Differentiable Fluids with Solid Coupling for Learning and Control – Supplementary Material

Tetsuya Takahashi[*,†], **Junbang Liang**[†], **Yi-Ling Qiao**[†], **Ming C. Lin**[†]
[*]Adobe, [†]University of Maryland at College Park

1 Details of Adjoint Method

Given an objective function $\phi(\mathbf{q}, \mathbf{f})$ with a state vector $\mathbf{q}$ and a control vector $\mathbf{f}$, the goal of the adjoint method is to efficiently compute the gradient of the objective function with respect to the control vector:

$$\frac{d\phi}{d\mathbf{f}} = \frac{\partial\phi}{\partial\mathbf{q}}\frac{d\mathbf{q}}{d\mathbf{f}} + \frac{\partial\phi}{\partial\mathbf{f}}, \tag{1}$$

with the constraint due to the forward simulation:

$$\mathbf{q} = \mathbf{F}(\mathbf{q}, \mathbf{f}). \tag{2}$$

This constraint can be linearized by differentiating (2) with respect to $\mathbf{f}$ to obtain

$$\frac{\partial\mathbf{q}}{\partial\mathbf{f}} = \frac{\partial\mathbf{F}}{\partial\mathbf{q}}\frac{\partial\mathbf{q}}{\partial\mathbf{f}} + \frac{\partial\mathbf{F}}{\partial\mathbf{f}}, \tag{3}$$

which can be rearranged as

$$\left(\mathbf{I} - \frac{\partial\mathbf{F}}{\partial\mathbf{q}}\right)\frac{\partial\mathbf{q}}{\partial\mathbf{f}} = \frac{\partial\mathbf{F}}{\partial\mathbf{f}}. \tag{4}$$

In the gradient formulation (1), direct computation of $\frac{d\mathbf{q}}{d\mathbf{f}}$ is known as extremely expensive since this term requires computing the gradient for each state with respect to each control. The adjoint method avoids this expensive computation by computing $\frac{\partial\phi}{\partial\mathbf{q}}\frac{d\mathbf{q}}{d\mathbf{f}}$ without explicitly forming $\frac{d\mathbf{q}}{d\mathbf{f}}$ as follows.

First, for clarity, we introduce a vector $\mathbf{s} = \left(\frac{\partial\phi}{\partial\mathbf{q}}\right)^T$ and a matrix $\mathbf{Y} = \frac{d\mathbf{q}}{d\mathbf{f}}$ such that $\frac{\partial\phi}{\partial\mathbf{q}}\frac{d\mathbf{q}}{d\mathbf{f}} = \mathbf{s}^T\mathbf{Y}$. If we rewrite (4) as $\mathbf{XYZ}$ with $\mathbf{X} = \left(\mathbf{I} - \frac{\partial\mathbf{F}}{\partial\mathbf{q}}\right)$ and $\mathbf{Z} = \frac{\partial\mathbf{F}}{\partial\mathbf{f}}$, and introduce the adjoint vector $\mathbf{r}$ such that $\mathbf{X}^T\mathbf{r} = \mathbf{s}$, we obtain $\mathbf{s}^T\mathbf{Y} = \mathbf{r}^T\mathbf{XY} = \mathbf{r}^T\mathbf{Z}$, i.e.,

$$\frac{\partial\phi}{\partial\mathbf{q}}\frac{d\mathbf{q}}{d\mathbf{f}} = \mathbf{r}^T\frac{\partial\mathbf{F}}{\partial\mathbf{f}}, \tag{5}$$

with the constraint ($\mathbf{X}^T\mathbf{r} = \mathbf{s}$):

$$\left(\mathbf{I} - \frac{\partial\mathbf{F}}{\partial\mathbf{q}}\right)^T\mathbf{r} = \left(\frac{\partial\phi}{\partial\mathbf{q}}\right)^T, \tag{6}$$

Copyright © 2021, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

which can be reformulated as

$$\mathbf{r} = \left(\frac{\partial\mathbf{F}}{\partial\mathbf{q}}\right)^T\mathbf{r} + \left(\frac{\partial\phi}{\partial\mathbf{q}}\right)^T. \tag{7}$$

The key for the efficiency of the adjoint method comes from the *sparse* structure of $\frac{\partial\mathbf{F}}{\partial\mathbf{f}}$ and $\frac{\partial\mathbf{F}}{\partial\mathbf{q}}$. Since the forward simulation typically involves control at simulation step i ($\mathbf{f}_i$, where i denotes simulation step index), and consecutive states ($\mathbf{q}_i$ and $\mathbf{q}_{i+1}$) only, we can efficiently compute $\frac{\partial\mathbf{F}_i}{\partial\mathbf{f}_i}$, $\frac{\partial\mathbf{F}_i}{\partial\mathbf{q}_i}$, and adjoint states with the recursive relation:

$$\mathbf{r}_i = \left(\frac{\partial\mathbf{F}_i}{\partial\mathbf{q}_i}\right)^T\mathbf{r}_{i+1} + \left(\frac{\partial\phi}{\partial\mathbf{q}_i}\right)^T, \tag{8}$$

where $\mathbf{r}_n = \left(\frac{\partial\phi}{\partial\mathbf{q}_n}\right)^T$. These reformulations finally allow for efficient computations of $\mathbf{r}^T\frac{\partial\mathbf{F}}{\partial\mathbf{f}}$ and thus the gradient $\frac{d\phi}{d\mathbf{f}}$.