```
pro SO2exosphere_distribution, input, output, npack, seed

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;
;; O source rate based on SO2 exosphere modeled by Vincent Dols. See notes.
;;
;; Written by Matthew Burger
;;
;; Version History
;;   3.1  11/23/201
;;    * 2nd try
;;   3.0: 11/23/2010
;;    * initial version - doesn't work
;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

num = n_elements(x)
case (1) of
  stregex(input.spatialdist.size, 'large', /fold, /bool): begin
    name_atmos = 'ATMOS_BOTH_LARGE'
    delta1 = 0.1  ;ATMOS_LARGE
    delta2 = 0.22  ;ATMOS_LARGE
    HZ = 1.10            ;ATMOS_LARGE
    shift_coef =   0.9   ;ATMOS_LARGE
    r1 = 1.42   ;ATMOS_LARGE
    r2 = r1 + 0.1    ;ATMOS_LARGE
    rmin = 1.04  ;distance where the rate drops to zero from the shifted center
    rmax = 2.16
    phi_drop = 2.  ;power index of the cos drop with longitude (2 + 2=4 for Z=1)
    end
  stregex(input.spatialdist.size, 'small', /fold, /bool): begin
    name_atmos = 'ATMOS_BOTH_SMALL'
    delta1 = 0.17  ;ATMOS_SMALL
    delta2 = 0.15  ;ATMOS_SMALL
    HZ = 0.95 ;ATMOS_SMALL
    shift_coef =   0.8   ;ATMOS_SMALL
    r1 = 1.02   ;ATMOS_SMALL
    r2 = r1 + 0.04   ;ATMOS_SMALL
    rmin = 1.04  ;distance where the rate drops to zero from the shifted center
    rmax = 2.16
    phi_drop = 7.  ;power index of the cos drop with longitude (2 + 7=9 for Z=1)
    end
  else: stop
endcase

;PLASMA DATA
;************
nel0 = 3778.0  ;upstream plasma densitycm-3
Bio = 1781.e-9 ; magn field at Io
vfl =  57.e3 ; upstream flow velocity m/s
mu0 =  4.* !pi *1.e-7 ;mgn permitivity
```

```
52  Valf =  Bio/sqrt(mu0 * (nel0 *1e6) * 22. * 1.67e-27) ;Alf velocity in m/s
53  Malf =   Vfl/Valf; Mach number
54  ANg_ALF =  atan(Malf) * 180./!pi; angle of alfven tube
55
56  ;; Determine r' = modified radial component
57  rr_pr = dindgen(1001)/1000 * rmax
58  fr_pr1 = exp(-(rr_pr-r1)^2/delta1^2) * (rr_pr GT 1)
59  fr_pr2 = 0.25*exp(-(rr_pr-r2)^2/delta2^2) * (rr_pr GT r1)
60  fr_pr = fr_pr1 + fr_pr2
61  r_pr = RandomDeviates_1d(rr_pr, fr_pr, npack, seed=seed)
62
63  ;; Determine latitudinal (z) and modified azimuthal (phi') components together
64  zz = (dindgen(201)/100-1)*2*Hz
65  pp_pr = dindgen(361)*!dtor
66
67  fz = exp(-(zz/Hz)^6)
68  f_zphi = dblarr(201,361)
69  for i=0,n_elements(zz)-1 do $
70    for j=0,n_elements(pp_pr)-1 do $
71      f_zphi[i,j] = fz[i] * (.5*(cos(!dpi-pp_pr[j])+1))^(2+phi_drop*abs(zz[i]))
72
73  RandomDeviates_2d, f_zphi, zz, pp_pr, npack, z, phi_pr, seed=seed
74
75  x_pr = r_pr * cos(phi_pr)
76  delX = shift_coef * Malf * abs(z)
77
78  *output.x0 = -(x_pr + delX)
79  *output.y0 = r_pr * sin(phi_pr)
80  *output.z0 = z
81
82  end
```