**MAPPING PLANETARY DATA BETWEEN DIFFERENT SHAPE MODELS OF AN IRREGULAR BODY.** B. Grieger, ESAC, Camino Bajo del Castillo s/n, Urb. Villafranca del Castillo, E-28692 Villanueva de la Cañada, Madrid, Spain (bgrieger@sciops.esa.int).

**Introduction:** Standard global map projections cannot display the complete surface of a highly irregular body such as the Rosetta target comet 67P/Churyumov-Gerasimenko (67P for short) because different points on the surface can have the same longitude and latitude. The Quincuncial Adaptive Closed Kohonen (QuACK) map [1], however, allows to display the complete comet. The map projection is defined by a special shape model (the QuACK shape model), see Fig. 1 that can be unfolded into a rectangular grid, see Fig. 2. In order to display data in the map, it has to be defined on this shape model.

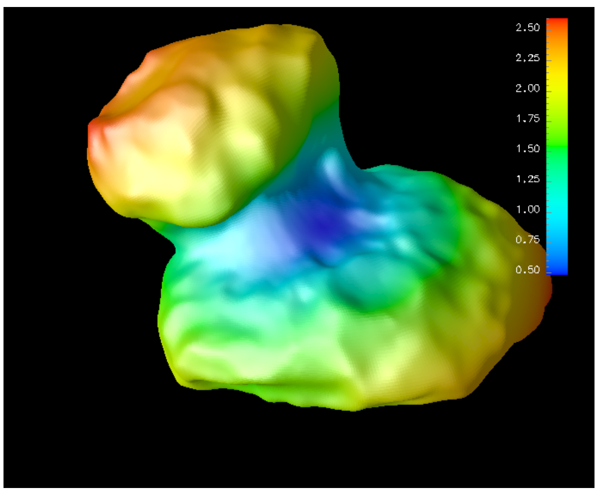


Figure 1: QuACK shape model. It consists of $401 \times 201$ grid points. The color encodes (just as example data) the distance of the surface point from the comet center.

Ideally, a data producer would project his data — e. g., imagery — directly onto the QuACK shape model. However, the QuACK shape model does not fit all purposes, because it is — for computational but also conceptual reasons — of relative low resolution, i. e., 80 000 plates, while state of the art shape models of 67P can have 12 million plates. Some Rosetta instrument teams have provided their data to ESA's Planetary Science Archive (PSA) with per pixel geometry information that stems from projection the data on such a high resolution shape model.

To enable PSA tools to display the data on the QuACK map, either the instrument teams need to redo their projection with the QuACK shape model or we need means to map data defined on the surface of one shape model to the surface of another shape model. Herein, we present such a mapping.

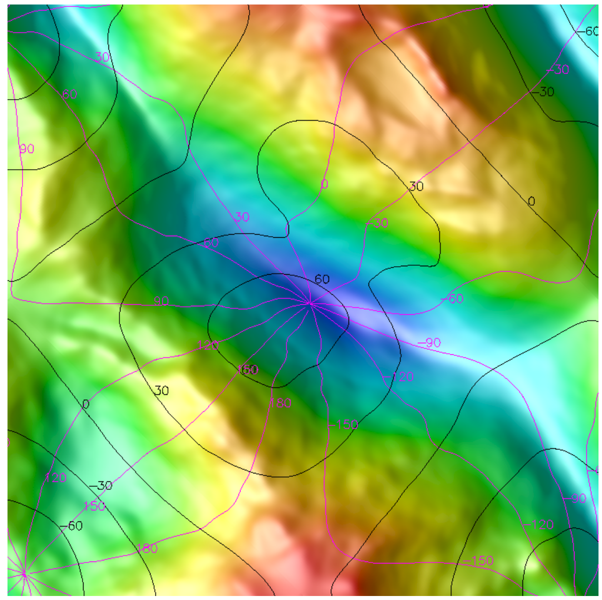

Figure 2: Unfolded QuACK map in quincuncial layout. The $401 \times 201$ data points are the same as in Fig. 1. Latitudes are depicted in black and longitudes in magenta. The low (bluish) diagonally running area is the neck, upper right is the head, and lower left is the body of the duck-like shape of 67P.

**Definition of the nearest surface point:** The definition of the surface point to which we want to map a non-surface point is per se straight forward: A non-surface point shall be mapped to the nearest point on the surface. However, there are a few subtleties involved.

The ambiguities in relating a non-surface point to the surface are illustrated in Fig. 3. We consider only a one-dimensional example, thus the shape is defined by a traverse, while a real shape model is defined by plates. Even in this simple example, we can see that the apparently straight forward definition "Pick the nearest surface point" can lead to different non-surface points being mapped to the same surface point and multiple solutions for the nearest surface point to a given non-surface point. The former is caused by the non-differentiable approximation of the surface, but the latter could also occur for a smooth surface if the distance of the non-surface point is larger than the radius of curvature.

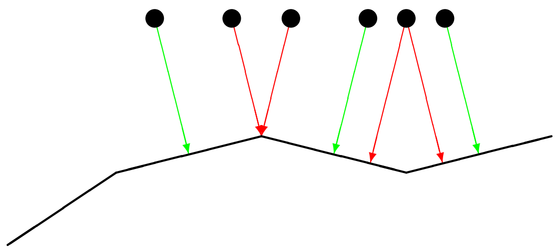

Figure 3: Illustration of the ambiguities in relating non-surface points to the surface.

When mapping high resolution data like images onto the shape, we would like to avoid distortions that could, as illustrated in Fig. 3, even lead to the omission of data. Therefore, we precompute the normals at the vertices of the shape model (as mean of the normals of adjacent plates) and define the nearest surface point as the one at which the local normal points exactly to the non-surface point, where the local normal is computed by bilinear interpolation from the (precomputed) normal at the corners of the plate. This defines the nearest surface point in a robust and unique way as long as the non-surface point is not too far from the surface.

**Efficiently finding the nearest surface point:** Apart from an appropriate definition of the optimal surface point, we need means to find it. It is computational prohibitive to search through all plates of a shape model for each non-surface point. Shape models may have millions of plates, and even the relatively low resolution QuACK shape model has 80 000. The non-surface points we want to map could even count many millions, thus the nested loops would have to check trillions of distances.

If we assume that we have already an approximate surface point for a non-surface point to map to, we can find the optimal solution by a relatively simple iterative procedure. We compute the local normal at the surface point (by bilinear interpolation between the corners of its plate) and consider a ray from the non-surface point in the inverse direction of the normal. The intersection of the ray with the surface is the updated surface point. If the surface is not too irregular (on the scale of the distance between the initial approximate surface point and the optimal surface point), repeated application of this procedure — possibly involving a damping factor — should quickly converge to the optimal point.

How can we get the initial approximate surface point? We precompute on a regular three-dimensional grid initial ray directions. For any non-surface point, we can then obtain an initial ray direction by trilinear interpolation in the grid cell where the point resides. Following this ray direction from the non-surface point gives the first approximate surface point. The initial ray directions at the grid points are computed as weighted sums of all inverted plate normals, each weighted with a negative power of the distance between grid point and plate center. We could also use the normals at the vertices (which we need to precompute anyway) instead of plate centers. The precomputation of the initial ray directions can be very computation time consuming, but it needs only to be done once.

**Conclusions:** The mapping of planetary data between different shape models can be useful for several purposes. We are here mainly aiming at data which has been provided to the PSA by the Rosetta instrument teams projected on a high resolution shape model. This data has to be mapped to the QuACK shape model in order to display it in a complete global map of the comet 67P. A subroutine implementing the described mapping of a non-surface point to the surface of the shape model will be added to other subroutines facilitating the QuACK map projection which are already available at GitHub:

`https://github.com/esaSPICEservice/QuACK`

**References:** Grieger, B. (2019) *A&A*, accepted.