This file contains the mathematical derivation and explanations of corresponding code at https://github.com/xinan-nancy-chen/rOMT.

Created by Xinan Chen at 04/13/21 and finished at 04/16/21.
I apologize for my poor handwriting if it causes your trouble to read.

A LaTeX version is expected in the future.

rOMT code. written by Xinan Chen on 04/13/21.

rOMT: regularized optimal mass transport.

Question in continuous version:

$$\begin{cases} \min\limits_{\rho, v} \int_0^1 \int_\Omega \rho(t,x)\|v(t,x)\|^2 dx\,dt \\ \text{s.t.} \begin{cases} \rho_t + \nabla\cdot(\rho v) = \delta\Delta\rho \quad ① \\ \rho(0,x)=\rho_0(x),\ \rho(1,x)=\rho_1(x) \end{cases} \end{cases}$$

where $\rho(t,x)$: density (function) at location x and time t.

$v(t,x)$: velocity field ....

① advection-diffusion equation.

Question in discrete version:

$$\begin{cases} \min\limits_{v} \Gamma(v) = \beta\cdot hd\cdot dt\ \underbrace{\rho^T}_{1\times mn}\underbrace{\left(I_m\otimes[I_n|I_n|I_n]\right)}_{3mn\times 3mn}\underbrace{(v\odot v)}_{3mn\times 1} \\ \qquad + \frac{1}{2}\|\rho_m-\rho_1^{obs}\|^2 + \frac{1}{2}\gamma\cdot hd\cdot dt\,\|\text{Grad}\,v\|^2 \\ \text{s.t.} \begin{cases} (I_n - dt\,\delta Q)\rho_{i+1} = S(v_i)\rho_i,\ i=0,\cdots,m-1. \quad (*) \\ \rho_0=\rho_0^{obs} \end{cases} \end{cases}$$

(ρ is $mn\times 1$; matching final observed image; smoothing v)

denote dimension of matrix / vector

where $\rho = \begin{pmatrix}\rho_1\\ \vdots \\ \rho_m\end{pmatrix}$ ($mn\times 1$) vertical vector

$v = \begin{pmatrix} v_0 \\ \vdots \\ v_{m-1}\end{pmatrix}$ ($3mn\times 1$) vertical vector

$\otimes$: Kronecker; $\odot$: Hadamard product.

Grad $= \nabla$ = gradient, is a matrix

The discretization is

for space, it is a $n_x \times n_y \times n_z$ cube of same length. hd = volume of each sub-cube (in our code, we use $hd = 1*1*1 = 1$).

in total $n = n_x\cdot n_y\cdot n_z$ cells,

for time, it is divided into nt equal intervals of length dt. (in this note, we use "m" to replace "nt" sometimes).

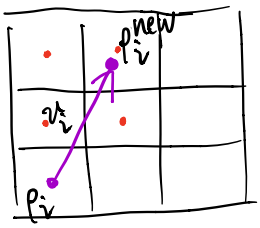
$t=0$	$t=dt$	$t=2\cdot dt$	....	$t=(m-1)dt$	$t=mdt$
$\rho_0=\rho_0^{obs}$ (given) $\xrightarrow{v_0}$	ρ_1 $\xrightarrow{v_1}$	ρ_2 $\xrightarrow{v_2}$	... $\xrightarrow{v_{m-2}}$	ρ_{m-1} $\xrightarrow{v_{m-1}}$	ρ_m

From $\rho_0=\rho_0^{obs}$ and v, we can derive ρ by (*) (corresponds to advecDiff.m in code).

in (*), it's the discretization of equation ①, which is a process of advection + diffusion.

So we solve it numerically as first advection, second diffusion.

for i, advection: $\rho_t + \nabla\cdot(\rho v) = 0$. ($\rho$: movement of ρ according to v) $\Rightarrow$ $\rho_i^{adv} = S(v_i)\rho_i$, where $S(v_i)$ is the averaging matrix w.r.t v_i



(corresponds to dTrilinear.m in code)

Particle in cell method:
move ρ_i according to v_i to get new location ρ_i^{new}
then split ρ_i^{new} into its nearest neighbor grid points by a ratio determinted by distance, i.e. the farther distance, the smaller the ratio.
After that, we get ρ_i^{adv} which is the distribution after advection.

diffusion: $\rho_t = 6\Delta\rho \Rightarrow \frac{\rho_{i+1} - \rho_i^{adv}}{dt} = Q\rho_{i+1}$ → Backwards scheme

Q is discretization of 6Δ on grid.

$\Rightarrow (I_n - dt\,Q)\rho_{i+1} = \rho_i^{adv}$

So combine the two steps together, we have (*).

Denote $L_n = I_n - dt\,Q$.

From (*) we have $\rho_{i+1} = L^{-1}S(v_i)\rho_i$, $i = 0, \cdots, m-1$.

So for $k = 1, \cdots, m$.

$$\rho_k = L^{-1}S(v_{k-1})L^{-1}S(v_{k-2})\cdots L^{-1}S(v_0)\rho_0$$

$\Rightarrow \rho_k = \rho_k(v_0, \cdots, v_{k-1})$ → w.r.t $v_0, \cdots, v_{k-1}$.

For energy functional $P(v)$, it has three terms,

$$P(v) = \beta P_1(v) + P_2(v) + \gamma P_3(v).$$

$P_1(v) \sim \rho \|\nabla v\|^2$

$$\rho = \begin{pmatrix} \rho_1 \\ \vdots \\ \rho_m \end{pmatrix} = \begin{pmatrix} \rho_1 & & \\ n & 1 & \\ & \vdots & \\ & & \rho_m \\ & & n \quad 1 \end{pmatrix}_{mn \times 1}$$

$$\Rightarrow P_1(v) = \text{hd} \cdot \text{dt} \cdot \sum_{i=1}^{m} \rho_i \odot \left(v_{i+1,x}^2 + v_{i+1,y}^2 + v_{i+1,z}^2 \right)$$

$$v = \begin{pmatrix} v_0 \\ \vdots \\ v_{m-1} \end{pmatrix} = \begin{pmatrix} v_0 & & \\ 3n & 1 & \\ & \vdots & \\ & & v_{m-1} \\ & & 3n \quad 1 \end{pmatrix}_{3mn \times 1} = \begin{pmatrix} x & & \\ y & & v_0 \\ z & & \\ & \vdots & \\ & & v_{m-1} \end{pmatrix}_{3mn \times 1}$$

We solve this optimization problem with Newton-Gaussian method:

$$g(v) = \nabla \Gamma(v) = \frac{\partial \Gamma(v)}{\partial v} \rightarrow \text{gradient of } \Gamma(v)$$

$$H(v) = \left(\frac{\partial^2 \Gamma(v)}{\partial v_i \partial v_j} \right)_{i,j} \rightarrow \text{Hessian matrix of } \Gamma(v)$$

Algorithm

```
v = initial guess;
for i = 1, ..., MaxIter
    Compute g(v), H(v);
    Solve Hs = -g for s;
    Do line search to find length l;
    v = v + ls;
end
```

Next we look into how to compute g and H.

recall that

$$\Gamma(v) = \beta \cdot hd \cdot dt \underbrace{\rho^T}_{1 \times mn} \underbrace{\left(I_m \otimes [I_n | I_n | I_n]\right)}_{3mn} (v \odot v) + \frac{1}{2} \| \rho_m - \rho_1^{obs} \|^2 + \frac{1}{2} \gamma \cdot hd \cdot dt \| \text{Grad } v \|^2$$

$$= \beta \Gamma_1(v) + \Gamma_2(v) + \gamma \Gamma_3(v)$$

$$\Rightarrow \quad g(v) = \frac{\partial P(v)}{\partial v} = \beta \frac{\partial P_1(v)}{\partial v} + \frac{\partial P_2(v)}{\partial v} + \gamma \frac{\partial P_3(v)}{\partial v}.$$

$$1^{\circ}. \quad \frac{\partial P_i(v)}{\partial v} = \text{hd-dt} \; \frac{\partial}{\partial v}\left(\underset{1\times mn}{\rho^T} \underset{mn\times 1}{M(v\odot v)}\right) \quad \text{where we Let } \underset{mn\times 3mn}{M} = I_m \otimes \left[I_n | I_n | I_n\right]$$

$$= \text{hd-dt} \; \underset{1\times mn}{\rho^T} \underset{mn\times 3mn}{\nabla\left[M(v\odot v)\right]} + \underset{1\times mn}{\left(M(v\odot v)\right)^T} \underset{mn\times 3mn}{\nabla \rho}$$

$$= \text{hd-dt} \left[\underset{1\times mn}{2\rho^T} \underset{mn\times 3mn}{M \, \text{diag}(v)} + \underset{1\times mn}{\left(M(v\odot v)\right)^T} \underset{mn\times 3mn}{J} \right]$$

v : $3mn \times 1$, v_i : $3n \times 1$

where $\underset{mn\times 3mn}{J} = \begin{pmatrix} J^1_{v_0} & & & \\ J^2_{v_0} & J^2_{v_1} & & \\ \vdots & \vdots & \ddots & \\ J^m_{v_0} & J^m_{v_1} & \cdots & J^m_{v_{m-1}} \end{pmatrix}$, $\underset{n\times 3n}{J^k_{v_j}} = \frac{\partial \rho_k}{\partial v_j}$, $k = 1, \cdots, m$. $j = 0, \cdots, m-1$.

Since $\rho_k = \rho_k(v_0, \cdots, v_{k-1}) \Rightarrow J^k_{v_j} = 0$ if $j \geq k$

$\Rightarrow$ J is lower triangular.

For $j \leq k-1$, $J^k_{v_j} = \frac{\partial}{\partial v_j}\left(L^{-1}S(v_{k-1})L^{-1}S(v_{k-2})\cdots L^{-1}S(v_0)\rho_0\right)$

$$= \underset{n}{\underbrace{L^{-1}S(v_{k-1})\cdots L^{-1}S(v_{j+1})L^{-1}}} \frac{\partial}{\partial v_j}\left(S(v_j)L^{-1}S(v_{j-1})\cdots L^{-1}S(v_0)\rho_0\right).$$

$$= \underset{n}{L^{-1}S(v_{k-1})\cdots L^{-1}S(v_{j+1})L^{-1}} \frac{\partial}{\partial v_j}\underset{3n}{\left(S(v_j)\rho_j\right)}$$

Since $S(v_j)$ is linear to v_j, then $\frac{\partial}{\partial v_j}(S(v_j)\rho_j)$ only depends on ρ_j.

↓ see details later in dTrilinear.m.

(5)

So we denote $\frac{\partial}{\partial v_j}(S(v_j)p_j) \triangleq B(p_j)$.

$$\Rightarrow J_{v_j}^{k} = L^{-1}S(v_{k-1})\cdots L^{-1}S(v_{j+1})L^{-1}B(p_j).$$

~~Thus we have $J_{v_j}^{k+1} = L^{-1}S(v_k)J_{v_j}^{k}$.~~

Thus the second term of $\frac{\partial \pi_i(v)}{\partial v}$, which is $\underset{1}{(M(v\odot v))^T}\underset{mn\ \ 3mn}{J}$

$$= \left[\underset{mn\ \ 3mn}{M}\begin{pmatrix}\begin{bmatrix}v_{0,x}^2\\ v_{0,y}^2\\ v_{0,z}^2\end{bmatrix}\\ \vdots \\ \begin{bmatrix}v_{m-1,x}^2\\ v_{m-1,y}^2\\ v_{m-1,z}^2\end{bmatrix}\end{pmatrix}\right]^T J$$

$$= \underset{mn}{\begin{bmatrix} I_n I_n I_n & & & \\ & I_n I_n I_n & & \\ & & \ddots & \\ & & & I_n I_n I_n \end{bmatrix}}\left(\begin{pmatrix}\begin{bmatrix}v_{0,x}^2\\ v_{0,y}^2\\ v_{0,z}^2\end{bmatrix}\\ \vdots \\ \begin{bmatrix}v_{m-1,x}^2\\ v_{m-1,y}^2\\ v_{m-1,z}^2\end{bmatrix}\end{pmatrix}_{3mn\ \ 1}\right)^T J$$

$$= \underset{1}{\begin{pmatrix}\|v_0^T\|^2 & \cdots & \|v_{m-1}^T\|^2\\ 1 & & n\end{pmatrix}}_{mn}\underset{mn}{\begin{pmatrix} {}^{n}J_{v_0}^{1}{}_{3n} & & & \\ J_{v_0}^{2} & J_{v_1}^{2} & & \\ \vdots & \vdots & \ddots & \\ J_{v_0}^{m} & J_{v_1}^{m} & \cdots & J_{v_{m-1}}^{m}\end{pmatrix}}_{3mn}$$

where $\underset{n}{\|v_i\|^2}_{1} = \begin{pmatrix} \|v_{loc=1}\|^2 \\ \vdots \\ \|v_{loc=n}\|^2 \end{pmatrix}_{n,1}$

$= \underset{1}{\left(D_1\right.}\underset{3n}{D_2 \cdots D_m\left.\right)}_{3mn}$

where $D_k = \sum_{i=k}^{m} \underset{1}{\|v_{i-1}^T\|^2}_{n} \underset{3n}{J_{v_{k-1}}^{i}}$, $k=1,\cdots,m$.

2° $\underset{1}{\frac{\partial P_2(v)}{\partial v}}_{3mn} = \frac{1}{2}\frac{\partial}{\partial v}\|\rho_m - \rho_1^{obs}\|^2 = \underset{1}{(\rho_m - \rho_1^{obs})^T}\underset{n}{\frac{\partial \rho_m}{\partial v}}_{3mn}$

$= (\rho_m - \rho_1^{obs})^T (J_{v_0}^m \; J_{v_1}^m \cdots J_{v_{m-1}}^m)$.

3° $\frac{\partial P_3(v)}{\partial v} = \frac{1}{2}hd\cdot dt \frac{\partial}{\partial v}\underset{3mn}{\|Grad\, v\|^2}_{3mn,1} = hd\cdot dt\,(Grad\, v)^T\, Grad$

$= hd\cdot dt\, v^T\, Grad^T\, Grad$.

For the Hessian matrix $H = \left(\frac{\partial^2 P}{\partial v_i v_j}\right)_{i,j} = \underset{3mn}{\frac{\partial g}{\partial v}}_{3mn}$

recall that $g = \beta\, hd\cdot dt\left[2\rho^T M\, diag(v) + [M(v\odot v)]^T J\right]$

$+ (\rho_m - \rho_1^{obs})^T \frac{\partial \rho_m}{\partial v} + \gamma\cdot hd\cdot dt\, v^T\, Grad^T\, Grad$

$$\Rightarrow H = \frac{\partial g}{\partial v} = \beta \cdot hd \cdot dt \Big[2\rho^T \nabla(M\,\mathrm{diag}(v)) + \nabla(2\rho)\, M\,\mathrm{diag}(v) + M(v \circ w)\nabla J + \nabla[M(v \circ w)]\, J \Big]$$

$$+ \left(\frac{\partial \rho_m}{\partial v}\right)^T \left(\frac{\partial \rho_m}{\partial v}\right) + (\rho_m - \rho_1^{obs}) \frac{\partial^2 \rho_m}{\partial v^2}.$$

$$+ \gamma \cdot hd \cdot dt\ Grad^T \cdot Grad.$$

numerically we approximate H with only first three terms in red, so

$$H \approx 2\beta \cdot hd \cdot dt \cdot \mathrm{diag}\Big(\underset{1}{\rho^T}\underset{mn}{M}\Big) + \underset{3mn}{\left(\frac{\partial \rho_m}{\partial v}\right)^T}\underset{n}{\left(\frac{\partial \rho_m}{\partial v}\right)}\underset{3mn}{}$$

$$+ \gamma \cdot hd \cdot dt\ Grad^T \cdot Grad.$$

Next we go to the code.

In driver.m, we have consecutive observed image $\rho_0^{obs}, \rho_1^{obs}, \dots, \rho_{S-1}^{obs}$.

but in our note, we only consider one loop: $\rho_0^{obs} \to \rho_1^{obs}$

in the next loop, we use $\rho_m \to \rho_2^{obs}$ where ρ_m is the last-time image computed by OMT from the previous loop.

In paramInitFunc.m, we load in the parameters in the model, as well as precompute some necessary matrices, and save in structure "par."

par.n = $[n_x, n_y, n_z]$

par.h1 = par.h2 = par.h3 = 1.

par.hd = h1·h2·h3 = hd = 1.

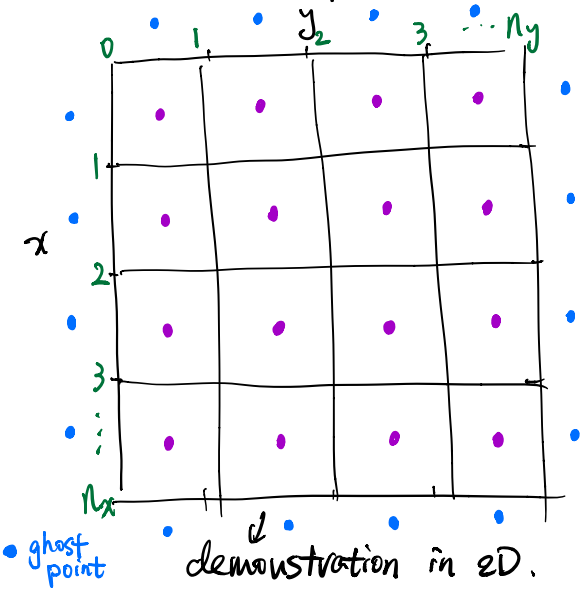
par.dt = dt in note

par.nt = m in note

par.sigma = 6 in note

We use boundary condition as cell-centered Neumann condition.

par.Xc, par.Yc, par.Zc are cell-centered grid.



demonstration in 2D.

par.Grad = gradient in space.

$$= \begin{pmatrix} Gx \to (n_x+1)\, n_y\, n_z \times n_x n_y n_z \\ Gy \to n_x (n_y+1)\, n_z \times n_x n_y n_z \\ Gz \to n_x n_y (n_z+1) \times n_x n_y n_z \end{pmatrix}.$$

When using Neumann boundary, we set ghost points on the spatial boundary, such that the value equals the nearest inside-neighbor.

In GNblock_u.m,

Rho_i = $\rho_0 = \rho_0^{obs}$ in note.

Abig = M in note.

phi = get_phi(Rho_i, u, nt, dt, par) → calculate $\Gamma(v)$. energy functional.

dmk2 = second term of $\frac{\partial \Gamma_i(v)}{\partial v}$.

$u = v$ in note.

Rho = advection (Rho_i, u, nt, dt, par) → get $\rho = \begin{pmatrix} \rho_1 \\ \vdots \\ \rho_m \end{pmatrix}$ from ρ_0 and v following the adv-diff eq.

Then in a loop, we get dmk2 — second term of $\frac{\partial \Gamma_i(v)}{\partial v}$.

vertical vector

$$g = \beta \frac{\partial \Gamma_1(v)}{\partial v} + \frac{\partial \Gamma_2(v)}{\partial v} + \gamma \frac{\partial \Gamma_3(v)}{\partial v}$$

$$= \beta \left[2\, hd \cdot dt\, \rho^T M\, diag(v) + dmk2 \right]^T + \underset{3mn}{\begin{pmatrix} J_{v_0}^{m^T} \\ J_{v_1}^{m^T} \\ \vdots \\ J_{v_{m-1}}^{m^T} \end{pmatrix}}_{3n} \underset{n}{(\rho_m - \rho_1^{obs})}_{1}$$

$$+ \gamma\, hd \cdot dt \cdot Grad^T Grad\, v$$

$[p^T M \mathrm{diag}(x)]^T = \mathrm{diag}(x) M^T p$

$$Hx = 2\beta \cdot hd \cdot dt \cdot \underbrace{\mathrm{diag}(p^T M)}_{3mn} x + \underbrace{\left(\frac{\partial \rho_m}{\partial v}\right)^T}_{n} \underbrace{\frac{\partial \rho_m}{\partial v} x}_{3mn\ 1} \quad \text{linear form.}$$

$$+ \gamma \cdot hd \cdot dt \ \mathrm{Grad}^T \mathrm{Grad}\, x$$

$$\left(H = 2\beta \cdot hd \cdot dt \cdot \mathrm{diag}(p^T M) + \left(\frac{\partial \rho_m}{\partial v}\right)^T \frac{\partial \rho_m}{\partial v} + \gamma\, hd \cdot dt\ \mathrm{Grad}^T \mathrm{Grad} \right) \text{matrix form}$$

$\frac{\partial \rho_m}{\partial v} x \rightarrow$ get_drNdu(x)

$\left(\frac{\partial \rho_m}{\partial v}\right)^T \left(\frac{\partial \rho_m}{\partial v} x\right) \rightarrow$ get_drNduT$\left(\frac{\partial \rho_m}{\partial v} x\right)$

Solve $HS = -g$ for S.

linear Search.

In advecDiff.m,

recall that, given ρ_0 and v, we can derive $\rho = \begin{pmatrix} \rho_1 \\ \vdots \\ \rho_m \end{pmatrix}$,

with $\begin{cases} \rho_i^{adv} = S(v_i)\rho_i \\ (I_n - dt\,Q)\rho_{i+1} = \rho_i^{adv}, \quad i = 0, \cdots, m-1. \end{cases}$

Mdis = Q in note.

rho = $\begin{pmatrix} \rho_0 \\ \rho \end{pmatrix} = \begin{pmatrix} \rho_0 \\ \rho_1 \\ \vdots \\ \rho_m \end{pmatrix}$ in note.

in note: t	0	1	2	...	m
in note: i	1	2	3	...	nt+1
in code	ρ_0	$\xrightarrow{v_0}$ ρ_1	$\xrightarrow{v_1}$ ρ_2	$\xrightarrow{v_2}$ ...	$\xrightarrow{v_{m-1}}$ ρ_m

S = dTrilinear(rho(:,i-1), ...) → get $S(v_i)$ in note.

rho(:,i) = S · rho(:,i-1) → get $S(v_i)\rho_i = \rho_i^{adv}$ in note.

rho(:,i) = pcg(I − dt * Mdis, rho(:,i))

↓

Solve linear system $(I_n - dtQ)\rho_{i+1} = \rho_i^{adv}$ for ρ_{i+1}.

only return $\begin{pmatrix} \rho_1 \\ \vdots \\ \rho_m \end{pmatrix}$.

To obtain dmk2 $= hd \cdot dt \cdot \underset{1}{[M(v \odot v)]^T} \underset{mn}{J} \underset{3mn}{}$

recall that

$$M(v \odot v) = \underset{1}{\left(\underset{1}{\|v_0^T\|^2}, \cdots, \underset{1}{\|v_{m-1}^T\|^2} \right)}_{mn}$$

$$J = \begin{pmatrix} {}_{n}J^1_{v_0} & & & \\ J^2_{v_0} & J^2_{v_1} & & \\ \vdots & \vdots & \ddots & \\ J^m_{v_0} & J^m_{v_1} & \cdots & J^m_{v_{m-1}} \end{pmatrix}_{mn}^{3mn} \triangleq \begin{pmatrix} \frac{\partial \rho_1}{\partial v} \\ \frac{\partial \rho_2}{\partial v} \\ \vdots \\ \frac{\partial \rho_m}{\partial v} \end{pmatrix}_{n}^{3mn}$$

$$\Rightarrow M(v \odot v) J \triangleq \underset{1}{\left(\underset{1}{D_1}, D_2, \cdots D_m \right)}_{3mn}$$

$$\text{where } D_k = \sum_{i=k}^{m} \underset{1}{\|v_{i-1}^T\|^2} \underset{n}{} J^i_{v_{k-1}} \underset{3n}{}, \quad k = 1, \cdots, m.$$

Thus we have the table

(**)

$$\begin{aligned}
D_m &= \|v_{m-1}^T\|^2 J^m_{v_{m-1}} \\
D_{m-1} &= \|v_{m-1}^T\|^2 J^m_{v_{m-2}} + \|v_{m-2}^T\|^2 J^{m-1}_{v_{m-2}} \\
&\vdots \\
D_2 &= \|v_{m-1}^T\|^2 J^m_{v_1} + \|v_{m-2}^T\|^2 J^{m-1}_{v_1} + \cdots + \|v_1^T\|^2 J^2_{v_1} \\
D_1 &= \|v_{m-1}^T\|^2 J^m_{v_0} + \|v_{m-2}^T\|^2 J^{m-1}_{v_0} + \cdots + \|v_1^T\|^2 J^2_{v_0} + \|v_0^T\|^2 J^1_{v_0}
\end{aligned}$$

E_m E_{m-1} $\Longleftarrow$ E_2 E_1

As long as we can computer $D_1, \cdots, D_m$.

even though $\underset{1}{(}\underset{3n}{D_1} \cdots \underset{}{D_m}\underset{3mn}{)}$ is of dimension $1 \times 3mn$.

it doesn't hurt to compute $\underset{m}{D} \triangleq \begin{pmatrix} D_1 \\ \vdots \\ D_m \end{pmatrix}_{3n}$ of dimension $m \times 3n$.

In observation of the special form of (**),

$$\underset{m}{D}\underset{3n}{} = \underset{m}{E_1}\underset{3n}{} + E_2 + \cdots + E_{m-1} + \underset{m}{E_m}\underset{3n}{}.$$

where $\underset{m}{E_i}\underset{3n}{} = \begin{pmatrix} \|v_{i-1}^T\|^2 J^i_{v_0} \\ \vdots \\ \|v_{i-1}^T\|^2 J^i_{v_{i-1}} \\ 0 \\ \vdots \\ 0 \end{pmatrix}_{3n} = \underset{m}{\operatorname{diag}}\left(\|v_{i-1}^T\|^2, m\right)_{mn} \begin{pmatrix} J^i_{v_0} \\ \vdots \\ J^i_{v_{i-1}} \\ 0 \\ \vdots \\ 0 \end{pmatrix}_{3n}$

$i = 1, \cdots, m.$

In practice, only need to compute those non-zero entries.

$$\Rightarrow \underset{i}{(E_i)}_{\text{first } i \text{ entries}} = \underset{i}{\operatorname{diag}}\left(\|v_{i-1}^T\|^2, i\right)_{in} \left(\frac{\partial p_i}{\partial v_{\text{first } i \text{ entries}}} \right)_{3n} \Big|_{\text{vertical order}}$$

We know that in our model, $nt = m$. $v = \begin{pmatrix} v_0 \\ \vdots \\ v_{m-1} \end{pmatrix}$.

to get $\dfrac{\partial p_i}{\partial v_{\text{first } i \text{ entries}}}$, we can assume in this case,

$\underline{nt = i}$, then $\downarrow$ is the final density function's derivative w.r.t. $v = \begin{pmatrix} v_0 \\ \vdots \\ v_{i-1} \end{pmatrix}$ with other parameters like "dt" unchanged.

Go back to the code,

in math g is of size $1 \times 3mn$.

but numerically we prefer to put g as vertical vector,

So we instead compute g^T.

Then $dmk2 = hd \cdot dt \underset{3mn \; mn}{J^T} \underset{1}{[M(v \odot v)]} = \begin{pmatrix} \underset{3n}{D_1^T} \\ \vdots \\ D_m^T \end{pmatrix}_{3mn \; 1}$

Similarly, we instead compute $\underset{3n}{\left(D_1^T \cdots D_m^T\right)}_m = D^T$.

$\Rightarrow \underset{3n \; m}{D^T} = \underset{3n \; m}{E_1^T} + \cdots + E_m^T$. where (E_i^T) first i entries (horizontal)

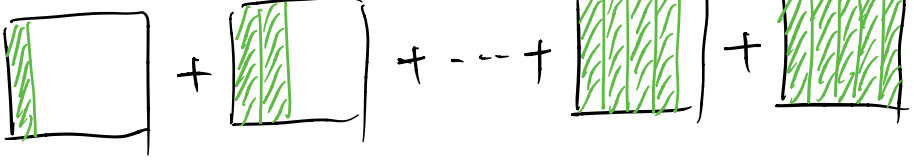
$$= \left(J_{v_0}^{i\,T} \cdots J_{v_{i-1}}^{i\,T}\right) \operatorname{diag}\left(\|v_{i-1}\|^2, i\right)$$

$$= \left(J_{v_0}^{i\,T}\|v_{i-1}\|^2, \cdots, J_{v_{i-1}}^{i\,T}\|v_{i-1}\|^2\right).$$

where $J_{v_j}^k = L^{-1}S(v_{k-1}) \cdots L^{-1}S(v_{j+1})L^{-1}B(p_j)$

In each entry, it is a matrix times a vector.

iteratively

dmk2 = 

So for $j = 1:m$.

$$dmk2(:, 1:j) = dmk2(:, 1:j) + hd \cdot dt \left(\underset{3n}{J_{v_0}^{j\,T}} \underset{n}{\| v_{j-1} \|^2}, \cdots, \underset{1}{J_{v_{j-1}}^{j\,T} \| v_{j-1} \|^2} \right)$$

end

where $\underset{n}{\| v_{j-1} \|^2} = \underset{n}{\left(I_n \,|\, I_n \,|\, I_n \right)} \underset{3n}{\begin{pmatrix} v_{j-1,x}^2 \\ v_{j-1,y}^2 \\ v_{j-1,z}^2 \end{pmatrix}} \underset{1}{} = A \left(v_{j-1} \odot v_{j-1} \right)$.

In `get_drNduT.m`

this function returns flattened

$$\underset{3n}{sensTx} = \underset{3n}{\left(J_{v_0}^{m\,T} \underset{n}{y}, \cdots, J_{v_{m-1}}^{m\,T} \underset{n}{y}\right)_m} = \underset{3mn}{\left(\frac{\partial p_m}{\partial v}\right)^T} \underset{n}{y}$$

$B = I_n - dt\,Q = L$ in note.

$Rho = (p_0\ p_1 \cdots p_m)$ in note.

By the expression of $J_{v_k}^m = L^{-1} S(v_{m-1}) \cdots L^{-1} S(v_{k+1}) L^{-1} B(p_k)$

where $B(p_k) = \frac{\partial}{\partial v_k}(S(v_k)\,p_k)$ (Δ)

we have table

$$\underset{3n}{J_{v_0}^{m\,T}} \underset{n}{y} = \underset{3n}{B(p_0)^T} \underset{n}{(L^{-1})^T} \underset{n}{S(v_1)^T} (L^{-1})^T \cdots S(v_{m-1})^T (L^{-1})^T y$$

$$J_{v_1}^{m\,T} y = B(p_1)^T (L^{-1})^T S(v_2)^T (L^{-1})^T \cdots S(v_{m-1})^T (L^{-1})^T y$$

$$\vdots \qquad\qquad \vdots$$

$$J_{v_{m-3}}^{m\,T} y = B(p_{m-3})^T (L^{-1})^T \underbrace{S(v_{m-2})^T (L^{-1})^T S(v_{m-1})^T (L^{-1})^T y}_{sens}$$

$$J_{v_{m-2}}^{m\,T} y = B(p_{m-2})^T (L^{-1})^T \underbrace{S(v_{m-1})^T (L^{-1})^T y}_{sens} \qquad sensI$$

$$J_{v_{m-1}}^{m\,T} y = B(p_{m-1})^T (L^{-1})^T y \qquad sensI$$

$sens$ $sensI$

It is obvious that as k increases, the expression will simplify. So it is better to compute from the end to the start.

Note that we can compute $B(\rho_k)$ and $S(v_k)$ at the same time in one function dTrilinear.m, which also can be seen from equation (4).

So start from $J_{v_{m-1}}^{mT} y$, we can have both $B(\rho_{m-1})^T (L^{-1})^T y$ and $S(v_{m-1})^T (L^{-1})^T y$, and the latter can be passed to the next loop as the vector.

$dt \cdot [T_x, T_y, T_z] = B(\rho_k)$ in note.
$S = S(v_k)$ in note.

Note that even though we write as L^{-1}, we are not computing the inverse, but by solving linear systems.

in **dTrilinear.m**,

this function returns averaging matrix P and derivatives T_x, T_y, T_z.

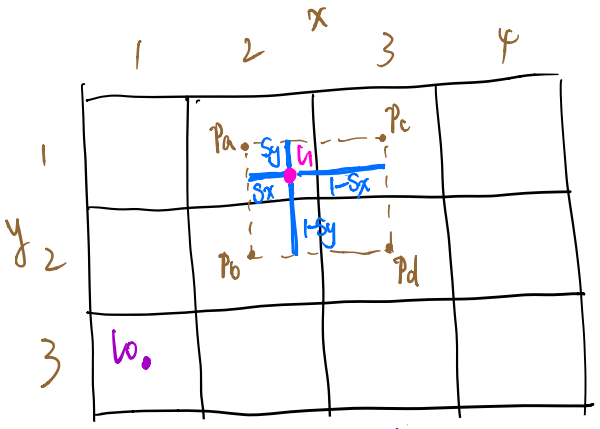
inputs: T — ρ_R in note

x, y, z — new location at next time — old location $+ dt \cdot \hat{v}_R$.

outputs: P — $S(\hat{v}_R)$

$$dt[T_x, T_y, T_z] - B(\rho_R) = \frac{\partial}{\partial \hat{v}_R}\left(S(\hat{v}_R)\rho_R\right).$$

$\rho_R^{adv} = S(\hat{v}_R)\rho_R$. how to get expression of $S(\hat{v}_R)$?



demonstrate in 2d:

l_0 is moved directly to l_1.

mass at l_1 can be split into 4 locations P_a, P_b, P_c, P_d at ratio

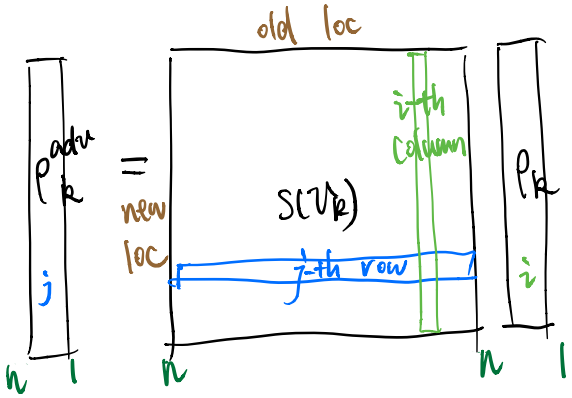
$(1-S_x)(1-S_y)$, $(1-S_x)S_y$,

$S_x(1-S_y)$, S_xS_y, respectively.

where $S_x = x_c + dt \cdot v_{k,x} - [x_c + dt \cdot v_{k,x}]$

$= dt \cdot v_{k,x} - [dt \cdot v_{k,x}], \cdots, \cdots$.

- starting grid point (x_c, y_c, z_c), l_0
- end point $(x_c + dt \cdot v_{k,x}, y_c + dt \cdot v_{k,y}, z_c + dt \cdot v_{k,z})$, l_1
- neighor grid points



in order to get matrix $S(V_k)$. The row-dimension is the new location axis; the column-dimension is the old-location axis.

The i-th column of $S(V_k)$ contains the ratio of splitting of old loction at i into new locations.

e.g. loc i is split at loc $j_1, \cdots, j_4$ with ratio $r_1, \cdots, r_4$.

then $\begin{cases} S(V_k)(i, j_1) = r_1, \\ S(V_k)(i, j_2) = r_2, \\ S(V_k)(i, j_3) = r_3, \\ S(V_k)(i, j_4) = r_4. \end{cases}$

The j-th row of $S(V_k)$ contains where the (mass of) new location j comes from.

In conclusion, $S(V_k)(j, i)$ = the ratio of mass at new location j coming from old location i.

So $S(V_k)$ is only w.r.t V_k, not including ρ_k.

Next we explore the expression of $B(\rho_k) = \frac{\partial}{\partial V_k}(S(V_k)\rho_k)$

according to the expression of ρ_k, it has nothing to do with V_k, then the dericative is actually posed on $S(V_k)$.

First of all, $B(\rho_k) = [B_x(\rho_k), B_y(\rho_k), B_z(\rho_k)]$.

$n \quad 3n \quad n \quad n \quad n \quad n \quad n$

where $B_x(\rho_k) = \frac{\partial}{\partial V_{k,x}}(S(V_k)\rho_k)$, --, ---.

We can just focus on $B_x(\rho_k)$, since the rest dimensions are straightforward if x-dimension is solved.

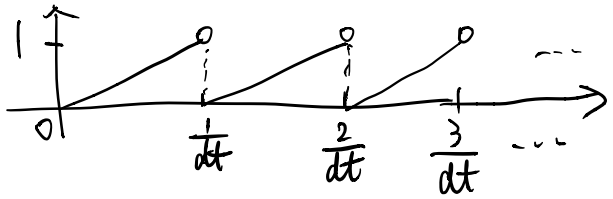
Take 2d as example, recall that the non-zero entries are $(1-S_x)(1-S_y)$, $S_x(1-S_y)$, $(1-S_x)S_y$, $S_x S_y$.

where $S_x = dt \cdot v_{k,x} - [dt \cdot v_{k,x}]$

$S_y = dt \cdot v_{k,y} - [dt \cdot v_{k,y}]$.

$\Rightarrow \frac{\partial S_x}{\partial v_{k,x}} = dt, \frac{\partial S_y}{\partial v_{k,x}} = 0.$

Note, for function $y(x) = dt \cdot x - [dt \cdot x]$



So $y(x)$ is differentiable on $\mathbb{R}/\mathbb{Z} \Rightarrow$ almost everywhere.

$\Rightarrow y'(x) = dt$, a.e.

$\Rightarrow \frac{\partial (1-S_x)(1-S_y)}{\partial v_{k,x}} = -dt(1-S_y), \frac{\partial (1-S_x)S_y}{\partial v_{k,x}} = -dt\, S_y$

$\frac{\partial S_x(1-S_y)}{\partial v_{k,x}} = dt(1-S_y), \frac{\partial S_x S_y}{\partial v_{k,x}} = dt \cdot S_y.$

$\Rightarrow B_x(\rho_k) = \underset{n}{T_x(v_k)}\ \underset{n}{}\ \text{diag}(\rho_k)_n$

where the entries of $T_x(v_k)$ are the entry-wise derivatives w.r.t $v_{k,x}$ of $S(v_k)$.

Similarly to get $T_y(v_k)$, $T_z(v_k)$.

$\Rightarrow \quad B(\rho_k) = dt \cdot [T_x, T_y, T_z]$

↘ ↓ ↓

in code.

in get_drNdu.m,

This function returns

$$\underset{n}{\left(}\underset{n}{J^m_{v_0}}\underset{3n}{,}\ J^m_{v_1},\cdots,J^m_{v_{m-1}}\underset{3mn}{\left.\right)}\underset{3mn}{\underset{1}{x}} = \underset{n}{\frac{\partial \rho_m}{\partial v}}\underset{3mn}{\underset{1}{x}}$$

If we write $x = \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix}$ (each x_i is $3n \times 1$), then the output is

$$(J^m_{v_0},\cdots,J^m_{v_{m-1}})\begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} = \underset{n}{\underset{3n}{\underset{1}{\sum_{i=1}^{m} J^m_{v_{i-1}} x_i}}}$$

$B = I_n - dt\,Q = L$ in note.

rho $= (\rho_0, \rho_1, \cdots, \rho_m)$ in note.

recall that $J^m_{v_R} = L^{-1}S(v_{m-1})\cdots L^{-1}S(v_{R+1})L^{-1}B(\rho_R)$.

where $B(\rho_R) = \frac{\partial}{\partial v_R}(S(v_R)\rho_R)$.

Then we have the table:

$$J^m_{v_0}x_1 = L^{-1}S(v_{m-1})L^{-1}S(v_{m-2})\cdots L^{-1}S(v_2)L^{-1}S(v_1)\underset{n}{L^{-1}B(\rho_0)}\underset{3n}{\underset{1}{x_1}}$$

$$J^m_{v_1}x_2 = L^{-1}S(v_{m-1})L^{-1}S(v_{m-2})\cdots L^{-1}S(v_2)\underset{n}{L^{-1}B(\rho_1)}\underset{3n}{\underset{1}{x_2}}$$

$$\vdots$$

$$J^m_{v_{m-2}}x_{m-1} = L^{-1}S(v_{m-1})\underset{n}{L^{-1}B(\rho_{m-2})}\underset{3n}{\underset{1}{x_{m-1}}}$$

$$J^m_{v_{m-1}}x_m = \underset{n}{L^{-1}}\underset{n}{B(\rho_{m-1})}\underset{3n}{\underset{1}{x_m}}$$

Caculating $B(p_k)$ and $S(p_k)$ can be done in the $(k+1)$-th loop.

So loop from 1 to m.