

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” calculates 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 differences of weekly data.
4. To divide the differentiations by means of absolute values.

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

Creation of "data.txt" by EXCEL

1. Get raw data from DataStream
2. Change to weekly data
3. Differentiation
4. Divide by the mean of absolute values

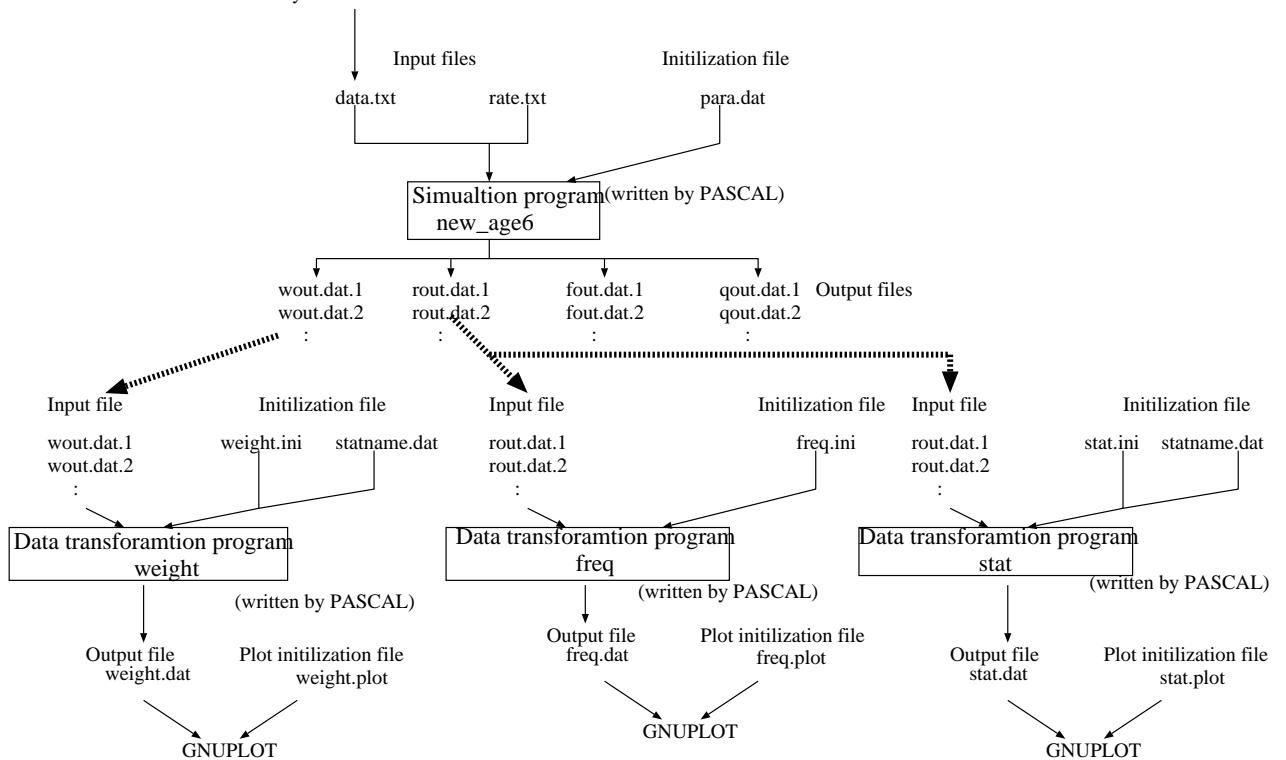


Figure 1: Overview

DataTerm 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
USOPERAET	"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
USRRTGENMB	US RETAIL SALES OF GENERAL MERCHANDISE STORES CURA
JPOCRSALG	JP RETAIL SALES:VOLUME VOLA
USO CRSALG	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

Example)

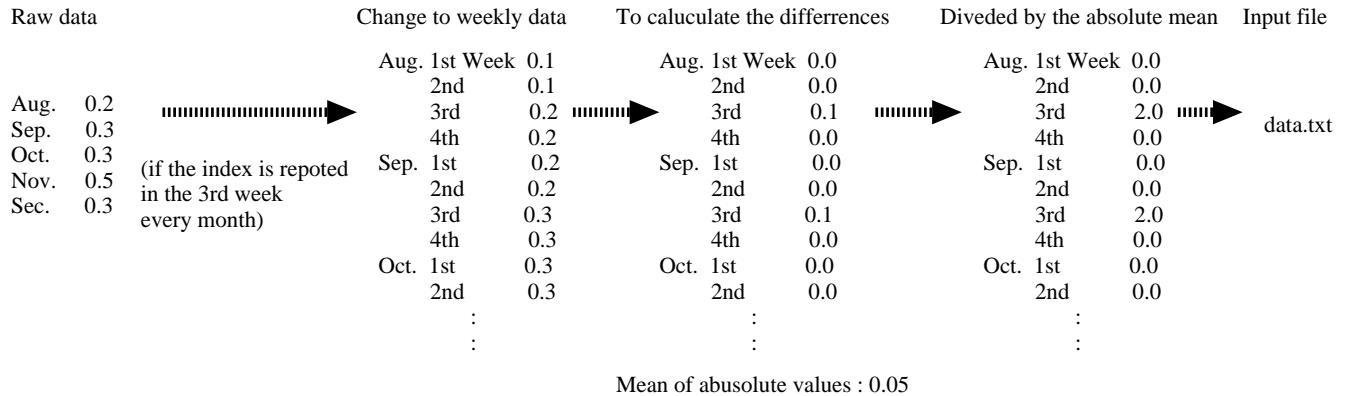


Figure 2: Example of creation of “data.txt”.

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_agedasi6”. 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.

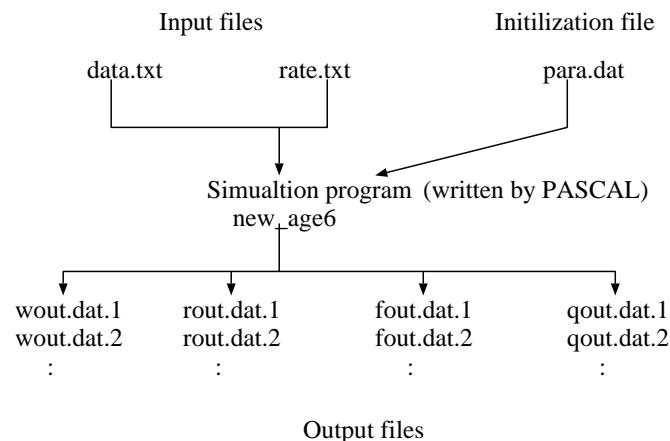


Figure 3: Simulation program

.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)
50	Percentage of selection in forecast period (Fixed)
25	Number of simulation 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 of week	Market average of factor 1's weights	Market std of factor 1's weights	Market average of factor 2's weights	Market std of 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.

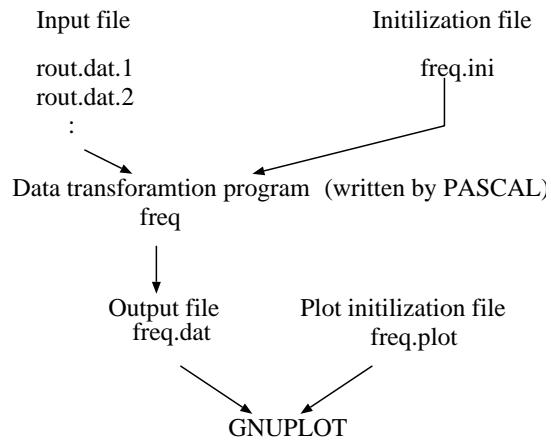


Figure 4: Data transformation program: freq

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.

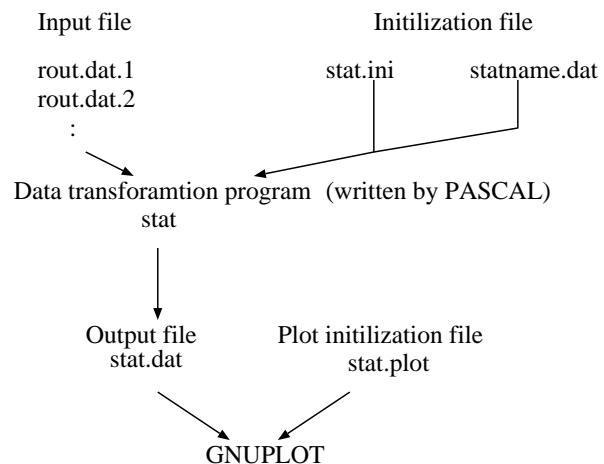


Figure 5: Stat

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.

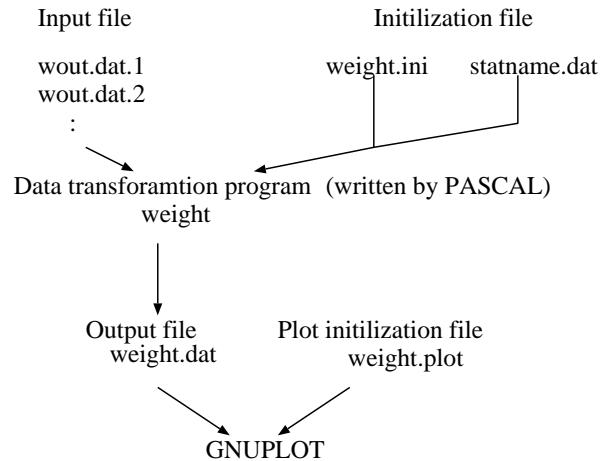


Figure 6: Data transformation program: weight

50 1 2 3 4 5 ... 47 48 49 50	Number of all simulation paths (Changeable) 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	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270	271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378	379	380	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399	400	401	402	403	404	405	406	407	408	409	410	411	412	413	414	415	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431	432	433	434	435	436	437	438	439	440	441	442	443	444	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459	460	461	462	463	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479	480	481	482	483	484	485	486	487	488	489	490	491	492	493	494	495	496	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511	512	513	514	515	516	517	518	519	520	521	522	523	524	525	526	527	528	529	530	531	532	533	534	535	536	537	538	539	540	541	542	543	544	545	546	547	548	549	550	551	552	553	554	555	556	557	558	559	560	561	562	563	564	565	566	567	568	569	570	571	572	573	574	575	576	577	578	579	580	581	582	583	584	585	586	587	588	589	590	591	592	593	594	595	596	597	598	599	600	601	602	603	604	605	606	607	608	609	610	611	612	613	614	615	616	617	618	619	620	621	622	623	624	625	626	627	628	629	630	631	632	633	634	635	636	637	638	639	640	641	642	643	644	645	646	647	648	649	650	651	652	653	654	655	656	657	658	659	660	661	662	663	664	665	666	667	668	669	670	671	672	673	674	675	676	677	678	679	680	681	682	683	684	685	686	687	688	689	690	691	692	693	694	695	696	697	698	699	700	701	702	703	704	705	706	707	708	709	710	711	712	713	714	715	716	717	718	719	720	721	722	723	724	725	726	727	728	729	730	731	732	733	734	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	756	757	758	759	760	761	762	763	764	765	766	767	768	769	770	771	772	773	774	775	776	777	778	779	780	781	782	783	784	785	786	787	788	789	790	791	792	793	794	795	796	797	798	799	800	801	802	803	804	805	806	807	808	809	810	811	812	813	814	815	816	817	818	819	820	821	822	823	824	825	826	827	828	829	830	831	832	833	834	835	836	837	838	839	840	841	842	843	844	845	846	847	848	849	850	851	852	853	854	855	856	857	858	859	860	861	862	863	864	865	866	867	868	869	870	871	872	873	874	875	876	877	878	879	880	881	882	883	884	885	886	887	888	889	890	891	892	893	894	895	896	897	898	899	900	901	902	903	904	905	906	907	908	909	910	911	912	913	914	915	916	917	918	919	920	921	922	923	924	925	926	927	928	929	930	931	932	933	934	935	936	937	938	939	940	941	942	943	944	945	946	947	948	949	950	951	952	953	954	955	956	957	958	959	960	961	962	963	964	965	966	967	968	969	970	971	972	973	974	975	976	977	978	979	980	981	982	983	984	985	986	987	988	989	990	991	992	993	994	995	996	997	998	999	1000
------------------	----	---------	---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	------

5 To plot the transformed data

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

5.1 Plot initialization file: stat.plot

```
set yrang [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)

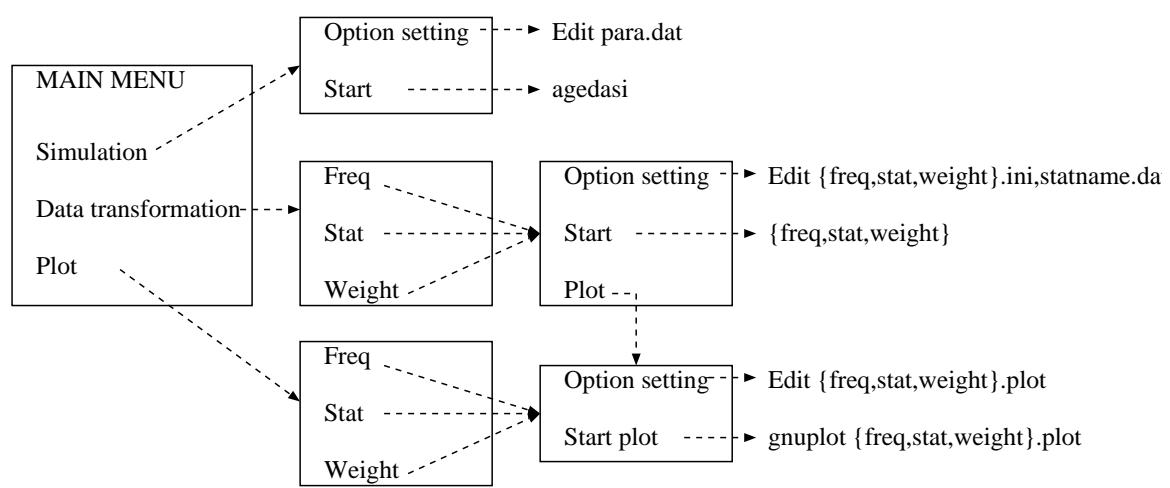


Figure 7: Interface Program by TCL/TK