Supplemental: Curl-Flow: Boundary-Respecting Pointwise Incompressible Velocity Interpolation for Grid-Based Fluids

1 Bilinear velocity interpolation is divergent

To see the problem with standard interpolants, define a linear interpolation function $lerp(a, b, t) = (1 - t)a + tb$ and construct the bilinearly interpolated velocity at a point $P = (x, y)$ (see Figure 1) as

$$u(x, y) = lerp(lerp(u_0, u_1, \alpha_u(x)), lerp(u_2, u_3, \alpha_u(x)), \beta_u(y))$$
$$v(x, y) = lerp(lerp(v_0, v_1, \alpha_v(x)), lerp(v_2, v_3, \alpha_v(x)), \beta_v(y))$$

where the α and β functions return edge fractions ($0 \leq \alpha, \beta \leq 1$) for the data indicated by their subscripts. This interpolated velocity is not analytically divergence-free in general:

$$\nabla \cdot \mathbf{u} = \frac{lerp(u_1 - u_0, u_3 - u_2, \beta_u(y)) + lerp(v_2 - v_0, v_3 - v_1, \alpha_v(x))}{h}$$
$$\neq 0$$

The grid cell width h appears due to the derivatives of α and β.

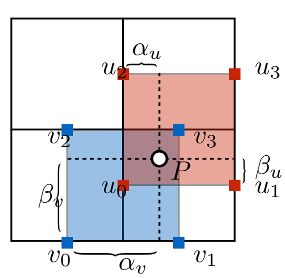

Figure 1: **Direct Velocity Interpolation:** To compute a pointwise velocity at point $P = (x, y)$ in a staggered velocity grid, one (bilinearly) interpolates the nearby velocity samples component by component.

2 Convergence table of Figure 25.

Table 1: Data for convergence test of Figure 25: error and the order of convergence (within parentheses) are measured using a L_∞ norm under grid refinement. Both direct (bilinear) velocity interpolation and Curl-Flow interpolation yield similar convergence.

Grid Resolution	16^2	32^2	64^2	128^2	256^2	512^2
Direct Velocity Interpolation (L_∞)	3.733e-02 (–)	9.554e-03 (1.966)	2.396e-03 (1.996)	5.927e-04 (2.015)	1.412e-04 (2.070)	2.824e-05 (2.322)
Curl-Flow (L_∞)	3.787e-02 (–)	1.073e-02 (1.819)	4.789e-03 (1.165)	1.845e-03 (1.376)	5.050e-04 (1.870)	4.183e-05 (3.594)