## Table of Contents



# Calculating topometrics for the Baspa Valley, NW Himalaya

```
%Download most recent version of topotools from github: https://
github.com/wschwanghart/topotoolbox
%on Linux systems, you can do:
%git clone https://github.com/wschwanghart/topotoolbox

addpath(genpath('/home/bodo/topotoolbox'))
rmpath(genpath('/home/bodo/topotoolbox\.git'));
```

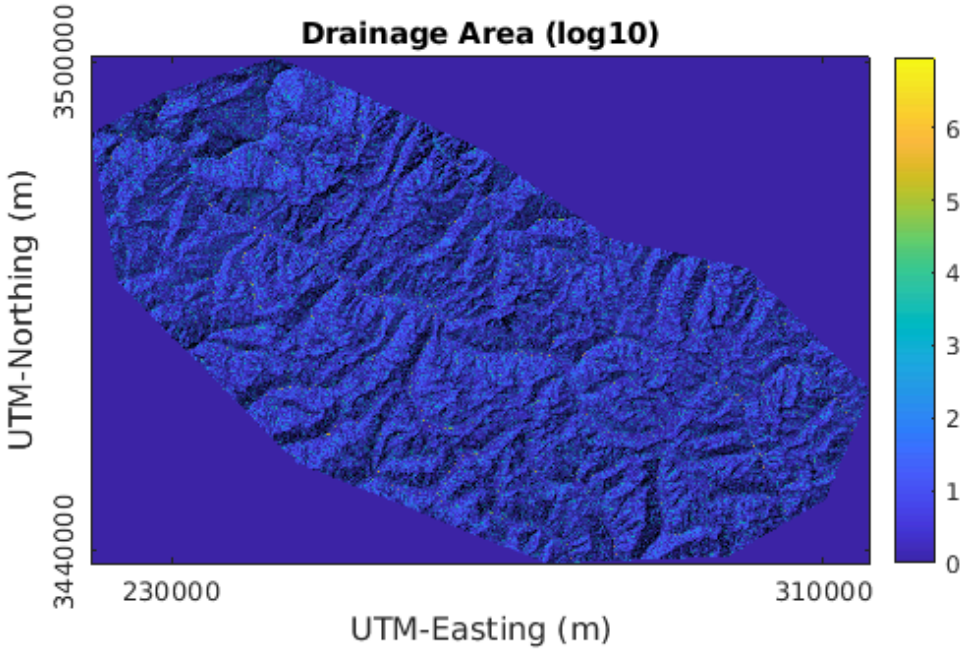
# Preparing data

This DEM is taken from the NASADEM (reprocessed SRTM1 with 30m spatial resolution). It has been reprojected to UTM46N with bilinear interpolation and stored in a compressed geotiff file. The gdal command used: gdalwarp -tr 30 30 -t_srs epsg:32646 -r bilinear -co COMPRESS=DEFLATE -co ZLEVEL=9 demLat_N28_N34_Lon_E075_E082.dem.nc demLat_N28_N34_Lon_E075_E082.UTM46N.WGS84.tif

# Load a DEM

```
dem_fname="Baspa_ALOS12_5m_UTM44N.tif";
Baspa_dem = GRIDobj(dem_fname);

%Verify that you have extracted the correct DEM:
%imageschs(Baspa_dem, Baspa_dem)

Baspa_dem_FIL = fillsinks(Baspa_dem);
Baspa_FD = FLOWobj(Baspa_dem_FIL,'preprocess','carve');
Baspa_FAC = flowacc(Baspa_FD);

figure
imageschs(Baspa_dem, log10(Baspa_FAC), ...
    'ticklabels','nice','colorbar',true);
colorbar
title('Drainage Area (log10)');
ylabel('UTM-Northing (m)', 'Fontsize', 12);
xlabel('UTM-Easting (m)', 'Fontsize', 12);
```

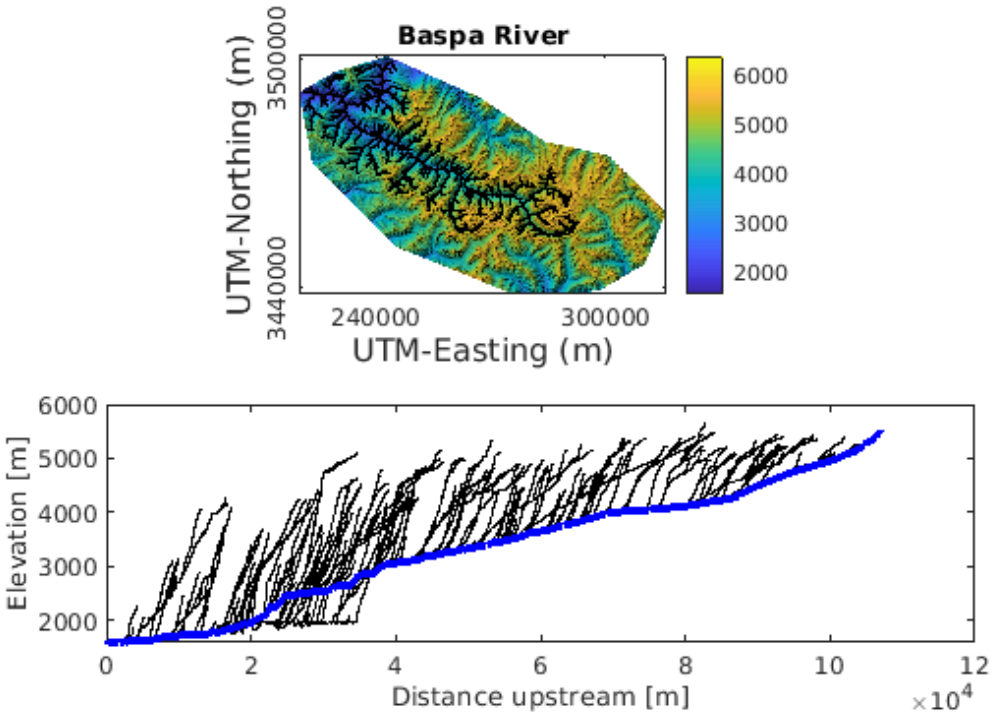
```
dem_resolution = Baspa_dem.refmat(2,1);

% get the DEM resolution, used to convert area threshold to meters
 squared
area_threshold = 1e6;
minApix = area_threshold/(dem_resolution*dem_resolution);
minApix = ceil(minApix);  % convert area threshold to meters specified
 above

Baspa_FAC_w = Baspa_FAC > minApix;  % masks flow accum grid (above
 threshold)

%extract river
Baspa_rivers_STR = STREAMobj(Baspa_FD, Baspa_FAC_w);
```




# extract largest catchment and plot

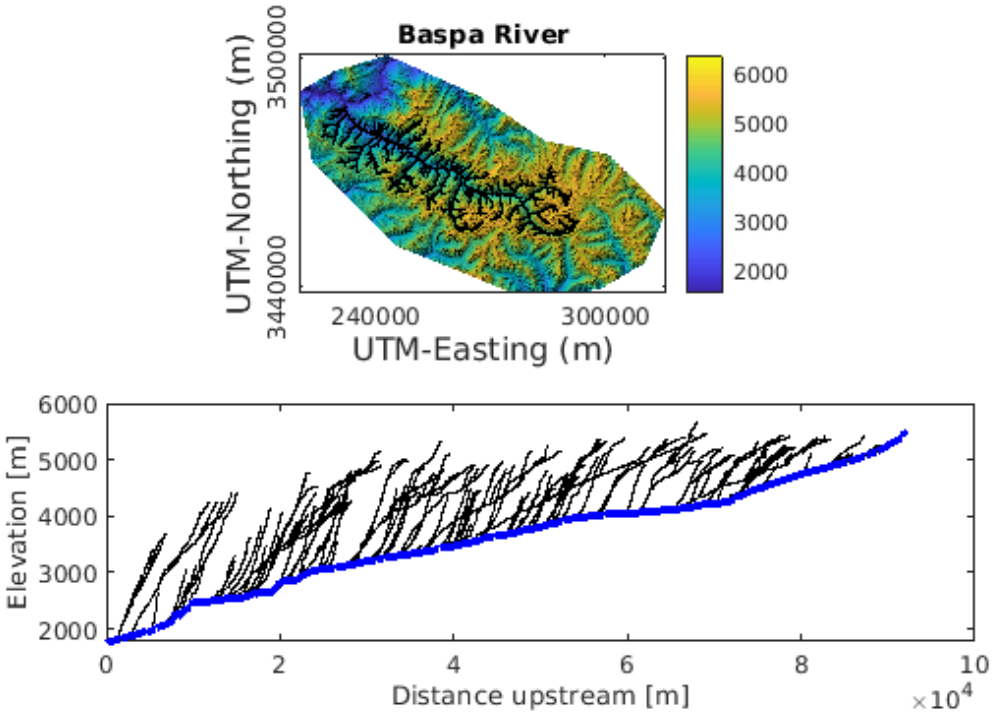
```
Baspa_rivers_STR = klargestconncomps(Baspa_rivers_STR, 1);
figure
clf
subplot(2,1,1)
imageschs(Baspa_dem, Baspa_dem, ...
    'ticklabels','nice','colorbar',true)
ylabel('UTM-Northing (m)', 'Fontsize', 12);
xlabel('UTM-Easting (m)', 'Fontsize', 12);
hold on
plot(Baspa_rivers_STR, 'k')
title('Baspa River')
```

```
subplot(2,1,2)
plotdz(Baspa_rivers_STR, Baspa_dem, 'color', 'k', 'linewidth', 0.5)
hold on
plotdz(trunk(Baspa_rivers_STR),Baspa_dem, 'color', 'b', 'linewidth',
 3)
```




```
[Baspa_dbasins, Baspa_dbasins_outlet] = drainagebasins(Baspa_FD,
 Baspa_rivers_STR);
%Baspa_catchment=232287.8876,3487440.8685
[Baspa_x, Baspa_y] =
 snap2stream(Baspa_rivers_STR,232287.8876,3487440.8685);
[Baspa_dbasins, Baspa_dbasins_outlet] = ...
    drainagebasins(Baspa_FD, Baspa_x, Baspa_y);
Baspa_dbasins.Z = logical(Baspa_dbasins.Z);
Baspa_river_mask = and(Baspa_FAC_w, Baspa_dbasins);

%mask out Baspa basin:
Baspa_rivers_STR = STREAMobj(Baspa_FD, Baspa_river_mask);

%replot only Baspa
figure
clf
subplot(2,1,1)
imageschs(Baspa_dem, Baspa_dem, ...
    'ticklabels','nice','colorbar',true)
ylabel('UTM-Northing (m)', 'Fontsize', 12);
xlabel('UTM-Easting (m)', 'Fontsize', 12);
hold on
plot(Baspa_rivers_STR, 'k')
title('Baspa River')
```

```
subplot(2,1,2)
plotdz(Baspa_rivers_STR, Baspa_dem, 'color', 'k', 'linewidth', 0.5)
hold on
plotdz(trunk(Baspa_rivers_STR),Baspa_dem, 'color', 'b', 'linewidth',
 3)
```




# calculate slope and area from DEM

set a minimum gradient (no place in DEM has a slope of 0)

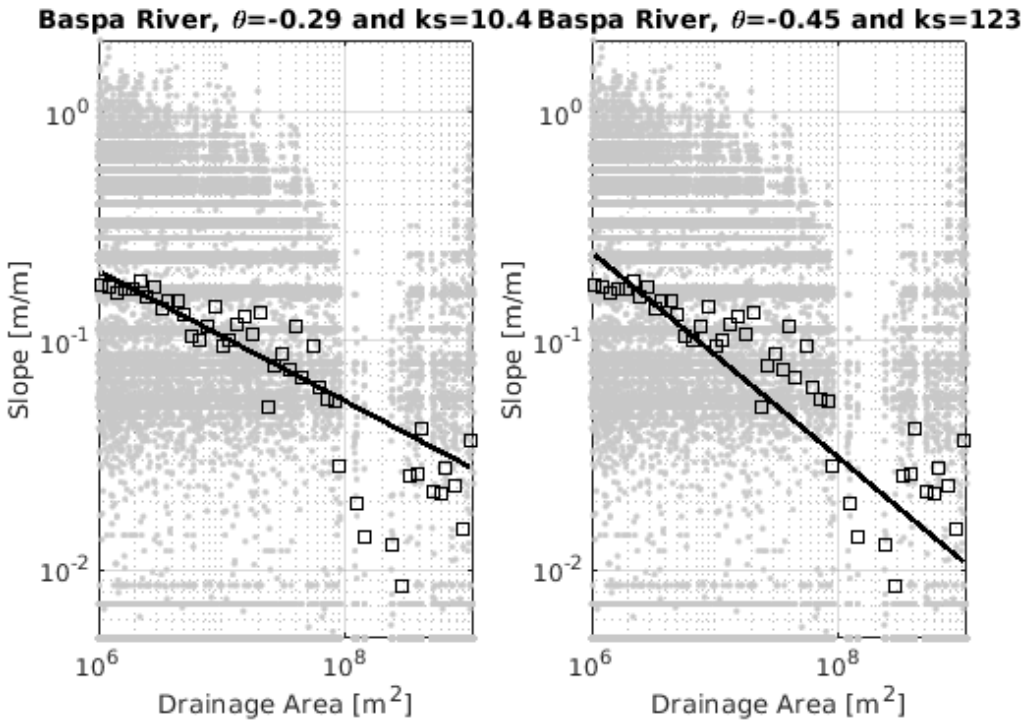
```
min_str_gradient = 5e-3;
Baspa_mg = imposemin(Baspa_FD, Baspa_dem, min_str_gradient);

Baspa_rivers_slope =
 gradient(Baspa_rivers_STR,Baspa_mg,'unit','tangent');
Baspa_rivers_area = Baspa_FAC.Z(Baspa_rivers_STR.IXgrid).*...
    (Baspa_FAC.cellsize).^2;
Baspa_rivers_slope_5eminus3 = ...
    find(Baspa_rivers_slope < 4e-3);
Baspa_rivers_slope(Baspa_rivers_slope_5eminus3) = [];
Baspa_rivers_area(Baspa_rivers_slope_5eminus3) = [];

%use slopearea function
Baspa_slopearea = ...
    slopearea(Baspa_rivers_STR, Baspa_mg,...
    Baspa_FAC, ...
    'areabinlocs', 'median', ...
    'gradaggfun', 'median', ...
    'streamgradient', 'robust', ...
```

```
    'mingradient', 1e-2, ...
    'plot', false, ...
    'areabins', 50);
```

# Figures

```
figure;
subplot(1,2,1)
loglog(Baspa_rivers_area, Baspa_rivers_slope, ...
    '.', 'color', [0.8 0.8 0.8])
xlabel('Drainage Area [m^2]')
ylabel('Slope [m/m]')
grid on
hold on
loglog(Baspa_slopearea.a, ...
    Baspa_slopearea.g, ...
    'ks', 'markersize' ,5)
aeval =
 logspace(log10(min(Baspa_slopearea.a)),log10(max(Baspa_slopearea.a)),10);
geval = Baspa_slopearea.ks(1)*aeval.^Baspa_slopearea.theta;
loglog(aeval, geval, 'k-', 'linewidth', 2)
title(['Baspa River, \theta=',num2str(Baspa_slopearea.theta,2),' and
 ks=',num2str(Baspa_slopearea.ks,3)])

subplot(1,2,2)
loglog(Baspa_rivers_area, ...
    Baspa_rivers_slope, '.', ...
    'color', [0.8 0.8 0.8])
xlabel('Drainage Area [m^2]')
ylabel('Slope [m/m]')
grid on
hold on
Baspa_slopearea = ...
    slopearea(Baspa_rivers_STR, Baspa_mg,...
    Baspa_FAC, ...
    'areabinlocs', 'median', ...
    'gradaggfun', 'median', ...
    'streamgradient', 'robust', ...
    'hist2', false, ...
    'mingradient', 1e-2, ...
    'plot', false, ...
    'areabins', 50, ...
    'theta', 0.45);
loglog(Baspa_slopearea.a, ...
    Baspa_slopearea.g, ...
    'ks', 'markersize' ,5)

% more flexible fitting of regression
% [beta,R,J,CovB,MSE,ErrorModelInfo] = ...
%     nlinfit(Baspa_slopearea.a,Baspa_slopearea.g,@(b,x) ...
%     b(1)*x.^b(2),[Baspa_slopearea.ks Baspa_slopearea.theta]);
%
%or use a graphical interface:
```

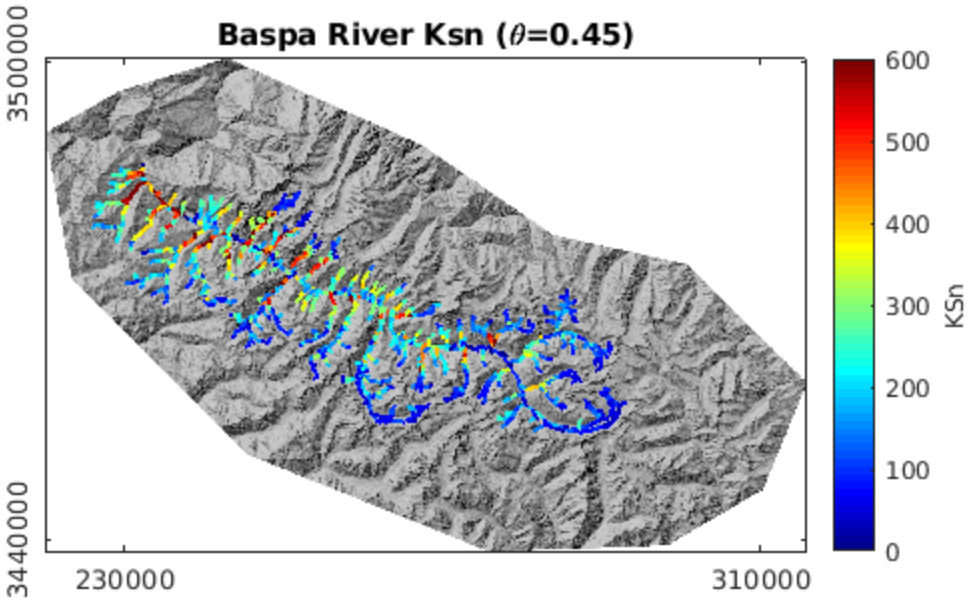
```
%nlintool(Baspa_slopearea.a,Baspa_slopearea.g,@(b,x) b(1)*x.^b(2),
[Baspa_slopearea.ks Baspa_slopearea.theta],0.05,...
%                  'area','gradient')
aeval =
 logspace(log10(min(Baspa_slopearea.a)),log10(max(Baspa_slopearea.a)),10);
geval = Baspa_slopearea.ks(1)*aeval.^Baspa_slopearea.theta;
loglog(aeval, geval, 'k-', 'linewidth', 2)
title(['Baspa River, \theta=',num2str(Baspa_slopearea.theta,2),' and
 ks=',num2str(Baspa_slopearea.ks,3)])
```

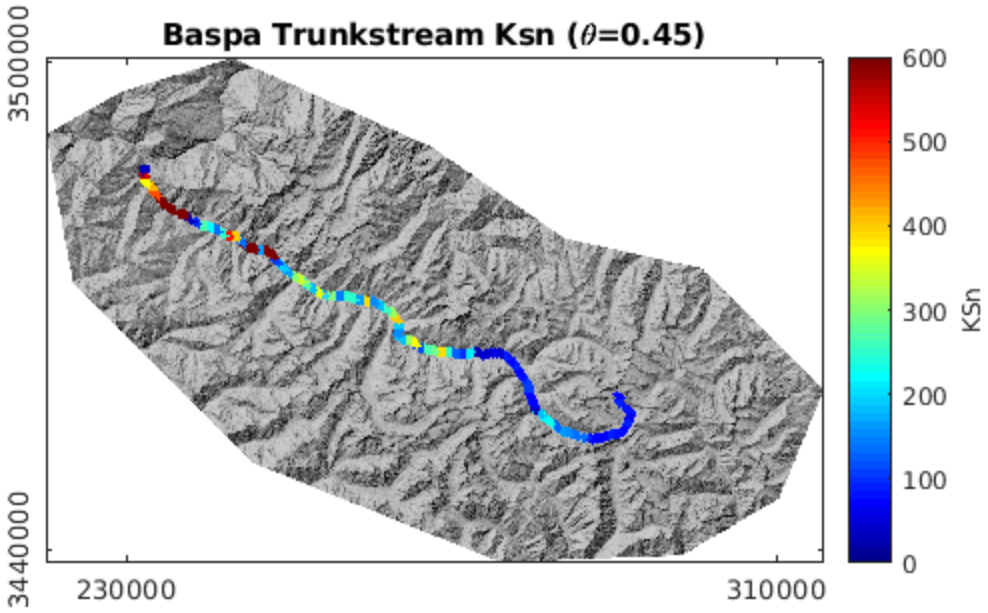



# Generate a map of steepness indices

```
Baspa_slopearea = ...
    slopearea(Baspa_rivers_STR, Baspa_mg,...
    Baspa_FAC, ...
    'areabinlocs', 'median', ...
    'gradaggfun', 'median', ...
    'streamgradient', 'robust', ...
    'mingradient', 1e-2, ...
    'plot', false, ...
    'areabins', 50);

g   = gradient(Baspa_rivers_STR, Baspa_dem);
a   = getnal(Baspa_rivers_STR, Baspa_FAC)*Baspa_FAC.cellsize^2;
Baspa_ksn = g./(a.^(-0.45));

Baspa_ksna = aggregate(Baspa_rivers_STR,...
    Baspa_ksn,'seglength',1000);
```

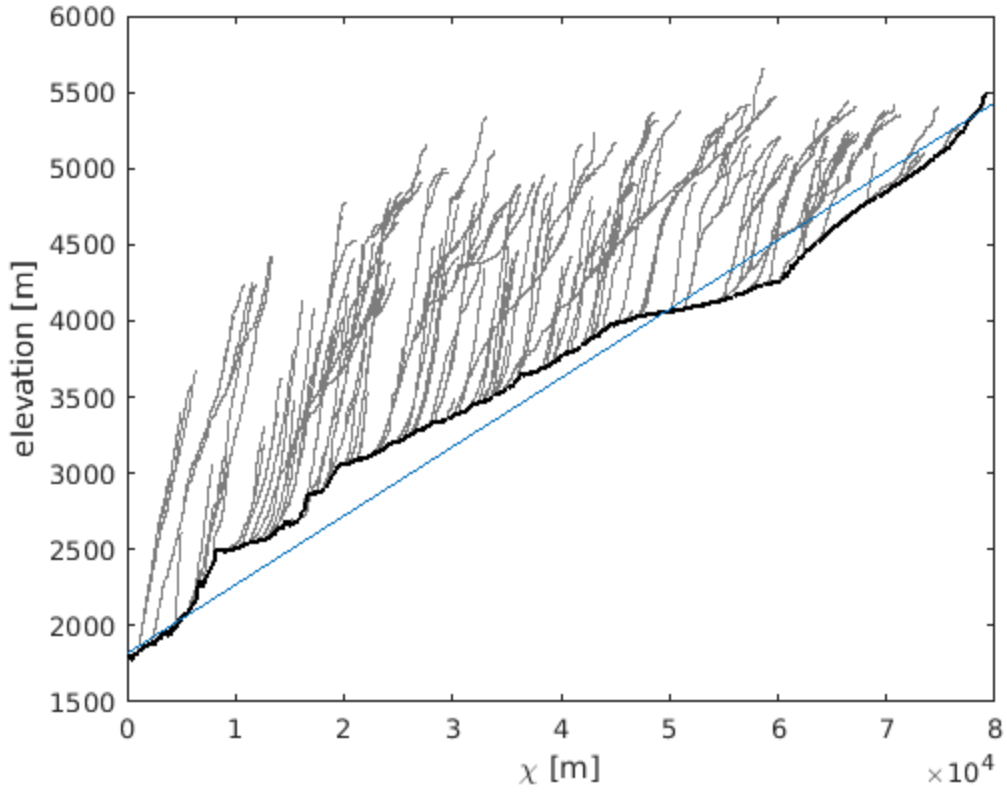
```
%repeat for trunk stream only (for separate plot):
g_trunk   = gradient(trunk(Baspa_rivers_STR), Baspa_dem);
a_trunk   = getnal(trunk(Baspa_rivers_STR),
 Baspa_FAC)*Baspa_FAC.cellsize^2;
Baspa_ksn_trunk = g_trunk./(a_trunk.^(-0.45));
Baspa_ksna_trunk = aggregate(trunk(Baspa_rivers_STR),...
    Baspa_ksn_trunk,'seglength',1000);

figure
imageschs(Baspa_dem,[],...
    'ticklabels','nice', ...
    'colormap',[.8 .8 .8],...
    'colorbar',false);
hold on
plotc(Baspa_rivers_STR, Baspa_ksna, ...
    'linewidth', 2)
caxis([0 600])
h   = colorbar;
colormap(jet)
h.Label.String = 'KSn';
box on
axis image
title('Baspa River Ksn (\theta=0.45)');

figure
imageschs(Baspa_dem,[],...
    'ticklabels','nice', ...
    'colormap',[.8 .8 .8],...
    'colorbar',false);
hold on
plotc(trunk(Baspa_rivers_STR), Baspa_ksna_trunk, ...
    'linewidth', 4)
caxis([0 600])
h   = colorbar;
colormap(jet)
h.Label.String = 'KSn';
box on
axis image
title('Baspa Trunkstream Ksn (\theta=0.45)');
```


Baspa River Ksn ($\theta$=0.45)
3500000
3440000
230000
310000
600
500
400
300
200
100
0
KSn




# Chiplot

```
c = chiplot(Baspa_rivers_STR,Baspa_dem,...
    Baspa_FAC,...
    'mnplot', false, ...
    'fitto', 'ts', ...
    'trunkstream', trunk(Baspa_rivers_STR), ...
    'a0', 1e6);
```




*Published with MATLAB® R2019a*