

INPUTS

The Jacobs Levy Markowitz Simulator

Bruce I. Jacobs, Kenneth N. Levy, and Harry M. Markowitz

Inputs

In order to produce a simulation of a financial market, JLMSim must be fed data in a manner that it expects. JLMSim reads data from a standard text file named `JLMSimInput.txt`. An example is provided with the JLMSim download.

There are two ways to create the financial system data for JLMSim. The user can create `JLMSimInput.txt` directly with a standard text editor. (The example file contains many comments that will aid the user who chooses to create the file manually.) Alternatively, the user can employ the Graphical User Interface (GUI) provided with JLMSim. The GUI—separate from JLMSim itself—generates the `JLMSimInput.txt` file automatically. With the Input GUI, the user performs the following steps:

1. creates or imports a UI Form file,
2. edits the UI Form file, and
3. generates a `JLMSimInput.txt` file.

The user can then run the JLMSim program.

To launch the Input GUI, the user clicks on the JLMSimInput shortcut listed under Start/Programs/JLMSim. When the program begins, it shows the screen illustrated in Figure 1. This screen represents a tabbed workbook. Each worksheet in the workbook represents a particular type of input to JLMSim or a different type of entity.

Most of the fields in a new workbook contain common default values. The user can accept these default values, modify them by entering information directly into each input field in each worksheet, or change all the information automatically by importing an existing UI Form file. Once the user is satisfied with the parameters for the simulation run, the data can be saved in UI Form for later use and exported in UI Form to a `JLMSimInput.txt` file.

This chapter describes the input data required by JLMSim and the use of the GUI for generating the data. Because JLMSim can operate in two modes—the Dynamic Analysis (DA) mode and the Capital Market Equilibrium (CME) mode—this chapter is divided into two parts, each of which describes the inputs for one of the operating modes. The JLMSim download includes two sample UI form files, `sample_da.txt` and `sample_cme.txt`, which demonstrate the input file format for each of these two modes.

The Dynamic Analysis (DA) Mode

The user can select the particular mode of operation—DA or CME mode—from the drop-down box in the Mode of Operation panel. The basic objective of JLMSim in DA mode is to model the evolution of market prices and volumes over time. This section describes the operation of the GUI generally and the specific inputs used for the DA mode.

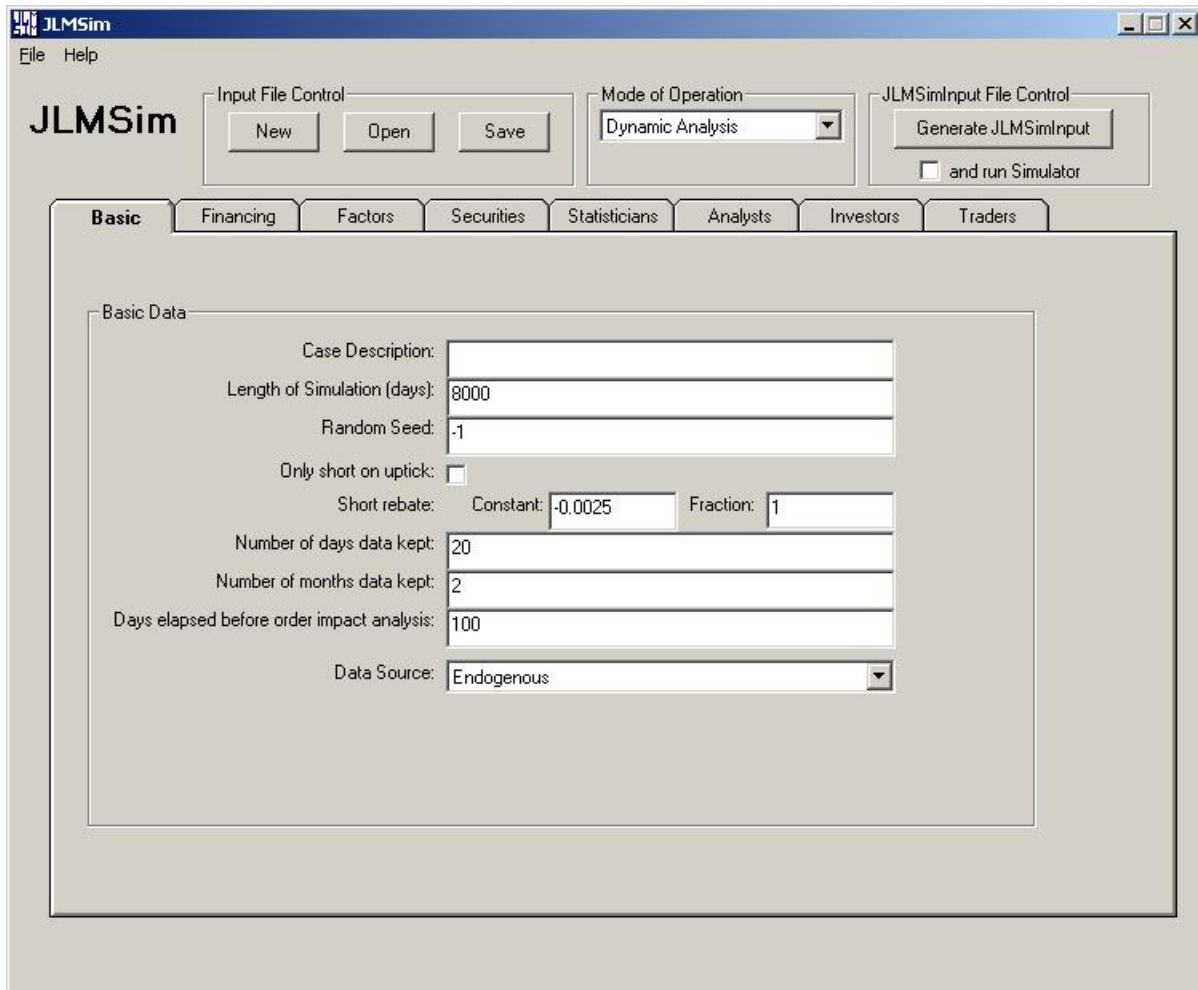


Figure 1: The New Workbook.

UI Form Files and JLMSimInput.txt Files

To create a new UI Form, the user can click the New button in the Input File Control pane. This creates a workbook (partially completed using reasonable default values) into which the user can type information. Alternatively, the user can load an existing UI Form file either by clicking the Open button in the Input File Control pane or by pulling down the File menu and choosing the Open menu item. In either case, the Input GUI will display a File Open dialog box, which enables the user to navigate to a desired folder and select an appropriate UI Form file. Once the user has loaded the sample UI Form file `sample_da.txt` into the Input GUI, all fields are completed. Figure 2 provides an example.

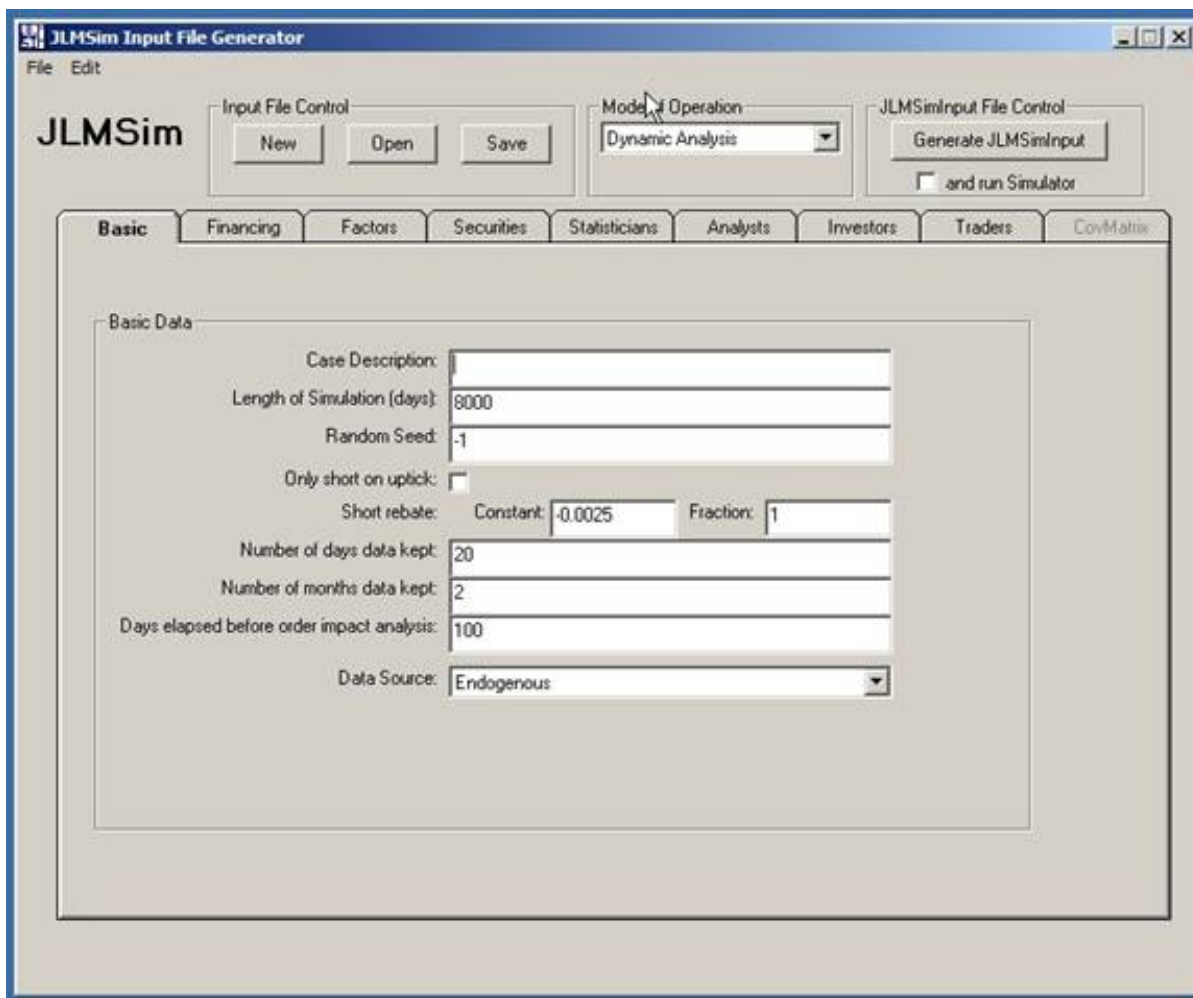


Figure 2: The Basic Worksheet in DA Mode.

When the user is satisfied with the simulation parameters that have been entered, the information can be saved into a new or an existing UI Form file by clicking the Save button in the Input File Control pane or by pulling down the File menu and choosing the Save menu item. In either case, the Input GUI will display a File Save dialog box, which enables the user to navigate to an appropriate folder and save the UI Form file.

To generate the `JLMSimInput.txt` file in preparation for use by the JLMSim program, the user clicks the Generate JLMSimInput button in the JLMSim File Control pane. This will reveal a File Save dialog box, which enables the user to navigate to the folder in which the JLMSim program resides and save the file as `JLMSimInput.txt`. Alternatively, the user can choose to save the `JLMSimInput.txt` file and run the JLMSim program immediately by checking the "and run Simulator" box in the JLMSimInput File Control pane and then clicking the Generate JLMSimInput button.

All simulation input information is represented in worksheets labeled "Basic," "Financing," "Factors," "Securities," "Statisticians," "Analysts," "Investors," and "Traders." (In the CME Mode, discussed later, the "Covariance Matrix" worksheet also becomes active). The user can navigate from worksheet to worksheet by clicking on the appropriately labeled tabs or by pulling down the Edit menu and selecting the appropriate item. The following subsections describe the information in each worksheet.

Basic Worksheet in DA Mode The basic worksheet in DA mode, shown in Figure 2, contains the following fields, which control general parameters of the simulation.

1. *Case Description.* This is a user-supplied mnemonic for identifying the particular case; for example, it may be an abbreviated description of input parameters. It should contain no spaces, colons, or characters not valid in a file name. The simulation uses this mnemonic in the names of several files that it creates for the run.
2. *Length of Simulation.* This is the number of days that the simulation should run.
3. *Random Seed.* This is the initial seed used by the internal random number generator. A value of -1 instructs the simulation program to use the default random seed. At the end of a simulation, a file named `>Messages...csv` is created; one of the final outputs of this file is the "next-to-be-used random seed." We recommend using this saved seed in a subsequent simulation run if the user wants a new, non-default random seed.
4. *Only short on uptick.* A check in this field indicates that the program should enforce an uptick rule. The absence of a check indicates that the rule will not be enforced.
5. *Short rebate.* The proceeds of any short sale, posted as collateral with the securities' lenders, earn daily interest (the short rebate) at an annual rate expressed as a decimal equal to the value in the "constant" field plus the value in the "fraction" field times the rate that the broker pays, as defined in the Financing worksheet discussed below.
6. *Number of days data kept.* Some estimation procedures use historical daily returns and some trading rules use recent end-of-day prices. The entry in this field indicates the number of days of data the program keeps for possible use in historical analysis.
7. *Number of months data kept.* Some estimation procedures use historical monthly returns. The entry in this field indicates the number of months the program keeps data for possible use in historical analysis.
8. *Days elapsed before order impact analysis.* JLMSim generates an `Order Impact Analysistxt` file as described in Chapter 3. JLMSim can skip sending output to this file for the early part of a simulation, if desired. The entry in this field indicates how many days are to be skipped before the start of the Order Impact Analysis.
9. *Data Source.* When JLMSim is in DA mode, the user can control the source of the data used to populate statisticians' databases before the start of the simulation. The user can select endogenous from the drop-down box, in which case historical returns will reflect actual returns in the simulation as these are generated. If the user selects exogenous instead, the return series used to estimate expected returns and covariances recycle the returns drawn from a factor model before the start of the simulation. The series used will cycle through the kept days or kept months, returning to the beginning of a series when the end of the series is reached. The choice of exogenous or endogenous affects the source of return data only, not the source of recent price data used by some trading rules.

The Financing Worksheet in the DA Mode

The financing worksheet, shown in Figure 3, controls simulation-wide information pertaining to the financing of trades. The worksheet contains the following fields.

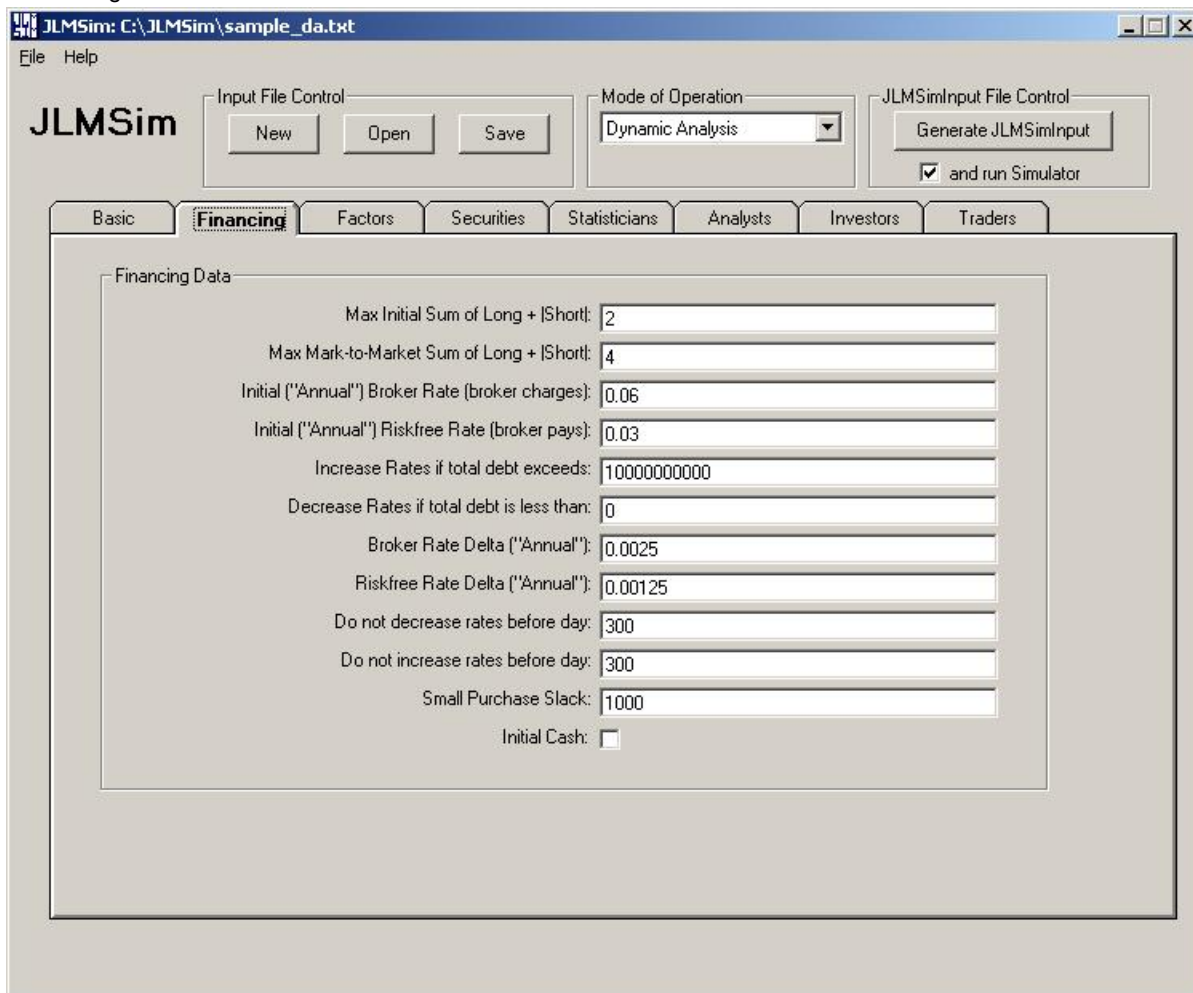


Figure 3: The Financing Worksheet in DA Mode.

1. *Max Initial Sum of Long + |Short|*. The entry in this field is a system-wide (all-investor) maximum on initial leverage. Leverage is a ratio with a numerator equal the sum of the value of long positions and the absolute value of short positions and a denominator equal to the equity in an account. Any transaction that would violate the initial leverage constraint cannot be executed. Individual investor templates may have constraints on initial leverage that are more restrictive than this system-wide constraint.
2. *Max Mark-to-Market Sum of Long + |Short|*. The entry in this field is the maintenance or mark-to-market constraint parameter. If leverage exceeds this value (because of market movements), then an account must liquidate enough assets to bring it back into compliance with this constraint.
3. *Initial (\"Annual\") Broker Rate (broker charges)*. The entry in this field controls the annual rates that brokers charge. This annual rate is divided by the number of trading days in a simulated year to get the daily rate.
4. *Initial (\"Annual\") Riskfree Rate (broker pays)*. The entry in this field controls the annual rates that brokers pay. This annual rate is divided by the number of trading days in a simulated year to get the daily rate.
5. *Increase Rates if total debt exceeds*. The entry in this field indicates the level of total indebtedness that will trigger an increase in interest rates. The program checks this trigger point once per simulated month. A value of 10,000,000,000, as shown in Figure 3, will deactivate the trigger.
6. *Decrease Rates if total debt is less than*. The entry in this field indicates the level of total indebtedness that will trigger a decrease in interest rates. The program checks this trigger point once per simulated month.
7. *Broker Rate Delta (\"Annual\")*. The entry in this field is the incremental increase or decrease applied to the annual rate brokers charge if debt trigger points are breached. The rate is increased by this amount if total debt is greater than or equal to the upper trigger point; it is decreased by this amount if total debt is less than the lower trigger point.
8. *Riskfree Rate Delta (\"Annual\")*. The entry in this field is the incremental increase or decrease applied to the annual rate brokers pay if debt trigger points are breached. The rate is increased by this amount if total debt is greater than or equal to the upper trigger point; it is decreased by this amount if total debt is less than the lower trigger point.
9. *Do not reduce rates before day*. The user may find it desirable to wait some interval of simulated time before lowering interest rates. This entry indicates the number of days the program should wait after the beginning of the simulation before lowering interest rates. If the entry in this field is set to a number greater than the number of days in the simulation, interest rates are never lowered.

10. *Do not increase rates before day.* The user may find it desirable to wait some interval of simulated time before increasing interest rates. This entry indicates the number of days the program should wait after the beginning of the simulation before raising interest rates. If the entry in this field is set to a number greater than the number of days in a simulation, interest rates are never raised.
11. *Small Purchase Slack.* The entry in this field is the minimum wealth, in number of dollars, that an investor must have before being allowed to purchase any further shares.
12. *Initial Cash.* This check box informs JLMsim whether or not investors' initial portfolios contain an initial cash allocation. If this box is not checked, the investors' portfolios will contain only the securities defined in the securities worksheet. If this box is checked, investors' initial portfolios may contain cash, and an additional entry field, "Initial Cash Fraction," appears.
13. *Initial Cash Fraction.* If investors are allowed to hold cash, this field determines the fraction of every investor's portfolio that is allocated to cash. For example, an entry of 0.3 indicates that 30% of every investor's portfolio is allocated to cash.

The Factors Worksheet

At the start of the simulation, the historical data in the statisticians' databases are generated by a factor model. The factors worksheet, shown in Figure 4, controls the parameters of the factor model.

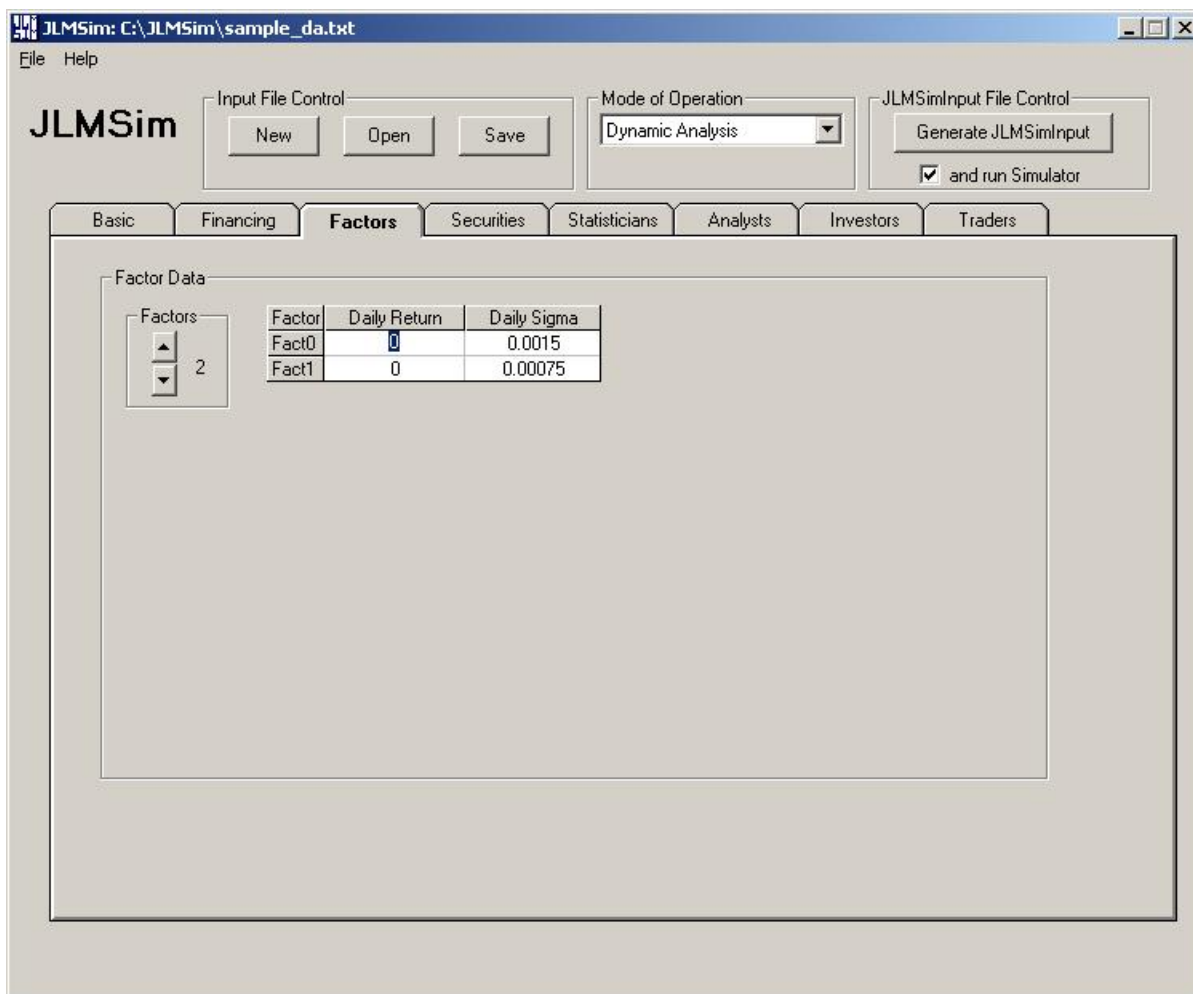


Figure 4: The Factors Worksheet.

The up-arrow and down-arrow buttons labeled Factors at the top left corner of the Factor Data pane control the number of factors that the program uses in the factor model. The number of factors in use is indicated immediately to the right of the buttons. Pressing the up button increases the number of factors. Each time the number of factors is increased, another row is added to the table in the worksheet, and another column is added to the table in the securities worksheet. Similarly, pressing the down button causes the numbers of factors to decrease.

Note that the factors are labeled from a base of zero. Thus a two-factor model has factors labeled "Fact0" and "Fact1." This labeling convention is the same as the one used in C++ and applies whenever multiple entities exist within JLMsim.

The table in the Factors worksheet requires two parameters for each factor.

1. *Daily Return*. This is the expected daily return provided by the factor, measured as a fraction. An entry of 0.0001 would indicate an expected return of one basis point (one-hundredth of 1 percent) per day.
2. *Daily Sigma*. This is the standard deviation of the daily return, also measured as a fraction. An entry of 0.0015 would indicate a standard deviation of 15 basis points per day.

The Securities Worksheet in DA Mode

The securities worksheet controls the number of securities (other than cash and loans) used in the simulation, as well as various parameters for each security. The number of securities is controlled by the up- and down-arrow buttons at the top left of the Security Data pane. The parameters pertaining to each security are contained in the table.

In the DA mode, the securities worksheet, shown in Figure 5, contains as many columns as there are factors in the factors worksheet, plus four other columns. The entries in the first column determine the price of each security at the start of the simulation. The second column is the average pre-simulation daily trading volume. The remaining columns are inputs for the factor model used to generate returns for the statisticians' databases before the start of the simulation. In particular, the third and fourth columns control each security's daily alpha and daily idiosyncratic risk. The last columns control each security's beta with respect to each factor.

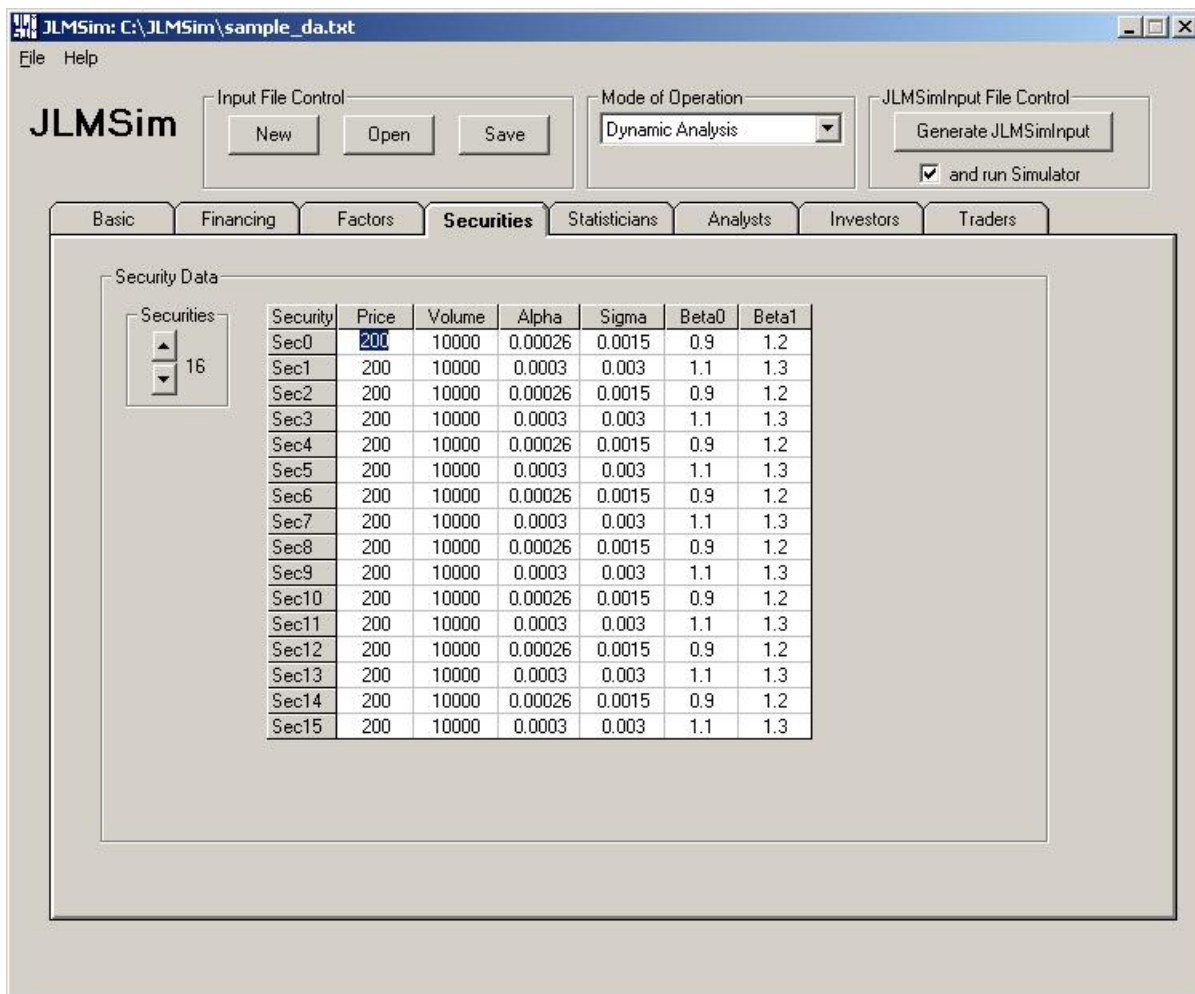


Figure 5: The Securities Worksheet in DA Mode.

The Statisticians Worksheet in DA Mode

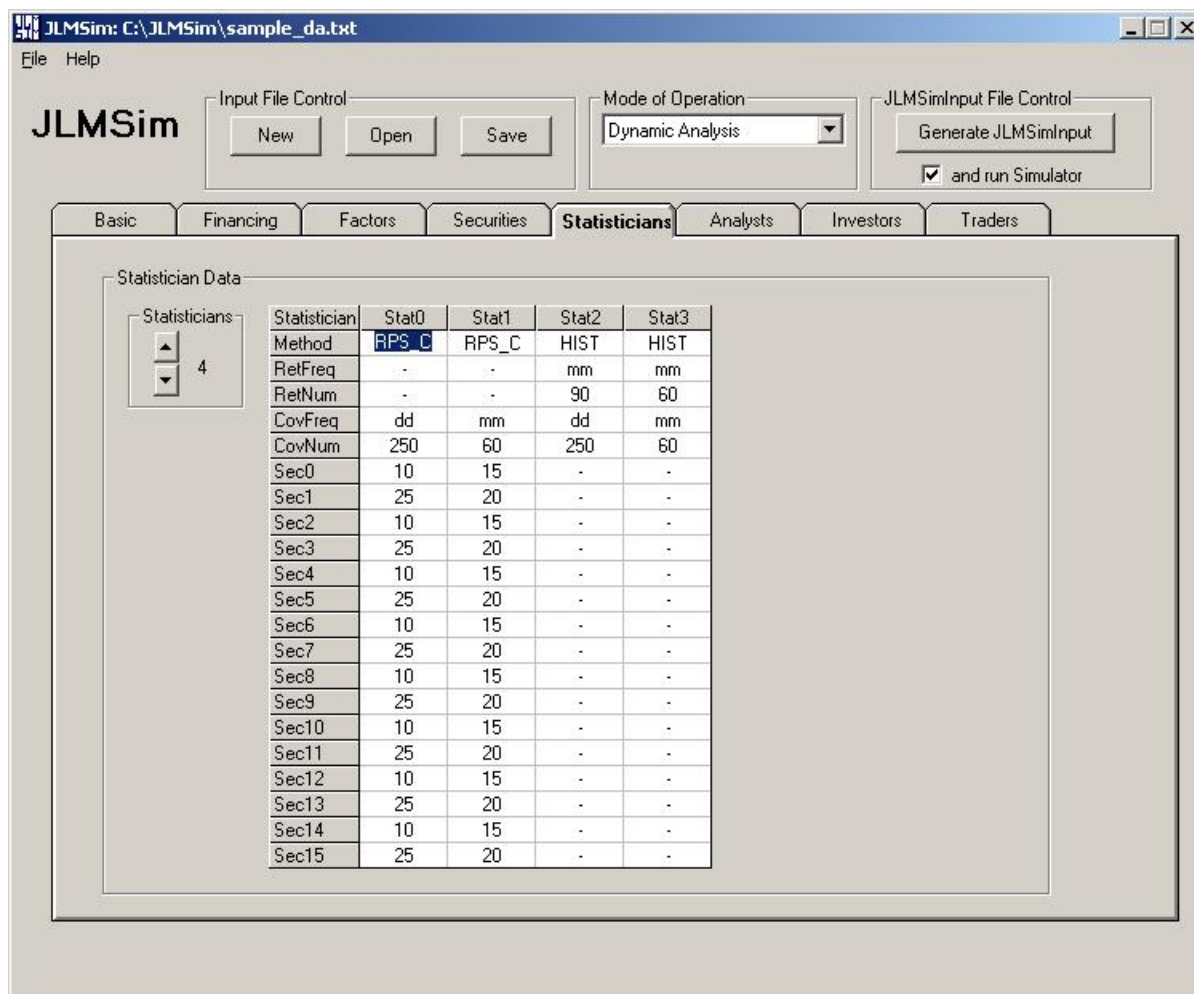


Figure 6: The Statisticians Worksheet in DA Mode.

The statisticians worksheet, shown in Figure 6, controls the number of statisticians and the following parameters for each statistician.

1. **Method.** The entry in this field determines the method that the corresponding statistician uses to estimate expected returns. At present, the following two methods are permitted:
 - o *RPS_C*, or returns per share constant, where a constant number of dollars returned per share is used to estimate expected returns. The expected return per dollar invested is this number divided by price.
 - o *HIST*, or historical returns, where historical average returns are used to estimate expected returns.
2. **RetFreq.** This row and the following one are used only if the statistician uses the *HIST* method of expected return estimation. The entry in this field determines the frequency of observation that the statistician uses when estimating expected returns. At present, the following two frequencies are permitted:
 - o *dd*, indicating daily observation, and
 - o *mm*, indicating monthly observation.
3. **RetNum.** The entry in this field determines the number of observations that the statistician uses when estimating expected returns. For example, a 60 in this field and an *mm* in the preceding field indicate that the statistician uses 60 months of data.
4. **CovFreq.** This row and the following one are analogous to the previous two rows but relate to the estimation of covariances rather than returns. The entry in this field determines the frequency of observation that the statistician uses when estimating covariances of returns. At present, the following two frequencies are permitted:
 - o *dd*, indicating daily observation, and
 - o *mm*, indicating monthly observation.
5. **CovNum.** The entry in this field determines the number of observations that the statistician uses when estimating covariances of returns.
6. **Sec0—Sec15.** The remaining rows in the table are visible and used if the statistician uses the *RPS_C* method to estimate returns. An entry in any one of these fields specifies the constant number of dollars that the statistician uses to estimate security return. For example, an entry of 10 with a corresponding security price of \$200 would indicate an expected return of 0.05 (i.e., a 5 percent return per dollar invested).

The Portfolio Analysts Worksheet in DA Mode

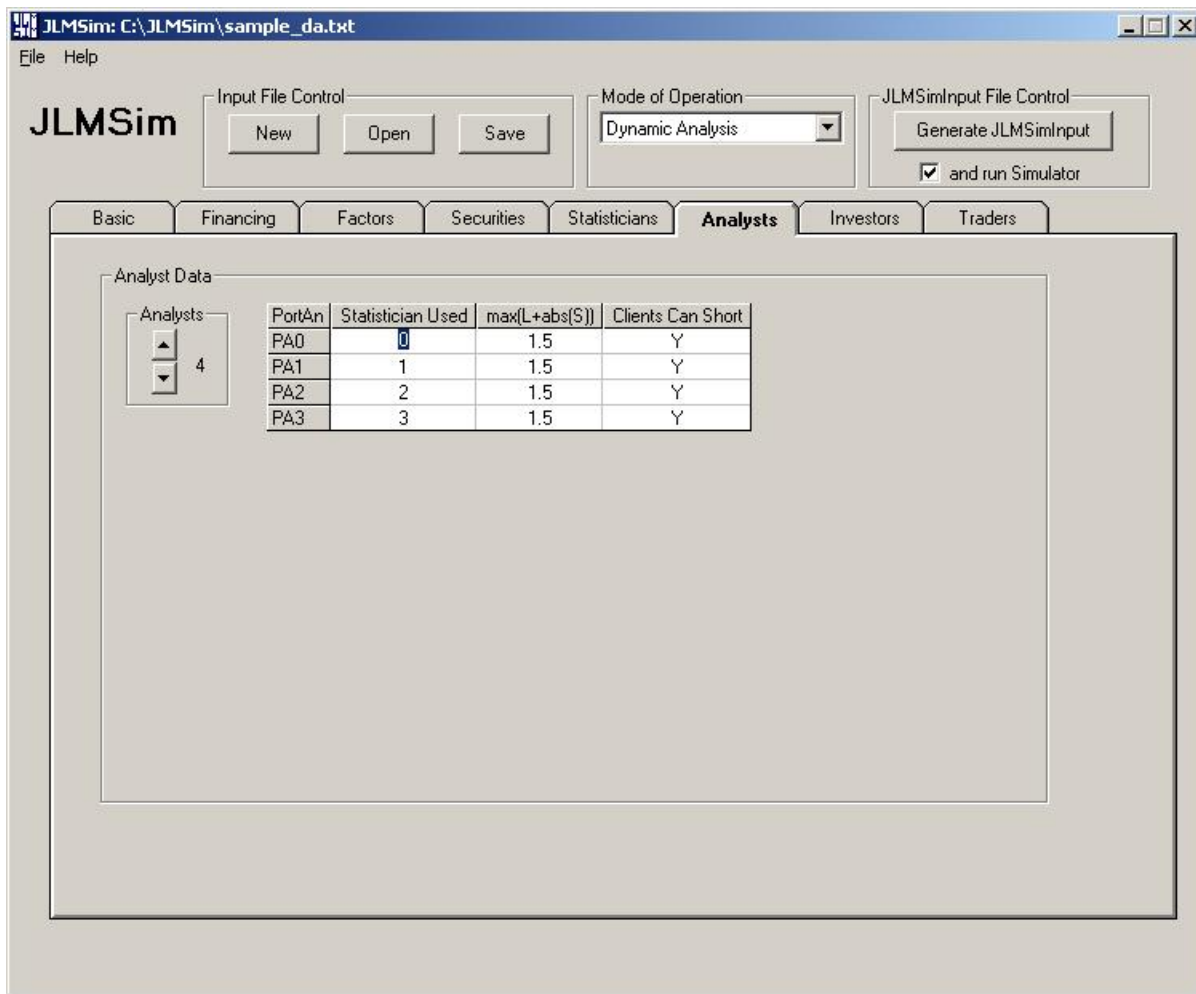


Figure 7: The Portfolio Analysts Worksheet in DA Mode.

The portfolio analysts worksheet, shown in Figure 7, controls the number of portfolio analysts used in the simulation, as well as the following parameters for each portfolio analyst.

1. *Statistician Used.* The entry in this field determines the statistician (as defined in the statisticians worksheet) that the portfolio analyst uses for obtaining estimates of returns, variances, and covariances.
2. *max(L + abs(S)).* The entry in this field is the upper bound on the leverage that the corresponding portfolio analyst can use when computing the efficient frontier. The upper bound actually used in the computation of the efficient frontier will be the smaller of the number presented here and the global "Max Initial" presented in the financing worksheet.
3. *Clients Can Short.* A Y in this entry indicates that the corresponding portfolio analyst's clients are allowed to sell securities short. An N indicates that they may not.

The Investor Templates Worksheet

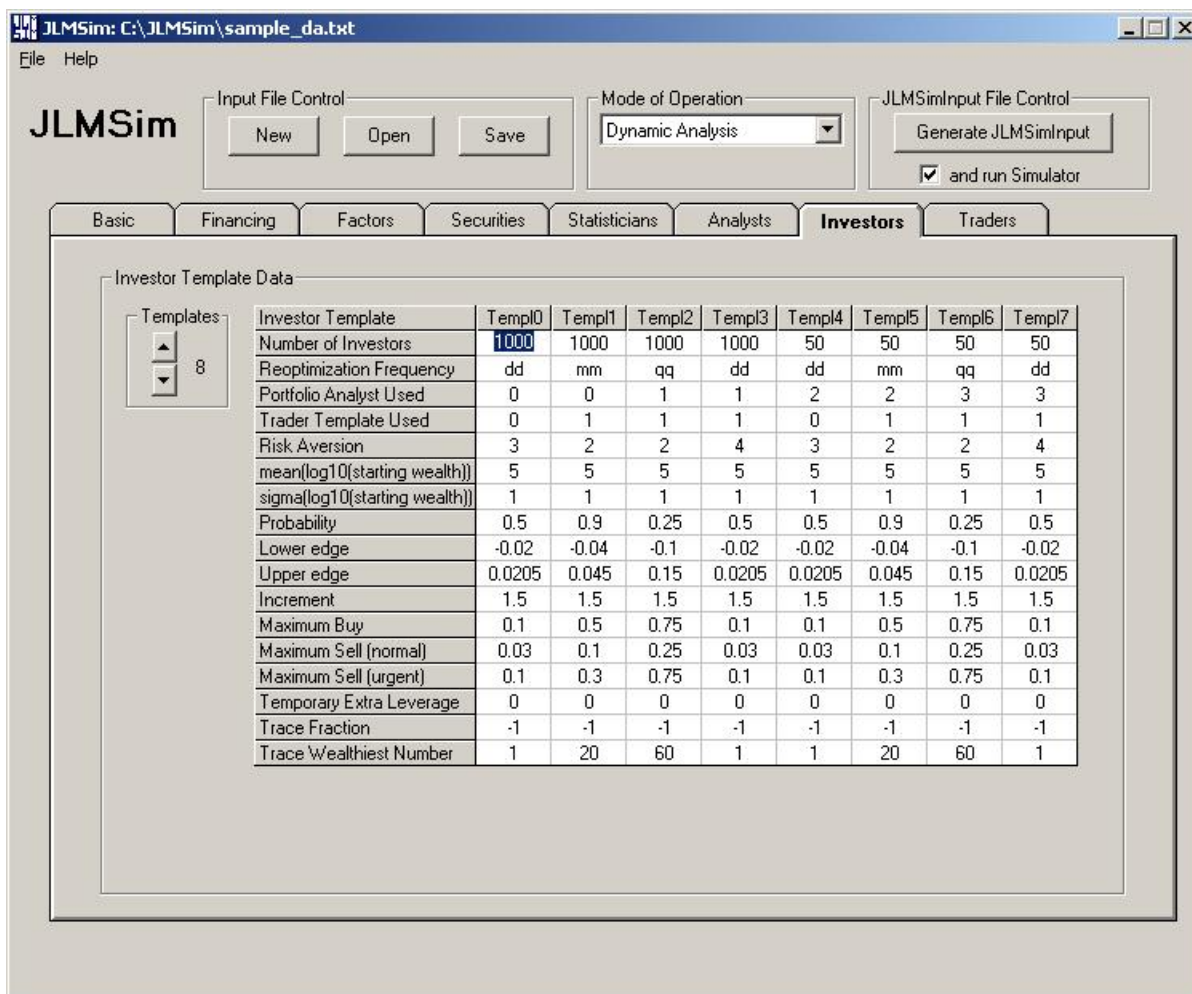


Figure 8: The Investor Templates Worksheet.

Each investor is stamped out from one or another investor template. The investor templates worksheet, shown in Figure 8, controls the number of investor templates and the following parameters pertaining to each.

1. *Number of Investors.* The entry in this field determines how many investors there are for the corresponding investor template.
2. *Reoptimization Frequency.* The entry in this field determines how frequently investors of the corresponding investor template reoptimize. At present, permitted frequencies are
 - o *dd*, indicating daily reoptimization,
 - o *mm*, indicating monthly reoptimization,
 - o *qq*, indicating quarterly reoptimization, and
 - o *yy*, indicating annual reoptimization.
3. *Portfolio Analyst Used.* The entry in this field determines which portfolio analyst (as described in the portfolio analysts worksheet) the investors of the corresponding investor template use.
4. *Trader Template Used.* The entry in this field determines which trader (as described in the trader templates worksheet) the investors of the corresponding investor template use.
5. *Risk Aversion.* The investors of a given template select portfolios that maximizes $E-KV$, where E is the expected portfolio return, V is the variance of the portfolio's return, and K is the investor's risk aversion. The entry in this field is the risk aversion level of investors of the corresponding template.
6. *mean(log10(starting wealth)).* JLMsim assumes that initial wealth of each investor in a particular investor template is drawn from a log-normal distribution. The entry in this field is the mean value of the base-10 logarithm of the starting wealth.
7. *sigma(log10(starting wealth)).* JLMsim assumes that the initial wealth of each investor in a particular template is drawn from a log-normal distribution. The entry in this field is the standard deviation of the base-10 logarithm of starting wealth.
8. *Probability.* This and the following two rows pertain to random deposits and withdrawals when the investor reoptimizes. The entry in this field is the probability that a deposit will occur as of any reoptimization. The user can suppress random deposits and withdrawals by setting this entry to zero.
9. *Lower edge.* Money flows for each investor template are drawn randomly from a uniform distribution. If a draw is negative, the flow is interpreted as a withdrawal; if a draw is positive, it is interpreted as a deposit. The entry in this field is the lower edge of the probability distribution, expressed as a fraction of investor equity.

10. *Upper edge*. The entry in this field is the upper edge of the uniform probability distribution from which money flows are drawn, expressed as a fraction of investor equity.
11. *Increment*. An investor may borrow to pay a withdrawal if there is not enough cash available. The entry in this field determines the maximum leverage permitted to finance a withdrawal. In particular, this entry is added to the portfolio analyst's $\max(L+\text{abs}(S))$ policy parameter to obtain the maximum leverage permitted to accommodate a withdrawal request. If a withdrawal cannot be paid out of cash or borrowing, it becomes a "withdrawal due" and will be paid when cash or borrowing capacity permits.
12. *Maximum Buy*. This row, and the two following ones, present parameters that reduce account turnover. The trades initiated after a reoptimization will attempt to move the investor in a straight line from current portfolio to desired portfolio, and the investor will attempt to move along this line as far as possible without violating any of the constraints defined by the entries in this and the following two rows. The entry in the Maximum Buy field defines the maximum amount of purchases as a fraction of the portfolio. For example, an entry of 0.10 indicates that, as of any reoptimization, the corresponding investor's purchases cannot exceed 10 percent of the equity of the portfolio.
13. *Maximum Sell (normal)*. The entry in this field defines the maximum total sales as a percentage of the portfolio, assuming the investor is not in an urgent situation. For example, an entry of 0.03 indicates that the investor cannot sell an amount greater than 3 percent of portfolio value. A situation is considered urgent if there are withdrawals due or if the required withdrawal would cause leverage to exceed the portfolio analyst's $\max(L+\text{abs}(S))$ policy parameter.
14. *Maximum Sell (urgent)*. The entry in this field defines the maximum sale per reoptimization if the investor's situation is urgent.
15. *Temporary Extra Leverage*. When a reoptimization calls for both purchases or short sales on the one hand and sales or cover actions on the other, the purchase or short sale orders are not placed if their execution alone (without execution of any sales or cover orders) would be "too large," where too large is defined to be the limit used in the optimization plus the temporary extra leverage specified in the entry in this field.
16. *Trace Fraction*. The entry in this field defines, for each investor template, how many investors will be followed in the trace file, which is described in Chapter 3. A number N indicates that one out of every N investors of the template is to be traced. For example, a value of 1000 indicates that one out of every 1000 investors will be traced. A value of -1 indicates that this feature should be ignored. The trace file tends to be very large, even when very few investors in any one template are traced.
17. *Trace Wealthiest Number*. An investor, whether or not selected by the "trace fraction" feature, will be traced if the investor is ranked by initial wealth within the wealthiest N investors; the user here specifies the value of N. A value of -1 indicates that this feature should be ignored.

The Trader Templates Worksheet in DA Mode

Trader	TT0	TT1	LiqTrad
Buy Alpha	0	0	0
Buy Alpha Increment	0	0	0
Buy Beta	0.98	1	1
Buy Beta Increment	0.01	0.01	0.01
Buy Review, Wait First	0.2	0.2	0.2
Buy Review, Wait Then	0.3	0.3	0.3
Buy Review, Wait Last	5	3	4000
Buy Max Changes	6	10	1000
Sell Alpha	0	0	0
Sell Alpha Increment	0	0	0
Sell Beta	1.02	1	0.99
Sell Beta Increment	-0.01	-0.01	-0.01
Sell Review, Wait First	0.2	0.2	0.2
Sell Review, Wait Then	0.3	0.3	0.3
Sell Review, Wait Last	5	3	4000
Sell Max Changes	6	10	1000
Recent	10	10	10
Max Bid Rule	2	2	2
Max Bid Parameter	5	5	100
Min Offer Rule	2	2	2
Min Offer Parameter	5	5	90

Figure 9: The Trader Templates Worksheet in DA Mode.

In DA mode, the trader templates worksheet, shown in Figure 9, controls the number of trader templates and parameters relating to each. The attributes of a trader template are parameters used in the trading strategy of its traders. The last trader template, labeled LiqTrad in the worksheet, contains the trading strategy parameters for the liquidation trader that liquidates investor accounts if liquidation is necessary in order to meet maintenance margin requirements. The table in the worksheet presents the following parameters, used by each template's traders in placing or revising bids and offers.

1. *Buy Alpha*. An initial bid is placed at alpha plus beta times the current price. The entry in this field defines the value of alpha for the corresponding trader template.
2. *Buy Alpha Increment*. If a purchase is not completed in a timely fashion, then orders are reviewed and the buy alpha is increased by the increment defined by the entry in this field.
3. *Buy Beta*. An initial bid is placed at alpha plus beta times the current price. The entry in this field defines the value of beta for the corresponding trader template.
4. *Buy Beta Increment*. If a purchase is not completed in a timely fashion, then orders are reviewed and the buy beta is increased by the increment defined by the entry in this field.
5. *Buy Review, Wait First*. Entries in this and the following two fields define the length of time traders from a given template wait before revising a bid. The entry "wait first" indicates the first wait, in days.
6. *Buy Review, Wait Then*. The entry in this field indicates the length of each subsequent wait, other than the last wait.
7. *Buy Review, Wait Last*. The entry in this field indicates how long to wait after the last change of bid before canceling the order.
8. *Buy Max Changes*. The entry in this field is the maximum number of times that bids are to be revised.
9. *Sell Alpha*. An initial offer is placed at alpha plus beta times the current price. The entry in this field defines the value of alpha for the corresponding trader template.
10. *Sell Alpha Increment*. If a sale is not completed in a timely fashion, then orders are reviewed and the sell alpha increment defined by the entry in this field is added to the sell alpha.
11. *Sell Beta*. An initial offer is placed at alpha plus beta times the current price. The entry in this field defines the value of beta for the corresponding trader template.
12. *Sell Beta Increment*. If a sale is not completed in a timely fashion, then orders are reviewed and the sell beta increment defined by the entry in this field is added to the sell beta.
13. *Sell Review, Wait First*. Entries in this and the following two fields define the length of time that traders from a given template wait before revising an offer. The entry in the "wait first" field indicates the first wait, in days.
14. *Sell Review, Wait Then*. The entry in this field indicates the length of each subsequent wait, other than the last wait.
15. *Sell Review, Wait Last*. The entry in this field indicates how long to wait after the last change of offer before canceling the order.
16. *Sell Max Changes*. The entry in this field is the maximum number of times that offers are to be revised.
17. *Recent*. The entries in this and the following four rows contain parameters for bid/offer rules that override the preceding ones. Traders from each template follow the bid or offer strategy described in the entries above until this strategy conflicts with the rules described in the entries below, in which case the latter rules are adopted. The rules below ensure that traders will not bid or offer at a price that is too far from recent prices. The entry in the recent field indicates how many days are to be considered as recent.
18. *Max Bid Rule*. The entry in this field is a code specifying a maximum bid rule. The permitted values of the code are:
 - 0, meaning the trader cannot bid more than the maximum recent price plus P percent,
 - 1, meaning the trader cannot bid more than the average recent price plus P times recent standard deviations, and
 - 2, meaning the trader cannot bid more than the average recent price plus P percent.
19. *Max Bid Parameter*. The entry in this field presents, for each trader template, the P parameter used in the maximum bid rule. The interpretation of this parameter is specified by the rule.
20. *Min Offer Rule*. The entry in this field is a code specifying the minimum offer rule. Possible values of this code are:
 - 0, meaning the trader cannot offer less than the minimum recent price minus P percent,
 - 1, meaning the trader cannot offer less than the average recent price minus P standard deviations, and
 - 2, meaning the trader cannot offer less than the average recent price minus P percent.
21. *Min Offer Parameter*. The entry in this field presents, for each trader template, the P parameter used in the minimum offer rule. The interpretation of this parameter is specified by the rule.

The Capital Market Equilibrium (CME) Mode

In the CME mode, JLMSim attempts to find equilibrium implied returns for securities based on the composition of the market portfolio and on the preferences of market participants.

The principal difference between the DA mode and the CME mode has to do with the formation of expected returns. In DA mode, statisticians form expected returns (as instructed by the JLMSim user), based on either historical average returns or fixed estimated expected returns per share. In CME mode, the statistician adjusts expected returns daily in an attempt to make the market weights of various stocks equal target market weights designated by the user. The amount of adjustment depends on parameters set by the JLMSim user. In this section, we describe specific inputs used for the CME mode.

To switch to CME mode, the user selects this mode from the drop-down box in the Mode of Operation panel. File Control operates in exactly the same manner as in the DA mode: all simulation information is represented in worksheets labeled "Basic," "Financing," "Factors," "Securities," "Statisticians," "Analysts," "Investors," and "Traders." CME mode contains an additional worksheet labeled

Covariance Matrix. As before, the user can navigate from worksheet to worksheet by clicking on the appropriately labeled tabs or can pull down the Edit menu and select the appropriate item. The following subsections describe the information in each worksheet.

The Basic Worksheet in CME Mode

The basic worksheet in CME mode is shown in Figure 10. In particular, this shows the appearance of the worksheet when the sample UI Form file `sample_cme.txt` is loaded into the Input GUI. The differences between this worksheet in the CME mode and its equivalent in the DA mode, shown in Figure 2, are the absence of the Data Source drop-down box and the inclusion of the CME Parameters field.

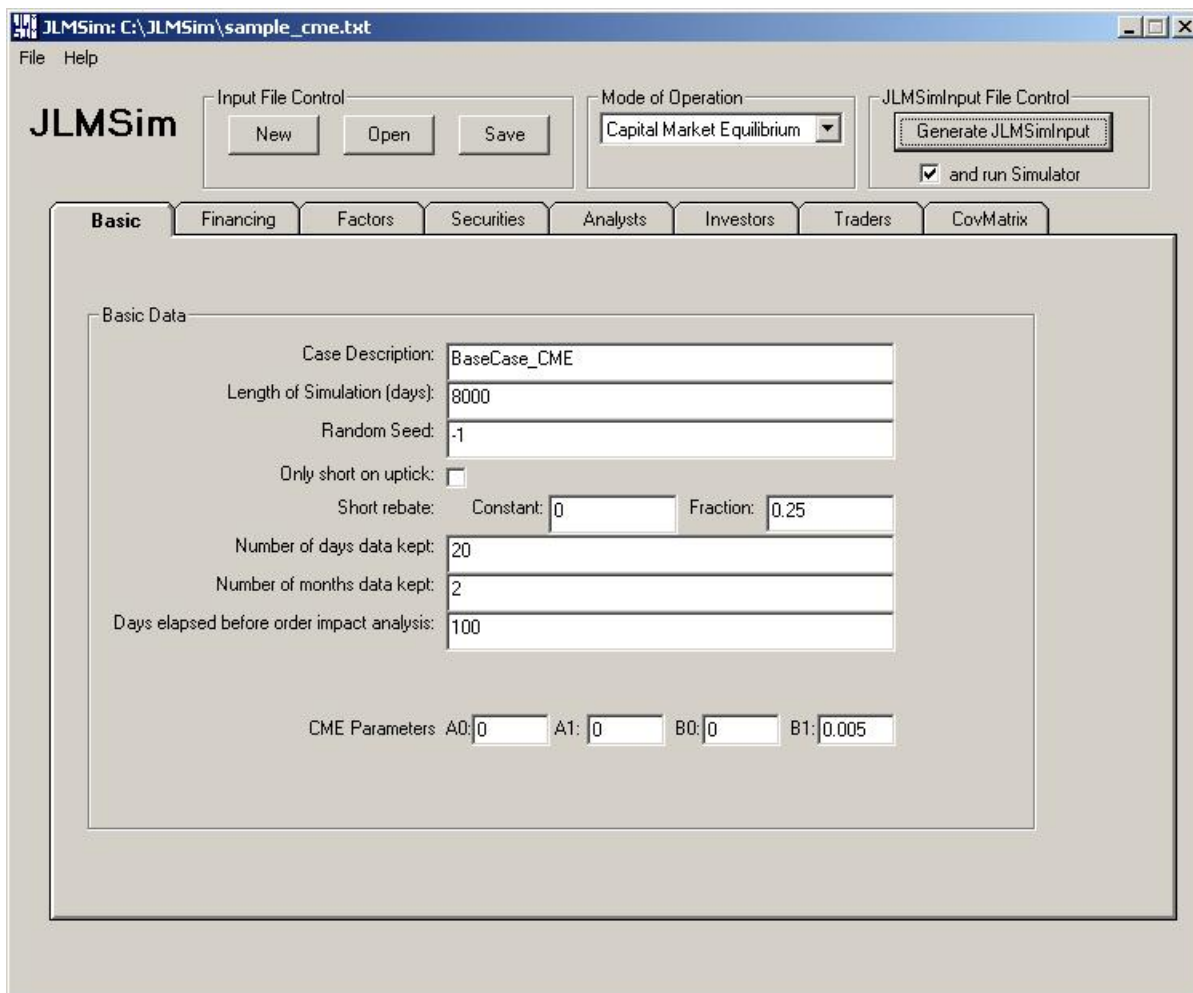


Figure 10: The Basic Worksheet in CME Mode.

The CME parameters are the four expected return estimate adjustment parameters A_0 , A_1 , B_0 , and B_1 . These are used as follows. For the i -th security (or asset class), let $\text{diff}(i)$ be the difference between the actual and target fraction of the security in the market portfolio. If $|\text{diff}(i)|$ does not exceed

$$A_0 + A_1 * \text{target}(i)$$

where $\text{target}(i)$ is the fraction of the security in the target market portfolio, then no action is taken. If $\text{diff}(i)$ is positive (negative) then

$$B_0 + B_1 * |\text{diff}(i)|$$

is subtracted from (added to) the expected return estimate for the security.

The Securities Worksheet in CME Mode

The securities worksheet in the CME mode, shown in Figure 11, differs from its appearance in the DA mode, shown in Figure 5, by the addition of two new columns—the Eq Pct column and the Init Ret column.

The entries in the Eq Pct column determine the equilibrium weights of each security in the target market portfolio. If the simulation allows for the presence of cash, then the difference between 100 percent and the sum of the entries in this column represents the equilibrium amount of cash. If cash is not present, the entries in this column should sum to 100 percent. The entries in the Init Ret column determine the initial estimates for security returns.

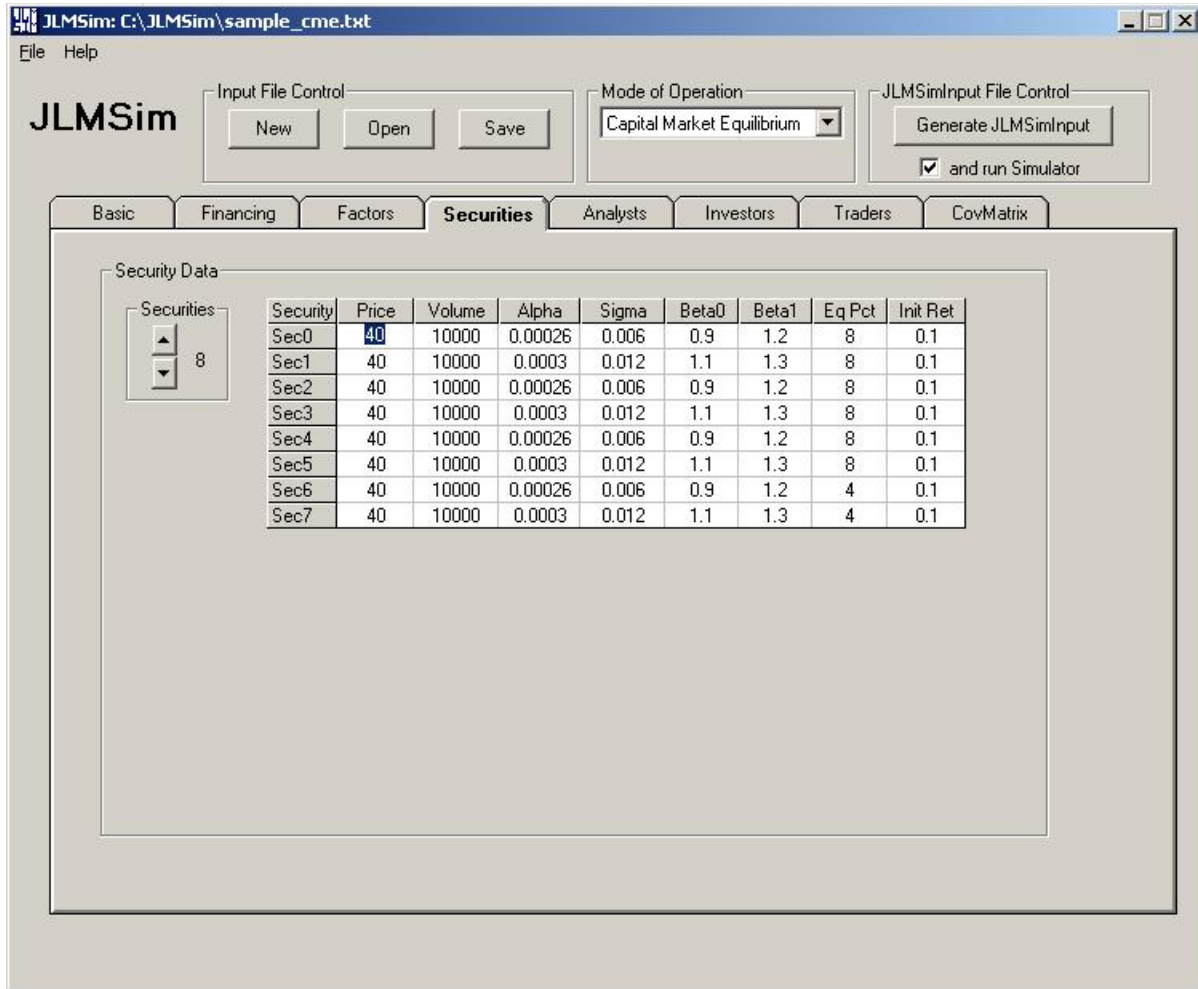


Figure 11: The Securities Worksheet in CME Mode.

The Statisticians Worksheet in CME Mode

In the CME mode, estimates are formed using the parameters described above, and the user has no further control over statisticians' parameters. Therefore, in this mode, the statisticians worksheet is disabled.

The Portfolio Analysts Worksheet in CME Mode

In the CME mode, the portfolio analysts worksheet, shown in Figure 12, differs from its appearance in DA mode, shown in Figure 7. In particular, the Statistician Used column is omitted.

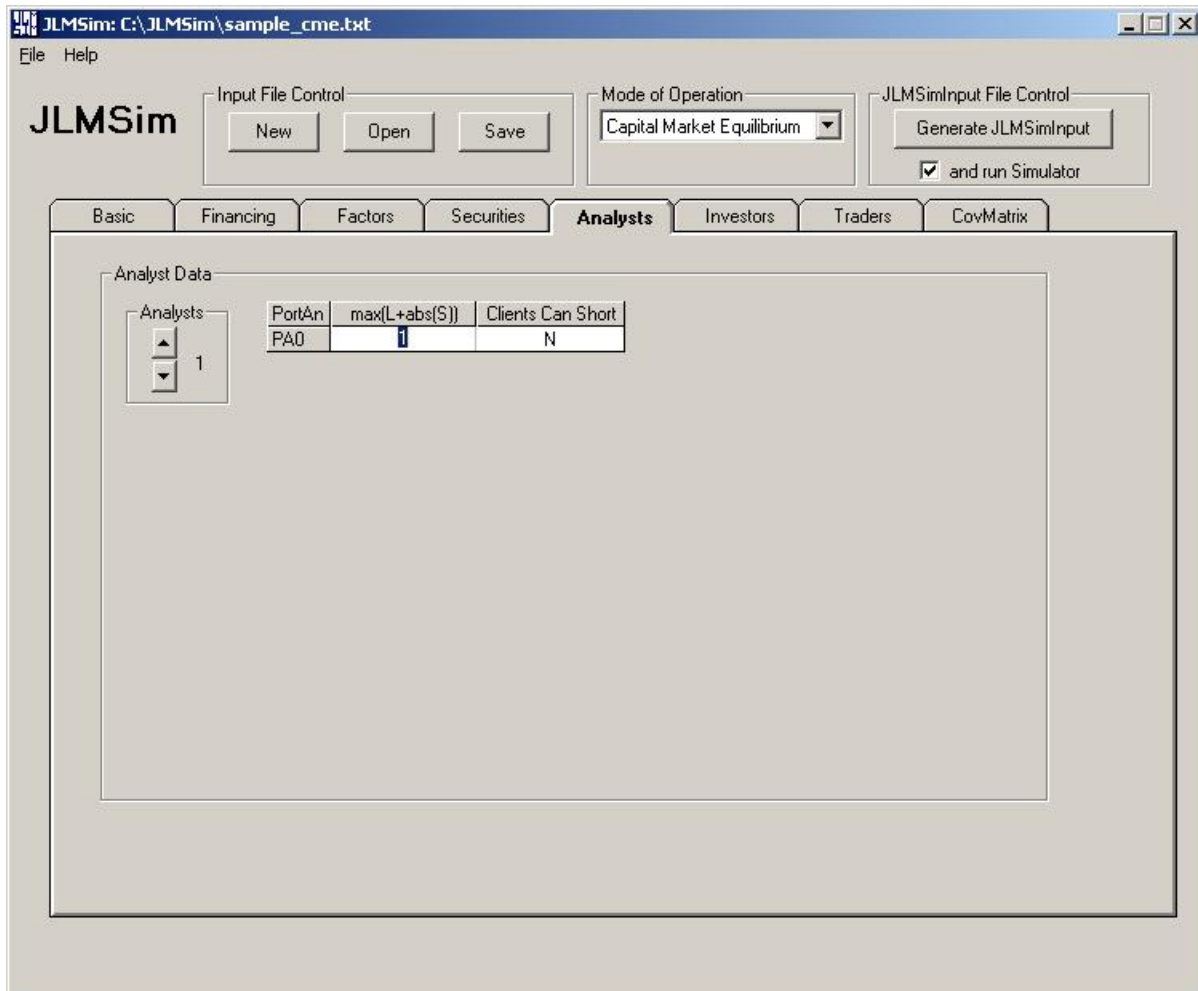


Figure 12: The Portfolio Analysts Worksheet in CME Mode.

The Investor Templates Worksheet in CME Mode

The investor templates worksheet in CME mode, shown in Figure 13, differs from its appearance in DA Mode, shown in Figure 8. The rows relating to random cash flows to or from the investors are omitted.

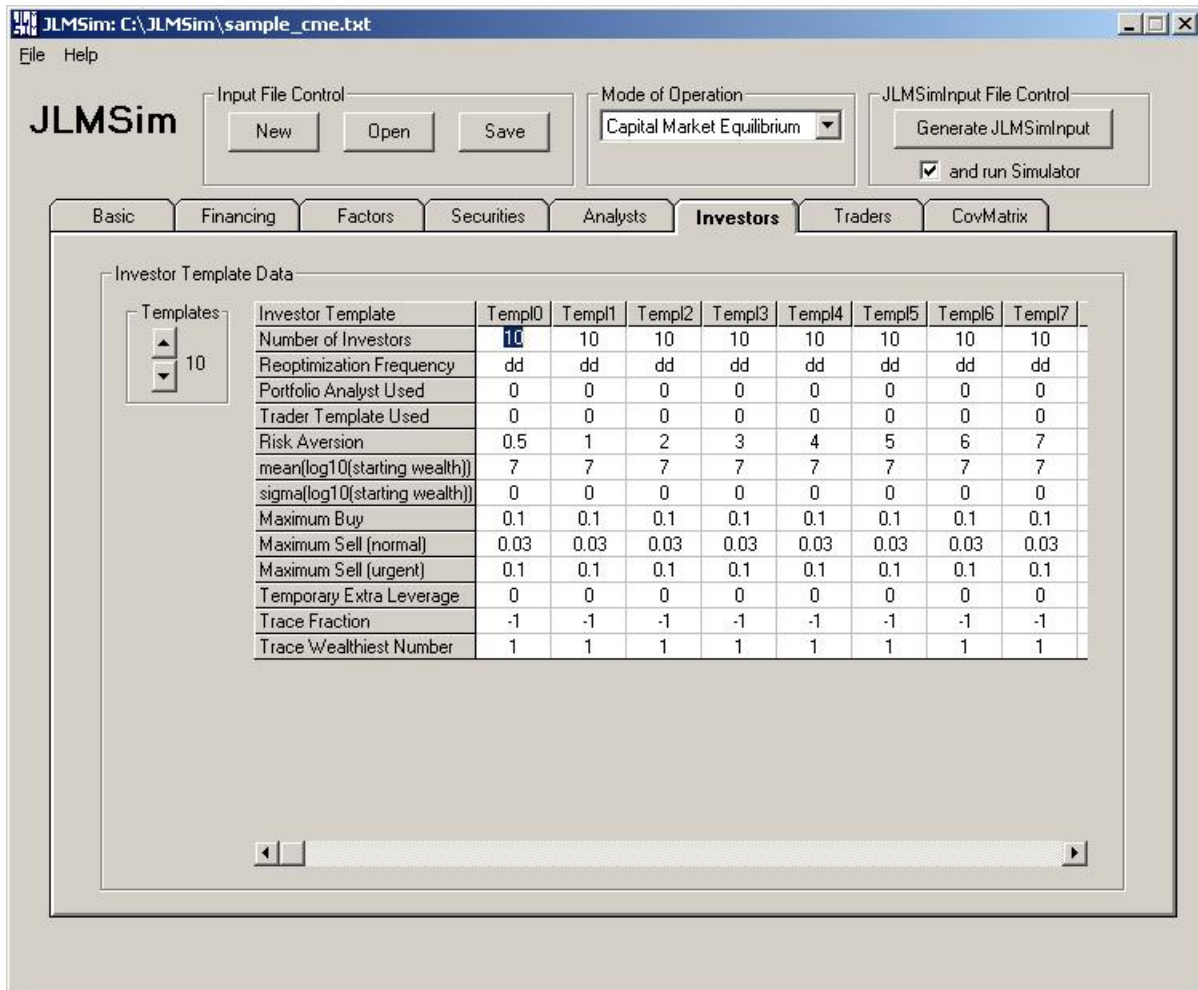


Figure 13: The Investor Templates Worksheet in CME Mode.

The Trader Templates Worksheet in CME Mode

In the CME mode, no liquidation trader is present. The trader templates worksheet, shown in Figure 14, thus differs from its appearance in the DA Mode (shown in Figure 9). The Liq Trad column is omitted.

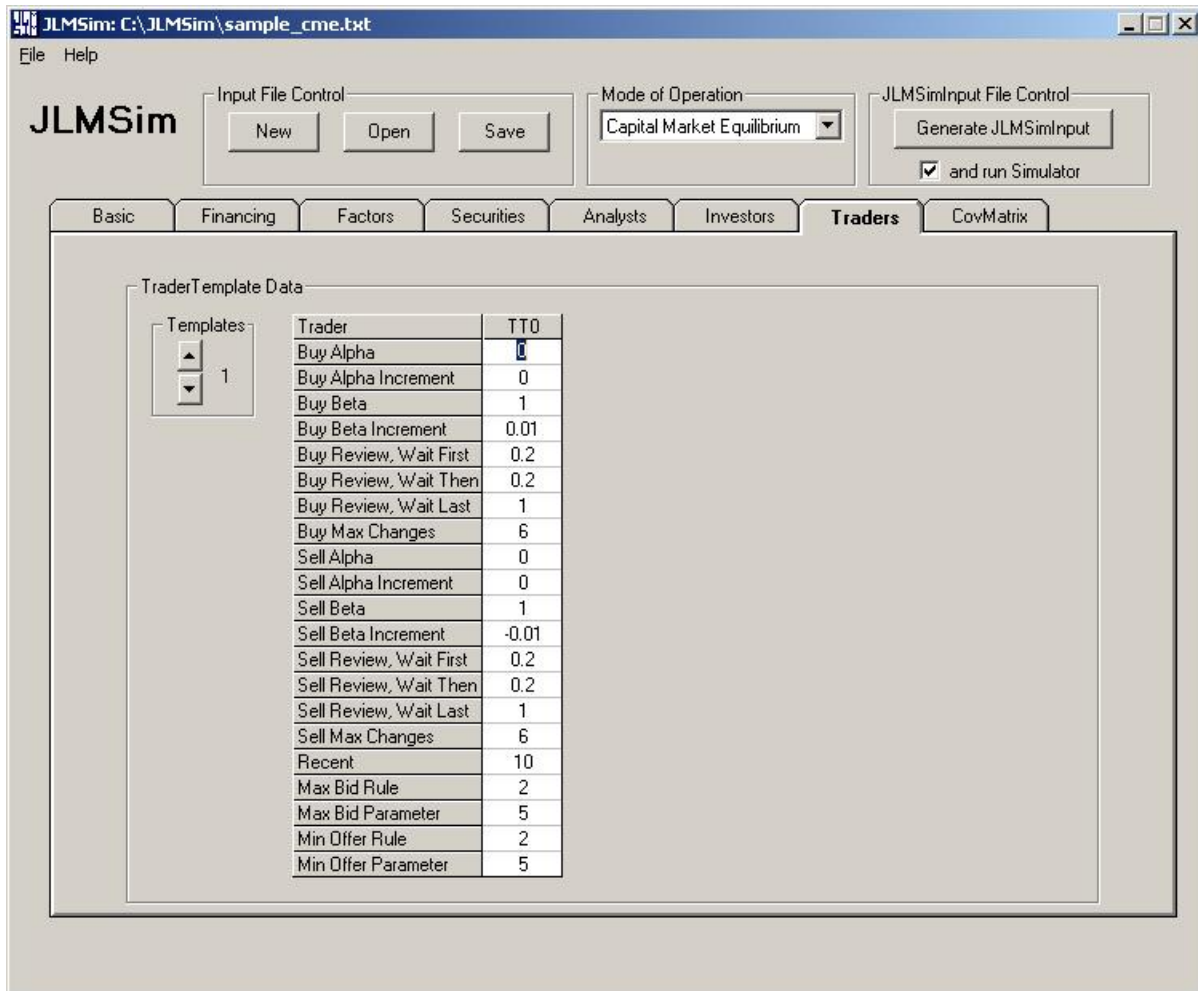


Figure 14: The Trader Template Worksheet in CME Mode.

The Covariance Matrix Worksheet

In the CME mode, JLMSim needs an estimate of the covariance matrix of security returns. This matrix can be entered into the covariance matrix worksheet of the Input GUI, shown in Figure 15. Four different approaches may be used.

1. *DF*: In this approach, JLMSim uses a covariance matrix generated from the daily factor model, as described in the factors worksheet.
2. *AC_UH*: In this mode, the upper half of the (symmetrical) annualized covariance matrix is entered, and the Input GUI fills in the lower half of the matrix.
3. *AC_LH*: In this mode, the lower half of the (symmetrical) annualized covariance matrix is entered, and the Input GUI fills in the upper half of the matrix.
4. *AC_BH*: In this mode, the entire annualized covariance matrix is entered.

The covariance matrix input approach is selected from the Covariance Matrix Model drop-down box in the Common Covariance Matrix Data panel.

Figure 15 shows the covariance matrix worksheet when the DF approach has been selected. If one of the other three covariance models is chosen, the table in the figure contains as many rows and columns as there are securities defined in the securities worksheet. The entry in row $R-i$ and column $C-j$ represents the annualized covariance between the returns of securities i and j as defined in the securities worksheet.

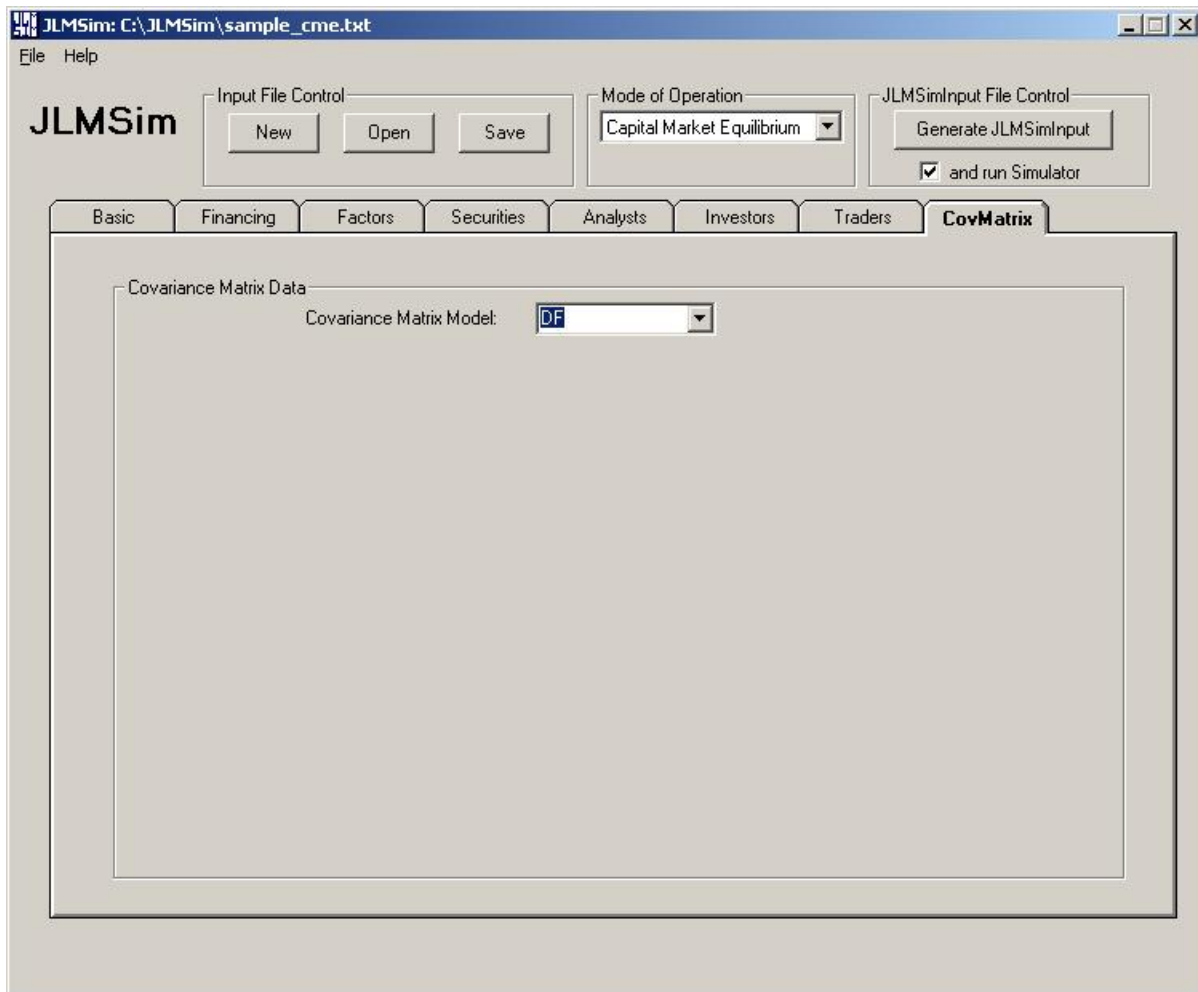


Figure 15: The Covariance Matrix Worksheet using the DF Approach.