

財團法人國家實驗研究院國家太空中心

全球定位科學應用研究委託案(3/3)

101年度「全球定位科學應用研究中心」

期末報告

工作項目(5.2)：大氣層掩星資料反演軟體及軟體使用手冊

(第二版)、測試報告

計畫名稱：掩星資料處理技術建置：大氣層掩星資料自主反
演技術發展

執行期間： 2012/02/18~2013/02/17

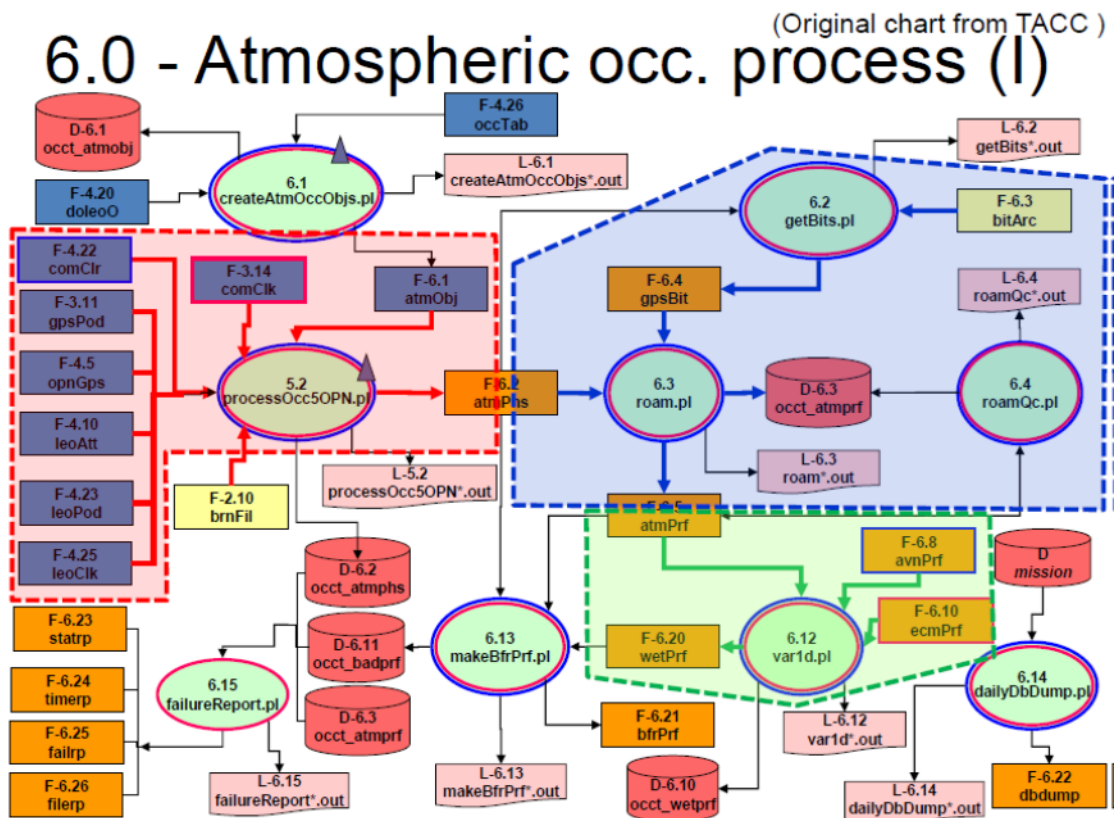
分項工作管理人：黃成勇

參與人員：黃成勇、蔡和芳、謝函蓁、吳靜美

中 華 民 國 1 0 2 年 2 月

一、摘要

本篇報告主要說明大氣掩星剖面反演，包含超額相位反演，大氣乾、濕溫度剖面反演，內容包含反演原理與各副程式功能說明。超額相位反演本軟體資料處理目標為從 GPS 衛星軌道和福衛三號軌道資訊以及三號衛星酬載 GOX 所收到的 50Hz 觀測資料反演到掩星超額相位(Excess Phase)。主要輸入資料為 POD 兩個天線所收到的 50Hz 參考資料和 OCC 兩個天線所收到的掩星觀測資料。乾溫度剖面本計畫目前使用 phase matching 法，計算偏折角，改進傳統電離層校正方式，忽略大氣水氣的影響，反演出大氣乾溫度剖面、折射率和壓力剖面。濕氣壓剖面反演須利用地面輔助觀測資料或大氣模式進行 one-D VAR 變分最佳化處理。目前前兩部分已經已 Perl 以及 Fortran 語言完成大部分的程式，第三部分已經使用 R Langrange 完成一維變分方法折射率資料反演的測試，下年度四月前將完成一維變分篇折角，以 Fortran 程式完成的套件。屆時大氣反演將有初版完整的反演套件。



圖一、大氣資料反演流程圖

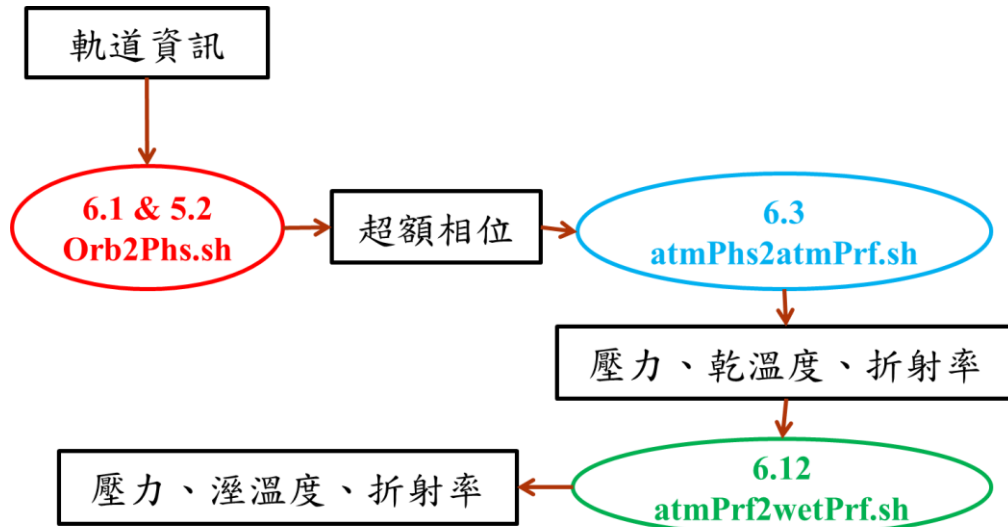
	Finish	Future Work
<u>Orbit →</u> <u>excess phase</u>	<ul style="list-style-type: none"> ● Occ. table for single difference ● Single difference for excess phase ● Cycle slip detection ● Time synchronization ● Relativity correction 	<ul style="list-style-type: none"> ➤ Zero difference & Double difference ➤ Orbit improvement (colaboration with Tuse-Pang Chen)
<u>Excess phase →</u> <u>atmPrf</u>	<ul style="list-style-type: none"> ● Ionospheric correction ● Cycle slip detection ● Forward model ● Open Loop (GPSBit) ● Bending angle optimization ● Phase matching method for bending angle ● Refraction, Pressure, Dry temperature 	<ul style="list-style-type: none"> ➤ High order ionospheric effect ➤ WO improvement ➤ Data QC improvement ➤ Background model improvement at Higt Altitude (>25km)
<u>atmPrf →</u> <u>wetPrf</u>	<ul style="list-style-type: none"> ● Ground temperature fixing or model temperature fixing method ● One-D Var (Refractivity) (test in R Langrange) 	<ul style="list-style-type: none"> ➤ Transfer to Fortran Langrange (1DVAR BA) ➤ Two dimation, end to end, forward model for bending angle

表一、列出三個模組:衛星軌道至超額相位、超額相位到乾剖面、乾剖面到濕剖面等各自已經完成的函示及其功能(藍色)，以及未來可以改進的部分(紅色)。

二、 大氣資料軟體處理程序

圖一為目前 CDAAC 反演大氣剖面資料的資料流程，主要分成三個部分。第一部分為圖上紅色區域，須入資料為 GPS 和低軌道衛星資料，衛星時鐘誤差和衛星姿態，及掩星觀測資料，透過一次差分或二次差分反演超額相位。第二部分為藍色區域，由超額相位經 Abel-inversion 反演 GPS 訊號路徑偏折角、大氣折射率、壓力和乾溫度剖面。第三部分為藍色區域部分，由大氣乾溫度剖面結合地

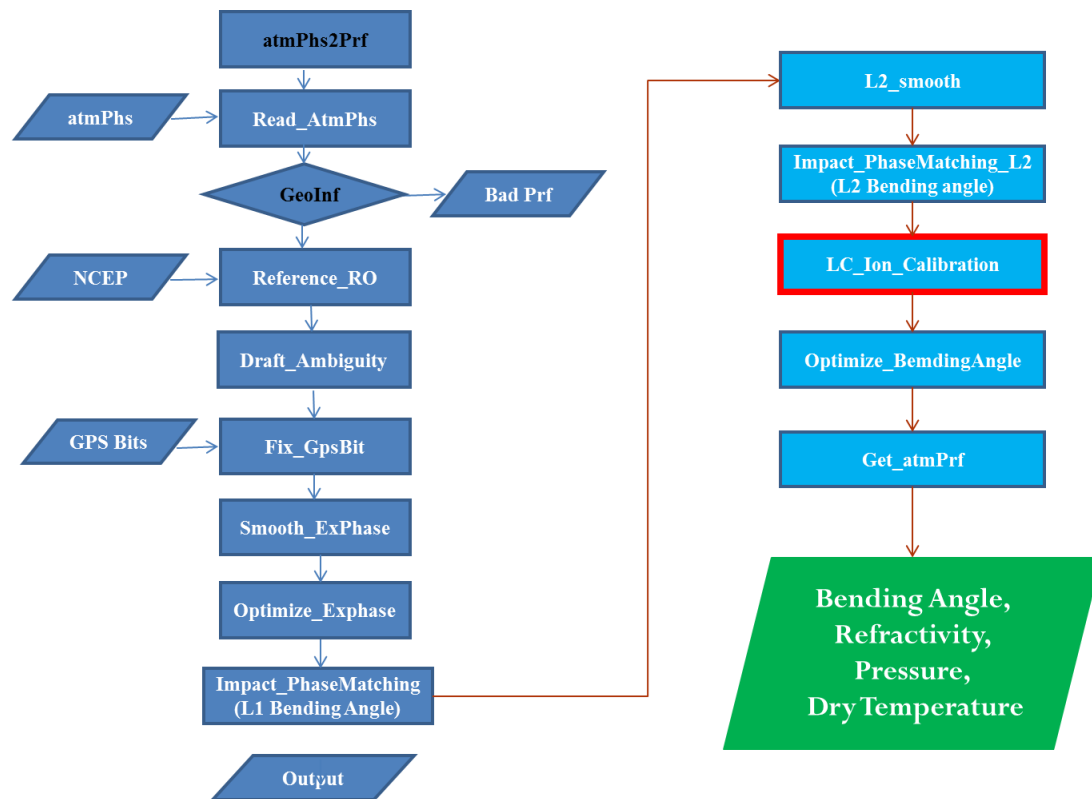
面輔助大氣觀測資料或大氣模式，經行 one-D VAR 處理反演出濕乾溫度剖面 and 濕氣壓剖面。下圖為目前本計畫大氣掩星資料流程簡圖，詳細資料流程會在之後報告做更詳細的說明。



圖二、大氣掩星反演資料處理流程

三、 超額相位反演至乾溫度剖線:

此部分資料處理輸入資料包含超額相位(atmPhs)，GPS navigation message (GPSBit)，大氣參考背景模式(NCEP or CIRA)，輸出資料為乾溫度剖線參數(atmPrf)，參數包含偏折角、折射係數、壓力以及溫度剖線。程式主要功能包含 10 公里以下 OPEN LOOP 接收訊號校正，訊號超額相位和偏折角模擬，Phase Matching 法計算偏折角、電離層影響誤差校正、偏折角最佳化、大氣參數(折射率、乾溫度、壓力)計算等。



圖三、大氣掩星剖面資料處理流程圖

副程式功能說明:

AtmPhs2Prf.f90:

為大氣掩星剖面主程式，主要功能為控制由大氣超額相位反演到大氣掩星剖面的掩星反演資料處理流程，呼叫副程式執行各項功能。

Read_atmPhs.f90

讀取 netcdf 格式的 atmPhs 檔案，目前所使用的套件為 netcdf_4.0.1 靜態函式庫。

GeoInf.f90

依低軌道衛星以及 GPS 衛星軌道計算掩星事件相關地理資訊，包含掩星事件地點，時間，掩星剖線的最高點以及最低點，並依掩星事件高低點以及事件發生判斷掩星觀測資料反演與否。

Reference_RO.f90

依 Ray Tracing 方式以及以 NCEP 大氣模式為背景條件，模擬大氣訊號路徑以及訊號延遲量。模擬出來的相位當作參考訊號，以做為修正觀測到的掩星訊號週波模稜值判定使用，尤其做為低高度，OPEN LOOP 校正使用。

Ref_bending_excess.f90

依 Ray Tracing 方式以及以 NCEP 大氣模式為背景條件，模擬大氣訊號路徑以及訊號延遲量以及掩星訊號偏折角。模擬出來的相位當作參考訊號，以做為修正觀測到的掩星訊號週波模稜值判定使用，尤其做為低高度，OPEN LOOP 校正使用。

Fix_gpsBit.f90

讀取 GPS bit 資料以及校正 OPEN LOOP，GPS RO 訊號在接近地表約 15 公里以下，低軌道衛星 GPS 接收器會使用 OPEN LOO 技術追鎖 GPS 掩星訊號，接收到的相位會隨 GPS bits 資料而有半波長的差異，因此需要須入 GPS bits 資料校正，已獲得分辨性較好的全波長觀測訊號。

Draft_Ambiguity.f90

初步偵測與校正 L1 以及 L2 訊號的週波模稜值，使訊號更接近真實情形以及模擬的參考訊號，以利接下來更精密的訊號週波模稜值判定。

Tools.f90

包含掩星反演軟體所需各項工具，如 Lagrange 內插，排序，高斯平滑，濾波等基本數學工具。

Half_Wavelength.f90

以半波長方式修補 OPEN LOOP 觀測資料，此方法用在缺乏 GPS bits 資料或 GPS bits 資料不完全時使用。

Gnufor2.f90

網路上免費程式，先在 Linux 下安裝 gnuplot 套裝軟體，在使用 Gnufor2 當作 Fortran 軟體畫圖介面，目前在城市開發階段，使用此軟體當作偵測工具，等軟體開發完成，則視是否需要安裝此軟體。

Smooth_ExPhase.f90

此程式主要功能為判定訊號的品質，依不同的時間窗區判定資料的好壞，並作為掩星剖面最低點的依據。

Optimize_ExPhase.f90

依照 Smooth_ExPhase 的成果，以及週波模稜為整數等限制條件，最佳化超額相位判定。

Impact_PhaseMatching.f90

依據 Jensen2004 年文獻，並加以修改而成的程式，用來計算掩星訊號偏折角，為掩星反演精確與否的關鍵程式。此程式的反演方法稱為 Phase Matching method，根據理論為 Born and Wolf, 1999 的 stationary point。

LC_Ion_Calibration

傳統無電離層項線性組合，以雙頻訊號計算出來的偏折角，進行無電離層組合，消除高層電離層電子濃度對大氣剖面的影響。

Optimize_BendingAngle.f90

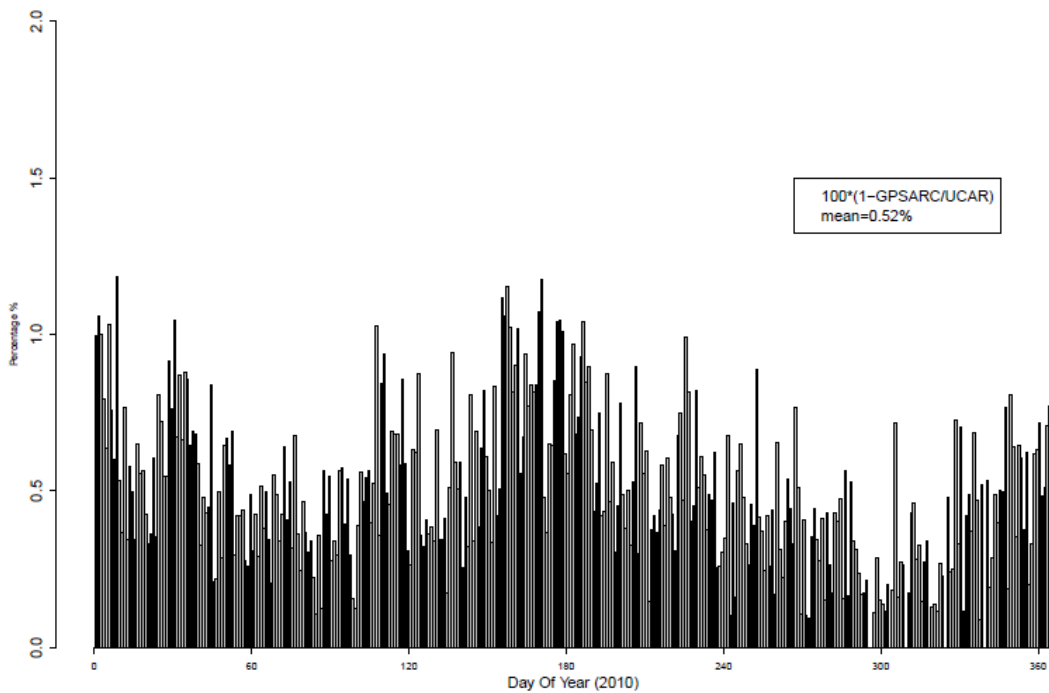
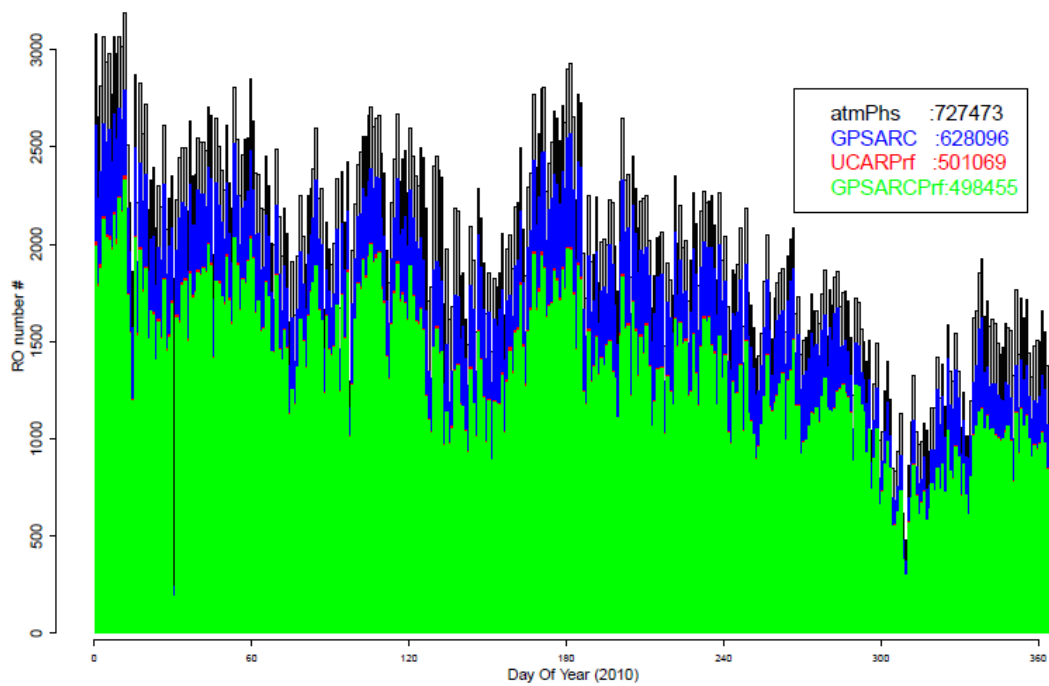
依據雙頻訊號的優劣，以及背景大氣模式條件，組合量測的訊號偏折角和背景條件偏折角，進行 30 公里以上偏折角的最佳化。

Get_atmPrf.f90

輸入偏折角，以 Abel inversion 計算乾溫度大氣參數，參數包含大氣折射係數，大氣壓力以及乾溫度垂直剖線。

成果分析

在此部份我們用了 2012 年一整年的掩星觀測資料，驗證大氣乾溫度剖面程式。在此區間中，福衛三號衛星總共觀測到 72 萬 7473 筆掩星觀測超額相位資料 (atmPhs)。UCAR 總共反演出 501069 筆乾溫度剖面，在 UCAR 反演的剖面中，我們反演出 498455 筆乾溫度剖面，為反演出來的比率約為 0.5。另外 UCAR 未反演出來的超額相位剖面中，我們還多反演出 127027 筆，因為目前不清楚 UCAR 未反演出來的原因，或是因為資料品質而不使用，所以上不可以說我們的反演的軟體比較好，此原因將待整套原形軟體完成後，做進一步的分析。在整個測試中，使用 Intel(R) Core(TM) i7 CPU(8 core) 2.93GHz CPU PC，8G ram，Linux FC16，同時使用 7 個 core,共花了約六天的時間。若以成功的剖面計算，平均單一剖面約花 6 秒的時間。下圖為 2012 一整年反演剖面數量的比較圖。黑色為掩星觀測到的掩星數量，藍色為自主程式反演的乾溫度剖面數量，紅色為 UCAR 反演出來的剖面數量，綠色為自主程式在紅色的掩星事件中可反演的掩星數目。下圖二顯示尚未反演出來的比率，平均尚未反演出來的比率約為 0.5%。



四、 超額相位計算原理

衛星訊號介紹

福衛三號四個 GPS 天線所量測到的訊號主要為 1Hz 的 CA 的虛擬距離 (version 1)、50Hz 的雙頻 L1 和 L2 的載波相位。虛擬距離用方程式表示如下:

$$Pr_{k,1}^p(t) = \rho_k^p(t) - cdt_k + cdt^p + I_{k,1,p}^p(t) + T_k^p(t) + d_{k,1,p}(t) + d_{1,p}^p(t) + \varepsilon_{1,p} \quad (1)$$

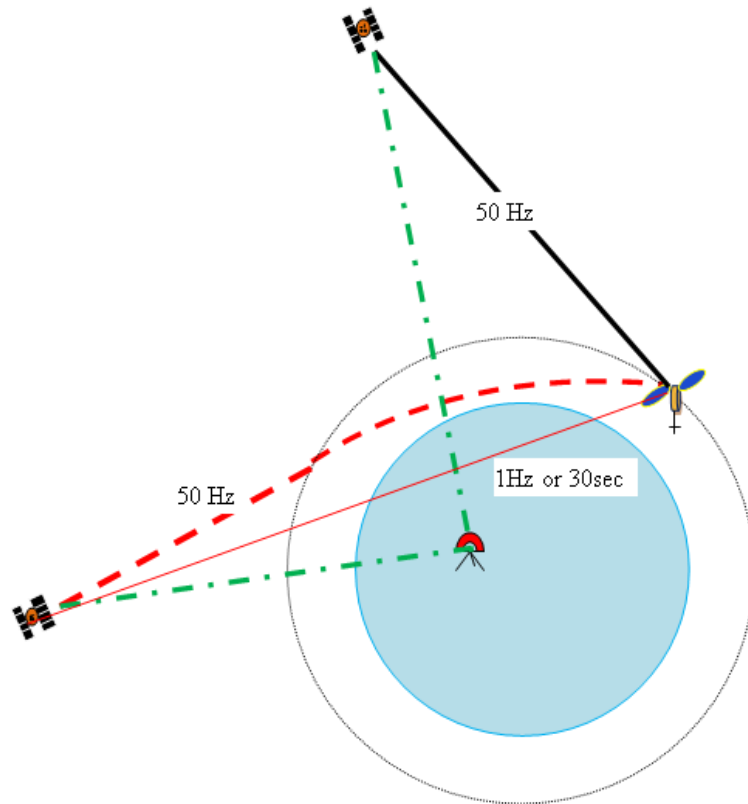
雙頻相位為

$$\Phi_{k,1}^p(t) = \rho_k^p(t) - cdt_k + cdt^p + \frac{c}{f_1} N_{k,1}^p(1) - I_{k,1,p}^p(t) + T_k^p(t) + d_{k,1,\Phi}(t) + d_{1,\Phi}^p(t) + \varepsilon_{1,\Phi} \dots (2)$$

$$\Phi_{k,2}^p(t) = \rho_k^p(t) - cdt_k + cdt^p + \frac{c}{f_2} N_{k,2}^p(1) - \frac{f_1^2}{f_2^2} I_{k,2,p}^p(t) + T_k^p(t) + d_{k,2,\Phi}(t) + d_{2,\Phi}^p(t) + \varepsilon_{2,\Phi} \dots (3)$$

上式中， c 為光速，數字 1 和 2 分別表示載波 L1 和 L2， ρ 為 GPS 衛星 p 和 GPS 訊號接收器 k 的直線距離， dt^p 和 dt^q 分別代表 GPS 衛星(p)和接收器(q)的時鐘誤差， I 為電離層延遲， T 為大氣延遲， $d_{p,1}$ 、 $d_{\phi,1}$ 和 $d_{\phi,2}$ 分別代表 GPS 接收器硬體電碼延遲和雙頻的相位延遲， $d^{p,1}$ 、 $d^{\phi,1}$ 和 $d^{\phi,2}$ 為 GPS 衛星的電碼延遲和雙頻的相位延遲。

一次差分



圖四、超額相位解算幾何圖

如圖四所示，掩星超額相位解算作業中主要有兩種方式，分別為一次差分和二次差分方式。一次差分為兩顆衛星和一顆低軌道衛星，如圖上黑色實線和紅色虛線進行差分，得到紅色虛線的超額相位。二次差分為利用二顆 GPS 衛星，一個低軌道衛星和一個地面 GPS 接收站訊號進行差分，即利用圖上兩條綠色虛線進行差分後在與低軌道衛星訊號一次差分後的結果進行差分，下列分別用方程式表示一次差分：

$$\Phi_{Leo,1}^{pq}(t) = \rho_{Leo}^{pq}(t) + cdt^{pq} + \frac{c}{f_1} N_{Leo,1}^{pq}(1) - I_{Leo,1,P}^{pq}(t) + d_{Leo,1,\Phi}^{pq}(t) + \varepsilon_{1,\Phi} \quad (4)$$

$$\Phi_{Leo,2}^{pq}(t) = \rho_{Leo}^{pq}(t) + cdt^{pq} + \frac{c}{f_2} N_{Leo,2}^{pq}(1) - \frac{f_1^2}{f_2^2} I_{Leo,2,P}^{pq}(t) + d_{Leo,2,\Phi}^{pq}(t) + \varepsilon_{2,\Phi} \quad (5)$$

和二次差分：

$$\Phi_{Leo,Gnd,1}^{pq}(t) = \rho_{Leo,Gnd}^{pq}(t) + \frac{c}{f_1} N_{Leo,Gnd,1}^{pq}(1) - I_{Leo,Gnd,1,P}^{pq}(t) - T_G^{pq}(t) + \varepsilon_{1,\Phi} \quad (6)$$

$$\Phi_{Leo,Gnd,2}^{pq}(t) = \rho_{Leo,Gnd}^{pq}(t) + \frac{c}{f_2} N_{Leo,Gnd,2}^{pq}(1) - \frac{f_1^2}{f_2^2} I_{Leo,Gnd,2,P}^{pq}(t) - T_G^{pq}(t) + \varepsilon_{2,\Phi} \quad (7)$$

在低軌道衛星上，大氣延遲可被忽略。比較二次差分和一次差分，一次差分無大氣延遲效應，但是仍有 GPS 軌道誤差、GPS 時鐘誤差和 GPS 硬體延遲，另外還有廣義相對論影響(下節描述)。因此一次差分所需要的軌道經度需求比二次差分需求高。因應未來與 UCAR 合作的不確定性，地面 GPS 輔助站的資料獲得量的不定性和即時性，本文直接採用 International GNSS Service 的 GPS 精密軌道和 GPS 時鐘誤差，校正 GPS 訊號一次差分的結果。

廣義相對論校正

廣義相對論也會影響 GPS 的時鐘誤差，其形成原因為衛星經過不同重力位能場時，重力加速度造成。由狹義相對論產生的時鐘誤差，已經經由 GPS 衛星直接校正，所以本文無須校正，另外低軌道衛星所受到的影響，也會經由一次差分消除，所以無需校正。重力對 GPS 造成的時鐘誤差為：

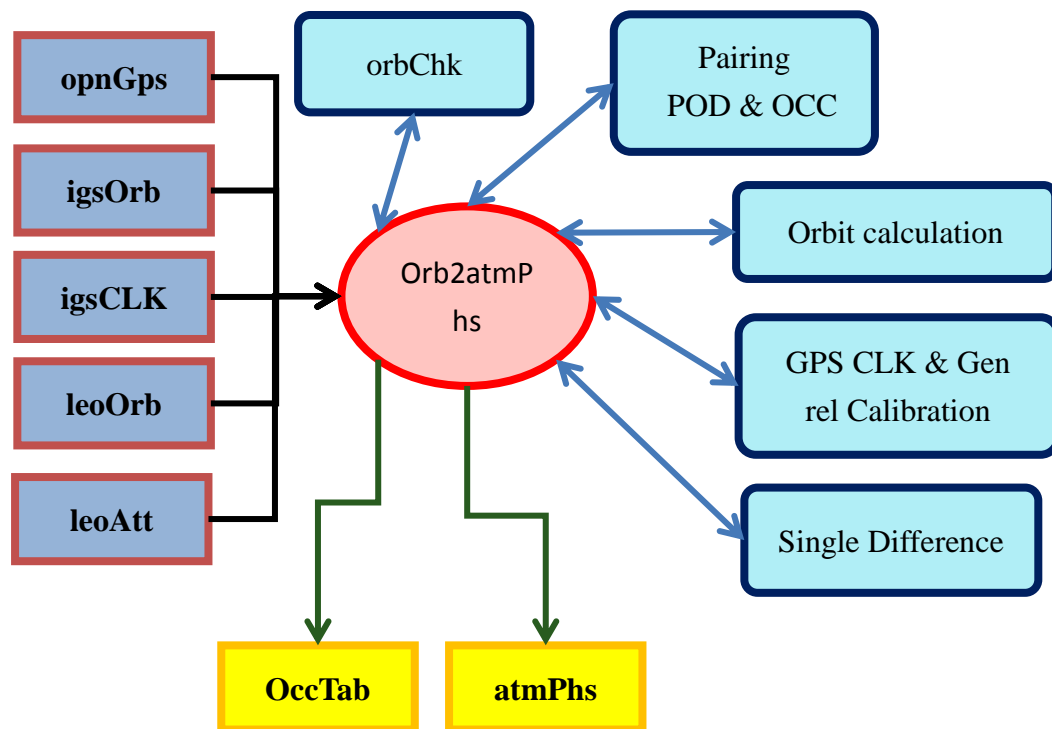
$$\Delta t_R(t) = -\frac{2}{c^2} \sqrt{a\mu e} \sin E = -\frac{2}{c^2} \mathbf{X} \cdot \dot{\mathbf{X}} \quad (8)$$

\mathbf{X} 和 $\dot{\mathbf{X}}$ 分別代表衛星在 ECI 座標下的位置和速度向量，其第二階重利差修正項為

$$\delta\rho_{r,rel}^S = \frac{2GM_E}{c^2} \ln\left(\frac{r^S+r_r+\rho_r^S}{r^S+r_r-\rho_r^S}\right) \quad (9)$$

G 為重力場常數， M_E 為地球重量， r^s 和 r_r 分別為衛星和接收器的地心半徑長。在忽略周波模稜值和噪音的情況下。

五、 成果報告



圖五、超額相位資料處理流程圖

輸入資料:

igsOrb : GPS 衛星軌道資料，內容包含每 15 分鐘 32 顆 GPS 衛星在地固座標的位置和時鐘誤差(SP3 format, from IGS)。

igsCLK : GPS 衛星每 30 秒鐘的衛星時鐘誤差(SP3 format, from IGS)。

leoOrb：低軌道衛星每一秒的衛星在地固座標的位置和時鐘誤差(SP3 format, from TACC)。

leoAtt：低軌道衛星姿態(VC1)，紀錄衛星飛行時本體及太陽能板相對三軸位置及轉動角度。

opnGps：包含 POD 和 OCC 天線接收到的 GPS 訊號，大部分目前的掩星觀測資料為 version11.0，內含 1Hz CA 電碼所量到的虛擬距離，50Hz 的雙頻載波訊號。

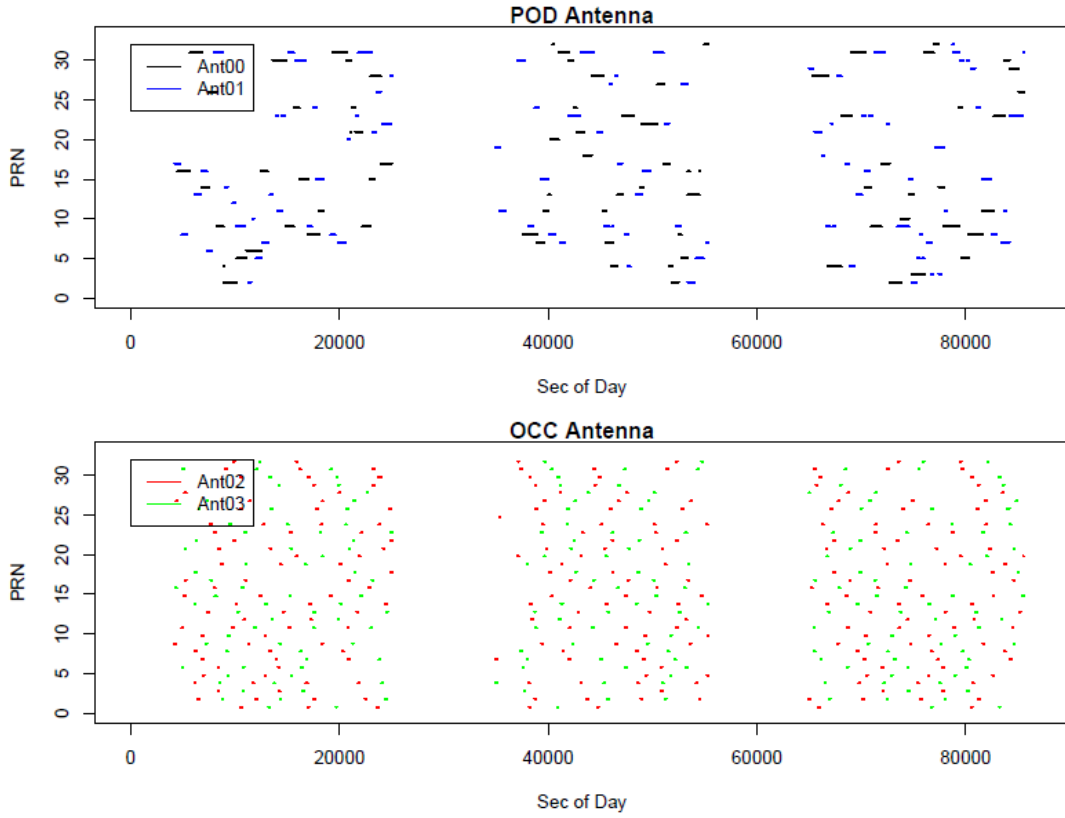
輸出資料

Occtab：建立一次差分掩星觀測事件表，輸出包含掩星事件發生位置、時間、POD 與 OCC 掩星配對等。

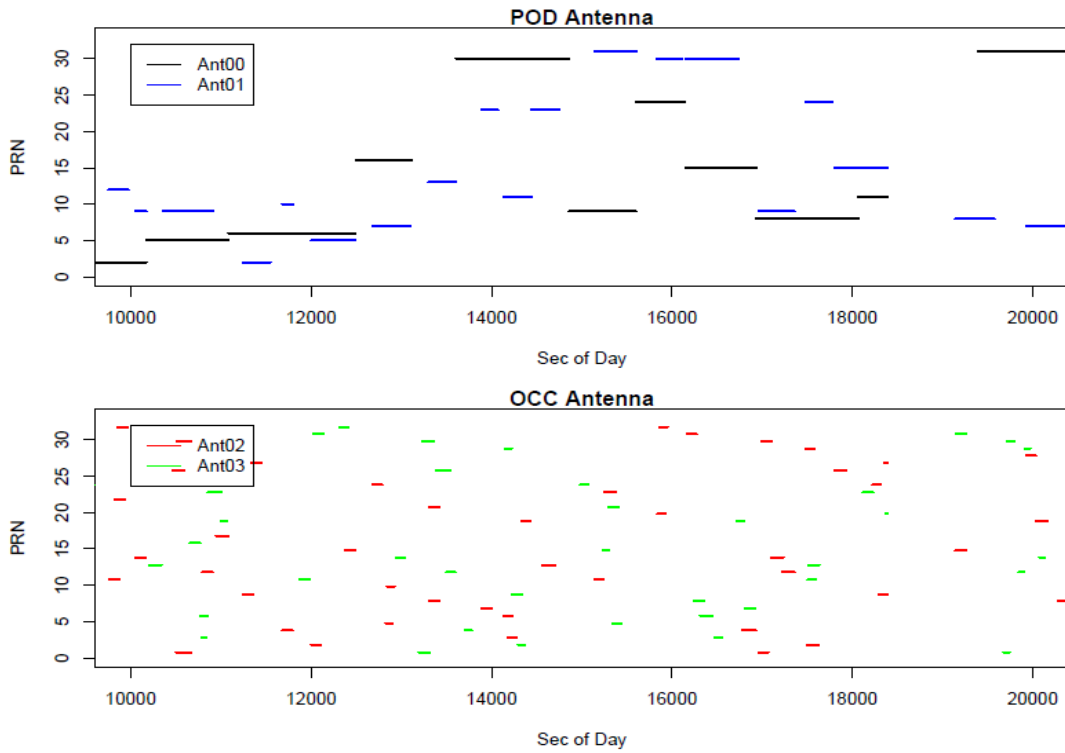
atmPhs：GPS 衛星軌道，低軌道衛星軌道、L1 和 L2 超額相位。

資料處理流程

1. 進行低軌道衛星確認，確定獲得低軌道衛星軌道的窗區，並進行衛星品質確認。
2. 將 POD 和 OCC 所量測的 50Hz 資料進行配對，並確認可獲得低軌道得到可以進行一次差分的掩星事件，輸出掩星觀測事件表。圖六和圖七顯示 POD(黑色藍色線)天線和 OCC(紅色和綠色)天線觀測到的 GPS 訊號時間表。將 OCC 天線的時間軸與 POD 天線同一間做匹配，即得掩星觀測一次差分表。
3. 讀取 GPS 衛星時鐘誤差，進行衛星時鐘誤差校正。
4. 計算衛星速度，衛星軌道半徑，廣義相對論影響校正。
5. 一次差分，獲得超額相位，訊號周波脫落偵測修復。



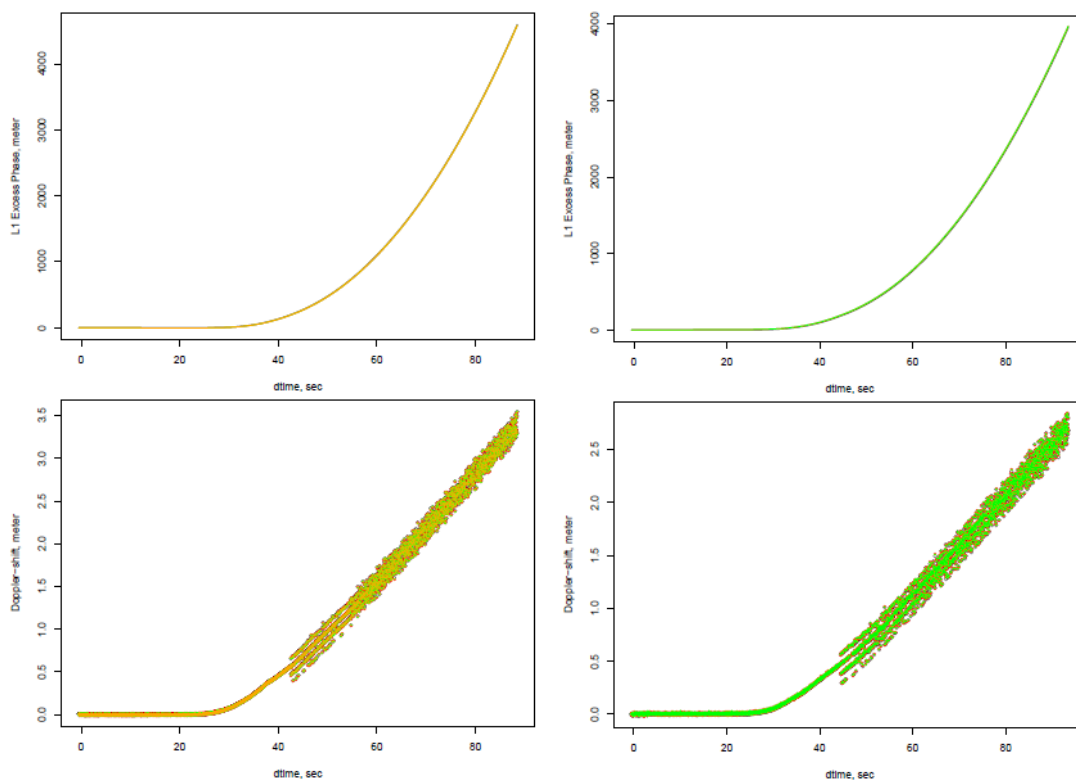
圖六，POD 和 OCC 天線觀測時間表



圖七，POD 和 OCC 天線觀測時間表

下圖為中心目前發展的軟體所反演出來的超額相位和都普勒平移量，與

UCAR(黑色實線)相當一致，



圖八、超額相位及都普勒平移量反演成果

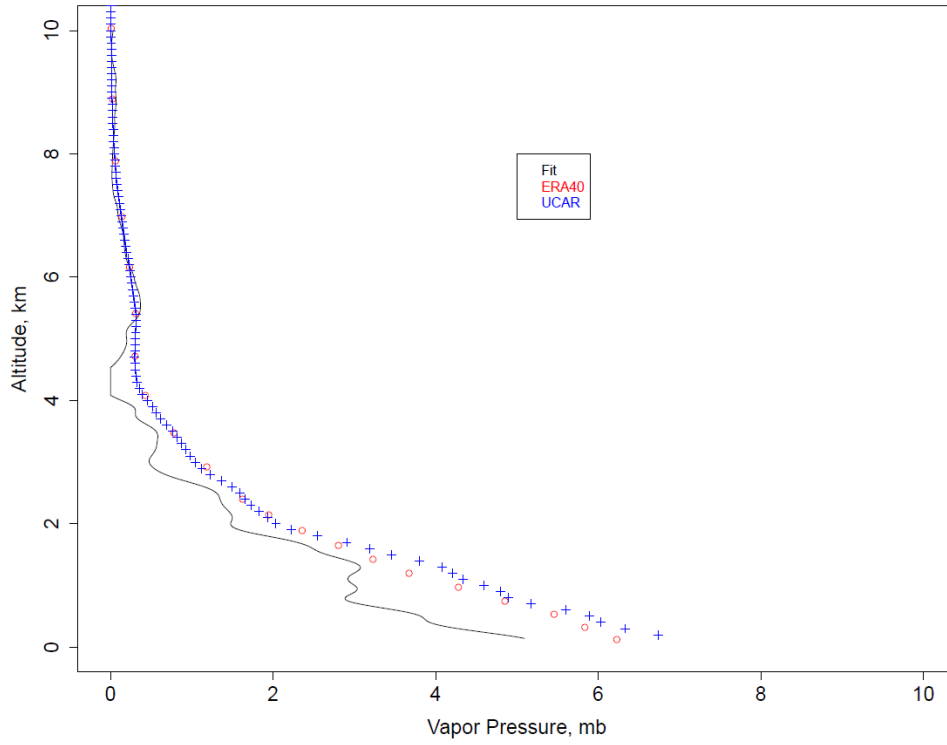
濕溫度剖面反掩

在失溫度剖面反演上，已經完成固定模式溫度剖面方法反演大氣濕溫度剖面。下圖舉出兩個反演的例子。圖上、黑色線為自主軟體反演出來的成果，藍色十字符號為 UCAR 目前在網上公布的結果，紅色為 ERA40 模式的成果。反演方式為裹用下列方程式

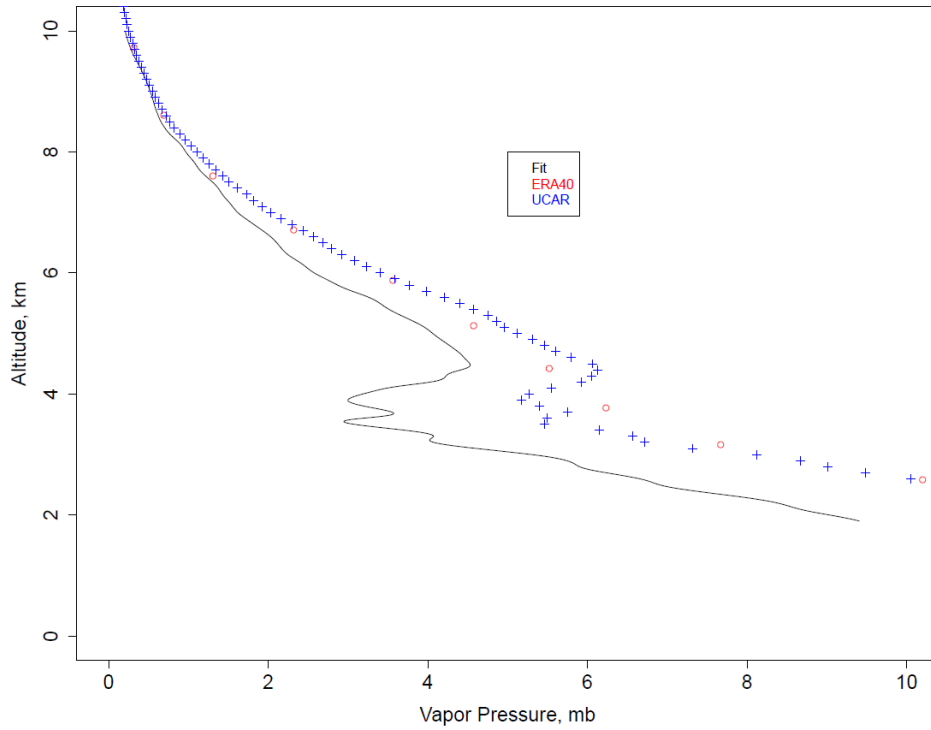
$$N = c_1 \frac{P}{T} + c_2 \frac{e}{T^2}$$

方程式中折射率 N 來自於乾溫度剖面，溫度剖面由模式提供，即可反演出大氣濕溫度剖面，溫度剖面可由乾溫度剖面的溫度折返點，加上地面溫度或由雲高溫度加以限制而得，取代模式溫度剖面。目前反演程式暫由 R Langrange 完成，下年度計畫將改由作業化效率較高的 Fortran 和 Perl 完成。

WetPrf_C001.2010.001.00.07.G32



WetPrf_C001.2010.001.00.38.G22



圖九、濕溫度剖面: 黑色線為自主軟體反演出來的成果，藍色十字符號為 UCAR 目前在網上公布的結果，紅色為 ERA40 模式的成果。

瞭解 CDAAC 掩星事件表建立流程

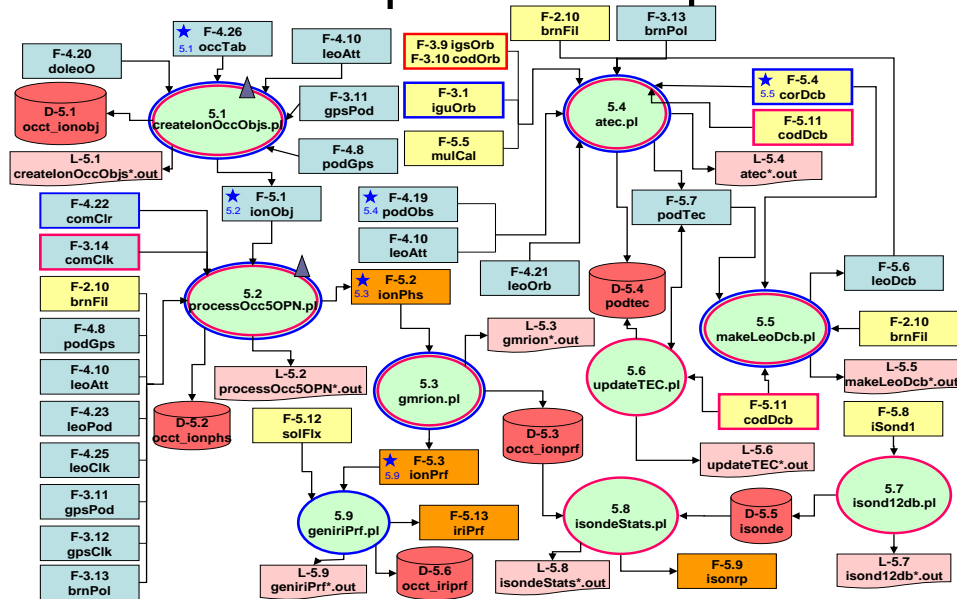
掩星事件表是一種清單，記錄掩星事件發生的 metadata，以提供大氣、電離層反演程序的必要資訊之一。CDAAC 程式在 LEO POD 程序之後，大氣及電離層反演程序之前，建立掩星事件表(occTab)，進而產生大氣掩星物件(atmospheric occultation objects: atmObj)，以及電離層掩星物件(ionospheric occultation objects: ionObj)，分別提供大氣及電離層反演程序使用。

電離層掩星物件其實為 perl 片段，其中包含電離層反演所需的輸入檔路徑，方便電離層反演程序處理。其中會比較掩星起訖時間，篩選掩星時間長度至少大於 30 秒的物件；小於 30 秒的物件會被捨棄。將代表同一掩星事件的前後兩半資料檔名並列於同一 perl 片段中(但直到一次差超量相位程序才處理銜接的問題)。不同的是，篩選電離層掩星事件的工作是根據 podGps 而不是 opnGps。

createIonOccObjs.pl: 主要從 occTab 中，篩選出實際發生的電離層掩星事件。此外，不以最高點和最低點的掩星觀測時間來判斷上升和下降掩星，而是以 GOX 天線和姿態 (leoAtt) 來決定。同樣到最後，程式將每一個 perl 片段存成每個小檔案，即得 ionObj。

在自行建立演興表上，我們已經建立完成大氣掩星表部分，內容包含超額相位資料觀測資料在 opnGps 檔案內的記憶體位置，方便加速資料讀取，一次差分所需的輔助資料所需檔案資料，基本幾何觀測資料，等等如掩星發生時間，地點等等。這份掩星表將成為整個大氣反演資料處理流程的控制表。

5.0 - Ionospheric occ. process



圖九、CDAAC 中 5.0 電離層資料處理流程。

輸入檔	資料內容
gpsPod	GPS 軌道資訊檔，GPS 衛星質心之 J2000 坐標和速度 File format: Bernese STD format
podGps	含 GPS-LEO 衛星雙頻電碼相位觀測量，1 秒取樣，與 opnGps 同為 GOX 接收。用於一次差程序，也被用在 LEO POD 程序，估算 LEO 衛星軌道。 File format: BINEX format 0x7f format
leoAtt	LEO 衛星姿態參數，含衛星姿態四元數、LEO 衛星的 ECEF 坐標和速度。用來輔助 LEO POD 程序解算 LEO 軌道、輔助 TIP 程序計算 TIP 指向方向，並繪製 attPlt 圖檔 LEO 姿態檔，該檔為 GFZ 制定的衛星姿態檔案格式 File format: txt
occTab	大氣及電離層掩星表。包含資訊為：掩星發生事件資料下載(地點、時間) (one dump of data for one LEO) File format: netCDF format

doleoO	This file is used to schedule calling of the leoOrb program which computes LEO orbits (positions and velocities of a Low Earth Orbiter with respect to time). File format: Perl
--------	--

表一、為掩星表製作時所需之輸入檔之資料

輸出檔	資料內容
createIonOccObjs*.out	記錄 createIonOccObjs.pl 讀入及處理(暫存檔)
ionObj	Ionospheric occultation object file. Contains file names of all data needed to process an ionospheric occultation (excess phase + inversion).

表二、為掩星表製作時所需之輸出檔之資料

輸出檔範例:

createIonOccObjs*.out	<pre> \$conId = '1'; @gpsPodfiles = (/pub/cosmic/level1b/gpsPod/gpsPod_2010.305.00.00.2880_2010.2640_std'); @leoClkfiles = (/pub/cosmic/level1b/leoClk/leoClk_2010.305.001.01_2010.2640_clk'); \$leoId = '921'; @leoPodfiles = (/pub/cosmic/level1b/leoPod/leoPod_2010.305.001.01_2010.2640_std'); </pre>
-----------------------	--

	@lrfiles	=	(/pub/cosmic/level1a/podGps/2010.305/podGps_2010.305.001.01.01_bnx');
	@mrfiles	=	(");
	\$occId	=	'745';
	\$occPrn	=	'31';
	\$occtag	=	'C001.2010.305.22.35.G31';
	\$setting	=	'1';
	\$startTime	=	'972685715';
	\$stopTime	=	'972686155';

程式流程:

- 1.將輸入的參數檔案解出 occTab 與 doleoO 的相關檔案分別定義為 occTabfile 與 doleoOfile
- 2.使用 PubFile 剖析 occTabfile,並取得對應的下載資訊 dumpid 和 leoid.
- 3.取得 occTabfile 的版本資訊 (for creating ionObj)
- 4.使用 PDF::NetCF 開啟 occTabfile,並取得所需資料放到相對應的變數
(1)occ_id, early_occ,late_occ, occ_type, occ_sat,
ion_start_time,ion_stop_time,setting,occ_yr,occ_doy,occ_doy,occ_hr,occ_min.
- 5.將輸入的參數檔案解出 occTab 與 doleoO 的相關檔案分別定義為 occTabfile 與 doleoOfile
- 6.使用 PubFile 剖析 occTabfile,並取得對應的下載資訊 dumpid 和 leoid.
- 7.取得 occTabfile 的版本資訊 (for creating ionObj)

8.使用 PDF::NetCF 開啟 occTabfile,並取得所需資料放到相對應的變數

(1)occ_id, early_occ,late_occ, occ_type, occ_sat,

ion_start_time,ion_stop_time,setting,occ_yr,occ_doy,occ_doy,occ_hr,occ_min.

9. Loop function

10.使用 PubFile 開啟並依先前得到的資訊初始化 ionObj

(1)取得 occtab(s)

(2)指定是 SION_OCC 或 RION_OCC(setting)(s)

(3)取得天線 ID: lrAntID 和 mrAntID 並根據先前下載資訊與 podfiles,取得相對應的檔案(irfiles(s), mrfiles(s))

(4)將所獲得的資訊寫入 ionObj.

Excess phase(超額相位)

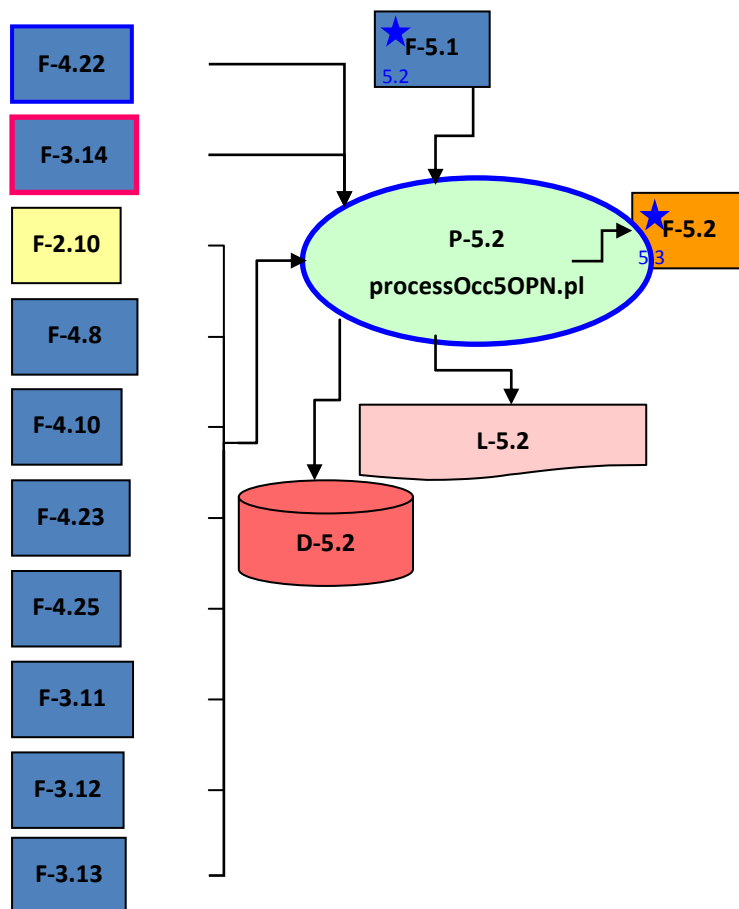
超額相位和是大氣超額相位以及電離層超額相位之和,但對大氣掩星而言,削減電離層貢獻量的步驟要到大氣反演程序才會進行;反之,電離層超額相位卻是以零次差估算的。由於後續的電離層反演法也只需相對的超額相位,因此周波未定值也忽略不計。

CDAAC中processOcc5OPN.pl程式主要估算大氣和電離層之超額相位。它同樣和Bernese 程式緊密結合,其中叫用多支Bernese 程式及類Bernese 程式進行演算。類Bernese 程式源自標準Bernese 程式,但經過CDAAC 人員增修而來。不論即時處理程序或後處理程序,皆採一次差演算程序估算大氣超額相位;以零次差估算電離層超額相位。

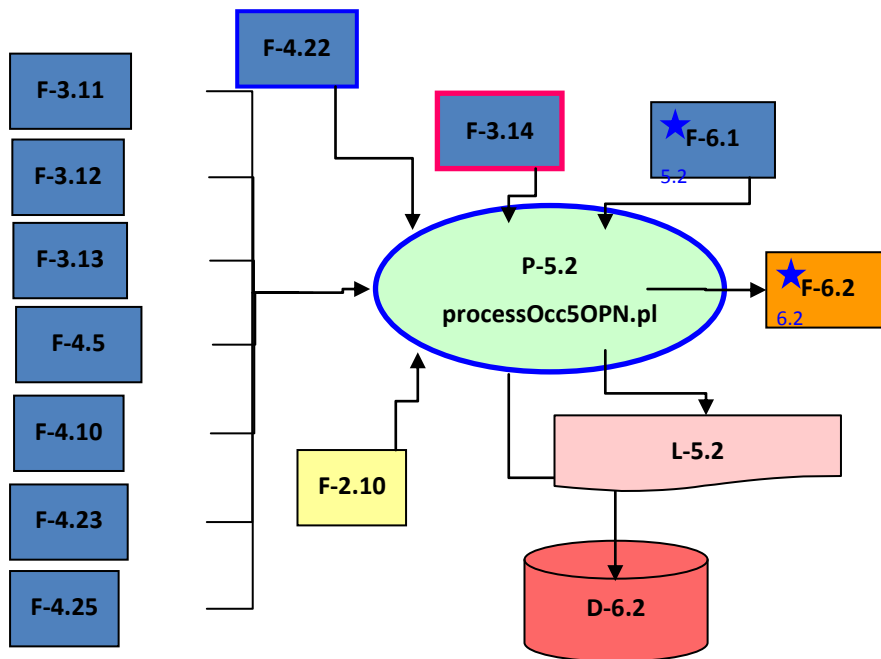
在大氣方面,本程式根據50 Hz 取樣的GPS 大氣觀測資料 (opnGps) 、低

軌衛星姿態檔(leoAtt)，配合POD 程序所得之GPS 和LEO 軌道(gpsPod 和leoPod)、GPS 時錶偏差值(即時comClr;後處理comClk)及LEO 時錶偏差值(leoClk)，估算大氣超額相位(atmPhs)。進行一次差的條件是，單一低軌衛星同時接收來自一顆掩蔽中的GPS 衛星的訊號和另一顆非掩蔽中的參考GPS 衛星的訊號。取掩星連線之相位觀測量與參考連線之無電離層相位觀測量做一次差，可消除LEO 時錶偏差項。其優點是可使GPS 觀測量中的誤差，例如多路徑效應、大氣殘差、熱雜訊(一種隨溫度與系統頻寬而變的內部雜訊)等，減至最小[Schreiner et al., 2005]。

在電離層方面，則以1 秒取樣的podGps 取代opnGps，配合上述資料估算電離層超額相位(ionPhs)。



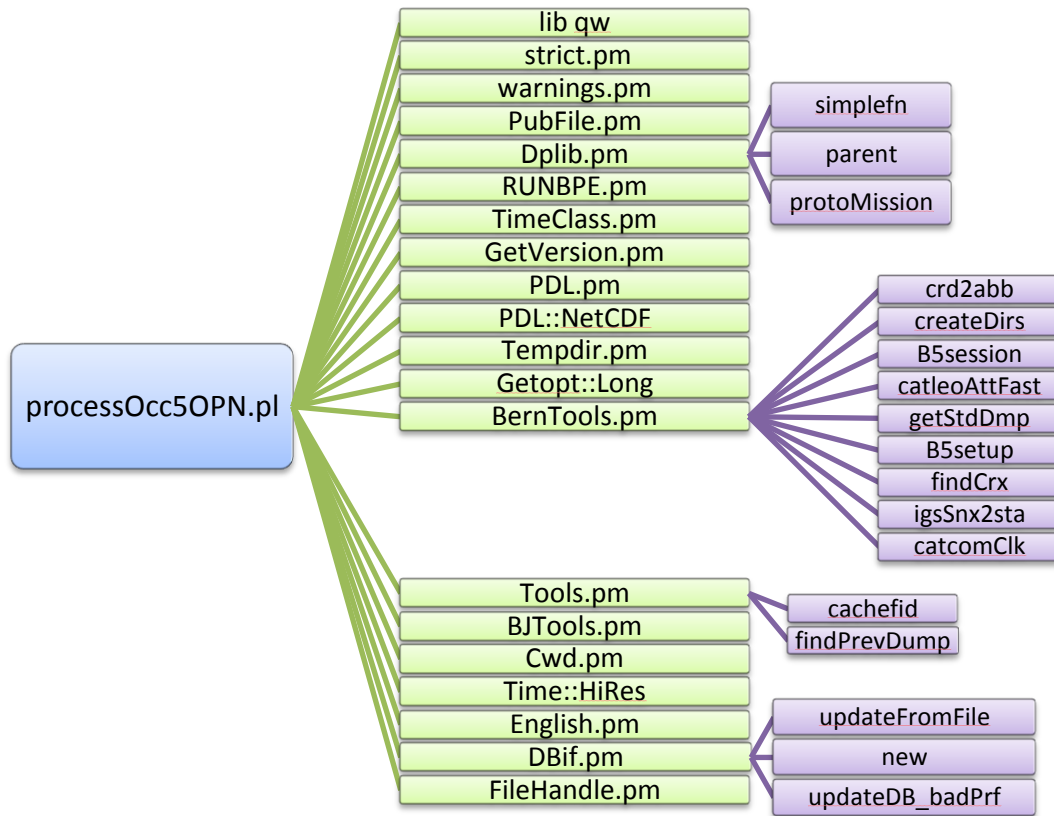
圖二、processOcc5OPN.pl此程式處理電離層資料之流程圖。



圖三、processOcc5OPN.pl此程式處理大氣層資料之流程圖。

processOcc5OPN.pl：建立電離層和大氣之超額相位檔。

本程式主要估算大氣和電離層之超額相位和。開迴路資料亦適用。這支程式可由二次差、一次差和零次差的計算方式處理資料，但目前指定以一次差的方式處理大氣資料；以零次差處理電離層資料，故只需叫用 GENPHS5OPN 程式。該程式雖為 CDAAC 所撰寫，但叫用多支 Bernese 或類 Bernese 程式以估算超額相位。類 Bernese 程式的命名規則很像 Bernese 程式，但實際上是 CDAAC 自行撰寫或修改自 Bernese 的程式。



圖三、processOcc5OPN.pl此程式所包含之perl模組。

輸入檔	資料內容
gpsPod	GPS 軌道資訊檔，GPS 衛星質心之 J2000 坐標和速度 File format: Bernese STD format
leoAtt	LEO 衛星姿態參數，含衛星姿態四元數、LEO 衛星的 ECEF 坐標和速度。用來輔助 LEO POD 程序解算 LEO 軌道、輔助 TIP 程序計算 TIP 指向方向，並繪製 attPlt 圖檔 LEO 姿態檔，該檔為 GFZ 制定的衛星姿態檔案格式 File format: txt
gpsClk	GPS 時錶偏差值，Bernese CLK 格式，取自 iguOrb
brnPol	地軸校正參數，Bernese ERP 格式
leoPod	低軌衛星質心之 J2000 坐標和速度，Bernese STD 格式

leoClk	低軌衛星時錶偏差值
comClk、 comClr	30 秒取樣之 GPS 衛星時錶偏差值 (前者 4 小時 1 檔；後者從前者取出最新資料，且涵蓋整個掩星觀測時間)
opnGPS	50 Hz 取樣的 GPS 大氣觀測資料 Atmospheric occultation data in a simple custom binary format. This data file contains all high rate atmospheric data sent us by the GNSS receiver on the LEO.
atmObj	Atmospheric occultation object file. Contains file names of all data needed to process an Atmospheric occultation (excess phase + inversion).
brnFil	含低軌衛星及天線名稱的參數檔，Bernese STA 格式。 在本程式中使用天線相位中心檔 (brnFil) 中，低軌衛星之掩星天線和參考天線名稱 (長 16 個位元組)。其中，掩星天線即為掩星連線上的 OCC 天線；參考天線即為參考連線上的 POD 天線。

表三、為超額相位製作時所需之輸入檔之資料

輸出檔	資料內容
processOcc5OPN *.out	記錄 processOcc5OPN.pl 讀入及處理(暫存檔)
atmPhs	大氣超量相位

表四、為超額相位製作時所需之輸出檔之資料

六、 自評表

原規劃目標	完成與否	說明
改善電離層校正方式，增加掩星資料反演比率	100%	
以一年以上的掩星觀測資料，進行乾溫度參數進行驗證並改進程式	100%	99.5% ， 6days for 1years atmphs profiles. 6sec/per RO
撰寫One-D VAR計算濕溫度參數程式(濕溫度，濕氣壓)	100%	已經完成由 R Langrange 撰寫的1DVAR 由折射率計算溼氣壓程式雛形，未來將輸入資料改為偏折角或超額相位，並改由 Fortran 程式撰寫。
改進周波脫落偵測方法，改善超額相位正確度與精度，及品質偵測(QC)程式	100%	

附錄

程式碼(processOcc5OPN.pl)

```
#!/ops/tools/bin/perperl
#
## Copyright (c) 1995-2004 University Corporation for Atmospheric Research
## All rights reserved
#
my $pkgdoc = <<'EOD'; # Save package level doc
#/**-----
# @file      processOcc5OPN.pl
#
# This is the script which creates one excess phase file given an occ object.
# It takes an atmospheric/ionospheric occ object as input and creates an
# atmPhs/ionPhs excess phase file as output.
# It also updates the database with information from the excess phase file.
# This script works for OPEN LOOP data.
#
# @author    Bill Schreiner
# @status
# @since     04/26/2005
# @version   $URL:
svn://ursa.cosmic.ucar.edu/branches/cdaac_3.1/src/genXphase5OPN/processOcc5OPN.pl $ $Id:
processOcc5OPN.pl 13792 2011-04-22 19:39:07Z vanhove $
# @cdaacTask no
# @usage     processOcc5OPN.pl [atm|ion]Obj_IIII.YYYY.DDD.HH.MM.GGG_SSS.VVVV_pl
[mission] [--parmsfile=parms1_pl]
# @          [--(no)db] [--(no)clean] [--(no)test] [--menu] [--dbonly] [--lvs=l08]
# @example   processOcc5OPN.pl atmObj_CHAM.2002.213.00.06.G09_0001.0002_pl
--parmsfile=processOcc5.parms1 --test
# @          Input is the name of an atmObj/ionObj file.
# @          By default, run script and clean out tmpdir. The mission can optionally be specified.
# @          If the mission is not specified, then the environment variable 'cdaac_mission' is
consulted
# @          instead. It is a good idea to specify the mission, since perperl can get confused if
# @          multiple missions are running at the same time.
# @          If --nodb is specified, do not populate the database.
# @          If --dbonly is specified, only populate the database.
# @          If --noclean is specified, then don't clean out tmpdir.
# @          If --test is specified, then use the GPSUSER directory in the CVS area
# @          otherwise (default) use GPSUSER from /ops/tools/parms
# @          If --lvs is set, then use specified antenna pcv's
# @          If --menu is specified, then do not run the Bernese programs,
# @          just set up the directory structure and call the MENU program
# @          this is used to edit the *.INP files for the program
# @          If --debug=pgm is set (eg --debug=GENPHS5OPN) then run up until this
# @          program is to be called, but instead of calling it, output a file containing
# @          the full path to the input file for that program. This allows easy debugging
# @          of that program (provided it is compiled with -g, of course!).
# @log      $log$
# -----*/
EOD
$|++;
#/** @call perlib::BernTools.pm:catcomClk */
#/** @call perlib::BernTools.pm:findCrx */
#/** @call perlib::BernTools.pm:igsSnx2sta */
```

```

#!/** @call perllib::Dplib.pm:protoMission */
#!/** @call perllib::Tools.pm:cachefid */
#!/** @call perllib::BernTools.pm:B5setup */
#!/** @call perllib::BernTools.pm:getStdDmp */
#!/** @call perllib::BernTools.pm:catleoAttFast */
#!/** @call perllib::BernTools.pm:B5session */
#!/** @call perllib::BernTools.pm:createDirs */
#!/** @call perllib::BernTools.pm:crd2abb */
#!/** @call database::DBif.pm:updateDB_badPrf */
#!/** @call database::DBif.pm:new */
#!/** @call perllib::Dplib.pm:parent */
#!/** @call perllib::Tools.pm:findPrevDump */
#!/** @call database::DBif.pm:updateFromFile */
#!/** @call perllib::Dplib.pm:simplefn */

#!/**使用方法: processOcc5OPN.pl [atm|ion]Obj_IIII.YYYY.DDD.HH.MM.GGG_SSSS.VVVV_pl
[mission] [--parmsfile=parms1_pl][--(no)db] [--(no)clean] [--(no)test] [--menu] [--dbonly] [--lvs=108] */
#!/** atm: 大氣層 ion 電離層 */
#!/** mission: 任務名稱*/
use lib qw(. /ops/tools/lib /ops/tools/bern5/BERN/BPE); # Add current directory to include path
use strict; # Uptight version of perl
use warnings;
use PubFile;
use Dplib;
use RUNBPE;
use TimeClass;
use GetVersion;
use PDL;
use PDL::NetCDF;
use Tempdir;
use Getopt::Long;
use BernTools;
use Tools;
use BJTools;
use Cwd;
use Time::HiRes qw(gettimeofday tv_interval);

use English;
use FileHandle;
STDOUT->autoflush(1); # Set pipes to immediate flush
STDERR->autoflush(1); # Set pipes to immediate flush

no warnings 'once'; # get rid of 'used only once' warnings for variables in obj:: and parms::
# 如果沒有檔案則跳出
if (@ARGV < 1) {
    print $pkgdoc;
    exit -1;
}

#
## Deal with command line options
# 處理 command line 的選項, 以下為選項預設值

my $parmsfile = 'processOcc5.parms1'; # default
my $lvs = ""; # use absolute pcv's
my $dbonly = 0;
my $db = 1; # populate database
my $clean = 1; # clean out tmpdir

```

```

my $test    = 0; # set to 1 to test with GPSUSER from CVS directory. Set to 0 for production use
from /ops/tools/parms
my $menu    = 0; # set to 1 to run the menu program, useful in filling out *.INP files
my $debug   = ""; # set to the name of the Bernese FORTRAN program to debug
GetOptions ("parmsfile=s" => \$parmsfile,
           "db!"      => \$db,
           "donly!"  => \$donly,
           "clean!"  => \$clean,
           "menu!"   => \$menu,
           "lvs=s"   => \$lvs,
           "debug=s" => \$debug,
           "test!"   => \$test) || die "Cannot parse command line options!";

# Input file
# infile 為輸入的那串指令, 如果沒有定義 mission, 則使用環境變數中的 mission 值取代
my $infile = shift;
my $mission = shift;

$ENV{cdaac_mission}=$mission if(defined($mission));

# Check mission
die "Please set environment variable: cdaac_mission to eg champ (or specify mission on the command
line)"
  unless (defined($ENV{cdaac_mission}));

# 檢查 mission 是否設定正確
do "/ops/tools/lib/DBif.pm"; # do this at run time to make sure mission is set correctly!

# netcdf object
# 定義 ncobj 物件, 之後使用
my ($ncobj);

#
## Extract some initial vars.
# 使用 PubFile 模組解析命令列, 並將解析結果填入各欄位 ($filetype, $yr, $doy, $leoid, $hr, $min,
$occsat, $occtag, $leoname)
my $pub = PubFile->parse($infile);
my $inpath = $pub->fullpath;
die "Cannot find input file: $infile" unless (-e $inpath);

my ($filetype, $yr, $doy, $leoid, $hr, $min, $occsat, $occtag, $leoname) =
  ($pub->filetype, $pub->yr, $pub->doy, $pub->leoid, $pub->hr, $pub->min, $pub->occsat,
  $pub->occtag, $pub->leoname);
my $leonum = $BernTools::satidmap{$leoname};
my $yr2d = sprintf("%02d", TimeClass::two_digit_year($yr));

print "Starting processOcc5OPN for File = $infile\n";

# 將 gettimeofday 塞入 t0 變數, 於執行 script 時使用
my $t0 = [gettimeofday];

# use prototype mission
# 再次確認 mission, uc 為 upper case, 把字串全變大寫
$mission = $ENV{cdaac_mission};
my $missionuc = uc $mission;

```

```

#protoMission : 由任務名稱取出原任務名稱，例如從任務 cosmicrt 中取出原任務名稱 cosmic。
my $protomission = Dplib::protoMission($mission);
my $protomissionuc = uc $protomission;
# 改成大寫的兩個變數似乎未使用到.....( missionuc, protomissionuc)
## Read in occ obj
package obj;
do $inpath;
package main;

#
## Read in parms
# If the user has the OPSTOOLS environment variable set, use parms files from there.
# 選擇 OPSTOOLS 目錄
my $parmsdir = $Dplib::parmsdir;
$parmsdir =~ s|/ops/tools|$ENV{OPSTOOLS}| if (exists($ENV{OPSTOOLS}));

# 如果代有選項 --test (test mode) , 則使用當下目錄
# If the user runs in --test mode, use local parms file D. Hunt 1/12/2009
my $srcdir = cwd();
$parmsdir = Dplib::parent($srcdir) if ($test);

die "Cannot find input parmsfile: $parmsfile" unless (-e "$parmsdir/genXphase5OPN/$parmsfile");
package parms;
# Use 'do' instead of 'require' since this forces the parms file to be read every time.
# This is important when using perperl! D. Hunt 2010.1.11
do "$parmsdir/genXphase5OPN/$parmsfile";
package main;

#
## Set variable for processing with absolute/relative phase centers
# 設定環境變數, absolute/relative phase centers....(For Bernese)
my $satabsrel = $lvs ? "SATELLIT.${lvs}" : "SATELLIT.";
my $phafilerelel = $lvs ? "${missionuc}.PHA.${lvs}" : "${missionuc}.PHA";

#
## Get subver, Output file
# 選則輸出 file 的 type 是 atmPhs 或 ionPhs
my $outfiletype = ($filetype =~ /atm/) ? "atmPhs" : "ionPhs";
# 由 GetVersion::GetVersion 決定輸出的 sub version
my $subver = GetVersion::GetVersion ('genXphase5OPN', $parmsfile, $outfiletype, $infile);
my ($subType, $version) = split(/\./,$subver);
# 根據以上資訊, 使用 PubFile->new 產生名稱為 outfile 的 PubFile 物件
my $outfile = PubFile->new (FILETYPE => $outfiletype, OCCTAG => $occtag, SUBVER =>
$subver)->fullpath;

# 如果有定義 dbonly, 則去做 UPDATEDB function( 本 Perl 最後幾行 )
goto UPDATEDB if ($dbonly);

# Create temp directory
# 建立 temp 目錄..
my $tmpdir;

# use eval to trap errors. close after execution.
# eval :將大括號裡面的字串視為 Perl 指令執行, 若有錯誤則印出
eval {

## Assemble LEO info

```

```

my $prochist = 'GENPHS5 1.0';

#
## Format times
# 初始化 startObj , stopObj 為 0,0
# 並根據 obj::setting, 以 TimeClass 初始並設定 startObj 與 stopObj
my ($startObj,$stopObj) = (0,0);
if ($obj::setting == 1) {          # Setting
    $startObj = TimeClass->new->set_gps($obj::startTime->inc_sec_gps(-$parms::occOverTop);
    $stopObj  = TimeClass->new->set_gps($obj::stopTime->inc_sec_gps($parms::occOverBot);
} else {                          # Rising
    $startObj = TimeClass->new->set_gps($obj::startTime->inc_sec_gps(-$parms::occOverBot);
    $stopObj  = TimeClass->new->set_gps($obj::stopTime->inc_sec_gps($parms::occOverTop);
}

# 取得 startGps 與 stopGps
my $startGps = $startObj->get_gps;
my $stopGps  = $stopObj->get_gps;

# 由 startObj 取出 start Year, start Day of Year, Hour, Minute, Second
my ($startYr, $startDoy, $hour, $minute, $second) = $startObj->get_yrdoyhms_gps;
# 由 start year 和 start day of year 求出 month 跟 day ..
my ($month, $day) = TimeClass::find_date ($startYr, $startDoy);

# 求出 開始日期 date 與結束日期 enddate
my $date    = $startObj->get_yrdoy_gps;
my $enddate = $stopObj->get_yrdoy_gps;

# 定義 dates 陣列, 並將 date, enddate, enddatep1 全部放入此陣列中
my @dates = ($date);
my $enddatep1 = TimeClass->new->set_gps($stopGps->inc_sec_gps(86400)->get_yrdoy_gps;
push @dates, $enddate unless ($enddate == $dates[0]);
push @dates, $enddatep1;

#
## Fiducial data overlap
# 算出 startGps 與 stopGps
$startGps += -$parms::fidOver+1;
if ($parms::proctype == 0) {          # DOUBLE DIFF
    $stopGps += $parms::fidOver-1;
} else {
    $stopGps += $parms::fidOver;
}
## Call getStdDmp to get gpsPod files for processing
# 根據 startGps 與 stopGps , 使用 BernTools 求出 gpsPolfile, gpsClkfile, gpsPodfile 供後續程式使用
my ($gpsPolfile, $gpsClkfile, $gpsPodfile) = BernTools::getStdDmp($startGps, $stopGps);

#
## Decide whether to process delta or double differences
# 決定 Delat 或 Double difference, 以下為初始預設值...
my $fidtrpfile = "";
my $fidtrptext = 1;
my %binexfiles = ();
my @scriptList = ();
my ($obsfil1,$obsfil2) = ("",""); # occ
my ($obsfil1r,$obsfil2r) = ("",""); # ref
my $occantid='00';
my $refantid='00';

```

```

# 取出檔名
my $scriptname = Dplib::simplefn($0);
if ($test) { #測試 mode
    my $foo = "$srcdir/TEMPDIR";
    $tmpdir = \$foo;
    mkdir $$tmpdir || die "Could not make TEMPDIR in " . cwd() . "\n";
} else { # 正常 mode
    $tmpdir = Tempdir->new($scriptname);
    $Tmpdir::delete = 0 unless ($clean); # remove temp dir after run unless --noclean is specified
}

system "rm -rf $$tmpdir/*"; # start from scratch!清除 temp 目錄

# Double differences 二次差 (但在 CDAAC 中未使用)
if ($parms::proctype == 0) { # DOUBLE DIFF

    print "Double Difference Processing\n";

    # Fiducial Data Ready
    ## Get fiducial trop file
    # 由 startGPS, stopGps 時間內尋找 fidTrp, 並回傳一資料結構給 filecahce.
    my $filecache = Tools::cachefid("fidTrp", $startGps, $stopGps);
    # 將 filecahce 所有符合條件的物件列出 which(($filecache->{START} <= $stopGps) &
    ($filecache->{END} >= $stopGps))
    my @trpfiles = map { $filecache->{LIST}[$_] } list which(($filecache->{START} <= $stopGps) &
    ($filecache->{END} >= $stopGps));
    unless (@trpfiles) {
        my $tStartStamp = TimeClass->new->set_gps($startGps)->get_stamp_gps;
        my $tStopStamp = TimeClass->new->set_gps($stopGps)->get_stamp_gps;
        die "No fidTrp file found covering the time span from $tStartStamp to $tStopStamp";
    }

    $fidtrpfile = $trpfiles[$#trpfiles];
    $fidtrptext = {"BERNTRP.TRP" => $fidtrpfile};
    # 根據 mission, 定義要執行的 script
    @scriptList = ($protomission eq 'gpsmet') ? qw(BXOBV3 GENPHS5OPN) : qw(BXOBV3 CODSPP
    GENPHS5OPN);
    die "Cannot process atmGps files, only opnGps files. Please use processOcc5.pl" if
    ($obj::occhrfiles[0] =~ /atmGps/);

    #
    ## Get BINEX fiducial data ready for bernese
    # 準備給 bernese 的 binex 地面站資料
    # 取得站名
    my ($stat) = ($obj::fidGpsfiles[0] =~ m/fidGps_(\w\w\w\w)\./);
    die "Cannot determine station name from input file: $obj::fidGpsfiles[0]" unless defined($stat);

    my $dat = "";
    if (@obj::fidGpsfiles) {
        my $offset = 0;
        foreach my $file (@obj::fidGpsfiles) {
            my $realfile = PubFile->parse(Dplib::simplefn($file))->fullpath; # swap in current mission,
            PUBNAME D. Hunt 1/12/2009
            open (IN, $realfile);
            my $size = -s $realfile;
            read (IN, $dat, $size, $offset);
            close IN;
        }
    }
}

```

```

    $offset += $size;
  }
  $binexfiles{"${stat}${doy}A.${yr2d}O"} = sub {$dat};

  if (defined($obj::occhrfiles[0]) && $obj::occhrfiles[0]) {
    my $pub = PubFile->parse($obj::occhrfiles[0]);
    $occantid = $pub->ant;
    $obsfil1 = $obj::occhrfiles[0];
    $obsfil2 = $obj::occhrfiles[1] if (defined($obj::occhrfiles[1]));
  }
  if (defined($obj::refhrfiles[0]) && $obj::refhrfiles[0]) {
    my $pub = PubFile->parse($obj::refhrfiles[0]);
    $refantid = $pub->ant;
    $obsfil1r = $obj::refhrfiles[0];
    $obsfil2r = $obj::refhrfiles[1] if (defined($obj::refhrfiles[1]));
  }
} else {
  die "No fiducial files found for prefix fidGps, station $stat";
}
}
#一次差 with hr clocks/ 零次差
elsif ($parms::proctype == 1 || $parms::proctype == 3) { # delta diff with hr clocks or zero-difference
processing

  if ($parms::proctype == 1) {
    print "Delta Difference Processing with HR clocks\n";
  } else {
    print "Zero-difference processing\n";
  }

  # occClk clk file
  my @clkfiles = ();
  $gpsClkfile = "$$tmpdir/BERNCLK.CLK";

  if ($ENV{versionMode} =~ /real/) {
    # 產出所需檔案
    my $pub = PubFile->parse($obj::leoPodfiles[$#{ $obj::leoPodfiles }]);
    my $clrsbver = GetVersion::GetVersion ('leoOrb5', "", "comClr");
    @clkfiles = PubFile->new(FILETYPE => 'comClr', DUMPID => $pub->dumpid => SUBVER =>
$clrsbver)->fullpath;
    die "No comClk files found" unless (@clkfiles);
    # 建立連結
    symlink $clkfiles[0], $gpsClkfile;

  } else {
    ## Use cachefid to get list of comClk files to concatenate together.
    my $filecache = Tools::cachefid ("comClk", $startGps, $stopGps);
    @clkfiles = map { $filecache->{LIST}{$_} } list which((( $filecache->{START} <= $startGps) *
($filecache->{END} >= $stopGps)));
    die "No comClk files found" unless (@clkfiles);

    # cat comClk files together
    if (@clkfiles > 1){
      BernTools::catcomClk($gpsClkfile, \@clkfiles);
    } else {
      symlink $clkfiles[0], $gpsClkfile;
    }
  }
}

```



```

}
# 定義 scriptList 為 GENPHS5OPN
@scriptList = qw(GENPHS5OPN);

die "Cannot process atmGps files, only opnGps files. Please use processOcc5.pl" if
($obj::occhrfiles[0] =~ /atmGps/);

if (defined($obj::occhrfiles[0]) && $obj::occhrfiles[0]) {
    my $pub = PubFile->parse($obj::occhrfiles[0]);
    $occantid = $pub->ant;
    $obsfil1 = $obj::occhrfiles[0];
    $obsfil2 = $obj::occhrfiles[1] if (defined($obj::occhrfiles[1]));
}
if (defined($obj::refhrfiles[0]) && $obj::refhrfiles[0]) {
    my $pub = PubFile->parse($obj::refhrfiles[0]);
    $refantid = $pub->ant;
    $obsfil1r = $obj::refhrfiles[0];
    $obsfil2r = $obj::refhrfiles[1] if (defined($obj::refhrfiles[1]));
}
}
# 電離層, 或一次差 with IGS clocks
elsif ($parms::proctype == 2) { # IONOSPHERE, OR deltadiff with IGS clks

    # must convert full path mrfiles and lrfiles to local PUBNAME and mission for testing
    # D. Hunt 1/12/2009
    my $mrfiles0 = eval { PubFile->parse(Dplib::simplefn($obj::mrfiles[0]))->fullpath };
    my $lrfiles0 = eval { PubFile->parse(Dplib::simplefn($obj::lrfiles[0]))->fullpath };
    if (defined($mrfiles0) && (-s $mrfiles0) && (!defined($parms::skipmrdata))) {
        my $pub = PubFile->parse($obj::mrfiles[0]);
        $occantid = $pub->ant;
        $obsfil1 = $mrfiles0;
        if (defined($obj::mrfiles[1])) {
            $obsfil2 = PubFile->parse(Dplib::simplefn($obj::mrfiles[1]))->fullpath;
            $obsfil2 = undef if (!-s $obsfil2);
        }
    }
    elsif (defined($lrfiles0) && -s $lrfiles0) {
        my $pub = PubFile->parse($obj::lrfiles[0]);
        $occantid = $pub->ant;
        $obsfil1 = $lrfiles0;
        if (defined($obj::lrfiles[1])) {
            $obsfil2 = PubFile->parse(Dplib::simplefn($obj::lrfiles[1]))->fullpath;
            $obsfil2 = undef if (!-s $obsfil2);
        }
    }
    } else {
        die "No MR or LR LEO Data Available\n";
    }
}
# 定義 scriptList 為 GENPHS5OPN
@scriptList = qw(GENPHS5OPN);
print "Ionospheric Occultation Processing, L4=L1-L2 OR\n";
print "Delta Difference Processing with IGS Clocks\n";
}

## Set up necessary environment variables and the standard options hash
## for RUNBPE

```

```

#
# 定義所有 BernTools::B5setup 所需資訊
# Save time objects
my $tstart = $startObj;
my $tstop = $stopObj;

# Determine if an attitude file is available. If so, set up INP file
# and directory structure to point to it. D. Hunt 9/22/2004
# Get a list of the names of the leoAtt files from /pub needed for this run
# If leoAtt files unavailable, try getting level0/chpAcc files
# If no attitude files found, then use default pointing
my @leoattfiles = ();
my $attfile;
my ($Time, $Att);

# Do this in post-processing only. D. Hunt 11/22/2006
if ($ENV{versionMode} =~ /post/) {
    for my $dat (@dates) {
        push (@leoattfiles, sort PubFile->new(FILETYPE => "leoAtt", DATE => $dat, LEO =>
$leoid)->glob);
    }
} else { # For real time, only use current and previous dump files. D. Hunt 11/22/2006
    my $dumpid = PubFile->parse($obj::leoPodfiles[-1])->dumpid;
    my $prevdumpid = Tools::findPrevDump ($dumpid);
    push (@leoattfiles, PubFile->new(FILETYPE => "leoAtt", DUMPID => $prevdumpid)->glob)
        if (defined($prevdumpid));
    push (@leoattfiles, PubFile->new(FILETYPE => "leoAtt", DUMPID => $dumpid)->glob);
}
if (@leoattfiles != 0) {
    $attfile = "$tmpdir/BERNLEO.ATT";
    # Uncomment for old (perhaps more correct, but very slow) catleoAtt
    # ($Time, $Att) = BernTools::catleoAtt(\@leoattfiles, $leoid, $attfile, $startGps-10, $stopGps+10,
"att", "rat", "sad", "ang");

    # New fast catleoAtt written in C
    BernTools::catleoAttFast(\@leoattfiles, $leoid, $attfile, $startGps-10, $stopGps+10);
} else {
    # Post-Processed, chpAcc or leoAtt
    $attfile = PubFile->new(FILETYPE => "chpAcc", YR => $yr, DOY => $doy,
        HR => 0, MIN => 0, DUR => 1440)->fullpath('nocreate');
}

my @attinfo = ();
if (-s $attfile) {
    @attinfo = ('LEO.ATT' => $attfile); # for createDirs
} else {
    print "\nNo leoAtt files found ..... USING Default Attitude\n\n";
}

# For GPS/MET, use L1P1 phase not L1CA.
my $usel1ca = ($protomission eq 'gpsmet') ? 0 : 1;

# For GPS/MET, use L1
my $usel1cafid = ($protomission eq 'gpsmet') ? 0 : 1;
my $usel1cafid = 0;

# For GPS/MET, Minimum epochs
my $minepo = ($protomission eq 'gpsmet') ? 0 : 30;

```

```

# REFSAT for ZD case
my $refprn = (!defined($obj::refPrn)) || ($obj::refPrn == -999) ? 0 : $obj::refPrn;

# GPSUSER is in the CVS directory if --test is specified, otherwise its in /ops/tools/parms
my $gpsuser = $test ? "$srcdir/GPSUSER" : "$Dplib::parmsdir/genXphase5OPN/GPSUSER";

# Switch in the correct PUBNAME and MISSION. Do not use the full path
# name from the atmObj. D. Hunt 1/8/2009
$obsfil1 = PubFile->parse(Dplib::simplefn($obsfil1))->fullpath if ($obsfil1);
$obsfil2 = PubFile->parse(Dplib::simplefn($obsfil2))->fullpath if ($obsfil2);
$obsfil1r = PubFile->parse(Dplib::simplefn($obsfil1r))->fullpath if ($obsfil1r);
$obsfil2r = PubFile->parse(Dplib::simplefn($obsfil2r))->fullpath if ($obsfil2r);

my $opts = BernTools::B5setup ($gpsuser, $tstart, $$tmpdir,
    {(MISSION => $missionuc,
    USEL1CA => $usel1ca, #0
    USEL1CAFID => $usel1cafid, #0
    MINEPO => $minepo, # 0
    LEONAME => $leoname,
    LEONUM => $leonum,
    PROCTYPE => $parms::proctype,
    OBSFIL1 => $obsfil1,
    OBSFIL2 => $obsfil2,
    OBSFIL1R => $obsfil1r,
    OBSFIL2R => $obsfil2r,
    GENRES => "$$tmpdir/GPSDATA_Q/THIS/ORB/ex.nc",
    STARTDOY => $doy, # day of year
    LI1 => $parms::ilink[0],
    LI2 => $parms::ilink[1],
    LI3 => $parms::ilink[2],
    LI4 => $parms::ilink[3],
    LS1 => $parms::isml4[0],
    LS2 => $parms::isml4[1],
    LS3 => $parms::isml4[2],
    LS4 => $parms::isml4[3],
    LN1 => $parms::isml4n[0],
    LN2 => $parms::isml4n[1],
    LN3 => $parms::isml4n[2],
    LN4 => $parms::isml4n[3],
    L4WINOPT => $parms::l4winopt,
    INOCHKL2SLIP => $parms::inochk2slip // 0,
    IREFL3L1 => $parms::irefl3l1 // 0,
    ISACK => $parms::isack,
    FIXREFGAP=> $parms::fixrefgap,
    FIXREFOUT=> $parms::fixrefout,
    REFMAXTOL=> $parms::refmaxtol,
    IDUR => $parms::idur,
    NDEG => $parms::ndeg,
    CFITTOL => $parms::cfittol,
    IRL => $parms::intreflink,
    SUBTYPE => $subType,
    VERSION => $version,
    SETTING => $obj::setting,
    LEOID => $leoid,
    DUMPID => "1",
    GNSSID => $obj::conId,
    OCCSAT => $obj::occPrn,
    REFSAT => $refprn,

```

```

        SAMPLING => $parms::fsamp,
        MAXZEN   => $parms::fmaxzen,
                SATABSREL=> $satabsrel,
        STARTGPS => ($tstart->get_mjd),
        STOPGPS  => ($tstop->get_mjd));

$oopts{FASTMENU} = 1; # speed processing
# $oopts{DEBUG} = 1; # Turn off debug to reduce number of kids spawned
$oopts{NOCHDIR} = 1;

#
## Specify and create directory hierarchy for BPE.
## (mostly just the CAMPAIGN directory).
## $tree contains a hash structure which is used by createDirs to create the necessary
## directory tree.

# Make sure session includes fiducial overlap
$tstart->inc_sec_gps(-$parms::fidOver+1);
if ($parms::proctype == 0) { # 二次差
    $tstop->inc_sec_gps($parms::fidOver-1);
} else {
    $tstop->inc_sec_gps($parms::fidOver);
}
#
## Only include station info for double difference processing. D. Hunt 2/21/2007
#
my $crdfile_string;
my @stainfo = ();
if ($parms::proctype == 0) {
    my $datep = TimeClass->new->set_yrdoym_gps($date)->inc_sec_gps(-86400)->get_yrdoym_gps;
    if ($ENV{versionMode} =~ /real/) {
        $crdfile_string = ReadFidTab5::makeBerneCrd($tstart->get_gps, 'IGS00', {DISTANCE =>
3650});
    } else {
        $crdfile_string = ReadFidTab5::makeBerneCrd($tstart->get_gps, 'IGS00', {PRECISE => 1,
DISTANCE => 3650});
    }
    @stainfo = (
        "BERNCRD.CRD" => sub { $crdfile_string },
        "BERNABB.ABB" => sub { BernTools::crd2abb($crdfile_string) },
        "BERNSTA.STA" => sub { BernTools::igsSnx2sta($datep) }
    );
}

my $pubname = defined($ENV{PUBNAME}) ? $ENV{PUBNAME} : '/pub';

# Get leoclk and leopod full paths using the correct PUBNAME and MISSION from
# environment variables. Do not use the full path from the atmobj!
# This allows one to run this in a test configuration
# D. Hunt, 1/8/2009
# 根據 PUBNAME 和 MISSION 取得 leoclok 和 leopod 的完整路徑
my $leoClkfile = PubFile->parse(Dplib::simplefn($obj::leoClkfiles[0]))->fullpath;
my $leoPodfile = PubFile->parse(Dplib::simplefn($obj::leoPodfiles[0]))->fullpath;
my $tree =
{
    STATUS    => sub {}, # empty file
    GPSTEMP   => {AUTO_TMP => 1},
    GPSDATA_Q => {THIS => {ATM    => $fidtrptext,

```

```

        DATPAN => 1,
        OBS    => 1,
        OUT    => 1,
        ORB    => {"GPS.CLK"      => $gpsClkfile,
                  "LEO.CLK"      => $leoClkfile,
                  "GPS.STD"      => $gpsPodfile,
                  "LEO.STD"      => $leoPodfile,
                  @attinfo,
                  "GPS.ERP"      => $gpsPolfile},
        SOL    => 1,          # map of Bernese names for input normal
equations files to CDAAC names
        RAW    => \%binexfiles,
        STA    => { @stainfo,
                  "BERNPHA.PHA" => "cp
$pubname/$mission/config/brnFil/$phafilereI",
                  "BERNCRX.CRX" => BernTools::findCrx($yr, $mission),
                  "LEOCC.STA"  => "cp
$pubname/$mission/config/brnFil/${missionuc}.$occantid.STA",
                  "LEOREF.STA" => "cp
$pubname/$mission/config/brnFil/${missionuc}.$refantid.STA",
                  "BERNSTA.STA" => sub { BernTools::igsSnx2sta($date) },
                  "ITRFCODE.BLQ" => "cp $ENV{X}/GEN/ITRFCODE.BLQ",
                  "SESSIONS.SES" =>
                    sub { BernTools::B5session
("$ENV{X}/SKL/SESSIONS.SES", $tstart, $tstop) }}}
};

# Create a directory hierarchy, including symlinks to files and directories and
# generated files under $$tmpdir according to $tree.
# 建立目錄，並建立 tree 與 tmpdir 連結
BernTools::createDirs ($$tmpdir, $tree);
system "chmod -R 0777 $$tmpdir";

#
## Now call the menu program (and exit) if requested by the user
#
#-----
#Useful for debug...
#chdir ("${ENV{XQ}}");
#foreach my $key (sort keys %ENV) { print "$key => ${ENV{$key}}\n"; }
#exec "gdb ${ENV{XQ}}/menu" if ($menu);
#-----
MENU:
exec "${ENV{XQ}}/menu ${ENV{U}}/PAN/MENU.INP" if ($menu);

printf "Elapsed = %5.2f\n", tv_interval ($t0); $t0 = [gettimeofday];

#
## Run all scripts in newly created processing area
#
# Set things up for debugging the specified program
if ($debug) {
    $opts->{DEBUG_PGM} = $debug;
    $opts->{DEBUG} = 1;
}

```

```

my $pid = 1;
# Run all scripts
#####
# 執行 script! (GENPHS5OPN)
for my $script0 (@scriptList) {
    my ($script, $optdir) = split /\_/, $script0;
    $optdir = "THIS" unless (defined($optdir));
    $$opts{OPT_DIR} = $optdir;
    print "Executing script: $optdir:$script...";
    $$opts{FASTMENU} = 0 if !($script eq "GENPHS5OPN"); # no speed processing
    $$opts{CMD_PATH} = '/ops/schrein/bin' if (exists($ENV{OPSTOOLS}) && $ENV{OPSTOOLS} =~
/schrein/ && $script eq "GENPHS5OPN");

    # Run local build of GENPHS5OPN if --test. D. Hunt 12/31/2008
    $$opts{CMD_PATH} = '.' if ($test && $script eq "GENPHS5OPN");

    RUNBPE->new(%$opts, SCRIPT => $script, PID => $pid++)->copyUarea->run;
    printf "Elapsed = %5.2f\n", tv_interval($t0); $t0 = [gettimeofday];
    if ($script eq 'CODSPP') {
        die "ProcessOcc5.pl ERROR: BAD CODSPP Solution" if (-s
"$tmpdir/GPSDATA_Q/THIS/OUT/CODXTR.DEL");
    }
}
}; # close of eval for trapping errors
#列出錯誤訊息
if ($?) {
    my $errstr = $?;
    # Add failed attempt to badPrf database
    DBif->new->updateDB_badPrf($outfile, $errstr)->close if (!$test);
    die $errstr;
}

# Update badPrf db if file not created.
# 如果執行錯誤, 則更新 badPrf db.
unless (-s "$tmpdir/GPSDATA_Q/THIS/ORB/ex.nc") {
    # Add failed attempt to badPrf database
    DBif->new->updateDB_badPrf($outfile, "Excess phase file not created")->close if (!$test);
    die "Excess phase file not created";
}

#
### Get info from netCDF file

my $bad = 0;
my $errstr = "";

eval {
    # PDF::NetCDF->New 一個 NetCDF 的物件, 指向 ncobj
    $ncobj = PDL::NetCDF->new ("$tmpdir/GPSDATA_Q/THIS/ORB/ex.nc");
    # 取得 NetCDF 物件 (ncobj)的資訊
    my $l2min = $ncobj->getatt('l2min')->at(0);
    my $refrms = $ncobj->getatt('refrms')->at(0);
    my $refmax = $ncobj->getatt('refmax')->at(0);
    my $fidoccrms = $ncobj->getatt('fidoccrms')->at(0);
    my $fidoccmx = $ncobj->getatt('fidoccmx')->at(0);
    my $fidrefrms = $ncobj->getatt('fidrefrms')->at(0);
}

```

```

my $fidrefmax = $ncobj->getatt('fidrefmax')->at(0);
# 檢查資訊是否符合範圍, 若不符合則印出 errstr, 並將 bad 設為 1, 以供接下來程式輸出錯誤訊息
if ($l2min < $parms::minrefl2snr) {
    $errstr .= "Failed: LOW L2 SNR ON REFERENCE LINK: $l2min < $parms::minrefl2snr";
}
if ($refrms > $parms::refrmstol) {
    $errstr .= "Failed: BAD L4 RMS FOR LEO REFSAT LINK: $refrms > $parms::refrmstol";
}
if ($refmax > $parms::refmaxtol) {
    $errstr .= "Failed: BAD L4 MAX RESIDUAL FOR LEO REFSAT LINK: $refmax >
$parms::refmaxtol";
}
if ($fidrefrms > $parms::fidrmstol) {
    $errstr .= "Failed: BAD L4 RMS FOR FIDUCIAL REFSAT LINK: $fidrefrms > $parms::fidrmstol";
}
if ($fidoccrms > $parms::fidrmstol) {
    $errstr .= "Failed: BAD L4 RMS FOR FIDUCIAL OCCSAT LINK: $fidoccrms > $parms::fidrmstol";
}
if ($fidrefmax > $parms::fidmaxtol) {
    $errstr .= "Failed: BAD L4 MAX RESIDUAL FOR FIDUCIAL REFSAT LINK: $fidrefmax >
$parms::fidmaxtol";
}
if ($fidoccmx > $parms::fidmaxtol) {
    $errstr .= "Failed: BAD L4 MAX RESIDUAL FOR FIDUCIAL OCCSAT LINK: $fidoccmx >
$parms::fidmaxtol";
}

if ($errstr) {
    $bad = 1;
} else {
    $errstr = "OK";
}

# Write to netcdf file.
# 將其餘資訊寫入 NetCDF 檔案並儲存
$ncobj->putatt ($occtag, 'fileStamp'); # add complete occ id as global attribute
$ncobj->putatt ($parmsfile,'parmsfile'); # add parameter file name as global attribute
$ncobj->putatt ('UCAR/CDAAC','center'); # add processing center as global attribute
$ncobj->putatt ($bad, 'bad');
$ncobj->putatt ($errstr, 'errstr');
$ncobj->close;
};
# 列出錯誤訊息
if ($?) {
    $errstr .= "Failed: Cannot read netCDF file: $?";
    system "mv $$tmpdir/GPSDATA_Q/THIS/ORB/ex.nc $outfile.unreadable";
    # Add failed attempt to badPrf database
    DBif->new->updateDB_badPrf($outfile, $errstr)->close if (!$test);
    die $errstr;
}
# 如果執行錯誤, 則更新 badPrf db.
if ($bad) {
    print "Badness found: \n$errstr\n";
    # Add failed attempt to badPrf database
    DBif->new->updateDB_badPrf($outfile, $errstr)->close if (!$test);
    die $errstr;
}
}

```

```
# Write file to pub, print "FILE CREATION".
# 建立 File.
system "rm -f $outfile" if (-e $outfile);
system "mv $$tmpdir/GPSDATA_Q/THIS/ORB/ex.nc $outfile";
print "FILE CREATION: ", Dplib::simplefn($outfile), "\n";

#
## Update the database for this occ
#
# 如果有定義 , 則做以下程式 UPDATEDB
UPDATEDB:

# update the database
DBif->new->updateFromFile($outfile)->close if (!$test);

print "Done!\n";

exit;
```