



OPIS ZAMÓWIENIA:

Zamówienie dotyczy uruchomienia na istniejącym już, dedykowanym serwerze bazy danych i katalogu o nazwie roboczej Toruń Set of Eclipsing Binaries with an Open On-Line Archive (TSEBOOLA), który ma być zbiorem informacji na temat obiektów (gwiazdowych układów podwójnych) badanych w ramach projektu naukowego, wraz z zestawem usług obliczeniowych, obsługiwanych z poziomu przeglądarki. W tym celu sieć komputerowa Centrum Astronomicznego im. M. Kopernika (CAMK) PAN w Toruniu została wyposażona w dedykowany serwer.

1. Baza danych

Baza TSEBOOLA ma przede wszystkim zawierać informacje o konkretnych układach podwójnych i być stworzona w SQL. Oprócz tego, powinna istnieć osobna baza danych użytkowników TSEBOOLA i powiązanych serwisów, stworzona w Mongo. Baza TSEBOOLA ma być widoczna publicznie tylko do odczytu, natomiast zarejestrowani i zalogowani użytkownicy mają mieć możliwość dodawania i edycji rekordów oraz korzystania z usług obliczeniowych. Przewidziane są trzy poziomy: dostęp otwarty dla wszystkich bez logowania (all), dostęp dla wszystkich zarejestrowanych użytkowników (users), dostęp priorytetowy dla jednego lub kilku wybranych użytkowników (owner, group).

Dla każdego obiektu podane mają być:

- Podstawowe informacje literaturowe dostępne ze źródeł kwerendowalnych (nazwa, położenie na niebie, jasności, itp). Nie muszą być zapisywane w bazie.
- Jedna lub więcej krzywych blasku i prędkości radialnych, tzn. tabele z następującymi kolumnami: czas pomiaru, wartość pomiaru, błąd pomiaru, źródło pomiaru. Tabele mają być przechowywane na lokalnym serwerze i dostępne do ściągnięcia przez użytkownika w formie tekstowej (plain text ascii).
- Model układu, będący zbiorem kilkunastu lub kilkudziesięciu parametrów fizycznych (w postaci liczb) wraz z ich niepewnościami (błędami).
- Modelowe krzywe blasku i prędkości radialnych, tzn. tabele z następującymi kolumnami: czas pomiaru, wartość modelowa, różnica między wartością zmierzoną a modelową. Tabele mają być przechowywane na lokalnym serwerze i dostępne do ściągnięcia w formie tekstowej (plain text ascii).
- Graficzna, interaktywna reprezentacja krzywych blasku i prędkości radialnych w postaci wykresów, z możliwością zbliżania/oddalania, zapisu do pliku graficznego (.png) i odczytu wartości przypisanych danemu punktowi (czas, pomiar, błąd) np. poprzez najechanie kursorem.
- Zestaw obserwacji spektroskopowych w postaci plików .fits, dostępnych do ściągnięcia.
- Odnośniki do odpowiedniej publikacji, jeśli dane źródło zostało opisane w recenzowanym czasopiśmie.

Przykładowy, prosty schemat strony startowej:

<https://app.moqups.com/1Fa1ff6Ue7/view/page/ad64222d5>

Przykład strony dla pojedynczego obiektu:

<https://app.moqups.com/1Fa1ff6Ue7/view/page/a493423e7>

Do stworzenia interfejsu preferowane jest wykorzystanie Angular, Vue lub React. TSEBOOLA powinna wykorzystywać tylko i wyłącznie infrastrukturę komputerową sieci CAMK Toruń, w szczególności bez korzystania z chmur.

Przekazanie wstępnej wersji do użytku wewnętrznego i testów powinno nastąpić nie później niż koniec lipca 2020. Publiczne udostępnienie bazy danych gotowej do edycji, wraz z usługą rejestracji/logowania i zabezpieczeniami powinno nastąpić nie później niż koniec października 2020.

2. Serwisy obliczeniowe

Dostępne serwisy obliczeniowe mają mieć postać przyjaznych interface'ów graficznych, z możliwością obsługi w oknie przeglądarki (MS Edge, Firefox, Chrome, Safari), komunikujących się z gotowymi już programami (tzn. tworzenie i wysyłanie plików startowych dla tych programów, uruchamianie programów, wyświetlanie wyników w oknie przeglądarki i możliwość ich zapisu). Interfejsy powinny być przygotowane w Angular, Vue lub React, w podobnym stylu graficznym jak UI bazy danych. Same obliczenia mogą być prowadzone nie na serwerze, a na innej maszynie podłączonej fizycznie do tej samej sieci, pracującej na systemie operacyjnym Linux. Serwisy powinny pracować w oparciu o środowisko Jupyter notebook i język Python. Wszystkie serwisy obliczeniowe powinny wykorzystywać tylko i wyłącznie infrastrukturę komputerową sieci CAMK Toruń, w szczególności bez korzystania z chmur.

W ramach dostępnych usług obliczeniowych, TSEBOOLA powinna oferować przynajmniej dwa z niżej wymienionych serwisów:

V2FIT

Dopasowywanie krzywych prędkości radialnych do danych wysłanych przez użytkownika (w postaci tabeli: czas, wartość, błąd) lub pobranych z głównej bazy danych. Preferowane jest wykorzystanie istniejącego kodu V2FIT (język Fortran 77). Kod ten nie jest dostępny publicznie. Serwis ma mieć interfejs z możliwością podania danych i ustawienia odpowiednich opcji, a następnie ma tworzyć odpowiednie pliki konfiguracyjne i wsadowe i uruchamiać program. Kod zwraca wartości dopasowanych parametrów z błędami (również do pliku tekstowego), oraz dwa rodzaje plików wyjściowych (tekstowe). Serwis ma wczytywać wyniki i wyświetlać je w interfejsie jako wartości liczbowe, tabele i wykres.

Przykładowy interfejs programu:

<https://app.moqups.com/1Fa1ff6Ue7/view/page/ab7491083>

Przykłady plików wsadowych (konfiguracyjny i przykładowe dane) oraz wyjściowych (zwracane parametry, dane i krzywe modelowe) stanowią załączniki 1 do 5 tego dokumentu.

JKTEBOP

Dopasowanie krzywych blasku do danych wysłanych przez użytkownika (w postaci tabeli: czas, wartość, błąd) lub pobranych z głównej bazy danych. Preferowane wykorzystanie istniejącego kodu JKTEBOP¹ (Fortran 77). Kod jest dostępny publicznie wraz z przykładowymi plikami wsadowymi, konfiguracyjnymi i wyjściowymi². Serwis ma mieć interfejs z możliwością podania danych i ustawienia odpowiednich opcji, a następnie ma tworzyć odpowiednie pliki konfiguracyjne i wsadowe i uruchamiać program. Po zakończeniu, serwis ma wczytywać wyniki i wyświetlać je w

1 <https://www.astro.keele.ac.uk/~jkt/codes/jktebop.html>

2 <https://www.astro.keele.ac.uk/~jkt/codes/jktebop-v34.tgz>

interfejsie jako wartości liczbowe, tabele i wykres. Interfejs może być zbliżony do przykładu pokazanego dla V2FIT.

iSpec

Analiza widm gwiazdowych wysłanych przez użytkownika (w postaci tabeli: długość fali, strumień; lub w postaci plików .fits). Preferowane wykorzystanie istniejącego kodu iSpec (Python). Kod jest dostępny publicznie³ i posiada zarówno GUI jak i scripter. Interfejs, który ma być przygotowany, może opierać się na już istniejącym GUI⁴. Kod iSpec jest rozbudowanym narzędziem, z wieloma opcjami i funkcjonalnościami. Dlatego, w razie wyboru tego serwisu do zaprogramowania, nie będzie wymagane jego pełne uruchomienie, jedynie wybranych części (do uzgodnienia) pozwalających na wykorzystanie podstawowych funkcji i możliwości programu.

RV calculator

Wyznaczanie prędkości radialnych (RV) gwiazd przy wykorzystaniu przynajmniej jednej z następujących metod matematycznych:

- jednowymiarowa funkcja korelacji wzajemnej (1D CCF)
<https://ui.adsabs.harvard.edu/abs/1979AJ....84.1511T>
- TODCOR
<https://ui.adsabs.harvard.edu/abs/1994ApJ...420..806Z>
Program wykorzystujący tę metodę już istnieje (Fortran 77) i jest obecnie używany. W przypadku wyboru tej metody preferowane jest wykorzystanie tego właśnie kodu.
- funkcja poszerzenia (BF)
<https://ui.adsabs.harvard.edu/abs/2002AJ....124.1746R>
- *Improved Least-Squares Deconvolution* (iLSD)
<https://ui.adsabs.harvard.edu/abs/2013A%26A...560A..37T>

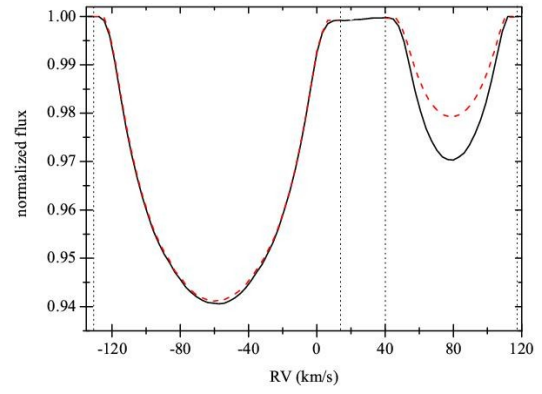
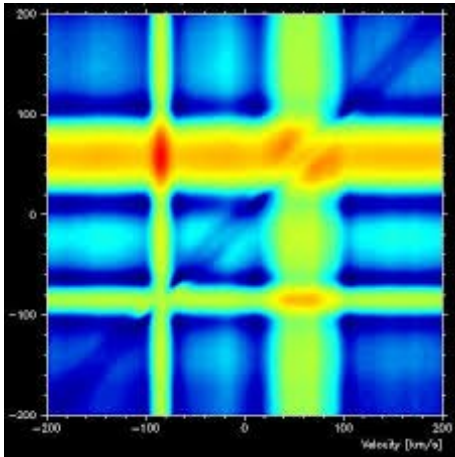
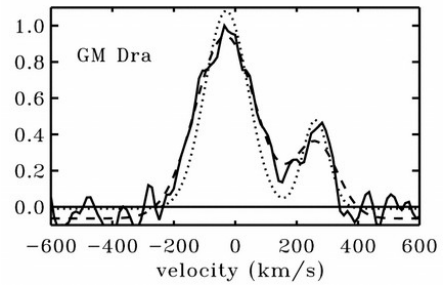
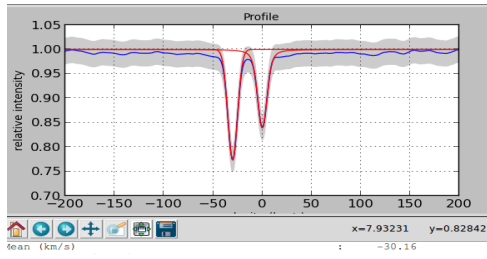
Danymi wejściowymi ma być widmo gwiazdy wysłane przez użytkownika (w postaci tabeli: długość fali, strumień; lub w postaci plików .fits), oraz wybrane widma wzorcowe, przechowywane na lokalnym serwerze (w postaci plików tekstowych lub .fits). Sugerowane jest wykorzystanie istniejącego już kodu (Fortran 77 / Python), ale dozwolone będzie własne zaimplementowanie jednej lub więcej z wymienionych metod. Serwis ma mieć interfejs z możliwością podania danych i ustawienia odpowiednich opcji, a następnie ma tworzyć odpowiedni plik konfiguracyjny i uruchamiać program. Interfejs może przypominać ten pokazany dla V2FIT, ale w zależności od wybranej metody różnice mogą być znaczne. Po zakończeniu obliczeń, serwis ma wczytywać wyniki i wyświetlać je w interfejsie jako wartości liczbowe i wykres.

Przykłady plików konfiguracyjnego i wyjściowych dla metody TODCOR stanowią Załącznik 6-8 tego dokumentu. Graficzna reprezentacja wyników pokazana jest na Rysunku 1.

Uruchomienie wybranych dwóch serwisów obliczeniowych powinno nastąpić nie później niż do końca grudnia 2020 i stanowić będzie zakończenie zadania.

³ <https://www.blancocuaresma.com/s/iSpec>

⁴ <https://www.blancocuaresma.com/s/iSpec/manual/introduction> oraz
http://obswww.unige.ch/~sblancoc/data/iSpec/iSpec_Users_Manual_v20150728.pdf



Rysunek1: Graficzna reprezentacja wyników obliczenia CCF (lewo góra), BF (pravo góra), TODCOR (lew dół) i iLSD (lewo dół).

Załącznik 1

Przykład pliku konfiguracyjnego dla programu V2FIT. Pierwsze linie to ustawienia ogólne (model i nazwy plików z pomiarami, poziom tolerancji dopasowania, forma liczenia niepewności). Za komendą PARAMS zaczynają się wartości startowe szukanych parametrów, w następującym formacie: flaga dopasowania (1 = dopasowywać, 0 = trzymać stałą wartość), ID parametru, wartość startowa, błąd względny. Symbol „#” oznacza linię nie brana pod uwagę przez program.

```
SB1 [input_file_1]
SB2 [input_file_2]
#CIRCPL
TOL 1.0d-10
BSTRAP 1000
#MCARLO 1000
PARAMS
0 PERIOD 6.7286092517d0 2.9d-8
1 TP 4953.5881745d0 14.6d-2
0 ECC 0.0d0 0.0d0
0 OMEGA 0.0d0 0.0
0 SOMDOT 0.0 0.0
0 INC 84.887 0.004d0
1 V01 17.7d0 0.0
1 K1 65.0d0 0.0
0 V02 0.0d0 0.0
1 K2 78.0d0 0.0
0 V11 0.0d0 0.0
0 V12 0.0d0 0.0
0 V21 0.0d0 0.0
0 V22 0.0d0 0.0
#-----if CIRCPL uncommented
#0 PLPER 77.025 0.129
#1 PLK 0.1 0.0
#1 PLTP 5430.0 0.0
#0 PLECC 0.0 0.0
#0 PLOM 0.0 0.0
```

Załącznik 2

Przykład pliku wsadowego z pomiarami prędkości radialnych dla programu V2FIT. Znaczenie kolumn: 1 – czas, 2 – pomiar, 3 – niepewność pomiaru. Kolumny rozseparowane spacjami lub tabulatorami. Symbol „#” oznacza linię nie brana pod uwagę przez program. Komendy „EFAC” i „EQUAD” to modyfikatory niepewności, komenda „JUMP 1 1” dodaje dodatkowy parametr do dopasowania. W interfejsie powinna być możliwość definiowania położenia i wartości tych komend, jak również edycji samych wartości pomiarów i niepewności, wyboru linii do wczytania, dodawania i usuwania linii.

W przypadku modelu SB2 generowane mają być dwa takie pliki. Nie muszą one posiadać jednakowej ilości linii, ani tych samych wartości w kolumnie 1.

```
EFAC 2.5
3853.6735 -61.414 1.09
3853.6902 -60.589 1.14
3854.6804 -9.716 1.04
3854.6978 -7.817 1.06
3855.6381 55.626 1.05
#3855.6546 56.980 1.11
3856.6581 89.260 0.96
3856.6740 88.171 0.95
3857.6752 60.224 1.03
3858.6816 -10.429 0.88
3858.6975 -12.298 0.96
3859.6785 -61.313 1.11
3859.6940 -60.947 0.98
JUMP 1 1
EQUAD 0.25
4606.354447 -53.631 0.117
4606.380649 -54.286 0.268
4612.364547 -15.733 0.258
4602.344046 44.082 0.392
4602.372331 46.661 0.442
4601.345582 -24.307 0.329
4601.371864 -24.761 0.262
4598.359986 21.701 0.417
4598.387449 18.873 0.371
#4597.353753 79.155 0.352
4597.379573 78.350 0.300
4595.381744 29.562 0.557
4594.336033 -40.459 0.698
4584.430998 53.129 0.243
4583.419625 87.563 0.256
```

Załącznik 3

Przykład pliku wyjściowego z pomiarami prędkości radialnych dla programu V2FIT. Znaczenie kolumn: 1 – czas, 2 – pomiar, 3 – zmodyfikowana niepewność pomiaru, 4 – rezydua dopasowania (różnica wartości zmierzonej i modelowej), 5 – faza orbitalna, 6 – kolumna nieaktywna, 7 – oryginalna niepewność pomiaru. Kolumny rozseparowane spacjami. Do stworzenia wykresów serwis powinien wczytać kolumny 1 do 5.

W przypadku modelu SB2 generowane są dwa takie pliki. Nie muszą one posiadać jednakowej ilości linii, ani tych samych wartości w kolumnie 1.

| | | | | | | |
|-------------|-----------|---------|----------|---------|---------|---------|
| 3853.673500 | -61.41400 | 2.72500 | -1.22798 | 0.55292 | 0.00000 | 1.09000 |
| 3853.690200 | -60.58900 | 2.85000 | -0.79909 | 0.55540 | 0.00000 | 1.14000 |
| 3854.680400 | -9.71600 | 2.60000 | 0.88236 | 0.70254 | 0.00000 | 1.04000 |
| 3854.697800 | -7.81700 | 2.65000 | 1.59629 | 0.70512 | 0.00000 | 1.06000 |
| 3855.638100 | 55.62600 | 2.62500 | 1.12031 | 0.84488 | 0.00000 | 1.05000 |
| 3856.658100 | 89.26000 | 2.40000 | 1.38795 | 0.99650 | 0.00000 | 0.96000 |
| 3856.674000 | 88.17100 | 2.37500 | 0.28248 | 0.99886 | 0.00000 | 0.95000 |
| 3857.675200 | 60.22400 | 2.57500 | 2.82461 | 0.14769 | 0.00000 | 1.03000 |
| 3858.681600 | -10.42900 | 2.20000 | 0.08130 | 0.29727 | 0.00000 | 0.88000 |
| 3858.697500 | -12.29800 | 2.40000 | -0.70953 | 0.29963 | 0.00000 | 0.96000 |
| 3859.678500 | -61.31300 | 2.77500 | -1.39300 | 0.44540 | 0.00000 | 1.11000 |
| 3859.694000 | -60.94700 | 2.45000 | -0.66401 | 0.44771 | 0.00000 | 0.98000 |
| 4606.354447 | -53.63100 | 0.29250 | 0.13698 | 0.41573 | 0.00000 | 0.11700 |
| 4606.380649 | -54.28600 | 0.67000 | 0.40272 | 0.41963 | 0.00000 | 0.26800 |
| 4612.364547 | -15.73300 | 0.64500 | -0.10250 | 0.30897 | 0.00000 | 0.25800 |
| 4602.344046 | 44.08200 | 0.98000 | -0.11198 | 0.81965 | 0.00000 | 0.39200 |
| 4602.372331 | 46.66100 | 1.10500 | 0.65718 | 0.82386 | 0.00000 | 0.44200 |
| 4601.345582 | -24.30700 | 0.82250 | -0.09781 | 0.67126 | 0.00000 | 0.32900 |
| 4601.371864 | -24.76100 | 0.65500 | -2.20656 | 0.67517 | 0.00000 | 0.26200 |
| 4598.359986 | 21.70100 | 1.04250 | -0.89842 | 0.22762 | 0.00000 | 0.41700 |
| 4598.387449 | 18.87300 | 0.92750 | -1.79015 | 0.23171 | 0.00000 | 0.37100 |
| 4597.379573 | 78.35000 | 0.75000 | 0.15236 | 0.08190 | 0.00000 | 0.30000 |
| 4595.381744 | 29.56200 | 1.39250 | 1.06141 | 0.78492 | 0.00000 | 0.55700 |
| 4594.336033 | -40.45900 | 1.74500 | -0.11351 | 0.62952 | 0.00000 | 0.69800 |
| 4584.430998 | 53.12900 | 0.60750 | -0.59496 | 0.15750 | 0.00000 | 0.24300 |
| 4583.419625 | 87.56300 | 0.64000 | -0.41489 | 0.00716 | 0.00000 | 0.25600 |

Załącznik 4

Przykład pliku wyjściowego z wartościami dopasowanych parametrów z programu V2FIT.

Pre-fit Parameters:

```
Period [days] =      6.72860925170000
Tp [MJD] =      4953.58817450000
Ecc =      0.0000000000000000E+000
omega [deg] =      0.0000000000000000E+000
Inclination [deg] =      84.88700000000000
V01 [km/s] =      17.700000000000000
K1 [km/s] =      65.000000000000000
V02 [km/s] =      0.0000000000000000E+000
K2 [km/s] =      78.000000000000000
V11 [km/s/d] =      0.0000000000000000E+000
V12 [km/s/d] =      0.0000000000000000E+000
V21 [km/s/d/d] =      0.0000000000000000E+000
V22 [km/s/d/d] =      0.0000000000000000E+000
SOMDot [deg/d] =      0.0000000000000000E+000
```

```
RV1 NObs =          36
RV1 RMS =      13.1024690414372
RV1 Chi^2 =      96395.6098270706
RV2 NObs =          36
RV2 RMS =      10.0756091111447
RV2 Chi^2 =      19089.1085622407
Chi^2/DoF =      1804.44872483299
NObs =           72
DoF =           64
```

ok

MINPACK info:

algorithm estimates that the relative error in the sum of squares is at most to
1 and that the relative error between x and the solution is at most tol

notes:

```
tol =      1.0000000000000000E-010
fvec is an output array of length NObs which contains the model function
evaluated at the parameter vector x
x is the parameter vector
fcn is the name of the subroutine which calculates the model function
```

RV1 NObs = 36
RV1 RMS = 0.993573268025470
RV1 Chi^2 = 30.3283915653110
RV2 NObs = 36
RV2 RMS = 1.17016626849278
RV2 Chi^2 = 37.1976301128527
Chi^2/DoF = 1.05509408872131
NObs = 72
DoF = 64

Post-fit Parameters:

Period (assumed) [days] = 6.72860925170000
Tp [MJD] = 4953.444955988491 +/- 0.1010110431891618E-02
Ecc (assumed) = 0.0000000000000000E+000
omega (assumed) [deg] = 0.0000000000000000E+000
Inclination (assumed) [deg] = 84.88700000000000
V01 [km/s] = 11.7676059402338 +/- 0.522349141156110
K1 [km/s] = 76.1228561277169 +/- 0.151005609043395
V02 (assumed) [km/s] = 0.0000000000000000E+000
K2 [km/s] = 78.8017435318653 +/- 0.182649760650746
V11 (assumed) [km/s/d] = 0.0000000000000000E+000
V12 (assumed) [km/s/d] = 0.0000000000000000E+000
V21 (assumed) [km/s/d/d] = 0.0000000000000000E+000
V22 (assumed) [km/s/d/d] = 0.0000000000000000E+000
SOmDot (assumed) [deg/d] = 0.0000000000000000E+000
JumpRV1 [km/s] = 0.164490474861765 +/- 0.543886527633881
JumpRV2 [km/s] = 0.253899776264858 +/- 0.549958390589577
JumpRV1 [km/s] = 0.829457985949952 +/- 0.537059313140835
JumpRV2 [km/s] = 0.650875842486974 +/- 0.542025715678800

Binary Parameters:

a1*sin(i) [km] = 7043268.70251915 +/- 13971.7968333900
a1 [AU] = 4.726943448117965E-002 +/- 9.376881145597414E-005
a2*sin(i) [km] = 7291132.81023937 +/- 16899.6725590055
a2 [AU] = 4.893292294015698E-002 +/- 1.134186410488048E-004
a12 [AU] = 9.620235742133663E-002 +/- 1.471610618989400E-004
a12 [RSun] = 20.6923109808751 +/- 3.165309602291831E-002
Sem. axis [mas] = 0.0000000000000000E+000 +/- 0.0000000000000000E+000
M1 [MSun] = 1.33450075517184 +/- 7.085866222161532E-003
M1 sin3i [MSun] = 1.31863360565681 +/- 6.680007839556661E-003
R1 [RSun,ZAMS] = 1.23368432668757
Theta1 [mas,ZAMS] = 0.0000000000000000E+000
M2 [MSun] = 1.28913402718300 +/- 6.257247621223937E-003
M2 sin3i [MSun] = 1.27380628587229 +/- 5.841368814180905E-003
R2 [RSun,ZAMS] = 1.20543370886122
Theta2 [mas,ZAMS] = 0.0000000000000000E+000
om1 = NaN
om2 = NaN
sin2i = 0.0000000000000000E+000

Załącznik 5

Fragment przykładowego pliku wyjściowego z programu V2FIT z modelową krzywą RV. Znaczenie kolumn: 1 – czas, 2 do 4 – nieistotne, 5 – faza orbitalna, 6 – faza orbitalna zmodyfikowana, 7 – modelowa wartość prędkości, 8 – nieistotna. Kolumny rozseparowane spacjami. Cały plik ma 2000 linii (~154 kB). Do stworzenia wykresów serwis powinien wczytać kolumny 1, 6 i 7.

W przypadku modelu SB2 generowane są dwa takie pliki. Oba jednakowej długości.

| | | | | | | | |
|-----------|-------|-------|----------|----------|----------|--------|--------|
| 4956.4764 | 0.000 | 0.000 | 0.000000 | 0.000000 | 0.000000 | 28.896 | -0.000 |
| 4956.4765 | 0.000 | 0.000 | 0.000000 | 0.001000 | 0.001000 | 28.895 | -0.000 |
| 4956.4766 | 0.000 | 0.000 | 0.000000 | 0.002000 | 0.002001 | 28.893 | -0.000 |
| 4956.4767 | 0.000 | 0.000 | 0.000000 | 0.003000 | 0.003001 | 28.891 | -0.000 |
| 4956.4769 | 0.000 | 0.000 | 0.000000 | 0.004000 | 0.004001 | 28.887 | -0.000 |
| 4956.4770 | 0.000 | 0.000 | 0.000000 | 0.005000 | 0.005001 | 28.882 | -0.000 |
| 4956.4771 | 0.000 | 0.000 | 0.000000 | 0.006000 | 0.006002 | 28.875 | -0.000 |
| 4956.4773 | 0.000 | 0.000 | 0.000000 | 0.007000 | 0.007002 | 28.868 | -0.000 |
| 4956.4774 | 0.000 | 0.000 | 0.000000 | 0.008000 | 0.008002 | 28.859 | -0.000 |
| 4956.4775 | 0.000 | 0.000 | 0.000000 | 0.009000 | 0.009003 | 28.850 | -0.000 |
| 4956.4776 | 0.000 | 0.000 | 0.000000 | 0.010000 | 0.010003 | 28.839 | -0.000 |
| 4956.4778 | 0.000 | 0.000 | 0.000000 | 0.011000 | 0.011003 | 28.827 | -0.000 |
| ... | | | | | | | |

Załącznik 6

Przykład pliku konfiguracyjnego dla programu wykorzystującego metodę TODCOR. Pliki `template_1` i `template_2` mogą być takie same. Program tworzy dwa pliki wynikowe `output_result_name-I.dat` `output_result_name-II.dat` z wynikami pomiarów, oraz opcjonalnie `output_error_name` do oszacowania błędów.

```
[input_spectrum_file] [template_1] [template_2] [output_result_name]
1 -15 130
0 50
20
6348. 6500. 2
6172. 6397. 3
6006. 6224. 4
5848. 6060. 5
5698. 5905. 6
5555. 5755. 7
5420. 5617. 8
5291. 5484. 9
5168. 5356. 10
5051. 5235. 11
4939. 5119. 12
4832. 5007. 13
4733. 4901. 14
4635. 4799. 15
4545. 4701. 16
4452. 4607. 17
4368. 4517. 18
4292. 4430. 19
4206. 4347. 20
4135. 4267. 21
4135. 6500.
0 50 [output_error_name]
```


Załącznik 8

Fragment przykładowego pliku wyjściowego z programu wykorzystującego metodę TODCOR, służącego do oszacowania błędów (`output_error_name`). Znaczenie kolumn: 1 – numer iteracji, 2 – wartość dla gwiazdy 1, 3 – wartość dla gwiazdy 2. Ilość linii (iteracji) ustala użytkownik w pliku konfiguracyjnym (ostatnia linia, druga pozycja). Serwis powinien obliczać dwa średnie odchylenia standardowe, ze wszystkich wartości z kolumn 2 i 3 i wyświetlać je użytkownikowi.

| | | |
|-----|---------------------|----------------------|
| 1 | 119.592000000000001 | -121.241000000000000 |
| 2 | 119.167000000000000 | -115.045999999999999 |
| 3 | 118.617000000000000 | -121.241000000000000 |
| 4 | 119.597000000000001 | -121.241000000000000 |
| 5 | 119.802000000000001 | -113.151000000000000 |
| 6 | 119.117000000000000 | -112.176000000000000 |
| 7 | 119.502000000000001 | -111.241000000000000 |
| 8 | 119.477000000000000 | -115.855999999999999 |
| 9 | 118.122000000000001 | -121.241000000000000 |
| 10 | 119.147000000000001 | -121.241000000000000 |
| 11 | 118.677000000000001 | -113.086000000000000 |
| ... | | |