A TPM for non-convex shapes

In this chapter, generalizations of our TPM (see chapter 3) are discussed to account for the effects of shape non-convexity beyond that of cratering. On globally non-convex shapes, facets may shadow one another and additionally exchange energy through direct and indirect radiation.

The work reported in this chapter is in progress. A first model variant has been developed in which shadowing is taken into account but mutual heating is not; detailed tests are on-going. This model variant aims primarily at simulations of eclipsing binary systems and has been used to analyze Spitzer observations of the binary system (617) Patroclus during two mutual events (see sect. 6.8).

A more general model variant, in which also mutual heating of facets is taken into account, has been partially designed but not yet implemented.

A.1 Physical background

Most available asteroid shape models are intrinsically convex such that the "convex-shape" TPM is applicable (see chapter 3). Several important asteroids, however, have well determined non-convex shapes, e.g. the spacecraft target NEAs (433) Eros and (25143) Itokawa (see also Fig. 1.1 on p. 2). Also, doubly tidally locked binary asteroid systems can be thermally modeled as a single, non-convex object (see sect. A.1.2).

Non-convexity of a body's shape adds to the complexity of modeling its thermal emission because facets may eclipse and/or occult one another (i.e. obstruct the line of sight towards the Sun and/or the observer, respectively), and furthermore they can directly and indirectly exchange energy by scattering and reabsorption of optical and thermal radiation, leading to mutual heating.

Mutual heating couples the temperatures on different facets and thus significantly increases the difficulty of determining surface temperatures. Shadowing is easier to model, requiring only the directions towards the Sun and the observer to be checked for possible obstructions before fluxes are calculated.

We have designed, implemented, and partially tested a TPM variant for non-convex bodies where shadowing effects are considered and mutual heating is neglected. We have partially designed, but not yet implemented, a variant in which mutual heating is fully taken into account. Mutual heating may be negligible if the global surface concavities are shallow such that only moderate fractions of the solid angle above the facets' local horizons are covered by other facets (see also Lagerros, 1997). Primarily, however, the model described herein is designed to model the thermal effects of eclipses in binary asteroid systems. The binary system is described as a single rigid body with a shape consisting of two disjoint parts. For eclipsing binaries, we expect the effect of mutual heating to be negligible relative to that of shadowing.

In the following, both model variants shall be described. As in the case of the convex model, geometric and thermal aspects are separated as far as possible. An auxiliary software was developed to convert asteroid shape models into a TPM-specific `.concave` format containing all geometric information required for thermal modeling.

A.1.1 Geometric aspects

While on a convex body all facet-specific geometric information required to determine the thermal flux emanating from it is its outbound surface-normal vector, more geometric information is required to determine whether or not a facet is obstructed by another facet, and furthermore to calculate the mutual heating among them. To model obstructions, it is required to know for each facet which other facets are visible from it and what their relative position is. To model mutual heating, it is required to know the *view factors* of all visible facets, a measure of the solid angle under which facets are visible to one another (view factors have been discussed in the context of hemispherical craters, see eqn. 3.23 on p. 64).

The surface temperature on a given facet is coupled to the surface temperatures of all facets in its *thermal unit*, which we define as the set of all facets with which it can exchange energy through direct or indirect radiation. Each asteroid shape model can be uniquely decomposed into disjoint thermal units. If lateral heat conduction can be neglected, temperatures inside one thermal unit are independent of temperatures inside any other thermal unit. For a convex object, each thermal unit consists of a single surface element, whereas a concavity in an otherwise convex object forms a contiguous thermal unit. In general, not all facets belonging to a thermal unit are directly visible to one another; e.g. a pyramid on a large sphere belongs to a thermal unit containing all points visible from the peak of the pyramid, although points on one side of the pyramid do not see points on the other side.

In an auxiliary program, asteroid shape models are read in (the same input formats are supported as for the convex model), all required geometric information is calculated and stored in `.concave` files, which can be read in by the non-convex TPM code. To generate a `.concave` file, outbound surface-normal vectors and the intrinsic diameter are determined like for a convex object (see sect. 3.2.1.a and 3.2.1.b).[1] Then, all facets are checked whether they are visible from one another. To be visible from one another

1. facets must be tilted towards one another, i.e. each must be above the other facet's local horizon. This is determined from the sign of the scalar product of the connecting vector and the outbound normal vectors.

2. additionally, the line of sight between the two facets must not be obstructed by another facet. Only facets above either horizon can do that.

To determine whether a facet a obstructs a line from facet b, one determines the intersection of that line with the plane containing facet a.[2] The vector of the intersection point can be uniquely decomposed as

$$\vec{v_1} + x \cdot (\vec{v_2} - \vec{v_1}) + y \cdot (\vec{v_3} - \vec{v_1}) \tag{A.1}$$

($\vec{v_i}$ denote the vertices of facet a), facet b is obstructed by facet a if $x + y \leq 1$. This routine is also used in the proper TPM code to determine whether facets are shadowed.

Thermal units are built up recursively by adding a facet which has not yet been associated with a thermal unit. The routine to add facets cycles through the list of facets visible from the added facet, determines whether the visible facet is already contained in the thermal unit and adds it otherwise (recursion).

[1] Note that it is assumed in our algorithms that the origin of the coordinate system is connected to all vertices by straight lines which lie entirely within the object (see footnote 1 on 51). Judging from visual inspection of computer renderings, this appears to be the case for all shape models considered in this thesis.

[2] We here only consider lines originating at the midpoint of facet b. This avoids model complications due to finite facet size, such as partially obstructed facets. If the line of sight between two facets is partially obstructed by a third facet, our approach risks defining one facet to be "visible" from the other, but not vice versa. To avoid this, visibility is only checked in one direction and the result is applied in both directions.

A.1.2 Doubly tidally locked binaries as special non-convex bodies

For the non-convex TPM code, it is not necessary to assume that the asteroid shape be contiguous. A rigid body consisting of two disjoint pieces is equivalent to a doubly tidally locked binary system which is at rest in a co-rotating coordinate system (i.e. the two spin periods are synchronized with the period of the circular mutual orbit, and both spin axes are perpendicular on the latter). While tidal friction is known to attract all binary systems towards this asymptotic attractor state, many binary systems are not doubly tidally locked; this variant of the TPM is not applicable to the latter. It was primarily designed for planning and analyzing our Spitzer observations of (617) Patroclus, which appears to be doubly tidally locked (see sect. 6.8).

It is straightforward to model the observable effects of eclipse and occultation events using a non-convex TPM code in which shadowing is accounted for. Both components are assumed to be convex. It is plausible that mutual heating from component to component can be neglected because their distance is typically much larger than either component's radius.

An auxiliary program has been developed to generate `.concave` files describing binary systems. To this end, two arbitrary `OBJ` shape files are read in, typically triangulated spheres. Both can be stretched (e.g. to obtain ellipsoids), one can be rescaled to obtain different-size components. After determining the outbound surface-normal vectors on each component, they are placed at a user-specified mutual distance. Both components are shifted by a distance proportional to the other component's volume, such that the z axis (i.e. the spin axis of the mutual orbit in the body-fixed system) runs through the center of mass. It is tacitly assumed that the center of mass of either component coincides with the origin of the respective shape model's coordinate system, and that the mass density is homogeneous throughout the system.

All geometric quantities, such as lists of shadowers, are determined like they would be determined for other shape files. If both components are convex, the resulting binary shape has one large thermal unit containing the two hemispheres facing one another, while each facet on the two opposite hemispheres forms its own thermal unit.

The intrinsic volume-equivalent diameter (which is stored in the `.concave` file) is determined from the sum of the individual components' volumes, where stretching and rescaling factors are taken into account. Note that in this special case the algorithm discussed in sect. 3.2.1.b to calculate the volume of a shape model would fail.

For a binary system, the two usual diameter definitions (area-equivalent or volume-equivalent; see sect. 2.2.1) generally lead to significantly different diameter values. For a system composed of two spheres with diameters D_1 and D_2, the total area-equivalent diameter $D_\mathcal{A}$ equals $\sqrt{D_1{}^2 + D_2{}^2}$, whereas the total volume-equivalent diameter D_V equals $\sqrt[3]{D_1{}^3 + D_2{}^3}$.

It must be noted that our TPM uses $D_\mathcal{A}$ to relate the H value with p_V (because H is based on area-dependent optical magnitudes), while D_V is used to scale model fluxes. Therefore, for given H and p_V values, model fluxes must be multiplied with

$$\frac{f_{\text{“true”}}}{f_\text{model}} = \left(\frac{D_V}{D_\mathcal{A}}\right)^2 = \frac{\left(D_1{}^3 + D_2{}^3\right)^{2/3}}{D_1{}^2 + D_2{}^2}. \tag{A.2}$$

Alternatively, the TPM code outputs “true” flux values if it is given an H value which is offset by $2.5\log_{10}(f_{\text{“true”}}/f_\text{model})$ from the H value quoted in the literature.

A.1.3 Temperature distribution without mutual heating

If mutual heating can be neglected, then the temperatures on individual facets are independent from one another. We can continue to use eqn. 3.11b on p. 55 to determine surface temperatures on smooth facets, with the sole change that μ_S is redefined to vanish when the facet is eclipsed. In the absence of thermal inertia, in particular, $T = \sqrt[1/4]{\mu_S}\ T_{SS}$. Analogously, μ_O is defined to vanish when the facet is occulted such that observable flux contributions from a facet continue to be proportional to μ_O.

A.1.4 Temperature distribution with mutual heating

The effect of mutual heating on smooth facets inside a thermal unit can in principle be determined along the lines discussed in sect. 3.2.3 on p. 60 for facets inside hemispherical craters. However, since view factors are no longer constant, no general analytic solutions can be found.

The first step towards a solution is to determine the radiation field at optical wavelengths, J_V, for all time steps by numerically solving eqn. 3.27 (with suitably defined $m_S(\vec{r})$ to take shadowing into account). Since asteroid Bond albedos are typically small, the perturbative approach eqn. 3.32 would be expected to converge rapidly. The integral therein is replaced by a sum over all triangular facets inside the thermal unit.

The temperature distribution then follows from eqn. 3.33 combined with eqn. 3.36, which must be solved together with the heat diffusion equation eqn. 3.11 below each facet. A TPM with mutual heating has not yet been implemented, we plan to tackle the temperature distribution using the following algorithm: After a suitable initialization of the temperature profile below each facet,[3] the system is propagated one time step further by iterating the following steps for each facet:

1. calculate the sub-surface temperature profile. If the heat diffusion is discretized in a fully explicit way, sub-surface temperatures for the new time step are fully determined by the old temperatures.

2. determine the amount of absorbed thermal flux emanating from other facets. Since the current surface temperatures are not yet known, approximate them with values from the previous time step. Recursively add higher scattering orders until convergence is reached to within a user-specified accuracy goal. Alternatively, cut off after a certain number of scattering orders (for $\epsilon = 0.9$, the reflectivity equals $1 - \epsilon = 0.1$, so the third scattering order may already be insignificant).

3. from this, combined with the pre-computed optical flux at the current time step and the current sub-surface temperature, calculate the current surface temperature from a discretization of the surface boundary condition (eqn. 3.36).

This requires that the temperature profiles of all facets must be calculated (and therefore kept in RAM) simultaneously. It remains to be studied at the code validation phase whether or not the approximation made in step 2 introduces significant systematic errors. A suitable test method might be to compare the mutual heating terms determined in step 2 with mutual heating terms which would result from the resulting surface temperatures.

After temperatures have been determined, observable fluxes remain to be calculated by summing up the Planck contributions from the individual facets taking account of obstructions of the observer's line of sight and multiple scattering. Due to the smallness of the infrared reflectivity ($1 - \epsilon \sim 0.1$), it is probably sufficient to truncate after the single-scattering order (see also Fig. 3.3 on p. 72).

[3] It should be verified at the code validation stage that model fluxes are independent of the particular initialization chosen; a good first estimate may lead to significant improvements in convergence speed.

A.1.5 Beaming

Without mutual heating If mutual heating by other facets is neglected, the temperature distribution inside a hemispherical crater can in principle be determined in a way which is analogous to that used for the convex TPM (see sect. 3.2.3), provided that $m_S(\vec{r})$ is set to zero when the crater is eclipsed by another facet.

If thermal inertia is neglected, an eclipse cools the crater down to 0 K instantaneously. An explicit numerical model of thermal conduction inside craters would be relatively easy to generalize such that it allows for the effect of eclipsing by other facets. Our approximative treatment of thermal conduction inside craters on convex bodies (see sect. 3.2.3.f), however, is not readily generalized to allow for eclipses.

Instead, we have reverted to an approximation similar to one proposed by Spencer et al. (1989); Spencer (1990), where beaming is crudely taken into account by multiplying surface temperatures with a globally constant factor $\eta^{-1/4}$. This is analogous to the NEATM η. Note, however, that here η solely reflects the effects of thermal-infrared beaming, whereas in the NEATM η reflects the *combined* effect of beaming and thermal inertia. Here, surface temperatures are first calculated for smooth facets, taking thermal conduction and shadowing fully into account. We expect this approximation to be appropriate for small phase angles and not too large values of thermal inertia, hence it should be feasible to model thermal-infrared observations of a Trojan binary such as (617) Patroclus, which is observed at small phase angles and is expected to be covered with regolith.

With mutual heating When hemispherical craters are added to the smooth facets within a thermal unit, the list of integral equations to be solved grows further. Inside a crater, the temperature distribution becomes more complex since there are additional sources of incoming flux to be considered. Additionally, the facet's thermal emission gains a different directional characteristics, influencing the mutual heating of facets. The number of integral equations which in principle all have to be solved simultaneously quickly becomes prohibitively large. A suitable approximation appears to be in order. To the best of our knowledge, no detailed modeling of this problem is available in the literature, Lagerros (1997) proposed a model in which beaming is only applied to the direct component.

It seems difficult to make use of the high symmetry of hemispherical craters in the presence of mutual heating by other facets (and their craters). It may be advantageous to give up the distinction between global shape and local craters, but rather to model roughness by modifying the shape model itself, i.e. by replacing planar facets with suitable rough facets (which are themselves triangulated). This opens a conceptually easy road to much more general thermal models of asteroid surface roughness allowing, in particular, for positive surface relief such as boulders. See also the discussion in sect. 7.3.4.c on p. 234.

The implementation of such roughness models is beyond our current scope. It is clear, however, that a TPM code suitable for this task must be optimized for numerical performance, chiefly because the number of facets contained in thermal units grows vastly if small-scale roughness is added, requiring relatively large amounts of RAM and CPU time (see below).

A.2 Implementation

Our implementation of the non-convex TPM is based on that of the convex TPM discussed in sect. 3.3. While the current version, which neglects mutual heating, is a helpful tool in itself, great care was taken to implement it in a way which makes it easy to add mutual heating at a later stage.

The class structure of the convex model is largely preserved. The abstract base classes `asteroid` and `ThermalModel` are reused without any modification, the definition of `TriangulatedConcave` differs marginally from that of `TriangulatedConvex`. An important new class is `ThermalUnit`, containing a list of pointers to `FacetConcave` objects. The latter contain all required precomputed geometric information for one facet, such as a list of visible facets and view factors. The routine `fluxModFactors` (defined within the scope of `ThermalModelConcave`) is passed a reference to a `ThermalUnit` object and determines the observable flux contribution from that unit. Flux contributions from all thermal units are summed up and converted into physical units by the calling routine within the scope of `TriangulatedConcave`.

`ThermalInertiaOnlyConcave` objects calculate temperatures and observable fluxes emanating from all facets within the thermal unit, taking shadowing effects (both towards the Sun and the observer) into account. To solve the heat diffusion equation, the same algorithm is used as in the convex model. As in the convex model, a vector containing μ_S for each time step is calculated before the diffusion equation is solved; in doing so, shadowing is accounted for. The algorithm to determine whether a facet is shadowed is identical to that described in sect. A.1.1 (eqn. A.1 on p. 247). Typically, identical discretization steps can be chosen as in the convex-model case. To model eclipse events on binary systems, it is important to consider the duration of eclipse events relative to the rotation period, since it may require a finer time resolution than is usual (see sect. 6.8.4).

Anticipating that in a more general TPM surface temperatures will be coupled through mutual heating, the code is designed in a way that keeps the thermal profiles of all facets belonging to a thermal unit simultaneously in the computer's RAM for all time steps. The number N of `double` variables required is

$$N = \texttt{nTime * nZ * facets.size()}, \tag{A.3}$$

`facets.size()` denotes the number of facets contained in the thermal unit. Typical values for a binary during eclipse are `facets.size()` ~ 2500, `nTime` $= 1000$, and `nZ` $= 30$, totaling to 75 million `double` variables requiring 600 MB of RAM. This has a critical impact on the numerical efficiency of the program unless it runs on a machine with significantly more RAM available (typically used machines, however, had only 512 MB of RAM).

We note that knowledge of the temperature profile at *all* time steps is not required for the primary purposes of our model, although it proved helpful in the validation phase. Consequently, the RAM demand could be cut down by a factor of `nTime`/2 if only two time steps would be stored simultaneously.[4] This algorithm has been adapted from the convex TPM, where only a single facet is handled at any time and therefore RAM is not a matter of concern.

As discussed in sect. A.1.5, we account for thermal-infrared beaming by multiplying temperatures with a globally constant factor $\eta^{-1/4}$. This is implemented in the class `ThermalInertiaEta` which inherits from `ThermalInertiaOnlyConcave`. All temperatures are rescaled by overwriting the routine `TSS` (defined within the scope of `ThermalModel`): the value returned by the base class is multiplied by $\eta^{-1/4}$ which is a constant within the scope of `ThermalInertiaEta`.

Functions on the main-routine level are adapted from the convex model, which determine χ^2 and the best-fit p_V for a given data set and scan the parameter space spanned by thermal inertia and η. Additionally, time-resolved model spectra can be output for any combination of thermal parameters.

[4] In addition to a vector containing surface temperatures for all time steps, which is currently not needed because surface temperatures are stored as zeroth element of the profile.

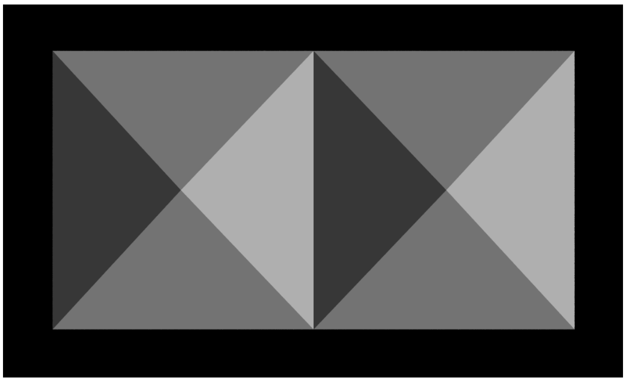

Figure A.1: One of the model figures used to verify that visible and invisible facets are correctly identified. It consists of two adjacent pyramids. If viewed from above, two facets see one another, the remaining six are isolated. Viewed from below, each inverse pyramid forms a thermal unit in which each facet sees the remaining three.

A.3 Validation

The model variant discussed in this chapter is still under development. Various validation test have been successfully performed, however.

Geometry The tool `OBJ2concave` to generate `.concave` files has been tested intensively to verify that it correctly distinguishes visible from invisible facets. To this end, model figures such as that depicted in Fig. A.1 were used or more complicated variants thereof.

Furthermore, the tool `binary` which generates binary `.concave` files has been checked by manual inspection of simple output files (e.g. a model binary consisting of two tetrahedra).

Thermal physics The concave TPM was verified to reproduce convex-model fluxes for spherical shapes. For a non-eclipsing binary viewed and irradiated pole on, the model flux equals the sum of the two individual spherical components, provided the diameter conversion (eqn. A.2 on p. 249) is applied; the validity of the latter has been numerically verified for a large range of diameter ratios.

The treatment of eclipses and occultations was qualitatively validated by calculating model fluxes for a synthetic binary system consisting of two equally sized spheres with the spin axis perpendicular on the viewing plane, leading to total eclipse and occultation events. There are two eclipses and occultations per mutual orbit, when respectively one component obstructs the other's line of sight towards the Sun or the observer, respectively. Without thermal inertia, eclipses have the same observable effect as occultations, both leading to a peak flux drop of 50 %. For increasing values of thermal inertia, the relative depth of the eclipse event decreases (shadowed regions gradually cool down but their thermal emission is not "switched off" immediately) while occultation events, which are purely geometric in nature, remain unchanged. Finally, for very large thermal inertia values, eclipse events become virtually invisible in the thermal infrared.

There appear to be flaws in the model code, however. Assuming the radar-derived shape model of (54509) YORP by Taylor et al. (2007), which contains major concavities, and the observing geometry of our Spitzer observations of this object (see sect. 6.4), synthetic model fluxes output by the non-convex TPM are ~ 2 times larger than convex-TPM fluxes. This is inconsistent with the expectation that shadowing effects should *reduce* model fluxes rather than *increase* them. While the convex-shape TPM is well tested, the non-convex TPM is not; additionally, "convex" fluxes are more consistent with expectations based on the NEATM. It is not clear at present what causes this discrepancy. Further testing of the non-convex TPM is required.