is the Fertilizer Optimization Table in cell range B42:H53.

<table>
<thead>
<tr>
<th>Crop</th>
<th>Application Rate - kg/Ha</th>
<th>Planted (Ha)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Urea</td>
<td>TSP</td>
</tr>
<tr>
<td>Maize</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Sorghum</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Upland rice, paddy</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Beans</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Finger Millet</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Soybeans</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Groundnuts, unshelled</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Total fertilizer needed</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Fertilizer and total costs</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

In this table the Application Rate – kg/Ha presents the amount of Urea, TSP, DAP, KCL, or xxx that must be applied on a per Ha basis and the Planted (Ha) per crop. Row 52 and 53 summarize the total fertilizer needed along with the cost for each product and the total fertilizer expense.

Results of the optimization anticipated effect on crop yield and revenue are presented in the Expected Average Effects per Ha Table in cell range B55:D63.

<table>
<thead>
<tr>
<th>Crop</th>
<th>Yield Increases</th>
<th>Net Returns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maize</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Sorghum</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Upland rice, paddy</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Beans</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Finger millet</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Soybeans</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Groundnuts, unshelled</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Summary information in the Expected Average Effects per Ha Table summarizes the expected Yield Increases in column C from cell range C57:C63. The Net Returns in column D from cell range D57:D63 shows the expected net returns per crop.

The final table in the optimized results section is the Total Expected Net Returns to Fertilizer in cell range B65:D67.

| Total net returns to investment in fertilizer | 0 |

The total net returns to investment in fertilizer in cell C66 sums together all expected returns to fertilizer use after subtracting the cost of the fertilizer.
To evaluate the optimization process of the Fertilizer Optimizer spreadsheet the following values should be inputted into the crop input parameters section.

Steps to enter this information into the crop input parameters section includes:

1) Leave the Area Planted (Ha) set to 0 for Maize and Sorghum. Only 2 Ha of land will be available for this optimization and up to 2 acres may be planted in either crop.

2) Solver will allocate the land according to which crop provides the highest net returns. The Min Planted (Ha) and Max Planted (Ha) need to be set to 0 Ha and 2 Ha for Maize and Sorghum. The Max Total (Ha) should be set to 2 Ha because this is the amount of land available in the optimization.

3) Set the Price/50 kg bag of Urea to $75 and Triple super phosphate, TSP to $125.

4) The Amount available to invest in fertilizer optimization will be $100. Having a budget constraint of $100 constraints the amount of fertilizer that may be applied by Solver.
To review the constraints present in the optimization, ensure the response functions are open on the right hand side of the spreadsheet by pressing CTRL + SHIFT + U on the keyboard at the same. Unhiding the response functions unprotects the spreadsheet and allows for the Solver Parameters dialog box to be opened by selecting the Data tab on the Excel spreadsheet and pressing the Solver button on the far right hand side of the menu.

Reviewing the Solver Parameters input window shows the following information:

1) Set the Objective function to maximize $V$89 (Net Returns) by changing cell ranges $O$33:$S$39, $C$16:$C$22 (Fertilizer Application Amounts and Area Planted).

2) Constrain cell ranges $AB$33:$AD$39 (Fertilizer Total) <= $Y$18:$A$24 (Fertilizer Total Maxes) to limit the total amount of fertilizer put on each crop by nutrient type.

3) Constrain cell ranges $C$16:$C$22 (Area Planted) >= $F$16:$F$22 (Min Ha) imposes a non-negativity requirement on the land constraint.
4) Constrain cell ranges $O$33:$S$39 (Fertilizer Application Amount) <= $T$18:$X$24 (Fertilizer Max Application Rate) to limit the amount of fertilizer that may be put on by each product.

5) Constrain cell ranges $C$23 (Total Hectares) <= $H$16 (Max Total) so the total land area amongst all of the crops cannot exceed the amount available in the optimization.

6) Constrain cell ranges $O$33:$S$39 (Fertilizer Application Amount) >= $O$18:$S$24 (Fertilizer Min Application Rate) imposes a non-negativity requirement on the fertilizer application amount.

7) Constrain cell ranges $C$16:$C$22 (Area Planted) <= $G$16:$G$22 (Max Ha) requires the land area planted for each individual crop to be less than or equal to the maximum for each one.

8) Constrain cell ranges $T$55 (Fertilizer Cost/Crop) <= $U$55 (Budget Constraint) keeps the cost of fertilizer purchased for the optimization to be less than or equal to the producer’s budget constraint.

Close the Solver Parameters input window after reviewing this information. The optimization may now be processed by pressing the Optimization button.

Upon successfully completing the optimization with a budget constraint of $100 the optimized results section should display the following information.

<table>
<thead>
<tr>
<th>Crop</th>
<th>Urea</th>
<th>TSP</th>
<th>DAP</th>
<th>KCL</th>
<th>xxx</th>
<th>Planted (Ha)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maize</td>
<td>33</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2.00</td>
</tr>
<tr>
<td>Sorghum</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Upland rice, paddy</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Beans</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Finger Millet</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Soybeans</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Groundnuts, unhshelled</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Total fertilizer needed</td>
<td>67</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Fertilizer and total costs</td>
<td>190</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Crop</th>
<th>Yield Increases</th>
<th>Net Returns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maize</td>
<td>1,202</td>
<td>190</td>
</tr>
<tr>
<td>Sorghum</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Upland rice, paddy</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Beans</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Finger millet</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Soybeans</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Groundnuts, unhshelled</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Total Expected Net Returns to Fertilizer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total net returns to investment in fertilizer</td>
</tr>
</tbody>
</table>

Results indicate that 2 Ha of Maize should be planted with 33 kg of Urea applied per Ha with an expected yield increase of 1,202 kg per Ha. Expected net returns to the limited investment
equate to $381. Next, change the budget constraint to $200 and press the Optimization button again.

Results from successfully completing the second optimization with a budget constraint of $200 should appear in the optimized results section with the following information.

<table>
<thead>
<tr>
<th>Crop</th>
<th>Application Rate - kg/ha</th>
<th>Planted (Ha)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maize</td>
<td>67</td>
<td>2.00</td>
</tr>
<tr>
<td>Sorghum</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Upland rice, paddy</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Beans</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Finger Millet</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Soybeans</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>Groundnuts, unshelled</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td><strong>Total fertilizer needed</strong></td>
<td>133</td>
<td>2</td>
</tr>
<tr>
<td><strong>Fertilizer and total costs</strong></td>
<td>200</td>
<td>200</td>
</tr>
</tbody>
</table>

Different than the first optimization, the amount of Urea applied per Ha basically doubles from 33 to 67 kg per Ha of Maize. The expected yield increase grows to 1,728 kg per Ha with expected net returns to fertilizer use of $491.

Finally, change the budget constraint to $300 and press the Optimization button again.
Results from the third optimization with a budget constraint of $300 should appear with the following information in the optimized results section.

Analysis of the third optimization shows the Urea and TSP application rates were increased to 79 and 6 kg per Ha of Maize. The expected Yield Increase for this particular example increases to 1,918 kg per Ha of Maize. The Total Expected Net Returns to this investment in fertilizer is $502. The total cost for fertilizer in this example is $266 (well below the budget constraint of $300).

When comparing the results of the three optimizations with budget constraints of $100, $200, and $300 the expected net returns and increase in crop yields do not increase proportionally. The non-linear response relationship provides the greater value for the first units of fertilizer applied and eventually a diminishing effect occurs. Solver has the ability to identify these fertilizer response relations across multiple crops when the budget constraint is limited.

After reviewing the example optimizations and gaining a basic understanding on how Solver functions, the following chapter explains how Macro programs function and automate many processes of the Fertilizer Optimizer. Chapter 3 explains the basics of Macro Programs along with how to access and edit them in the Fertilizer Optimizer.
CHAPTER 3: Macro Program Basics

Macro programs embedded within the Ugandan Fertilizer Optimizer automate common features and procedures for users. The Develop tab within Excel must be enabled to open the Microsoft for Visual Basic for Application (VBA) Editor to allow for accessing, editing, and creating Macro program. Chapter 3 explains how to accomplish these tasks along with an overview on Macros Programs.

3.1 Accessing Macro Code

The Developer tab in Excel must be enabled to access or edit macros within the Fertilizer Optimization spreadsheet. When enabled the Developer tab appears on the Quick Access Toolbar.

![Developer Tab](image)

3.1.1 Adding Developer Tab to Excel

The following series of steps shows how to add the Developer tab on the Quick Access Toolbar in Excel:

![Step 1](image)

![Step 2](image)
1) Select the File tab on the Quick Access Toolbar
2) Select Options on File drop down menu

After selecting Options the Excel Options window appears.

3) Select Customize Ribbon on the left hand side of the Excel Options window
4) In the Main Tabs box, checkmark the Developer option
5) Select OK

The Developer Tab now appears on the Quick Access Toolbar.

3.1.2 Unlocking Fertilizer Optimizer Macros

Macros enabled in the Fertilizer Optimizer are password protected. The following series of steps explain how to access the Visual Basic for Application Editor and unlock the password protected Macros in the Fertilizer Optimizer spreadsheet.
1) Select the Developer tab on the Quick Access Toolbar
2) Click the Visual Basic Button (typing Alt + F11 on the keyboard at the same time can bypass Step 1 and 2)

After selecting the Visual Basic Button (or the Alt + F11 keyboard shortcut) the Microsoft Visual Basic for Application editor appears.

The Visual Basic editor has the ability to create and modify Macros Programs. The code within the editor for a macro usually begins with the keyword Sub and ends with the keyword End Sub. Each macro that does a specific task in the Fertilizer Optimizer is called a subroutine. In the editor, a module refers to a group of macros and several modules compiled together form a project.
Several different projects may exist together in the same Excel workbook at the same time. Projects typically encompass a set of modules that have a specific purpose. Password protecting a project allows a user to guard the integrity of compiled macros.

The Project Explorer window of the Visual Basic editor displays a list of all projects embedded within the Excel spreadsheet.

![Project Explorer window](image)

Reviewing the list in the Project Explorer window shows all of the different projects available in the Excel spreadsheet. This list may vary depending upon the computer which the spreadsheet is viewed due to the different types of add-ins enabled. In the list above, the FertilizerOptimizer project contains all of the modules which run the Fertilizer Optimizer spreadsheet.

Double clicking the FertilizerOptimizer prompts the Password window to appear.

![Password window](image)

3) Enter in the password (case sensitive): **Sdm4760**
4) Click OK

Properly entering the password causes the FertilizerOptimizer project to unlock and expand in the Project Explorer window.
5) Click the + icon sign to expand the drop list of Modules

The drop down Modules list displays the seven different modules used in the Fertilizer Optimizer spreadsheet including the Fix_Land_Area_Req_Checkbox, Help, Hide_Protect, Optimizer_Solver, Print_Output, Reset_Form, and Unprotect_Unhide.

After completing these steps, the Macros for the Fertilizer Optimizer have now been unlocked and may be edited.
3.2 Structure of Macro Programs

The Microsoft Visual Basic for Application Editor serves as the area in the Uganda Fertilizer Optimizer to access, create, or edit the Macros Programs. After completing section 3.1, the Microsoft Visual Basic for Application Editor should be displayed and the Fertilizer Optimizer Project unlocked similar to the image below.

![Microsoft Visual Basic for Application Editor](image)

3.2.1 Visual Basic for Application Editor

The following section explains the basics of the Microsoft Visual Basic for Application Editor and how to navigate in this window.

The Project Explorer of the Editor appears by default along the right hand side of the window. This area organizes all projects in the current spreadsheet. As noted in section 3.1, not all projects are available for editing due to certain projects being password protected. The number and names of the different projects displayed in this Project Explorer will vary depending upon the computer in which the Editor is opened.
A closer review of the Project Editor from above shows four different projects exist in Project Explorer including: atpvaen.xls, FertilizerOptimizer, Simetar, and Solver. Trying to access any of the projects other than the FertilizerOptimizer will prompt the user for a password. Simetar and Solver Projects are add-ins for Excel specific to the user’s computer. By default the Tool requires that the Solver add-in be installed to function appropriately. This project should appear in the Editor.

Clicking on the + icon next to the Modules and References folders under the Fertilizer Optimizer Project expands the contents to display the different macros and reference present in the Uganda Fertilizer Optimizer.

As the expanded contents shows, the Uganda Fertilizer Optimizer has eight macros including the Fix_Land_Area_Req_Checkbox, Help, Hide_Protect, Optimizer_Solver, Print_Output, Reset_Form, and Unprotect_Unhide and one reference being the Reference to Solver.XLAM.

Chapter 4 presents a detailed explanation on how the eight macros function. The Reference to Solver.XLAM serves as a designation noting whether any of the modules in a project reference another project in the spreadsheet. In this particular spreadsheet the Optimizer_Solver Macro calls upon Solver and causes the Reference to Solver.XLAM to appear in the References folder.
The Programming window serves as the area in which the code for a macro may be edited. Several different methods exist to access the programming window of a module. Right clicking on a module, displays an expandable drop down list.

Selecting View Code will display the Programming Window for a particular module. Also, rapidly left clicking on a module twice can also display the Programming window.
The Programming window inside of the Visual Basic for Application Editor may be moved, resized, minimized, expanded, or closed as any other window can in a Microsoft operating environment.

Multiple Programming windows may be open at the same time inside of the Visual Basic for Application Editor. A user may want to have multiple windows open to compare code between different macros. Ease of navigation around the Editor increase with use. The following section continues to elaborate on the use of the Editor and outlines the basic layout of macro code.

3.2.2 Layout of Code

A module contains a macro that has a series of statements designed to accomplish a particular task in Excel. These tasks may be basic to complex depending upon the request performed by the macro. The amount of code embedded within a module usually reflects the number and intricacy of the tasks performed in the program. This should not be the sole indicators of a program’s ability as a relatively short macro may still perform a considerable number of tasks.

To explain the basic layout of code for a macro, the Programming window with the Help Macro serves as an example.
Major sections to the layout of the Fertilizer_Optimization_2_17_2013 – Help (Code) Programming Window include:

**Fertilizer_Optimization_2_17_2013 – Help (Code)** – Programming Window heading located in the top blue pane of the window. The heading of the programming window indicates the name of the Excel workbook in which the macro is embedded followed by the name of the macro. For the above example the name of the spreadsheet is Fertilizer_Optimization_2_17_2012 followed by a – (dash) and the name of the Macro called Help. The – (dash) of the title heading are automatically included in the Programming Window heading to separate the name of the spreadsheet and Macro. The (Code) notation reminds the programmer that the word preceding the brackets represents the name of the Macro.

**Sub Help()** – This statement in a light purple font represents the introductory code for a program. The word **Sub** indicates the lines of code are a subroutine meaning the set of commands are designed in a manner to achieve a specific purpose. The word **Help** followed by **Sub** indicates that this series of codes are for the Help Macro. To signify the end of a macro the words **End Sub** must be included as the last statement in the lines of code.

**Lines starting with a single quote (’)** – Any line that starts with a single ’ (apostrophe) and a space represents a comment in a program. Excel ignores any text wrote in these statements and the comment font has a light greenish tint to indicate that text only serves as notes in the Macro. Comments serve as good reference for programmers to indicate how a series of commands or statements process when run.

‘ **Help Macro** – This comment line indicates the name of the program as the Help Macro. Typically the first three lines after **Sub** statement of a macro are comments with the middle line indicating the name of the program.
‘Selects cell A1 in the Help and Instructions Tab’ – This comment line indicates the following series of code in the spreadsheet will select cell A1 in the Help and Instructions Tab of the Excel workbook.

Sheets (‘Help and Instructions’).Select – This code makes the Help and Instruction tab the active page in the Excel workbook.

Range (“A1”). Select – This line tells Excel to select and make cell A1 the active tab in the Excel workbook. Once selected as the active cell in the workbook the user can read the series of instructions provided on the screen to aid troubleshooting common problems in the Fertilizer Optimizer spreadsheet.

End Sub – Ending statement for the Sub Help subroutine. End Sub does not include the name of the Help Macro in the statement.

Macros embedded in the Fertilizer Optimizer spreadsheet vary in length reflecting the complexity of the activities carried out by the code. All macros embedded within the Fertilizer Optimizer spreadsheet follow a similar format to the Help macro. Many of the commands are similar between the programs due to the common activities shared by the macros. The following section goes into further detail regarding command statements shared between macros of the Fertilizer Optimizer.

3.2.3 Command Statements

The following section explains basic programming standards, statements, and commands utilized in the macros of the Fertilizer Optimizer. Although used to achieve different purposes, the language of the commands remain relatively similar between the macros and by understanding the basics of the code allows for further editing and troubleshooting problems within the statements.

Macro Heading

The heading of each macro program follows a similar format. Maintaining a consistent heading allows the programmer to quickly identify the name of the macro program that may need editing or error checking.

Sub Unprotect_Unhide()
' Unprotect_Unhide Macro
'

Each Macro in the Fertilizer Optimizer follows a similar heading layout as the Sub Unprotect_Unhide () example displays. To start the code the word Sub must be typed along with the Macro name Unprotect_Unhide and a pair of (). The Macro name must not contain any spaces so _ (underscores) can be used to separate words. Typically after the Sub heading, three comments follow with the middle one containing the entire name of the Macro.
**Sub Statements**

To mark the beginning and ending of a macro, program statements must be provided to indicate where the code starts and stops.

```vba
Sub Reset_Form()
    '...
End Sub
```

Each Macro must contain the `Sub Reset_Form()` and `End Sub` statements. The `Sub Reset_Form()` represents the start of the subroutine coupled with the name of the Macro. A series of related commands that have the goal of completing a particular task are referred to as a subroutine. At the completion of these tasks, the statement `End Sub` marks the completion of the subroutine and stops the macro.

**Range Statement**

User editing cells in Excel commonly select a range to apply a certain kind of formatting or perform a calculation on them.

```vba
' Select a cell
    Range("A1").Select
```

One of the most basic statements for VBA with application to Excel programming involves selecting a cell in a spreadsheet. The `Range("A1").Select` code tells which cell(s) in Excel to select and become the active range. For the purpose of these statements, a range may refer to one or more cells in a particular order. Also, when selecting cell(s) with this statement quotation marks and brackets must enclose ("A1") the range.

```vba
' Reset Land Area Requirement to 0
    Range("F16:G22, H16") = 0
```

Another basic command statement involves setting cell(s) or a series of cells in Excel equal to a specific value. The `Range("F16:G22, H16")` for this examples takes two different ranges of cells including F16 to G22 and H16 and set them = 0. Different than the prior example, the selection of different ranges can be accomplished by using a comma to separate the series. Also, to select a series of cell such as F16 to G22 the ranges must be divided by using a : (colon). All cells being set equal to a specific value must be enclosed once again by quotation marks and brackets.

```vba
' Set maize min and max areas planted
    Range("F16:G16") = Range("C16")
```

Rather than setting a series of cells equal to a specific constant, a range may be specified to a value defined by a user. The `Range("F16:G16")` takes the value of `Range("C16")` inputted by the user in the Excel workbook. The order in which a series of cells are assigned a specific value cannot be reversed. In the example, the order cannot be stated as
Range("C16") = Range("F16:G16") which would result in an error. To assign a series of cells a constant or imputed value from the user, the amount must be stated after the range which will be assigned the specific value.

**If Statements**

Many users of Excel have exposure to using if statements as function in cells. The basic premise to using this function involves a logic test to determine whether or not an argument is true. Depending upon the solution to this logic test, a value or action can be performed if the argument is true or a different value or action may be carried out if the logic test is false.

```vba
' If Statement for Maize
    If Range("C16") <= 0.01 Then
        Range("C16") = 0
        Range("O33:S33") = 0
    End If
```

Comparable to if statements of Excel, the if statements of VBA operate in a similar manner. In the above example, the logic test If Range("C16") <= 0.01 determines whether the contents of C16 is less than or equal to 0.01. If the argument is true the commands of Range("C16") = 0 and Range("O33:S33") = 0 will be carried out and the code ends with the statement End If. If the logic test is false the two Range commands will not be carried out and the if statement will cease with the End If operation.

The number of commands embedded within an if statement may be numerous, but the statements must be listed between the If Range and End IF code. Multiple if statements can be embedded within other if statement to check different levels of logic arguments with the code. Each if statement within another if argument must have the End If code to cease the operation of each statement. To simplify using multiple if arguments, the programmer can also introduce a series of if statements that are not embedded within each other to test for a specific argument.

**Hide/Unhide Columns in a spreadsheet**

The majority of calculations performed in the Fertilizer Optimizer spreadsheet occur in the hidden columns of K to AJ. These columns are hidden to protect the integrity of calculations performed in the Fertilizer Optimizer spreadsheet. To edit or test these calculations, columns K:AJ must be unhidden in the spreadsheet.

```vba
' Hide columns L thru AI
    Columns("K:AJ").Hidden = True
```

To hide columns K to AJ in the Excel spreadsheet the command `Columns("K:AJ").Hidden = True` hides the selected columns due to `.Hidden = True`.

```vba
' Unhide columns L thru AI
    Columns("K:AJ").Hidden = False
```
To unhide columns K to AJ in the Excel spreadsheet the statement `Columns("K:AJ").Hidden = False` unhides the selected columns due to `.Hidden = False`.

In each of these examples the notation `Columns("K:AJ")` stays the same with the columns being hidden enclosed by the ("K:AJ") quotation marks and brackets. Also, `.Hidden = remains the same, but one of the two options of either True or False must be used. True will hide the columns K to AJ while False unhides the columns.

**Protect/Unprotect the Spreadsheet**

By clicking on the Review Tab in Excel and selecting the Protect Sheet Icon, the spreadsheet becomes protected from editing. A password can also be entered to password protect the workbook. Only those cells unlocked (allowed for editing) can be changed when the spreadsheet is protected. Hidden columns cannot be unhidden when the spreadsheet is protected. By protecting the spreadsheet when columns K to AJ are hidden the user of the workbook cannot accesses formulas and functions needed for the Fertilizer Optimizer. This protects the integrity of the spreadsheet and reduces the likelihood of user induced errors.

```
' Password protect the spreadsheet
ActiveSheet.Protect Password:="52Fre"
```

To password protect the Fertilizer Optimizer tab of the Fertilizer Optimizer spreadsheet the code `ActiveSheet.Protect Password:="52Fre"` protects the active sheet of the workbook.

```
' Unprotect sheet incase the document was protected
ActiveSheet.Unprotect Password:="52Fre"
```

To unprotect the password protected active sheet where the optimization is carried out the command statement `ActiveSheet.Unprotect Password:="52Fre"` unprotects the Fertilizer Optimizer tab of the Fertilizer Optimizer spreadsheet.

In each example the `ActiveSheet. and Password:="52Fre"` remain the same for the command statements. The major difference between the two command statements exists between the words Protect and Unprotect. The word Protect will password protect the spreadsheet when combined with the code of the statement. Including the Un in front of protect will Unprotect the password protected spreadsheet when included with other code of the command. The password required to unprotect the spreadsheet is given by the portion of the command statement `Password:="52Fre"`. As previously discussed, the 52Fre phrase serves as the password for the spreadsheet.

**Specialized Statements for Solver**

The Solver add-in of Microsoft Excel may be run using the VBA code of Excel. Automating this add-in with the use of Macro reduces the chances of user induced errors from improperly inputting target cells, objective function cells, or constraints.

```
' Clears all previous objective functions or constraints from solver dialog box
```
SolverReset

To begin the process of inputting various cell references or constraints into solver all prior conditions should be erased. In order to erase all of these references the code SolverReset removes all prior conditions which rests the Solver add-in. Before using VBA code to manipulate Solver erasing all prior conditions reduces the chance of having inappropriate references or constraints occurring in the optimization.

' Maximize $V$89 (Net Returns) by changing $O$33:$S$39,$C$16:$C$22 (Fertilizer Application Amounts and Area Planted)
   SolverOk SetCell:="$V$89", MaxMinVal:=1, ValueOf:="0",
   ByChange:="$O$33:$S$39, $C$16:$C$22"

The above statement inputs the Set Object, To Max/Min/Value Of, and By Changing Variable Cells of Solver. For the Fertilizer Optimizer, the Set Objective is done by the command SolverOk SetCell:="$V$89", where cell $V$89 contains the objective function for the optimization. The statement MaxMinVal:=1, ValueOf:="0" tells solver to optimize cell $V$89 to a maximum value. The MaxMinVal:= can assume three different values where 1 = Maximize, 2 = Minimize, or 3 = Match a Specific Value (Value of ). When option 3 is selected the ValueOf:="0", must be set to a specific value in which the optimization is attempting to assume. Whenever option 1 or 2 are selected the ValueOf:="0" can remain defaulted to zero as this value will be ignored.

For the objective function to achieve the optimization’s goal a certain range of cells must be changed to achieve a particular value. The By Changing Variable Cells in Solver are inputted using the statement ByChange:="$O$33:$S$39, $C$16:$C$22". The two ranges of $O$33:$S$39 and $C$16:$C$22 represents the fertilizer application rate and land area planted that may be changed during the optimization. When more than one range exists in Solver, the cell references are separated using a , (comma).

' Constraint $O$33:$S$39(Fertilizer Application Amount) >= $O$18:$S$24 (Fertilizer Min Application Rate)
   SolverAdd CellRef:="$O$33:$S$39", Relation:=3,
   FormulaText:="$O$18:$S$24"

Multiple constraints exist in the Fertilizer Optimization in order to find a solution bound within a feasible range. The command SolverAdd CellRef:="$O$33:$S$39", Relation:=3,
   FormulaText:="$O$18:$S$24" adds the constraint in Solver to have the fertilizer application rate $O$33:$S$39 must be >= the minimum fertilizer application rate of "$O$18:$S$24". The Relation:= can assume three different conditional values of 1 = less than or equal to (<=), 2 = value equal to (=), or 3 = greater than or equal to (>=). Multiple ranges cannot be combined on either side of an optimization constraint.

' Cancels the solver solution window from appearing
   SolverSolve userFinish:=True
After all objective or variable cells and constraints are inputted to Solver, the optimization may proceed by pressing the Solve button. The command statement \texttt{SolverSolve} selecting the Solve button of Solver and allows the optimization to complete. After the successful completion of an optimization, Solver displays a Solver Results window. In the Fertilizer Optimizer the Solver Results window does not appear to limit unneeded feedback to the user.

**Print Preview Output**

The Print Preview option in Excel allows a user to view output of the spreadsheet before the selection is printed. By clicking File and Print in Excel 2010 or the Office 2007 round icon and Print Preview in Excel 2007 (CTRL + F2 keyboard shortcut for either version of Excel) displays the contents of the workbook before printing.

```vba
' Select A1 to I64 for print preview
Range("A1:I68").Select
Selection.PrintOut Copies:=1, Preview:=True
```

Range A1 to I68 displays the Fertilizer Optimizer, Expected Average effects per Ha, and Total Net Returns to Fertilizer tables from the Fertilizer Optimizer spreadsheet. Printing this range displays optimized results from the Tool. To select this range the statement

```vba
Range("A1:I68").Select
```

makes A1 to I68 the active range. Next, the statement

```vba
Selection.PrintOut Copies:=1, Preview:=True
```

takes the active selection and displays the range in Print Preview. The command

```vba
Selection.PrintOut Copies:=1, Preview:=True
```

indicates to Excel only to print one copy of the active selection while \texttt{Preview:=True} allows the Print Preview option. For the \texttt{Preview:=} indicating True allows Excel to display the active select in Print Preview while \texttt{False} skips this option and sends the active selection directly to the printer for printing. Also, the \texttt{Copies:=} indicates the number of copies which should be printed after the user is finished reviewing the Print Preview option. Increasing the number from 1 to 2 will increase to two.

After reviewing these commands the programmer can now interpret various statements present throughout the macros code of the Fertilizer Optimizer. The actual code for each macro of the Fertilizer Optimizer is presented in Chapter 4 along with a discussion to elaborate on how the specific program operates.
CHAPTER 4: Fertilizer Optimizer Macro

Macro programs embedded in the Uganda Fertilizer Optimizer run Solver to determine the optimal fertilizer rate and land allocation specific to the user’s farm and constraints. Many of the equations and functions required to run the optimization are hidden and password protected. Chapter 4 explains the purpose of each program and how the macros function.

4.1 Fix_Land_Area_Req_Checkbox Macro

Users of the Fertilizer Optimizer can either provide a fixed land allocation for each crop where Solver does not vary the amount of hectares planted or allow the Tool to determine the optimal acreage allocation in addition to fertilizer application rates. In either case constraints are established to limit the range of possible solutions. Versions of the Tool where the Land Area Requirements input area to the right of the Crop Selection and Prices Table do not exist the following Macro will not be present in the spreadsheet.

Sub Fix_Land_Area_Req_Checkbox()
    ' Fix_Land_Area_Checkbox
    ' Set maize min and max areas planted
    Range("F16:G16") = Range("C16")
    ' Set sorghum min and max areas planted
    Range("F17:G17") = Range("C17")
    ' Set upland rice min and max areas planted
    Range("F18:G18") = Range("C18")
    ' Set beans min and max areas planted
    Range("F19:G19") = Range("C19")
    ' Set finger millet min and max area planted
    Range("F20:G20") = Range("C20")
    ' Set soybeans min and max area planted
    Range("F21:G21") = Range("C21")
    ' Set groundnuts min and max area planted
    Range("F22:G22") = Range("C22")
    ' Set Max Total (Ha) = Total hectares
    Range("H16") = Range("C23")
End Sub
The Fix_Land_Area_Req_Checkbox macro runs with the selection of the checkbox to the right of the Fix land area req. in cell F23. Running this macro sets the Min Planted (Ha) and Max Planted (Ha) equal to the Area Planted (Ha) inputted by the user. Also, the Max Total (Ha) becomes set equal to the Total hectares of the Area Planted (Ha) inputted by the user. The constraints of setting the min and max equal to the area planted specifies the exact hectares of each crop planted.

When the Fix land are req. checkbox of the Land Area requirements form is selected cell L25 has a value of TRUE assigned. If the Fix land area req. checkbox is not selected cell L25 assumes a default value of FALSE. The Optimizer_Solver Macro references the value of L25 to check the cell’s content with an if statement on whether to call or run the Fix_Land_Area Req_Checkbox Macro before carrying out the optimization. Solver may provide a substantially different answer if the user leaves the Fix land area req. checkbox unchecked.
4.2 Help Macro

Users of the Fertilizer Optimizer may experience problems when operating the Tool. The help Macro aids the user in identifying common problems and solutions for correcting the errors.

Sub Help()
   ' Help Macro
   ' Selects cell A1 in the Help and Instructions Tab
   Sheets("Help and Instructions").Select
   Range("A1").Select
End Sub

The Help Macro activates when the user selects the Help button located in the upper left corner of the Fertilizer Optimization tab. When the user clicks this button the code selects the Help and Instructions tab. Also, the second part of the code selects cell A1 of the active sheet so the user can read directions for solving problems on how to enable Macro features or enable the Solver add-in.
4.3 Hide_Protect Macro

Hiding the underlying formulas and equations of the Fertilizer Optimizer along with protecting the spreadsheet ensures the Tool’s integrity. Using the Hide_Protect Macro simplifies the hiding and protecting process for the programmer.

Sub Hide_Protect()
' Hide_Protect
' Unprotect sheet incase the document was protected
    ActiveSheet.Unprotect Password:="52Fre"

' Hide columns L thru AI in spreadsheet
    Columns("L:AI").Hidden = True

' Password protect the spreadsheet
    ActiveSheet.Protect Password:="52Fre"
    Range("D10").Select
End Sub

The Hide_Protect Macro runs when the programmer selects the Hide & Protect button located near cells L3 and L4 in the Fertilizer Optimization tab. When columns L to AI are hidden in the spreadsheet, the Hide & Protect button cannot be viewed by the programmer or user. The Hide_Protect Macro may also be run by selecting the Macros button in the Developer tab on the Quick Access Toolbar in Excel.

The purpose of the Hide_Protect Macro involves hiding columns L to AI in the spreadsheet so a user does not access the different formulas or equations in the optimization portion of the spreadsheet. To hide these columns the Macro first unprotects the spreadsheet (in case the active sheet has password protection). Next, the program hides column L to AI in the spreadsheet and then password protects the spreadsheet. The active cell then becomes cell D10 and the spreadsheet is ready for use.
4.4 Optimize_Solver Macro

After a user inputs all appropriate data including Crop Selection and Prices, Land Area Requirements, Fertilizer Selection and Prices, and a Budget Constraint the Tool has all necessary data. The Optimize_Solver Macro uses the input data to determine an optimal solution which maximizes net returns to fertilizer uses. Results from the fertilizer optimization are provided in the output tables including the Fertilizer Optimization, Expected Average Effects per Ha, and Total Expected Net Returns to Fertilizer.

Sub Optimize_Solver()
    
    ' Unprotect the spreadsheet if columns
    ActiveSheet.Unprotect "52Fre"

    ' If statement to check if fix land area req. checkbox is selected
    If Range("L25") = True Then
        Call Fix_Land_Area_Req_Checkbox
    End If

    ' Assign values from Area Planted Ha Check (L16:L21) to Area Planted Ha (C16:C21)
    Range("C16") = Range("L16")
    Range("C17") = Range("L17")
    Range("C18") = Range("L18")
    Range("C19") = Range("L19")
    Range("C20") = Range("L20")
    Range("C21") = Range("L21")
    Range("C22") = Range("L22")

    ' Reset all fertilizer application levels to 0
    Range("$O33:$S39") = 0
    Range("$V$89").Select

    ' Clears all previous objective functions or constraints from solver dialog box
    SolverReset

    ' Maximize $V$89 (Net Returns) by changing $O$33:$S$39, $C$16:$C$22
    SolverOK SetCell:="$V$89", MaxMinVal:=1, ValueOf:="0",
    ByChange:="$O$33:$S$39, $C$16:$C$22"

    ' Constraint $O$33:$S$39(Fertilizer Application Amount) >= $O$18:$S$24
    SolverAdd CellRef:="$O$33:$S$39", Relation:=3,
    FormulaText:="$O$18:$S$24"
' Constraint $O33:$S39(Fertilizer Application Amount) <= $T18:$X24 (Fertilizer Max Application Rate)
   SolverAdd CellRef:="$O33:$S39", Relation:=1,
   FormulaText:="$T18:$X24"

' Constraint $AB33:$AD39(Fertilizer Total) <= $Y18:$AA24 (Fertilizer Total Maxes)
   SolverAdd CellRef:="$AB33:$AD39", Relation:=1,
   FormulaText:="$Y18:$AA24"

' Constraint $T55(Fertilizer Cost/Crop) <= $U55 (Budget Constraint)
   SolverAdd CellRef:="$T55", Relation:=1, FormulaText:="$U55"

' Constraint $C16:$C22(Area Planted) >= $F16:$F22 (Min ha)
   SolverAdd CellRef:="$C16:$C22", Relation:=3,
   FormulaText:="$F16:$F22"

' Constraint $C16:$C22(Area Planted) <= $G16:$G22 (Max ha)
   SolverAdd CellRef:="$C16:$C22", Relation:=1,
   FormulaText:="$G16:$G22"

' Constraint $C23(Total Hectares) <= $H16 (Max Total)
   SolverAdd CellRef:="$C23", Relation:=1, FormulaText:="$H16"

' Cancels the solver solution window from appearing
   SolverSolve userFinish:=True

' If statements to check minimum area planted
' If Statement for Maize
   If Range("C16") <= 0.01 Then
      Range("C16") = 0
      Range("O33:S33") = 0
   End If

' If Statement for Sorghum
   If Range("C17") <= 0.01 Then
      Range("C17") = 0
      Range("O34:S34") = 0
   End If

' If Statement for Upland Rice
   If Range("C18") <= 0.01 Then
      Range("C18") = 0
      Range("O35:S35") = 0
   End If

' If Statement for Beans
   If Range("C19") <= 0.01 Then
      Range("C19") = 0
      Range("O36:S36") = 0
   End If
' If Statement for Finger millet
    If Range("C20") <= 0.01 Then
        Range("C20") = 0
        Range("O37:S37") = 0
    End If

' If Statement for Soybeans
    If Range("C21") <= 0.01 Then
        Range("C21") = 0
        Range("O38:S38") = 0
    End If

' If Statement for Groundnuts Unshelled
    If Range("C22") <= 0.01 Then
        Range("C22") = 0
        Range("O39:S39") = 0
    End If

' Reposition the active window
    Range("B4:2").Select
    ActiveWindow.SmallScroll Down:=10

' If statement to password protect the spreadsheet if columns L thru AI are not hidden
    If Columns("L:AI").Hidden = True Then
        ActiveSheet.Protect Password:="52Fre"
    End If

End Sub

The Optimize_Solver Macro runs when the user selects the Optimize button located underneath the Budget Constraint Table. When a user clicks the Optimize button the macro uses the farmer specific input data with the Solver Add-in to determine an optimized solution that maximizes returns to fertilizer use on a constrained financial investment. The code first unprotects the password protected spreadsheet so values may be changed in the necessary cells. Next, an if statement calls or runs the Fix_Land_Area_Req_Checkbox Macro if the checkbox to the right of the Fix land area req. in cell F23. This code is rerun in case the user has made any changes to the land allocations after previously selecting the checkbox which ran the Fix_Land_Area_Req_Checkbox Macro.

After the completion of the two if statements, a series of range statements assigns values from L16:L22 to the correlating cells in C16:C22. A series of if statements embedded in the worksheet from L16:L22 checks whether a particular crop should be considered in the optimization. If the area planted remains at 0 the Solver add-in will not consider the crop in the optimization. Those crops needing consideration in the optimization will either be assigned a fixed amount if the Fix land area req. checkbox is selected or a small default value of 0.0001 if a Max Planted (Ha) of greater than 0 has been assigned.
Next, the code in the Optimize_Solver Macro sets cells O33:S39 to a value of 0 which resets the fertilizer application amounts from any prior optimization. After resetting these cells, the code makes V89 (objective function) the active selection. The SolverReset statement then opens the Solver dialog box and clears out all prior objective functions and constraints. The macro then proceeds with setting the objective function in cell V89 to a maximum value by changing the fertilizer application rate in cells O33:S39 and the land area planted in cells C16:C22.

The Optimizer_Solver Macro code then proceeds with adding seven constraints into the Solver dialog box. First the fertilizer application amount in cells 033:S39 must be greater than or equal to the fertilizer minimum application rate in cells O18:S24. Also, the fertilizer application amount in cells O33:S39 must be less than or equal to the fertilizer maximum application rate in cells T18:AA24. The third constraint sets the fertilizer totals in cells AB33:AD39 to be less than or equal to the fertilizer total maxes in Y18:AA24. To keep fertilizer costs under the budget constraint, the sum of the fertilizer costs per crop in cell T55 must be less than or equal to the budget constraint in cell U55.

The fifth constraint keeps the area planted in cells C16:C22 greater than or equal to the minimum area planted in cells F16:F22. Also, the sixth constraint requires the area planted in cells C16:C22 to be less than or equal to the maximum area planted in cells G16:G22. The final constraint keeps the total hectares in cell C23 less than or equal to the max total hectares in cell H16. After inputting all of the optimization constraints, the VBA code selects solve on the Solver dialog box. Solver then runs the optimization algorithm to find the optimal solution. The macro cancels the option to display the Solver Results window. Finally, a series of if statements check to make sure that Solver does not provide any acreage allocation which is less than 0.01 acres for any particular crop. In the event an if statement finds a solution which is less than 0.01 acres, the allocation for that particular crop has a value of 0 assigned. The Macro then password protects the spreadsheet if columns L:AI are hidden.
4.5 Print_Output Macro

Upon the completion of an optimization the user may want to print the recommendations for acreage allocations and fertilizer application rates. The Print_Output Macro selects the output and takes the user to a preview of the active selection before printing.

```
Sub Print_Output()
'
' Print_Output Macro
'
'
' Select A1 to I68 for print preview
  Range("A1:I68").Select
  Selection.PrintOut Copies:=1, Preview:=True

End Sub
```

The Print_Output Macro runs when the user selects the Print Output button located underneath the Total Expected Net Returns to Fertilizer Table. When the user clicks the Print Output button, the code makes Range A1 to I68 the active selection. Next, the code proceeds to take the active selection to a preview before printing. In this preview the user can change the number of copies being printed and select the option on whether to print the selection in color or black and white.

Excel 2007 and 2010 have slightly different previews for the viewing of an active selection before printing. In Excel 2007 Print Preview refer to the viewing of a document before printing and the user must enter into a different window to change the various print options. In Excel 2010 the preview and print options are found in the same view. In either case the Print_Output Macro brings the user to a preview where the user can see the active selection before printing.
4.6 Reset_Form Macro

Resetting the input forms of the Fertilizer Optimizer including the Crop Selection and Prices, Land Area Requirements, Fertilizer Selection and Price, and Budget Constraint to the predefined default values erases all previously entered data and optimized solutions. The Reset_Form Macro selects and resets the input forms to specified default values allowing the user to enter new data for a different scenario.

Sub Reset_Form()
  ' Reset_Form Macro
  
  ' Reset Crop Selection and Prices
  Range("C10:D10, C11:D11") = "xxx"
  Range("C16:D22") = 0

  ' Reset Land Area Requirement to 0
  Range("F16:G22, H16") = 0

  ' Uncheck Land Area Requirement by changing L24 to a value of False
  Range("L25") = "False"

  ' Reset Selection and Prices
  ' Assign default values to urea
  Range("C27") = "46%"
  Range("D27, E27") = "0%"
  ' Assign default values to TSP
  Range("C28, E28") = "0%"
  Range("D28") = "46%"
  ' Assign default values to DAP
  Range("C29") = "18%"
  Range("D29") = "46%"
  Range("E29") = "0%"
  ' Assign default values to KCL
  Range("C30, D30") = "0%"
  Range("E30") = "60%"
  ' Assign default values to xxx
  Range("B31") = "xxx"
  Range("C31:E31") = "%"
  ' Assign price values of 0
  Range("F27:F31") = 0

  ' Reset all Fertilizer Application amounts to 0
  Range("O33:S39") = 0

  ' Reset budget constraint to 0
  Range("C34") = 0

  ' Reposition the user's window
  Range("V78").Select
End Sub

The Reset_Form Macro runs when the user selects the Reset Form button located underneath the Budget Constraints Table. When the user clicks the Reset Form button the code resets the input forms of the Fertilizer Optimizer to default values for a new optimization. First the code selects the Producer name and Prepared By cells of C10 to C11 and sets these cells to a value of “XXX”. Next, the cells of Crop Selection and Prices including the Area Planted (Ha) and Expected Grain Value/kg of C16:D22 are set to a value of 0. Constraints for the Land Area Requirements involving the Min Planted (Ha), Max Planted (Ha), and Max Total (Ha) also are assigned a value of 0.

Next, to reset the Fix land area req. checkbox in the Land Area Requirements form cell L24 is assigned a value of FALSE. When the Fix land area req. checkbox is selected cell L24 has a value of TRUE automatically assigned and when deselected the value changes to FALSE. The Reset_Form Macro then proceeds to setting the Fertilizer Selection and Prices form back to default values. The default values for N, P2O5, and K2O are set in terms of percentages and the Price/50 kg bag assume a value of 0. Finally, the Reset_Form Macro assigns a value of 0 to the Budget Constraint and repositions the screen of the Fertilizer Optimizer for entering the next set of data for a new optimization.
4.7 Unprotect_Unhide Macro

Unprotecting and unhiding the formulas and equations of the optimization allow for updating or editing the Fertilizer Optimizer. Using the Unprotect_Unhide Macro along with a keyboard shortcut speeds up the process so the programmer may easily work on the Tool.

```vba
Sub Unprotect_Unhide()
' Unprotect_Unhide Macro
'
' Unprotect the spreadsheet and unhide columns K thru AJ
    ActiveSheet.Unprotect Password:="52Fre"
    Columns("K:AJ").Hidden = False
    Range("O33").Select
End Sub
```

The Unprotect_Unhide Macro runs when the programmer selects CTRL + SHIFT + U on the keyboard. After pressing these three keys, the code first unprotects the password protected spreadsheet and then proceeds to unhide columns K to AJ containing the major formulas and equations of the Tool. Finally, the code selects O33 as the active cell. The programmer may now proceed with editing the different equations and formulas of the Fertilizer Optimizer spreadsheet.

The purpose of hiding columns in-between K to AJ remains in protecting the integrity of formulas and equations. Also, password protecting this spreadsheet guards against users from mistakenly editing the formulas or equations in columns L to AI. As long as these columns stay unhidden the spreadsheet is unprotected and the programmer may go about editing the spreadsheet. Upon running the Hide_Protect Macro the spreadsheet becomes protected and the only editable cells are the ones not locked (input ranges in tables with the yellow row headings).

The eight Macros programs in Section 4.1 to 4.8 of Chapter 4 automate the processes of the Fertilizer Optimizer spreadsheet. These programs may occasionally fail due to different errors. Chapter 5 explains how to troubleshoot common errors with Macros in Excel and correct the problems. Additional references are also provided in Chapter 5 to elaborate further on VBA for Macros programs.
CHAPTER 5: Troubleshooting Macros

Macro Programs embedded in the Uganda Fertilizer Optimizer may malfunction due to an error occurring within the program. Chapter 5 explains ways to troubleshoot these errors and methods to remedy the problems. Additional references are also provided in the chapter to aid users in developing and editing macro programs in Excel.

5.1 Common Problems

Programmers working on editing or enhancing Macro programs in the Uganda Fertilizer Optimizer spreadsheet may come across errors during the testing phase of the Excel spreadsheet. Also, including additional Macros in the spreadsheet may cause problems. In either case the Tool must be error free and validated before being released. Due to the popularity of Excel with VBA Macros many users and programmers have documented their problems on various discussion board postings or websites across the Internet.

The authors of this programmer manual have found describing Macro problems combined with key search words on Internet search engines such as Google produces helpful discussion board postings and informative YouTube clips. One particular website has not served as the sole source for referencing various problems with the Fertilize Optimizer spreadsheet, but a variety of different VBA for Excel websites and short video clips for troubleshooting various problems and issues has provided the greatest benefit.

When searching for different VBA references or solutions for errors, start by using terms describing the need coupled with phrases such as VBA Excel 2010 or VBA Excel macros. Refine the internet searches until a reputable source may be located. Performing additional searches usually leads to an appropriate source or discussion posting. Many VBA programmers have documented these problems on various blog discussion postings of various programming websites. Readers can join and access these websites for free to learn tips on VBA programming and read about different ways to troubleshoot errors.

5.2 Additional References

The following links provide a short list to reputable reference websites used in the design and validation phase of the Fertilizer Optimizer spreadsheet. The links to these resources were current during the writing of this manual and may become broken over time (perform a quick search engine query for the title of the link if a hyperlink does not work).

The following reference provides a good overall source of information on the Visual Basic for Application in Excel 2010 (applicable to earlier version of Excel as well) from Microsoft.

*Getting Started with VBA in Excel 2010 – Microsoft Office 2010:*

The second reference serves as a good discussion board website that helps group participants and readers troubleshoot problems in Excel. Archives in these discussion forums are particularly useful.
The two links previously provided are just examples of the different kinds of websites available for free to users and programs using VBA in Excel. Internet search engine queries may provide better resources. Always look for reputable websites when evaluating the value of a particular resource.
References


