

A Very Brief Introduction into Nodal-Discontinuous Galerkin Methods

Alex Alekseenko

Department of Mathematics,
California State University, Northridge

February 25, 2019

Discontinuous Galerkin Approximations

Galerkin discretizations of functions of single variable

Let $\phi_1(v), \dots, \phi_s(v)$, be linearly independent on $[a, b]$.

Galerkin approximation:

$$f(\vec{v}) \approx \sum_i f_i \phi_i(v).$$

To find f_i , we choose $\psi_i(v)$, $i = 1, \dots, s$ and consider s equations

$$\int_a^b f(v) \psi_j(v) dv = \int_a^b \sum_i f_i \phi_i(v) \psi_j(v) dv = \sum_i f_i \int_a^b \phi_i(v) \psi_j(v) dv$$

Define $b_i = \int_a^b f \psi_j dv$, $D_{ji} = \int_a^b \phi_i \psi_j dv$. Solve

$$b = Df$$

If $\psi = \phi$ the approximation is referred as Galerkin, if $\psi \neq \phi$ the approximation is referred as Petrov-Galerkin.

Galerkin discretizations of functions of single variable

Let $\phi_1(v), \dots, \phi_s(v)$, be linearly independent on $[a, b]$.

Galerkin approximation:

$$f(\vec{v}) \approx \sum_i f_i \phi_i(v).$$

To find f_i , we choose $\psi_i(v)$, $i = 1, \dots, s$ and consider s equations

$$\int_a^b f(v) \psi_j(v) dv = \int_a^b \sum_i f_i \phi_i(v) \psi_j(v) dv = \sum_i f_i \int_a^b \phi_i(v) \psi_j(v) dv$$

Define $b_i = \int_a^b f \psi_j dv$, $D_{ji} = \int_a^b \phi_i \psi_j dv$. Solve

$$b = Df$$

If $\psi = \phi$ the approximation is referred as Galerkin, if $\psi \neq \phi$ the approximation is referred as Petrov-Galerkin.

Galerkin discretizations of functions of single variable

Let $\phi_1(v), \dots, \phi_s(v)$, be linearly independent on $[a, b]$.

Galerkin approximation:

$$f(\vec{v}) \approx \sum_i f_i \phi_i(v).$$

To find f_i , we choose $\psi_i(v)$, $i = 1, \dots, s$ and consider s equations

$$\int_a^b f(v) \psi_j(v) dv = \int_a^b \sum_i f_i \phi_i(v) \psi_j(v) dv = \sum_i f_i \int_a^b \phi_i(v) \psi_j(v) dv$$

Define $b_i = \int_a^b f \psi_i dv$, $D_{ji} = \int_a^b \phi_i \psi_j dv$. Solve

$$b = Df$$

If $\psi = \phi$ the approximation is referred as Galerkin, if $\psi \neq \phi$ the approximation is referred as Petrov-Galerkin.

Galerkin discretizations, several variables

Let $\phi_1(\vec{v}), \dots, \phi_s(\vec{v})$, be linearly independent.

Galerkin approximation:

$$f(t, \vec{x}, \vec{v}) = \sum_i f_i(t, \vec{x}) \phi_i(\vec{v}) + R(t, \vec{x}, \vec{v})$$

the coefficients are defined by requiring that $\langle \psi_i, R \rangle_{L^2(V)} = 0$, for some $\psi_i(\vec{v})$, $i = 1, \dots, s$. By taking $L^2(V)$ scalar product with $\psi_j(\vec{v})$ obtain

$$\langle f, \psi_j \rangle_{L^2(V)} = \sum_i f_i \langle \phi_i, \psi_j \rangle_{L^2(V)}$$

and coefficients f_i are determined by inverting the matrix

$D_{ij} = \langle \phi_i, \psi_j \rangle_{L^2(V)}$. (Usually, $\psi = \phi$, orthogonal, but not required)

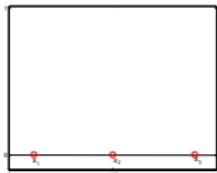
Governing equations are also projected on $\psi_j(\vec{v})$. Integration by parts may be used on terms with derivatives. Discretizations of derivatives may be introduced. (Actually, almost always they are)

Grid vs. Galerkin approximations

Consider the problem of approximating $\sin(x)$ on $[0, \pi]$:

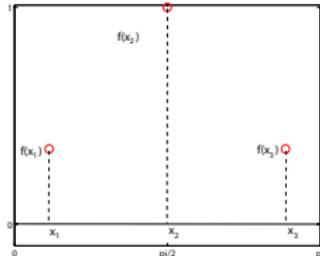
Grid approximation

Choose grid points x_i on $[0, \pi]$:



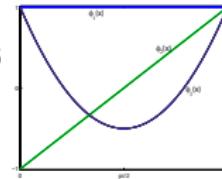
The function is approximated by

$$f_i = f(x_i), \text{ at } x_i, i = 1, 2, 3$$



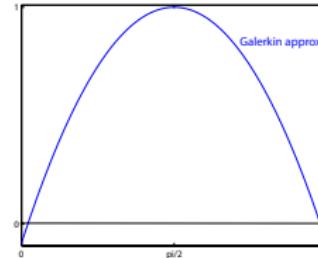
Galerkin approximation

Choose basis functions $\phi_i(x)$ on $[0, \pi]$:



The function is approximated by

$$\hat{f}(x) = \sum_{i=1}^3 \alpha_i \phi_i(x)$$



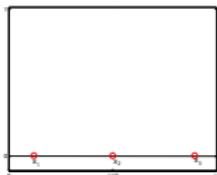
Distinction vanishes in nodal-Galerkin approaches ($\alpha_i \neq f(x_i)$)!

Grid vs. Galerkin approximations

Consider the problem of approximating $\sin(x)$ on $[0, \pi]$:

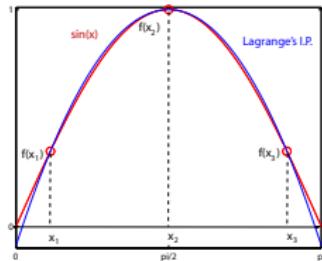
Grid approximation

Choose grid points x_i on $[0, \pi]$:



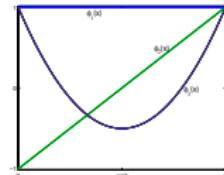
The function is approximated by

$$f_i = f(x_i), \text{ at } x_i, i = 1, 2, 3$$



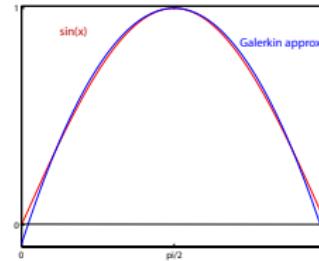
Galerkin approximation

Choose basis functions $\phi_i(x)$ on $[0, \pi]$:



The function is approximated by

$$\hat{f}(x) = \sum_{i=1}^3 \alpha_i \phi_i(x)$$



Distinction vanishes in nodal-Galerkin approaches ($\alpha_i \neq f(x_i)$)!

Galerkin Approximations

Steps to construct Galerkin discretizations

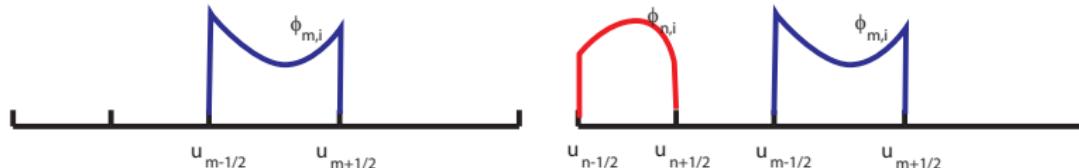
- Select Galerkin space by selecting the Galerkin basis.
- Substitute the Galerkin approximation into the problem.
- Take the scalar product of each equation with a test function.
- Derive the **weak form** of the equations. in the case of DEs, integrate by parts.
- Deduce system for the Galerkin coefficients

Why using Galerkin approximations?

- they are backed by powerful mathematics
- allow high order approximations
- have inherent mechanisms for extrapolation
- allow adaptive refinement
- may allow economic approximation using specialized basis functions
- can be converted into grid methods (nodal-Galerkin approach)

Discontinuous Galerkin Methods

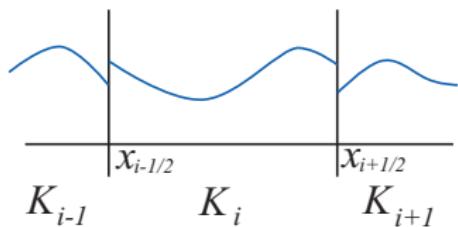
The interval U is partitioned into $U_m = [u_{m-1/2}, u_{m+1/2}]$. As a rule, $\varphi_{i,m}(u)$ are supported in U_m .



As a rule $\varphi_{i,m}(u)$ are orthogonal on U_m so $\int_U \varphi_{i,m} \varphi_{j,n} du = \delta_{mn} \delta_{ij}$. The solution is approximated by a discontinuous function

$$f(\vec{u}) \Big|_{U_m} \approx \sum_{i=1}^k \hat{f}_{i,m} \varphi_{i,m}(\vec{u})$$

Values of f on the boundaries of U_m are approximated by a *numerical flux*. Approximation of integrals of f is straightforward. Derivatives of f are approximated in a weak form using numerical flux (Cockburn, 1999).

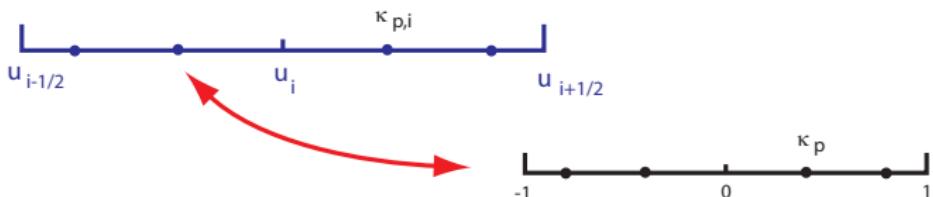


Nodal Discontinuous Galerkin Discretizations in the Velocity Variable

Nodal-Galerkin Basis Functions

(Hesthaven and Warburton, 2007) Let $U_i = [u_{i-1/2}, u_{i+1/2}]$,
 $\Delta u_i = u_{i+1/2} - u_{i-1/2}$, $u_i = (u_{i+1/2} + u_{i-1/2})/2$. Let κ_p , $p = 1, \dots, s$ be
the nodes of Legendre's quadrature of order $2s - 1$ and

$$\kappa_{p,i} = \frac{\Delta u_i}{2} \kappa_p + u_i \quad \text{or} \quad \kappa_p = \frac{2(\kappa_{p,i} - u_i)}{\Delta u_i}$$



Define basis functions

$$\lambda_{p,i}(u) = \frac{(u - \kappa_{1,i}) \cdot \dots \cdot (u - \kappa_{p-1,i})(u - \kappa_{p+1,i}) \cdot \dots \cdot (u - \kappa_{s,i})}{(\kappa_{p,i} - \kappa_{1,i}) \cdot \dots \cdot (\kappa_{p,i} - \kappa_{p-1,i})(\kappa_{p,i} - \kappa_{p+1,i}) \cdot \dots \cdot (\kappa_{p,i} - \kappa_{s,i})}$$

Basis Functions cont.

Define the basis functions

$$\lambda_{p,i}(u) = \frac{(u - \kappa_{1,i}) \cdot \dots \cdot (u - \kappa_{p-1,i})(u - \kappa_{p+1,i}) \cdot \dots \cdot (u - \kappa_{s,i})}{(\kappa_{p,i} - \kappa_{1,i}) \cdot \dots \cdot (\kappa_{p,i} - \kappa_{p-1,i})(\kappa_{p,i} - \kappa_{p+1,i}) \cdot \dots \cdot (\kappa_{p,i} - \kappa_{s,i})}$$

Recall that

$$\lambda_{p,i}(\kappa_{q,i}) = \begin{cases} 1, & \text{if } p = q \\ 0, & \text{if } p \neq q \end{cases}$$

The basis functions are orthogonal (recall that the Gauss quadrature with s nodes is exact on polynomials of degree $2s - 1$):

$$\int_{U_i} \lambda_{p,i}(u) \lambda_{p,i}(u) du = \sum_{r=1}^s w_r \lambda_{p,i}(\kappa_{r,i}) \lambda_{q,i}(\kappa_{r,i}) = \begin{cases} w_p, & \text{if } p = q \\ 0, & \text{if } p \neq q \end{cases}$$

where w_r are the Gauss quadrature weights.

Basis Functions cont.

The basis functions are orthogonal (recall that the Gauss quadrature with s nodes is exact on polynomials of degree $2s - 1$):

$$\int_{U_i} \lambda_{p,i}(u) \lambda_{p,i}(u) du = \sum_{r=1}^s w_r \lambda_{p,i}(\kappa_{r,i}) \lambda_{q,i}(\kappa_{r,i}) = \begin{cases} w_p, & \text{if } p = q \\ 0, & \text{if } p \neq q \end{cases}$$

More importantly,

$$\int_{U_i} u \lambda_{p,i}(u) \lambda_{p,i}(u) du = \sum_{r=1}^s w_r \kappa_{r,i} \lambda_{p,i}(\kappa_{r,i}) \lambda_{q,i}(\kappa_{r,i}) = \begin{cases} w_p \kappa_{r,i}, & \text{if } p = q \\ 0, & \text{if } p \neq q \end{cases}$$

In two dimensions and three dimensions, use

$$\lambda_{p,i}(u) \lambda_{q,j}(v) \quad \text{and} \quad \lambda_{p,i}(u) \lambda_{q,j}(v) \lambda_{r,l}(w)$$

Galerkin Coefficients in Nodal-DG Methods

The DG approximation reduces to the approximation by the Lagrange interpolating polynomial of degree $s - 1$ on nodes $\kappa_{p,i}$:

$$f_{p,i}(t, \vec{x}) = f(t, \vec{x}, \kappa_{p,i})!$$

Compare this to evaluating and solving the system

$$\int_{I_i} f(v) \psi_{j,i}(v) dv = \sum_l f_{l,i} D_{jl}$$

where

$$D_{jl} = \int_{I_i} \phi_{j,i}(v), \psi_{l,i}(v) dv$$

Matrix D may not be diagonal.

Kinetic Equations

The Boltzmann Equation

The evolution of the velocity distribution function is governed by the Boltzmann equation:

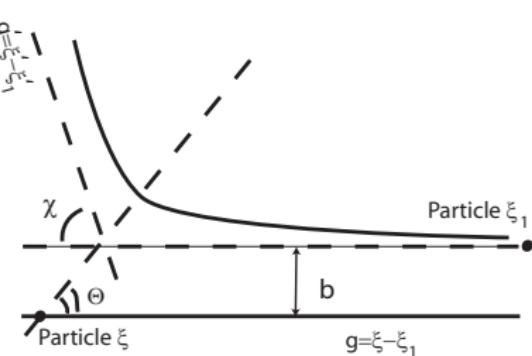
$$\frac{\partial}{\partial t} f(t, \vec{x}, \vec{v}) + \vec{v} \cdot \vec{\nabla}_x f(t, \vec{x}, \vec{v}) = Q(f, f),$$

where (no external forces, binary collisions, single species)

$$Q(f, f) = \int_{R^3} \int_0^{2\pi} \int_0^{b_0} (f' f'_1 - f f_1) |g| b \, db \, d\varepsilon \, dv_1$$

Where $f = (t, \vec{x}, \vec{v})$, $f = (t, \vec{x}, \vec{v}_1)$,
 $f' = (t, \vec{x}, \vec{v}')$, and \vec{v} and \vec{v}_1 are pre-collisional and \vec{v}' and \vec{v}'_1 are post collisional velocities.

Five dimensional integration requires $O(n^5)$ operations at each point of six dimensional phase space where n is the number of D.O.F. in one dimension.



The Model Equations

In the regimes when gas is close to continuum a good approximation to the Boltzmann equation is given by model equations where molecular collisions are modelled by relaxation:

$$\frac{\partial}{\partial t} f(t, \vec{x}, \vec{v}) + \vec{v} \cdot \vec{\nabla}_x f(t, \vec{x}, \vec{v}) = \nu(f_0 - f(t, \vec{x}, \vec{v})),$$

where f_0 is either Maxwellian (Bhatnagar-Gross-Krook, 1954) or a Gaussian (ellipsoidal-statistical BGK, Holway 1966) distribution function or some other distribution function.

ν is called the collision frequency and can be large ($\nu \approx 1/\text{Kn}$).

Because model equations depend only on a few macroparameters, their evaluation requires $O(n^3)$ operations at each point of three dimensional physical space

Nodal DG Formulation

If the basis functions are chosen as above. Let velocity cells and velocity ordinates within a cell be numbered by two indices say $(\kappa_{p,i}^u, \kappa_{p,i}^v, \kappa_{p,i}^w)$ is a node number p on cell i . Then the discrete velocity Boltzmann equation is given by

$$\begin{aligned}\partial_t f_{p,i}(t, \vec{x}) + \kappa_{p,i}^u \partial_x f_{p,i}(t, \vec{x}) + \kappa_{p,i}^v \partial_y f_{p,i}(t, \vec{x}) + \kappa_{p,i}^w \partial_z f_{p,i}(t, \vec{x}) \\ = \int_{V_i} \varphi_{p,i} Q(f, f) \quad (1)\end{aligned}$$

Similarly, nodal DG discretization of a model equation takes form:

$$\begin{aligned}\partial_t f_{p,i}(t, \vec{x}) + \kappa_{p,i}^u \partial_x f_{p,i}(t, \vec{x}) + \kappa_{p,i}^v \partial_y f_{p,i}(t, \vec{x}) \\ + \kappa_{p,i}^w \partial_z f_{p,i}(t, \vec{x}) = \nu(t, \vec{x}) (f_0(\kappa_{p,i}^u, \kappa_{q,i}^v, \kappa_{r,i}^w) - f_{p,i}(t, \vec{x})) \quad (2)\end{aligned}$$

DG Discretizations in the Velocity Variable

(AGG, 2010) Partition the velocity space (or a bounded subset) by V_i , $i = 1, \dots, M$. On each V_i introduce $\lambda_{l,i}(\bar{u})$ (e.g., polynomials). Seek the solution in the form

$$f(t, \bar{x}, \bar{u})|_{V_i} = \sum_l f_{l,i}(t, \bar{x}) \lambda_{l,i}(\bar{u})$$

Substitute into the kinetic eqn., multiply by a basis function, integrate:

$$\mathbf{D}_i \partial_t \mathbf{f}_i(t, \bar{x}) + \partial_x \mathbf{T}_i^u \mathbf{f}_i(t, \bar{x}) + \partial_y \mathbf{T}_i^v \mathbf{f}_i(t, \bar{x}) + \partial_z \mathbf{T}_i^w \mathbf{f}_i(t, \bar{x}) = \int_{V_i} Q(f, f) \lambda_{m,i}.$$

where

$$T_{ml,i}^u = \int_{V_i} u \lambda_{l,i} \lambda_{m,i}, \quad T_{ml,i}^v = \int_{V_i} v \lambda_{l,i} \lambda_{m,i}, \quad T_{ml,i}^w = \int_{V_i} w \lambda_{l,i} \lambda_{m,i},$$
$$D_{ml,i} = \int_{V_i} \lambda_{l,i} \lambda_{m,i}.$$

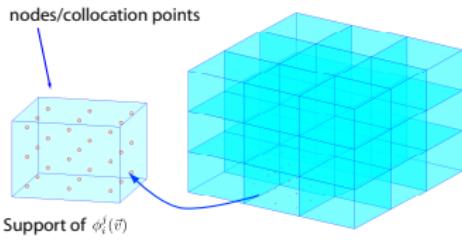
For Nodal-DG bases, matrices $T_{ml,i}^*$ are diagonal!

Discontinuous Galerkin Discretizations of the Collision Operator

Nodal DG Discretizations in 3D Velocity Variable

A rectangular velocity domain is partitioned into $K_i = U_j \times V_k \times W_l$. (Hesthaven and Warburton, 2007) Let $\kappa_{p,i}$, $p = 1, \dots, s$ be the nodes of Legendre's quadrature on U_i . Define

$$\lambda_{p,i}(u) = \prod_{\substack{r=1,s \\ r \neq p}} \frac{u - \kappa_{r,i}}{\kappa_{p,i} - \kappa_{r,i}}$$



In three dimensions use $\phi_{j,i}(\vec{u}) = \lambda_{p,j}(u)\lambda_{q,k}(v)\lambda_{r,l}(w)$, j runs over all (p, q, r) . Up to truncation errors of integration, the Galerkin coefficients are the nodal values of the function .

$$f(u)|_{K_i} = \sum_j f_{j,i} \phi_{j,i}(\vec{u}), \quad \text{where} \quad f_{j,i} = f(\kappa_{p,i}, \kappa_{q,i}, \kappa_{r,i}).$$

Nodal-DG discretizations are obtained by substituting the Galerkin approximation into the equation, multiplying by a basis function and integrating over the velocity domain:

$$\partial_t f_{p,i}(t, \vec{x}) + \kappa_{p,i}^u \partial_x f_{p,i}(t, \vec{x}) + \kappa_{p,i}^v \partial_y f_{p,i}(t, \vec{x}) + \kappa_{p,i}^w \partial_z f_{p,i}(t, \vec{x}) = \int_{V_i} \varphi_{p,i} Q(f, f)$$

The Bilinear Form of a Galerkin Projection

Consider a moment projection with a kernel $\varphi(\vec{\xi})$

$$I_\varphi = \int_{R^3} \varphi(\vec{\xi}) \int_{R^3} \int_0^{2\pi} \int_0^{b_*} (f(\vec{\xi}') f(\vec{\xi}_1') - f(\vec{\xi}) f(\vec{\xi}_1)) |g| b db d\varepsilon d\vec{\xi}_1 d\vec{\xi}$$

Replace it with (e.g., Kogan, 1995)

$$I_\varphi = \int_{R^3} \int_{R^3} \int_0^{2\pi} \int_0^{b_*} \frac{|g|}{2} f(t, \vec{x}, \vec{\xi}) f(t, \vec{x}, \vec{\xi}_1) (\varphi(\vec{\xi}') + \varphi(\vec{\xi}_1') - \varphi(\vec{\xi}) - \varphi(\vec{\xi}_1)) b db d\varepsilon d\vec{\xi}_1 d\vec{\xi}$$

From first principles of rarefied gas dynamics it follows that the last equation can be re-written as

$$I_\varphi = \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{\xi}) f(t, \vec{x}, \vec{\xi}_1) A(\vec{\xi}, \vec{\xi}_1; \varphi) d\vec{\xi}_1 d\vec{\xi}$$

where information about molecular collisions is encoded in

$$A(\vec{\xi}, \vec{\xi}_1; \varphi) = \frac{|g|}{2} \int_0^{2\pi} \int_0^{b_*} (\varphi(\vec{\xi}') + \varphi(\vec{\xi}_1') - \varphi(\vec{\xi}) - \varphi(\vec{\xi}_1)) b db d\varepsilon.$$

The Development of Fast Methods for BE

$$I_\varphi = \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{\xi}) f(t, \vec{x}, \vec{\xi}_1) A(\vec{\xi}, \vec{\xi}_1; \varphi) d\vec{\xi}_1 d\vec{\xi}$$

where information about molecular collisions is encoded in

$$A(\vec{\xi}, \vec{\xi}_1; \varphi) = \frac{|g|}{2} \int_0^{2\pi} \int_0^{b_*} (\varphi(\vec{\xi}') + \varphi(\vec{\xi}_1') - \varphi(\vec{\xi}) - \varphi(\vec{\xi}_1)) b db d\varepsilon.$$

Challenges:

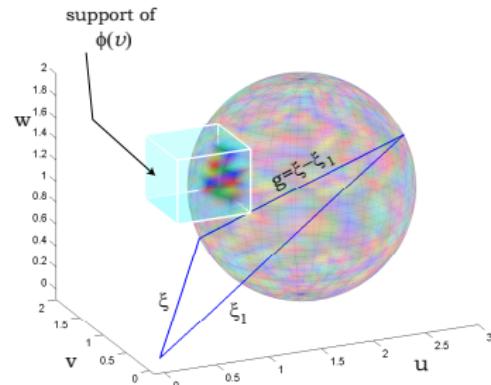
- Evaluating six dimensional integral for every basis function leads to $O(n^9)$ operations, where n is the number of velocity points in one velocity dimension. Want to reduce complexity to $O(n^6)$ at the least, ideally $O(n^3 \log n)$ and faster methods (with controlled accuracy).
- Storing $A(\vec{\xi}, \vec{\xi}_1; \varphi)$ for each basis function $\varphi(\vec{u})_{ij}$ requires $O(n^9)$ memory units. Want to reduce to at least $O(n^6)$, or less.

Collision Kernel. Shift Invariance

(Alekseenko and Josyula (2012), Also see Grohs, Hiptmair and Pintarelli (2015))

$$A(\vec{\xi}, \vec{\xi}_1; \varphi) = |g|^\alpha \int_{\mathbb{S}^2} \phi_{i;j}(\vec{v}') b_\alpha(\theta) d\sigma.$$

Depends on φ and the collision model, information on collisions



Theorem. $A(\vec{v}, \vec{v}_1; \varphi)$ is invariant with respect to a shift, namely

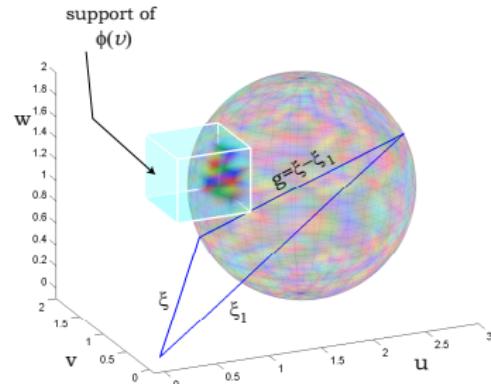
$$A(\vec{v} + \vec{\eta}, \vec{v}_1 + \vec{\eta}; \varphi(\vec{v} - \vec{\eta})) = A(\vec{v}, \vec{v}_1; \varphi)$$

Collision Kernel. Shift Invariance

(Alekseenko and Josyula (2012), Also see Grohs, Hiptmair and Pintarelli (2015))

$$A(\vec{\xi}, \vec{\xi}_1; \varphi) = |g|^\alpha \int_{\mathbb{S}^2} \phi_{i;j}(\vec{v}') b_\alpha(\theta) d\sigma.$$

Depends on φ and the collision model, information on collisions



Theorem. $A(\vec{v}, \vec{v}_1; \varphi)$ is invariant with respect to a shift, namely

$$A(\vec{v} + \vec{\eta}, \vec{v}_1 + \vec{\eta}; \varphi(\vec{v} - \vec{\eta})) = A(\vec{v}, \vec{v}_1; \varphi)$$

Collision Information Mapping. Scaling Invariance

Consider substitution $\hat{\vec{v}} = \vec{v}/C_\infty$ in the collision kernel for HS, VHS, L-J potentials

$$\begin{aligned} A(\vec{v}/C, \vec{v}_1/C; \phi_{i;j}(C\vec{v})) &= \frac{|g|^\alpha}{C^\alpha} \int_{\mathbb{S}^2} \phi_{i;j}(C(\vec{v}'/C)) b_\alpha(\theta) d\sigma \\ &= \frac{|g|^\alpha}{C^\alpha} \int_{\mathbb{S}^2} \phi_{i;j}(\vec{v}') b_\alpha(\theta) d\sigma \\ &= \frac{1}{C^\alpha} A(\vec{v}, \vec{v}_1; \phi_{i;j}). \end{aligned}$$

Theorem. For $b(\theta, \varepsilon)|g|^\alpha$ potentials, $A(\vec{v}, \vec{v}_1; \varphi)$ is invariant with respect to isotropic scaling, specifically

$$A(v, v_1; \phi(u)) := \frac{1}{C_\infty} A(v C_\infty, v_1 C_\infty; \hat{\phi}(u/C_\infty)). \quad (3)$$

Collision Information Mapping. Scaling Invariance

Consider substitution $\hat{\vec{v}} = \vec{v}/C_\infty$ in the collision kernel for HS, VHS, L-J potentials

$$\begin{aligned} A(\vec{v}/C, \vec{v}_1/C; \phi_{i;j}(C\vec{v})) &= \frac{|g|^\alpha}{C^\alpha} \int_{\mathbb{S}^2} \phi_{i;j}(C(\vec{v}'/C)) b_\alpha(\theta) d\sigma \\ &= \frac{|g|^\alpha}{C^\alpha} \int_{\mathbb{S}^2} \phi_{i;j}(\vec{v}') b_\alpha(\theta) d\sigma \\ &= \frac{1}{C^\alpha} A(\vec{v}, \vec{v}_1; \phi_{i;j}). \end{aligned}$$

Theorem. For $b(\theta, \varepsilon)|g|^\alpha$ potentials, $A(\vec{v}, \vec{v}_1; \varphi)$ is invariant with respect to isotropic scaling, specifically

$$A(v, v_1; \phi(u)) := \frac{1}{C_\infty} A(v C_\infty, v_1 C_\infty; \hat{\phi}(u/C_\infty)). \quad (3)$$

Convolution Formulation of DG Galerkin Projection

Theorem. $A(\vec{v}, \vec{v}_1; \varphi)$ is invariant with respect to a shift, specifically

$$A(\vec{v} + \vec{\eta}, \vec{v}_1 + \vec{\eta}; \varphi(\vec{u} - \vec{\eta})) = A(\vec{v}, \vec{v}_1; \varphi)$$

In the case of **uniform** DG discretizations, $\varphi^j(\vec{w}) = \varphi(\vec{w} + \vec{\xi}_j)$. Then,

$$\begin{aligned} I_{\varphi^j} &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v}, \vec{v}_1; \varphi^j) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v} + \vec{\xi}_j, \vec{v}_1 + \vec{\xi}_j; \varphi^j(\vec{u} - \vec{\xi}_j)) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v} + \vec{\xi}_j, \vec{v}_1 + \vec{\xi}_j; \varphi) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{w} - \vec{\xi}_j) f(t, \vec{x}, \vec{v}_1 - \vec{\xi}_j) A(\vec{w}, \vec{w}_1; \varphi) d\vec{w}_1 d\vec{w} \end{aligned}$$

A.A., T. Nguyen, and A. Wood (2018) introduce

$$I(\vec{\xi}) = \int_{R^3} \int_{R^3} f(\vec{v} - \vec{\xi}) f(\vec{v}_1 - \vec{\xi}) A(\vec{v}, \vec{v}_1; \varphi) d\vec{v}_1 d\vec{v}$$

Observe that on uniform grid $I_{\varphi^j} = I(\vec{\xi}_j)$.

Convolution Formulation of DG Galerkin Projection

Theorem. $A(\vec{v}, \vec{v}_1; \varphi)$ is invariant with respect to a shift, specifically

$$A(\vec{v} + \vec{\eta}, \vec{v}_1 + \vec{\eta}; \varphi(\vec{u} - \vec{\eta})) = A(\vec{v}, \vec{v}_1; \varphi)$$

In the case of **uniform** DG discretizations, $\varphi^j(\vec{w}) = \varphi(\vec{w} + \vec{\xi}_j)$. Then,

$$\begin{aligned} I_{\varphi^j} &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v}, \vec{v}_1; \varphi^j) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v} + \vec{\xi}_j, \vec{v}_1 + \vec{\xi}_j; \varphi^j(\vec{u} - \vec{\xi}_j)) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v} + \vec{\xi}_j, \vec{v}_1 + \vec{\xi}_j; \varphi) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{w} - \vec{\xi}_j) f(t, \vec{x}, \vec{v}_1 - \vec{\xi}_j) A(\vec{w}, \vec{w}_1; \varphi) d\vec{w}_1 d\vec{w} \end{aligned}$$

A.A., T. Nguyen, and A. Wood (2018) introduce

$$I(\vec{\xi}) = \int_{R^3} \int_{R^3} f(\vec{v} - \vec{\xi}) f(\vec{v}_1 - \vec{\xi}) A(\vec{v}, \vec{v}_1; \varphi) d\vec{v}_1 d\vec{v}$$

Observe that on uniform grid $I_{\varphi^j} = I(\vec{\xi}_j)$.

Convolution Formulation of DG Galerkin Projection

Theorem. $A(\vec{v}, \vec{v}_1; \varphi)$ is invariant with respect to a shift, specifically

$$A(\vec{v} + \vec{\eta}, \vec{v}_1 + \vec{\eta}; \varphi(\vec{u} - \vec{\eta})) = A(\vec{v}, \vec{v}_1; \varphi)$$

In the case of **uniform** DG discretizations, $\varphi^j(\vec{w}) = \varphi(\vec{w} + \vec{\xi}_j)$. Then,

$$\begin{aligned} I_{\varphi^j} &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v}, \vec{v}_1; \varphi^j) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v} + \vec{\xi}_j, \vec{v}_1 + \vec{\xi}_j; \varphi^j(\vec{u} - \vec{\xi}_j)) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v} + \vec{\xi}_j, \vec{v}_1 + \vec{\xi}_j; \varphi) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{w} - \vec{\xi}_j) f(t, \vec{x}, \vec{v}_1 - \vec{\xi}_j) A(\vec{w}, \vec{w}_1; \varphi) d\vec{w}_1 d\vec{w} \end{aligned}$$

A.A., T. Nguyen, and A. Wood (2018) introduce

$$I(\vec{\xi}) = \int_{R^3} \int_{R^3} f(\vec{v} - \vec{\xi}) f(\vec{v}_1 - \vec{\xi}) A(\vec{v}, \vec{v}_1; \varphi) d\vec{v}_1 d\vec{v}$$

Observe that on uniform grid $I_{\varphi^j} = I(\vec{\xi}_j)$.

Convolution Formulation of DG Galerkin Projection

Theorem. $A(\vec{v}, \vec{v}_1; \varphi)$ is invariant with respect to a shift, specifically

$$A(\vec{v} + \vec{\eta}, \vec{v}_1 + \vec{\eta}; \varphi(\vec{u} - \vec{\eta})) = A(\vec{v}, \vec{v}_1; \varphi)$$

In the case of **uniform** DG discretizations, $\varphi^j(\vec{w}) = \varphi(\vec{w} + \vec{\xi}_j)$. Then,

$$\begin{aligned} I_{\varphi^j} &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v}, \vec{v}_1; \varphi^j) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v} + \vec{\xi}_j, \vec{v}_1 + \vec{\xi}_j; \varphi^j(\vec{u} - \vec{\xi}_j)) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v} + \vec{\xi}_j, \vec{v}_1 + \vec{\xi}_j; \varphi) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{w} - \vec{\xi}_j) f(t, \vec{x}, \vec{v}_1 - \vec{\xi}_j) A(\vec{w}, \vec{w}_1; \varphi) d\vec{w}_1 d\vec{w} \end{aligned}$$

A.A., T. Nguyen, and A. Wood (2018) introduce

$$I(\vec{\xi}) = \int_{R^3} \int_{R^3} f(\vec{v} - \vec{\xi}) f(\vec{v}_1 - \vec{\xi}) A(\vec{v}, \vec{v}_1; \varphi) d\vec{v}_1 d\vec{v}$$

Observe that on uniform grid $I_{\varphi^j} = I(\vec{\xi}_j)$.

Convolution Formulation of DG Galerkin Projection

Theorem. $A(\vec{v}, \vec{v}_1; \varphi)$ is invariant with respect to a shift, specifically

$$A(\vec{v} + \vec{\eta}, \vec{v}_1 + \vec{\eta}; \varphi(\vec{u} - \vec{\eta})) = A(\vec{v}, \vec{v}_1; \varphi)$$

In the case of **uniform** DG discretizations, $\varphi^j(\vec{w}) = \varphi(\vec{w} + \vec{\xi}_j)$. Then,

$$\begin{aligned} I_{\varphi^j} &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v}, \vec{v}_1; \varphi^j) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v} + \vec{\xi}_j, \vec{v}_1 + \vec{\xi}_j; \varphi^j(\vec{u} - \vec{\xi}_j)) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A(\vec{v} + \vec{\xi}_j, \vec{v}_1 + \vec{\xi}_j; \varphi) d\vec{v}_1 d\vec{v} \\ &= \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{w} - \vec{\xi}_j) f(t, \vec{x}, \vec{v}_1 - \vec{\xi}_j) A(\vec{w}, \vec{w}_1; \varphi) d\vec{w}_1 d\vec{w} \end{aligned}$$

A.A., T. Nguyen, and A. Wood (2018) introduce

$$I(\vec{\xi}) = \int_{R^3} \int_{R^3} f(\vec{v} - \vec{\xi}) f(\vec{v}_1 - \vec{\xi}) A(\vec{v}, \vec{v}_1; \varphi) d\vec{v}_1 d\vec{v}$$

Observe that on uniform grid $I_{\phi^j} = I(\vec{\xi}_j)$.

Discrete Convolution Form

$$I(\vec{\xi}) = \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} f(\vec{v} - \vec{\xi}) f(\vec{v}_1 - \vec{\xi}) A(\vec{v}, \vec{v}_1; \varphi) d\vec{v}_1 d\vec{v}$$

The convolution form is discretized in a finite velocity domain using native nodal-DG Gauss quadratures

$$I_{ij} := I_i(\vec{\xi}_j) = \sum_{i', i''=1}^s \sum_{j'=1}^{M^3} \sum_{j''=1}^{M^3} f_{i'; j' - j} f_{i''; j'' - j} A_{i, i', i''; j', j''}$$

where $f_{i'; j' - j} = f(t, \vec{x}, \vec{v}_{i'; j' - j})$,
 $A_{i, i', i''; j', j''} = A(\vec{v}_{i'; j'}, \vec{v}_{i''; j''}; \phi_{i;c})(\omega_{i'} \Delta \vec{v}/8)(\omega_{i''} \Delta \vec{v}/8)$ and the three dimensional indices i' and i'' run over the velocity nodes within a single velocity cell and indices j' and j'' run over all velocity cells.

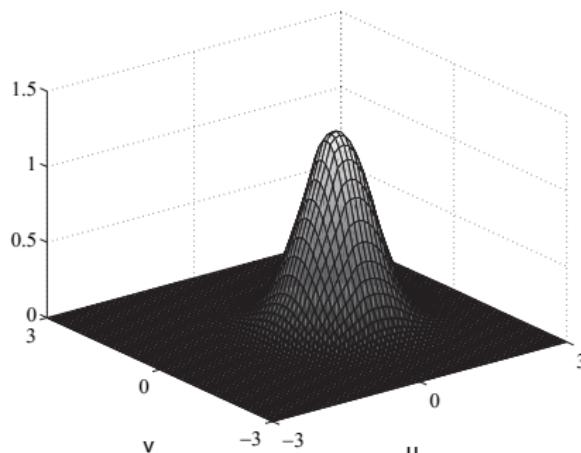
Some shifted indices $j' - j$ point outside of the velocity domain. In direct evaluation, missing values are replaced with zeros.

Extension to Circular Convolution

$$I_{i,j} := I_i(\vec{\xi}_j) = \sum_{i',i''=1}^s \sum_{j'=1}^{M^3} \sum_{j''=1}^{M^3} f_{i',j'-j} f_{i'',j''-j} A_{i,i',i'';j',j''}$$

There is a considerable wealth of theory on fast evaluation of discrete convolution. (Nussbaumer, 1982)

Most are related to circular convolution. This requires to extend the solution and kernel $A_{i,i',i'';j',j''}$ periodically.



Evaluating Circular Convolution in $O(n \log n)$

Let x_n and y_n be periodic sequences with period N . An N -point circular convolution is defined as (Nussbaumer, 1982)

$$z_I = \sum_{n=0}^{N-1} x_n y_{I-n}.$$

Discrete Fourier transform:

$$\mathcal{F}[x]_k = \sum_{n=0}^{N-1} W^{kn} x_n, \quad x_I = \frac{1}{N} \sum_{k=0}^{N-1} W^{-Ik} \mathcal{F}[x]_k, \quad \text{where} \quad W = e^{-i2\pi/N}.$$

Application of the discrete Fourier transform (or any other number theoretical transform) yields

$$\mathcal{F}[z]_k = \mathcal{F}[x]_k \mathcal{F}[y]_k.$$

- $\mathcal{F}[x]_k$ and $\mathcal{F}[y]_k$ can be computed in $O(N \log N)$.
- $\mathcal{F}[z]_k$ are computed in $O(N)$ operations.
- Finally, the values of z_I are obtained in another $O(N \log N)$ operations.

Convolutions of non-periodic sequences are reduced to circular convolutions. A difficulty is **aliasing**.

Discrete Convolution Form

We extend the solution $f_{i;j}$ and kernel $A_{i,i',i'';j,j''}$ periodically in j , $j = 1 \dots, M$ — the cell number.

$$I_{i;j} := I_i(\vec{\xi}_j) = \sum_{i',i''=1}^s \sum_{j'=1}^{M^3} \sum_{j''=1}^{M^3} f_{i';j'-j} f_{i'';j''-j} A_{i,i',i'';j',j''}$$

Theorem Let f_{j_u,j_v,j_w} be a three-index sequence that is periodic in each index with period M and let $A_{j'_u,j'_v,j'_w;j''_u,j''_v,j''_w}$ be a M -periodic six-dimensional tensor. The multi-dimensional discrete Fourier transform of discrete collision operator can be represented as

$$\begin{aligned} \mathcal{F}[I]_{k_u,k_v,k_w} &= M^3 \sum_{l_u,l_v,l_w=0}^{M-1} \mathcal{F}^{-1}[f]_{k_u-l_u,k_v-l_v,k_w-l_w} \mathcal{F}^{-1}[f]_{l_u,l_v,l_w} \\ &\quad \mathcal{F}[A]_{k_u-l_u,k_v-l_v,k_w-l_w,l_u,l_w,l_w} \end{aligned}$$

The Algorithm

Use of DFT allows us to calculate the collision operator in $O(s^9 M^6)$ operations.

- ① evaluate $\mathcal{F}^{-1}[f_i]_{k_u, k_v, k_w}$ in $O(s^3 M^3 \log M)$ operations where s^3 is the number of velocity nodes in each velocity cell.
- ② Next we directly compute the convolution in Fourier space

$$\mathcal{F}[I_{i,i',i''}]_{k_u, k_v, k_w} = M^3 \sum_{l_u, l_v, l_w=0}^{M-1} \mathcal{F}^{-1}[f_{i'}]_{k_u - l_u, k_v - l_v, k_w - l_w} \mathcal{F}^{-1}[f_{i''}]_{l_u, l_v, l_w} \mathcal{F}[A_{i,i',i''}]$$

complexity of this step is $O(s^9 M^6)$.

- ③ Calculate $\mathcal{F}[I_i]_{k_u, k_v, k_w}$. in a complexity $O(s^9 M^3)$ operations.
- ④ Recover $\mathcal{F}^{-1}[\mathcal{F}[I_i]]_{j_u, j_v, j_w} = I_{i;j_u, j_v, j_w}$. in $O(s^3 M^3 \log M)$ operations

Overall, the algorithm has the numerical complexity of $O(s^9 M^6)$. We note that $s = s_u = s_v = s_w$ can be kept fixed and the number of cells M^3 in velocity domain can be increased if more accuracy is desired.

Efficiency

M	DFT		Direct		Speedup
	time, s	α	time, s	α	
9	1.47E-02		1.25E-01		8.5
15	3.94E-01	6.43	4.91E+00	7.18	12.5
21	3.09E+00	6.14	7.80E+01	8.21	25.2
27	1.64E+01	6.65	6.05E+02	8.15	36.7

Table: CPU times for evaluating the collision operator directly and using the Fourier transform.

Aliasing. Split vs. Non-Split form.

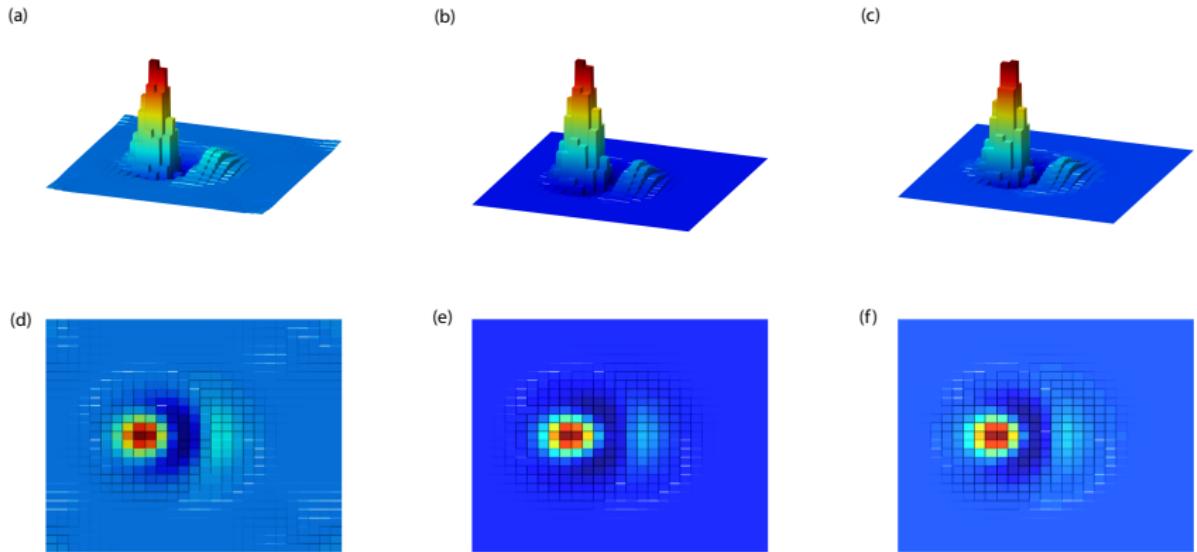


Figure: Evaluation of the collision operator using split and non-split forms: (a) and (d) the split form evaluated using the Fourier transform; (b) and (e) the split form evaluated directly; (c) and (f) the non-split form evaluated using the Fourier transform.

Conservation Errors. Split vs. Non-Split form.

Error in Conservation of Mass				
	Split		Non-split	
n	Fourier	Direct	Fourier	Direct
9	0.37	1.26	1.71E-5	1.92E-5
15	0.10	1.20	1.45E-5	1.71E-5
21	0.18	1.18	0.67E-5	0.93E-5
27	0.18	1.18	0.61E-5	0.86E-5

Error in Conservation of Temperature				
	Split		Non-split	
n	Fourier	Direct	Fourier	Direct
9	3.51	1.69	1.71E-2	1.84E-2
15	0.29	1.25	1.64E-3	3.15E-3
21	1.38	1.24	5.61E-5	1.75E-3
27	1.37	1.24	5.40E-4	1.05E-3

Table: Absolute errors in conservation of mass and temperature in the discrete collision integral computed using split and non-split formulations.