Thermal model for analysis of Mars infrared mapping

Hugh H. Kieffer   File= /xtex/tes/krc/jpap.tex

May 11, 2009

Contents

May 11, 2009

Liens:

Check T_s oscillation when have night frost
tdisk.f:: type 5x except 52 need to be updated
update IDL routines in *helplist.tex*
Complete Sections: 4.8, 6
Check Figs 1 and 2

1 Introduction

This paper describes a numerical model used extensively for computing planetary surface temperatures. The KRC numerical model has evolved over a period of four decades and has been used for a variety of planet, satellite and comet problems, but use has concentrated on Mars. The model uses a one-layer atmosphere but does allow condensation and global pressure variation; the model can output surface kinetic and planetary (nadir view from space) bolometric temperatures, along with a variety of parameters related to subsurface-layer and atmosphere temperatures, seasonal polar cap mass, heat-flow and numerical performance parameters.

The program is designed to compute surface and subsurface temperatures for a global set of latitudes at a full set of seasons, with enough depth to capture the annual thermal wave, and to compute seasonal condensation mass. For historic reasons (it originated in the era of kilo-Hz processors) the code has substantial optimization. It allows sloped surfaces and two zones of different sub-surface materials. There are generalities that allow this code set to be used for any ellipsoid with any spin vector, in any orbit (around any star); with or without an atmosphere (including condensation); this is also the source of some of the complexity.

In response to an oft-asked question, the acronym KRC is simply K for conductivity, R for "rho" (ρ) for density, and C for specific heat; the three terms in thermal inertia (TI).

KRC uses explicit forward finite differences and is coded in FORTRAN; model development began 1968, and was used to support the Viking when computing a single case for 19 latitudes at 40 seasons took an hour on a large university main frame computer[1]. For this reason, the code was highly optimized for speed and uses layer thickness increasing exponentially downward and time steps that increase by factors of two deeper into the subsurface where stability criteria are met. The code is modularized based on time scale and function, and there is extensive use of Commons. The version used for Viking was described briefly [appendix] in [32]. The KRC model was used in many analyzes of the Viking IRTM data, derivatives were used to study sublimating comets [58] and ring and satellite eclipses [4, 19]. The code has undergone step-wise revision, a major change being 2002 replacement of a down-going steady IR flux equivalent to fixed fraction of the noon insolation with the atmosphere described here, in which version it has been the basis for analysis of THEMIS and MER Mini-TES results. The code now allows temperature-dependent thernal conductivity.

A guide to running KRC is in the file **helplist.txt**; see Appendix E. For THEMIS, a "one-point" capability was included that allows input of a set of points defined by season, latitude, hour and a few major physical parameters; KRC will produce the surface kinetic temperature and planetary brightness temperature for these points; see Section 4.10.

1.0.1 Use for recent missions

Although the thermal models for the MGS Thermal Emission Spectrometer (TES) data production were based on the Mellon-Jakosky-Haberle model, which has some heredity from KRC [35, 47], KRC has been used in the analysis of TES data [36, 37]. Extensive comparison of the Mellon model and KRC was done in development of the MGS TES production code.

Determination of thermal inertia using the KRC model has been used in selecting all landing sites on Mars; Pathfinder: [23], MER: [12, 25], Phoenix: [1], MSL: (M. Golombek, personal communication). Post-landing assessment has shown the forecasts of rock abundance to be close. [22, 21]

Standard data reduction of the Odyssey Thermal Emission Imaging System (THEMIS) uses the KRC model,[10, 11, 48, 56]. KRC was used in analysis and surface thermal observations by Mini-TES, [24, 17].

[1] An IBM 360-91, at that time the largest (4 Mbyte memory) and fastest (16 * 1 MIPS) un-classified computer

KRC thermal modeling has been used for study of general nature of the Martian surface [5, 6, 7, 15, 3], Chapter 9 in [9]; and detailed sites: [2, 13, 16, 17, 18, 20, 26, 49].

KRC models are the basis for the surface temperature estimate to be used for the black-body emission correction to Mars Reconnaissance Orbiter (MRO) Compact Reconnaissance Imaging Spectrometer for Mars(CRISM) reflection spectra, [34].

KRC [30] and derivatives [43, 44] have been used in study of seasonal slab ice. The capability to model temperatures at the bottom of conical pits was added to study the potential volatile sublimation in freshly exposed trenches to be dug by the Phoenix mission.

1.0.2 Some other thermal models used for Mars

A finite-difference thermal model used for estimating depth to liquid water stability [14] was made publicly available. A derivative of this model and KRC was used to study ground ice stability [39].

The martian atmosphere has a significant effect on surface temperature, both in the physical temperature of the surface being influenced by the dusty atmosphere's modification of the insolation that reaches the surface, and on the apparent temperature measured remotely by infrared radiometry [27]. Thermal models which treat the atmosphere in detail, such as a dusty radiative/convective column [27] or that include lateral heat transport such in a General Circulation Model (GCM) [28, 8], generally take two to several orders of magnitude longer to compute.

The model used by the U. Colorado group [35] has indirect heritage from KRC and uses a similar subsurface, but has a multi-layer radiative/convective atmosphere; this model was used for TES standard data production.

The goal of the KRC model has been to account for the first-order effects of the atmosphere, while preserving the speed and flexibility to deal with surface effects such as layered materials and sloping surfaces. A complicating factor in treating the atmosphere more fully is that the opacity of Mars's atmosphere can vary considerably in space and time [54], only GCM's with surface dust interaction model this.

A model similar to KRC was used largely for Mars' polar studies [42, 40, 41].

1.0.3 Notation here

Program and routine names are shown as **PROGRM [,N]** , where **N** indicates a major control index. Code variable names are shown `VARIAB`. Input parameters are shown as INPUT. File names are shown as *file*.

For convenience, some physical parameter default values are shown within square brackets at their point of mention and some are listed in Table 1. All units are SI, except the use of days for orbital motion. The sample input file (Appendix A), includes all input parameters.

2 Physical representation

2.1 Planetary Orientation and Orbit

KRC can accept either fixed heliocentric range and sub-solar declination, or Keplerian orbital elements and a fixed planet orientation (direction of the spin axis); in both cases, "seasons" are at uniform increments of time. An associated program set, **PORB**, accesses files containing the elements for all the planets [52] and a few comets; this program set pre-calculates the orbital elements for any epoch, converts them into rotation matrices for the chosen epoch and creates an ASCII parameter set that is then incorporated into the input file for KRC. For TES and the initial THEMIS modeling, the martian elements were evaluated for epoch 1999; Mars' spin-axis orientation was based on pre-Viking data, and differs from the current best estimates [51] by about 0.3°. Within KRC, the orbital position of Mars is computed for each "season", yielding the heliocentric range, the sub-Solar latitude, and the seasonal indicator L_s.

Planetary orientations have been updated to [51] and mean elements have been updated to [50].

For Mars, the maximum error in ecliptic longitude is under 30 arc-sec, corresponding to about 1/60 of a sols' motion, which is negligible compared to the assumption of ignoring Martian longitude.

2.2 Atmosphere

KRC uses a one-layer atmosphere that is gray in the solar and thermal wavelength ranges. Radiative exchange with the Sun, space and the surface determines the atmosphere energy balance and its temperature variation. The columnar mass (and the surface pressure) can vary with season and surface elevation. A uniformly-mixed dust loading is allowed to modify the visual and thermal opacity. Direct and diffuse illumination are computed using a double-precision 2-stream Delta-Eddington model, with single scattering albedo ϖ and Henyey-Greenstein asymmetry parameter G_H. The thermal opacity is a constant factor C_2 times the visual opacity. An option allows an extension of twilight past the geometric terminator.

The current local solar-wavelength atmospheric opacity of dust, τ, can vary with atmospheric pressure: $\tau = \tau_0 \cdot P/P_0$

2.2.1 Delta Eddington 2-stream

A Delta-Eddington model is used for atmosphere scattering and fluxes (**deding2.f**); output parameters are normalized to unit solar irradiance along the incident direction at the top of atmosphere; so they must be multiplied by S_M to get flux.

Scattering parameters used are the aerosol single scattering albedo ϖ and the scattering asymmetry parameter [29]; both are input constants. The computded values include:

Planetary (atm plus surface system) albedo: `BOND`
Direct beam at the bottom, includes both collimated and aureole: $F_{\|}$ = `COLL`
Diffuse irradiances: $I_{i,j}$ =
i: [1, = isotropic [2, = asymmetric
j: ,1]= at top of atmosphere ,2] = at bottom of atm

The net diffuse flux is $F_{\ominus} = \pi[I_{1j} \pm \frac{2}{3} I_{2j}]$ where + is down, $F^{\downarrow}_{\ominus}$; - is up, $F^{\uparrow}_{\ominus}$. [53, eq. 8]

The total down-going solar flux through the day at the bottom of the atmosphere is

$$S'_t = S_M \left(\cos i_0 \; F_{\|} + F^{\downarrow}_{\ominus} \right) \qquad \text{eq : st} \tag{1}$$

where the diffuse component is $F^{\downarrow}_{\ominus} = \pi \left(I_{1,2} + \frac{2}{3} I_{2,2} \right)$

Solar heating of the atmosphere, by conservation of energy, is

$$H_V = S_M \left(\mu_0 - F^{\uparrow}_{\ominus}(0) - (1 - A_s) \left[\mu_0 \; F_{\|} + F^{\downarrow}_{\ominus}(\tau) \right] \right) \qquad \text{eq : aheat} \tag{2}$$

where $\mu_0 \equiv \cos i_0$.

2.2.2 Twilight

Twilight is allowed to account for having a turbid atmosphere. It is implemented as having the diffuse downward illumination depend upon an incidence angle scaled to go to 90° when the Sun is `TWILI` below the geometric horizon.

Because of the twilight extension, there can be a small negative energy balance near twilight. Physically, this is lateral scattering and does not strictly fit a one-layer model. There is no solar heating of the atmosphere during twilight.

2.2.3 Atmospheric IR radiation

The IR opacity is approximated as $\tau_R = P/P_0 \cdot (C_1 + C_2 \tau)$ where C_1 represents the opacity of the "clear" atmosphere, primarily due to the 15μm band, and C_2 is the IR/visual opacity ratio for dust.

The fractional thermal transmission of the atmosphere at zenith is roughly $e^{-\tau_R}$. The fractional absorption is $\beta \equiv 1. - e^{-\tau_R}$.

The fractional transmission of planetary (thermal) radiation in a hemisphere is:

$$e^{-\tau_e} \equiv \int_0^{90} e^{-\tau / \cos\theta} \cos\theta \sin\theta \; d\theta \tag{3}$$

Numerical integration shows that the effective hemispheric opacity is, within about 0.05 in the factor,

$$\tau_e \sim [1.0 < (1.50307 - 0.121687 * \ln \tau_R) < 2.0]\, \tau_R; \tag{4}$$

this is used in the effective absorption $\beta_e \equiv 1. - e^{-\tau_e}$.

The hemispheric downward (and upward) emission from a gray slab atmosphere is: $R_\Downarrow = \sigma T_a^4 \beta_e$. The IR heating of the atmosphere is:

$$H_R = \epsilon \sigma T_s^4 (1. - e^{-\tau_e}) - 2R_\Downarrow \quad = \quad \sigma \beta_e (\epsilon T_s^4 - 2T_a^4) \qquad \text{eq : rheat} \tag{5}$$

To estimate the back-radiation from a clear atmosphere, a synthetic transmission spectrum of the Mars atmosphere with a nominal amount of water vapor (provided by David Crisp) was multiplied by blackbody spectra for a range of temperatures to determine the fraction of radiation blocked, see Figure 3. A coefficent of 0.11 ± 0.004 covers the range 187K to 293K.

2.2.4 Atmospheric temperature

The atmospheric temperature is assumed to follow radiative energy conservation:

$$\frac{\partial T_a}{\partial t} = \frac{H_R + H_V}{c_p M_a} \qquad \text{eq : dTa} \tag{6}$$

where $M_a = P/G$ is the mass of the atmosphere and c_p is its specific heat at constant pressure.

Because the atmospheric temperature variation has significant time lag relative to the surface, one can use the surface temperature from the prior time step (typically 1/384 of a sol) with little error.

If the computed atmospheric temperature at midnight drops below the CO_2 saturation temperature for one scale height above the local surface, it is bounded at this value and the excess energy loss is converted to snow. If there is frost on the ground, this snow mass is added to the frost; otherwise it is ignored in the heat budget, which strictly does not conserve energy.

$$\Delta M = \Delta T c_p M_a / L \qquad \text{eq : dMa} \tag{7}$$

The nadir planetary temperature is given by

$$\sigma T_P^4 = \epsilon \sigma T_S^4 (e^{-\tau_R}) + \sigma T_a^4 (1 - e^{-\tau_R}) \implies T_P = \left[\epsilon(1. - \beta) T_S^4 + \beta T_a^4\right]^{1/4} \qquad \text{eq : Tp} \tag{8}$$

2.3 Geometry and Starting Conditions

2.3.1 Geometry

The diurnal variation of insolation onto the surface at the bottom of the atmosphere is computed for the current season and latitude. The incidence angle from zenith onto the horizontal plane (i_0) or sloped surface is computed by:

$$\cos i = \sin \delta \sin(\theta + s_N) - \cos \delta \cos(\theta + s_N) \cos(\phi + s_E) \tag{9}$$

where

$\delta =$ the solar declination,
$\theta =$ latitude,
$\phi =$ hour angle from midnight,
$s_N =$ north component of surface slope,
$s_E =$ east component of surface slope,

Direct (collimated) insolation is computed for the local surface, which may be sloped in any direction and has incidence angle i_2;

Direct insolation is zero when either i_0 or i_2 is $> 90°$. Diffuse illumination is based on i_0, with the optional extension into twilight (see Section 2.2.2). For a sloped surface, the solid angle of skylight is reduced and light reflected off the regional surface (presumed Lambert and of the same albedo) is added; the Delta-Eddington downward diffuse radiance is multiplied by DIFAC $= 1 - \alpha + \alpha A$, where $\alpha = (1 - \cos i_2)/2$ is the fraction of the upper hemisphere obscured by ground. For the small flat bottom of conical pits, $\alpha = \sin^2\left(\frac{\pi}{2} - s\right)$ where s is the slope of the pit wall from horizontal.

If a sloping surface (or pit) is used, the regional horizontal surface (or pit wall) is assumed to be at the same temperature, which becomes a poor approximation for steep slopes.

The incident flux at the top of atmosphere is: $I = S_M \cos i_0$, where $S_M \equiv \frac{S_o}{U^2}$, S_o is the solar constant and U is heliocentric range of Mars in Astronomical Units..

2.3.2 Starting conditions: Diurnal-average equilibrium

For the first season, the atmosphere temperature is set based on the equilibrium for no net heating of the atmosphere or surface, using the diurnal average of insolation (see Eq. 2 and Eq. 5):

$$\langle H_V \rangle + \langle H_R \rangle = 0 \qquad \text{eq : VR} \tag{10}$$

Surface radiation balance, from Eq. 13 for a flat surface with no net sub-surface heat flow:

$$\epsilon\sigma\langle T_s^4 \rangle = (1. - A)\langle S'_{(t)} \rangle + \epsilon\sigma\beta_e\langle T_a^4 \rangle \qquad \text{eq : sbal} \tag{11}$$

Expandsion of $\langle H_R \rangle$ using Eq. 5 and substitution into Eq. 11, yields ;

$$\langle T_a^4 \rangle = \frac{\langle H_V \rangle/\beta_e + (1 - A)\langle S'_{(t)} \rangle}{\sigma(2 - \epsilon\beta_e)} \qquad \text{eq : Ta4} \tag{12}$$

Substitute into Eq. 11 to get $\langle T_s \rangle$. For computational simplicity, the average top-of-atmosphere insolation is used as an approximation for $\langle S'_{(t)} \rangle$; this slightly over-estimates the temperature of the atmosphere at the start of the first season.

The planetary heating values are based on the average surface temperatures from the prior season; this is similar to allowing some long-term lag in total atmospheric temperature response.

2.4 CO_2 Frost condensation and Sublimation

The local frost condensation temperature TFNOW may be either fixed at an input value TFROST, or derived from the local surface partial pressure at the current season.

The relation between condensation/sublimation temperature and partial pressure is taken to be the Clausius-Clapeyron relation: $\ln P_c = a - b/T$, in **CO2PT** with a=27.9546 [Pascal] and b=3182.48 [1/Kelvin], derived from [31] page 959.

If frost is present, $E = W \cdot \Delta t$ energy is used to modify the amount of frost; $\Delta M = -E/L$, where L is the latent heat of sublimation. The frost albedo may depend upon insolation, and there may be an exponential attenuation of the underlying ground albedo; see §2.4.1. The amount of frost at each latitude is carried (asymptotic prediction) to the next season.

2.4.1 Effective Albedo

A thick frost deposit can have a constant albedo, or be linearly dependent upon the insolation as described by [38, 33]. It should be noted that it is now known that regions of the seasonal caps can have virtually constant low albedo, [33, 57].

As the seasonal frost thins, the effective albedo of the surface continuously approaches that of underlying soil. $A = A_f + (A_s - A_f)e^{-M/M_e}$ where M_e is the frost required, kg m^{-2}, for unity scattering attenuation..

2.4.2 Global and local pressure

The total amount of atmosphere is set by the annual mean surface pressure at the reference elevation, P_0, input as PTOTAL.
The current global pressure P_g =PZREF, can be any of the following:
1) constant at P_0
2) P_0 times the normalized Viking Lander pressure curve VLPRES [55]
3) Based on depletion of atmospheric CO_2 by growth of frost caps; P_0 minus the total frost mass at the end of the prior season. Requires that the number of latitudes NLAT> 8.

The initial partial pressure of CO_2 at zero elevation is $P_{c0} = P_0 \cdot (1.-$non-condensing fraction) =PCO2M. The current CO_2 partial pressure at zero elevation is $P_{cg} = P_{c0} + (P_g - P_0)$ =PCO2G.

Both the current local total pressure and CO_2 partial pressure scale with surface elevation and scale height: $P \propto e^{-z/\mathcal{H}}$. The scale height is: $\mathcal{H} = T_a\mathcal{R}/\mathcal{M}G$; where T_a is the mean atmospheric temperature over the prior day (or season), $\mathcal{R}$ is the universal gas constant, $\mathcal{M}$ is the mean molecular weight of the atmosphere (43.5), and G is the martian gravity.

Local current dust opacity scales with local total pressure: $\tau = \tau_0 \cdot P/P_0$. The atmospheric saturation temperature is evaluated at one scale height above the local surface.

2.5 Boundary conditions

2.5.1 Level Surface

The surface condition for a frost-free level surface is :

$$W = (1.-A)S'_{(t)} + \Omega\epsilon R_{\Downarrow} + \frac{k}{X_2}(T_2 - T) - \Omega\epsilon\sigma T^4 \qquad \text{eq : w} \tag{13}$$

where A is the current surface albedo, $S'_{(t)}$ is the total solar radiation onto the surface as in Eq. 1, $R_{\Downarrow}$ is the down-welling thermal radiation (assumed isotropic), k is the thermal conductivity of the top layer, X_2 is the depth to the center of the first soil layer, and T is the kinetic temperature of the surface. Most constant terms are pre-computed, see Table 1. Ω is the visible fraction of the sky; for level surfaces, $\Omega = 1$. The boundary condition is satisfied when W=0.

When frost is present, the values in Eq. 13 are replaced with ϵ_F, A_F, and T_F, where subscript F indicates the frost values, and no iteration is done; leaving W as a non-zero quantity to change the frost amount. See Section 2.4.

2.5.2 Slopes and Conical Holes

The surface condition for a planar sloped surface or a conical pit with a small flat bottom modifies the interaction with the radiation field. This is simplified by using the crude approximation that the surfaces visible to the point of computation are at the same temperature and have the same brightness where illuminated. Then

$$S'_t = S_M\left[F_{\|}\cos i_2 + \Omega F_{\ominus}^{\downarrow} + \alpha A(G_1 F_{\|} + \Omega F_{\ominus}^{\downarrow})\right] \qquad \text{eq : pit} \tag{14}$$

$F_{\|}$ is the collimated beam in the Delta-Eddington model and $F_{\ominus}^{\downarrow}$ is the down-going diffuse beam. G_1 is the fraction of the visible surrounding surface which is illuminated. Within the brackets in Eq. 14,
the first term is the direct collimated beam, DIRECT
the second is the diffuse skylight directly onto the target surface, DIFFUSE
the third term is light that has scattered once off the surrounding surface, BOUNCE

For a sloped surface, G_1 is taken as unity. As a first approximation, for pits $G_1 = (90 - i)/s \;\; < 1)$ where s is the slope of the pit walls. For a flat-bottomed pit, $i_2 = i$ when the sun is above the pit slope, and $\cos i_2 = 0$ when below.

2.5.3 Layering of materials and sub-surface scaling

Beginning with layer IC, all lower layers can have their conductivity, density and volume specific heat reset to COND2, DENS2, and SPHT2 respectively. If LOCAL is set true, then the physical thickness of these layers scales with the local thermal diffusivity; otherwise, the geometric increase of physical layer thickness continues downward unaltered.

2.5.4 Base of model

Normally, the base of the model is treated as insulating. However, there are also options for it to be held at a fixed temperature, which is useful to model subsurface H_2O ice.

2.6 Relation of thermal inertia to particle size

The relation of TI to particle diameter is based on laboratory measurements,[45]; the specific relation shown in Figure 8 is for P=600 Pascal, density=1600 kg/m^3 and specific heat=625.

A histogram was made of the Thermal inertia determined from TES data, [46], although this used a different thermal model. The TI source data are available at http://lasp.colorado.edu/inertia/2007/ ; these data were weighted by area. Most areas are in TI range of 100:500; values above about 200 are increasingly affected by a rock population or real bedrock.

3 Numerical Methods

The recent addition of temperature-dependent conductivity is treated as a variation to the pre-2008 version.

3.1 Basic Method

The user inputs the thermal inertia I, the bulk density ρ, and the specific heat of the material C_p. Thermal conductivity k is computed from $I^2/(\rho C_p)$. The thermal diffusivity is $\kappa = \frac{k}{\rho C_p}$. While k, ρ, and C_p do not independently influence the surface temperature for a homogeneous material, they set the spatial scale of the subsurface results; $\texttt{SCALE} = \sqrt{kP/\pi\rho C_p}$.

KRC uses layers that increase geometrically in thickness by a factor RLAY. In order to simplify the innermost code loops, KRC places the radiating surface between the first and second model layers.

Symbols used:

B_i = thickness of layer i [m]. Top (virtual) layer = FLAY * SCALE.
C_p specific heat of the material
H = heat flow$=-k\frac{dT}{dz}$
i = layer index, layer 1 is above the physical surface
subscript +(-) is shorthand for $i+(-)1$; i+.5 is the lower boundary of the layer
I = thermal inertia $\equiv \sqrt{k\rho C_p}$
k = thermal conductivity
P = PERSEC = diurnal period in seconds
R = RLAY = ratio of thickness of succeeding layers
t = time
T = temperature
X = X = depth to middle of each layer [m]
ρ = bulk density
κ = Thermal diffusivity $\equiv \frac{k}{\rho C_p}$

Basic differential equation of heat flow is :

$$\frac{\partial T}{\partial t} = \frac{-1}{\rho C_p}\frac{\partial}{\partial z}\left(-k\frac{\partial T}{\partial z}\right) \quad = \quad \frac{k}{\rho C_p}\frac{\partial^2 T}{\partial z^2} \tag{15}$$

Expressed for numerical calculations:

$$\frac{\Delta T_i}{\Delta t} = -\frac{H_{i+1/2} - H_{i-1/2}}{B_i \rho_i C_{p_i}} \tag{16}$$

Use steady-state relations to find heat flow at interface between two layers: $H = -k\nabla T$

$$H_{i+.5} = -\frac{T' - T_i}{B_i/2}k_i \ \text{ or } \ T' - T_i = -\frac{H_{i+.5}B_i}{2k_i} \tag{17}$$

where T' is the temperature at the interface.

$$\text{similarly} \ \ T_{i+1} - T' = -\frac{H_{i+.5}B_{i+1}}{2k_{i+1}}$$

$$\text{Thus} \ \ T_{i+1} - T_i = -\frac{H_{i+.5}}{2}\left(\frac{B_i}{k_i} + \frac{B_{i+1}}{k_{i+1}}\right)$$

$$\text{or} \ \ H_{i+.5} = -\frac{2(T_{i+1} - T_i)}{\frac{B_i}{k_i} + \frac{B_{i+1}}{k_{i+1}}} \qquad \text{Similarly} \qquad H_{-.5} = -2\frac{T - T_-}{\frac{B}{k} + \frac{B_-}{k_-}}$$

For uniform layer thickness in uniform material, the standard form of explicit forward difference is

$$\frac{\Delta T_i}{\Delta t} = \frac{\kappa}{B^2}\left[T_+ - 2T_i + T_-\right]$$

3.2 Finite difference scheme for Exponential layer thickness

The depth parameter is scaled to the diurnal thermal skin depth. For variable layer thickness:

$$\frac{\Delta T_i}{\Delta t_i} = \frac{2}{B_i \rho_i C_{p_i}} \left[\frac{T_+ - T_i}{\frac{B_i}{k_i} + \frac{B_+}{k_+}} - \frac{T_i - T_-}{\frac{B_i}{k_i} + \frac{B_-}{k_-}} \right] \tag{18}$$

Formulate as

$$\Delta T_i = F_{1_i} \left[T_+ + F_{2_i} T_i + F_{3_i} T_- \right] \qquad \text{eq : deltaT} \tag{19}$$

Define intermediate constants for each layer:

$$F_{1_i} = \frac{2\Delta t_i}{B_i \rho_i C_{p_i}} \cdot \frac{1}{\frac{B_i}{k_i} + \frac{B_+}{k_+}} \equiv \frac{2\Delta t_i}{\rho_i C_{p_i} B_i^2} \cdot \frac{k_i}{1 + \frac{B_+}{B_i} \frac{k_i}{k_+}} \qquad \text{eq : F1} \tag{20}$$

and

$$F_{3_i} = \left(\frac{B_i}{k_i} + \frac{B_+}{k_+} \right) \cdot \frac{1}{\frac{B_i}{k_i} + \frac{B_-}{k_-}} \equiv \frac{1 + \frac{B_+}{B_i} \frac{k_i}{k_+}}{1 + \frac{B_-}{B_i} \frac{k_i}{k_-}} \qquad \text{eq : F3} \tag{21}$$

and

$$F_{2_i} = -(1 + F_{3_i}) \qquad \text{eq : F2} \tag{22}$$

Then the inner-most loop is Eq. 19 followed by

$$T_i = T_i + \Delta T_i \tag{23}$$

The input parameter FLAY specifies the thickness of the top "virtual" layer is dimensionless units (in which the diurnal skin depth is 1.0), so that the scaled thickness of the uppermost layer in the soil is FLAY*RLAY, and the physical depth of its center in meters is 0.5*FLAY*RLAY*SCALE. Normally (LP2 set true) a table of layer thickness, depth, (both scaled and in meters), overlying mass, and numerical convergence factor is printed out at the start of a run.

3.2.1 Extension to temperature-dependent conductivity

The conductivity must be evaluated at each layer and time-step. Because conductivities and layer thicknesses appear largely as ratios, calculate these as infrequently as possible. $k_{(T)}$ is implimented as a third-degree polynomial; expecting that the linear and quadratic term will cover the variation in gas conductivity and the cubic term the radiation effect. To minimize roundoff problems, the polynomial uses a scalesd independant variable $T' = (T - T_{off}) * T_{mul} = (T - 220.) * 0.01$. Because KRC allows two materials, this requires 8 additional coefficients.

Compute once per run: $F_{C_{p_i}} = \frac{2\Delta t_i}{\rho_i C_{p_i} B_i^2}$ and $F_{B_i} = B_+/B_i$

Compute once per time step: $F_{k_i} = k_i/k_+$

Then Equations 20 and 22 become

$$F_{1_i} = F_{C_{p_i}} \cdot \frac{k_i}{1 + F_{B_i} F_{k_i}} \qquad \text{eq : F1t} \tag{24}$$

and

$$F_{3_i} = \frac{1 + F_{B_i} F_{k_i}}{1 + 1/\left(F_{k_-} F_{B_-} \right)}. \qquad \text{eq : F3t} \tag{25}$$

Eq. 22 remains the same, but must be evaluated for every layer and time-step.

3.2.2 Solving the upper boundary condition

When there is no surface frost, the net energy into the upper boundary must be zero. From Eq. 13, find

$$\frac{\partial W}{\partial T} = -F_7 - 4F_5T^3 \tag{26}$$

where the constants are $F_7 = k/X_2$ and $F_5 = \Omega\epsilon\sigma$. Note that this assumes that the temperature gradient in top half of layer 2 is linear.

The corresponding estimate of ΔT is applied iteratively until $\mid \Delta T \mid <$ GGT. If $\frac{\Delta T}{T} > 0.8$, the routine will stop with a warning that the solution has gone unstable. If this normalized increment is > 0.1, then δ is reduced by 70% to improve stability.

The surface kinetic temperature for a balanced boundary condition, Eq. 13, is iterated with Newton convergence until δ, the change in T, is $<$ GGT
if $\mid \delta \mid /T > 0.1$, then δ is reduced by 70% to improve stability
If $\mid \delta \mid /T > 0.8$, it is assumed that the model has gone unstable and it is terminated.

If frost is present, the unbalanced energy W is applied to condensation or sublimation.

3.2.3 Stability and Binary time expansion

The classic convergence stability criterion is $\frac{\Delta t}{(\Delta Z)^2}\kappa < \frac{1}{2}$ or $(\Delta z)^2$, equivalent to $B^2 > 2\Delta t\kappa$. A convergence safety factor is defined as $/\sqrt{2\Delta t \cdot \kappa}$. The process was found to be numerically unstable if this factor is less than about 0.8. The routine will stop with an error message if the safety factor is anywhere less than one. As the layer thickness increases with depth, the routine will repeatedly double the time interval if all the following conditions are met:
The safety factor is larger than 2
The layer is at least the 3rd down
The remaining time intervals are divisible by 2
No more than MAXBOT time doublings will be done

To handle potential large jumps in diffusivity that are allowed between two materials, an initial calculation of the convergence factor for the upper layer of the lower material is made without time-doubling. If this is < 1, then the thickness of this and all lower layers is increased to be just stable. If this factor is greater than the input safety factor, then the number of allowed time-doubling in the upper material is set accordingly.

The numerous input parameters that control the time-depth grid and convergence are based upon extensive testing done during the code development.

3.2.4 Starting conditions

For the first season, the model starts at 18 Hours with the surface temperature normally set to the equilibrium surface temperature of a perfect conductor as calculated in Eq. 11. The bottom temperature is also normally set to this value. The input parameter IB allows the option of setting the initial bottom temperature to TDEEP or also the surface temperature to this value; the latter case is useful for studying details of the disappearance of seasonal frost

Once the top and bottom temperatures are set, all intermediate layer temperatures are set by linear interpolation with depth. The initial atmosphere temperature is always set to the equilibrium values using Eq. 12.

Second-degree perturbation is applied at the end of the (third) day; this jumps the mean temperature of all layers and the lower boundary to equal the mean surface temperature.

3.2.5 Jump perturbations

A logical flag LRESET is normally false. It is set True on day NRSET or later of the first season if the lower boundary is adiabatic, but never on the last day of calculation in a season.

On a day when LRESET is true, the summation for average layer temperatures is restarted. At the end of the day, all layer temperatures are offset by $\langle T_s \rangle - \langle T_i \rangle$ so as to yield no net heat flow.

There is an option to instead perturb temperatures based on a linear plus fractional quadratic function of depth between the diurnal average surface and diurnal average bottom temperatures: if DRSET is not zero, then the layer temperature offsets are, using $x = z_i/z_n$ where $_n$ is the bottom layer:

$$\Delta T_i = (\langle T_s \rangle - \langle T_n \rangle)\,(x + \mathrm{DRSET} \cdot x(1-x))$$

3.2.6 Convergence criteria and parameters

At each time step, if there is not frost, the surface boundary equation is iterated until the change in surface temperature is less than GGT.

The test for how many days to run for a season is as follows:

$\Delta_T =$ is the RMS change of layer temperatures, including the virtual layer, from the prior day, stored at the end of each day in DTMJ

The test for making the next day the last is: either the temperature change over the last two days is nearly constant, or the temperature change is small; i.e.:

$$\mid 1 - \frac{\Delta_T}{\Delta_{T,j-1}} \mid \leq \mathsf{DDT} \qquad \text{or} \qquad \Delta_T \leq \mathsf{DTMAX}$$

where $\Delta_{T,j-1}$ is forced to be at least 10^{-6} . Normally, DDT $= 0.002$ and GGT $= 0.1$ and DTMAX $= 0.1$

After computation of the last day, there is a final check that convergence has continued: the temperature change has decreased or it is still small; i.e.:

$$\Delta_T \leq \Delta_{T,j-1} \qquad \text{or} \qquad \Delta_T \leq \mathsf{DTMAX}$$

If these tests fail, and there are days left in the season, then daily calculations are resumed.

3.2.7 Prediction to next season

Calculations run from midnight to midnight. When convergence has been reached, commonly in fewer days than separate seasons, the results at the last 3 midnights, y_1, y_2, y_3, are used to forecast asymptotically the model result at the end of the season, $y = b_0 + b_1 r^x$ where x is the number of sols remaining in the season. Normally, this will use a fit over the last 3 midnights; for convenience reformulated as

$$y = y_3 + c_1((1. - r^x) \qquad \mathrm{eq:pred} \tag{27}$$

where $r = \frac{y_3 - y_2}{y_2 - y_1}$ is the ratio of the last two changes, and $c_1 = \frac{y_3 - y_2}{1/r - 1}$. If the fit is not asymptotic (e.g., if $r \geq 1$), or if the forecast distance (from the last computed midnight) is less than 0.9 sols, the routine will do a linear prediction using the most recent two points. In addition, lower and upper limits can be specified, e.g.., to keep a temperature from falling below a frost point.

3.3 Comparison to other thermal models

3.3.1 Comparison to Ames GCM

A KRC run for the Viking lander 1 latitude and surface pressure was run with solar dust opacity $\tau_0 = 0.3$ to compare with the AMS GCM (data kindly provided by Robert Haberle). Both models were "spun up" for 20 days, the GCM model used a infrared/solar opacity ratio of 1.0 whereas KRC used 0.5.

The GCM surface temperatures are lower than KRC, the most at night. This results largely from having a deep sub-surface model and starting all layers too cold for this season.

The resulting temperatures for $L_s = 100°$ are shown in Fig. 4. The down-going radiation fluxes at the surface for both models are shown in Fig. 5

The atmospheric temperature for the GCM is a layer average weighted by mass. The KRC and GCM atmospheric temperatures have similar phase, with minima near 8H and maxima near 17H, however, the KRC down-going infrared radiance lags the GCM slightly, as expected because the GCM near-surface atmospheric layers dominate the down-going flux and track the surface temperature more closely than the KRC one-layer atmosphere.

The KRC atmosphere down-going infrared radiances are similar to the GCM if the ID/solar opacity ratio is 0.5.

Variations on the KRC model were run to simulate the effect of having a deep model and starting at 180K for a 20 day run, as did the GCM run; these yield surface temperatures about 3 K cooler than a realistic subsurface temperatures.

A specific test case was chosen for comparison of the KRC one-layer atmosphere with the multi-layer radiative, conductive and convective coupled atmosphere of a full Global Circulation Model (GCM), [28]; the Viking-1 landing site, Latitude 22°N, elevation -3.1 km, $L_s = 100°$, $\tau = 0.3$, visible/IR opacity ratio 1.0, surface pressure of 7 millibar, bolometric albedo of 0.25, thermal inertia 270 $\mathrm{J\,m^{-2}s^{-1/2}K^{-1}}$, with a run-up time of 20 sols. The comparison with KRC run with the same conditions is shown in Figures 1 and 2; except that the base KRC model used a ratio of thermal/solar opacity=1.0. The GCM used an initial "run-up" of 20 sols and inhibited lateral atmospheric dynamics. KRC used a 3 year start-up (spaced 1/40'th of Martian year) followed by a 20-sol run-up every sol and ; the latter is more realistic for sub-surface conditions and raises all temperatures about 2 K.

3.3.2 Comparison to Mellon model

For TES standard processing, Mellon models were generated at 8 sol intervals and 5°latitude spacing for 10 thermal inertia's spaced logarithmically; for 3 sets of albedo, dust opacity, and average surface pressure. KRC models were generated on the same grid, except only latitudes 85,60,30 and 0, both N and S and for the middle value of albedo (0.25), dust opacity (0.5), and average surface pressure (600 Pa). The same values were used for all physical parameters identified in the Mellon models. The diurnal surface temperature curves for three thermal inertia and three latitudes are shown in Figure 6. KRC models are a few degrees warmer, the greatest at night and for low thermal inertia. A seasonal comparison of is shown in Figure 7; The models track each other closely except for the lowest inertia at 30S near $L_s = 90°$, when CO_2frost forms at night [?? investigate].

4 Architecture

The main program can run one or more "cases", which are normally independent except for retaining the input parameters that are not explicitly changed. However, "linking" runs that transfer forward the current conditions (layer temperatures, frost budget, atmospheric temperature) is possible; see Section 4.9.

The main program, **KRC** calls **TDAY,1** once with a flag that precomputes everything possible about the subsurface numerical scheme and then for each season calls

TSEAS which determine the distance and declination of the Sun, and then calls

TLATS which loops over latitudes, calculating insolation and atmospheric parameters for one latitude and calling

TDAY,2 which does the layer calculations for each time step and each "day" needed to reach convergence.

In addition to the FORTRAN version, IDL interfaces to KRC exist.

4.1 Main program, KRC

KRC explicitly sets all common areas to zero (not necessary on most modern computers), defines all logical units, sets physical constants, and asks for the input file name and the output log file name. It calls **TCARD,1** to read the input file up to before the first potential change card.

If **TCARD** reports having detected the one-point mode; the 1-point flag is set in common so all routines will know, the initial input file is closed, and the input list of points and the output table file are opened. **TCARD,2** is called to get the first 1-point.

KRC now starts a case and calls **TSEAS**. The remainder of **KRC** is logic and looping control for seasons or 1-points, additional cases and control of optional printing and binary output files

4.2 Input: TCARD

All input other than the initial two file names is handled by one routine, **TCARD**, which reads lines from the input file[s]. The first integer on a card controls the action to be taken, there are 12 possibilities, described in *helplist.tex*, which allow changing all integer, real, logical input parameters, the latitude set, the elevations set, titles and file names, and 1-point sets. It also tests for formal errors on array sizes, loop limits and some physical constants. A sample input file is shown in Appendix A

4.3 Seasons: TSEAS

This routine initializes or increments the season counter and computes the current Modified Julian Date (offset from 2,440,000). If LSC true (rare), it calls **TCARD,2** to read any parameter changes and then **TDAY,1** to set up the subsurface model. The Sun-planet geometry is obtained from **PORB** [Planet ORBit] and the L_s computed; the same geometry is used throughout a "season". Notification of execution time thus far will be sent to the terminal if LNOTIF is True and the season count is a multiple of NMOD.

If SVALB if True, then at each season the soil albedo will be derived by linear interpolation in L_s of the albedo table. A similar seasonal variation for atmospheric opacity is available by setting SVTAU True.

The routine **TLATS** is called to do calculations for all latitudes, after which there are options to print a diurnal surface temperature table and layer minimum/maximum temperature table by calling the routine **TPRINT**, which handles nearly all printout.

For certain kinds of binary output (K4OUT < 0) and if the season counter is JDISK and if the run is not a continuation from a prior file, then **TDISK,5** is called to write the Common area `KRCCOM` to disk. if K4OUT is not zero, for every season starting with JDISK, the results for that season are written to the binary output file.

4.4 Latitude calculations: TLATS

TLATS is called once per season. Using solar geometry information in Common, it calculates insolation-related values that are constant across latitude. It sets the reference-level surface pressure as described in §2.4.2; if based on polar cap mass, then the routine **TINT** is called to do the global integration; there is an option for **TPRINT,8** to print the global properties.

Looping over latitude, the local surface pressure is calculated. The solar radiation absorbed at the surface and in the atmosphere are calculated for every time-of-day step, including consideration of insolation-dependent frost albedo if that was specified. There is an option to print the values for each hour. The equilibrium temperature conditions are computed; and starting conditions are set to the ending conditions for the prior season, if any.

The routine **TDAY,2** is called to do the diurnal calculations for one latitude. Based on the number of sols required to reach convergence, temperatures at midnight for all layers, the surface, the bottom, and the atmosphere, along with the amount of frost, are predicted to the end of the season; see §3.2.7. There are options to print a convergence summary and hourly radiation conditions.

4.5 Diurnal calculations: TDAY

The routine **TDAY** continues the inner loops for depth, time of day, and days to convergence; most of the execution time is in this routine. It is coded to minimize the computation time. There are two major sections; **TDAY,1** sets up the subsurface layer and time grid, checking for stability. It computes and saves values that are independent of surface conditions. **TDAY,2** solves the boundary conditions and the diffusion equation, including atmospheric temperature.

TDAY,2 has an outer loop for days-to-convergence; this resets some summations and for the last day and sets the time steps for print and disk output. A middle loop runs over time-of-day; it interpolates the upper two layers to the surface temperature, implements the lower boundary condition, and sets the number of layers to be used for this time step.

There are two inner loops to solve the diffusion equation, they have a total of 10 indexing (two with fixed offsets) and five floating point operations per layer.

After the inner loops, the middle loop finds the new surface heat flow, solves the boundary conditions, computes the new atmospheric temperature and checks for saturation, and modifies any frost amount. If frost appears new or disappears entirely, the frost flag is set appropriately and surface albedo and emissivity reset. If the current day is the last to be computed this season, and the time is on an hour, then hourly conditions are saved and printed.

In the outer loop, midnight conditions and daily averages are saved. Convergence conditions and iteration counts are checked to see if a jump should be done or if the next day can be the last or if the routine is finished.

4.6 Disk Output: TDISK

TDISK handles all binary input/output. It can write a variety of contents for each season. These have been developed over time to address various research issues. Planetary temperature is that defined in Eq. 8.

Direct access files: a record for each season

-1 KRCCOM plus LATCOM. Only this version supports restarting from a specific season.

0 one record of KRCCOM, latitudes and elevations; then records each season for of hourly surface kinetic and planetary brightness temperatures for every latitude

1:49 KRCCOM and DAYCOM for the last latitude.

One large array for all seasons, latitudes and cases, with KRCCOM loaded into a “virtual” part of the array. The number of cases depends upon number of layers, latitudes and seasons, and is described in *helplist.tex.*

51 Surface and planetary temperatures for every hour, latitude and season. Plus, for every season, the date, L_s, PZREF, dust opacity and total frost.

52 For each latitude, surface, planetary and atmosphere hourly temperature and diurnal layer extremes and NDJ4, DTM4, TTA4, FROST4, AFRO4, HEATMM. Plus, for every season, the date, L_s, PZREF, dust opacity and total frost.

54 Surface temperature at 1 and 13 Hours, diurnal-average upward heat flow, midnight frost amount and bottom temperature.

55 For one latitude, 10 items related to temperatures, frost and heat-flow. Useful for large number of seasons.

56 Designed for seasonal cap studies; hourly surface and planetary temperatures, plus several parameters at midnight for each latitude, plus several global parameters each season.

4.7 Commons

All commons are contained in separate files that are included into routines at compilation. The primary file contains constants that set the sizes of arrays, and the main Common, **KRCCOM** which contains all input parameters, some physical constants, and all the major loop indices.

LATCOM contains results for latitudes

DAYCOM contains layer temperature extremes and the values at midnight, several conditions at the end of each iteration day, radiation and surface temperature values at each time step, and indices of time-doubling layers.

HATCOM contains arrays related to heat flow and irradiance

UNITS contains logical unit assignments, open/closed flags and error message indices.

FILCOM contains all file names

PORBCM contains planetary geometry and rotation matrices

4.8 Print file

Tprint output; Describe each flag

Describe print options

Sample layer table

4.9 Linked Runs

Seasonal details by continuing with 1-day season

KRC has the ability to continue from the vertical temperature profile at the end of a prior case, as long as the physical distribution of the layers is not changed.

By continuing from memory and incrementing the total number of seasons, it is possible to continuously change parameters such as the atmospheric opacity.

4.9.1 Routine when2start

If a specific season is desired, an IDL routine is available to compute the proper initial date.

```
function when2start, ls,j5,del
;_Titl WHEN2START Calc starting date for KRC to reach Ls on specific season step
; ls    in.     Float.  L_sub_S target
; j5    in.     Integer. Step on which to reach target
; del   in.     Float. KRC season step size OR negative  of either:
;               number of Sols. <36 OR intervals per Mars-years >35
```

This can be used multiple times for linked runs. For example, to "spin up" for 3 Mars' years of 40 seasons each, then run daily for 20 sols and end at $L_s = 100°$, call in reverse order; because this routine treats indices as 1-based, if second date-interval set adds N intervals, use N+1 as target index.

print, when2start(100.,21,1.0275)
    Will print a starting date of 11896.82, at which time $L_s = 90.86$
print, when2start(90.86,120,17.1745)
    will print the needed starting date of 9853.04

4.10 One-point version (an alternate input)

To support the THEMIS team, an interface to the KRC system was built that computes the temperature for a single condition. The user generates a file 'one.inp' that contains lines of specific times and conditions.

The input file `Mone.inp` is set to do one latitude for 2 seasons. All the iteration and convergence parameters can be set in this file to achieve the accuracy desired. The input file contains a change-card 10 which points to 'one.inp' as the file of specific points.

4.10.1 Guide to running in one-point mode

A parameter initialization file Mone.inp is provided. It sets the KRC system into a reasonable mode for one-point calculations.

A line near the end of that file points to a file 'one.inp' which can contain any number of one-point conditions. 'one.inp' is intended to be edited to contain the cases desired; however, it must maintain the input format of the sample file.

First line is a title, which can be changed freely. The second line is an alignment guide for the location lines and should not be modified; both these lines must be present.

Each following line must start with an '11 '; this is a code that tells the full-up KRC that this is a one-point line. The next 9 fields are read with a fixed format, and each item should be aligned with the last character of the Column title. All items must be present, each line must extend at least to the m in Azim; comments may extend beyond that, but they will not appear in the output file. Be sure to have a ¡CR¿ at the end of the last input line.

The fields (after the 11) in the one-point input are:
Ls L_S season, in degrees
Lat Aerographic latitude in degrees
Hour Local time, in 1/24'ths of a Martian Day
Elev Surface elevation (relative to a mean surface Geoid), in Km
Alb Bolometric Albedo, dimensionless
Inerti Thermal Inertia, in SI units
Opac Atmospheric dust opacity in the Solar wavelength region
Slop Regional slope, in degrees from horizontal
Azim Azimuth of the down-slope direction, Degrees East of North.

The two additional columns in the output file are:
TkSur Surface kinetic temperature
TbPla Planetary bolometric brightness temperature

—– Comments on the One-point model.

The initialization file of 2002mar08 is set to compute the temperatures at the season requested without seasonal memory. It uses layers that extend to 5 diurnal skin depths. It does not treat the seasonal frost properly, so don't believe the results near the edge of the polar cap. Execution time on a circa 2001 PC may be the order of 0.01 seconds per case.

The underlying model is the full version of KRC. By modifying the initialization file, you can compute almost anything you might want. If you choose to try this, best to read the remainder of this document.

5 Use

Users guide available in *helplist.tex*; see Appendix E. For normal runs, the user will be prompted for the name of the input file and the names of a print file.

5.1 Symbols and variables

In Table 1; computation frequency is indicated as:
C = Input constant
F = Firm-coded constant
O = Once
S = Every "season" (may be as frequent as each sol)
H = Every "Hour" (24 times per sol)
R = Rapid: every time-step (Nominal is 384 times per sol)
SR = every time step for one day each season
subscript $[f]$ means that frost values are used if frost is present.
'MARS' indicates that the values were taken from reference [31] at the listed page.

6 Sample Applications and execution time

Polar cap edge and global pressure

nyear @ 1/40 + last year at 1/sol

Redo the Viking models: 19 latitudes, 25 layers, 120 seasons, 3 cases; execution time on a circa 2007 PC was 3.5 seconds.

THEMIS image: 1 lat, 1 season, 8 azimuths, 4 slopes, 5 elevations

Grid interpolation possibilities

References

[1] R. Arvidson, D. Adams, G. Bonfiglio, P. Christensen, S. Cull, M. Golombek, J. Guinn, E. Guinness, T. Heet, R. Kirk, A. Knudson, M. Malin, M. Mellon, A. McEwen, A. Mushkin, T. Parker, F. Seelos, K. Seelos, P. Smith, D. Spencer, T. Stein, and L. Tamppari. Mars Exploration Program 2007 Phoenix landing site selection and characteristics. *Journal of Geophysical Research (Planets)*, 113(E12):0–+, June 2008.

[2] R. E. Arvidson, R. C. Anderson, P. Bartlett, J. F. Bell, D. Blaney, P. R. Christensen, P. Chu, L. Crumpler, K. Davis, B. L. Ehlmann, R. Fergason, M. P. Golombek, S. Gorevan, J. A. Grant, R. Greeley, E. A. Guinness, A. F. C. Haldemann, K. Herkenhoff, J. Johnson, G. Landis, R. Li, R. Lindemann, H. McSween, D. W. Ming, T. Myrick, L. Richter, F. P. Seelos, S. W. Squyres, R. J. Sullivan, A. Wang, and J. Wilson. Localization and Physical Properties Experiments Conducted by Spirit at Gusev Crater. *Science*, 305:821–824, August 2004.

[3] R. E. Arvidson, F. Poulet, R. V. Morris, J.-P. Bibring, J. F. Bell, S. W. Squyres, P. R. Christensen, G. Bellucci, B. Gondet, B. L. Ehlmann, W. H. Farrand, R. L. Fergason, M. Golombek, J. L. Griffes, J. Grotzinger, E. A. Guinness, K. E. Herkenhoff, J. R. Johnson, G. Klingelhöfer, Y. Langevin, D. Ming, K. Seelos, R. J. Sullivan, J. G. Ward, S. M. Wiseman, and M. Wolff. Nature and origin of the hematite-bearing plains of Terra Meridiani based on analyses of orbital and Mars Exploration rover data sets. *J. Geophys. Res. (Planets)*, 111(E10):12–+, November 2006.

[4] H. H. Aumann and H. H. Kieffer. Determination of particle sizes in Saturn's rings from their eclipse cooling and heating curves. *Astron. Jour.*, 183:305–311, 1973.

[5] J. L. Bandfield. High-resolution subsurface water-ice distributions on Mars. *Nature*, 447:64–67, May 2007.

[6] J. L. Bandfield and C. S. Edwards. Derivation of martian surface slope characteristics from directional thermal infrared radiometry. *Icarus*, 193:139–157, January 2008.

[7] J. L. Bandfield and W. C. Feldman. Martian high latitude permafrost depth and surface cover thermal inertia distributions. *J. Geophys.Res.(Planets)*, 113(E12):8001–+, August 2008.

[8] J. R. Barnes, J. B. Pollack, R. M. Haberle, C. B. Leovy, R. W. Zurek, H. Lee, and J. Schaeffer. Mars atmospheric dynamics as simulated by the NASA AMES General Circulation Model. II - Transient baroclinic eddies. *J. Geophys. Res.*, 98:3125–3148, February 1993.

[9] J.F. Bell. *The Martian Surface: Composition, Mineralogy , and Physical Properties*. Cambridge University Press, 2008.

[10] P. R. Christensen, J. L. Bandfield, J. F. Bell, N. Gorelick, V. E. Hamilton, A. Ivanov, B. M. Jakosky, H. H. Kieffer, M. D. Lane, M. C. Malin, T. McConnochie, A. S. McEwen, H. Y. McSween, G. L. Mehall, J. E. Moersch, K. H. Nealson, J. W. Rice, M. I. Richardson, S. W. Ruff, M. D. Smith, T. N. Titus, and M. B. Wyatt. Morphology and composition of the surface of Mars: Mars Odyssey THEMIS results. *Science*, 300:2056–2061, 2003.

[11] P. R. Christensen, H. Y. McSween, J. L. Bandfield, S. W. Ruff, A. D. Rogers, V. E. Hamilton, N. Gorelick, M. B. Wyatt, B. M. Jakosky, H. H. Kieffer, M. C. Malin, and J. E. Moersch. Evidence for magmatic evolution and diversity on Mars from infrared observations. *Nature*, 436:504–509, July 2005.

[12] P. R. Christensen, S. W. Ruff, R. Fergason, N. Gorelick, B. M. Jakosky, M. D. Lane, A. S. McEwen, H. Y. McSween, G. L. Mehall, K. Milam, J. E. Moersch, S. M. Pelkey, A. D. Rogers, and M. B. Wyatt. Mars Exploration Rover candidate landing sites as viewed by THEMIS. *Icarus*, 176:12–43, July 2005.

[13] P. R. Christensen, S. W. Ruff, R. L. Fergason, A. T. Knudson, S. Anwar, R. E. Arvidson, J. L. Bandfield, D. L. Blaney, C. Budney, W. M. Calvin, T. D. Glotch, M. P. Golombek, N. Gorelick, T. G. Graff, V. E. Hamilton, A. Hayes, J. R. Johnson, H. Y. McSween, G. L. Mehall, L. K. Mehall, J. E. Moersch, R. V. Morris, A. D. Rogers, M. D. Smith, S. W. Squyres, M. J. Wolff, and M. B. Wyatt. Initial Results from the Mini-TES Experiment in Gusev Crater from the Spirit Rover. *Science*, 305:837–842, August 2004.

[14] S. M. Clifford and C. J. Bartels. The Mars Thermal Model (marstherm): a FORTRAN 77 Finite-Difference Program Designed for General Distribution. In *Lunar and Planetary Institute Conference Abstracts*, pages 142–143, March 1986.

[15] K. S. Edgett and P. R. Christensen. The particle size of Martian aeolian dunes. *J. Geophys. Res*, 96:22765–+, December 1991.

[16] W. C. Feldman, M. C. Bourke, R. C. Elphic, S. Maurice, J. Bandfield, T. H. Prettyman, B. Diez, and D. J. Lawrence. Hydrogen content of sand dunes within Olympia Undae. *Icarus*, 196:422–432, August 2008.

[17] R. L. Fergason, P. R. Christensen, J. F. Bell, M. P. Golombek, K. E. Herkenhoff, and H. H. Kieffer. Physical properties of the Mars Exploration Rover landing sites as inferred from Mini-TES-derived thermal inertia. *Journal of Geophysical Research (Planets)*, 111(E10):2–+, February 2006.

[18] R. L. Fergason, P. R. Christensen, and H. H. Kieffer. High resolution thermal inertia derived from THEMIS: thermal model and applications. *J. Geophys.Res (Planets)*, 111:E12004, 2006.

[19] I. Gatley, H. H. Kieffer, E. Miner, and G. Neugebauer. Infrared observations of Phobos from Mariner 9. *Astrophys. Jour.*, 190:497–503, 1974.

[20] T. D. Glotch and P. R. Christensen. Geologic and mineralogic mapping of Aram Chaos: Evidence for a water-rich history. *J. Geophys. Res.(Planets)*, 110(E9):9006–+, September 2005.

[21] M. P. Golombek, R. E. Arvidson, J. F. Bell, P. R. Christensen, J. A. Crisp, L. S. Crumpler, B. L. Ehlmann, R. L. Fergason, J. A. Grant, R. Greeley, A. F. C. Haldemann, D. M. Kass, T. J. Parker, J. T. Schofield, S. W. Squyres, and R. W. Zurek. Assessment of Mars Exploration Rover landing site predictions. *Nature*, 436:44–48, July 2005.

[22] M. P. Golombek, R. A. Cook, T. Economou, W. M. Folkner, A. F. C. Haldemann, P. H. Kallemeyn, J. M. Knudsen, R. M. Manning, H. J. Moore, T. J. Parker, R. Rieder, J. T. Schofield, P. H. Smith, and R. M. Vaughan. Overview of the Mars Pathfinder Mission and Assessment of Landing Site Predictions. *Science*, 278:1743–+, December 1997.

[23] M. P. Golombek, R. A. Cook, H. J. Moore, and T. J. Parker. Selection of the Mars Pathfinder landing site. *J. Geophys. Res.(Planets)*, 102:3967–3988, February 1997.

[24] M. P. Golombek, L. S. Crumpler, J. A. Grant, R. Greeley, N. A. Cabrol, T. J. Parker, J. W. Rice, J. G. Ward, R. E. Arvidson, J. E. Moersch, R. L. Fergason, P. R. Christensen, A. Castaño, R. Castaño, A. F. C. Haldemann, R. Li, J. F. Bell, and S. W. Squyres. Geology of the Gusev cratered plains from the Spirit rover transverse. *J. Geophys. Res. (Planets)*, 111(E10):2–+, January 2006.

[25] M. P. Golombek, J. A. Grant, T. J. Parker, D. M. Kass, J. A. Crisp, S. W. Squyres, A. F. C. Haldemann, M. Adler, W. J. Lee, N. T. Bridges, R. E. Arvidson, M. H. Carr, R. L. Kirk, P. C. Knocke, R. B. Roncoli, C. M. Weitz, J. T. Schofield, R. W. Zurek, P. R. Christensen, R. L. Fergason, F. S. Anderson, and J. W. Rice. Selection of the Mars Exploration Rover landing sites. *J. Geophys. Res.(Planets)*, 108:8072–+, December 2003.

[26] R. Greeley, R. E. Arvidson, P. W. Barlett, D. Blaney, N. A. Cabrol, P. R. Christensen, R. L. Fergason, M. P. Golombek, G. A. Landis, M. T. Lemmon, S. M. McLennan, J. N. Maki, T. Michaels, J. E. Moersch, L. D. V. Neakrase, S. C. R. Rafkin, L. Richter, S. W. Squyres, P. A. de Souza, R. J. Sullivan, S. D. Thompson, and P. L. Whelley. Gusev crater: Wind-related features and processes observed by the Mars Exploration Rover Spirit. *J. Geophys. Res.(Planets)*, 111(E10):2–+, January 2006.

[27] R. M. Haberle and B. M. Jakosky. Atmospheric effects on the remote determination of thermal inertia on Mars. *Icarus*, 90:187–204, 1991.

[28] R. M. Haberle, J. B. Pollack, J. R. Barnes, R. W. Zurek, C. B. Leovy, J. R. Murphy, H. Lee, and J. Schaeffer. Mars atmospheric dynamics as simulated by the NASA AMES General Circulation Model. I - The zonal-mean circulation. *J. Geophys. Res.*, 98:3093–3123, February 1993.

[29] J. H. Joseph, W. J. Wiscombe, and J. A. Weinman. The delta-Eddington approximation for radiative flux transfer. *Journal of Atmospheric Sciences*, 33:2452–2459, 1976.

[30] H. H. Kieffer. Cold jets in the Martian polar caps. *Jour. Geophys. Res., Planets*, 112(E11):8005–+, August 2007.

[31] H. H. Kieffer, B. M. Jakosky, C. W. Snyder, and Eds. M. S. Matthews. *Mars.* University of Arizona Press, Tucson, 1992. 1498 pp.

[32] H. H. Kieffer, T. Z. Martin, A. R. Peterfreund, B. M. Jakosky, E. D. Miner, and F. D. Palluconi. Thermal and albedo mapping of Mars during the Viking primary mission. *J. Geophys. Res.*, 82:4249–4291, 1977.

[33] H. H. Kieffer, T. N. Titus, K. F. Mullins, and P. Christensen. Mars south polar spring and summer behavior observed by TES: Seasonal cap evolution controlled by frost grain size. *J. Geophs. Res.*, 105(E4):9653–9699, 2000.

[34] T. Z. Martin. Thermal correction of MRO CRISM data using photoclinometry and slope-dependent thermal models for the Martian surface. In *Bulletin of the American Astronomical Society*, volume 36 of *Bulletin of the American Astronomical Society*, pages 1160–+, November 2004.

[35] M. T. Mellon, B. M. Jakosky, H. H. Kieffer, and P. R. Christensen. High-resolution thermal inertia mapping from the Mars Global Surveyor Thermal Emission Spectrometer. *Icarus*, 148:437–455, December 2000.

[36] S. A. Nowicki and P. R. Christensen. Rock abundance on Mars from the Thermal Emission Spectrometer. *Journal of Geophysical Research (Planets)*, 112(E11):5007–+, May 2007.

[37] M.M. Osterloo, V.E. Hamilton, J.L. Bandfield, T.D. Glotch, A.M. Baldridge, P.R. Christensen, L.L. Tornabene, and F.S. Anderson. Chloride-bearing materials in the Southern highlands of Mars. *Science*, 319:1651, 2008.

[38] D. A. Paige. The annual heat balance of the Martian polar caps from Viking observations. *Ph.D. thesis*, California Inst. of Technology., 1985.

[39] D. A. Paige. The thermal stability of near-surface ground ice on Mars. *Nature*, 356:43–45, 1992.

[40] D. A. Paige, J. E. Bachman, and K. D. Keegan. Thermal and albedo mapping of the polar regions of Mars using Viking thermal mapper observations 1. north polar region. *J. Geophys. Res.*, 99:25,959–25,991, 1994.

[41] D. A. Paige and K. D. Keegan. Thermal and albedo mapping of the polar regions of Mars using Viking thermal mapper observations 2. south polar region. *J. Geophys. Res.*, 99:25,993–26,013, 1994.

[42] D. A. Paige and S. E. Wood. Modeling the Martian seasonal CO_2 cycle. *Icarus*, 99:15–27, 1992.

[43] S. Piqueux and P.R. Christensen. North and south subice gas flow and venting of the seasonal caps of Mars: A major geomorphological agent. *J. Geophys. Res.*, 113:E06005, 2008.

[44] S. Piqueux, C.S. Edwards, and P.R. Christensen. Distribution of the ices exposed near the south pole of Mars using Thermal Emission Imaging System (THEMIS) temperature measurements. *J. Geophys. Res.*, 113:E08014, 2008.

[45] M. A. Presley and P. R. Christensen. Thermal conductivity measurements of particulate materials 2. results. *J. Geophys. Res.*, 102(E3):6551–6566, March 1997.

[46] N. E. Putzig and M. T. Mellon. Apparent thermal inertia and the surface heterogeneity of Mars. *Icarus*, 191:68–94, 2007.

[47] N. E. Putzig, M. T. Mellon, K. A. Kretke, and R. E. Arvidson. Global thermal inertia and surface properties of Mars from the MGS mapping mission. *Icarus*, 173:325–341, February 2005.

[48] A. D. Rogers, P. R. Christensen, and J. L. Bandfield. Compositional heterogeneity of the ancient Martian crust: Analysis of Ares Vallis bedrock with THEMIS and TES data. *Journal of Geophysical Research (Planets)*, 110(E9):5010–+, May 2005.

[49] S. W. Ruff, P. R. Christensen, R. N. Clark, H. H. Kieffer, M. C. Malin, J. L. Bandfield, B. M. Jakosky, M. D. Lane, M. T. Mellon, and M. A. Presley. Mars' "White Rock" feature lacks evidence of an aqueous origin: Results from Mars Global Surveyor. *Jour. Geophys. Res.*, 106(15):23921–23928, October 2001.

[50] K. P. Seidelmann, editor. *Explanatory Supplement to the Astronomical Almanac.* University Science Books, Mill Valley, California, 2005.

[51] P. K. Seidelmann, B. A. Archinal, M. F. A'Hearn, D. P. Cruikshank, J. L. Hilton, H. U. Keller, J. Oberst, J. L. Simon, P. Stooke, D. J. Tholen, and P. C. Thomas. Report of the IAU/IAG Working Group on Cartographic Coordinates and Rotational Elements: 2003. *Celestial Mechanics and Dynamical Astronomy*, 91:203–215, March 2005.

[52] P. K. Seidelmann, L. E. Doggett, and M. R. Deluccia. Mean elements of the principal planets. *Astron. Jour.*, 79:57–+, January 1974.

[53] E. P. Shettle and J. A. Weinman. The Transfer of Solar Irradiance Through Inhomogeneous Turbid Atmospheres Evaluated by Eddington's Approximation. *Journal of Atmospheric Sciences*, 27:1048–1055, October 1970.

[54] M. D. Smith. Interannual variability in TES atmospheric observations of Mars during 1999-2003. *Icarus*, 167:148–165, January 2004.

[55] J. E. Tillman, N. C. Johnson, P. Gettorp, and D. B. Percival. The Martian annual atmospheric pressure cycle: years without great dust storms. *J. Geophys. Res.*, 98:10963–10971, 1993.

[56] T. N. Titus, H. H. Kieffer, and P. N. Christensen. Exposed water ice discovered near the south pole of Mars. *Science*, 299:1048–1051, 2003.

[57] T. N. Titus, H. H. Kieffer, and K. F. Mullins. Slab ice and snow flurries in the Martian polar night. *J. Geophys. Res.*, 106(E10):23,181–23,196, 2001.

[58] P. R. Weissman and H. H. Kieffer. Thermal modeling of cometary nuclei. *Icarus*, 47:302–311, 1981.

A Sample input file for Mars

Below is a typical input file for Mars. All parameter values should be right-aligned with the parameter name above it. The line beginning "08 Sep 29" and the following block of floating-point numbers specifies the planetary spin axis, the orbit, and contains the associated rotation matrices. All lines below that are "change cards" allowing modification of most parameters; each specified by type (the first number; 1=real 2=integer 3=logical, values greater than 3 have special meaning, explained in the Helplist), location within type (the 2'nd number), new value (the 3'rd number) and a comment which will be printed (4'th item). By FORTRAN convention, everything after a "/" is not read, and thus allows notation in this file. A line beginning with a 0 terminates a set of change cards and starts a new KRC "case". A 2'nd consecutive 0 will terminate the program.

It is possible (by setting LSC True) to read change cards at each season; this requires care to not change any dimensions.

```
 0 0 / KOLD: season to start with;  KEEP: continue saving data in same disk file
Default values for all parameters. 19 latitudes with mean Mars elevation
    ALBEDO     EMISS   INERTIA     COND2     DENS2    PERIOD SPEC_HEAT   DENSITY
       .25      1.00     200.0       3.4     928.0    1.0275      630.     1600.
      CABR       AMW   -ABRPHA    PTOTAL     FANON      TATM     TDEEP   SpHeat2
      0.11      43.5     -0.00     510.0      .055      200.     180.0     1300.
      TAUD     DUSTA    TAURAT     TWILI      ACR2     -ARC3     SLOPE    SLOAZI
       0.3       .90       0.5       0.0       0.5     -0.00       0.0       90.
    TFROST    CFROST    AFROST     FEMIS       AF1       AF2    FROEXT      FD32
     146.0   589944.       .65      0.95      0.54    0.0009       50.       0.0
      RLAY      FLAY     CONVF     DEPTH     DRSET       DDT       GGT     DTMAX
    1.2000     .1800    2.0000       0.0       0.0     .0020       0.1       0.1
      DJUL    DELJUL  SOLARDEC       DAU      HLON    SOLCON      GRAV    Atm_Cp
   10322.34  17.1745      00.0     1.465        .0     1368.     3.727     735.9
        N1        N2        N3        N4        N5       N24        IB        IC
        20       384        16        19       120        24         0         0
     NRSET      NMHA      NRUN     JDISK     IDOWN       I14       I15     KPREF
         3        24         1        81        -7        45        65         1
     K4OUT     JBARE      NMOD    IDISK2                                     end
        52         0         5         0                                       0
    LP1    LP2    LP3    LP4    LP5    LP6 LPGLOB   LVFA   LVFT  debug
      F      T      F      F      F      F      F      F      F      F
  LPORB   LKEY    LSC LNOTIF  LOCAL   LD16 LPTAVE Prt.78 Prt.79   LONE
```

```
      T      F      F      T      T      F      F      F      F      F
LATITUDES: in 10F7.2  _____7 _____7 _____7 _____7 _____7 _____7 _____7
 -87.50 -80.00 -70.00 -60.00 -50.00 -40.00 -30.00 -20.00 -10.00   0.00
  10.00  20.00  30.00  40.00  50.00  60.00  70.00  80.00  87.50  -0.00
Elevations: in 10F7.2 _____7 _____7 _____7 _____7 _____7 _____7 _____7
   3.51   2.01   1.39   1.22   0.38   0.48   1.17   1.67   1.26   0.17
  -0.94  -1.28  -1.99  -2.51  -3.52  -4.08  -4.51  -4.38  -2.57  -0.00
  08 Sep 29 10:41:33 =RUNTIME.  IPLAN AND TC=   4.0 0.55000
   4.000000       0.5500000     0.8650615      0.3229325E-01   5.000821
  0.9340634E-01   1.523671       12882.95       686.9650      0.9229904
   5.544495       24.62280      0.000000      0.4093198       0.000000
   0.000000       0.000000      0.000000       6.159676      0.4662921
  0.4172604E-01   0.6197483       4.381073       0.000000       1.228627
  0.6619807        0.000000       1.391099      0.1075499     -0.3195100E-01
  0.2263214      -1.246176     -0.5861457     -0.8611114E-01  0.8908045
  0.4461527     -0.9063585      0.1158813     -0.4063075     -0.4136413
 -0.4393618      0.7974096      0.9138050     -0.4049719     -0.3095386E-01
  0.4054090      0.9140893      0.9184200E-02  0.2457525E-01 -0.2094154E-01
  0.9994786     -0.3252879     -0.8556869     -0.4024770      0.9456150
 -0.2943530     -0.1384504      0.7823110E-07  0.4256245     -0.9048999
8 0 0 ’/work1/krc/mars/master.t52’ / Disk file name
1 12 540.  ’PTOTAL set to yield 7  mb at VL1 @ Ls=100’
1 35 4. ’CONVF’ / push time doubling start deeper
0/
0/
0/
```

Below is an example of an elaborate set of change cards that looks in detail at the temperatures through the first 40 sols of ice freshly exposed at the bottom of a conical pit. It uses 3 latitudes and does 5 cases; the first is ice freshly exposed to a full hemisphere of sky, followed by pits with slopes of 45 and 25 degrees, then these two pits with a different initial ice temperature

```
7 7 7 ’Pit dug to ice by Phoenix’ / New title
8 0 0 ’../output/phx4.t52’ / Disk file name/
1  1 .20 ’Albedo’
1  3 2025.3 ’Inertia for ice’ /
1  7 1300.  ’Spec heat’ / for ice
1  8 928. ’Density’ / for ice
1 15 185. ’TDEEP’ /
1 17 0.2 ’TAUD’
1 39 .001 ’GGT: set to avoid ending early’ / set for daily output
1 41 11920.2 ’DJUL’ / starting date
1 42 1.0275 ’DELJUL  1 sol’ / set for daily output
2  1 19 ’Num Layers’ /
2  3 1  ’N3: set to run each day’ / set for daily output
2  4 3  ’N4’ / number of latitudes
2  5 40 ’N5’ /  total number of seasons = sols
2  7 2  ’IB start all =TDEEP’ /
2 12 1  ’JDISK start immediately’ /
2 17 52 ’K4OUT: 6 items’/  2 17 51 ’K4OUT: 30 layers’/
4 77 77 ’New Latitudes’ / Must be N4 of them in  10F7.2
  65.00  70.00  72.00 -10.00   0.00  10.00  25.00  45.00  70.00  22.00
5 77 77 ’New Elevations’ / Must be N4 of them in  10F7.2
   -3.5   -3.5   -3.5   00.0   00.0   00.0   00.0   00.0   00.0   -3.1
0/
1 24 -400. ’Azimuth. Set flag to indicate  a pit’ /
1 23 45. ’Slope’ / slope of pit wall
```

```
0/
1 23 65. 'Slope' /
0/
1 23 45. 'Slope' /
1 15 220. ' TDEEP' /
0/
1 23 65. 'Slope' /
0/
0/
0/
```

B Tables

Table 1: Symbols and variables

Sym	Name in	Input File label	Value+	Description and basis
-bol	Code	or Equation	frequency	
A	`AS`		S,R$_f$	Current bolometric albedo.
c_p	`ATMCP`	Atm_Cp	860. C	Atm. specific heat at constant pressure. J K^{-1} kg^{-1}, MARS p.855
C_1	`CABR`	CABR	0.11 C	Clear atmosphere IR absorption.
C_2	`TAURAT`		0.5 C	IR/vis relative opacity. Viking VIS & IRTM opacities. MARS p.1022,5
F_3	`FAC3`	$(1-A_{[f]})$	S,R$_f$	Surface solar absorbtance.
F_4	`FAC4`	$1+1/$RLAY	O	Layer factor.
F_5	`FAC5`	$\Omega\epsilon\sigma$	O	Surface thermal emission factor.
$4F_5$	`FAC45`	$4\Omega\epsilon\sigma$	O	Surface thermal emission factor
F_6	`FAC6`	$\Omega\epsilon_{[f]}$	O	Surface emission factor.
F_7	`FAC7`	$\frac{k}{X_2}$	O	Layer scaling.
F_8	`FAC8`	$e^{-\tau_R}\epsilon_{[f]}$	O	Fraction of surface blackbody reaching top-of-atmosphere.
F_9	`FAC9`	$\sigma(1-e^{-\tau_e})$	O	
G	`GRAV`	GRAV	3.727 C	Martian gravity. m s^{-1}
G_H	`G0`	ARC2	0.5 C	Henyey-Greenstein asymmetry. MARS p.1030
$\mathcal{H}$	`SCALEH`		S	Scale height in km. Based on TATM*
H_R	`ADGR`		SR	Solar heating of atm. Wm^{-2}
H_V	`ADGR`		SR	Solar heating of atm. Wm^{-2}
i			SR	Incidence angle from zenith onto a horizontal surface.
i_2			SR	Incidence angle onto local slope; from SLOPE and SLOAZI
k	`COND`	COND	C	Thermal conductivity of the soil. Wm^{-1}K^{-1}
M			R$_f$	Columnar mass of CO$_2$frost kg m^{-2}
$\mathcal{M}$	`AMW`		43.5 C	Atomic weight of general atmosphere. (g/mole).
P_0	`PTOTAL`	PTOTAL	689.7 C	Global annual mean surface pressure. Pa
P_g	`PZREF`		S	Current pressure at reference level. Pa
P	`PRES`		S	Current local surface pressure. Pa
$R_{\Downarrow t}$	`ATMRAD`	$F_9T_a^4$	R	Hemispheric emission from a gray slab atmosphere. Wm $^{-2}$
S_o	`SOLCON`	SOLCON	1368. C	Solar constant. Wm^{-2}
S_M	`SOL`	S_o/U^2	S	Solar flux at Mars. Wm^{-2}
$S'_{(t)}$	`ASOL`		SR	Total insolation onto [sloped] surface. Wm^{-2}
T	`TSUR`		R	Surface kinetic temperature. Kelvin
T_a	`TATM`	TATM	200. C*	Temperature of the atmosphere. Kelvin
T_a	`TATMJ`		R	Temperature of the atmosphere. Kelvin
T_P	`TPFH`		R	Nadir planetary temperature. Kelvin
t			-	Time from midnight. "Hour"
U	`DAU`		S	Heliocentric range. Astronomical Units
W	`POWER`		R	Energy into the surface boundary. Wm^{-2} s^{-2}

Computation frequency is indicated as:

C = Input constant
F = Firm-coded constant
O = Once
S = Every "season" (may be as frequent as each sol)
H = Every "Hour" (24 times per sol)
R = Rapid: every time-step (Nominal is 384 times per sol)
SR = every time step for one day each season

subscript $[f]$ means that frost values are used if frost is present.

'MARS' indicates that the values were taken from reference Mars92=[31] at the listed page.

Table 2: Symbols and variables: Continued

Sym -bol	Name in Code	Input File label or Equation	Value+ frequency	Description and basis
α	`1-SKYFAC`	$(1-\alpha)$	S	Fraction of upper hemisphere occupied by ground = slope/180°
β	`BETA`	$1-e^{-\tau_R}$	S	Vertical thermal absorption of atmosphere
β_e	`BETH`	$1-e^{-\tau_e}$	S	Hemispheric thermal absorption of atmosphere
γ	`TWILFAC`		S	Twilight extension factor = 90/(90+twilight)
δ	`[R]SDEC`		S	Solar declination.
$\epsilon_{[f]}$	`EMIS`		S,R_f	Surface emissivity. FEMIS for frost
θ	`DLAT`		S	Latitude. θ_2 = latitude + slope north
μ_0	`COSI`		R	Cosine of the incidence angle
ϖ	`OMEGA`	DUSTA	0.9 C	Dust grain single scattering albedo. MARS p.1030
Ω	`SKYFAC`	$\equiv 1-\alpha$	SR	Fraction of the sky (upper hemisphere) that is visible to the surface
σ	`SIGSB`	5.67051e-8	F	Stephan-Boltzman constant. W m^{-2} K^{-4}
τ_0	`TAUD`	TAUD	0.2 C	Nominal solar-range dust opacity
τ	`OPACITY`		S	Current local dust opacity
τ_e	`TAUEFF`		S	Effective thermal opacity of the atmosphere
τ_R	`TAUIR`		S	Thermal opacity, zenith
ϕ	`ANGLE`		R	Hour angle from midnight, ϕ_2 = hour angle + slope east
$\langle\ \rangle$				diurnally-averaged value
	`TWILI`	TWILI	1.0 C	Central angle extension of twilight, degrees
	`DTAFAC`	$\Delta t/(c_p \frac{P}{g})$	O	Atmosphere heating factor. $s^2 m^2 K$ W^{-1}
	`FEMIT`	$\Omega \epsilon_f \sigma T_f^4$	O	Frost thermal emission.

Table 3: Sample layer table

```
RUN-CASE   1- 1    05 Nov 19 16:45:41   PAGE=  3
 Conductiv.= 3.400E+00  Dens*Cp= 1.206E+06  Diffu.= 2.818E-06  Scale= 2.822E-01
         ___THICKNESS____   _______CENTER_DEPTH_____ CONVERGENCE
 LAYER     scale    meter     scale    meter   kg/m^2   factor
    1    0.1800   0.0508   -0.0900  -0.0254     0.000   0.000
    2    0.2160   0.0610    0.1080   0.0305    56.568   2.851
    3    0.2592   0.0731    0.3456   0.0975   124.450   2.053
    4    0.3110   0.0878    0.6307   0.1780   205.908   2.956
    5    0.3732   0.1053    0.9729   0.2746   303.658   2.129
    6    0.4479   0.1264    1.3834   0.3904   420.958   3.065
    7    0.5375   0.1517    1.8761   0.5295   561.718   2.207
    8    0.6450   0.1820    2.4673   0.6963   730.630   3.178
    9    0.7740   0.2184    3.1768   0.8965   933.324   2.288
   10    0.9288   0.2621    4.0282   1.1368  1176.557   3.295
   11    1.1145   0.3145    5.0498   1.4251  1468.437   2.372
   12    1.3374   0.3774    6.2758   1.7711  1818.693   3.416
   13    1.6049   0.4529    7.7469   2.1863  2239.000   2.460
   14    1.9259   0.5435    9.5123   2.6845  2743.368   3.542
   15    2.3111   0.6522   11.6308   3.2823  3348.610   2.550
   16    2.7733   0.7826   14.1730   3.9997  4074.901   3.672
   17    3.3279   0.9392   17.2236   4.8606  4946.449   5.288
   18    3.9935   1.1270   20.8843   5.8937  5992.308   7.615
   19    4.7922   1.3524   25.2771   7.1334  7247.337  10.965
Bottom layers for time doubling:      2    4    6    8   10   12   14   19
```

Table 4: Fortran Code set

Name	description
	Primary routines
KRC	Planet surface thermal model; top routine, MGS-TES version
TSEAS	Advance one ”season” along planets orbit
TLATS	Latitude computations
TDAY	Day and layer computations
	Input / output routines
TCARD	Read input file and changes
TDISK	Save/read results at the end of a season; Version with BINF5
TPRINT	Printed output routine
	Specific task routines
ALBVAR	Compute frost albedo as linear function of insolation
ALSUBS	Convert between L_s and days into a Martian year
AVEDAY	Average daily exposure of surface to sunlight.
CO2PT	CO_2 pressure/temperature relation
DEDING2	Delta-Eddington 2-stream solution for single homogeneous layer
EPRED	Exponential Prediction of numerical iteration
TINT	Spherical integrals over globe
VLPRES	Viking lander pressure curves
	Orbit geometry routines
PORB	Computes planetary angles and location for specific time.
PORB0	Planetary orbit. Read pre-computed matrices and do rotation; minimal for KRC
ECCANOM	Iterative solution of Keplers equations for eccentric orbit
ORBIT	Compute radius and coordinates for elliptical orbit
	Utility routines listed in Makefile
Fortran	catime.f datime.f idarch.f sigma.f vaddsp.f xtreme.f binf5.f white1.f
C	b2b.c r2r.c u_move1.c u_move4.c u_swapn.c primio.c pio_bind_c.c
C	binf5_bind.c b_alloc.c b_c2fstr.c b_f2cstr.c b_free.c
	Other routines
IDLKRC	Interface to IDL. Planet surface thermal model MGS-TES version

C Figures

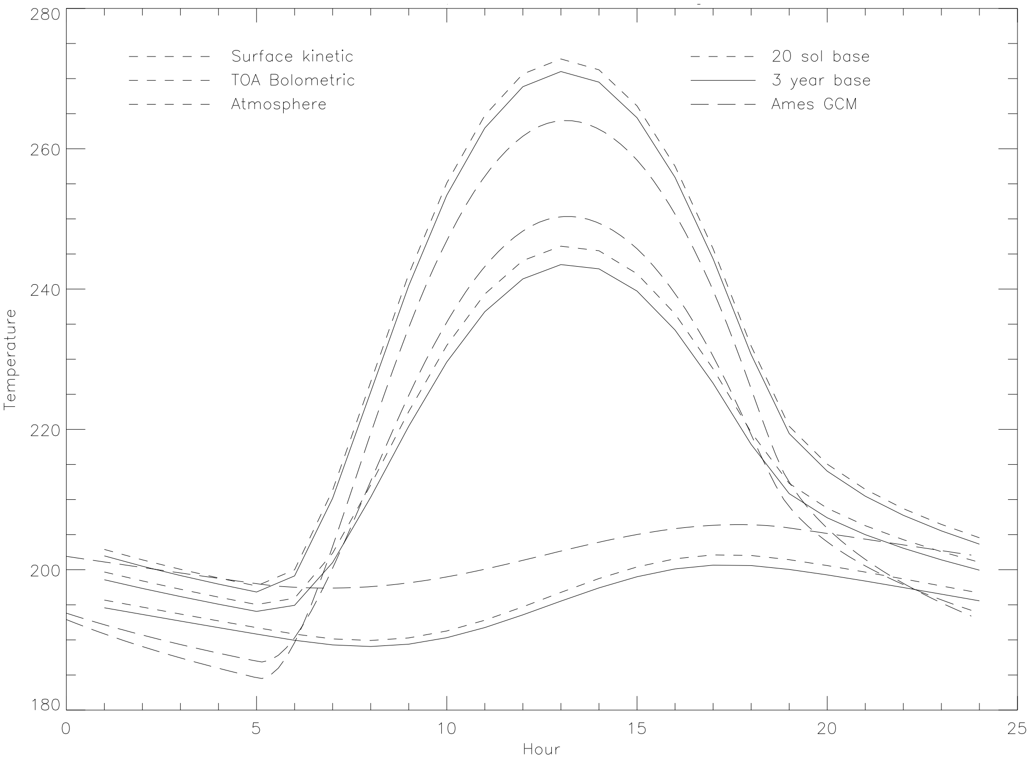

Figure 1: Comparison with GCM results. Calculations for the Viking Lander 1 site at $L_s = 100°$. The input parameter are the same for all models, see text. The upper three lines are surface kinetic temperature through 24 Hours; dashed line is KRC results with a 20-sol run-up, solid line is KRC with a 3 year run-up, and the long-dash line the GCM results with a 20-sol run-up. The lower three curves are the mass-weighted temperature of the atmosphere, and the central three curves are the top-of-atmosphere bolometric temperature.

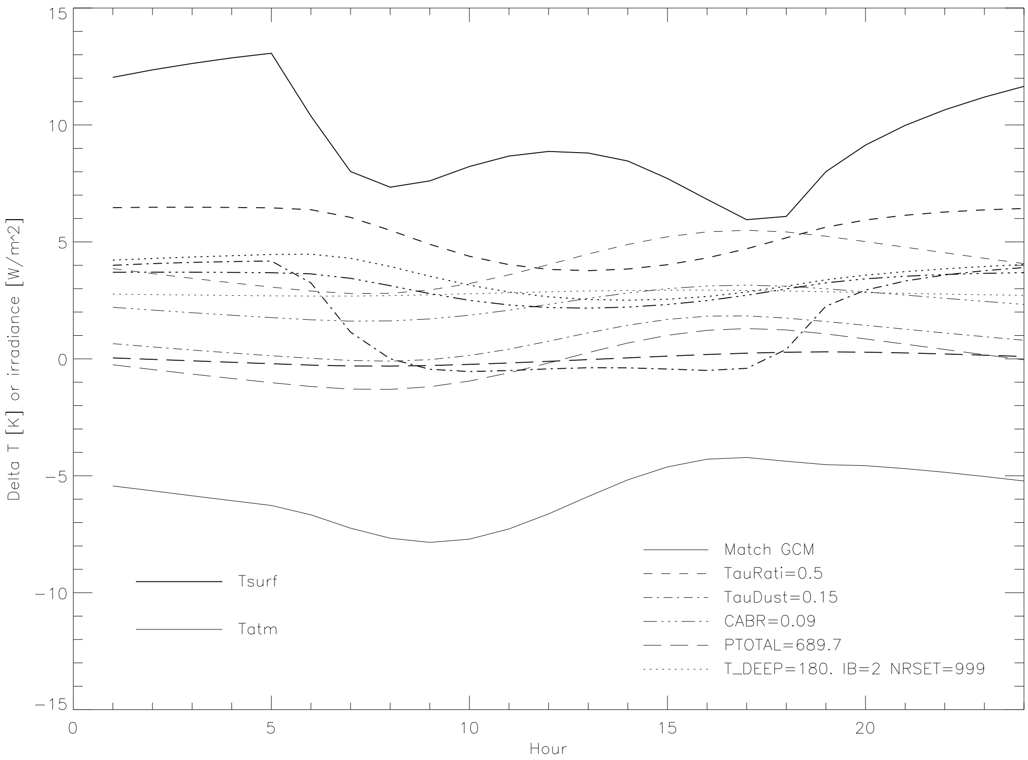

Figure 2: Difference of KRC base model from the GCM model and effect of modifying KRC atmosphere parameters. The solid lines KRC base - GCM temperatures; thick for surface temperature and thin for atmosphere temperature.

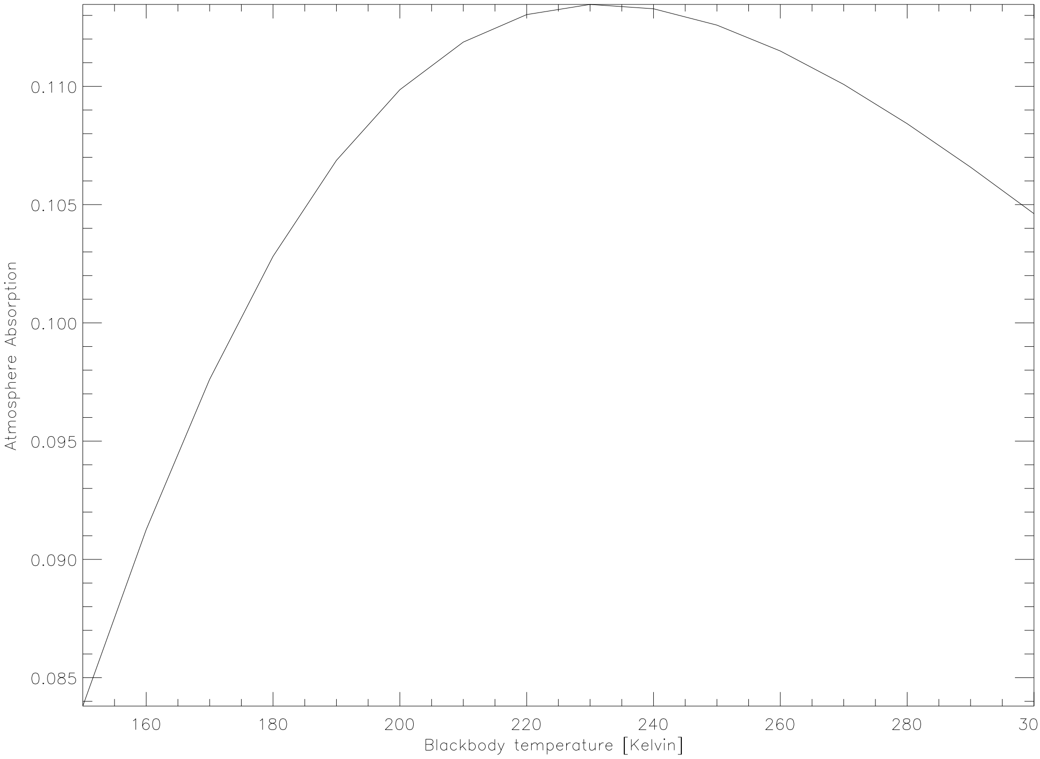

Figure 3: Blocking of thermal radiation by a dust-free Mars atmosphere. Ordinate is the fraction of blackbody radiation absorbed by a nominal atmosphere of 7000[?] Pa CO$_2$with a nominal amount of water vapor. Abscissa is the blackbody temperature.

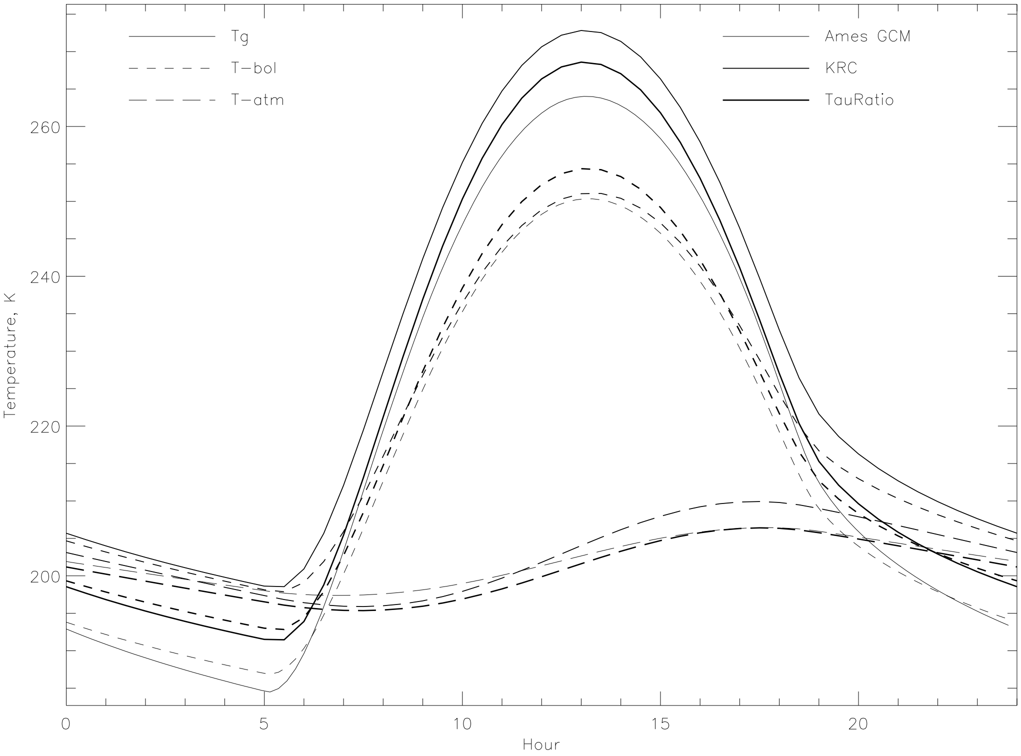

Figure 4: Surface kinetic, mass-weighted atmosphere and planetary model temperatures for the Viking 1 landing site at $L_s = 100°$. Light lines are data from the AMES GCM; heavy lines from KRC. Thermal inertia of 270, surface albedo 0.25.

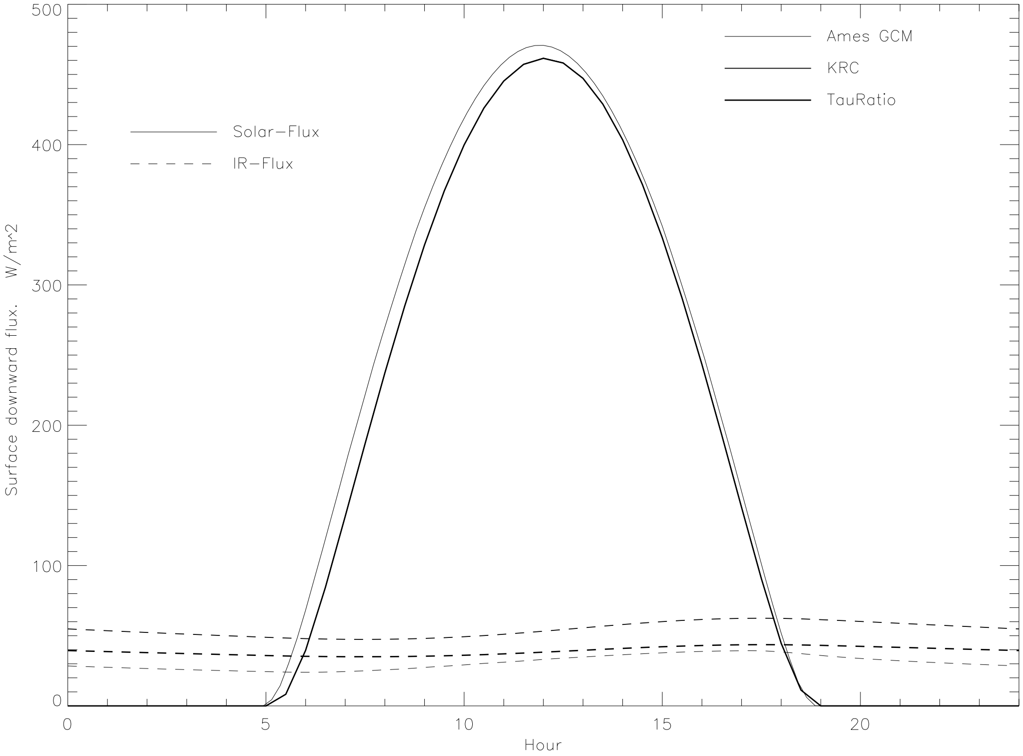

Figure 5: Downward solar and infrared radiation fluxes at the surface for the AMES GCM and KRC (heavy lines) models.

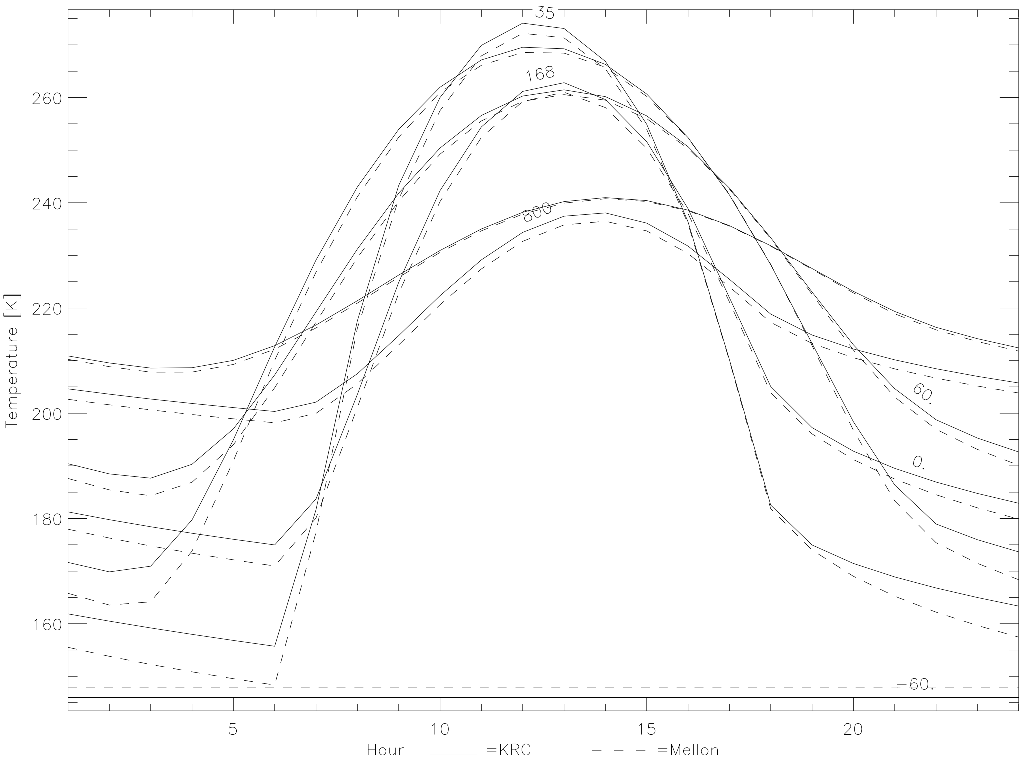

Figure 6: Comparison of KRC with the Mellon models used for TES standard production. Diurnal curves for TI of 35,168 and 800 (labeled for latitude 0) for latitudes 60S, 0 and 60N (labeled for inertia 168), all at $L_s = 100°$. Both models had seasonal frost all day long at 60S. KRC models are a few degrees warmer, the greatest at night and for low thermal inertia.

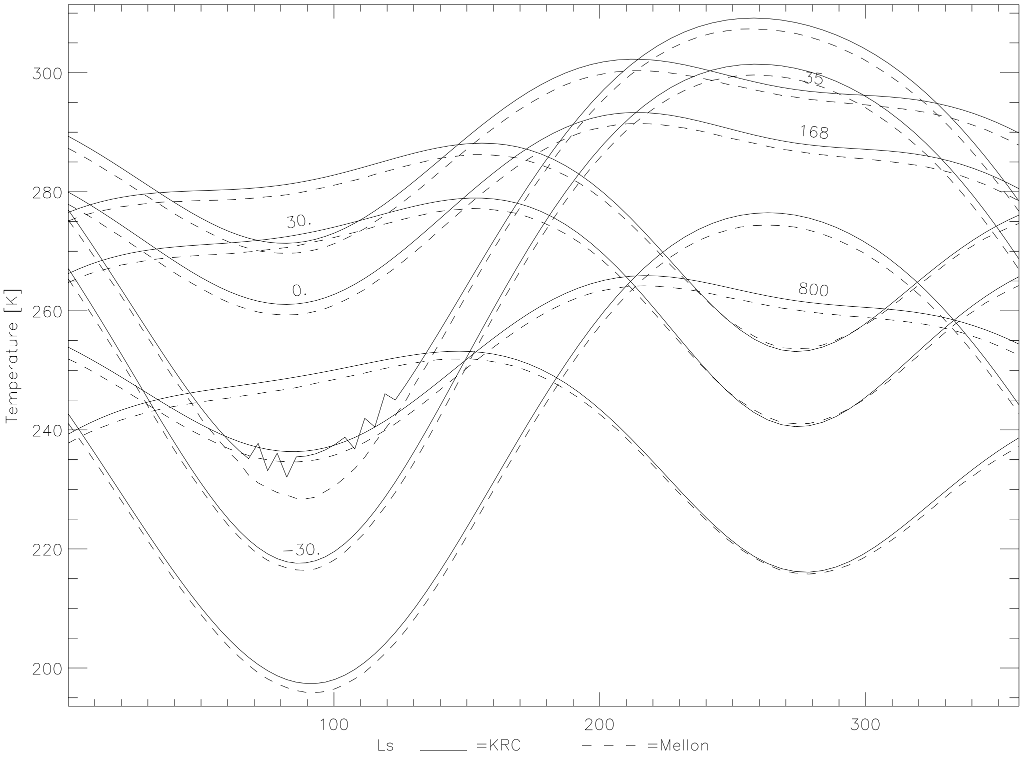

Figure 7: Comparison of KRC with the Mellon models used for TES standard production. Seasonal curves for TI of 35,168 and 800 for latitudes 30S, 0 and 30N, all at 13H. The models track each other closely except for the lowest inertia at 30S near $L_s = 90°$, when CO_2frost forms at night.

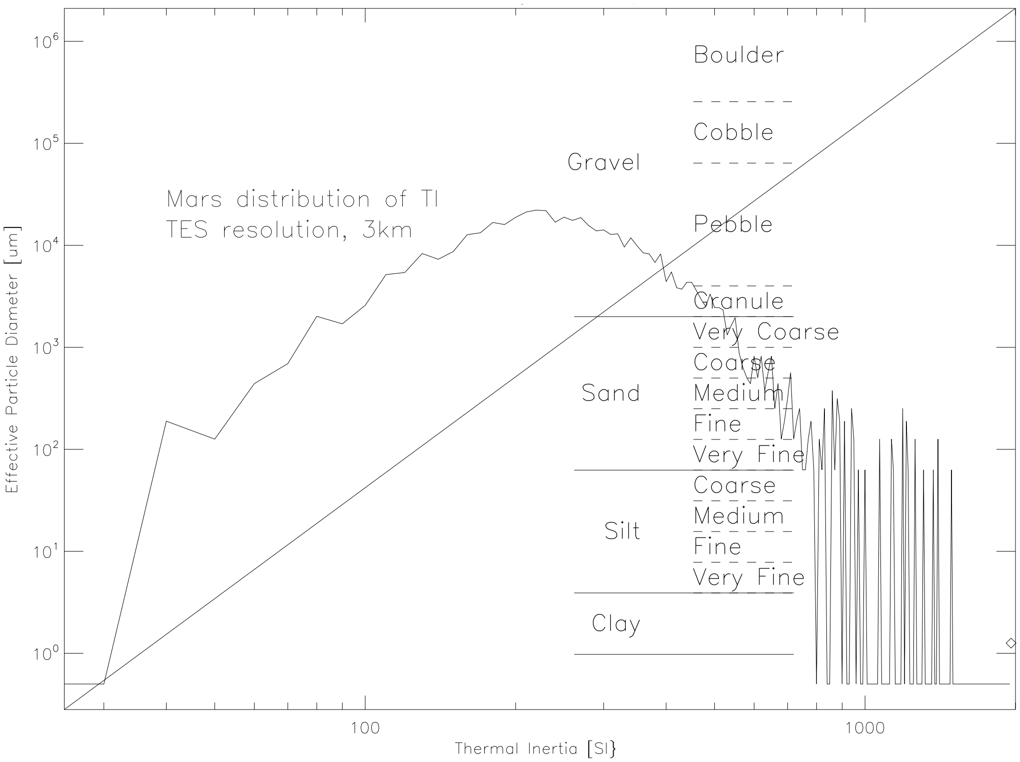

Figure 8: Nominal relation between thermal inertia and effective particle size, shown as the nearly straight line; specific conditions are P=600 Pascal, density=1600 kg/m3 and specific heat=625. The size designations are standard Wentworth scale. The areal distribution of thermal inertia on Mars, between latitudes 80S and 75N is shown as the jagged line, derived from mapping using TES data, [47] Note the log scales; most areas are in TI range of 100:500; values above the peak are increasingly affected by a rock population or real bedrock. The small diamond at high TI indicates the sum of all values above the plotted range.

D Binary file format

Other than direct access files, binary output files are in “bin5” format, described below. None contain the optional free text section.

```
           Description of the "bin5" file system
               Hugh Kieffer  2002sep02

The bin5 file system is designed to allow transfer of binary arrays between
different languages, operating systems and hardware types. Each bin5 file
contains a leading 512 byte (or multiple thereof) region that is ASCII text
and describes the array which follows it.

The first few words are integers separated by spaces, these define
the array size and type
  The first integer is the number of dimensions to the array, N.
  The next N integers are the sizes of each dimension, in the order of
    most-rapidly varying index first,

Then comes an integer that defines the word type; this follow the IDL
convention. Values greater than 4 are not supported in Fortran as of 2002sep.

       1 Byte
       2 Integer
       3 Longword integer
       4 Floating point
       5 Double-precision floating
       6 Complex floating
       7 String
       8 Structure
       9 Double-precision complex
      10 Pointer
      11 Object reference
      12 Unsigned Integer
      13 Unsigned Longword Integer
      14 64-bit Integer
      15       Unsigned 64-bit Integer

The next integer is the number of elements in the array, this is redundant with
the product of the dimension sizes.

Files written after 1999march should have a final integer that indicates the
number of leading ASCII bytes in the file; this is always a multiple of 512.

Next is some text bounded by << .. >>; the first part will usually be
"<<IDL_SIZE + headlen " , the latter part should be the creation date of the
file, e.g.," Tue Jul 9 05:08:18 2002 >>"

This is followed by the "header", which is free text describing the file. Files
from Hugh Kieffer may contain Keyword=value sections and/or embedded small
arrays. Embedded arrays use a separator,such as |,#, or ^, between each element,
the separator is doubled to indicate the limits of the embedded array.  E.g.,
Geom=||7.|0.|7.|0.|0.|1.||  Such arrays are created/read by strum.pro

The last 5 bytes of the ASCII section should always be "C_END"; these should be
immediately preceeded by a 5- or 8-byte indication of the source hardware
```

architecture, e.g. "x86 ".

It may be convenient to define an alias to look at the ASCII section:

```
  alias bhead  'dd count=1 if=\!*'  # display first 512  bytes of one binary file
  alias bhead2 'dd count=2 if=\!*'  # display first 1024 bytes of one binary file
```

The Flagstaff group has software for writing/reading bin5 files in IDL, Fortran and C.

Example:

```
 2 32 2 4 64 512 <<IDL_SIZE + headlen Tue Jul  9 05:08:18 2002 >>synmoonspec:
Multiply factor for ROLO irrad model r311f and Solar model= solar_bb. Synthetic
spectrum: Match mean and StdDev of Apollo with Breccia fraction: 0.05. Geom=
||7.|0.|7.|0.|0.|1.||


                                x86  C_END
```

This example defines a single-precision floating-point array sized (32,2)

E Helplist

A guide to setting up the input file is contained in helplist.txt, which is included here.

```
  KRC: PLANETARY SURFACE TEMPERATURES
HELPLIST.TXT  2009 Feb
        Hugh Kieffer. Original code ~1969, many revisions.

Major changes:

2002jul12-17 Replace atmosphere with Delta-Eddington model, and atmospheric
temperature based on solar and IR energy balance.

2008nov-2009feb Add capability for temperature-dependant thermal conductivity;
and revision of KRCCOM.

The evolution of KRC code is contained in evolve.txt.



================================================================================

METHOD

See the LaTeX document for a more detailed description: tes/krc/jpap.tex

Program is designed to compute surface and subsurface temperatures for a global
set of latitudes at a full set of seasons, with enough depth to capture the
annual thermal wave, and to compute seasonal condensation mass. For historic
reasons, the code has substantial optimization. There are generalities that
allow this code set to be used for any solid body with any spin vector, in any
orbit (around any star); this is also the source of some of the complexity.

Method is explicit forward finite differences with exponentially increasing
layer thickness and binary time increase with depths where allowed by stability.
Depth parameter is scaled to the diurnal thermal skin depth.  Initially starts
at 18 hours with the mean temperature of a perfect conductor.  Second degree
perturbation is applied at the end (midnight) of the (third) day; this jumps the
mean temperature of all layers and the lower boundary to equal the mean surface
temperature.

Boundary condition treatment:
  Perturbation solution of quartic equation at surface for each iteration;
    temperature gradient assumed uniform in top interval.
  Lower boundary may be insulating or constant-temperature.

Atmospheric Radiation:
KRC uses a one-layer atmosphere that is grey in both the solar and infra-red
regions. parametric atmosphere. The default atmospheric parameters are based on
estimates of Mars' gas and aerosol properties.
  Delta-Eddington model for insolation; direct onto sloped surface and diffuse,
with possible twilight extension.
  Atmosphere temperature based on Delta-Eddington solar absorption and IR opacity
[Pre 2002jul16
  First-order treatment of scattering of solar radiation.
```

Diurnal temperature is modeled as sinusoidal with phase shift.]

Keplerian orbital motion; seasons are at uniform increments of time. Mean orbital elements are pre-calculated for any epoch (all planets and several comets) by the PORB code set.

Units are SI; days for orbital motion. (Revised from cal-cgs, 97july)

Options:
  Different Physical properties below a set layer (IC).
  Regional slope
  Three ways to handle seasonal global pressure variation

Atmosphere condensation:
Global integral of CO2 frost-gas budget can control surface pressure.
Allows different surface elevation for each latitude zone.
  Zonal frost saturation temperature tracks local surface pressure.
Option for cap albedo to depend upon mean daily insolation.

CONVERGENCE NOTES

Convergence prediction routine can't jump more than one time constant (TAU=X**2/2) for the total thickness. Therefore, if X(N1) is small, make DDT smaller than usual. If DELJUL is much smaller than (X(N1))**2/2, then DDT can be as large as 0.3. Otherwise DDT must be about 0 for the prediction routine to work well (it assumes the 3rd derivative to be 0).

- - - - - - - - - - INPUT FILE - - - - - - - - - -

All parameters for KRC are set by a formatted text file. An example is master.inp , which has default values for a 19 latitude set for a run of three martian years, with the last output to disk. Parameter values are listed below their titles, which are in many cases identical to the code name, and last charater of the title is above the last location in the field. Thus, integer values MUST be aligned. Titles with a leading "[" indicate that the value is not used. The recommended procedure is to copy master.inp and edit only the values you wish to change. The number of lines of Latitudes and Elevations must match the value of N4, e.g., 2 lines for N4=11:20, entries beyond the N4 position may be left blank or contain the end of the line. The 13 lines following Elevations are a geometry matrix for Mars orientation and orbit in 2005, and should not be touched; they can be replaced by running PORBMN carefully.

The first input line is always KOLD,KEEP (I*), which sets file usage.
  These are described near the end of this help file under DISK BINARY FILES.
The second line is free text where you can outline the purpose of your run.
 If KOLD=0, then a full set of input values is read.

Change lines may follow immediately after the geometry matrix (see PARAMETER CHANGES section below) . The end of definition of a "case" is indicated by a "0/" line. Two successive "0/" lines ends the run.

Items with numbers inset 2 spaces below are computed, not input.
The source code for 'krccom.inc' indicates which subroutine sets many of the parameters; as the routine name in lowercase just below the parameter name.
 -

```
Type 4 Title (20A4) 80 characters of anything to appear at top of each page.

Type 1 Real parameters  (8F10.2) ================================
   Surface Properties
1 ALB Surface albedo
2 EMIS Surface emissivity
3 SKRC Surface thermal inertia [J m^-2 s^-1/2 K^-1] { cal cm * 4.184e4}
4 COND2 Lower material conductivity (IC>0)
5 DENS2 Lower material density (IC>0)
6 PERIOD Length of solar day in days (of 86400 seconds)
7 SPHT Surface specific heat [J/Kg/K]  {cal/g/K * 4184.}
8 DENS Surface density [kg/m^3] {g/cubic cm. *10}
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   Atmospheric Properties
9 CABR Atmospheric infrared back radiation coefficient
                2002jul16 IR opacity of dust-free atmosphere
10 AMW Molecular weight of the atmosphere
11 [ABRPHA UNUSED [Phase of ABRAMP, degrees relative to midnight]
12 PTOTAL Global annual mean surface pressure at 0 elev., Pascal[=.01mb]
13 FANON Mass-fraction of mean atmosphere that is non-condensing
14 TATM Atm temp for scale-height calculations
-   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -
15 TDEEP Fixed bottom temperature. Used if IB<=1.
16 SPHT2 Lower material specific heat (IC>0)
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  Dust  & Slope Properties
17 TAUD Mean visible opacity of dust, solar wavelengths
18 DUSTA Single scattering albedo of dust
19 TAURAT Ratio of thermal to visible opacity of dust
20 TWILI Twilight extension angle [deg]
21 ARC2 Henyey-Greenstein asymmetry factor
  moon        = eclipse start time in local Hours
22 [ARC3 NOT USED  coeff. for planetary heating
  moon        = eclipse duration in seconds 0=no eclipse
23 SLOPE Ground slope, degrees dip. Only pit may slope beyond pole.
24 SLOAZI Slope azimuth, degrees east from north. <-360 is a pit
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
    Frost Properties
25 TFROST Minimum Frost saturation temperature
  may be overridden by local saturation temperature (LVFT)
26 CFROST Frost latent heat [J/Kg] {cal/gm*4184.  [ Not used if
27 AFROST Frost albedo, may be overridden (LVFA) [ TFROST never
28 FEMIS Frost emissivity   [ reached
29 AF1 constant term in linear relation of albedo to solar flux
30 AF2 linear term in relation of albedo to solar flux units=1/flux
   Afrost = AF1 + AF2 * <cos incidence> SOLCON / DAU^2
31 FROEXT Frost required for unity scattering attenuation coeff. [Kg/m^2]
                the greater of this and 0.01 is always used.
32 fd32    UNUSED
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   Thermal Solution Parameters
33 RLAY Layer thickness ratio
34 FLAY First layer thickness (in skin depths)
35 CONVF Safety factor for classical numerical convergence
 0 for no binary time division of lower layers
 >0.8 for binary time division. Larger is more conservative
```

```
36 DEPTH Total model depth (scaled) (overrides FLAY if not 0.)
37 DRSET   Perturbation factor in jump convergence. If = 0., then
  all layers reset to same average as surface layer. Else,
                  does quadratic curve between surface and bottom averages
38 DDT Convergence limit of temperature RMS 2nd differences
39 GGT Surface boundary condition iteration test on temperature
40 DTMAX Convergence test: RMS layer T changes in a day
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   Orbit Geometry & Constants
41 DJUL Starting Julian date of run -2440000(N5>0)
42 DELJUL Increment between seasons in Julian days (if N5>1)
43 SDEC Solar declination in degrees. (if N5=0)
44 DAU Distance from Sun in astronomical units (if N5=0)
45 SUBS Aerocentric longitude of Sun, in degrees. For printout
  only. Computed from date unless N5=0(for printout only)
46 SOLCON Solar constant Applied Optics 1977 v.16, p.2693: 1367.9 W/m^2
                    1366.2 Based on figure in Frohlich, Observations of
    irradiance variations, Space Sci. Rev.,94,15-24,2000
47 GRAV Surface gravity.  MKS-units
48 AtmCp   Specific heat at constant pressure of the atmosphere [J/Kg/K]
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   Temperature dependent conductivity.  Ignored unless LKOFT set.
49     CONUP0  Constant coef for upper material
50     CONUP1  Linear   in k=c0+c1x+c2x^2+c3x^3 where x=(T-220)*0.01
51     CONUP2  Quadratic    "
52     CONUP3  Cubic coeff. "
53     CONLO0  Constant coef for lower material
54     CONLO1  Linear      as for CONUP above
55     CONLO2  Quadratic    "
56     CONLO3  Cubic coeff. "
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
COMPUTED REAL*4 VALUE
  57    HUGE = 3.3E38  nearly largest  REAL*4 value
  58    TINY = 2.0E-38  nearly smallest REAL*4 value
  59    EXPMIN = 86.80  neg exponent that would almost cause underflow
  60    fd60(2) Spare
  61
  62    RGAS = 8.3145  ideal gas constant  (MKS=J/mol/K)
  63    TATMIN  Atmosphere saturation temperature
  64    PRES    Local surface pressure at current season
  65    OPACITY Solar opacity for current elevation and season
  66    TAUIR   current thermal opacity at the zenith
  67    TAUEFF  effective current thermal opacity
  68    TATMJ   One-layer atmosphere temperature
  69  SKYFAC fraction of upper hemisphere that is sky
  70  TFNOW frost condensation temperature at current latitude
  71  AFNOW frost albedo  at current latitude
  72  PZREF Current surface pressure at 0 elevation, [Pascal]
  73 SUMF Global average columnar mass of frost [MKS]
  74 TEQUIL Equilibrium temperature ( no diurnal variation)
  75 TBLOW Numerical limit (Blowup) temperature
  76 HOURO Output Hour requested for "one-point" model
  77 SCALEH Atmospheric scale height
  78 BETA Atmospheric IR absorption
  79 DJU5 Current Julian date (offset 2440000 ala PORB convention)
  80 DAM Half length of daylight in degrees
  81 EFROST Frost on the ground at current latitude [Kg/m^2] {g/cm^2 * 10.}
  82 DLAT Current latitude
  83 COND  Top material Thermal conductivity (for printout only)
  84 DIFFU Top material Thermal diffusivity (for printout only)
  85 SCALE Top material Diurnal skin depth (for printout only)
  86 PI pi
  87 SIGSB Stephan-Boltzman constant (set in KRC)
  88 RAD Degrees/radian

Type 2 Integer Parameters (8I10) ===================================

1 N1 # layers (including fake first layer) (lim MAXN1)
2 N2 # 'times' per day (lim MAXN2). Must be an even number,
  should be a multiple of N24 and NMHA.
3 N3 Maximum # days to iterate for solution (lim MAXN3)
  98sep03 This can be 1, but then must use  DELJUL ~= PERIOD
If N3 lt 3, first day starts on midnight. else at 18H
4 N4 # latitudes (lim MAXN4=19). Global integrations done for N4>8
5 N5 # 'seasons' total for this run. If 0, then DAU and SDEC will be
used as entered for a single season.
6 N24 # 'hours' per day stored, should be divisior of  N2 (lim MAXNH)
7 IB Bottom control: 0=insulating, 1=constant temperature
  2=start all layers =TDEEP & constant temperature
8 IC First layer (remember that 1 is air) of changed properties.
  0 or 999=homogeneous, or 3 to N1-2
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
9 NRSET # days before reset of lower layers;  >N3=no reset
10 NMHA # 'hour angles' per day for printout (no limit)
11 NRUN Run #; appears in some printout
12 JDISK Season count that disk output is to begin
13 IDOWN Season at which to read change cards
14 I14 Index in FD of flexible print
15 I15  ""
16 KPREF Mean global pressure control. 0=constant
1= follows Viking Lander curve  2=reduced by global frost, but
                then N4 must be >8, and latitudes must be monotonic increasing
                and must include both polar regions (no warning for your failure)
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
17 K4OUT Disk output control: See details in DISK BINARY FILES section
Three modes of direct access Fortran files;  one case per file.
      -=KRCCOM(once), then TSF & TPF;
      0=KRCCOM,LATCOM each season
   1:49=KRCCOM,DAYCOM for the last latitude; each season
 Modes of bin5 file for multiple cases
   51=(Hours, 2 min/max,  lat, seasons, cases)
   52=(hours,  7 items, lat, seasons, cases)
   54=[many seasons, 5 items,lats, cases]
   55=[many seasons,9 items, cases]
   56=[packed T hour and depth, latitude,season,case]

18 JBARE J5 season count at end of which to set frost amount to 0. 0=never
19 NMOD Spacing of season for notification. minimum of 1
20      IDISK2  NOPE Season count at which to reset deep layer temperatures
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
COMPUTED I*4 VALUES
  21 KOLD    Season index for reading starting conditions
  22    id22(6) 22 and 23 used as flags for season-variable ALB and TAUD
  28 NFD Number of real items read in
  39 NID Number of integer items read in
  30 NLD Number of logical items read in
  31 N1M1 Temperature vrs depth printout limit (N1-1)
  32 NLW Temperature vrs depth printout increment
  33 JJO Index of starting time of first day
  34 KKK Total # separately timed layers
  35 N1PIB N1+IB Used to control reset of lowest layer
  36 NCASE Count of input parameter sets in one run
  37 J2 Index of current time of day
  38 J3 Index of current day of iteration
  39 J4 Index of current latitude
  40 J5 Index of current "season"

Type 3 Logical Parameters (10L7) ==================================

1 LP1 Print program description. TPRINT(1)
2 LP2 Print all parameters and change cards (2)
3 LP3 Print hourly conditions on last day (3)
4 LP4 Print daily convergence summary (4)
5 LP5 Print latitude summary (5)
6 LP6 Print TMIN and TMAX versus latitude and layer (6)
7 LPGLOB Print global parameters each season
8 LVFA Use variable frost albedo. Uses AF1 & AF2 (real # 29,30)
9 LVFT Use variable frost temperatures
10 LKOFT   Use temperature-dependent conductivity
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
11 LPORB Call PORB1 just after full input set
12 LKEY Read change item from terminal after main input set
13 LSC Read change cards from input file at start of each season
14 LNOTIF spare
15 LOCAL Use each layer for scaling depth
16 LD16 Print hourly table to FORT.76 [TLATS]
17 LPTAVE Print <T>-<TSUR> at midnight for each layer [TDAY]
18 LD18 Output to fort.78 [TLATS] insolation and atm.rad.coefficents
19 LD19 Output to fort.79 [TLATS] insolation and atm.rad. arrays
20 LONE (Computed) Set TRUE if KRC is in the "one-point" mode
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
followed in 'krccom' by:
[real*4] TITLE(20) 80-character title
[real*4] DAYTIM(5) 20-character run  date and time
===========================================================

Latitude(s) (10F7.2) N4 latitudes in degrees, no internal separations.
Latitudes to be in order; south to north. [[If last latitude is
.LE. 0, will assume symmetric results for global integrations]]

Elevation(s) (10F7.2) N4 values in Km corresponding to latitudes

Orbital Parameters (LPORB=T) Format identical to that produced by PORB
program set ASCII file output. So these can be directly pasted with an
editor. see PORBCM.INC
 - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

PARAMETER CHANGES

Fortran List Directed. Change the values in KRCCOM
White-separated, a "/" terminates the read and leaves remaining values unchanged
The 4 items are: Integer Integer Numeric_value 'Text' / Comment

```
    1: Type (integer) see table below
    2: Index in array (integer), as listed in table above
    3: New value, numeric, will read as real and convert. 0.=false.
    4: Reason, text string within single quotes
   [ after a / (forward slash) nothing is read, so you can use for comments]
```

The print file will list each change as read, followed by the title of the changed item. It is a good idea to look at this print to be sure you changed what you intended.

Type	Meaning	Valid Index
0	End of Current Changes	
1	Real Parameter	1-56
2	Integer Parameter	1-20
3	Logical Parameter	1-20
4	New Latitude Card(s) Follow	
5	New Elevation Card(s) Follow	
6	New Orbital Parm Cards Follow (LPORB Must be True)	
7	Text becomes new Title	

8	Text becomes new disk or season-variation file name
	if index=22, read variable ALBEDO
	if index=23, read variable TAUD	

9	Complete new set of input follows	
10	Text becomes new One-Point input file name	
11	This is a set of parameters for "one-point" model For this type, 9 values must appear in a rigid format	
12	Set of 2*4 coefficents for T-dep conductivity. List-directed IO	

To start variable albedo, use input card:

```
  8 22 0 'AlbedoFileName' / Variable albedo text file name
```

Can revert to constant albedo by hokey technique of using a bad name. E.g.,

```
  8 22 0 'badName' / turn variable albedo off
```

Files of text table of value versus season will be read at the start of a run. These will apply to ALL latitudes. See example valb1.tab

Variable Tau done the same way, with 22 being replaced with 23

COMMON /LATCOM/ see latcom.inc
COMMON /DAYCOM/ see daycom.inc

Because the binding routines to IDL are intolerant of any errors, the items in the above commons have not been changed, Rather, in 2004July, and additional common was added as a "catch-all" for any new items.

COMMON /HATCOM/ see hatcom.inc

Error Returns:
"Parameter error in TDAY(1)" : Convergence factor < .8 classic.
Instability anticipated.
"UNSTABLE; Layer..... TDAY(1):

DRSET: 0=>Reset by delta_average_T for each layer:
else: reset by {linear + DRSET*quadratic}*{<surf>-<botm>}
TDAY: LRESET Reset midnight T's for all but top layer.
      LDAY Last day computations

------------------- Handy things ------------------------------

The first "hour" in printout and output arrays is 1/24 (strictly, 1/N24) of a sol after midnight. E.g., the last time is midnight, not the first.

Atmospheric scale height, SCALEH, depends upon physical constants and
  TATMAVE which (2007nov) is always = TATM, input. and GRAV , input

----- DISK BINARY FILES ----------

The routine TDISK is used to read or write direct-access binary files or bin5 files. The first season to write is specified by JDISK, all following seasons will go to the same file. For direct-access files, each file record consists of KRCCOM plus LATCOM or KRCCOM plus DAYCOM.

Disk output is largely controlled by the KRC and TSEAS routines.

 - - - Items which control file I/O - - - - -

KOLD & KEEP on first input line
  KOLD: 0= input card set follows; else=disk record number to start from,
 then will read any change cards.
If LPORB in old file was True, then there must be a PORB card set
  as the set of lines following the KEEP,KOLD line
  KEEP: 0= close disk file after reading seasonal record KOLD;
>0= value of JJJJJ at which to start saving seasons in same disk
file [overrides JDISK].
  To start from a prior seasonal run, need to determine the record
corresponding to the desired season;
 KOLD=J5_target - JDISK(old) ; >0
set KEEP=1, change card J5=number of new seasons, set K4OUT.

JDISK sets the first season to save results

N5   sets the last season to run

K4OUT sets the record content:
 -      Will output first record of KRCCOM,ALAT,ELEV, then records of TSF & TPF
 0 (normal) Will output records of KRCCOM+LATCOM
 +n<=50 Will output records of KRCCOM+DAYCOM for the last computed latitude.

 > 50 Will write custom bin5 file at the end of a run, with dimensionality from 3 to 5 (more possible). All 5x outputs allow multiple cases, each with a "prefix" for each case consisting of with 4 size integers (converted to Float) followed by KRCCOM; after this may come vectors of parameters versus season. The next-to-last dimension is increased to allow room for the prefix to be embedded in the bin5 array. KRC input items that would change any of the bin5 dimensions are not allowed to change between cases. Each dimension is adjusted to the

necessary size. Each case has the same structure; this simplifies coding although some items are then present redundantly. The number of cases allowed is set by the size of one case, and printed as MASE at the end of the first case in the print output. Cases beyond the maximum that can be stored will be executed, but not saved.

```
The first 4 words of the prefix, and of thus of the bin5 array, are:
(1)=FLOAT(NWKRC)   ! Number of words in KRCCOM
(2)=FLOAT(IDX)     ! 1-based index of dimension with extra values
(3)=FLOAT(NDX)     ! Number of those extra
(4)=FLOAT(NSOUT)   ! [Available of other use]

    51=(N24 hours, 2: TSF TPF, N4 lats, NDX+ seasons, cases)
The prefix section contains: sub_array(seasons,5)(0-based index)
  0)=DJU5  1)=SUBS  2)=PZREF  3)=TAUD  4)=SUMF

    52=(N24 hours, 7 items, N4 lats, NDX+ seasons, cases)
The 7 items are:  1)=TSF  2)=TPF  3)=TAF  4)=DOWNVIS  5)=DOWNIR
6) packed with [NDJ4,DTM4,TTA4,      followed by TIN(2+
7) packed with [FROST4,AFRO4,HEATMM, followed by TAX(2+
  The number of layers for TIN and TAX is the smaller of: the number computed
and that fit here.
The prefix is identical to Type 51

    54= (seasons, 5 items, NDX +nlat, cases)
        Items are (0-based index):
0= TSF=surface temperature at 1 am, 1= TSF at 13 hours,
2= HEATMM=heat flow, 3= FROST4=frost amount,
4= TTB4 = predicted mean bottom temperature
        The prefix contains DJU5

    55= (seasons,NDX+ items,cases).  For seasonal studies at one latitude
ITEMS intended to be recoded as needed. Initial version is 9 items:
        [Tsur@ 1am,3am,1pm, spare, Tplan @1am,1pm, Surface heat flow,
        frost budget, T_bottom]
        The prefix contains DJU5
 Can hold very large number of seasons and cases.
THIS MODE DOES NOT SUPPORT CONTINUATION RUNS

    56= [vectors&items, latitudes, NDX+ seasons, cases]
The first dimension is: TSF for all hours, TPF at all hours,
 T4 for all layers at midnight, then FROST4,HEATMM,TTA4
The prefix is identical to Type 51
```

Once a disk file is opened, any records written will go into that file until a new filename is specified (Type 8 Change line), which closes the current file.

To run & save various cases for a single season, set N5 and JDISK to 1.

To extract a detailed day by saving DAYCOM to disk, set JDISK=N5, set a new file name, and set K4OUT to desired latitude index (normally 1):

To run continuously with output every K ((1-3) days, set DELJUL=K*PERIOD
this will force prediction terms to near 0.
setting N3=1 will turn off all prediction.
set GGT large (to avoid iteration for convergence)

set NRSET=999 (to avoid reset of layers)

To continue run with new parameters (e.g., DELJUL)
3 21 1 'flag set to continue'
  Note: changing DELJUL will cause reset of DJUL
Must increase the value of N5: 2 5 bigger 'Increase stopping season'
Reset will not occur because J5 continues incrementing

ASCII Output Files

 krc.prt general; results, what is output is controlled by LP1:6 & LPGLOB

fort.76
tlats.f: mimic Mike Mellon ASCII files

```
        if (ld16) then
          write(76,761)subs,dlat,alb,skrc,taud,pres
 761      format(/,'      Ls      Lt       A       I    TauD       P'
 762      format(f7.2,f9.3,f8.3,f9.3)
            write(76,762)qh,tsfh(i),adgr(j),qs

          do i=1,n24
            j=(i*n2)/n24
            qh=i*qhs
            qs=(1.-alb)*asol(j) ! absorbed insolation
            write(76,762)qh,tsfh(i),adgr(j),qs
          enddo
```

fort.78
tlats.f: for average and maximum:

```
        if (ld18) write(78,*)cosi_(i), t_(i),ADGs(i),ADGP(i)
        if (ld18) write(78,*)j5,j4,sol,ave_a,adgir,c52,beta
```

fort.79
tlats.f: for each time-step

```
       if (ld19) write(79,*)adgr(jj),qa,direct,diffuse
```

   col 1 = downgoing thermal radiation
   col 2 = total insolation reaching surface
   col 3 = direct fraction of insolation
   col 4 = diffuse fraction of insolation

------------------ To run two material types (2000jan23)

Set IC to the first layer to have the lower material properties (>= 3)
Set COND2 to the lower material conductivity
Set DENS2 to the lower material density
Set SPHT2 to the lower material specific heat
If LOCAL is False, then initial setting of all layer thicknesses is based upon the scale of the upper material; if it is set True, the thickness of the lower layers is set by their scale.

  TDAY no longer allows unstable (thin) layers, and will increase the thickness of the layer IC to satisfy the convergence safety factor FCONV if needed. However, the code to check on convergence was retained.

------------------ Setting temperature-dependant conductivity

Basic Flag is L10=LKofT . If this is true, then the 8 input parmeters CONUP0 to CONLO3 must be set to yield thermal conductivity as a function of temperature for the upper and lower materials. k=c0 +c1x + c2x^2 +c3x^3 where x=(T-200.)*0.01

One way to generate the coefficents is to run for each of the upper and lower materials the IDL program koftop, which calls koftfit, which calls spspread; this last mimics the Piqueux relation for un-cemented soils. koftop allows change of its parameters, including grain radius and pressure, and will print the required parameters.

Below are sample coefficients based on Sylvain Piqueux's numerical model for un-cemented soils; the fit error is <0.1% over 120-320K. Left column is grain radius in micrometers, then the four normalized coefficients ready for inclusion in a KRC input file, followed by the thermal inertia at 220K for nominal density and specific heat.

R(mu)	c0	c1	c2	c3	Iner
10.	0.008274	0.000735	-0.000376	0.000148	89.8
20.	0.012379	0.001280	-0.000629	0.000250	109.9
50.	0.021485	0.002647	-0.001201	0.000483	144.7
100.	0.032051	0.004528	-0.001874	0.000761	176.8
200.	0.046023	0.007569	-0.002743	0.001129	211.8
500.	0.068387	0.014075	-0.003874	0.001687	258.2
1000.	0.086303	0.021288	-0.004146	0.002099	290.1
2000.	0.103743	0.030909	-0.003141	0.002535	318.0
5000.	0.127172	0.049907	0.002019	0.003469	352.1
10000.	0.149810	0.074734	0.011546	0.004939	382.2
20000.	0.185706	0.119913	0.030938	0.007877	425.5
50000.	0.283361	0.250283	0.089327	0.016714	525.6

RUNNING THE "ONE-POINT" MODE (2002mar08)

A parameter initialization file Mone.inp is provided. It sets the KRC system into a reasonable mode for one-point calculations. Do not change that file unless you have read this entire file.

A line near the end of that file points to a file 'one.inp' which can contain any number of one-point conditions. 'one.inp' is intended to be edited to contain the cases you want; however, it must maintain the input format of the sample file.

First Line is any title you wish. It must be present.
The second line is an alignment guide for the location lines. It must be there.

Each following line must start with an '11 '; this is a code that tells the full-up KRC that is a one-point line. The next 9 fields are read with a fixed format, and each item should be aligned with the last character of the Column title. All items must be present, each line must extend at least to the m in Azim; comments may extend beyond that, but they will not appear in the output file. Be sure to have a <CR> at the end of the last input line.

The fields (after the 11) in the one-point input are:
Ls L_sub_S season, in degrees
Lat Aerographic latitude in degrees
Hour Local time, in 1/24'ths of a Martian Day
Elev Surface elevation (relative to a mean surface Geoid), in Km
Alb Bolometric Albedo, dimensionless
Inerti Thermal Inertia, in SI units
Opac Atmospheric dust opacity in the Solar wavelength region
Slop_ Regional slope, in degrees from horizontal
Azim Azimuth of the down-slope direction, Degrees East of North.

The two additional columns in the output file are:
   TkSur Surface kinetic temperature
TbPla Planetary bolometric brightness temperature

Try running the binary file first. If that fails, a Makefile is provided to complile and link the program; simply enter "make krc" and pray. If this fails, have your local guru look over the Makefile for local dependancies. Suggestions of making the Makefile more universal are welcome.

To run the program, change to the directory where the program was built, and enter "krc". You should get a prompt:

```
       ?* Input file name or / for default =
       Mone.inp
```

If the initialization file still has this name and is in the same directory, enter a single "/" and <CR>. Otherwise, enter the full pathname to the initialization file, with no quotes and no blanks.

A second prompt is for the name of the output file:

```
       ?* Print file name or / for default =
       krc.prt
```

Again, if this is satisfactory, simply enter / <CR> , else enter the desired file path-name.

----- Comments on the One-point model.

The initialization file of 2002mar08 is set to compute the temperatures at the season requested without seasonal memory. It uses layers that extend to 5 diurnal skin depths. It does not treat the seasonal frost properly, so don't believe the results near the edge of the polar cap. Execution time on a circa 2001 PC may be the order of 0.01 seconds per case.

The underlying model is the full version of KRC. By modifying the initialization file, you can compute almost anything you might want. If you choose to try this, best to read all of this document.

Reading type 5x files

Routines do not access files directly unless specifically listed.

DEFINEKRC Define structures in IDL that correspond for Fortran commons
Calls: None other than IDL
Firm code of common definitions. Must be recoded if a Fortran *.inc changes

```
READKRCCOM  Read a KRCCOM structure from a bin5 file
uses 3-element HOLD array. Returns a structure of krccom
Options to open or close bin5 file or read one case
Calls: DEFINEKRC
Files: bin5
HOLD is: 0]=logical unit  1]=number of words in a case  2]=# cases in the file

MAKEKRCVAL Make string of selected KRC inputs: Key=val
Calls: DEFINEKRC

KRCHANGE  Find changes in KRC input values in common KRCCOM
Calls:  READKRCCOM  MAKEKRCVAL
Reads and stores krccom for first case. For each additional case, makes a
list of any changes in the flaot, integer or logical input values.

KRCCOMLAB Print KRC common input items
 all items via arguments
Calls: None

KRCLAYER  Compute center depth of KRC layers
 all items via arguments
Calls: none

 KRCCOMLAB Print KRC common input items
 all items via arguments
Calls: None

KRCSIZES  Compute array and common sizes for KRC Fortran
Test procedure to compute array sizes or hours.
Must recode if any size in *.inc changes
```

The following listing of all commons was generated by these Linux commands:

```
cd /home/hkieffer/krc/tes
rm allinc.txt
cat krccom.inc latcom.inc daycom.inc hatcom.inc filcom.inc units.inc
../porb/porbcm.inc > allinc.txt
```

```
C_Titl  KRCCOM.INC  common for input and transfer variables
C_Limitations
IMPLICIT REAL*4 (A-H,O-Z), INTEGER*4 (I,J,K,M,N), LOGICAL*4 (L) !std.,+L
C Here are all the dimension-defining parameters for items in any common
PARAMETER (MAXN1 =30)    ! dimension of layers
PARAMETER (MAXN2 =384*4)   ! dimension of times of day: 384=24*16
PARAMETER (MAXN3 =16)    ! dimension of iteration days
PARAMETER (MAXN4 =37)    ! dimension of latitudes
PARAMETER (MAXN5 =161)    ! dimension of seasons
PARAMETER (MAXN6 =6)    ! dimension of saved years
PARAMETER (MAXNH =48)    ! dimension of saved times of day
PARAMETER (MAXBOT=6)    ! dimension of time divisions
PARAMETER (MAXN1P=MAXN1+1) ! dimension layer temperature points
PARAMETER (NUMFD=56+32, NUMID=40, NUMLD=20)! number of each type
PARAMETER (NWKRC=NUMFD+NUMID+NUMLD+25+2*MAXN4)
! above is size of common in 32-bit words
C_Description.
```

```
C  Cset -> routine which sets value.  seas=TSEAS  lat=TLAT
C *= initially set as input, this routine may reset value.
C lines 1-5 = real: input via  TCARD
C   6 =   " : transfer
C 7-8 = integer: input via  TCARD
C 9-A =    "   : transfer
C B-C = logical: input via  TCARD
C D   = title (input via tcard) and run_time (transfer)

COMMON /KRCCOM/
     1 ALB,EMIS,SKRC,COND2,DENS2, PERIOD,SPHT,DENS,CABR,AMW          ! 1:10
     2,ABRPHA,PTOTAL,FANON,TATM,TDEEP, SPHT2,TAUD,DUSTA,TAURAT,TWILI ! 11:20
     3,ARC2,ARC3,SLOPE,SLOAZI,TFROST, CFROST,AFROST,FEMIS,AF1,AF2    !   :30
     4,FROEXT,FD32,RLAY,FLAY,CONVF, DEPTH,DRSET,DDT,GGT,DTMAX        !   :40
Cset   *day1
     5,DJUL,DELJUL,SDEC,DAU,SUBS, SOLCON,GRAV,ATMCP,CONUP0,CONUP1    !   :50
Cset          v total of 56 input
     5,CONUP2,CONUP3,CONLO0,CONLO1,CONLO2, CONLO3,HUGE,TINY          !   :58
     &,EXPMIN,FD60(2),RGAS,TATMIN,PRES,OPACITY,TAUIR,TAUEFF,TATMJ    !   :68
Cset   *seas     ----seas-------
     6,SKYFAC,TFNOW,AFNOW,PZREF,SUMF, TEQUIL,TBLOW, HOURO,SCALEH,BETA !  :78
Cset   -----------lats--------- tint ---lats-------      ---lats----
     6,DJU5,DAM,EFROST,DLAT,COND,  DIFFU,SCALE,PI,SIGSB,RAD       !79:88
Cset   seas lat day2   lats -------day1------- --main------
     7,N1,N2,N3,N4,N5,  N24,IB,IC,NRSET,NMHA                      !  1:10
     8,NRUN,JDISK,IDOWN,I14,I15,  KPREF,K4OUT,JBARE,NMOD,IDISK2   ! 11:20
     9,KOLD,KVALB,KVTAU,ID24(4),  NFD,NID,NLD                     ! 21:30
Cset             ----card---
     A,N1M1,NLW,JJO,KKK,N1PIB,  NCASE,J2,J3,J4,     J5   ! 31:40
Cset   ---day1- lat ----day1-    main -day2- lats  seas
     B,LP1,LP2,LP3,LP4,LP5,    LP6,LPGLOB,LVFA,LVFT,LKOFT         !  1:10
     C,LPORB,LKEY,LSC,LNOTIF,LOCAL,   LD16,LD17,LD18,LD19,LONE   ! 11:20
     D,TITLE(20),DAYTIM(5)   ! these two make up 100 bytes
Cset    tcard    tprint
     &,ALAT(MAXN4)  ! latitude in degrees. set in  TCARD
     &,ELEV(MAXN4)  ! elevation in km. set in  TCARD
C
REAL*4    FD(NUMFD)
        INTEGER*4 ID(NUMID)
        LOGICAL*4   LD(NUMLD)
EQUIVALENCE (FD(1),ALB), (ID(1),N1), (LD(1),LP1)

C_Hist 85may12  Hugh_H_Kieffer  USGS_Flagstaff major revision
C 86jul01 HHK add comments of where variable is set
C  97feb11  HHK revise many locations and sizes
C  97sep08  HHK add  SCALEH,BETA   97sep11 replace  DIMENSION
C  98MAY26  HHK remove  L from implicit integer
C  00jan23  HHK redefine  BK->COND2  BRC->DENS2  FD16->SPHT2
C 2002mar01 HHK Make implict L logical*4
C 2002mar07 HHK LD20-->LONE.  move ALAT & ELEV from LATCOM, make id(21)=kold
C 2002jul16 HK Add OPACITY to common
C 2004jul07 HK Move dimension-defining parameters from other commons into KRCCOM
C 2008oct02-15 HK Move MAXBOT from DAYCOM to here. Replace ID22 with KVALB &
C    KVTAU and redefine ABRAMP to be AMW. New maximum dimensions.
C 2008nov13 HK Add T-dependent conductivity parameters.
C_End ____________________________________________________________________
```

```
C_Titl  LATCOM.INC   common for latitude-dependant items in  KRC
PARAMETER (NWLAT= (9+ 3*MAXN1 + 2*MAXNH) *MAXN4) ! size of common in 4-byte words
C_arg: all are set in  TLATS
      COMMON /LATCOM/
     & NDJ4(MAXN4)  ! # days to compute solution for each latitude
     &,DTM4(MAXN4)  ! rms temperature change on last day
     &,TST4(MAXN4)  ! predicted equilibrium temperature of ground
     &,TTS4(MAXN4)  ! predicted mean surface temperature for each latitude
     &,TTB4(MAXN4)  ! predicted mean bottom temperature
     &,FROST4(MAXN4)  ! predicted frost amount kg/m^2.
     &,AFRO4(MAXN4)  ! frost albedo.
     &,TTA4(MAXN4)  ! predicted final atmosphere temperature
     &,TTX4(MAXN4)  ! spare
     &,TMN4(MAXN1,MAXN4) ! predicted convergence midnight temperature
     &,TIN(MAXN1,MAXN4)  ! minimum hourly layer temperature
     &,TAX(MAXN1,MAXN4)  ! maximum hourly layer temperature
     &,TSF(MAXNH,MAXN4)  ! final hourly surface temperature
     &,TPF(MAXNH,MAXN4)  ! final hourly planetary temperature
C_Desc
C_Hist  85may12  Hugh_H_Kieffer    97feb11  HHK add  ELEV
C  97sep08  HHK add  TPF
C 2002mar09 HHK  Move ALAT & ELEV from here to KRCCOM
C 2002jul13 HK Add TTA4 and TTX4
C 2002aug15 HK NWLAT is computed, = 2793
C 2004jul07 HK Move dimension-defining parameters from other commons into KRCCOM
C 2009feb   HK With MAXN1=30, MAXNH=48, MAXN4=37, NWLAT is 7215
C_End _______________________________________________________________________

C_Titl  DAYCOM.INC   common for layer and time items in  KRC
PARAMETER (NWDAY = 6*MAXN1+1  + (5+MAXN1)*MAXN3 + 3*MAXN2
1    + 2*MAXNH + MAXBOT) ! size of this common in 4-byte words
C_Desc
C  ASOL &  ADGR are set in  TLATS, the rest are set in  TDAY
      COMMON /DAYCOM/ XCEN(MAXN1) ! Depth at layer centers [m]
     &, SCONVG(MAXN1) ! Classical convergence factor for each layer
     &, BLAY(MAXN1) ! Layer thicknesses [m]
     &, TMIN(MAXN1) ! Minimum layer temperatures of day on the hour
     &, TMAX(MAXN1) ! Maximum layer temperatures of day on the hour
     &, TTJ(MAXN1P) ! Layer temperatures (TTJ(1) is surface temperature)
     &, TT1(MAXN1,MAXN3)! Temperatures at start of day for each layer and day
     &, TTS(MAXN3) ! Mean daily surface temperatures
     &, TTB(MAXN3) ! Mean daily bottom temperatures
     &, TTA(MAXN3) ! End-of-Day Atmospheric temperatures
     &, DTMJ(MAXN3) ! RMS daily temperature change
     &, FRO(MAXN3) ! Daily frost amounts. [kg/m^2]
     &, ASOL(MAXN2) ! Insolation at each time of day, direct + diffuse
     &, ADGR(MAXN2) ! Atm. solar heating at each time of day
     &, TOUT(MAXN2) ! Surface temperatures of solution at each time of day
     &, TSFH(MAXNH) ! Hourly surface temperatures at solution
     &, TPFH(MAXNH) ! Hourly planetary temperatures at solution
     &, N1K(MAXBOT) ! Binary time division layers
C_Hist 84jun15  Hugh_H_Kieffer  97feb11  HHK add  ADGR
C    97mar03 correct  NWDAY       97sep08  HHK add  TPFH
C 2002jul12 HK Add TTA, ADGR was down-going atmospheric  IR radiance
C 2002aug15 HK NWDAY is computed, =1951
```

```
c 2002oct30 HK set N2 to 8*384 as trial  NWday=10015
C 2004jul07 HK Move dimension-defining parameters from other commons into KRCCOM
C 2008oct02 HK Move MAXBOT and MAXN1P from here to KRCCOM
C 2008nov13 HK Change names: X->XCEN  T->TTJ  TT->TT1  TLAY->BLAY
C_End ___________________________________________________________________________

C_Titl  HATCOM.INC   common to store post 2003 items in  KRC
C     NWHAT= 1+(3+4*MAXNH+MAXN6*MAXN1)*MAXN4 ! size of this common
      COMMON /HATCOM/ HEAT1M ! Mean upward heat flow into surface on last day
     &, HEATMM(MAXN4) !  " " for all latitudes <W m^-2>
     &, TEXTRA(MAXN4,2)  ! Extrapolation in surface/bottom temperature
     &, TAF(MAXNH,MAXN4)  ! final hourly atmosphere temperature, not predicted
     &, TOFALB(MAXNH,MAXN4) ! hourly top-of-atm albedo, not predicted
     &, DOWNVIS(MAXNH,MAXN4)    ! hourly net downward solar flux
     &, DOWNIR(MAXNH,MAXN4)     ! hourly net downward thermal flux
     &, TMN4Y(MAXN6,MAXN1,MAXN4)! midnight temperatures (year,layer,lat.)
C_Desc Designed for seasonal heatflow into annual frost
C_Hist 2004jul05 Hugh Kieffer
C 2004Oct05 HK Add DOWNVIS and DOWNIR
C 2009apr22 HK Add TMN4Y
C_End ___________________________________________________________________________

C_Titl FILCOM.INC  Common for file names
        COMMON /FILCOM/ FINPUT,FOUT,FDISK,FVALB,FVTAU,TITONE
CHARACTER*80    FINPUT,FOUT,FDISK,FVALB,FVTAU
CHARACTER*20 TITONE ! title for each one-point line
C_Hist   85oct14  Hugh_Kieffer  97feb12 increase string length
C 2002mar01 HK increase string length from 60
C 2006sep09 HK Add FVALB,FVTAU
C 2009may10 HK Add TITONE
C_End___________________________________________________________________________

C_Titl  UNITS.INC common  /UNITS/ for logical units and errors  KRC
        INTEGER*4 IOPM,IOKEY,IOIN,IOSP,IOERR,IOD1,IOD2,IOD3,IRTN,IERR
        LOGICAL*4 LOPN1,LOPN2,LOPN3,LOPN4
COMMON /UNITS/
     & IOPM ! prompt, usually terminal screen
     &,IOKEY ! interactive input, usually terminal keyboard
     &,IOIN ! input; input disk file or terminal
     &,IOSP ! printer (spooled)
     &,IOERR ! error messages, commonly =  IOSP
     &,IOD1 ! disk unit for explanation file (temporarily open)
     &,IOD2 ! disk output files
     &,IOD3 ! spare
     &,IRTN ! subroutine # in which error occured
     &,IERR ! error return code
     &,LOPN1,LOPN2 ! status of logical units
     &,LOPN3,LOPN4 !  .TRUE. means that one is currently open
C_Hist 1985----  Hugh_H_Kieffer
C 2004jul06 HK Explicit type statements
C_End____________________________________________________________________________

C_Titl  PORBCM.INC   include for PORB system COMMOM
C_Vars
      COMMON /PORBCM/ PLANUM,TC, ODE,CLIN,ARGP,ECC,SJA,TJO,PERIOD
     1 ,ZFQA,ZFQB,  SIDAY,TLO,  OBL, CQA,CQB,  P17(2)
     2 ,ZFEB,ZFEC,XFEXB,ARGV,SLP, P24
     3 ,PHOXX(3),PHEXX(3),HFXX(3), FQ(9),EO(9),HO(9)
     5 ,PI, DAYTIM(5), IOK,IOS,IOP,IOD
C_Desc   All angles in radians
C Conditions at request epoch
C
C         BEWARE  PI and PERIOD would conflict with krccom
C
C  1 PLANUM PLANET NUMBER, OUT FROM THE SUN (>9 ARE OTHER OBJECTS)
C  2 TC   TIME IN CENTURIES FROM REFERENCE DATE (1950.0)
C  3 ODE LONGITUDE OF THE NODE
C  4 CLIN INCLINATION
C  5 ARGP ARGUMENT OF PERIAPSIS
C  6 ECC ECCENTRICITY
C  7 SJA SEMI-MAJOR AXIS (ASTRONOMICAL UNITS)
C  8 TJO JULIAN DATE OF PERIHELION
C  9 PERIOD PERIOD OF THE ORBIT (JULIAN DAYS)
C 10 ZFQA PLANET POLE ORIENTATION IN EARTH EQUITORIAL SYSTEM: DECLINATION
C 11 ZFQB   "     "      "        "    "       "        "     RIGHT ACEN.
C 12 SIDAY SIDERIAL ROTATION PERIOD OF BIDY (HOURS)
C 13 TLO JULIAN DATE OF ZERO LONGITUDE TOWARDS PLANETS VERNAL EQUINOX ??
C 14 OBL OBLIQUITY OF THE EARTH'S AXIS
C 15 CQA REFERENCE STAR, IN EARTH EQUATORIAL: DECLINATION
C 16 CQB    "        "   "    "       "     : RIGHT ACENSION
C 17 P17(2) SPARE
C
C  items computed from epoch constants
C 19 ZFEB Planets pole in ecliptic system; longitude
C 20 ZFEC    " " " "  co-latitude
C 21 XFEXB Angle along planets equator from ecliptic to planet orbit
C 22 ARGV ANGLE FROM PLANETS ORBITAL NODE TO PLANETS VERNAL EQUINOX
C 23 SLP L-SUB-S AT PERIAPSIS
C 24 P24 SPARE
C
C 25 PHOXX(3) VECTOR FROM FOCUS (SUN) TO PLANET: ORBITAL PLANE COORDINATES
C 28 PHEXX(3) "     "     "    "    "     "    EQUTORIAL COORDINATES
C 31 HFXX(3) VECTOR FROM PLANET TO SUN IN PLANET-FIXED COORDINATES
C 34 FQ(9) ROTATION MATRIX: TO PLANET-FIXED FROM EARTH-EQUITORIAL
C 43 EO(9) ROTATION MATRIX: TO ECLIPTIC FROM ORBITAL
C 52 HO(9) ROTATION MATRIX: TO HELIOCENTRIC PLANET-FIXED, FROM ORBITAL
C --- end of the 60 real variables transfered to disk files
C 61 PI 3.14159...
C 62 DAYTIM(5) run_time in  20 ASCII bytes
C 67 IOK LOGICAL UNIT FOR INPUT (KEYBOARD)
C 68 IOS LOGICAL UNIT FOR PROMPT (SCREEN)
C 69 IOP LOGICAL UNIT FOR OUTPUT (PRINTER)
C 70 IOD LOGICAL UNIT FOR DATA FILE
C_Hist  1997jan30 and earlier  Hugh Kieffer
C  2009mar17 HK Number the positions in comments
C_End
```

IDL routines 2009may10

A number of IDL routines have been written to interface with the KRC system. The following are KRC-specific:

```
definekrc  #  Define structures in IDL that correspond for Fortran commons.
   Calls BYTEPAD
delcase    # Show delta between arrays changing only last index. Calls HISTFAST
lookrc     # Read any type 5x KRC bin5 models; look at change between cases.
   Calls READKRC5*
krccomlab  # Print KRC common input items. Calls 0
krchange   # Find changes in KRC input values in common KRCCOM.
   Calls READKRCCOM MAKEKRCVAL
krclayer   # Compute center depth of KRC layers. Calls 0
makekrcval # Make string of selected KRC inputs: Key=val. Calls DELAST0
readkrc52  # Read RKC type 52 or 51 bin5 file; post 2004jul21. Calls BIN5
readkrc54  # Read KRC type 54 or 55 bin5 file. Calls BIN5
readkrc56  # Read KRC type 56 bin5 file. Calls BIN5
readkrccom # Read a KRCCOM structure from a bin5 file. Calls DEFINEKRC
tstp2ta    # Convert T_surf and T_plan to T_atm. Calls 0
when2start # Calc starting date for KRC to reach Ls on specific season step.
   Calls 0
```

The following are utility routines called directly or indirectly by the KRC routines:

```
bin5.pro # Write/Read numeric binary files with 'standard' header. Calls 0
bytepad  # Create a Byte version of a string, padded with trailing blanks.
 Calls 0
chart    # Strip-chart plot of several variables. Calls PSYMLINE
color24bit # Generate 256 longwords to emulate nice 8-bit color table. Calls 0
delast0  # Delete trailing 0's past the decimal point. Calls 0
getp     # Modify single numeric value; with prompt and limit tests. Calls 0
getpan   # Modify any elements of numeric array, with prompt and limit tests.
 Calls 0
getpsn   # Interactive input any elements of a string array, with prompt. Calls 0
graph    # Interface to graphics devices. Calls SETCOLOR
histfast # Robust, easy histogram plot, with statistics, opt row weights.
 Calls  MEAN_STD SUBTITLE
kon91    # Common minimal functionality in the kon case statement.
 Calls GETPINTS GRAPH  MAKE99 SETCOLOR TV2JPG TV2LP
label_curve # Place an oriented label on a curve . Calls RNDEX RTERP1
locate   # Find lower index of interval in ordered vector containing x. Calls 0
tes      # Convert Martian season L_s <-> Julian day. Calls 0
make99   # Make/print list of user options for a program. Calls 0
mean_std # Mean and standard deviation of a vector. Calls 0
psymline # Hughs convention for transfering PSYM and LINESTYLE in one. Calls 0
rndex    # Finds floating-point index of within a monotonic array. Calls LOCATE
rterp1   # real interpolation in a vector. Calls 0
setcolor # Set or modify colors, lines, plot-symbols, #plots/page.
 Calls COLOR24BIT  GETP  GETPAN  TOOTHB  ST0
st0      # Make minimal string for numbers, or string arrays. Calls DELAST0
strword1 # Extract first word from a string or strarr. Calls 0
toothb   # Add a toothed color scale-bar to a window or TVPLEX panel.
 Calls GETPAN
```

The IDL program lookrc contains code to read all type 5x files and compare multiple cases. ALthough there is a lot of speciality code, all functions are isolated in elements of a large case statement. One could extract parts of the

```
code to start your own routines.

.rnew lookrc
@ 11, edit file path; the default extension will be appended for each type
@ 5x, where x is 1,2,4,5,6  to read the file
@ 40, will list a guide to what was read

=============================================================================
Notes on how some aspects of the code work:

>> New file name:
TCARD reads a card of Type 8, (and index is not 22 or 23)
 it calls  TDISK(4,0), which closes current file and sets  LOPN2=.FALSE.
   TCARD then moves new file name into common
KRC checks if current (new) values of N5 and JDISK call for file output;
  with  LOPN2=.FALSE., KRC calls CALL TDISK (1,0) to open new file.

>> End of a case and end of a run:
TCARD sets KOUNT=0 at entry; this is incremented for every card except those of
type 0 ( or less) or type 11 (one-point mode). When type 0 is encountered, if
KOUNT is positive, does normal check of changes before return with IR=1 to
indicate start of a new case; if KOUNT is zero, returns with IR=5 and prints
'END OF DATA ON INPUT UNIT'

>> Setting one-point mode.
This can be done only in the first case, and there is no way to leave the
one-point mode except to end the run.

TCARD encounters: " 10 * filename" as change card in the initial case.
  sets this as new input file name, then returns with IRET=4
 [Thus, nothing following this change card in initial file is read]
KRC closes prior input file, opens the new one, and reads past first two lines
    then calls TCARD to read first one-point line and sets LONE=true
    and drops into the top of the "case" loop.
The master one-point should have a single latitude, no binary output file.
The small number of layers, days to converge, and seasons ignores the seasonal
effect.

One-point request values are read by TCARD @ 310, which computes starting DJUL


>> Starting conditions and date

Initial N5-JDISK sets the size of output files. There could be any number of
interior seasons where parameter changes are made; based on successive values of
IDOWN.

KRC initially calls TCARD(1
For each case loop, sets IQ=TCARD_return. If one-point mode, sets IQ=1

TSEAS uses IQ as key. It this is 1, then sets J5=0 and sets DJU5 to season -1.
else, increments J5 and increments DJU5 with current DELDUL. This allows use of
variable resolution dates. (so J5 never 0 when TCARD(2 called)
If J5 equals IDISK2, then TSEAS calls TCARD(2 to read changes, and proceeds to
next season.
```

TLATS uses J5 as the key; if it is <= 1, then starts from equilibrium conditions, else uses predictions from prior season

The default is that change cards cause a fresh calculation of starting conditions. Exceptions are when J5=IDOWN>0 at TCARD entry

>> Use of common PORBCM
Contents are described in porbcm.inc,
PORBCM is filled by TCARD calling PORB0, which reads the first 60 items in 5G15.7 from the input file and sets the value of PI. KRC references porbcm.inc but does not use it. TSEAS uses a few items to calculate LsubS. TYEAR uses the value for length of year.

F Evolution log of KRC

This log of modifications may not be complete.

```
evolve.txt  Notes on evolution of KRC code, after the first 15 years.

85May~10-14.    Dave Paige visits Flagstaff. We create new directory
[hkieffer.krc.mars] in which MARS version of code is put. MARS version
has larger LATCOM (JLAT changed to real*4).  HELPLIST revised.
  After Dave left, found that TDISK had not had JLAT change,
hence was not writing LATCOM to disk.
Made new plot version, starting with 84jun comet version, but with
almost entire revision using NCAR_1 routines, including new
MCURVE1, CONREC1 and GO1.  Never got all the bugs out.

85Jun24-28  Paul Weissman visit. Found error in COMA2, otherwise no changes
from Jun 84 comet version.  Linked and ran and duplicated older runs.

85sep05-07 Combine the comet and mars versions into single routines which use
the larger LATCOM. Only external change is reversing the meaning of
LD18.  Major restructuring of TLATS and TDAY to accommodate both comet
and Mars; use LD20 .TRUE. if Mars, .FALSE. if comet.  Other routines
needed no changes.
Revise HELPLIST.
Create directory KRC.COM] for the comet-particular stuff. Move older
routines for .KRC] to .KRC.COM]. Move all the routines which support
both comet and mars from .KRC.MARS] to .KRC].
Delete plot routines dating from 1984 which used the smaller LATCOM.

85oct14 Add COMMON FILCOM of file names; print these in TPRINT.
Change meaning of FROST4 and AFRO4 for comet.
Minor changes in printout sequence.
Change TYPE to WRITE(IOPM  in TDISK.

86oct Paul finds erroneous factor of PI in computing coma diffuse radiance;
change made to source code only, not linked.

87mar29 Remove incorporation of albedo in the solar incident flux ASOL.
Add ADGDIF (diffuse solar flux) to KRCCOM.INC.
Recompile TLATS,TDAY, Link.

87jun30 TDAY: Avoid /0 if DTMJ(JJJ)=0 at "done" test.

87sep11 .MARS] TDAY & TLATS: Special versions for metamorphism.
  Use ZLAT(17:19) for input of metamorphism and sublimation constants.
  Use TT(J,MAXN3) to transfer metamorphism rate.
  TYEAR: version of TLATS which averages daily insolation and includes
    PORB in the insolation calculation. Uses AVEDAY.

87oct01 ALL Separate the use of NMHA for storage and N24 for printout.
TLATS: replace CFSOLAR with AVEDAY.

87nov22  .YEAR] versions of DAYCOM and LATCOM with larger MAXN24 and MAXN2
meant for use with TYEAR.
Most routines compiled into .YEAR] with the these.
Will need to redo TDISK if it is to be used.
```

```
87nov22 TCARD: Add report if input integers are reset into valid range.
  NMHA no longer constrained.
TPRINT: redo some formats.
TDAY: Add error report if convergence is unstable.
KRC: set IOERR=IOPM rather than IOSP.
        Force parameter print through call to TPRINT(2) if TDAY(1) error occurs.
.MARS]TLATS:  now includes variance tests.

88sep08 .MARS]TDAY: Test moving layer T limit tests and metamorph from N24
into each time loop; so that they are done 1536 instead of 40 times
per "day". Negligible effect; <.01 degree in TMETA.

97fall-98summer Incorporate one-layer atmosphere with many parameters that can
be tuned to mimic Haberle-Jakosky model. Wrote LaTeX description. Build TES
look-up code for computation of thermal inertia from TES observations; this
interfaces with Mike Mellon model set.

1997sep  idlkrc.f  Build this IDL interface to call KRC.

98sep01-07 Add section to TDISK for output of bin5 files type 51 and 52. Minor
code cleanup, avoiding divide-by-zero if atmosphere parameters were zero Make
KRC/moon version of TLATS and TDAY by removing all atmosphere code, and
including eclipse section in TLATS; commons left the same even though
atmosphere results not calculated.

1999dec krc.f tcard.f Add option to continue from current condition

2002mar07 alsubs.f Created. Adopted from l_sub_s.pro

2002mar07 krc.f tcard.f Major change. Add option for "one-point" rapid runs for
Surface T

2002jul Major change. Incorporate Delta-Eddington atmosphere. Found that double
precision required within deding2.f

2002aug04 tdisk.f Add output file type 53=(combo at 1 lat, 2+80 seasons, 10
cases). Recode logic

2002nov01 tseas.f  Have DJU5 increment by current DELJUL for each season

2004jul06 tdisk  Add file style 54  Add the Common  HATCOM

2004sep28 porb.f Change name of called routine ROTATE to ROTVEC to avoid library
conflicts

2004sep28 tlats.f Add tests to avoid round-off to negative fluxes at night so
code would run at ASU

2004sep30-Oct5 tday.f tdisk.f Add storage of surface downward fluxes every hour
on last day. Revise file style 52 to include them (and a spare variable).

2005nov18 tlats  Add optional solar zenith angle limit

2005nov19 tprint Add print of depth to top of 2nd layer
```

```
2005dec28 tlats  Fix bug using ZENLIM. Additional comments

2006jan25 tlats  Modify SKYFAC from linear with slope to (1+cos s)/2

2006apr12 tdisk  Change file style 54 to have both 1am and 1pm surface Temp.

2006apr22 tdisk  Allow flexible number of cases for output file type 52 and 54

2006apr30 tdisk  Add TTB4 to type 54 output

2006sep09 tcard  Correct error:  REAL*4 LSUBS should have been ALSUBS

2006sep09  Allow seasonally variable albedo and TauD.
  three new routines: seasalb, seastau, readtxt360;
  and changes in tseas and tcard and filcom.inc

2008oct02-25
Found error in calculation of planetary temperature; was using
  hemispheric integral 1-BETA instead of exp(-Tau_IR); so in effect using tau
  that was too large by factor of tau_eff/ Tau_IR.
Modify: krc.f tseas.f tlats.f tday.f tdisk.f tprint.f tcard.f
 krccom.inc daycom.inc
Use slope azimuth as a flag for a pit of slope SLOPE.
Put in proper SKYFAC for conical pit.
Allow for Snow formation in cold atmosphere and fall to surface.
Move MAXBOT from daycom.inc to krccom.inc.
Replace ID22(1) and (2) with KVALB and KVTAU.
Move AMW fixed value in tlats.f into krccom in place of ABRAMP
  move other physical and hardware-dependent constants into krc and krccom.
Add output file type 56 in tdisk.
Replace dual use of GGT by using new DTMAX for daily convergence.
Major modification of output Type 52.
Use IDOWN as season index at which to read some changes
TMN4Y(
2008nov11-2009feb
Add temperature-dependant conduction option
  Use L10 as logical variable for k-of-T
  Lengthen KRCCOM and move some inputs around
  recode tday to have constant and T-dependant options
 Update IDL routines that deal with KRCCOM
Remove max # seasons.

2009apr22
Add ability to forecast  deep layers based on storage of midnight values.

Scheme 1. [Coded, but not refined. Mostly a stub] Store at integral years prior
to forecast season index. If have 3 or more values, use EPREAD; should be safe
to use on all layers. Because of possible jump perturbation on the first day,
best to not have that be one of the stored dates. Does nothing if IDISK2=0.  [To
remove, delete TMN4Y from hatcom, delete call to TYEARP in TSEAS, remove
tyearp.o from sources in Makefile]

Scheme 2.[Not implimented] Rolling storage of all seasons for the past year. At
forecast season; evaluate thermal delay to surface and make [complicated]
forecast.
```

G Code details

deding2.f has inputs of:

`OMEGA` dust single scattering albedo
`G0` dust asymmetry parameter
`ASUR` surface albedo
`COSI` cosine of incidence angle
`TAU` dust vertical opacity

and outputs:

`BOND` Planetary (atm plus surface system) albedo
`COLL` Direct beam at bottom = collimated + aureole
`RI[2,2]` Diffuse irradiances:

[1,= I_0 = isotropic   [2, = I_1 = asymmetric
,1]= at top of atmosphere   ,2] = at bottom of atm