1 Overview of the system

The simulation process is divided into the four steps (Fig. 1):

1. To create two input data files. “data.txt” is a data file for economic fundamentals which is calculated by EXCEL. “rate.txt” is a data file for weekly rates.

2. To conduct simulation. The main simulation program is “new_ age6”. Its source code is “new_ agedasi6.pas” written by PASCAL. It is translated to C code by the command “p2c” and compiled by “cc” or “gcc”.Its output files are simulation data files (rout.dat.*), market averages of factors’ weights (wout.dat.*), supply and demand files (qout.dat.*), and rate change frequency files (fout.dat.*).

3. To transform the output data. There are three data transformation programs written by PASCAL.

   (a) A program “freq” counts frequency of simulated rates in some rate ranges every week. Its input file is rout.dat.*, its initialization file is freq.ini, and its output file is freq.dat.

   (b) A program “stat” calculates mean and standard deviation of simulated rates included in the file “statname.dat”. Its input file is rout.dat.*, its initialization file is stat.ini and statname.dat, and its output file is stat.dat.

   (c) A program “weight” calculated mean and standard deviation of all simulated weights. Its input file is wout.dat.*, its initialization file is weight.ini and statname.dat, and its output file is weight.dat.

4. To plot the transformed data. After data transformation, we can get graphs of the results by GNUPLOT.

These steps can be conducted in an interface program “agedasi.tcl” which is written by TCL/TK. Now it is in construction.

2 To create two input data files.

2.1 data.txt

“Data.dat” is a data file of normalized data about economic indexes from DataStream. It is used as an input file of the main simulation program “new_ age6”. It is calculated by Excel as follows:

1. To download raw data from DataStream. Now 40 data are downloaded (Table 1).

2. To change the raw data to weekly data.

3. To calculate diffrences of weekly data.

4. To divide the differentiations by means of absolute values.

Fig. 2 is an example of creation of “data.txt”.

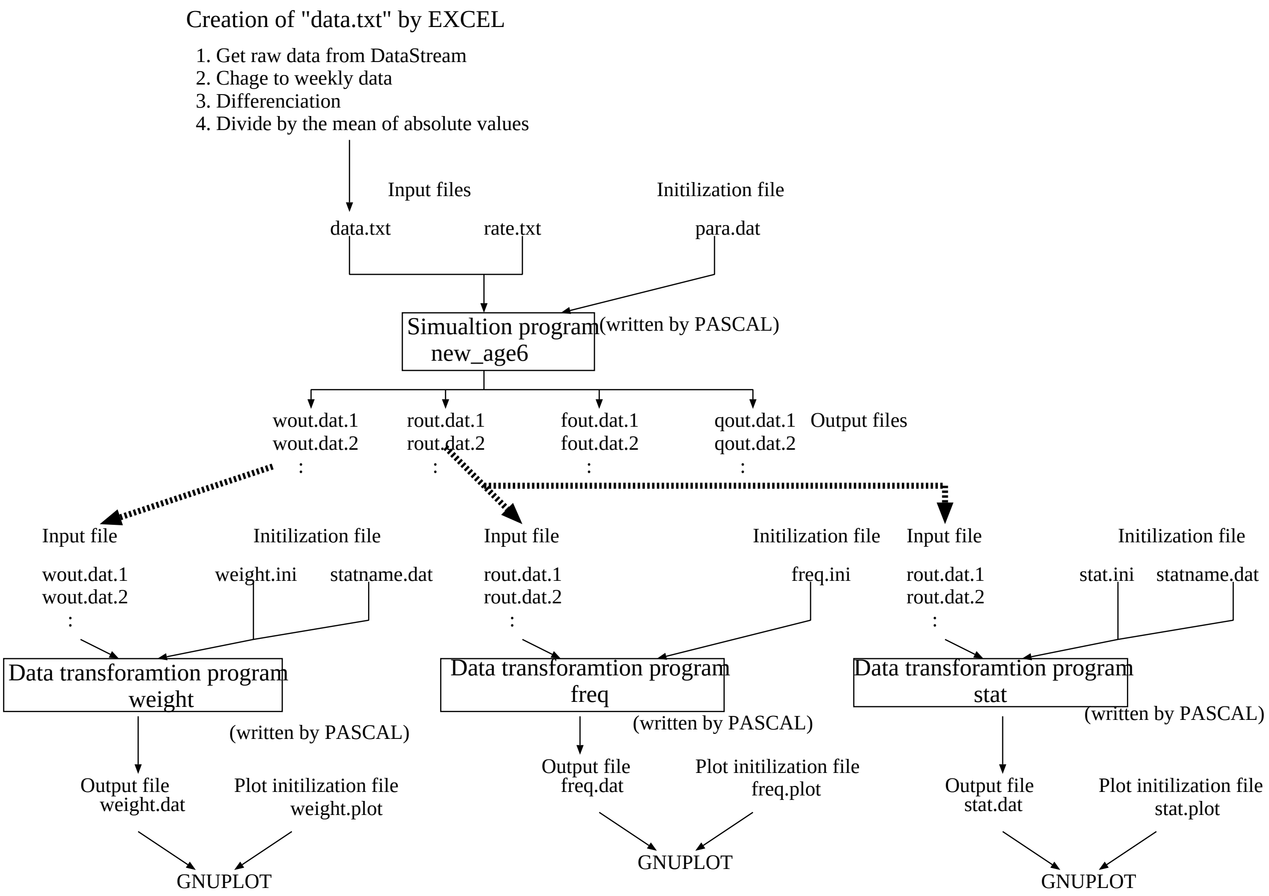

Figure 1: Overview

DataSterm Code	Contents
JPI99B.CB	JP GROSS DOMESTIC PRODUCT CURA
USI99B.CB	US GROSS DOMESTIC PRODUCT CURA
JPCURBALA	JP CURRENT BALANCE$TERMS (DISCONTINUED) SEE JPCURYENA CURN
JPCURBALB	JP CURRENT BALANCE S/A$TERMS (DISCONTINUED) SEE JPCURYENB CURA
USCPXFSEE	”US CONSUMER PRICE INDEX-ALL ITEMS EXCEPT FOOD,SHELTER & ENERGY”
USCPXFDEE	US CONSUMER PRICE INDEX - ALL ITEMS LESS FOOD AND ENERGY SADJ
JPCPGNRLF	JP CONSUMER PRICE INDEX - GENERAL
JPHOUSSTF	JP HOUSING STARTS NADJ
USUNRATEE	US TOTAL UNEMPLOYMENT RATE SADJ
USUNNAGCE	US UNEMPLOYED WAGE & SALARY WRKRS-NONAGRICULTURAL PRIV INDS.
JPOCUNEPE	JP UNEMPLOYMENT SADJ
JPI66..CE	JP INDUSTRIAL PRODUCTION INDEX - MINING AND MANUFACTURING VOLA
USI66..CE	US INDUSTRIAL PRODUCTION VOLA
USPURCHS	US NATIONAL ASSOCIATION OF PURCHASING MANAGERS’ INDEX SADJ
USOPERATE	”US CAPACITY UTILISATION RATE, ALL INDUSTRY SADJ”
USCAPMANE	”US CAPACITY UTILISATION RATE, MANUFACTURING SADJ”
USOCFBALB	US TRADE BALANCE (F.O.B. - F.O.B.) CURA
USRTGENMB	US RETAIL SALES OF GENERAL MERCHANDISE STORES CURA
JPOCRSALG	JP RETAIL SALES:VOLUME VOLA
USOCRSALG	US RETAIL SALES:VOLUME VOLA
USFINGPRE	US PRODUCER PRICE INDEX - FINISHED GOODS SADJ
USPPIGFFE	US PRODUCER PRICE INDEX-FINISHED GOODS LESS FOODS & ENERGY SADJ
JPOCPRODF	JP PRODUCER PRICES(MANUFACTURED GOODS) - TOTAL NADJ
USCURACBB	US CURRENT ACCOUNT BALANCE CURA
JPOCIOPCE	JP CAPACITY UTILISATION (MANUFACTURING) SADJ
JAPDOWA	NIKKEI 225 STOCK AVERAGE - PRICE INDEX
DJCMP65	DOW JONES COMPOSITE 65 STOCK AVERAGE - PRICE INDEX
JPDISCR	JAPAN DISCOUNT RATE - MIDDLE RATE
ICJPY10	”JAPAN (JPY) IR SWAP,10Y - MIDDLE RATE”
JAPLONG	JAPAN LONG TERM (GOVT.BONDS) - RED. YIELD
JPBIL3M	JAPAN BILL DISCOUNT THREE MONTH - MIDDLE RATE
JAP3MBL	JAPAN BILLS 3 MONTH - MIDDLE RATE
USDISCR	US DISCOUNT RATE - MIDDLE RATE
USFEDFD	US FEDERAL FUNDS - MIDDLE RATE
USTBL3M	US-TREASURY BILL 3 MONTH - MIDDLE RATE
USBD10Y	US TREASURY -10 YEAR BENCHMARK BOND - RED. YIELD
USM2WNA	US MONEY SUPPLY M2 NOT SEASON/ADJ. - ES
USM2WSA	US MONEY SUPPLY M2 SEASONALLY ADJ. - ES
USM3WNA	US MONEY SUPPLY M3 NOT SEASON/ADJ. - ES
USM3WSA	US MONEY SUPPLY M3 SEASONALLY ADJ. - ES

Table 1: the 40 raw data from DataStream

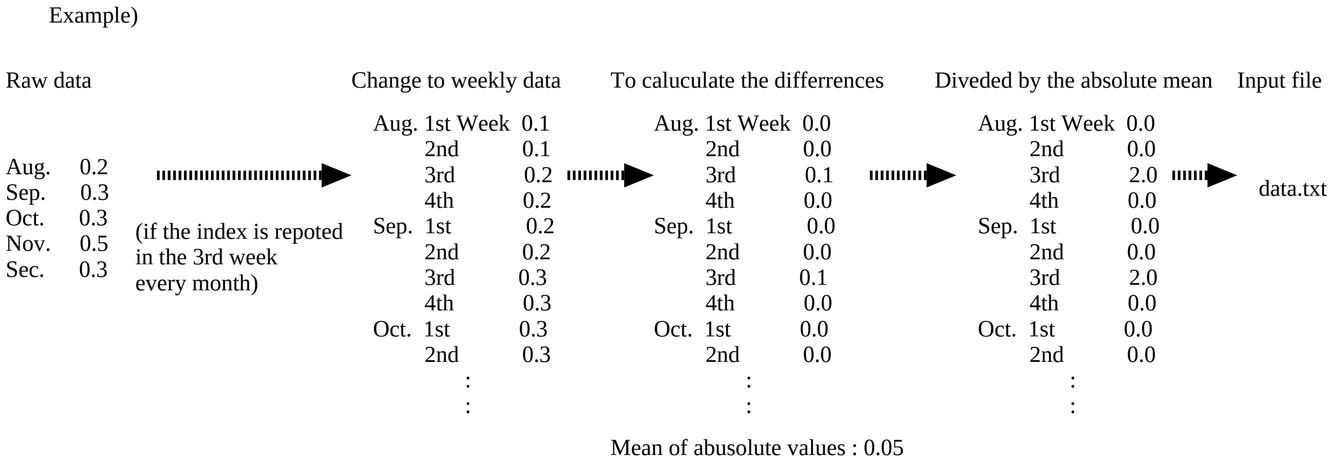

Figure 2: Example of creation of “data.txt”.

2.2 rate.txt

“Rate.txt” is a weekly data file about closing yen-dollar rate of Tokyo FX market every Friday. They are downloaded from DataStream. DataSteam code is usjapyn. The start date of “rate.txt” must be the same as that of “data.txt”.

```
108.55
109.1
110.15
110.6
```

3 To conduct simulation

The main simulation program (Fig. 3) is “new_ age6”. Its source code is “new_ agedasi6.pas” written by PASCAL. It is translated to C code by the command “p2c” and compiled by “cc” or “gcc”.Its output files are simulation data files (rout.dat.*), market averages of factors’ weights (wout.dat.*), supply and demand files (qout.dat.*), and rate change frequency files (fout.dat.*).

3.1 Initialization file: para.dat

See table 2.

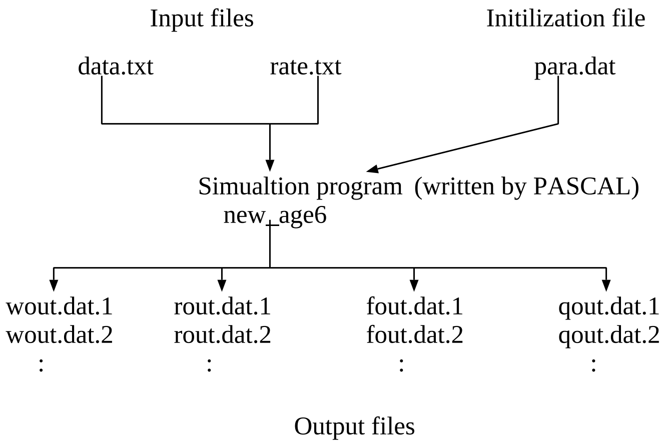

Figure 3: Simulation program

`.1`	Minimal interval of simulated rate value (Fixed)
`50`	Number of intervals (Fixed)
`1`	Mean value of Trend Factor 1 (Fixed)
`2`	Mean value of Trend Factor 2 (Fixed)
`1`	Mean value of Trend Factor 2 (Fixed)
`30`	Number of dealers (Changeable)
`58`	Start week of training period (Changeable)
`216`	End week of forecast period (Changeable)
`164`	Length of training period (Changeable)
`.3`	Probability of crossover in training period (Fixed)
`.003`	Probability of mutation in training period (Fixed)
`.5`	Percentage of selection in training period (Fixed)
`.0`	Probability of crossover in forecast period (Fixed)
`.0`	Probability of mutation in forecast period (Fixed)
`.0`	Percentage of selection in forecast period (Fixed)
`50`	Number of simulation times (Changeable)
`25`	Number of training times (Changeable)
`write`	Type of data outputs (Fixed)
`3`	Maximum size of alle (Fixed)

Table 2: para.dat

3.2 Output Files

3.2.1 result/rout.dat.*

See table 3.

Number of week	Simulated rate	Actual rate	Simulated rate change	Actual rate change	Trading volume
52	99.60000	99.60000	-0.55000	-0.55000	0
53	101.36000	101.36000	1.76000	1.76000	0
54	98.55000	98.55000	-2.81000	-2.81000	0
55	99.45000	99.45000	0.90000	0.90000	0
56	99.36000	99.36000	-0.09000	-0.09000	0

Table 3: rout.dat.*

3.2.2 result/wout.dat.*

See table 4.

Number	Market average of	Market std of	Market average	Market std of	···
of week	factor 1's weights	factor 1's weights	of factor 2's weights	factor 2's weights	
52	-2.4526	0.0000	-1.3621	0.0000	···
53	-2.4526	0.0000	-1.3621	0.0000	···

Table 4: wout.dat.*

4 To transform the output data

To transform the output data. There are three data transformation programs written by PASCAL.

4.1 freq

A program “freq” (fig. 4) counts frequency of simulated rates in some rate ranges every week. Its input file is rout.dat.*, its initialization file is freq.ini, and its output file is freq.dat.

4.1.1 Initialization file: freq.ini

See table 5.

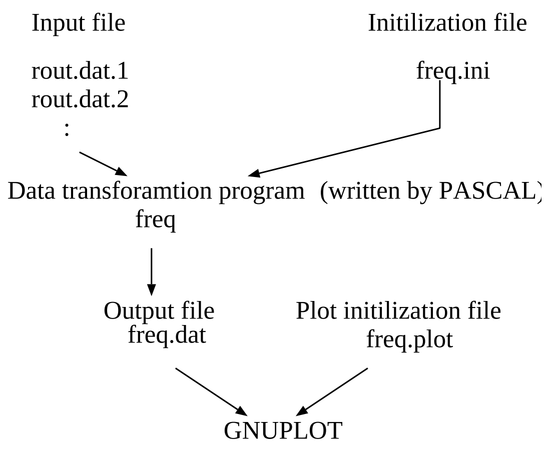

Figure 4: Data transformation program: freq

`95`	Minimum value of range (Fixed)
`145`	Maximum value of range (Fixed)
`25`	Number of frequency intervals (Fixed)
`1`	Start number of simulation file (Changeable)
`50`	Number of simulation time (Changeable)
`50`	Maximum size of frequency (Fixed)
`result/`	Name of the directory which has rout.dat.* (Changeable)

Table 5: freq.ini

4.1.2 Input file

{ dir }/rout.dat.{ start .. start + ntime -1 }

4.2 Output file: freq.dat

See the following table.

Week Rate range Numbers of simulation paths in the rate range

```
187      96.0000     0
187      98.0000     0
187     100.0000     0
187     102.0000     1     3
187     104.0000     0
187     106.0000     0
187     108.0000     2     27     28
187     110.0000     1     24
187     112.0000     2     7      39
187     114.0000     3     22     25     48
187     116.0000     1     17
187     118.0000     2     8      31
187     120.0000     5     9      14     16     40     49
187     122.0000     6     1      15     21     42     43     44
187     124.0000     3     13     29     32
187     126.0000     3     11     36     46
187     128.0000     3     12     26     33
187     130.0000     1     30
187     132.0000     5     18     19     35     45     50
187     134.0000     4     10     20     37     41
187     136.0000     4     6      23     38     47
187     138.0000     2     4      5
187     140.0000     1     34
187     142.0000     1     2
187     144.0000     0
```

4.3 stat

A program "stat" (fig. 5) calculates mean and standard deviation of simulated rates included in the file "statname.dat". Its input file is rout.dat.*, its initialization file is stat.ini and statname.dat, and its output file is stat.dat.

4.3.1 Initialization file: stat.ini

See table 6.

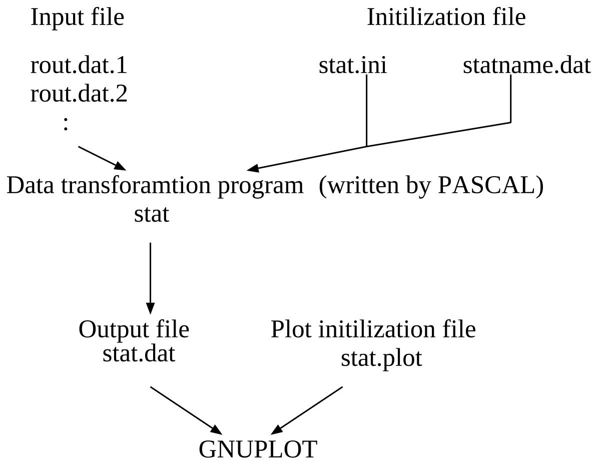

Figure 5: Stat

`result/`	Name of the directory which has rout.dat.* (Changeable)

Table 6: stat.ini

4.3.2 Initialization file: statname.dat

See table 7.

`50`	Number of all simulation paths (Changeable)
`1 2 3 4 5 ··· 47 48 49 50`	Numbers of simulation paths (Changeable)

Table 7: statname.dat

4.3.3 Input files

{ `dir` }/rout.dat.{ *numbers in* `statname.dat` }

4.3.4 Output file: stat.dat

See the following table.

```
# Number of path 43
# Paths  17 22 31 1 40 8 14 21 47 4 5 9 37 25 43 33 12
# week  volume  mean    real    std    error
  162   0.000 122.930 122.930  0.000  -0.000
  163   0.000 124.330 124.330  0.000  -0.000
```

4.4 weight

A program "weight" (Fig. 6) calculated mean and standard deviation of all simulated weights. Its input file is wout.dat.*, its initialization file is weight.ini and statname.dat, and its output file is weight.dat.

4.4.1 Initialization program: weight.ini

See table 8.

`43`	Number of factors (Changeable)
`result/`	Name of the directory which has wout.dat.* (Changeable)

Table 8: weight.ini

4.4.2 Initialization program: statname.dat

See table 9.

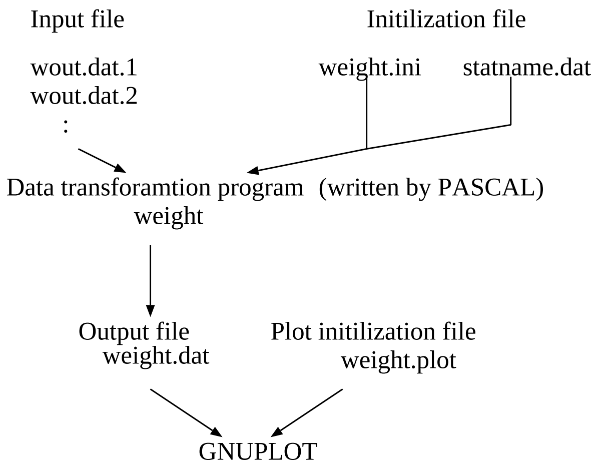

Figure 6: Data transformation program: weight

50	Number of all simulation paths (Changeable)
1 2 3 4 5 ··· 47 48 49 50	Numbers of simulation paths (Changeable)

Table 9: statname.dat

4.4.3 Input files

{ dir }/wout.dat.{ *numbers in* statname.dat }

4.4.4 Output file: weight.dat

See the following table.

```
# Number of path 50
# Paths  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 3
# week     Factor 1  Factor 2  Factor 3  Factor 4   Factor 5  Factor 6  Factor 7  Factor 8   Factor
   36      0.61662   0.48532   0.30691   -0.06922   0.37964   0.25921   0.40272   -0.08178   0.52868
   37      0.60838   0.50171   0.34075   -0.03746   0.34256   0.25274   0.38834   -0.05066   0.51063
```

5 To plot the transformed data

After data transformation, we can get graphs of the results by GNUPLOT. It needs plot nitialization file.

5.1 Plot initialization file: stat.plot

```
set yrange [95:140]
set xtics ("1996/1/5" 105,"1996/2/2"
109,"1996/3/1" 113,"1996/4/5" 118,"1996/5/3" 122,"1996/6/7"
127,"1996/7/5" 131,"1996/8/2" 135,"1996/9/6" 140,"1996/10/4"
144,"1996/11/1" 148,"1996/12/6" 153,"1997/1/3" 157,"1997/2/7"
162,"1997/3/7" 166,"1997/4/4" 170,"1997/5/2" 174,"1997/6/6"
179,"1997/7/4" 183,"1997/8/1" 187,"1997/9/5" 192,"1997/10/3"
196,"1997/11/7" 201,"1997/12/5" 205,"1998/1/2" 209,"1998/2/6"
214,"1998/3/6" 218)
set grid
set nokey
set xrange [164:216]
plot "stat.dat" u 1:3:5 w e,"stat.dat" u 1:4 w l
pause -1
```

Plot command is “gnuplot stat.plot”.

6 Interface Program by TCL/TK (in Construction)

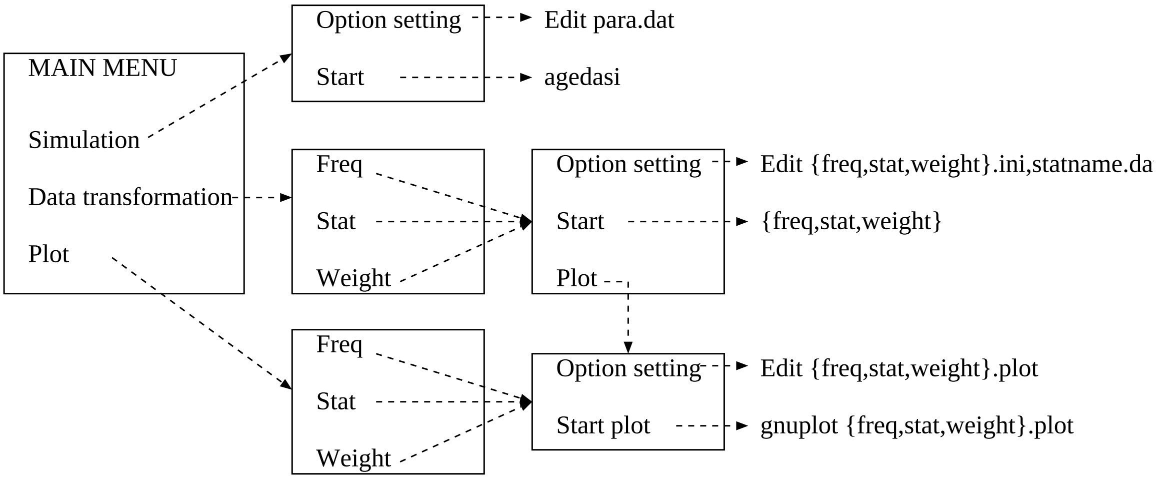

Figure 7: Interface Program by TCL/TK