Spherical shell coordinates

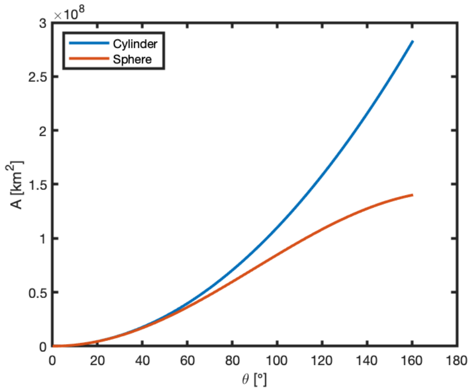
```
clear
set_demo_defaults;
R_mars = 3389508;     % [m] Mars' mean radius
grav = 3.711;    % [m/s^2] grav. acceleration on Mars
```

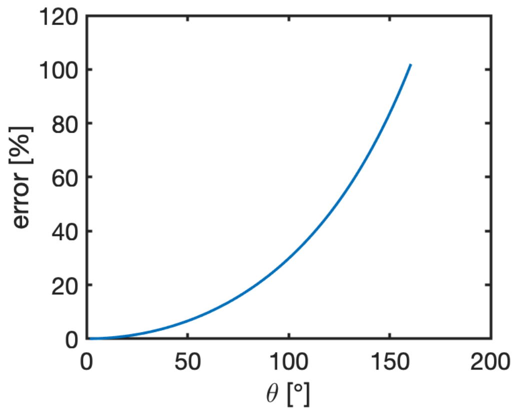
We have seen that moving to cylindrical coordinates removed the ambiguity in the interpretation of the polar recharge. To properly incorporate precipitation we need to go to a geometry with a meaningful surface area. In 1D linear coordinates the surface area is an arbitrary function of the undetermined width. In cylindrical coordinates we have a proper surface area, but given that the southern highlands aquifer stretches halfway through the northern hemisphere the we have a huge error in the actual surface area compared to a sphere, see figure. The same would be true for any estimate of the actual groundwater volume.

```
theta_bnd = acos(1/3);
theta_bnd_deg = rad2deg(theta_bnd)
```

```
theta_bnd_deg = 70.5288
```

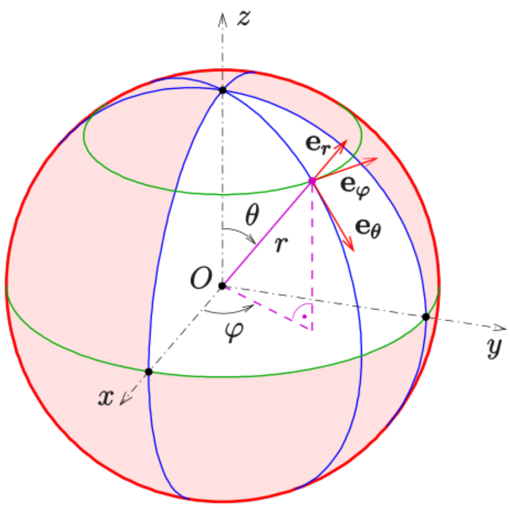
```

theta_vec = linspace(0,90+theta_bnd_deg,100);
l = R_mars*deg2rad(theta_vec); % [m] distance to dichotomy bnd

A_cyl = pi*l.^2;
A_cap = 2*pi*R_mars^2*(1-cos(deg2rad(theta_vec)));

figure('position',[10 10 900 600])
subplot 121
plot(theta_vec,A_cyl/1e6,theta_vec,A_cap/1e6)
xlabel '\theta [\circ]', ylabel 'A [km^2]', pbaspect([1 .8 1])
legend('Cylinder','Sphere','location','northwest')

subplot 122
plot(theta_vec,(A_cyl-A_cap)./A_cap*100)
xlabel '\theta [\circ]', ylabel 'error [%]', pbaspect([1 .8 1])
```





Spherical coordinates

This motivates us to discretize the discrete operators in spherical coordinates. The definition of standard variables in spherical coordinates is shown in the figure below.



Here r is the radial coordinate, θ is the co-lattitude and φ is the circumfrencial coordinate. The associated definition of the gradient and the divergence

- $\nabla h = \frac{\partial h}{\partial r}\hat{\boldsymbol{\rho}} + \frac{1}{r}\frac{\partial h}{\partial \theta}\hat{\boldsymbol{\theta}} + \frac{1}{r\sin\theta}\frac{\partial h}{\partial \varphi}\hat{\boldsymbol{\varphi}}$

- $\nabla \cdot \mathbf{q} = \frac{1}{r^2}\frac{\partial(r^2 q_r)}{\partial r} + \frac{1}{r\sin(\theta)}\frac{\partial}{\partial\theta}(\sin(\theta)\,q_\theta) + \frac{1}{r\,\sin(\theta)}\frac{\partial q_\varphi}{\partial\varphi}$

where $\mathbf{q} = [q_r q_\theta, q_\varphi]$.

The southern highlands aquifer is in a spherical shell, so that $r = R_{\text{Mars}}$ is fixed. To obtain a one-dimensional model we assume no change in the circumferential dirrection, φ, so that $\partial/\partial\varphi = 0$. Therefore, the remaining independent variable is the co-lattitude, θ, and one-dimensional operators in sherical shell geometry, $x = \theta$, are

- $\nabla h = \frac{1}{R_{\text{Mars}}}\frac{\mathrm{d}h}{\mathrm{d}x}$
- $\nabla \cdot \mathbf{q} = \frac{1}{R_{\text{Mars}}\sin(x)}\frac{\mathrm{d}}{\mathrm{d}x}(\sin(x)\,q)$

In spherical shell coordinates both the divergence and the gradient change. Again we have to amend the function `build_ops.m`.

Discrete operators

The discrete divergence and gradient matrix in spherical shell geometry can therefore be obtained as follows:

```
Grid.xmin = 0.1; Grid.xmax = 1; Grid.Nx = 35;
Grid = build_grid(Grid);
[D,G,I] = build_ops(Grid);

% Modification for spherical shell
Grid.R_shell = R_mars;
Rf = spdiags(sin(Grid.xf),0,Grid.Nx+1,Grid.Nx+1);
Rcinv = spdiags(1./(Grid.R_shell*sin(Grid.xc)),0,Grid.Nx,Grid.Nx);
D = Rcinv*D*Rf;
G = G/Grid.R_shell;
L = -D*G;
```

Similar to the cylindrical coordinates we evaluate terms outside the divergence at cell centers and terms inside the divergence at cell faces.

Spherical shell aquifer with precipitation

Dimensional

The equations for the steady confined aquifer with precipitation on a spherical shell are given by

$$\frac{1}{R_{\text{Mars}}\sin(\theta)}\frac{\mathrm{d}}{\mathrm{d}\theta}\left(\sin(\theta)\,bK\,\frac{1}{R_{\text{Mars}}}\frac{\mathrm{d}h}{\mathrm{d}x}\right) = q_p \text{ on } \theta \in [0, \theta_b]$$

with the boundary conditions

$$\left.\frac{\mathrm{d}h}{\mathrm{d}\theta}\right|_0 = 0 \text{ on } h(\theta_b) = h_o$$

Th parameter values are as before

```
yr2s = 60^2*24*365.25;  % second per year
rho = 1e3;              % [kg/m^3] desity of water
grav = 3.711;           % [m/s^2] grav. acceleration on Mars
k = 1e-11;              % [m^2] permeability (Hanna & Phillips 2005)
mu = 1e-3;              % [Pa s] water viscosity
ho = -500;              % [m] sea level
b = 5e3;                % [m] aquifer thickness
theta_bnd = pi-acos(1/3);  % [rad] angel dichotomy boundray from south pole
% derived values
K = k*rho*grav/mu;      % [m/s] hydraulic conductivity
```

Dimensionless

The angle, θ, in radiants is defined as ratio of the arc length, s, to the radius, R, of the circle, $\theta = s/R$, and hence dimensionless and of order one. We define the characteristic scale $h_c = q_p R_{\mathrm{Mars}}^2/(bK)$ and the associated dimensionless head $h' = (h - h_o)/h_c$. This induces the scale $q_c = q_p R_{\mathrm{Mars}}/b$ for the flux. The dimensionless equations are

$$-\frac{1}{\sin\theta}\frac{\mathrm{d}}{\mathrm{d}\theta}\left(\sin\theta\frac{\mathrm{d}h'}{\mathrm{d}\theta}\right) = 1 \text{ on } \theta \in [0, \theta_b]$$

with the boundary conditions

$$\left.\frac{\mathrm{d}h'}{\mathrm{d}\theta}\right|_0 = 0 \text{ and } h'(\theta_b) = 0.$$

The dimensionless flux is simply $q' = -\frac{\mathrm{d}h'}{\mathrm{d}\theta}$. The only dimensionless parameter is therfore the angle of the dichotomy boundary θ_b.

Analytic solutions

The dimensionless analytic solution is given by

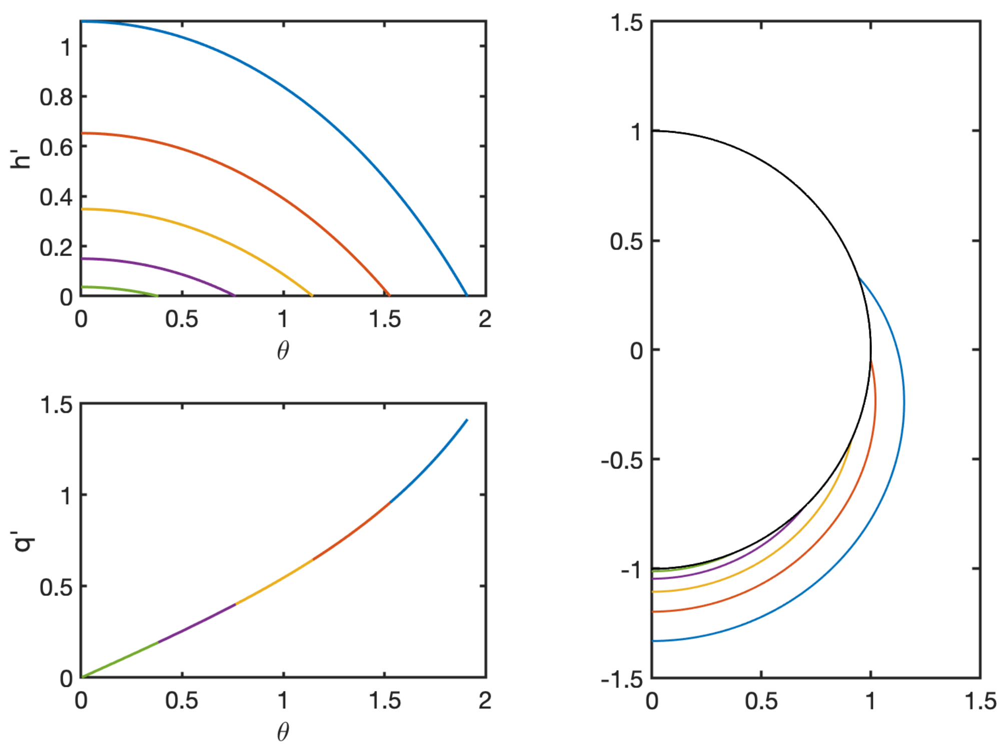
$$h' = \log\left(\frac{\cos\theta + 1}{\cos\theta_b + 1}\right) \text{ and } q' = \frac{1 - \cos\theta}{\sin\theta} = \csc\theta - \cot\theta.$$

```
hD_ana = @(theta,theta_bnd) log((cos(theta)+1)/(cos(theta_bnd)+1));
qD_ana = @(theta) csc(theta) - cot(theta);
```

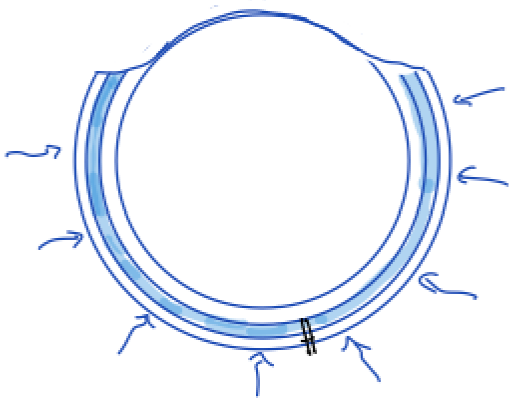
The solution is show in the figure below for increasing co-lattidudes of the dichotomy boundary

```
h_scale_plot = 0.3;
theta_bnd_vec = [theta_bnd:-theta_bnd/5:theta_bnd/5];

figure('position',[10 10 900 600])
for i = 1:length(theta_bnd_vec)
```

```
    theta =  linspace(0,theta_bnd_vec(i),1e2);
    subplot(2,2,1)
    plot(theta,hD_ana(theta,theta_bnd_vec(i))); hold on
    subplot(2,2,3)
    plot(theta,qD_ana(theta)); hold on
    subplot(2,2,[2 4])
    x_h = (1+hD_ana(theta,theta_bnd_vec(i))*h_scale_plot).*sin(theta);
    z_h = (1+hD_ana(theta,theta_bnd_vec(i))*h_scale_plot).*cos(theta);
    plot(x_h,-z_h,'-','linewidth',1.5), hold on
end
subplot(2,2,1)
ylabel 'h''', xlabel '\theta'
subplot(2,2,3)
ylabel 'q''', xlabel '\theta'
subplot(2,2,[2 4])
theta_sphere = linspace(0,pi,5e2);
x_base = sin(theta_sphere);
z_base = cos(theta_sphere);
plot(x_base,z_base,'k','linewidth',1.5)
axis equal
xlim([0 1.5]), ylim(1.5*[-1 1])
```



Whe looking at the picture of the head on the sphere, keep in mind that this is not the groundwater table, we are still looking at a confined aquifer that looks something like the sketch below



Numerical solution

The construction of the modified divergence and gradient will be integrated into the function `build_ops.m` and can be activated by a new field in the `Grid` structure called `Grid.geom = 'spherical_shell';`.

Note, for the flux computations the vectors Grid.V and Grid.A have to be aupdated appropritely!

Note also that all the other functions (build_bnd.m, solve_lbvp.m, comp_flux.m) **do not** need to be updated.

```
theta_ana = linspace(0,theta_bnd,1e2);

Grid.xmin = 0;
Grid.xmax = theta_bnd;
Grid.Nx = 1e1;
Grid.geom = 'spherical_shell';
Grid.R_shell = 1;
Grid = build_grid(Grid);

% Operators
[D,G,I] = build_ops(Grid);
L = -D*G;
fs = ones(Grid.Nx,1);

% Boundary conditions
BC.dof_dir = [Grid.dof_xmax];
BC.dof_f_dir = Grid.dof_f_xmax;
BC.g = hD_ana(Grid.xc(Grid.dof_xmax),theta_bnd);
BC.dof_neu = [];
BC.dof_f_neu = [];
BC.qb = [];
[B,N,fn] = build_bnd(BC,Grid,I);

hD = solve_lbvp(L,fs,B,BC.g,N);
qD = comp_flux(D,1,G,hD,fs,Grid,BC);
```

```
figure('position',[10 10 900 600])
subplot 121
plot(theta_ana,hD_ana(theta_ana,theta_bnd)), hold on
plot(Grid.xc,hD,'o','markerfacecolor','w','markersize',8)
xlabel '\theta [rad]', ylabel 'h'' ', pbaspect([1 .8 1])
legend('analytic','numeric','location','southwest')

subplot 122
plot(theta_ana,qD_ana(theta_ana)), hold on
plot(Grid.xf,qD,'o','markerfacecolor','w','markersize',8)
xlabel '\theta [rad]', ylabel 'q'' ', pbaspect([1 .8 1])
legend('analytic','numeric','location','northwest')
```