NUMERICAL SIMULATIONS IN OPTIMAL TRANSPORT
Sam Dulin\textsuperscript{1}, Scott McIntyre\textsuperscript{2}
\textsuperscript{1}University of Virginia, \textsuperscript{2}University of California, Berkeley

Moving Dirt Optimally

Building earthworks for Napoleon’s troops, Gaspard Monge (1781) asked: How to fill a hole with dirt as efficiently as possible?

\( \mu = \) pile of dirt

\( \nu = \) construction to be made from dirt

\( c(x, y) = \) cost to transport one unit of dirt

Specifying \( c(x, y) = |x - y|^2 \), what transportation map will move the dirt from \( \mu \) to \( \nu \) while minimizing the total \( c \)?

Fluid Transport

Benamou & Brenier (2000) translated this into fluid dynamics:

\( \rho(x, t) = \) where the dirt is (distribution) at time \( t \)

\( v(x, t) = \) how the dirt flows (velocity field)

Suppose \( \mu \) flows to \( \nu \) along \( \rho \) being pushed along by \( v \); which \( (\rho, v) \) pair minimizes the cost of transport?

We derive differential equations describing the optimal \( (\rho, v) \):

\[
\begin{align*}
\rho_t + \text{div}(\rho v) &= 0 & \text{“} \rho \text{ pushes dirt along } \rho' \text{”} \\
v_t + (v \cdot \nabla)v &= 0 & \text{“} v \text{ is optimal”}
\end{align*}
\]

Goal: Approximate solutions to this using numerical techniques

Acknowledgements

We would like to thank our mentors Dr. Jianbo Cui, Dr. Luca Dieci, Dr. Haomin Zhou, the Georgia Tech School of Mathematics, and NSF Grant DMS-1851843.

Discretizing Space

Instead of knowing \( \rho \) and \( v \) everywhere, we sample mesh points

infinite-dimensional (PDEs) \rightarrow finite-dimensional (ODEs)

Shooting Differential Equations

Runge-Kutta (RK) methods solve \( y' = f(t, y) \) when we know \( y(0) \).

When we know partial info about \( y(0) \) and \( y(1) \), we take a "shot" at what \( y(0) \) is and use a RK method to see how close it hits \( y(1) \). Finding a shot that hits is a root-finding problem.

Getting Good Guesses

To find a good guess for \( f(r) = 0 \), we can work up to it by solving an easier problem and using that root as the initial guess for a slightly harder problem closer to \( f \).

Examples

Shooting multiple shots throughout the interval improves stability.

Finding Roots

We want to solve \( f(r) = 0 \). If \( x_0 \approx r \), then iterate:

\[ x_{n+1} = x_n - (DF(x_n))^{-1}f(x_n) \]

\( DF \) is the Jacobian (derivative) of \( f \).