User's Manual of "*TAgen_auto.py*"

Version 1.1

(27 Aug, 2025)

Table of contents

1. Overview

■ Outline:

- The "***TAgen_auto.py***" is a simulation tool for generating a "***TA***" (averaged atomic time scale) with daily prediction and dynamical weighting. This ***TA*** is based on the same principle as the "***TAp***" explained in *[Ref. 1]*.
- The "***TAgen_auto.py***" is almost the same as "***TAgen_pred.py***", except it adopts a dynamical weighting that is inversely proportional to the Allan variance for each clock. These Allan variances are updated with every calculation.
- In "***TAgen_pred.py***", the weights are fixed so that the automatic optimization function (a major advantage of ***TA***) is ineffective. In contrast, "***TAgen_auto.py***" optimizes ***TA*** using dynamical weighting.

■ Aim:

- Let's learn the effectiveness of the weighting depending on the Allan variance updated in each calculation.

※ This program is only a tutorial example of the ***TA*** algorithm, and there is room for improvement depending on the situation for actual utility.

※ Another program "***TAgen_basic.py***" is recommended to learn the principle of ***TA***. And "***TAgen_pred_py***" is better for learning the process of making "***TA* with prediction**".

■ Technical comments:

- The program requires the measurement "***UTC(k) – clock$_i$***" data files, the parameter tables "***clock_file_list.txt***" and "***param_list.txt***" located under the designated directory and folders.
- The frequency stabilities of the clock data and the calculated ***TA*** are evaluated using the Allan Deviation (*ADEV*), which is calculated by the free Python library "***AllanTools***" *[Ref. 2]*.

■ Reference:

[1] Y. Hanado, "*Time scale algorithms ~basics to applications~ Lecture*", BIPM Summer School, September 11, 2025

[2] Web site of *"AllanTools" (https://allantools.readthedocs.io/en/latest/)*

2. Setting of environment

[1] Install *"Python" in your computer*

- Set the path to the working directory.
- The following Python modules should be installed in advance;

  - *numpy*
  - *matplotlib*
  - *allantools*

For example, from the command prompt;

```
C:\Mywork>python -m pip install numpy
```

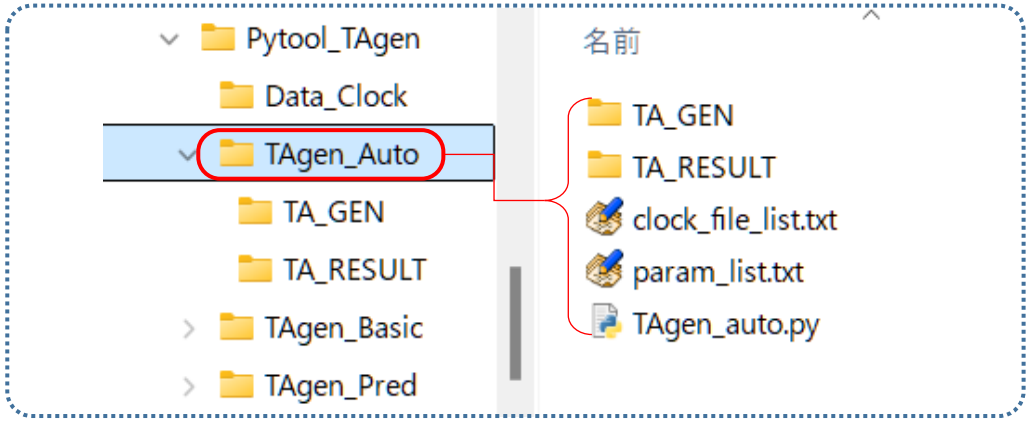
※ Although *"Spyder"* is not necessary for running the tools, it is useful for debugging and modifying them.

https://www.spyder-ide.org/

[2] Construct directories :

Make ***"/Pytool_TAgen"***,
"/Pytool_TAgen/TAgen_Auto",
"/Pytool_TAgen/TAgen_Auto/TA_GEN/", *and*
"/Pytool_TAgen/TAgen_Auto/ TA_RESULT/".

- The directory name "***Pytool_TAgen***" can be changed as you like, but **don't change the names of the directories under *"/Pytool_TAgen/"*.**



3. Preparation for starting the program

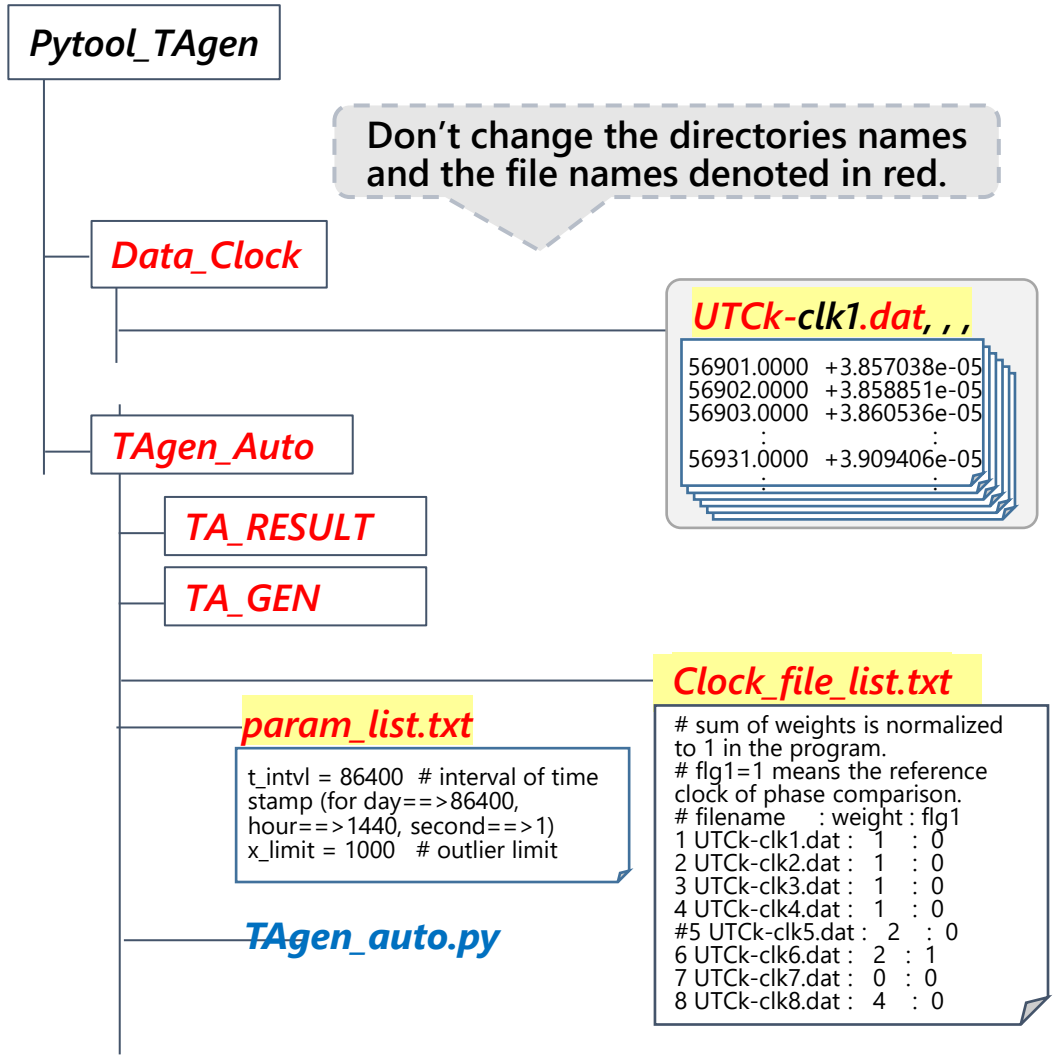
TAgen_auto.py

[1] Copy the Python program : "*TAgen_auto.py*"

[2] Copy or make the necessary files :

- Clock measurement data :
  "*/Pytool_TAgen/Data_Clock/UTCk-(clock$_i$).dat*" *(p.30)*
  - The "*UTCk-(clock$_i$).dat*" are measurement clock data used for the TAp calculation.
- Clock-data name list file :
  "/Pytool_TAgen/TAgen_Auto/clock_file_list.txt" *(p.28)*
  - "*clock_file_list.txt*" provides the conditions of each clock in the averaging process.
- Parameter list file :
  "/Pytool_TAgen/TAgen_Auto/param_list.txt" *(p.29)*
  - Parameters that are fixed within the program but can be changed on demand are provided in "*param_list.txt*".

※ Details of each file are provided in *Appendix A-1.2*.



4. Operation of each function

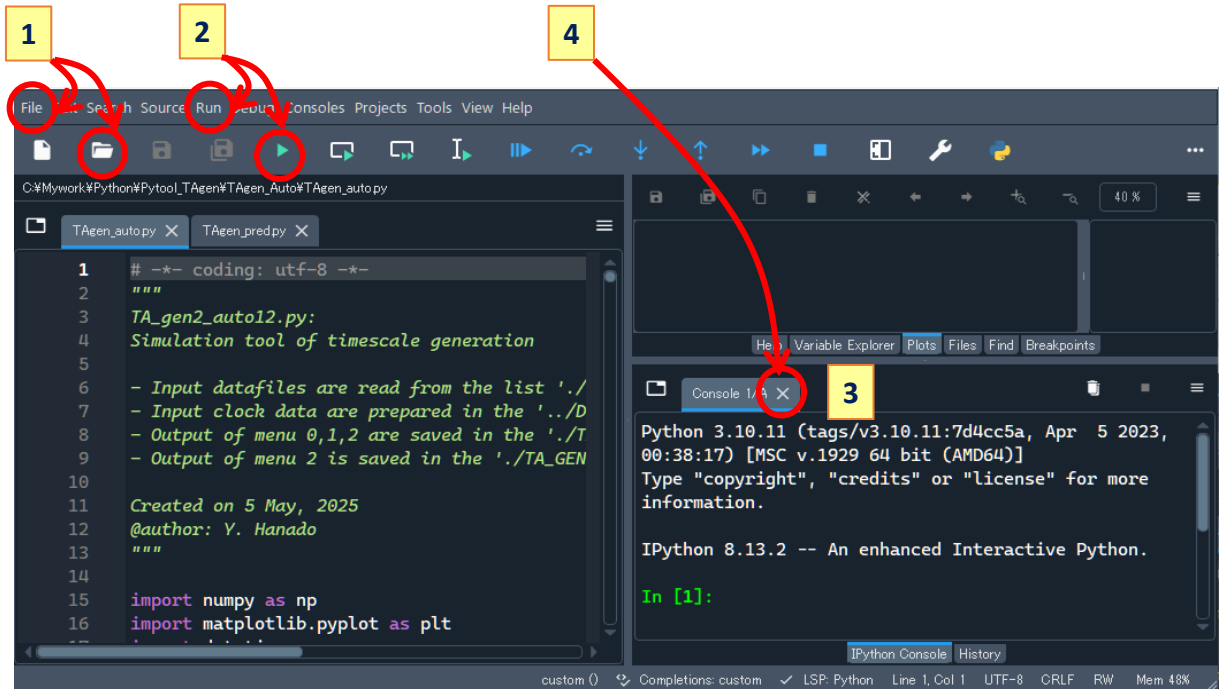
4.1. Run "*/TAgen_Auto/TAgen_auto.py*"

■ Case (1) : via "*Spyder*"

1. Select the following Python program from the menu of "***File --> open -> /Pytool_TAgen/TAgen_Auto/TAgen_auto.py***". 1
2. Click "▶" or "***Run***". 2

※ The result is shown in the "***Console***" window. 3

※ You can exit the program by clicking "***X***" on the 4 console.

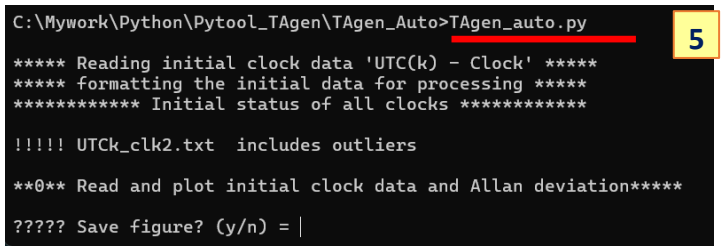


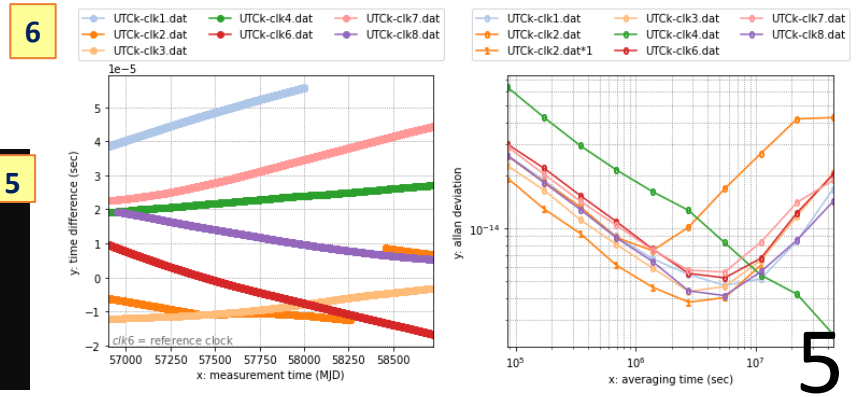
■ Case (2) : via command prompt

1. Go to the directory including "***TAgen_auto.py***" in advance.
2. Run "***TAgen_auto.py***" --> You can proceed with the program in the command prompt console. 5

※ The graph is shown in another window. 6

※ You can exit the program by "***Ctrl+Z***".





4. Operation of each function

4.1. Run "*/TAgen_Pred/TAgen_auto.py*"

■ Initial processing

[1] Read the parameters from "*param_list.txt*" *(p.29)*

- Parameters that are fixed within the program but can be changed on demand are provided from this parameter list.

[2] Read the parameter from "*clock_file_list.txt*" *(p.28)*

- This parameter list provides the conditions of each clock in the averaging process.
- The clocks expected to be used in the ***TAp*** calculation are assigned here with some weights.

[3] Read the assigned clock data *"UTCk-(clock$_i$).dat"* *(p.30)*

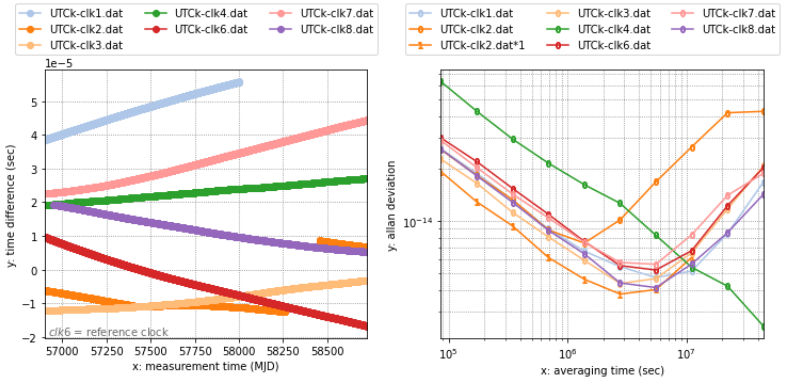
- Clocks with "***weight=0***" are not ignored in this initial process but excluded in the averaging process of "*menu-2*". 1
- Alert of outlier : outlier limit is set in "***param_list.txt***".

[4] Fix the available MJD range

- The available data range ***[MJD1, MJD2]*** is set as wide as possible.
- The MJD with enough number of clocks can be available.
- In all assigned clock data files, the available oldest date is set to be ***MJD1*** and the latest date is set to be ***MJD2***.

```
Python 3.10.11 (tags/v3.10.11:7d4cc5a, Apr  5 2023, 00:38:17)
[MSC v.1929 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 8.13.2 -- An enhanced Interactive Python.

runfile('C:/Mywork/Python/Pytool_TAgen/TAgen_Auto/TAgen_auto
.py', wdir='C:/Mywork/Python/Pytool_TAgen/TAgen_Auto')

..... Reading initial data of 'UTC(k) - Clock'
..... Formatting the initial data for processing

************ Initial status of all clocks ************

!!!!! UTCk-clk2.dat  includes outliers
```

1




```
**0** Read and plot initial clock data and Allan deviation*****

????? Save figure? (y/n) =
```

4. Operation of each function

4.1. Run "*/TAgen_Pred/TAgen_auto.py*"

■ Initial processing (Cont.)

[5] Combine the clock data within the assigned range

- The combined data file is "***all_raw_clk.dat***". *(p.35)*
- If data is missing, "***nan***" is recorded.

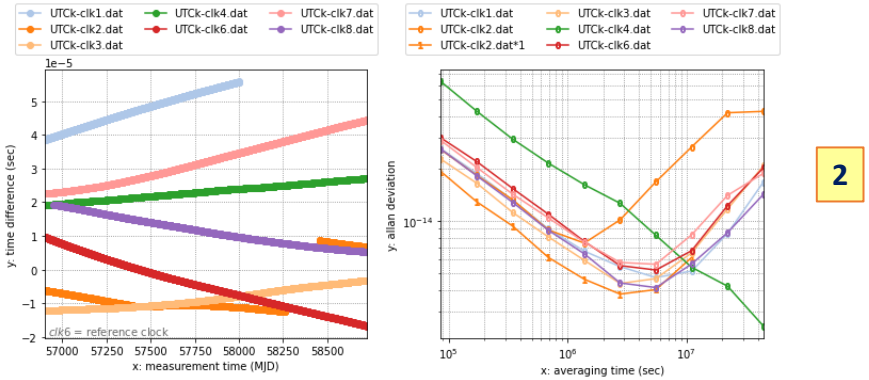
[6] Plot all the clock data of "***UTC(k) – clock$_i$***" *(p.40)* 2

- From the plots, clock conditions can be confirmed.
  - Left fig. : time difference of "***UTC(k) – clock$_i$***".
  - Right fig. : *ADEV* of "***UTC(k) – clock$_i$***".

※ *ADEV* is calculated by "***AllanTools***" *[Ref. 2]*, and saved in "***all_adev.dat***". *(p.37)*

[7] Save the results, if required

- If input "***y***" for "*Save figure (y/n) = "* , 3
  four results are saved under the following directory:
  "***./TA_RESULT/(time)/***" with a time stamp of data saving.
  - ***log_(time).txt*** : history of the process *(p.39)*
  - ***fig _(time).pdf*** : all figures *(p.40)*
  - ***all_raw_clk_(time).dat*** : "***UTC(k) – clock$_i$***" *(p.35)*
  - ***all_adev_(time).dat*** : *ADEV* of "***UTC(k) – clock$_i$***" *(p.37)*



2

0 Read and plot initial clock data and Allan deviation*****

????? Save figure? (y/n) = y 3

::::: save logfile = ./TA_RESULT/0818_104212/log_0818_104212.txt
::::: save figfile = ./TA_RESULT/0818_104212/fig_0818_104212.pdf
::::: save alldata = ./TA_RESULT/0818_104212/all_raw_clk_0818_104212.dat
::::: save ADEVdat = ./TA_RESULT/0818_104212/all_adev_0818_104212.dat

==========
Menu of the data processing
---
0. Plot the "UTC(k)-Clock" data
1. Assign the MJD range
2. Make TA with daily prediction
and dynamical weighting
10. Exit
==========
????? Input menu number ? >>

4. Operation of each function

TAgen_auto.py

4.1. Run "*/TAgen_Auto/TAgen_auto.py*"

Clock_file_list.txt

```
# sum of weights is normalized
to 1 in the program.
# flg1=1 means the reference
clock of phase comparison.
# filename    : weight : flg1
1 UTCk-clk1.dat :  1   : 0
2 UTCk-clk2.dat :  1   : 0
3 UTCk-clk3.dat :  0   : 0
4 UTCk-clk4.dat :  1   : 0
#5 UTCk-clk5.dat :  1   : 0
6 UTCk-clk6.dat :  1   : 1
7 UTCk-clk7.dat :  1   : 0
8 UTCk-clk8.dat :  1   : 0
```

UTCk-clk1.dat

56901.0000	+3.857038e-05
56902.0000	+3.858851e-05
:	:
57999.0000	+5.576701e-05
58000.0000	+5.578885e-05

all_raw_clk_0818_104212.dat

fname	UTCk-clk1.dat	UTCk-clk2.dat	UTCk-clk3.dat	UTCk-clk4.dat	UTCk-clk6.dat	UTCk-clk7.dat	UTCk-clk8.dat
Weight	1.0	1.0	1.0	1.0	2.0	0.0	4.0
flag	0	0	0	0	1	0	0
cid	1	2	3	4	6	7	8
56901	+3.857038e-05	-6.184638e-06	-1.215249e-05	+1.918367e-05	+9.635976e-06	+2.258480e-05	+nan
56902	+3.858851e-05	-6.197517e-06	-1.215115e-05	+1.919092e-05	+9.617261e-06	+2.258759e-05	+nan
:	:	:	:	:	:	:	:
58725	+nan	+6.693075e-06	-3.187783e-06	+2.707895e-05	-1.676407e-05	+4.440954e-05	+5.273111e-06
58726	+nan	+6.687379e-06	-3.180825e-06	+2.708426e-05	-1.677650e-05	+4.442301e-05	+5.270004e-06

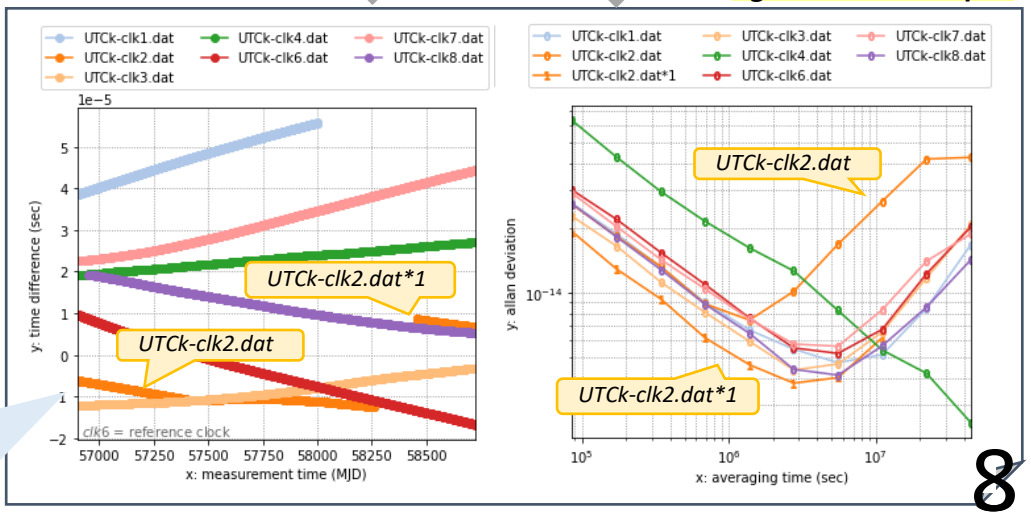
- The combined data of all "***UTC(k)-clock$_i$***" except *clk5*.
- The ***weight, flag*** and ***cid*** are the copies from "***Clock_flie_list.txt***".
- *ADEV*s are calculated from this data.

all_adev_0818_104212.dat

Tau(s)	UTCk-clk1.dat	UTCk-clk2.dat	UTCk-clk2.dat*1	UTCk-clk3.dat	UTCk-clk4.dat	UTCk-clk6.dat	UTCk-clk7.dat	UTCk-clk8.dat
MJDspan	**56901-58000**	**56901-58260**	**58457-58726**	**56901-58726**	**56901-58726**	**56901-58726**	**56901-58726**	**56952-58726**
86400	+2.585226e-14	+2.559433e-14	+1.914455e-14	+2.255109e-14	+6.289309e-14	+2.989441e-14	+2.887262e-14	+2.576366e-14
172800	+1.871863e-14	+1.824332e-14	+1.284166e-14	+1.643061e-14	+4.258263e-14	+2.186610e-14	+2.024180e-14	+1.811728e-14
345600	+1.314144e-14	+1.320590e-14	+9.326690e-15	+1.116636e-14	+2.944345e-14	+1.532961e-14	+1.422621e-14	+1.275807e-14
:	:	:	:	:	:	:	:	:
22118400	+8.453397e-15	+4.172218e-14	+nan	+1.168016e-14	+4.207961e-15	+1.222042e-14	+1.399682e-14	+8.546454e-15
44236800	+1.668442e-14	+4.249225e-14	+nan	+2.068302e-14	+2.484683e-15	+2.016610e-14	+1.872125e-14	+1.418509e-14

- ***MJDspan*** is the period used in the *ADEV* calculation.
- If there are not sufficient samples, *ADEV* cannot be calculated and is expressed as "***nan***".

fig_0818_104212.pdf



log_0818_104212.txt

```
# Data log at 2025.08.18_10h42m_12s
#
**0** Read and plot initial clock data and Allan deviation*****
::::: Input clocklist  : clock_file_list.txt
::::: Input clock file : ['UTCk-clk1.dat', 'UTCk-clk2.dat', 'UTCk-clk3.dat',
'UTCk-clk4.dat', 'UTCk-clk6.dat', 'UTCk-clk7.dat', 'UTCk-clk8.dat']
::::: data time interval (t_intval): 86400(sec)
::::: Outlier limit (x_limit))    : 1000.0
```

- Operated menu processes are serially added to the log file.

- "***UTCk-clk2.dat***" includes anomalies on *MJD 58261 ~58456*, and they are treated as missing data.
- The data are divided into two blocks before/after the anomalies.
- *ADEV*s are calculated for each block.

4. Operation of each function

TAgen_auto.py

4.2. Select the menu of data processing

■ **Input the menu number :** 1

0. Plot the "*UTC(k)* – *Clock$_i$*" data

- Read all the data assigned in "***clock_file_list.txt***" *(p.28)*, and plot the time differences vs "***UTC(k)***" and their *ADEV*s.
- **This operation initializes all the processing to date.**

1. Assign the MJD range

- The specified range will be used for all subsequent processing unless "*menu-0*" is selected.

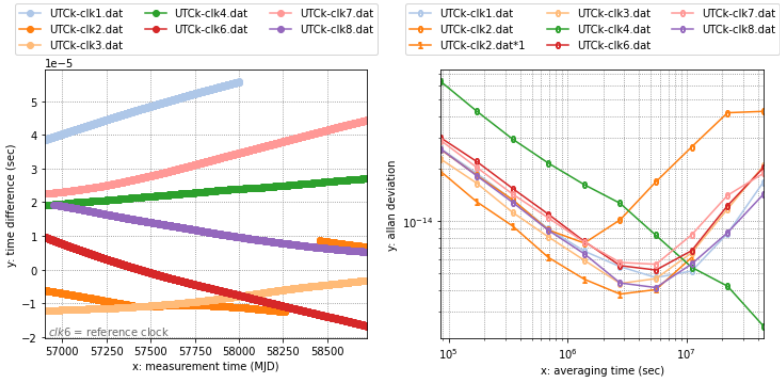
2. Make *TA* with daily prediction and dynamical weighting *1

- Calculate the ***TA*** using daily prediction and dynamical daily weighting ∝*1/Allan variance*.

10. Exit

- Exit the program.

1 : See "TAp**" in [Ref. 1].*



0 Read and plot initial clock data and Allan deviation*****

????? Save figure? (y/n) = y

::::: save logfile = ./TA_RESULT/0818_104212/log_0818_104212.txt
::::: save figfile = ./TA_RESULT/0818_104212/fig_0818_104212.pdf
::::: save alldata = ./TA_RESULT/0818_104212/all_raw_clk_0818_104212.dat
::::: save ADEVdat = ./TA_RESULT/0818_104212/all_adev_0818_104212.dat

==========
Menu of the data processing
---
0. Plot the "UTC(k)-Clock" data
1. Assign the MJD range
2. Make TA with daily prediction and dynamical weighting
10. Exit
==========
????? Input menu number ? >> 1

4. Operation of each function

TAgen_auto.py

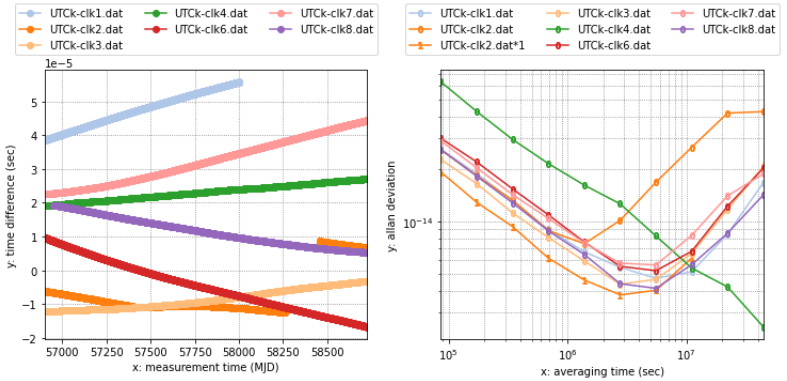
4.2.1. If "*menu-0*" is selected: 1

- **Plot the "*UTC(k) – Clock$_i$*" data**
  - Function is the same as for the "*Initial processing [1]-[7]*". *(pp.6-7)*
  - **Selecting this menu during operation will reset the results of the previous operation.**

```
=========================================
     Menu of the data processing
-----------------------------------------
  0.  Plot the "UTC(k)-Clock" data
  1.  Assign the MJD range
  2.  Make TA with daily prediction
          and dynamical weighting
 10.  Exit
=========================================
????? Input menu number ? >> 0
```

1

!!!!! UTCk-clk2.dat includes outliers




```
**0** Read and plot initial clock data and Allan deviation*****

????? Save figure? (y/n) =
```

4. Operation of each function

TAgen_auto.py

4.2.2. If "*menu-1*" is selected: 1

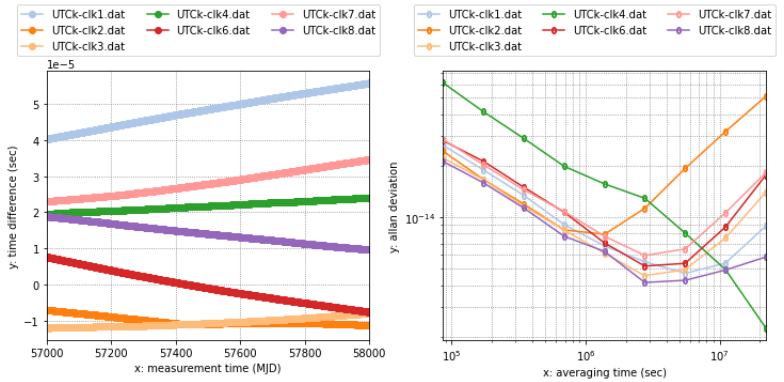
■ Assign the MJD range

- The data range is defined as [***MJD1, MJD2***].
- Enter "***MJD1***" and "***MJD2***" separated by a space. 2
- If the input values are outside the available range, re-entering is requested with an error message. 3
- If the input values are within the available range, the data processing range is updated. 4

※ **This new data range is adopted in any subsequent processes unless the initialization is done using "*menu-0*".**

- Save the results, if required. 5

```
=======================================
    Menu of the data processing
---------------------------------------
 0.  Plot the "UTC(k)-Clock" data
 1.  Assign the MJD range
 2.  Make TA with daily prediction
       and dynamical weighting
10.  Exit
=======================================
????? Input menu number ? >> 1          1
**1** Assign the MJD range for calculation in [56901,58726].
??1?? Input MJD1 and MJD2 with spacing = 56910 58800   2
**1** Selected range is : [ 56910  -  58800 ]
!!!!! MJD2 should be equal or smaller than  58726 . !!!!!   3
**1** Assign the MJD range for calculation in [56901,58726].
??1?? Input MJD1 and MJD2 with spacing = 57000 58000   4
 **1** Selected range is : [ 57000  -  58000 ]
```




```
**0** Read and plot initial clock data and Allan deviation*****
????? Save figure? (y/n) =          5
```

4. Operation of each function

4.2.3. If "*menu-2*" is selected: 1

■ Make TA with daily prediction and dynamical weighting

- This menu calculates the time series of "$x_i(t_k)$" day by day.

$$\hat{x}_i(t_k) = x_i(t_{k-1}) + \hat{y}_i(t_{k-1}) \cdot (t_k - t_{k-1}) \quad \text{Eq. (10)}\ ^{*1}$$

$$x_s(t_k) = \sum_i w_i(t_k)\{\hat{x}_i(t_k) - X_{is}(t_k)\} \quad \text{Eq. (9)}\ ^{*1}$$

$$x_i(t_k) = x_s(t_k) + X_{is}(t_k) \quad \text{Eq. (11)}\ ^{*1}$$

※ $x_i(t_k)$: time difference of ***clock***$_i$ vs TAp, is the value to obtain TAp
※ $x_s(t_k)$: the clock with a suffix "s" means the ***Refclk***.
※ t_{k-1} : one day before t_k
※ $X_{is}(t_k)$: measurement time difference of ***clocki*** vs ***Refclk***. (calculated from the "***UTC(k) – clock***$_i$" data)
※ $\hat{x}_i(t_k)$: prediction of $x_i(t_k)$
※ $\hat{y}_i(t_{k-1})$: rate of ***clock***$_i$ (calculated from the past values of x_i) *2
※ $w_i(t_k)$: weight of ***clock***$_i$ (calculated from the ADEV of past x_i) *3

```
=======================================
    Menu of the data processing
---------------------------------------
 0.  Plot the "UTC(k)-Clock" data
 1.  Assign the MJD range
 2.  Make TA with daily prediction
           and dynamical weighting
10.  Exit
=======================================
????? Input menu number ? >> 2
```

1

*1 : *Refer to [Ref. 1]*

*2 : $$\hat{y}_i(t_k) = \frac{x_i(t_k) - x_i(t_k - T_{rate})}{T_{rate}}$$ *Eq. (12)* *1

- T_{rate} *is given by the input value from the console. (p.16)*

*3 : $$w_i(t_k) = \frac{1}{\sigma_y^2(T_{wgt})} \qquad \sum_i w_i = 1$$

- $\sigma_y(T_{wgt})$ *is ADEV calculated from the past* x_i.
- T_{wgt} *is given by the input value from the console. (pp.16-17)*

4. Operation of each function

TAgen_auto.py

4.2.3. If "*menu-2*" is selected:

■ **Outline of the process**

" [No.]" are shown in the following pages

#1 Preparation

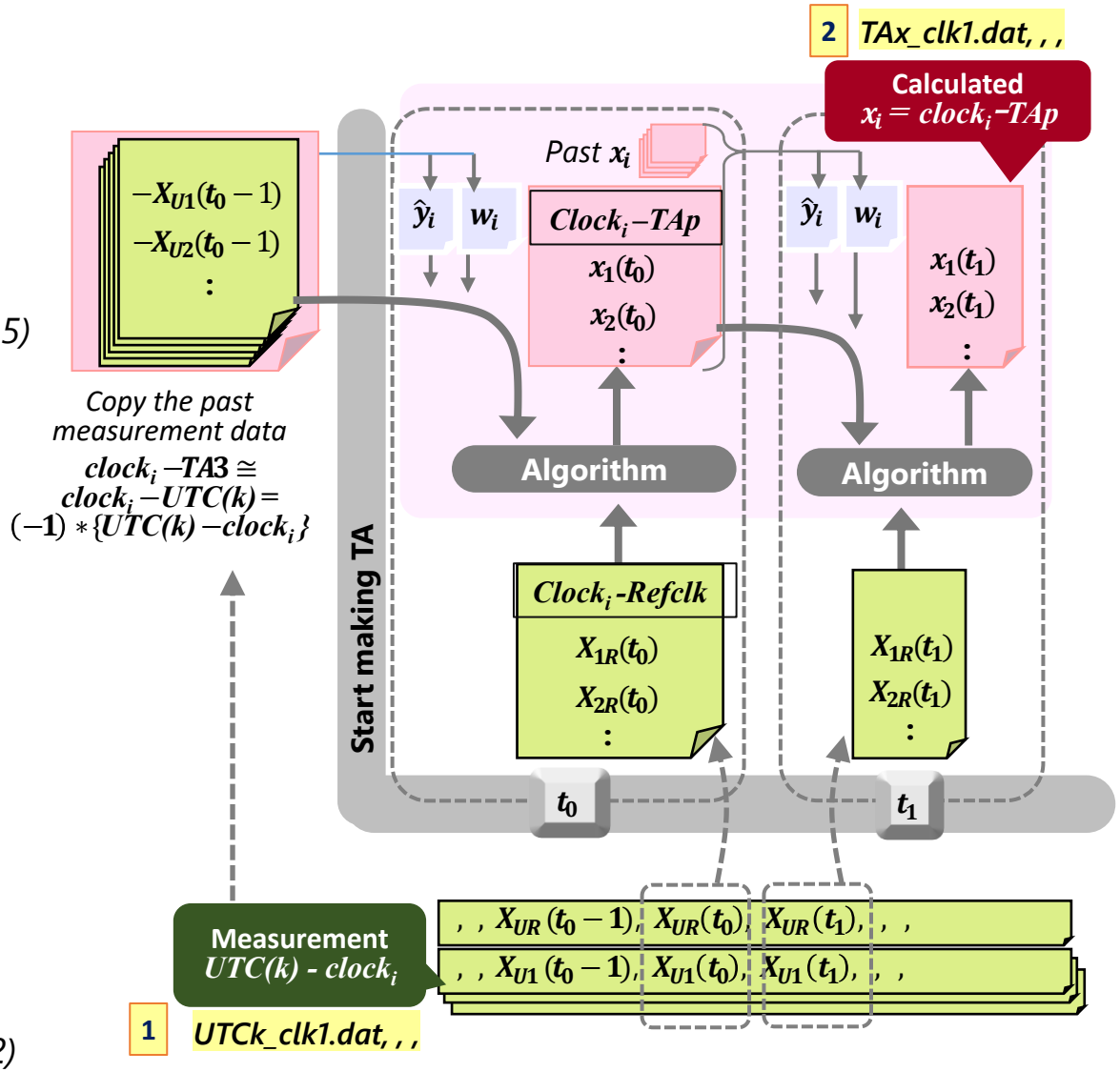
- Take over the parameters and clock data from the results of Initial processing ➔ [1][2][3][4] *(pp.14-15)*
  - Obtain the contents of "***param_list.txt***"
  - Obtain the contents of "***clock_file_list.txt***"
  - Obtain the contents of "***UTCk_(clock$_i$).dat***" 1
  - Obtain the range ***[MJD1, MJD2]***
- Set the parameters for rate & weight ➔ [6] *(pp.16-18)*
- Prepare the necessary data set ➔ [5][7] *(p.15,19)*
  - Calculate the "***clock$_i$ – Refclk***" from "***UTC(k) – clock$_i$***".
  - Copy the past "***clock$_i$ – UTC(k)***" into the data file of x_i :

#2 Calculation of daily *TAp*

- Iterative daily calculation ➔ [8] *(pp.20-21)*
  - Set the normalized weights
  - Calculate the prediction $\hat{x}_i$ of each clock
  - Calculate the daily x_i and accumulate it to "***TAx_(clock$_i$).dat***" 2

#3 Plotting & saving the results

- Calculate "***UTC(k) – TAp***" for evaluation ➔ [9] *(p.21)*
- Plot and save "***UTC(k) – TAp***" and its *ADEV* ➔ [10][11] *(p.22)*
  - The Result files are saved under "***./TA_RESULT/***".



4. Operation of each function

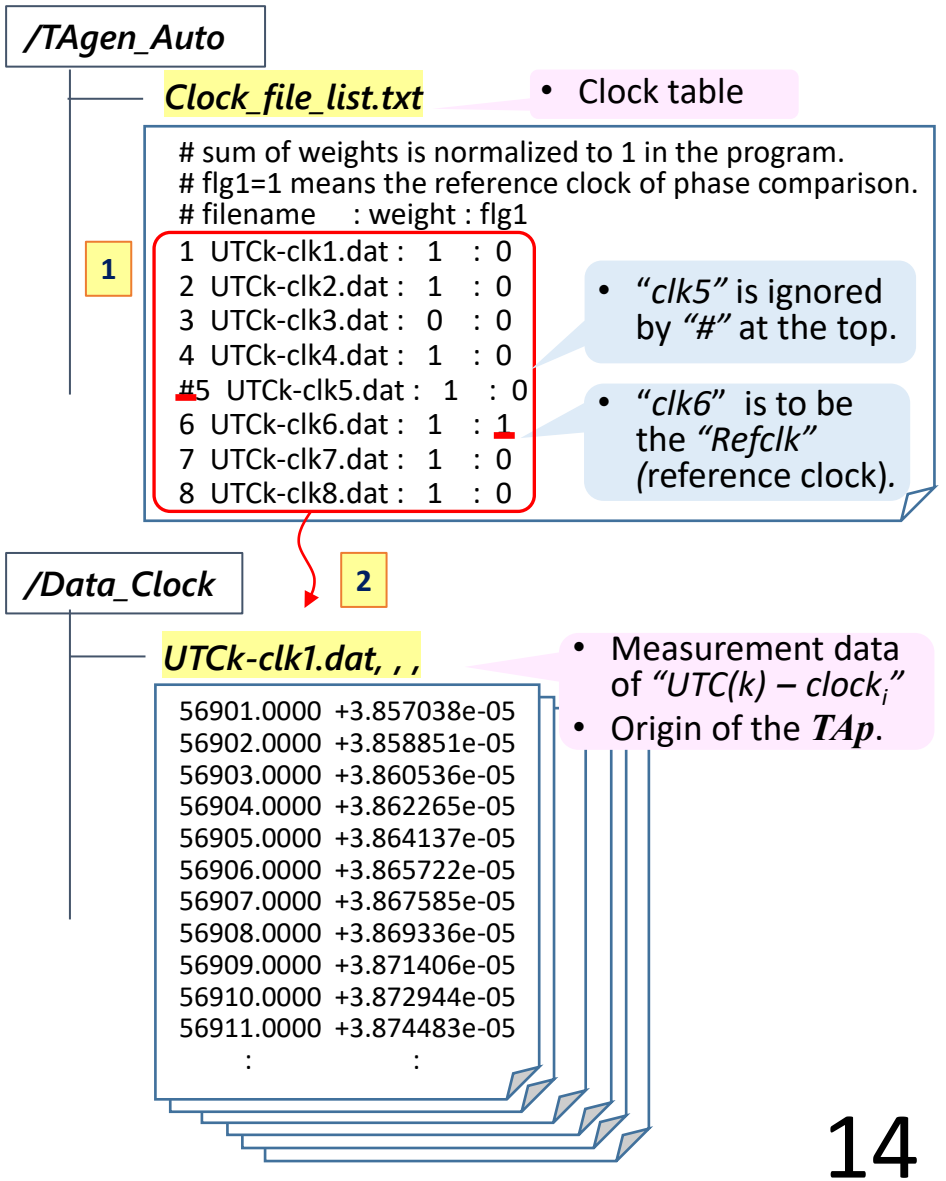
4.2.3. If "*menu-2*" is selected:

■ #1 Preparation

[1] Read the parameters from "*param_list.txt*"
← Initial Processing [1] *(p.6)*

[2] Read the parameter from "*clock_file_list.txt*" 1
← Initial Processing [2] *(p.6)*

[3] Read the assigned clock data "*UTCk-(clock$_i$).dat*" 2
← Initial Processing [3] *(p.6)*



4. Operation of each function

4.2.3. If "*menu-2*" is selected:

■ **#1 Preparation** (Cont.)

[4] Fix the available MJD range ← Initial Processing [4] *(p.6)*

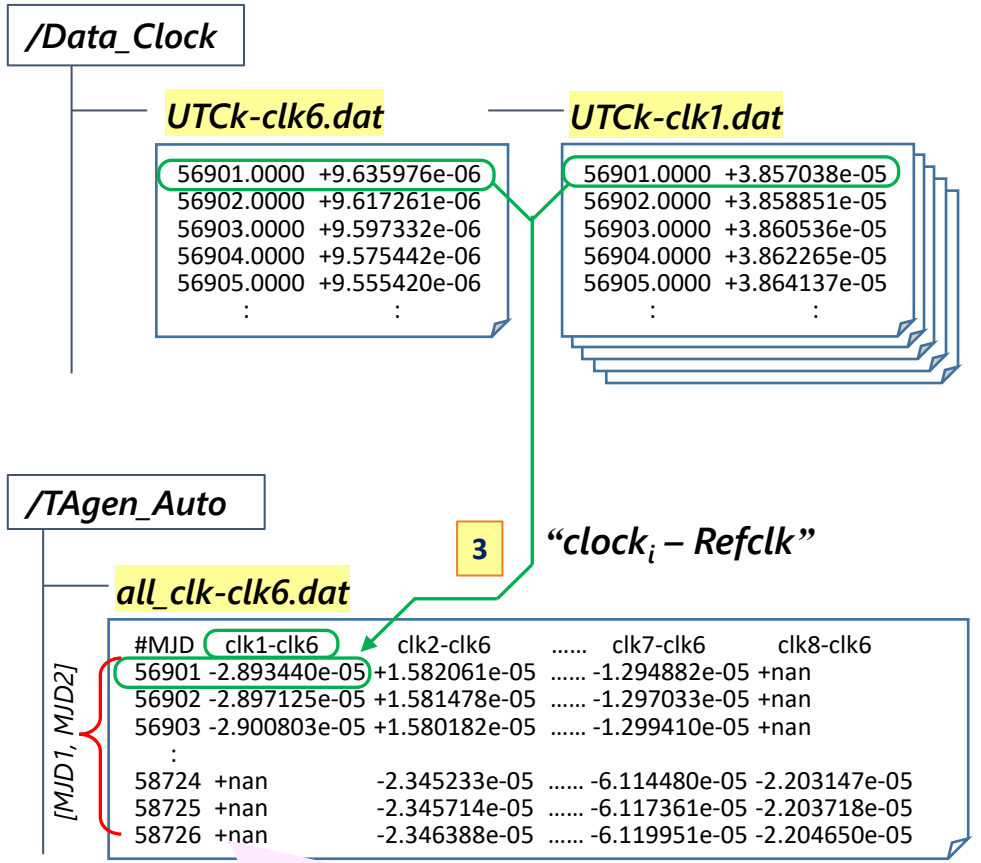
※ If ***[MJD1, MJD2]*** is defined by *"menu-1"* in advance, that range is adopted as a priority.

※ Mathematically, ***TAp*** can be calculated al least one clock. However, concentrating the weight on a single clock reduces the advantages of ***TAp***.

※ To avoid this problem, actual algorithms usually impose restrictions on the maximum weights and minimum number of clocks.

[5] Prepare the measurement data set of $X_{is}(t_k)$

- Calculate the measurement time difference of "***clock$_i$ – Refclk***" 3 from ***"UTCk-(clock$_i$).dat"***.
- Results are saved in the file ***"all_clk-(Refclk).dat"***. *(p.31)*



- Data file of all "$X_{is}(t_k) = clock_i - clock6$" except for the *clock$_i$* with the mark *"#"* in *"clock_file_list.txt"*

4. Operation of each function

4.2.3. If "*menu-2*" is selected:

■ #1 Preparation (Cont.)

[6] Set the parameters for rate and weight

[6.1] Enter T_{rate} as the span for calculating the rate

4

- Clock rate is calculated from the slope of the latest T_{rate} days.

$$\hat{y}_i(t_k) = \frac{x_i(t_k) - x_i(t_k - T)}{T_{rate}} \quad \boxed{Eq.\ (12)}\ (p.12)$$

- Any value for T_{rate} is permitted within the assigned MJD span.

※ The most suitable value is the one that gives the lowest ADEV.
(For a typical commercial Cesium clock, this may be around 30~60 days.)

```
=====================================
    Menu of the data processing
-------------------------------------
 0.  Plot the "UTC(k)-Clock" data
 1.  Assign the MJD range
 2.  Make TA with daily prediction
         and fixed clock weights
10.  Exit
=====================================
????? Input menu number ? >> 2

!!!!! UTCk-clk2.dat  includes outliers

*2** Set the span for clock rate calculation.
???? Span in days (try 10 or 30 or 90) = 40      4
**2** The rate span =  40

**2** Set the ADEV_tau for setting the clock weight.
???? Span in days (16 or 32 or 64 or 128) = 30   5
**2** The ADEV tau =  30
```

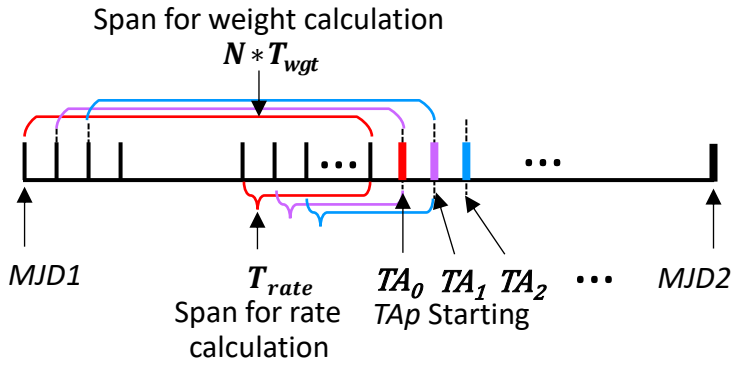
[6.2] Enter T_{wgt} as the span for calculating weights

5

- The clock weight is inversely proportional to the Allan Variance $\sigma_y{}^2(T_{wgt})$, and is updated daily.

$$w_i(t_k) \propto \frac{1}{\sigma_y{}^2(T_{wgt})} \quad Here \sum_i w_i = 1 \quad (p.12)$$

- $\sigma_y(T_{wgt})$ is calculated from the past latest x_i values within the "$N * T_{wgt}$" range using the Python library "***AllanTools***" *[Ref. 2]*. (Cont.)



4. Operation of each function

TAgen_auto.py

4.2.3. If "*menu-2*" is selected:

■ #1 Preparation

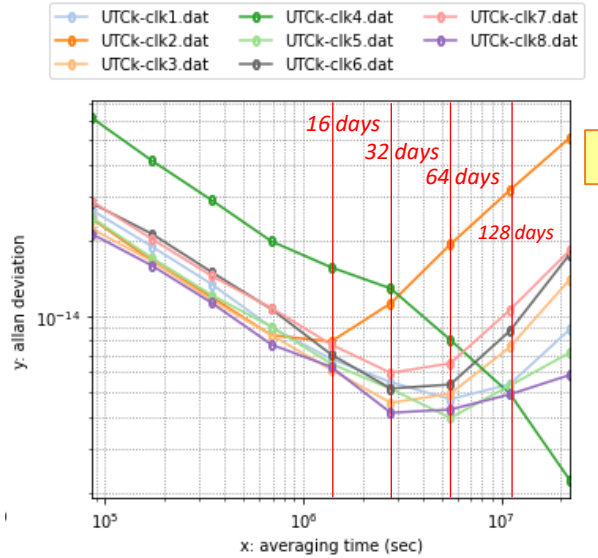
[6] Set the parameters for rate and weight

[6-2] Enter T_{wgt} as the span for calculating weights (Cont.)

$$w_i(t_k) \propto \frac{1}{{\sigma_y}^2(T_{wgt})} \qquad Here \sum_i w_i = 1$$

※ **As for the appropriate setting of *Twgt_N* :**

- A clock can join the ***TAp*** calculation if its rate and weight values can be prepared. This means that the clock's participation date (here T_0) must satisfy the conditions of "$T_0 > T_{rate}$" and "$T_0 > N * T_{wgt}$".
- To improve the accuracy of ADEV, extending the range of "$N * T_{wgt}$" is desirable. For that, a substantial amount of data stock is required, particularly for large T_{wgt}. In that case, the start date of ***TAp*** (named as TA_0 here) will be later, so that the time series of ***TAp*** becomes shorter.
- Due to these conflicting conditions (improving the precision of ADEV and realizing the longest time series of ***TAp***), the N value must be carefully determined.
- N is given as ***Twgt_N*** in "***param_list.txt***". *(p.29)*



6

※ **As for the determination way of T_{wgt} :**

- Any value for T_{wgt} within the assigned MJD span is allowed in the program. 5
- In this program, *"16, 32, 64, 128 days"* are suggested for T_{wgt}. 6
- The ADEV is calculated in Octave; the available τ for $\sigma_y(\tau)$ are "*1, 2, 4,,, days*". 6
- If the input T_{wgt} value does not correspond with the available above "τ"s, the $\sigma_y(\tau)$ by the closest τ to T_{wgt} is used for the weight calculation. 5

4.2.3. If "*menu-2*" is selected:

■ #1 Preparation

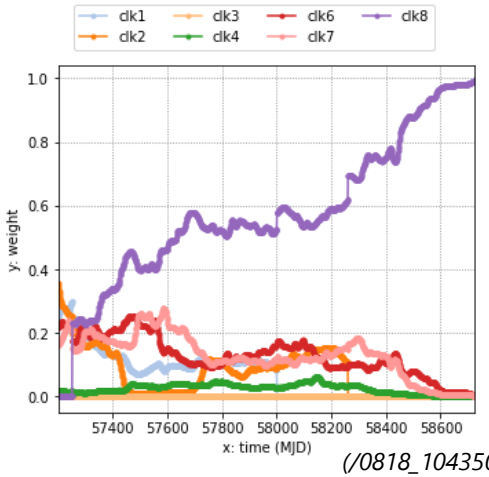
[6] Set the parameters for rate and weight (Cont.)

[6.3] Select whether to set the weight limit or not 7

※ Maximum weight limit is given in the "***param_list.txt***" *(p.29)*.

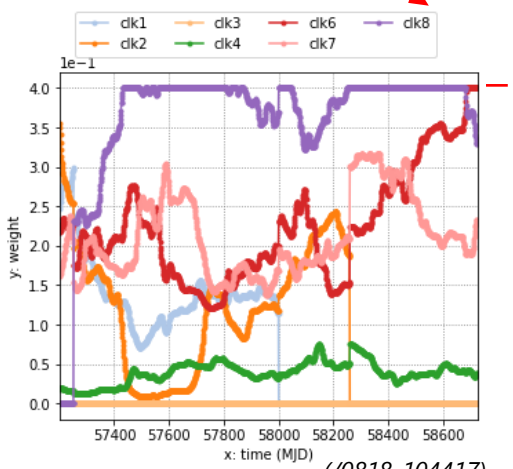
- If "***n***", the weights are simply calculated as shown in [6.2]. *(pp.16-17)*
- If "***y***", each weight is limited by the given limit.

limit OFF



(/0818_104350)

limit ON



(/0818_104417)

Saturation at the maximum limit =0.4 8

*Clock weights behavior during the **TA** calculation.*

```
=======================================
    Menu of the data processing
---------------------------------------
 0.  Plot the "UTC(k)-Clock" data
 1.  Assign the MJD range
 2.  Make TA with daily prediction
         and fixed clock weights
10.  Exit
=======================================
????? Input menu number ? >> 2

!!!!! UTCk-clk2.dat  includes outliers

*2** Set the span for clock rate calculation.
????? Span in days (try 10 or 30 or 90) =  40
**2** The rate span =  40

**2** Set the ADEV_tau for setting the clock weight.
????? Span in days (16 or 32 or 64 or 128) = 30
**2** The ADEV tau =  30

????? Set the weight_limit? (y/n) =  y   8
**2** weight limit =  0.4   8
```

※ *Both are the results under the following condition:*
rate span=40, ADEV tau=30, weight limit =0.4.

4. Operation of each function

TAgen_auto.py

4.2.3. If "*menu-2*" is selected:

■ #1 Preparation

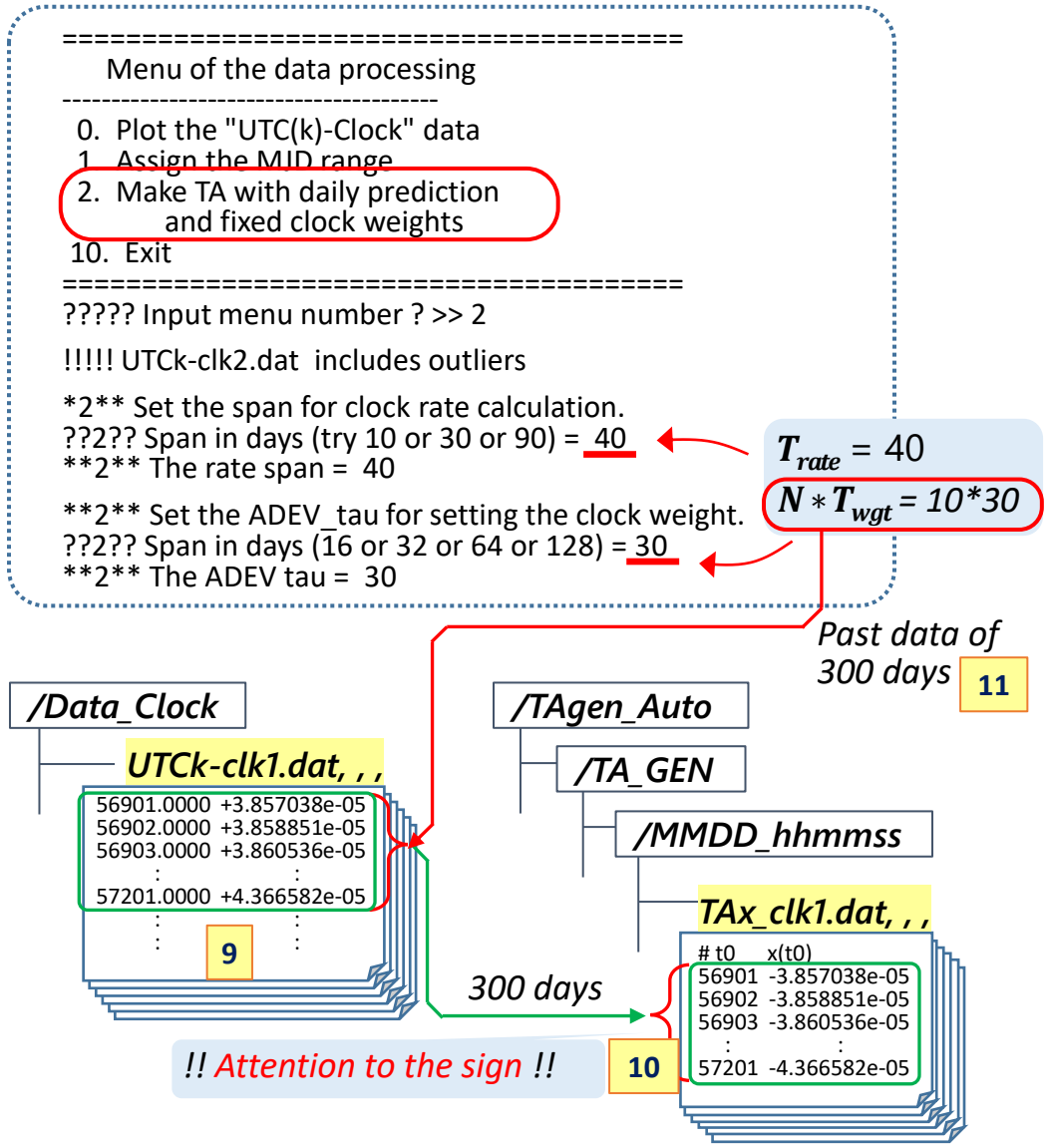
[7] Prepare the initial data set of x_i in the file "*TAx_(clock$_i$).dat*" to start the *TAp* calculation

- As shown in *Eq.(10) (p.12)*, the data set of past x_i and $\hat{y}_i$ values are required to start the TAp calculation.
- In particular, $\hat{y}_i$ needs the x_i over the latest T_{rate} days, and w_i needs the x_i over the latest "$N * T_{wgt}$" days. *(pp.16-17)*
- As no past x_i are at the beginning, "***UTC(k) – clock$_i$***" is used as a substitute for the initial x_i. 9

$$clock_i(t) - UTC(k)(t) \rightarrow clock_i(t) - TAp(t) \equiv x_i(t) \quad (t < \text{start date})$$

- This past "***clock$_i$ – UTC(k)***" data over the necessary range*1 are copied in "***TAx_(clock$_i$).dat***". 10

**1 : This range is determined by the longer one of T_{rate} or "$N * T_{wgt}$".* 11



4. Operation of each function

TAgen_auto.py

4.2.3. If "*menu-2*" is selected:

■ #2 Calculation of daily TAp

[8] Calculate $x_i(t_k)$ for "$TA_0 \le t_k \le MJD2$"

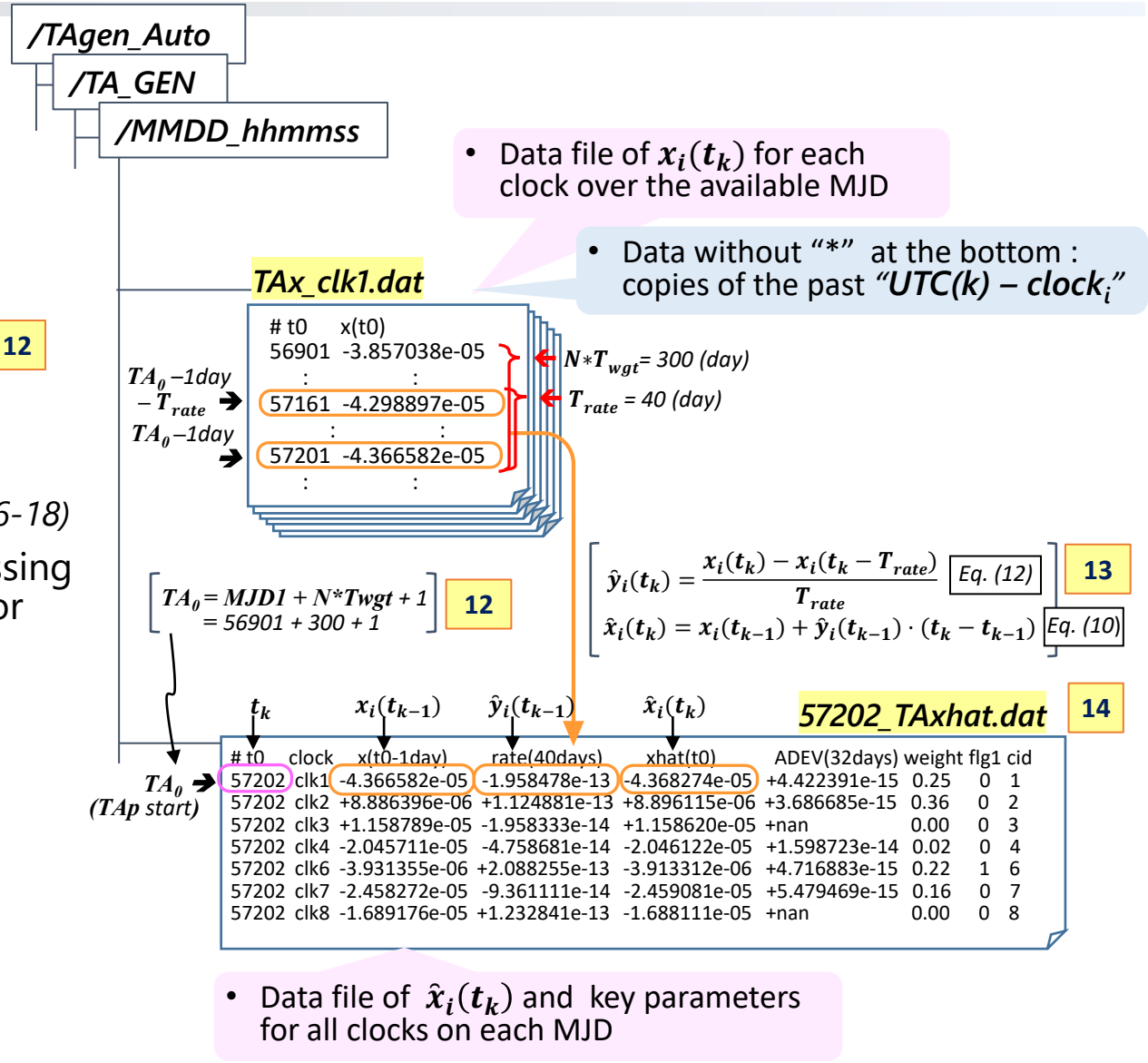
- The TAp calculation start date is "$TA_0 = MJD1+T+1$". (T is the longer of T_{rate} or $N*T_{wgt}$.) **12**

[8.1] Set the weights for the available clocks

- The weights are calculated as shown in [6.2][6.3]. *(pp.16-18)*
- If the measurement data for a clock is not normal (missing or an outlier), the weight of this clock becomes zero for that date.

[8.2] Calculate the prediction $\hat{x}_i(t_k)$

- $\hat{x}_i(t_k)$ is calculated by *Eq.(10)* *(p.12)* using the past x_i.
- "$\hat{y}_i(t_{k-1})$" is calculated as denoted in [6.1]. *(p.16)* **13**
- The calculated daily results are saved in the files ***"(MJD)_TAxhat.dat"***. *(p.32)* **14**



4. Operation of each function

TAgen_auto.py

4.2.3. If "*menu-2*" is selected:

■ #2 Calculation of daily TAp (Cont.)

[8] Calculate $x_i(t_k)$ for "$TA_0 \le t_k \le MJD2$"

[8.3] Calculate the daily $x_i(t_k)$

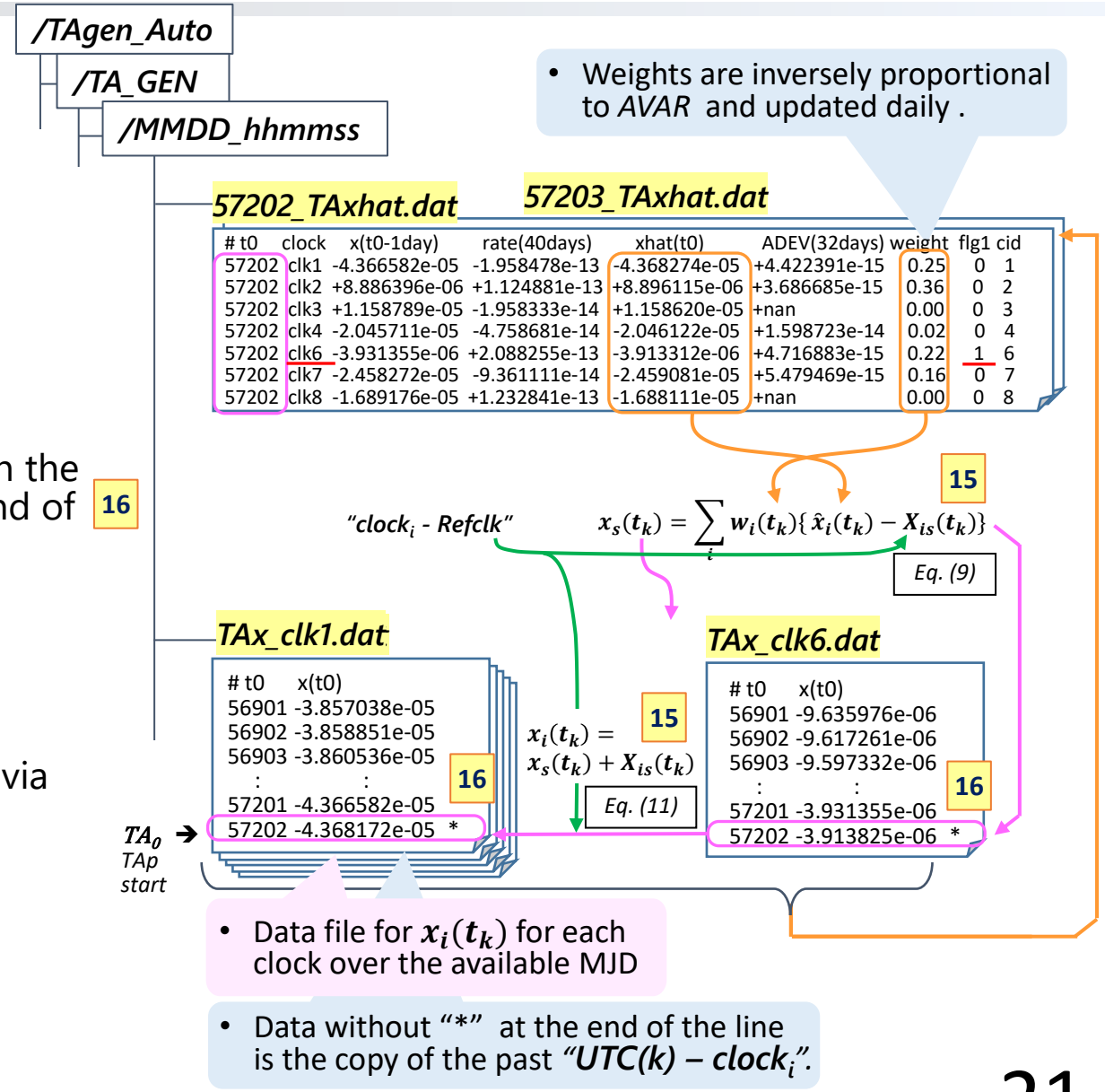
- Calculate $x_i(t_k)$ of each clock by Eq. (9)(11). (p.12). 15
- The calculated time series of $x_i(t_k)$ are accumulated in the files "***TAx_(clock$_i$).dat*** with the star mark " * " at the end of the line. 16

■ #3 Plotting & saving the results

[9] Calculate "*UTC(k)-TAp*" for evaluation

- TAp can be compared with other timescales or clocks via common reference, for example ***UTC(k)***.
- "***UTC(k) – TAp***" is obtained as follows (*Refer to [Ref. 2]*).

$$UTC(k) - TAp = \underbrace{[UTC(k) - h_i]}_{\text{"UTCk-(clock}_i\text{).dat"}} + \underbrace{[h_i - TAp]}_{\text{Calculated } x_i} \quad \text{Eq. (13)}$$



4. Operation of each function

TAgen_auto.py

4.2.3. If "*menu-2*" is selected:

■ #3 Plotting & saving the results (Cont.)

[10] Plot the results 12

- Left : time difference vs ***UTC(k)*** *(p.40)*
- Middle : *ADEV*s of left data *(p.40)*
- Right : clock weights *(p.41)*

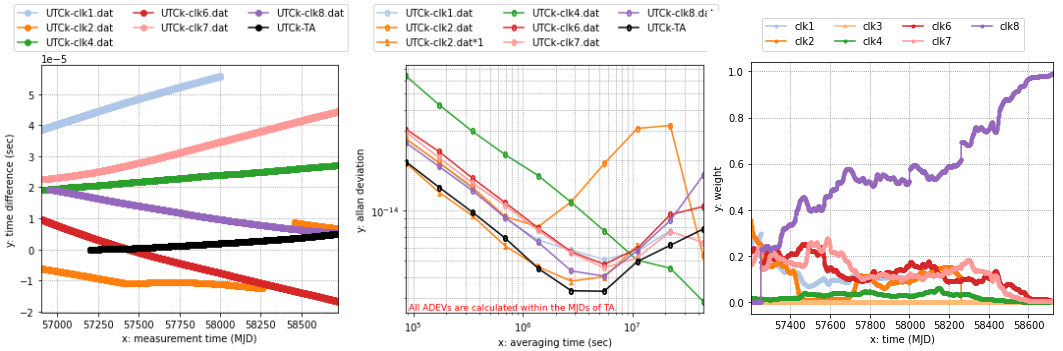
[11] Save the results, if required 13

- Results are saved at ***"./TA_RESULT/(time)/".*** *(pp.35-41)*

- ***Log_(time).txt*** : history of the process
- ***fig _(time).pdf*** : figure of "***UTC(k) – clock$_i$***" & *ADEV*
  all_raw_clk_TA_(time).dat : time difference vs ***UTC(k)***
- ***all_adev_(time).dat*** : *ADEV* of "***UTC(k) – clock$_i$***"
- ***all_weight_(time).dat*** : daily calculated weights
- ***wgt_fig_(time).dat*** : figure of all weights

```
????? Span in days (recommended 10 or 30 or 90) =  30

***** Calculating a timescale for  57202
                    :
***** Calculating a timescale for  58726
!!!! 58726 clk1 ADEV impossible
!!!! 58726 clk2 ADEV impossible
      58726 clk4 adev(32days)= 1.184538e-14
      58726 clk6 adev(32days)= 5.907734e-15
      58726 clk7 adev(32days)= 7.411210e-15
      58726 clk8 adev(32days)= 3.930297e-16

!!!!! UTCk-clk2.dat  includes outliers
!!!!! UTCk-clk2.dat  includes outliers
```

12




```
**2** make a timescale with prediction *****

????? Save figure? (y/n) = y
::::: save logfile =  ./TA_RESULT/0818_104350/log_0818_104350.txt
::::: save figfile =  ./TA_RESULT/0818_104350/fig_0818_104350.pdf
::::: save alldata =  ./TA_RESULT/0818_104350/all_raw_clk_TA_0818_104350.dat
::::: save ADEVdat =  ./TA_RESULT/0818_104350/all_adev_0818_104350.dat
::::: save weights =  ./TA_RESULT/0818_104350/all_weight_0818_104350.dat
::::: save wgt_fig =  ./TA_RESULT/0818_104350/wgt_fig_0818_104350.pdf
```

13

4. Operation of each function

TAgen_auto.py

Pytool_TAgen/TAgen_Auto/TA_RESUT/0818_104350/

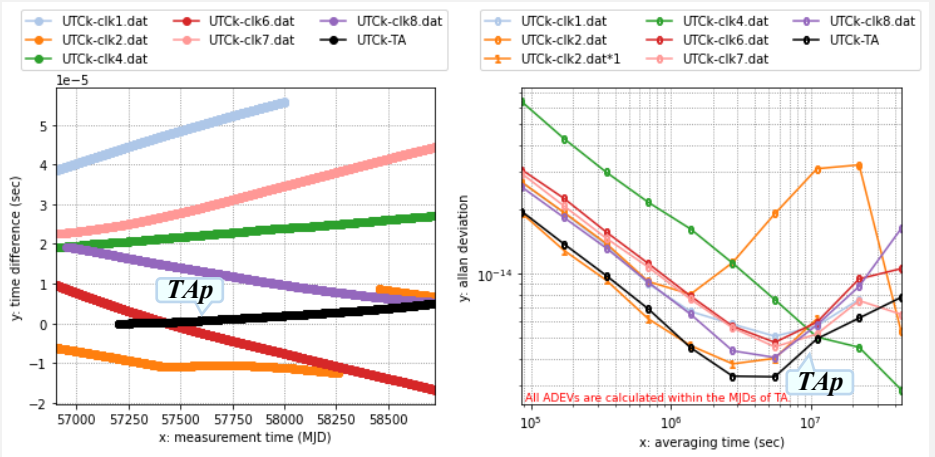
4.2.3. If "*menu-2*" is selected:

all_raw_clk_TA_0818_104350.dat

fname	UTCk-clk1.dat	UTCk-clk2.dat	UTCk-clk3.dat	UTCk-clk4.dat	UTCk-clk6.dat	UTCk-clk7.dat	UTCk-clk8.dat	UTCk-TA
weight	1.0	1.0	0.0	1.0	1.0	1.0	1.0	0.0
flag	0	0	0	0	1	0	0	3
cid	1	2	3	4	6	7	8	0
56901	+3.857038e-05	-6.184638e-06	-1.215249e-05	+1.918367e-05	+9.635976e-06	+2.258480e-05	+nan	+nan
:	:	:	:	:	:	:	:	:
57201	+4.366582e-05	-8.886396e-06	-1.158789e-05	+2.045711e-05	+3.931355e-06	+2.458272e-05	+1.689176e-05	+nan
57202	+4.368122e-05	-8.896972e-06	-1.158562e-05	+2.045840e-05	+3.913326e-06	+2.459225e-05	+1.688237e-05	-4.990688e-10
:	:	:	:	:	:	:	:	:
58726	+nan	+6.687379e-06	-3.180825e-06	+2.708426e-05	-1.677650e-05	+4.442301e-05	+5.270004e-06	+5.026417e-06

TAp calculation starts from $MJD1+N*T_{wgt}+1$ = 56901 +300 +1

fig_0818_104350.pdf

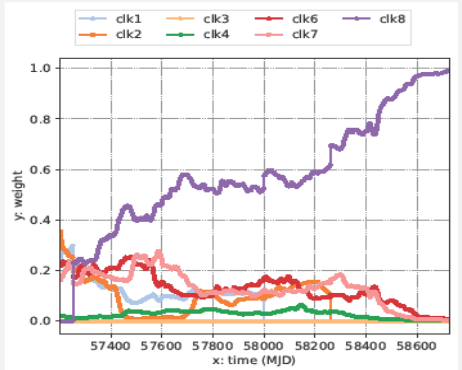


all_weight_0818_104350.dat

# MJD	clk1	clk2	clk3	clk4	clk6	clk7	clk8
57202.0	0.247	0.356	0.000	0.019	0.217	0.161	0.000
:	:	:	:	:	:	:	:
58726.0	0.000	0.000	0.000	0.001	0.004	0.003	0.992

Daily calculated weight is inversely proportional to Allan variance. This case is without the maximum weight limit.

Wgt_fig_0818_104350.pdf



log_0818_104350.txt

```
# Data log at 2025.08.18_10h43m_50s
#
**0** Read and plot initial clock data and Allan deviation*****
::::: Input clocklist   : clock_file_list.txt
::::: Input clock file  : ['UTCk-clk1.dat', 'UTCk-clk2.dat', 'UTCk-clk3.dat', 'UTCk-
clk4.dat', 'UTCk-clk6.dat', 'UTCk-clk7.dat', 'UTCk-clk8.dat']
::::: data time interval (t_intval): 86400(sec)
::::: Outlier limit (x_limit))     : 1000.0

**2** make a timescale with prediction *****
::::: saved TAxhat data : ./TA_GEN/0522_172900/(MJD)_TAxhat.dat
::::: saved TAx data : ./TA_GEN/0522_172900/TAx_(clock).dat
::::: interval for rate estimation (TArate_span)          : 40(days)
::::: ADEV tau for clock weight (wgt_adev_tau0)       : 32(days)
::::: max weight limit (wgt_lmt)                          : OFF
::::: sample numbers for ADEV for weighting (Twgt_N)) : 10
```

*If there are not sufficient samples, ADEV cannot be calculated, and is expressed as "**nan**".*

***MJDspan** is the period used in the ADEV calculation.*

all_adev_0818_104350.dat

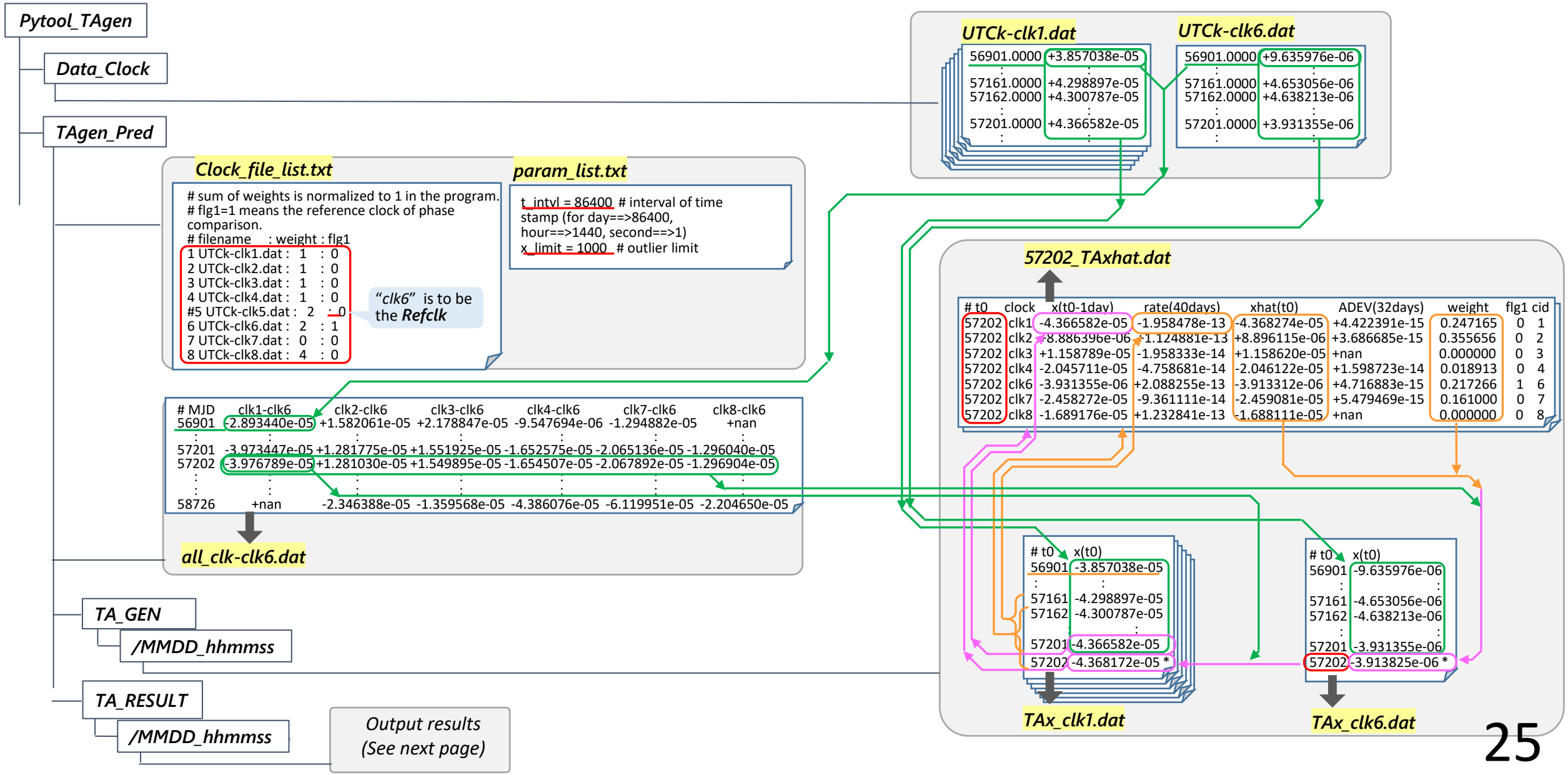
Tau(s)	UTCk-clk1.dat	UTCk-clk2.dat	UTCk-clk2.dat*1	UTCk-clk4.dat	UTCk-clk6.dat	UTCk-clk7.dat	UTCk-clk8.dat	UTCk-TA
MJDspan	57202-58000	57202-58260	58457-58726	57202-58726	57202-58726	57202-58726	57202-58726	57202-58726
86400	+2.671314e-14	+2.672821e-14	+1.914455e-14	+6.349398e-14	+3.056365e-14	+2.933094e-14	+2.537478e-14	+1.950581e-14
172800	+1.919744e-14	+1.931040e-14	+1.284166e-14	+4.272830e-14	+2.246428e-14	+2.070340e-14	+1.829287e-14	+1.375601e-14
345600	+1.359101e-14	+1.383787e-14	+9.326690e-15	+2.980696e-14	+1.566171e-14	+1.464787e-14	+1.317546e-14	+9.803913e-15
691200	+8.965388e-15	+9.188727e-15	+6.149651e-15	+2.154658e-14	+1.114723e-14	+1.072491e-14	+9.112607e-15	+6.841218e-15
1382400	+6.661882e-15	+8.011344e-15	+4.614288e-15	+1.618386e-14	+7.863938e-15	+7.597546e-15	+6.468798e-15	+4.512655e-15
2764800	+5.808297e-15	+1.127373e-14	+3.789893e-15	+1.119640e-14	+5.678597e-15	+5.582861e-15	+4.362013e-15	+3.320030e-15
5529600	+5.100534e-15	+1.905124e-14	+4.020702e-15	+7.564302e-15	+4.776232e-15	+4.551563e-15	+4.058927e-15	+3.297101e-15
11059200	+5.704858e-15	+3.093775e-14	+6.159347e-15	+5.057043e-15	+5.936237e-15	+5.225535e-15	+5.760070e-15	+4.961455e-15
22118400	+7.589139e-15	+3.222987e-14	+nan	+4.528442e-15	+9.449766e-15	+7.483117e-15	+8.720073e-15	+6.200905e-15
44236800	+nan	+5.349799e-15	+nan	+2.866345e-15	+1.056197e-14	+6.455195e-15	+1.625726e-14	+7.754905e-15

A-1. Input / Output files :

A-1.1. Relation of the files

A-1.1.1. Relation of the files (1)

TAgen_auto.py



A-1.1.1. Relation of the files (2)

TAgen_auto.py

Pytool_TAgen/TAgen_Auto/TA_RESULT/0818_104350/

Clock_file_list.txt

```
# sum of weights is normalized
to 1 in the program.
# flg1=1 means the reference
clock of phase comparison.
# filename     : weight : flg1
1 UTCk-clk1.dat :  1   : 0
2 UTCk-clk2.dat :  1   : 0
3 UTCk-clk3.dat :  0   : 0
4 UTCk-clk4.dat :  1   : 0
#5 UTCk-clk5.dat :  1   : 0
6 UTCk-clk6.dat :  1   : 1
7 UTCk-clk7.dat :  1   : 0
8 UTCk-clk8.dat :  1   : 0
```

UTCk-clk1.dat

```
56901.0000 +3.857038e-05
     :          :
57201.0000 +4.366582e-05
57202.0000 +4.368122e-05
     :          :
```

all_raw_clk_TA_0818_104350.dat

fname	UTCk-clk1.dat	UTCk-clk2.dat	UTCk-clk3.dat	UTCk-clk4.dat	UTCk-clk6.dat	UTCk-clk7.dat	UTCk-clk8.dat	UTCk-TA
Weight	1.0	1.0	0.0	1.0	1.0	1.0	1.0	0.0
flag	0	0	0	0	1	0	0	3
cid	1	2	3	4	6	7	8	0
56901	+3.857038e-05	-6.184638e-06	-1.215249e-05	+1.918367e-05	+9.635976e-06	+2.258480e-05	+nan	+nan
:	:	:	:	:	:	:	:	:
57201	+4.366582e-05	-8.886396e-06	-1.158789e-05	+2.045711e-05	+3.931355e-06	+2.458272e-05	+1.689176e-05	+nan
57202	+4.368122e-05	-8.896972e-06	-1.158562e-05	+2.045840e-05	+3.913326e-06	+2.459225e-05	+1.688237e-05	-4.990688e-10
:	:	:	:	:	:	:	:	:
58726	+nan	+6.687379e-06	-3.180825e-06	+2.708426e-05	-1.677650e-05	+4.442301e-05	+5.270004e-06	+5.026417e-06

all_adev_0818_104350.dat

Tau(s)	UTCk-clk1.dat	UTCk-clk2.dat	UTCk-clk2.dat*1	UTCk-clk4.dat	UTCk-clk6.dat	UTCk-clk7.dat	UTCk-clk8.dat	UTCk-TA
MJDspan	57202-58000	57202-58260	58457-58726	57202-58726	57202-58726	57202-58726	57202-58726	57202-58726
86400	+2.671314e-14	+2.672821e-14	+1.914455e-14	+6.349398e-14	+3.056365e-14	+2.933094e-14	+2.537478e-14	+1.950581e-14
172800	+1.919744e-14	+1.931040e-14	+1.284166e-14	+4.272830e-14	+2.246428e-14	+2.070340e-14	+1.829287e-14	+1.375601e-14
:	:	:	:	:	:	:	:	:
11059200	+5.704858e-15	+3.093775e-14	+6.159347e-15	+5.057043e-15	+5.936237e-15	+5.225535e-15	+5.760070e-15	+4.961455e-15
22118400	+7.589139e-15	+3.222987e-14	+nan	+4.528442e-15	+9.449766e-15	+7.483117e-15	+8.720073e-15	+6.200905e-15
44236800	+nan	+5.349799e-15	+nan	+2.866345e-15	+1.056197e-14	+6.455195e-15	+1.625726e-14	+7.754905e-15

log_0818_104350.txt

```
# Data log at 2025.08.18_10h43m_50s
#
**0** Read and plot initial clock data and Allan deviation*****
::::: Input clocklist   : clock_file_list.txt
::::: Input clock file  : ['UTCk-clk1.dat', 'UTCk-clk2.dat', 'UTCk-clk3.dat', 'UTCk-
clk4.dat', 'UTCk-clk6.dat', 'UTCk-clk7.dat', 'UTCk-clk8.dat']
::::: data time interval (t_intval): 86400(sec)
::::: Outlier limit (x_limit))     : 1000.0

**2** make a timescale with prediction *****
::::: saved TAxhat data : ./TA_GEN/0818_104218/(MJD)_TAxhat.dat
::::: saved TAx data : ./TA_GEN/0818_104218/TAx_(clock).dat
::::: interval for rate estimation (TArate_span)          : 40(days)
::::: ADEV tau for clock weight (wgt_adev_tau0)           : 32(days)
::::: max weight limit (wgt_lmt)                          : OFF
::::: sample numbers for ADEV for weighting (Twgt_N))  : 10
```

all_weight_0818_104350.dat

# MJD	clk1	clk2	clk3	clk4	clk6	clk7	clk8
57202.0	0.247	0.356	0.000	0.019	0.217	0.161	0.000
:	:	:	:	:	:	:	:
57251.0	0.299	0.254	0.000	0.017	0.227	0.203	0.000
57252.0	0.231	0.196	0.000	0.013	0.175	0.154	0.231
:	:	:	:	:	:	:	:
58000.0	0.084	0.088	0.000	0.027	0.159	0.115	0.526
58001.0	0.000	0.097	0.000	0.030	0.173	0.125	0.575
:	:	:	:	:	:	:	:
58725.0	0.000	0.000	0.000	0.001	0.004	0.003	0.992
58726.0	0.000	0.000	0.000	0.001	0.004	0.003	0.992

wgt_fig_0818_104350.pdf

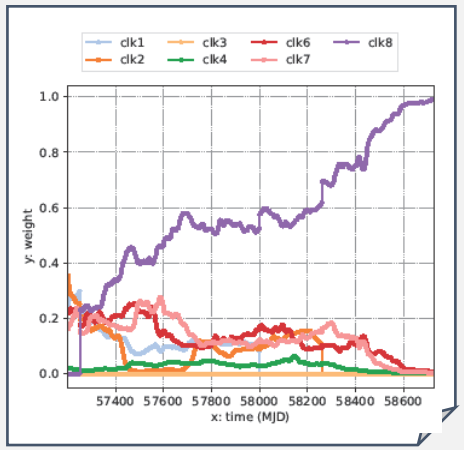
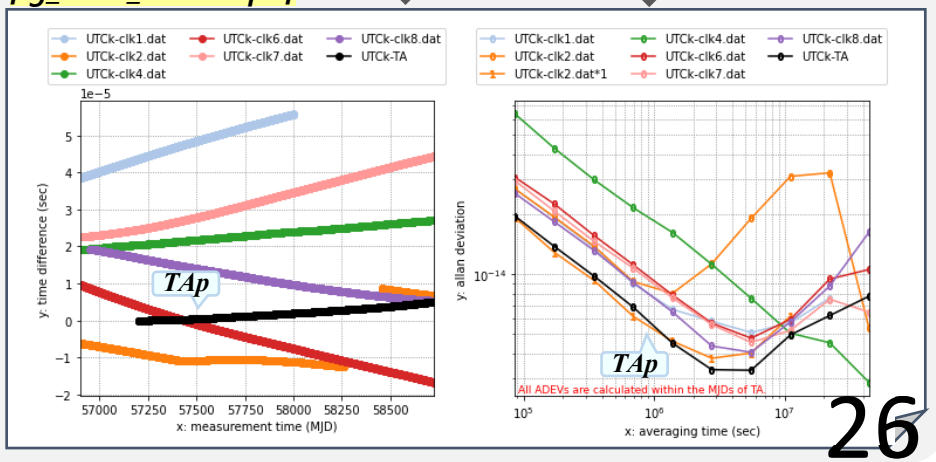


fig_0818_104350.pdf



A-1. Input / Output files :

A-1.2. Files used in the calculation

A-1.2.1 *"clock_fie_list.txt"*

■ *"/Pytool_TAgen/TAgen_Auto/**clock_file_list.txt**"*

Outline

- "***clock_file_list.txt***" provides the conditions of each clock in the averaging process.
- This table should be prepared in advance.

Details / Rules of use

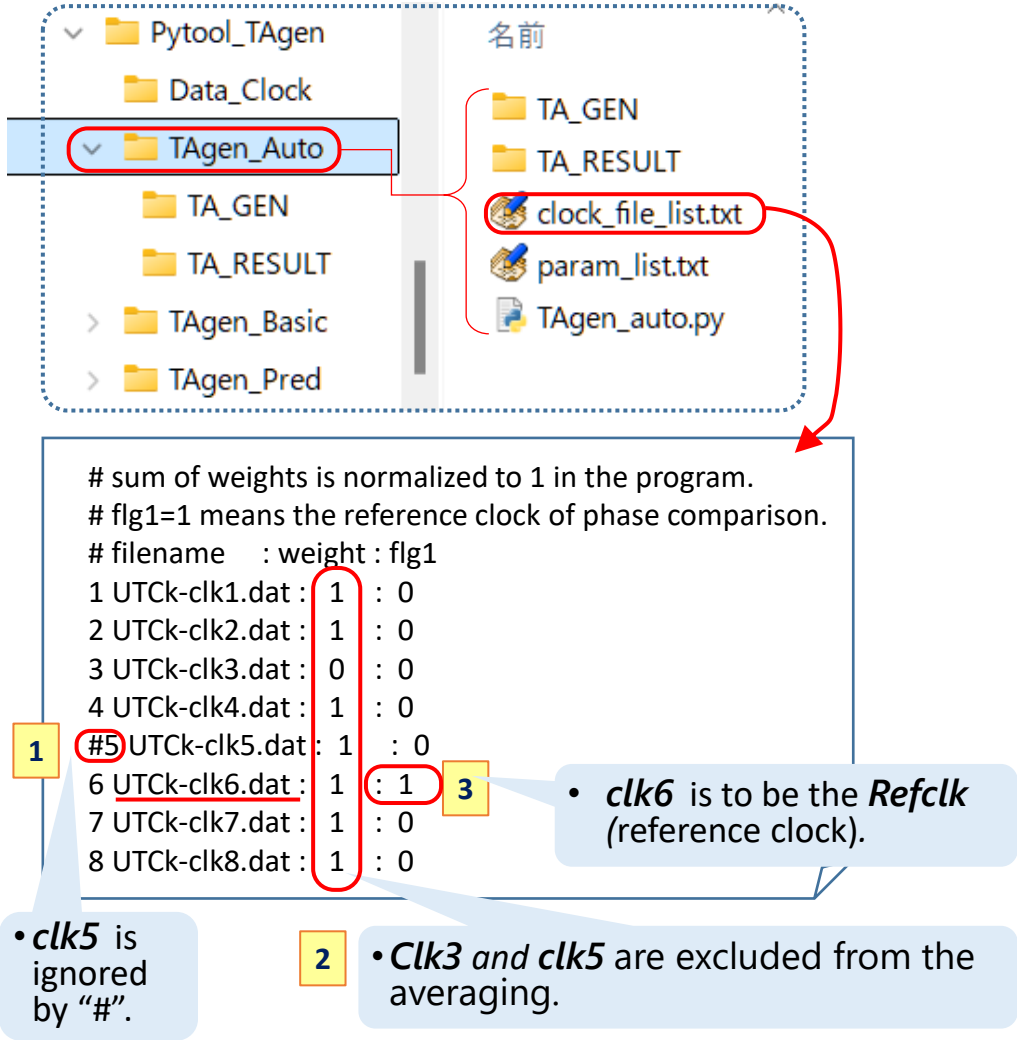
- The line starting with "#" is ignored. 1

- 1st column : file No. (defines plot color)
- 2nd column : data file name
- 3rd column : weight flag*1
- 4th column : flag of the reference clock *2.

Note

*1 : The clock with "*weight =0*" is excluded in the averaging. 2

*2 : The clock with "***flg1=1***", the reference clock named as "***Refclk***" hereafter, is used as the base clock to get the time difference between clocks. 3



A-1.2.2 *"param_list.txt"*

■ *"/Pytool_TAgen/TAgen_Auto/**param_list.txt**"*

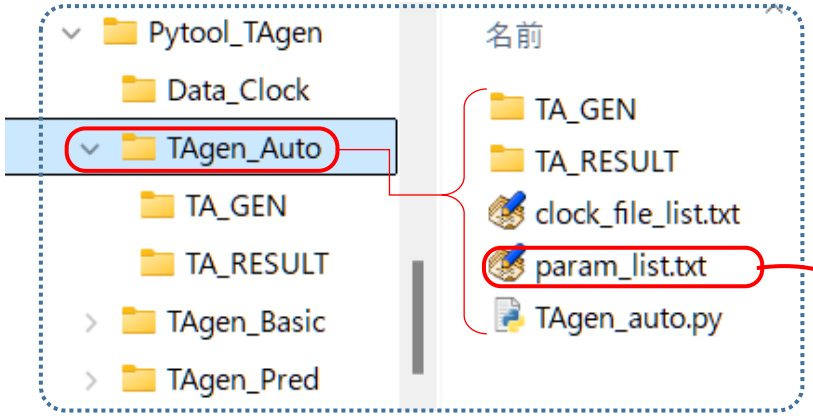
Outline

- *"**param_list.txt**"* provides the parameters for the ***TAp*** calculation.
- Parameters that are fixed within the program but can be changed on demand are stored here.
- This table should be prepared in advance.

Details / Rules of use

- These parameters are used with the same names in the program.
- The text after "#" provides the explanation of the parameter. 1

- ***t_intvl*** : interval of time stamp (for day → *86400*, hour → *1440*, second → *1*)
- ***x_limit*** : outlier limit
- ***Twgt_N*** : sample numbers required for the ADEV (for weighting)
- ***wgt_lmt*** : maximum weight limit
  (larger than *1/(available clocks for averaging)*, smaller than *1*)




```
t_intvl = 86400  # interval of time stamp (for day==>86400,
                   hour==>1440, second==>1)
x_limit = 1000   # outlier limit
Twgt_N  = 10     # sample numbers required for the ADEV (for
weighting)
wgt_lmt = 0.4    # maximum weight limit (larger than 1/(available
clocks for averaging), smaller than 1)
```

1

A-1.2.3 *"UTCk_(clock$_i$).dat"*

■ *"/Pytool_TAgen/Data_Clock/**UTCk-(clock$_i$).dat**"*

Outline

- *"**UTCk-(clock$_i$).dat**"* are the measurement clock data as the materials of ***TAp*** calculation.
- This data files should be prepared in advance.

Details / Rules of use

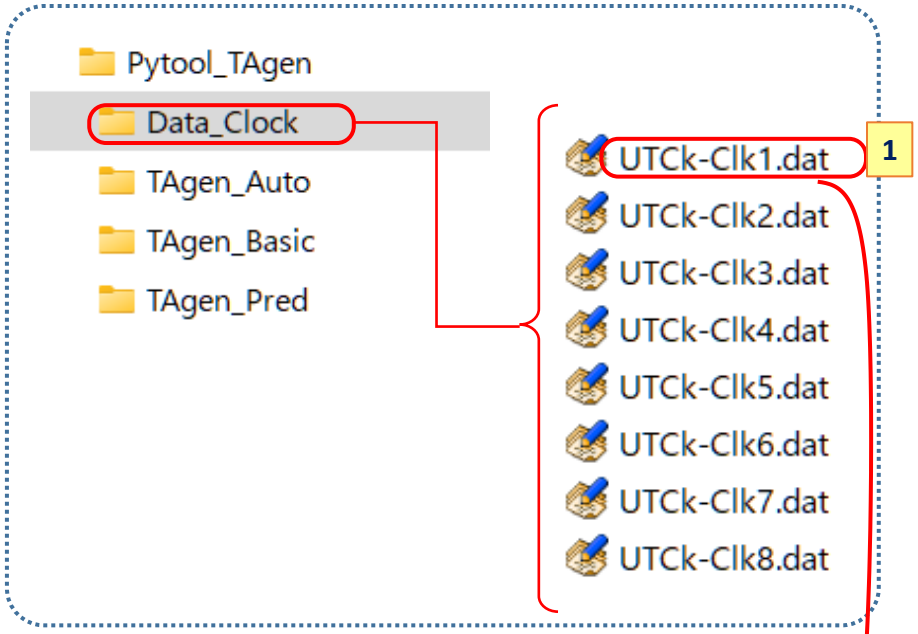
- One file contains data from a single clock.
- The file name should be *"**UTCk-****.dat**"*, where the four characters "********" identify each clock. 1
- 1st column : observation time stamp 2
- 2nd column : measured time difference of *"**UTC(k) – clock$_i$**"* (in second) 3

Note

※ The Interval of the 1st column should be specified as ***t_intvl*** in *"**param_list.txt**"*. *(p.29)* 4

※ This sampling interval must be constant and continuous.

※ Sufficient data is required for each clock to start the calculation. *(p.19)*



A-1.2.4 *"all_clk_(Refclk).dat"*

■ *"/Pytool_TAgen /TAgen_Auto/**all_clk_(Refclk).dat**"*

Outline

- *"**all_clk_(Refclk).dat**"* is the merged data of time difference *"**clock$_i$ – Refclk**"* calculated from the *"**UTCk-(clock$_i$).dat**"* files.
- This file is an intermediate product during *"menu-2"*.

Details / Rules of use

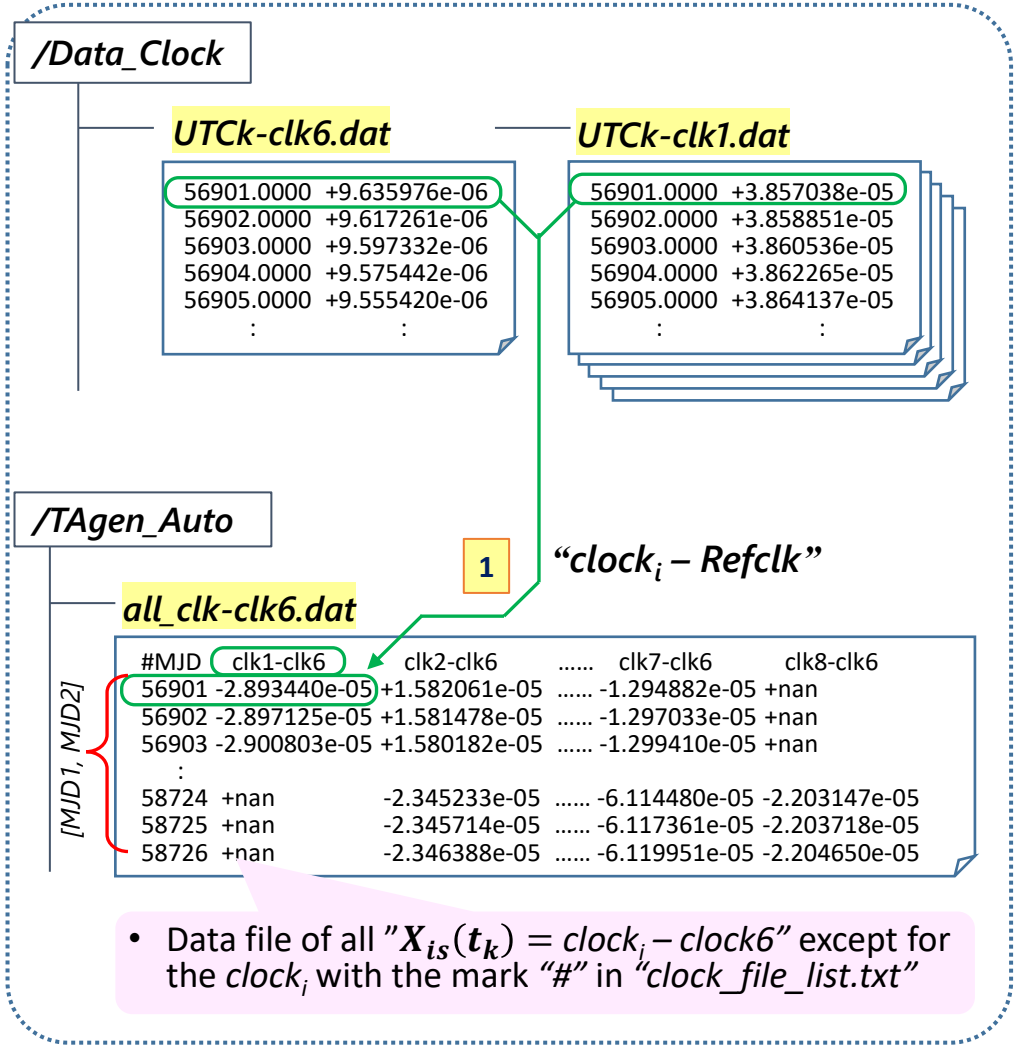
- ***Refclk*** is assigned by the ***flg1*** in *"**clock_file_list.txt**"*.

- 1st column : observation time stamp
- 2nd column : measured time difference of *"**clock$_i$ – Refclk**"* 1

:

Note

※ The clock with "#" in *"**clock_file_list.txt**"* is excluded. *(p.28)*



A-1.2.5 *"(MJD)_TAxhat.dat"*

■ *"/Pytool_TAgen/TAgen_Auto/TA_GEN/(time)/(MJD)_TAxhat.dat"*

Outline

- *"(MJD)_TAxhat.dat"* is the intermediate product for checking the TAp calculation. The main result is $\hat{x}_i(t_k)$.
- These data are intermediate products during *"menu-2"* : one file is for one day result.

Details / Rules of use 1

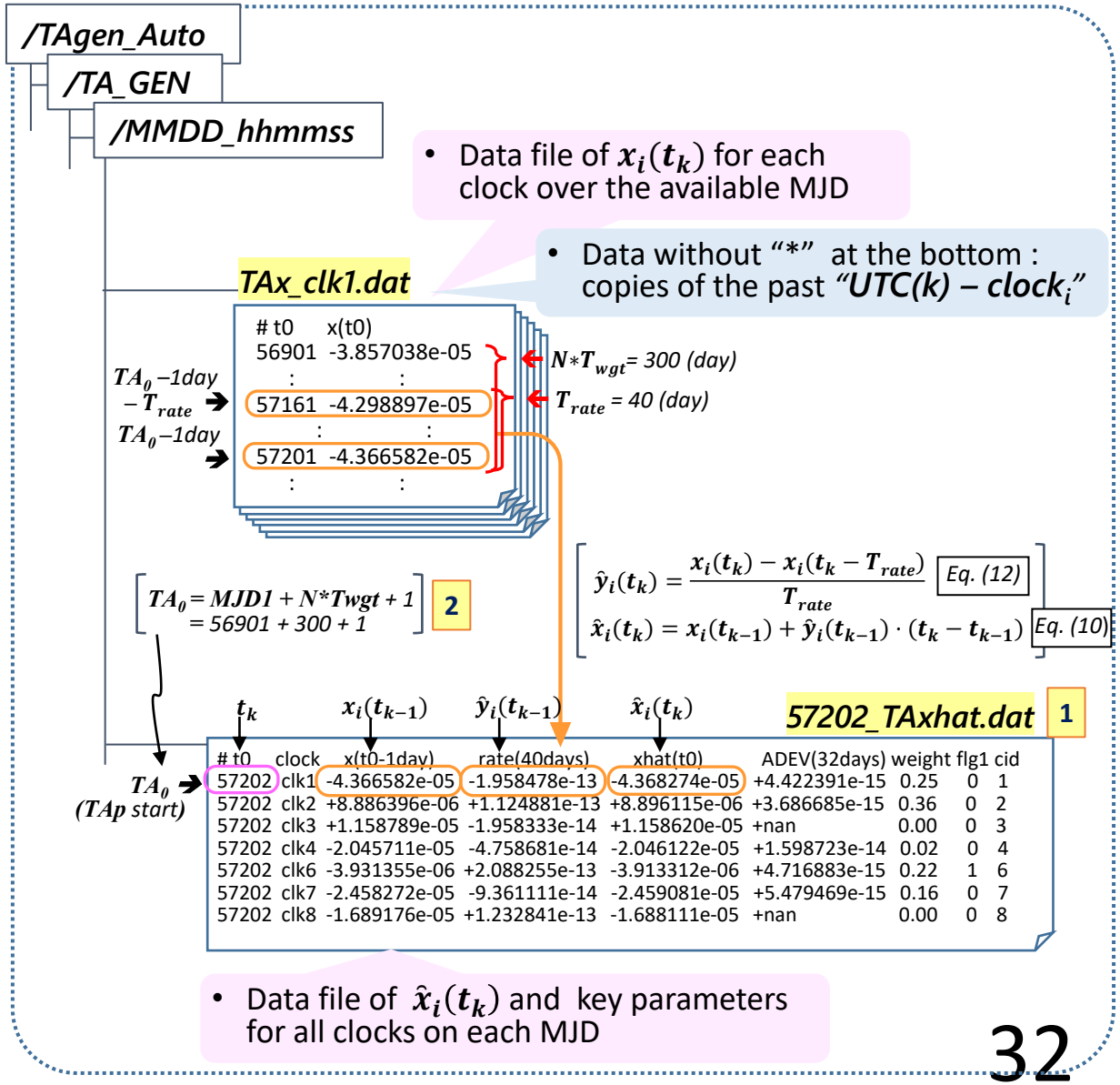
- 1st column : t_k: MJD to determine TAp
- 2nd column : clock name
- 3rd column : $x_i(t_{k-1})$: x_i on the previous day of t_k
- 4th column : $\hat{y}_i(t_{k-1})$ *(Eq. (12) on p.12)*
- 5th column : $\hat{x}_i(t_k)$ *(Eq. (10), on p.12)*
- 6th column : normalized clock weight
- 7th column : $flg1$
- 8th column : clock ID

Note

※ T_{rate} is given from the console at *"menu-2"*.

※ The TAp start date is determined by *"$TA_0 = MJD1+T+1$". (p.20)* 2

※ $MJD1$ is the oldest *MJD* when the available clock data is provided, or is specified by *"menu-1"*.



A-1.2.6 "*TAx_(clock$_i$).dat*"

TAgen_auto.py

■ "*/Pytool_TAgen/TAgen_Pred/TA_GEN/(time)/**TAx_(clock$_i$).dat***"

Outline

- "***TAx_(clock$_i$).dat***" is the time series of x_i of *clock$_i$* which is the targeted product of ***TAp*** calculation.

Details / Rules of use

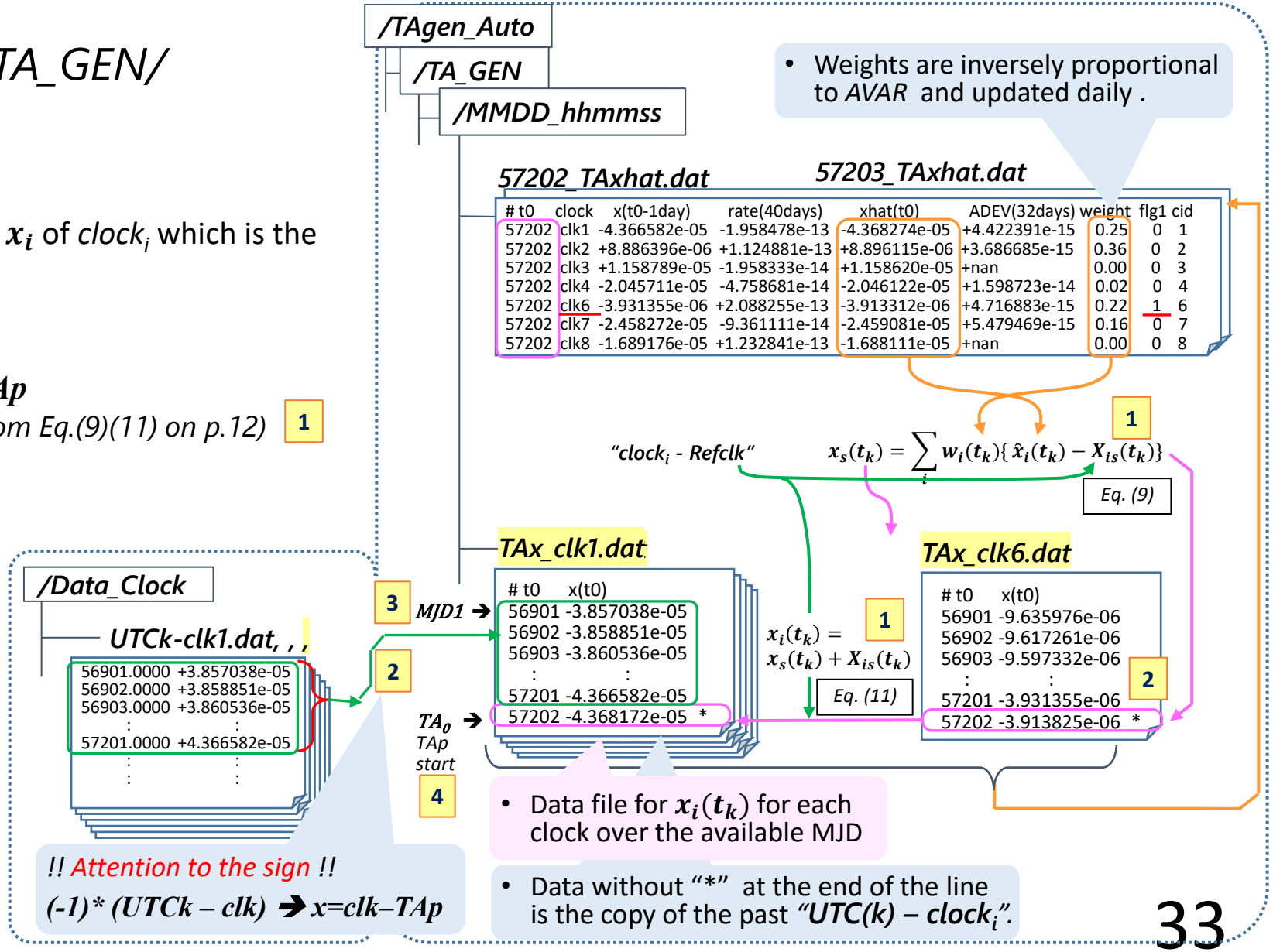
- 1st column : t_k: MJD to determine ***TAp***
- 2nd column : $x_i(t_k)$ of *clock$_i$ (derived from Eq.(9)(11) on p.12)* 1

Note

※ Data without "*******" at the bottom of the line are the copies of the past "***UTC(k) – clock$_i$***". 2

※ The first date ***MJD1*** is the oldest date when the available clock data is provided, or is specified by 3 "*menu-1*".

※ The ***TAp*** start date is determined by "$TA_0 = MJD1 + T + 1$". *(p.20)* 4



A-1. Input / Output files :

A-1.3. Output files of results

A-1.3.1 *"all_raw_clk_(time).dat"*

■ *"/Pytool_TAgen/TAgen_Auto/TA_RESULT/ (time)/**all_raw_clk_(time).dat**"*

Outline

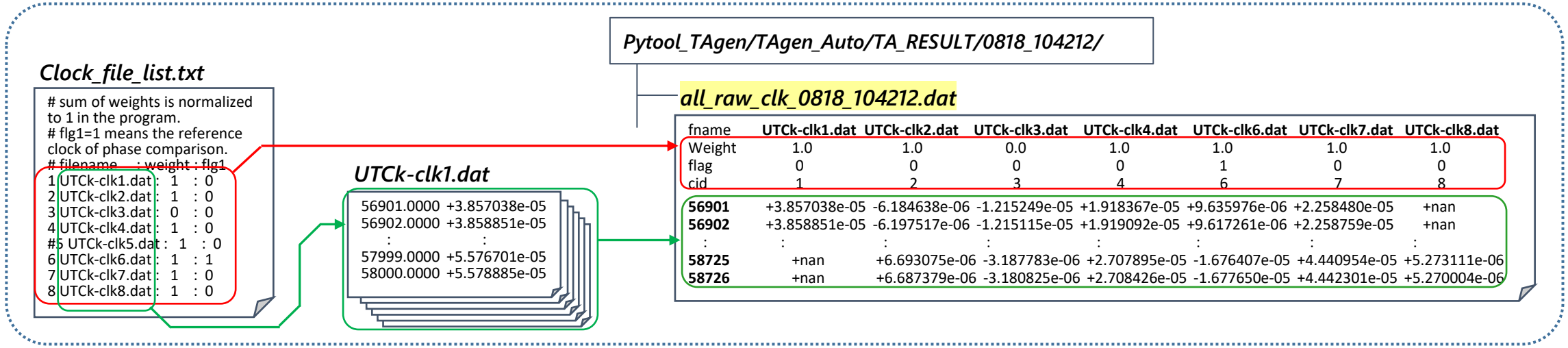
- *"**all_raw_clk_(time).dat**"* is the combined data of '***UTC(k) – clock$_i$***', which is used for the left plot of *"**fig_(time).pdf**"* (*p.40*)
- This data file is generated at the end of *Initial processing* on demand.

Details / Rules of use

- 1st line : clock file name
- 2nd line : *weight* : copy of the 3rd column of *"clock_file_list.txt"*
- 3rd line : *flag* : copy of the 4th column of *"clock_file_list.txt"*
- 4th line : *cid* : copy of the 1st column of *"clock_file_list.txt"*
- 5th line ~ : copy of the "***UTC(k) – clock$_i$***" of "***UTCk_(clock$_i$).dat***" (*p.30*)
- *(time)* : ***MMDD_hhmmss*** : time stamp of data saving
  (***MMDD***: month & date, ***hhmmss*** : hour & minute & second)

Note

※ The clock with "**#**" in "***clock_file_list.txt***" (*p.28*) is excluded.



A-1.3.2 *"all_raw_clk_TA_(time).dat"*

TAgen_auto.py

■ *"/Pytool_TAgen/TAgen_Auto/TA_RESULT/ (time)/**all_raw_clk_TA_(time).dat**"*

Outline

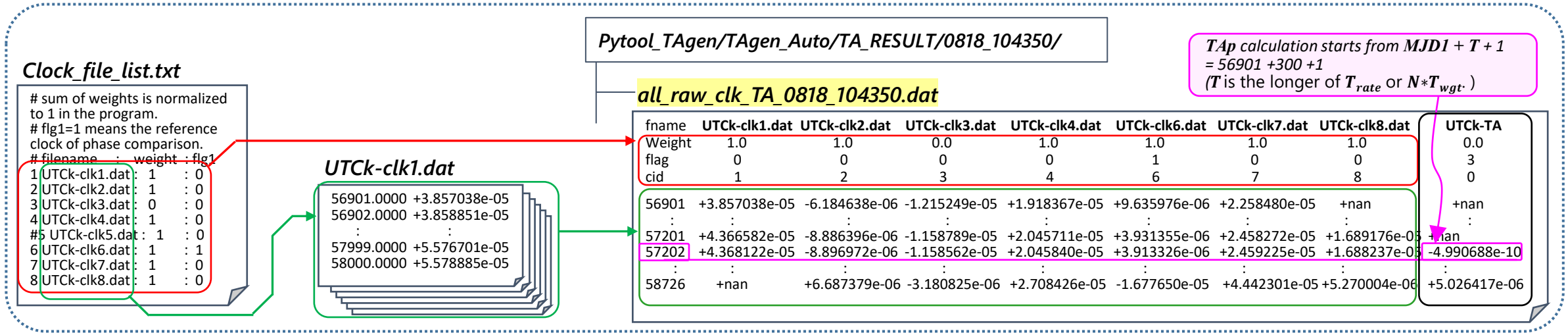
- *"**all_raw_clk_TA_(time).dat**"* is the combined data of *'**UTC(k) – clock$_i$**'* and *'**UTC(k) – TAp**'*.
- This data file is generated at the end of *"menu-2"* on demand.

Details / Rules of use

- The rules except for the last column data are the same as *"**all_raw_clk_(time).dat**"*. *(p.35)*
- The last column:
  - The values of *weight, flag, cid* are not used in the program.
  - ***TAp*** calculation starts from ***MJD1 + T + 1day.***
    (***T*** is the longer of T_{rate} or $N*T_{wgt}$.) *(p.20)*

Note

※ The clock with *"#"* in *"**clock_file_list.txt**"* *(p.28)* is excluded.



A-1.3.3 *"all_adev_(time).dat"*

TAgen_auto.py

■ *"/Pytool_TAgen/TAgen_Auto/TA_RESULT/ (time)/**all_adev_(time).dat**"*

Outline

- *"**all_adev_(time).dat**" is the Allan deviation of the data of "**all_raw_clk_(time).dat**" or "**all_raw_clk_TA_(time).dat**".*
- This data file is generated at the end of *Initial processing* or *"menu-2"* on demand.

Details / Rules of use

- 1st line : clock file name
- 2nd line : ***MJDspan*** : the period used in the *ADEV* calculation. 1
- 3rd line ~ : Allan deviation calculated by by "***AllanTools***" *[Ref. 2]*.
- *(time)* : ***MMDD_hhmmss*** : time stamp of data saving
  (***MMDD*** : month & date, ***hhmmss*** : hour & minute & second)

Note

※ If there are not sufficient samples, *ADEV* cannot be calculated and is expressed as ***"nan"***. 2

※ If the data is outlier (> ***x_limit, "nan"***), the data sequence is divided using the outlier as boundary. 3

※ If the data has discontinuous and divided into several blocks, the *ADEV* is plotted for each block and marked with ***"*1", "*2"***,,, . 4

*("UTCk_clk2.dat" shows the ADEV of [block-0], and "UTCk_clk2.dat*1" shows the ADEV of [block-1].)*

Pytool_TAgen/TAgen_Auto/TA_RESULT/0818_104350/

all_raw_clk_TA_0818_104350.dat

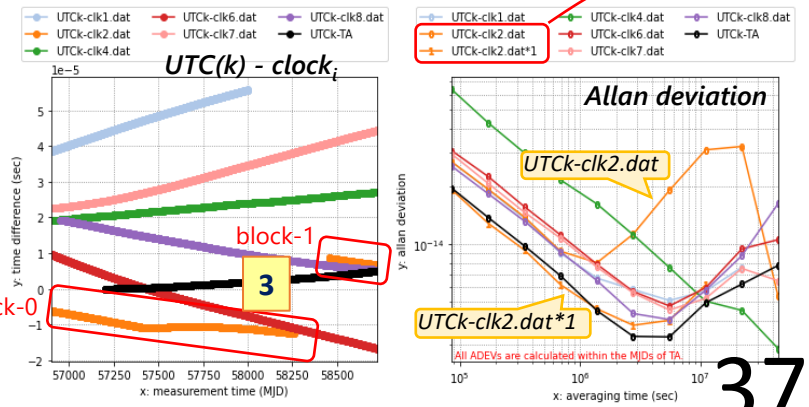
all_adev_0818_104350.dat

1

Tau(s)	UTCk-clk1.dat	UTCk-clk2.dat	UTCk-clk2.dat*1	UTCk-clk3.dat	UTCk-clk4.dat	UTCk-clk6.dat	UTCk-clk8.dat	UTCk-TA
MJDspan	57202-58000	57202-58260	58457-58726	57202-58726	57202-58726	57202-58726	57202-58726	57202-58726
86400	+2.671314e-14	+2.672821e-14	+1.914455e-14	+6.349398e-14	+3.056365e-14	+2.933094e-14	+2.537478e-14	+1.950581e-14
172800	+1.919744e-14	+1.931040e-14	+1.284166e-14	+4.272830e-14	+2.246428e-14	+2.070340e-14	+1.829287e-14	+1.375601e-14
:	:	:	:	:	:	:	:	:
11059200	+5.704858e-15	+3.093775e-14	+6.159347e-15	+5.057043e-15	+5.936237e-15	+5.225535e-15	+5.760070e-15	+4.961455e-15
22118400	+7.589139e-15	+3.222987e-14	+nan	+4.528442e-15	+9.449766e-15	+7.483117e-15	+8.720073e-15	+6.200905e-15
44236800	+nan	+5.349799e-15	+nan	+2.866345e-15	+1.056197e-14	+6.455195e-15	+1.625726e-14	+7.754905e-15

4 2

fig_0818_104350.pdf 4



A-1.3.4 *"all_weight_(time).dat"*

■ *"/Pytool_TAgen/TAgen_Auto/TA_RESULT/ (time)/**all_weight_(time).dat**"*

Outline

- *"**all_weight_(time).dat**"* is the list of clock weights actually used in the $\boldsymbol{TAp}$ calculation.
- The clock weight is inversely proportional to the Allan Variance $\boldsymbol{\sigma_y^2(T_{wgt})}$, and is updated daily.

$$\boldsymbol{w_i(t_k) \propto 1/\sigma_y^2(T_{wgt})} \quad \text{here} \quad \boldsymbol{\sum_i \ w_i = 1}.$$

- $\boldsymbol{\sigma_y(T_{wgt})}$ is calculated from the past latest $\boldsymbol{x_i}$ values within the *"*$\boldsymbol{N * T_{wgt}}$*"* range using the Python library *"**AllanTools**" [Ref. 2]*.
- The detail of the weight calculation is shown in "4.2.3. [6.2][6.3]". *(pp.16-18)*
- This data file is generated at the end of "*menu-2*" on demand.

Note

※ The reason of zero weight:
1 Initial setting, 2 Missing of clock data, and 3 Anomaly of clock data.

Details / Rules of use

- 1st line : clock name
- 2nd line ~ : clock weights actually used in the $\boldsymbol{TAp}$ calculation
- *(time)* : $\boldsymbol{MMDD\_hhmmss}$: time stamp of data saving
  ($\boldsymbol{MMDD}$: month & date $\boldsymbol{hhmmss}$: hour & minute & second)

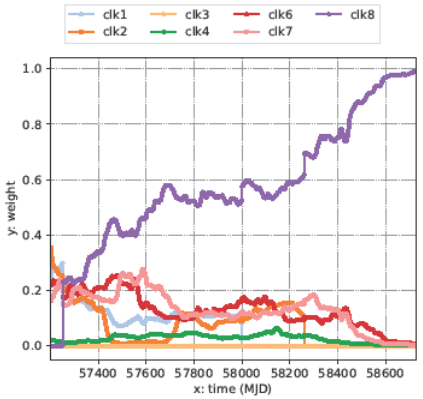
Pytool_TAgen/TAgen_Auto/TA_RESULT/0818_104350/

all_weight_0818_104350.dat

# MJD	clk1	clk2	clk3	clk4	clk6	clk7	clk8
57202.0	0.247	0.356	0.000	0.019	0.217	0.161	0.000
:	:	:	:	:	:	:	:
57251.0	0.299	0.254	0.000	0.017	0.227	0.203	0.000
57252.0	0.231	0.196	0.000	0.013	0.175	0.154	0.231
:	:	:	:	:	:	:	:
58000.0	0.084	0.088	0.000	0.027	0.159	0.115	0.526
58001.0	0.000	0.097	0.000	0.030	0.173	0.125	0.575
:	:	:	:	:	:	:	:
58260.0	0.000	0.107	0.000	0.034	0.092	0.148	0.619
58261.0	0.000	0.000	0.000	0.038	0.103	0.165	0.694
:	:	:	:	:	:	:	:
58726.0	0.000	0.000	0.000	0.001	0.004	0.003	0.992

2 (clk8) 2 3 1

wgt_fig_0818_104350.pdf



A-1.3.5 *"log_(time).txt"*

■ *"/Pytool_TAgen/TAgen_Auto/TA_RESULT/ (time)/**log_(time).txt**"*

Outline

- *"**log_(time).dat**"* is the history of operation process.
- This file is generated at the end of *initial processing* or *"menu-2"* on demand.

Details / Rules of use

- Operated menu processes are serially added to this log file.
- The recorded parameters are actually used in the calculation.
- *(time)* : ***MMDD_hhmmss*** : time stamp of data saving
  (***MMDD*** : month & date ***hhmmss*** : hour & minute & second)

Note

※The serially recording is reset when *"menu-0"* is selected.

Pytool_TAgen/TAgen_Auto/TA_RESULT/0818_104350/

log_0818_104350.txt

```
# Data log at 2025.08.18_10h43m_50s
#
**0** Read and plot initial clock data and Allan deviation*****
::::: Input clocklist   : clock_file_list.txt
::::: Input clock file  : ['UTCk-clk1.dat', 'UTCk-clk2.dat', 'UTCk-clk3.dat', 'UTCk-clk4.dat', 'UTCk-clk6.dat', 'UTCk-clk7.dat', 'UTCk-clk8.dat']
::::: data time interval (t_intval): 86400(sec)
::::: Outlier limit (x_limit))     : 1000.0

**2** make a timescale with prediction *****
::::: saved TAxhat data : ./TA_GEN/0818_104218/(MJD)_TAxhat.dat
::::: saved TAx data : ./TA_GEN/0818_104218/TAx_(clock).dat
::::: interval for rate estimation (TArate_span)      : 40(days)
::::: ADEV tau for clock weight (wgt_adev_tau0)   : 32(days)
::::: max weight limit (wgt_lmt)                  : OFF
::::: sample numbers for ADEV for weighting (Twgt_N)) : 10
```

A-1.3.6 *"fig_(time).pdf"*

■ *"/Pytool_TAgen/TAgen_Auto/TA_RESULT/ (time)/**fig_(time).pdf**"*

Outline

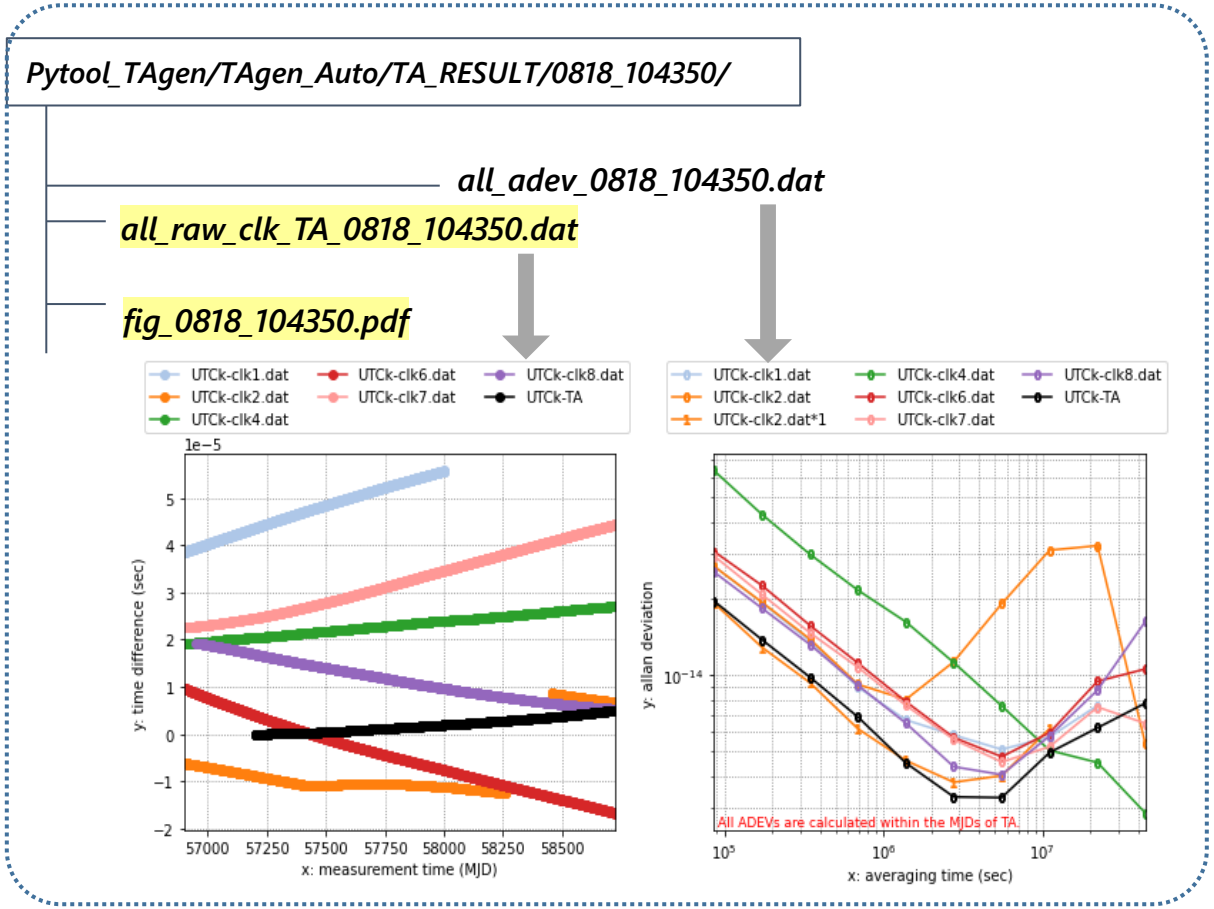
- *"**fig_(time).pdf**" is the plots of time differences vs **UTC(k)** and their Allan deviations.*
- *This figure file is generated at the end of Initial processing or "menu-2".*

Details / Rules of use

- Left figure :
  - time differences of ***clock*$_i$** or ***TAp*** vs ***UTC(k)***
  - ***"all_raw clk_(time).dat"*** or ***"all_raw clk_TA_(time).dat"*** is used for this plot. *(p.35 or 36)*
- Right figure :
  - Allan deviations of the data for the left plot
  - ***"all_adev (time).dat"*** is used for this plot. *(p.37)*
- *(time)* : ***MMDD_hhmmss*** : time stamp of data saving
  (***MMDD*** : month & date ***hhmmss*** : hour & minute & second)

Note

- Only the clocks with non-zero weights are plotted.
- The black line shows the results of ***TAp***.



A-1.3.7 *"wgt_fig_(time).pdf"*

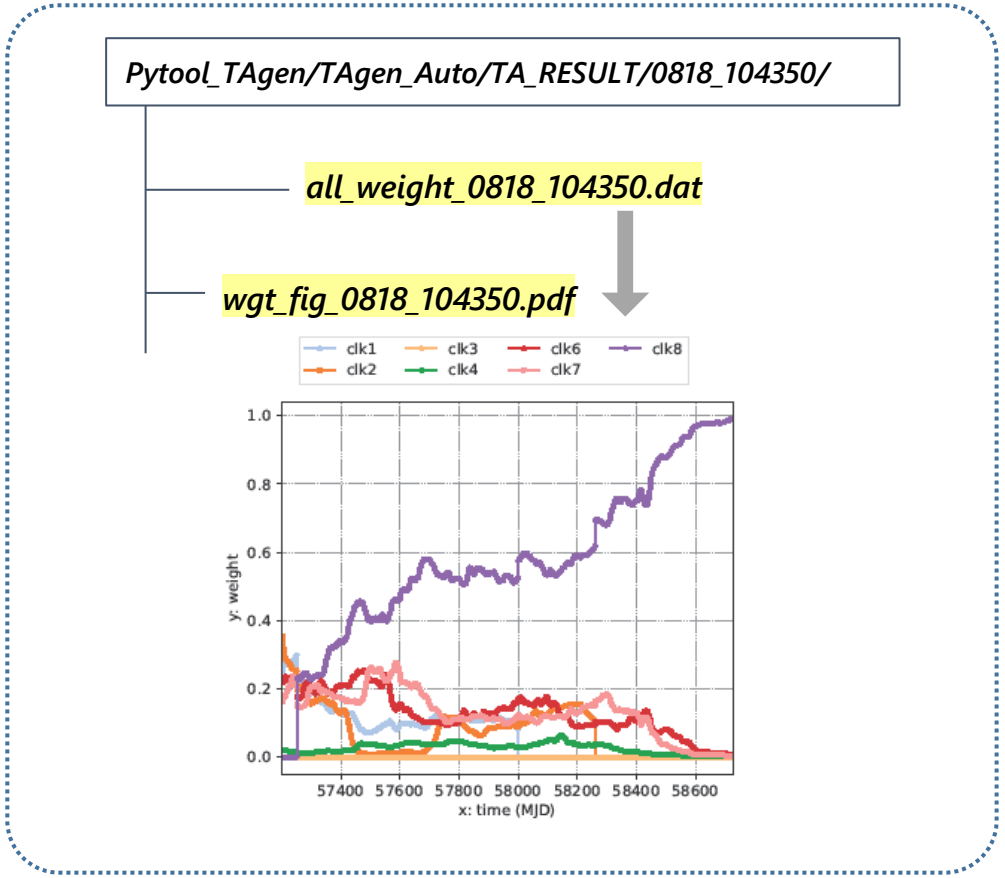
■ *"/Pytool_TAgen/TAgen_Auto/TA_RESULT/ (time)/**wgt_fig_(time).pdf**"*

Outline

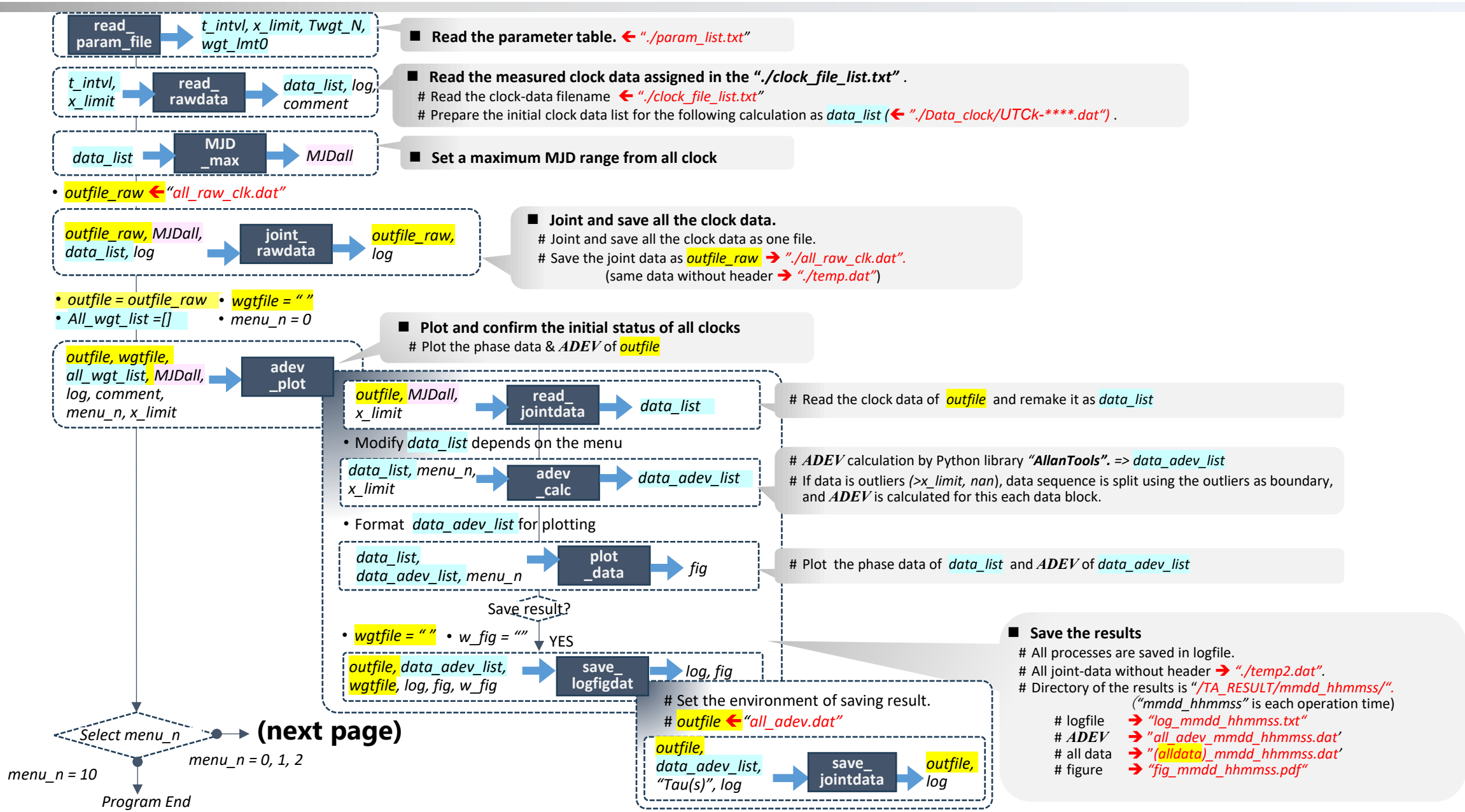
- *"**wgt_fig_(time).pdf**"* is the plots of actually used clock weights.
- This figure file is generated at the end of *"menu-2"*.

Details / Rules of use

- *"**all_weight_(time).dat**"* is used for this plot. *(p.38)*
- *(time)* : ***MMDD_hhmmss*** : time stamp of data saving
  (***MMDD*** : month & date ***hhmmss*** : hour & minute & second)



A-2. Construction and flow of the program


read_ param_file
t_intvl, x_limit, Twgt_N, wgt_lmt0
■ Read the parameter table. ← "./param_list.txt"
t_intvl, x_limit
read_ rawdata
data_list, log, comment
■ Read the measured clock data assigned in the "./clock_file_list.txt" .
Read the clock-data filename ← "./clock_file_list.txt"
Prepare the initial clock data list for the following calculation as data_list (← "./Data_clock/UTCk-****.dat") .
data_list
MJD _max
MJDall
■ Set a maximum MJD range from all clock
• outfile_raw ← "all_raw_clk.dat"
outfile_raw, MJDall, data_list, log
joint_ rawdata
outfile_raw, log
■ Joint and save all the clock data.
Joint and save all the clock data as one file.
Save the joint data as outfile_raw → "./all_raw_clk.dat".
(same data without header → "./temp.dat")
• outfile = outfile_raw
• wgtfile = " "
• All_wgt_list =[]
• menu_n = 0
■ Plot and confirm the initial status of all clocks
Plot the phase data & ADEV of outfile
outfile, wgtfile, all_wgt_list, MJDall, log, comment, menu_n, x_limit
adev _plot
outfile, MJDall, x_limit
read_ jointdata
data_list
Read the clock data of outfile and remake it as data_list
• Modify data_list depends on the menu
data_list, menu_n, x_limit
adev _calc
data_adev_list
ADEV calculation by Python library "AllanTools". => data_adev_list
If data is outliers (>x_limit, nan), data sequence is split using the outliers as boundary, and ADEV is calculated for this each data block.
• Format data_adev_list for plotting
data_list, data_adev_list, menu_n
plot _data
fig
Plot the phase data of data_list and ADEV of data_adev_list
Save result?
YES
• wgtfile = " "
• w_fig = ""
outfile, data_adev_list, wgtfile, log, fig, w_fig
save_ logfigdat
log, fig
■ Save the results
All processes are saved in logfile.
All joint-data without header → "./temp2.dat".
Directory of the results is "/TA_RESULT/mmdd_hhmmss/".
("mmdd_hhmmss" is each operation time)
logfile → "log_mmdd_hhmmss.txt"
ADEV → "all_adev_mmdd_hhmmss.dat"
all data → "(alldata)_mmdd_hhmmss.dat"
figure → "fig_mmdd_hhmmss.pdf"
Set the environment of saving result.
outfile ← "all_adev.dat"
outfile, data_adev_list, "Tau(s)", log
save_ jointdata
outfile, log
Select menu_n
(next page)
menu_n = 0, 1, 2
menu_n = 10
Program End

main program

: See the detail on *p.43*

Select menu_n

menu_n = 10

Program End

menu_n = 0

■ **Plot the "*UTC(k)-Clock$_i$*" data**
Reset all the processes so far and plot the initial clock data

- data_list = [ ]

t_intvl, x_limit → read_rawdata → data_list, log, comment
← "./clock_file_list.txt"
← "./Data_clock/UTCk- ****.dat"

data_list → MJD_max → MJDall

outfile_raw, MJDall, data_list, log → joint_rawdata → outfile, log
→ "./all_raw_clk.dat"
→ "./temp.dat"

- wgtfile = " "
- All_wgt_list =[]

outfile, wgtfile, all_wgt_list MJDall, log, comment, menu_n, x_limit → adev_plot
← "all_adev.dat"
→ "./temp2.dat".
→ "log_(time).txt"
→ "all_adev_(time).dat"
→ "(alldata)_(time).dat"
→ "fig_(time).pdf"

menu_n = 1

■ **Assign the MJD range**
Manually set the MJD range for the further processing

Input *MJDall* is the maximum-MJD
MJD1(=min) & *MJD2*(=max) should be within *MJDall*
Outut *MJDall* is [*MJD1*,,,*MJD2*]

MJDall, log → MJD_limit → MJDall, log

outfile,, MJDall, data_list, log → joint_rawdata → outfile, log
→ "./all_raw_clk.dat"
→ "./temp.dat"

- wgtfile = " "
- All_wgt_list =[]

outfile, wgtfile, all_wgt_list MJDall, log, comment, menu_n, x_limit → adev_plot
← "all_adev.dat"
→ "./temp2.dat".
→ "log_(time).txt"
→ "all_adev_(time).dat"
→ "(alldata)_(time).dat"
→ "fig_(time).pdf"

menu_n = 2

(next page)

main program

: See the detail on *p.43*

Select menu_n
- menu_n = 2
- menu_n = 10 → Program End

• *inflie_clklist* = *"./clock_file_list.txt"*

Inflie_clklist → clk_init_wgt → *clk_initparam*

■ **Get initial weight from** *"./clock_file_list.txt"*.
The weight of the reference clock (*flag1*=1) is set to be zero.

• *inflie_raw* ← *"./all_raw_clk.dat"*

infile_raw, *MJDall*, *x_limit* → TA_clk_clk → *diff_list*

■ **Prepare *the TAx$_{i0}$ (= clk_i – reference clock)* from the measurement raw data (*"./all_raw_clk.dat"*)**
The clock with *flag1*=1 is the reference clock (*"clk0"*).
Save the result (*diffdata*) ➔ *"./all_clk-(Refclk).dat"*

• *TAgen_dir* = *"./TA_GEN/"*

TAgen_dir → set_dirpath → *dt_str, dt_com, TAgen_path*

■ **Prepare the directory for data saving** (*"./TA_GEN/"*)

MJDall → TA_set_span_for_Ratewgt → *TArate_span, TAwgt_span*

■ **Prepare the data span for rate prediction**
Any span is OK within *MJDall*.
TArate_span : interval for calculating a slope
TAwgt_span : τ of Allan variance for calculating a weight

wgt_lmt0 → TA_set_weight_limit → *wgt_lmt*

■ **Set the weight_limit??**
If *'y'* : *wgt_lmt* = the value in the *"param_list.txt"*
else : *wgt_lmt* = *"OFF"*

■ **Make TA with daily prediction and fixed clock weights**
Fixed weight is given by *"./clock_file_list.txt*

• *initdate_span = TAwgt_span * Twgt_N*

infile_raw, *MJDall*, *TAgen_path*, *TArate_span*, *initdate_span*, *x_limit* → TA_make_init_x → *TAx_list*, *TA_MJD0*

TAx_list ➔ */ TAx_(clock$_i$).dat"*

■ **Prepare initial bunch of data for starting TA calculation**
initdate_span is necessary range for *AVAR(tau=TAwgt_span)*.
TAx_list : initial data stock for starting TA calculation.
[[clkname, [[MJD],[clk-UTCk)]], wgt, flg, id],,,]
TA_MJD0 : the MJD of starting TA calculation

• ***for MJD in range(TA_MJD0, MJDall[-1]+1):***

TAx_list, *MJD*, *TAwgt_span*, *clk_initparam*, *x_limit*, *Twgt_N*, *wgt_lmt*, *t_intvl* → TA_calc_wgt → *TAx_list*, *wgt_addinfo*

wgt_lmt, *wgt1* → re_weight_under_lmlit → *wgt3*

■ **Calculate the clock weight from the AVAR of past TAx**
The clock with *wgt=0* at *"clk_file_list.txt"* is excluded from the TA calculation.
Weight = 1/AVAR(tau)
tau ~ TAwgt_span (selected from "AllanTools" output)
AVAR is calculated from the past *TAx* during the *TAwgt_span*Twgt_N.*
If *wgt_lmt* is not *"OFF"*, maximum weight is limited as *wgt_lmt.*
Lastly renew the weight values in the former *TAx_list*

TAx_list, *MJD*, *TAgen_path*, *TArate_span* → TA_predict_x → *TAxhat_list*

■ **Calculate the *TAx* prediction (= *xhat*) for next day**
Slope is calculated in the latest span of TA*rate_span*
*xhat = x[i] + slope * (t[i] - t[i-1])*
TAxhat_list : [[clkname, t1, x(t0), slope/86400, xhat(t1), wgt, flg, cid],,,]
(t1 : assigned MJD, *t1 – t0* is 86400 sec)

• ***(cont. for loop:)***

TAx_list, *TAxhat_list*, *diff_list*, *MJD*, *TAgen_path*, *TArate_span*, *TAwgt_span*, *wgt_addinfo*, *x_limit*, *Twgt_N*, *wgt_lmt* → TA_calc_x → *TAx_list*, *wgt_list*, *comment*, *log_TA*

wgt_lmt, *wgt1* → re_weight_under_lmlit → *wgt3*

■ **Calculate and save the *TAx***
Sum the available weights and do their normalization
TA_x0 = Sum of *{wgt_n * (TA_xhat - df_x)} for* each clock
* *TA_x0* : *TAx* of reference clock
* *TA_xhat* : predicted *TAx* of each clock
* *df_x* : measured time diff (each clock - reference clock)
* *wgt_n* : normalized clock weight
# *TA_x = df_x + TA_x0*
* *TA_x* : *TAx* of each clock
# *TAx_list_new* : *[[clkname, [[MJD], [TAx]], [wgt_n], flg, id],,,]*
➔ rename as *TAx_list*
wgt_list : *[[clkname, wgt_n, cid],,,]*
Directory of the results is *"/TA_GEN/mmdd_hhmmss/".*

TAx_list ➔ */ Tax_(clock$_i$).dat"*
TAxhat_list ➔ */ (MJD)_TAxhat.dat"*

• *all_wgt_list = [[MJD, wgt_list],,,]*

■ **Save weight**
wgtfile = *"all_weight.dat"*

all_wgt_list → save_weight → *wgtfile* ➔ *"./all_weight.dat"*

infile_raw, *TAx_list*, *comment*, *TA_MJD0*, *x_limit* → TA_ref_TA → *UTCk_TA_list*, *comment*

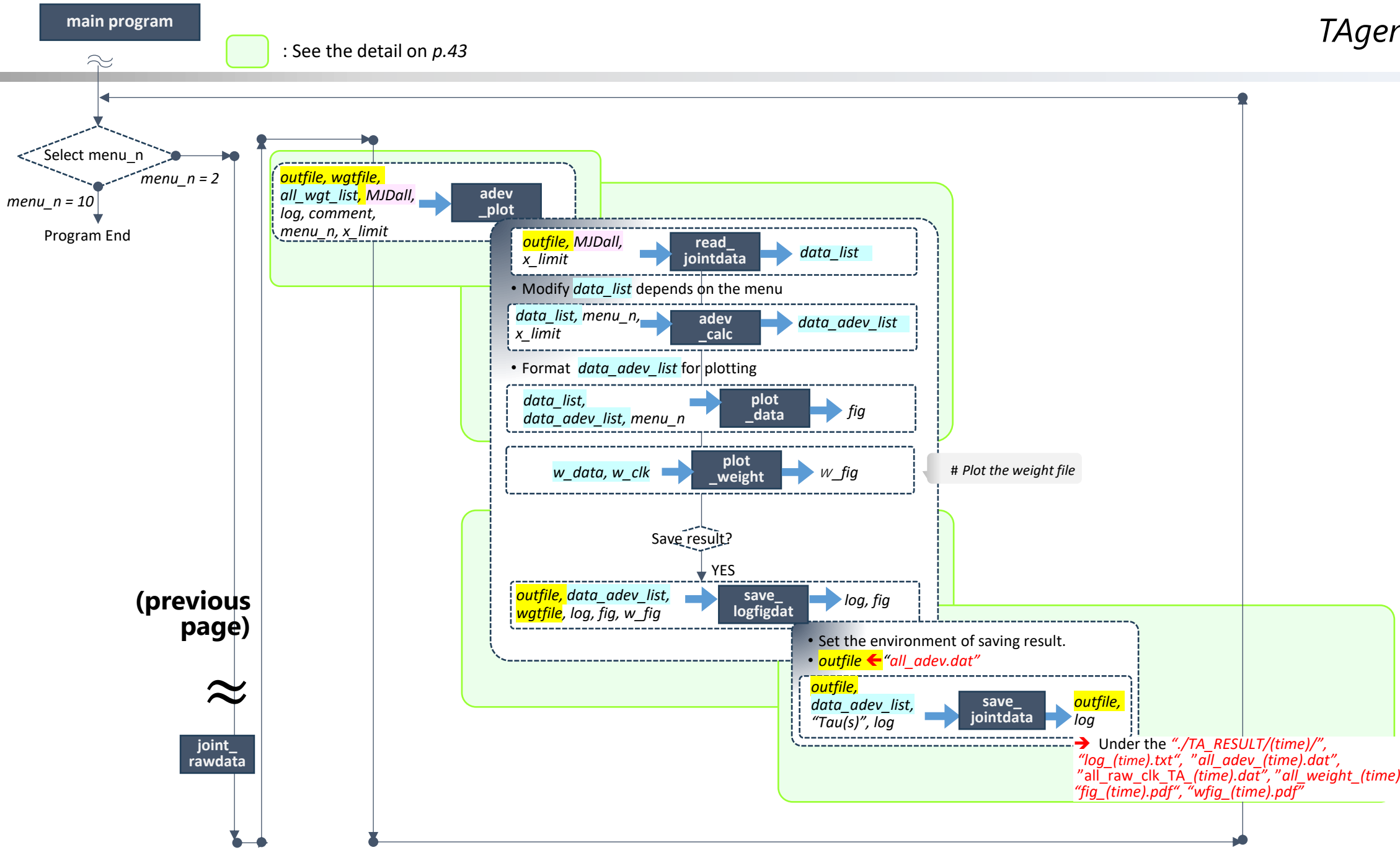
■ **Get the (UTCk -TA)**
"UTCk-TA" = "clk0-TA"(from *TAx_list*) +*"UTCk-clk0"*(from *"infile_raw"*)
UTCk_TA_list : appended the *["UTCk-TA", data, "0.0", "3", "0"]*

• *outfile* = *"./all_raw_clk_TA.dat"*

• *log = log + log_TA*

outfile, *MJDall*, *UTCk_TA_list*, *log* → joint_rawdata → *outfile*, *log* ➔ *"./all_raw_clk_TA.dat"* ➔ *"./temp.dat"*

(next page)


main program
: See the detail on p.43
Select menu_n
menu_n = 2
menu_n = 10
Program End
outfile, wgtfile, all_wgt_list, MJDall, log, comment, menu_n, x_limit
adev _plot
outfile, MJDall, x_limit
read_ jointdata
data_list
• Modify data_list depends on the menu
data_list, menu_n, x_limit
adev _calc
data_adev_list
• Format data_adev_list for plotting
data_list, data_adev_list, menu_n
plot _data
fig
w_data, w_clk
plot _weight
w_fig
Plot the weight file
Save result?
YES
outfile, data_adev_list, wgtfile, log, fig, w_fig
save_ logfigdat
log, fig
• Set the environment of saving result.
• outfile ← "all_adev.dat"
outfile, data_adev_list, "Tau(s)", log
save_ jointdata
outfile, log
➔ Under the "./TA_RESULT/(time)/", "log_(time).txt", "all_adev_(time).dat", "all_raw_clk_TA_(time).dat", "all_weight_(time).dat" "fig_(time).pdf", "wfig_(time).pdf"
(previous page)
joint_ rawdata