Eleventh International Conference on Computational Fluid Dynamics (ICCFD11), Maui, Hawaii, USA, July 11-15, 2022 **ICCFD11-2022-3103**

Synthesizing Turbulent Channel Flow

A Proposed Form of Solution to Steady Turbulence in Channels

Dr. John William Poduska, Sr.
Weston, MA USA

Yaglom and Lumley in "A Century of Turbulence, 2001" whimsically state:

> *"We believe that, even after 100 years, turbulence studies are still in their infancy. We are naturalists, observing butterflies in the wild. We are still discovering how turbulence behaves, in many respects. We do have a crude, practical, working understanding of many turbulence phenomena, but certainly nothing approaching a comprehensive theory, and nothing that will provide predictions of an accuracy demanded by engineers."*

Abstract: There has long been a need for a fully theoretical basis for describing turbulent flows. We propose a formal representation of the random velocities using a few ordinary smooth non-random functions together with an ordinary Stochastic Integral. We call this the ***Gaussian Transform*** or ***GXF***. For example, for **3D** Channel Flow, we need only two **CDF**'s (Cumulative Distribution Functions) and four ordinary correlation functions. The simpler case of **2D** Channel Flow requires only one **CDF** and one correlation function. Decaying Isotropic Turbulence (**HIT**) requires only one **CDF** and two correlations functions. We will develop these functions for **2D** Channel Flow; we will describe how to compute them numerically from **DNS** results; we will describe the synthesis of multiple realizations of the flow; and we will apply the **NSE** to this representation to come to a full solution to **2D** Channel Flow.

We will show that there ***IS*** a comprehensive theory which provides a complete closed form solution to certain turbulence problems.

We show the existence of such a theory. We do ***not*** (yet) provide the complete answer.

Keywords: Theoretical Turbulence, Multidimensional Stochastic Integrals, Ritz-Galerkin approximation, Metalog **CDF** approximations, Stationary Ergodic Stochastic Processes

1. Introduction

1.1. The Base Case

We study steady state turbulence in the **PPF** configuration – fully-developed turbulence in a high aspect ratio rectangular channel with **x**-streamwise, **y**-cross channel (±L), **z**-spanwise. We adopt the usual assumptions: Newtonian Fluid, constant viscosity and density, incompressible, smooth no-slip walls, stationary on time, stationary/homogeneous on **x** and **z**. "Reynolds Averaging" applies. Our parlance is that the flow is ***stationary*** and ***ergodic*** on **(x,z,t)**.

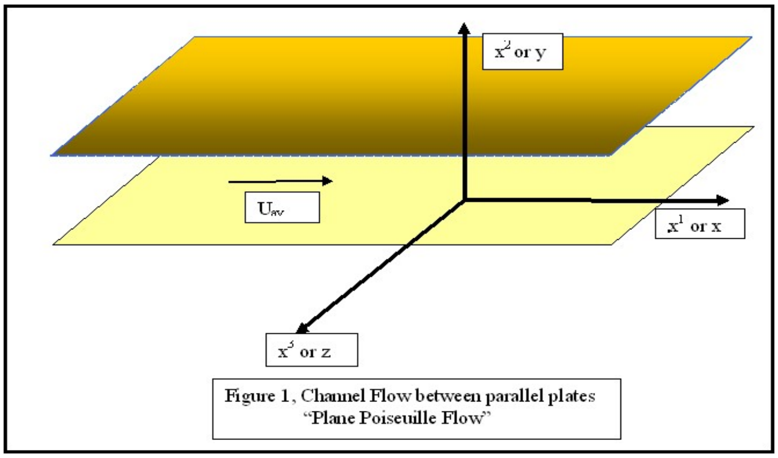

Figure 1.1.1. Standard configuration for **3D PPF** – Steady channel flow between two smooth Parallel Plates. The flow is along the **x**-axis, and the wall normal is the **y**-axis. The **z**-axis is spanwise. For **2D PPF**, the **z**-axis quantities and dependencies are eliminated.

Over the past few years, there have been several comprehensive **DNS** studies[1] of **PPF** over a range of Reynolds numbers. These studies provide an excellent database for developing insight into the functions needed to describe **PPF** turbulence.

<u>Our Goal</u>: We seek a ***closed-form*** equation for the velocities of this steady turbulent channel flow – much like:

$$u^i(x,y,z,t) = \left\{ \begin{array}{l} \text{A closed-form expression involving a few deterministic functions} \\ \text{and a simple random function, namely Brownian Motion} \end{array} \right\} \quad (1.1.1)$$

With this, we can synthesize the flow from first principals. And we can approximate the solution to the equations of motion as close as we like with (say) a Best Fit methodology. All to be shown below.

It is worth noting that physical assumption of the ***continuum*** together with the mathematical assumption of ***ergodicity*** (Reynolds Averaging) are severe constraints and make the stated goal possible.

1.2. The Random Turbulent Velocities We Want

Each **DNS** run (and every experimental record) yields a sample $\mathbf{u^i(x,y,z,t)}$ – one member of an ***ensemble*** of solutions. Following Wiener [1] and Lumley[2] [3], we choose to represent the ensemble of velocities by $\mathbf{u^i(x,y,z,t,\alpha)}$ where the parameter $\boldsymbol{\alpha\in[0,1]}$ selects a sample from the ensemble. Thus, any average is obtained by simple (perhaps Lebesgue) integration:

$$\overline{\text{foo}}(y) = \int_0^1 \text{foo}(x,y,z,t,\alpha)\,d\alpha \quad (1.2.1)$$

A useful way to view this representation is this: $\mathbf{u^i(x,y,z,t,\alpha)}$ is a simple ***deterministic*** function of 5 parameters – **x,y,z,t,α** – and α is selected ***at random*** from **[0,1]**. This allows us to treat $\mathbf{u^i(x,y,z,t,\alpha)}$ like any other ordinary function with a specific value for a given set of parameters.

[1] See for examples, the Lee-Moser [4] and Mortensen-Langtangen [5] listed in the references.
[2] Also, by Steinhaus, Kolmogorov, and Yaglom [2] – among others.

Our principal goal is to show that closed-form formulas exist to represent $\mathbf{u^i(x,y,z,t,\alpha)}$ with a few smooth functions and the Wiener Process $\mathbf{r(s,\alpha)}$ – aka Brownian motion.

1.3. Organization of this Paper

Our research is based on a tool we call the ***Gaussian Transform*** or **GXF** for short. This paper is organized as follows:

1. Develop the **GXF** tool in **1D**.
   This simple case will illustrate the basic principles.
2. Develop the Solution for **2D PPF**.
   The **2D** case is much simpler than the **3D** case.
   The **3D** case follows with simple concepts but complex algebraic gymnastics.
3. Measure the **cdf**'s and Autocorrelation Functions.
   Tailored **DNS** runs of **2D PPF** provide numerical approximation of the required analytic functions.
4. Approximate the **cdf** and Autocorrelation Functions:
   Use existing approximation tools to approximate the **cdf** and Autocorrelation Functions with a few parameters. Then optimize these parameters with some "Best Fit" and/or Galerkin process.

There is much more to be said and done to exploit this **GXF** mechanism. The promise is great: the ability to synthesize – i.e. compute – ***ANY*** statistical quantity – including velocities and pressures from a fixed closed-form equation. Our goal is to show that such closed-form equations exist to represent $\mathbf{u^i(x,y,z,t,\alpha)}$ with a few smooth functions and a convolution integral using the Wiener Process $\mathbf{r(s,\alpha)}$.

2. Using the cdf – Cumulative Distribution Function

2.1. A Simple Scalar Velocity Example

Ultimately, we want to find an equation for $\mathbf{u^i(x,y,z,t,\alpha)}$ with $\alpha \in [0,1]$. To illustrate the process, we consider **PPF** – channel flow – and we pick a scalar velocity at a fixed point in (**x,y,z**) space in a direction specified by the unit vector λ^i thus:

$$u(t,\alpha) = \lambda^i u^i(x,y,z,t,\alpha) \qquad (2.1.1)$$

So, $\mathbf{u(t,\alpha)}$ is stationary and ergodic on "**t**".

This random function has a $\mathbf{cdf^u(..)}$ (***cumulative distribution function for u***) defined as:

$$\mathcal{Pr}\left\{u(t,\alpha) \le \lambda\right\} = \text{cdf}^u(\lambda) \quad \text{stationary, i.e. not-dependent on "t"} \qquad (2.1.2)$$

To illustrate, consider a Gaussian Random Variable.

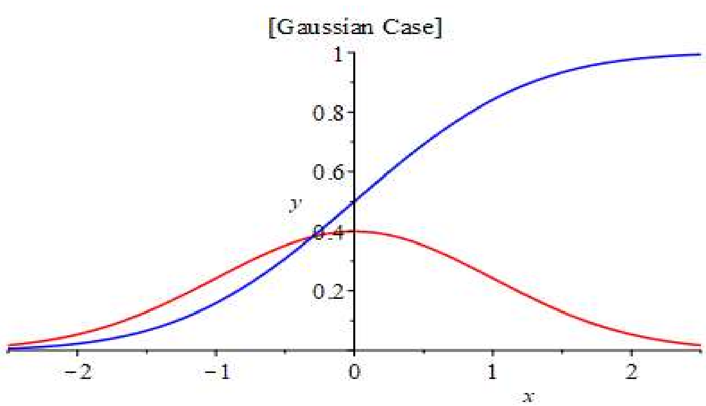

Figure 2.1. The **cdf** and **pdf** for the ***Standard Normal Gaussian Distribution***. In this case the mean is 0 and the variance is 1

This has the well-known monotone, strictly-increasing **cdf** and the famous "bell-shape" **pdf**.

For physical situations, $\mathbf{cdf^u(..)}$ has very "nice" properties – it is monotone strictly-increasing with a **range=[0,1]** and **domain=[±∞]**. It has a smooth derivative known as the $\mathbf{pdf^u(..)}$ (***probability density function for u***), often shaped like the "bell" curve.

2.2. The Associated "quantile" Random Function

We construct the associated "***quantile***" Random Function as follows:

$$q(t,\alpha) = \mathrm{cdf}^u\left(u(t,\alpha)\right) \quad \text{Compose the } \mathbf{cdf} \text{ with the velocity} \tag{2.2.1}$$

What is the ***Cumulative Distribution Function*** (**cdf**) of **q**?

$$\begin{aligned}
&\mathcal{Pr}\{u(t,\alpha)\le\lambda\} = \mathrm{cdf}^u(\lambda) && \text{Standard Definition}\\
&\{u(t,\alpha)\le\lambda\} \Rightarrow \{\mathrm{cdf}^u(u(t,\alpha))\le \mathrm{cdf}^u(\lambda)\} && \text{Apply } \mathrm{cdf}^u(\lambda) \text{ to both sides}\\
&\mathcal{Pr}\{\underbrace{\mathrm{cdf}^u(u(t,\alpha))}_{=q(t,\alpha)} \le \underbrace{\mathrm{cdf}^u(\lambda)}_{\triangleq\mu}\} = \underbrace{\mathrm{cdf}^u(\lambda)}_{\triangleq\mu} && \text{Because } \mathrm{cdf}^u(\lambda) \text{ is monotone}\\
&\mathcal{Pr}\{q(t,\alpha)\le\mu\} = \mu && \text{!!! The Uniform Distribution}
\end{aligned} \tag{2.2.2}$$

So, $\mathbf{q(t,\alpha)}$ has a linear-rectangular distribution, regardless of the original $\mathbf{u(t,\alpha)}$! This demonstrates – as expected – that ***ALL*** single point statistics of $\mathbf{u(t,\alpha)}$ are derived from the $\mathbf{cdf^u}$.

2.3. The Associated "Normal" Random Function

Now the ***Standard Normal*** distribution is a ***Gaussian*** distribution with **mean=0** and **variance=1**. The Normal $\mathbf{cdf^N(..)}$ and its inverse (quantile function) are[3]:

$$\begin{aligned}
&\mathrm{cdf}^N(\lambda) = \frac{1}{2}\left[1+\mathrm{erf}\left(\frac{\lambda}{\sqrt{2}}\right)\right] && \text{The "Normal" Distribution}\\
&\mathrm{icdf}^N(\mu) = \sqrt{2}\cdot\mathrm{ierf}\left(2\mu-1\right) && \text{The "Normal" Quantile Function}
\end{aligned} \tag{2.3.1}$$

Then, construct another ***random function*** as:

$$\phi(t,\alpha) = \mathrm{icdf}^N\left(q(t,\alpha)\right) = \mathrm{icdf}^N\left(\mathrm{cdf}^u(u(t,\alpha))\right) \triangleq M_{u2\phi}(u(t,\alpha)) \tag{2.3.2}$$

This $\boldsymbol{\phi(t,\alpha)}$ is a ***Normal Gaussian Random Function*** with **Mean=0** and **Variance=1**! Moreover, $\boldsymbol{\phi(t,\alpha)}$ is ***smooth***, ***stationary***, and ***ergodic*** on "**t**" because $\mathbf{u(t,\alpha)}$ and $\mathbf{q(t,\alpha)}$ are so. The inverse is:

$$u(t,\alpha) = \mathrm{icdf}^u\left(\mathrm{cdf}^N(\phi(t,\alpha))\right) \triangleq M_{\phi 2u}(\phi(t,\alpha)) \tag{2.3.3}$$

Thus (as expected) all single point statistics – including mean, variance, skewness, kurtosis/flatness, characteristic function, etc. – are contained in $\mathbf{cdf^u(..)}$ independent of "**t**".

Note that both $\mathbf{M_{\phi2u}}$ and $\mathbf{M_{u2\phi}}$ are monotone strictly-increasing because each is the composition of two monotone strictly increasing functions.

<u>Summary</u>: This demonstrates that there exist ***monotone strictly increasing functions***:

[3] The Wikipedia article "Normal Distribution" is quite good.

$$M_{u2\phi}(\lambda) \triangleq \text{icdf}^{N}\left(\text{cdf}^{u}(\lambda)\right)$$
$$M_{\phi2u}(\lambda) \triangleq \text{icdf}^{u}\left(\text{cdf}^{N}(\lambda)\right) \qquad (2.3.4)$$

Such that

$$\phi(t,\alpha) = M_{u2\phi}\left(u(t,\alpha)\right)$$
$$u(t,\alpha) = M_{\phi2u}\left(\phi(t,\alpha)\right) \qquad (2.3.5)$$

So, we represent **u(t,α)** ***exactly*** by a Monotone Function $\mathbf{M_{\phi2u}(..)}$ and a Normal Random Variable **φ(t,α).**

It is worth repeating that **φ(t,α)** is indeed a standard Normal Random Variable which completely and faithfully represents **u(t,α)**. It is differentiated from other Normal **RV**'s by its Autocorrelation Function, often referred to as **B(τ)** for Gaussian **RV**'s.

3. The Wiener Machinery

3.1. The Wiener Process

The "***Wiener Process***" is an idealization of "***Brownian Motion***" named for the Scottish Botanist Robert Brown. The Wiener Process **r(t,α)** is continuous on **t** for any $\alpha \in$ **[0,1]**.

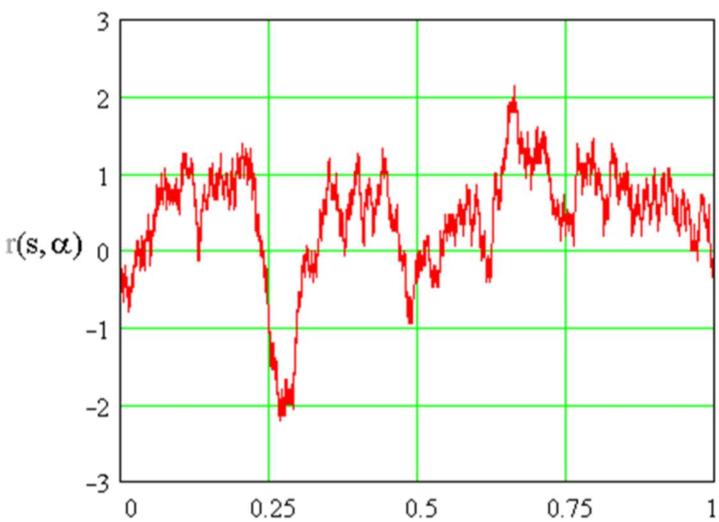

Figure 1, A Sample of the Wiener Process

Figure 3.1.1. A construction of the Wiener Process. Four Thousand Bernoulli Trials were integrated (summed) to form this sample function. The striking characteristics of a curve that is continuous, but not differentiable, and not bounded in variation are apparent. The fractal dimension is 3/2.

We choose a version of the function **r(t,α)** which has some interesting and convenient properties. With regard to **t**: for every α, **r(t,α)** is: continuous on **t**, unbounded in variation, and a fractal with fractal-dimension **3/2**. With regard to α: for almost all **t**, **r(t,α)** is: continuous on α, and for all **t** it is Square Integrable. Moreover, **r(t,α)** has increments which are Gaussian distributed and are mutually independent for non-overlapping intervals.

These properties are enough to completely specify the Wiener Process **r(t,α)**. Most important, this simple process has enough power to be the basis of a complete representation of any Stochastic Process.

3.2. The Wiener Stochastic Convolution Integral

The ***Wiener Stochastic Convolution Integral*** is:

$$\phi(t,\alpha)=\int\Phi(t-s)\,dr(s,\alpha) \qquad \text{Usually taken over infinite limits} \tag{3.2.1}$$

Here **ϕ(t,α)** is a ***Gaussian Process*** which is stationary and ergodic over all time. It is defined as the convolution of an ***unknown*** deterministic function **Φ(s)**, and the ***known*** Wiener Process **r(s,α)**[4]. It is an ordinary Riemann-Stieltjes integral where **dr(s,α)** is an increment whose variance is **ds**.

The derivative and the autocorrelation function are defined by:

$$\begin{aligned}\frac{d}{dt}\phi(t,\alpha)&=\int\left(\frac{d}{dt}\Phi(t-s)\right)dr(s,\alpha) && \text{The derivative is well defined}\\ B(\tau)&=\int\big(\Phi(t+\tau)\Phi(t)\big)dt && \Phi(s)\ \text{Uniquely defines Autocorrelation Function}\end{aligned} \tag{3.2.2}$$

Moreover, **Φ(s)** is square-integrable and has a proper Fourier Transform. These characteristics are critical to solving the **NSE**.

It is also well established[5] that every Stationary, Gaussian random function with finite mean and variance is ***completely*** specified by its mean and auto-correlation function, and any autocorrelation function can be generated with the appropriate kernel to the Wiener Integral.

3.3. Principal Conclusion for 1D

Any physical velocity **u(t,α)** which is stationary and ergodic on "**t**" and has bounded mean and variance is completely defined by two smooth functions and the Wiener Process as follows:

$$\begin{aligned}u(t,\alpha)&=M_{\phi 2u}\big(\phi(t,\alpha)\big) \qquad \text{where:}\quad M_{\phi 2u}(..)\triangleq icdf^{u}\big(cdf^{N}(..)\big)\\ \phi(t,\alpha)&=\int\Phi(t-s)\,dr(s,\alpha)\end{aligned} \tag{3.3.1}$$

With a kernel completely defined by the autocorrelation function of **ϕ(t,α)**.

Importantly, these two functions – $\mathbf{M_{\phi 2u}(..)}$ and **Φ(..)** – can be measured experimentally and/or by **DNS** results.

3.4. Extension to Two or More Stationary Parameters

Recall the ***Wiener Stochastic Convolution Integral*** for one stationary parameter – "**t**":

$$\phi(t,\alpha)=\int\Phi(t-s)\,dr(s,\alpha) \tag{3.4.1}$$

We now extend this integral to ***two*** stationary parameters – "**x**" and "**t**" – as:

$$\phi(x,t,\alpha)=\int\Phi(x-x_1,t-t_1)\,dr(x_1,t_1,\alpha) \qquad \text{Taken over the whole } \{x,t\} \text{ surface} \tag{3.4.2}$$

This integral is a ***surface*** integral, and **dr(x,t,α)** is a surface element whose variance is **dσ(x,t)** – the ***area*** of the surface element. This heuristic description can be made fully rigorous.[6]

Of particular importance, note that **ϕ(x,t,α)** is:

---

[4] Said in a jocular but informative way, we represent an ***extraordinary*** function **ϕ(t,α)** (perhaps a velocity) as the convolution of an unknown but ***ordinary*** function **Φ(..)** with a known but ***bizarre*** function **r(s,α)**! The Stieltjes form integral does indeed exist because **Φ(..)** is bounded in variation and **r(s,α)** is continuous. The literature has a persistent and annoying error in claiming that the integral does not formally exist as a Stieltjes Integral. Yaglom noted that this is an example of the classic "separation of variables" technique.

[5] See Yaglom [2] "Stationary Random Functions", especially section 1.3.

[6] See Wiener [1] and Poduska [6].

1. Stationary and Homogeneous on "**x**" and "**t**" independently. So, ***all*** statistical properties of $\phi(x,t,\alpha)$ are independent of "**x**" and/or "**t**".
2. Ergodic on "**x**" and "**t**" independently. So, Ensemble (α) Averages, Space (**x**) averages, and Time (**t**) averages are all equal.

Extending the **1D** case, the derivatives and the autocorrelation function are defined by:

$$\begin{aligned}
&\frac{\partial}{\partial x}\phi(x,t,\alpha) = \int\left(\frac{\partial}{\partial x}\Phi(x-x_1,t-t_1)\right)dr(x_1,t_1,\alpha)\\
&\frac{\partial}{\partial t}\phi(x,t,\alpha) = \int\left(\frac{\partial}{\partial t}\Phi(x-x_1,t-t_1)\right)dr(x_1,t_1,\alpha)\\
&B(\xi,\tau) = \iint\left(\Phi(x+\xi,t+\tau)\cdot\Phi(x,t)\right)dxdt
\end{aligned} \tag{3.4.3}$$

Moreover, $\Phi(x,t)$ is square-integrable and has a proper Fourier Transform. These characteristics are critical to solving the **NSE**. Finally, by extension from the **1D** case, every 2-parameter, Stationary, Gaussian random function with finite mean and variance is ***completely*** specified by its mean and auto-correlation function, and any autocorrelation function can be generated with the appropriate kernel to the Wiener Integral.

4. Steady 2D Plane Poiseuille Flow

4.1. Basic Random Quantities

2D flows are rare in the physical world – two examples are large scale atmospheric flows and soap-films. Nevertheless, we expect the study of **2D PPF** to yield valuable insights into the full **3D** flow. So, we examine this simplified case first before we come to the full **3D PPF** flow.

Steady, fully developed, **2D PPF** is stationary and ergodic on "**x**" and "**t**" independently. Since – by conservation of mass – the divergence of the velocity is zero, the flow is completely defined by one stream function $s(x,y,t,\alpha)$ with a double order zero at $y=\pm L$. So:

$$\begin{aligned}
&s(x,y,t,\alpha) \quad \text{is the 2D PPF Stream Function}\\
&u^x(x,y,t,\alpha) = +\partial_y s(x,y,t,\alpha)\\
&u^y(x,y,t,\alpha) = -\partial_x s(x,y,t,\alpha)
\end{aligned} \tag{4.1.1}$$

Thus for **2D PPF** we need only one scalar stream function, viz $s(x,y,t,\alpha)$.

The procedure for implementing the **GXF** for **2D PPF** follows closely the **1D** case covered above.

4.2. The 2D PPF Associated quantile Function

The $\mathbf{cdf^s}(\lambda,y)$ of $s(x,y,t,\alpha)$ is independent of "**x**" and "**t**" but is dependent on "**y**", hence:

$$\begin{aligned}
&cdf^s(\lambda,y) && \text{cdf of } s(x,y,t,\alpha)\\
&q^s(x,y,t,\alpha) = cdf^s\left(s(x,y,t,\alpha),\, y\right) && \text{Associated Quantile Function}\\
&\phi^s(x,y,t,\alpha) = icdf^N\left(q^s(x,y,t,\alpha)\right) && \text{Associated Normal Function}
\end{aligned} \tag{4.2.1}$$

Restated: $\phi^s(x,y,t,\alpha)$ is a ***Normal*** Gaussian function directly derived from $s(x,y,t,\alpha)$ by:

$$\begin{aligned}
&\phi^s(x,y,t,\alpha) = M^s_{s2\phi}\left(s(x,y,t,\alpha),\, y\right)\\
&M^s_{s2\phi}(\lambda,y) = icdf^N\left(cdf^s(\lambda,y)\right)
\end{aligned} \tag{4.2.2}$$

The $M^s_{s2\phi}(\lambda,y)$ is continuous and strictly increasing on λ. Thus, it has a proper λ-inverse namely $M^s_{\phi 2s}(\lambda,y)$.

4.3. The Autocorrelation Function

Any stationary Gaussian Random Function is completely defined by its mean and auto-correlation function[7]. The associated $\boldsymbol{\phi^x(x,y,t,\alpha)}$ is ***Standard Normal*** – i.e. $\boldsymbol{\mu=0}$ and $\boldsymbol{\sigma=1}$ – but with an unspecified auto-correlation function. The **2D** Wiener Stochastic Convolution Integral applied to this case is:

$$\phi^s(x,y,t,\alpha)=\int\Phi^s(x-x_1,y,t-t_1)\,dr(x_1,t_1,\alpha) \qquad (4.3.1)$$

The auto-correlation function is:

$$B^s(\xi,y,\tau)=\int\Phi^s(x+\xi,y,t+\tau)\Phi^s(x,y,t)\,dx\,dt \qquad (4.3.2)$$

Given any $\mathbf{B^s(\xi,y,\tau)}$, a corresponding $\mathbf{\Phi^s(x,y,t)}$ can be readily defined[8].

4.4. The Principal Conclusion for 2D PPF

The $\boldsymbol{\phi^s(x,y,t,\alpha)}$ thus obtained is statistically identical to any other **2D** Standard Normal Random function with the same autocorrelation function $\boldsymbol{\phi^s(x,y,t,\alpha)}$. Thus we can formulate the random variable $\mathbf{s(x,y,t,\alpha)}$ from the Wiener integral as follows:

$$\begin{aligned}&\phi^s(x,y,t,\alpha)=\int\Phi^s(x-x_1,y,t-t_1)\,dr(x_1,t_1,\alpha) && \text{Gaussian Process}\\ &s(x,y,t,\alpha)=M^s_{\phi 2s}\left(\phi^s(x,y,t,\alpha)\right) && \text{Physical Stream Function}\end{aligned} \qquad (4.4.1)$$

<u>So, we claim this</u>: With two smooth functions and an integral with a **2D** Wiener process, we can compute any statistical property of **2D PPF** flow. We can also generate a "**DNS**" run with a computer-generated Wiener Process and visualize the result. **QEF**

5. Outline: Measuring cdf's and Autocorrelation Functions

5.1. A Prototypical DNS Run for 2D PPF

Suppose we run a **DNS** simulation with a high enough **Re** to have fully developed turbulence in a **2D PPF** setting. Suppose this **DNS** is discretized with enough points on the **x**-axis, **y**-axis, and **t**-time to get a reasonably accurate picture of the flow. Assume the stream function, velocity and pressure data is stored in a **Nx*Ny*Nt** array. In fact, we will want this array for many Reynolds numbers.

A promising structure is this:

1. $\mathbf{Nx=2^{10}}$**, Lx=6π**, Discretized using Discrete Fourier Transform
2. $\mathbf{Ny=2^{10}+1}$**, Ly=2**, Discretized using Cubic or Quintic Splines
3. **RK4 Runge-Kutta,** Time stepping

Then the procedure is:

1. Time step the **NSE** (in convective form) for the two velocities $\mathbf{u^x}$ and $\mathbf{u^y}$.
2. Solve the Poisson equation for the stream function **s**, using cubic/quintic splines.
3. Derive updated $\mathbf{u^x}$ and $\mathbf{u^y}$. which automatically satisfy Mass Conservation.
4. Record the **Nx*Ny*Nt** array.

---

[7] See Yaglom "Stationary Random Functions", especially section 1.3.

[8] Essentially, the Fourier Transform of $\mathbf{B^x(..)}$ is the Fourier Transform of $\mathbf{\Phi^x(+)*\Phi^x(-)}$.

This structure yields **Nx=Ny=Nt=2^{10}=1k** axis points. Assuming 32 bytes of data per entry, we need **32GB** storage for the whole array. This is a manageable task for **2D PPF**. It is a much harder task (at least **Nz** times harder) for **3D PPF**.

This simulation can be done in Python and run on a robust desktop computer in quite reasonable time.

5.2. Estimating the cdf

For **2D PPF**, we want to estimate the **$cdf^s(\lambda,y)$** based on the **DNS** results. For each of the **y**-axis points, the stored array has **Nx*Nt** values of **s**. We could make a histogram of the **Nx*Nt** values, but a far simpler process is to simply sort the **Nx*Nt** values for each **y**-point. Then normalize the ordinal by the total count **Nx*Nt** to get a reasonable piecewise linear approximation to the **cdf**[9]. There are some much more sophisticated approximation techniques – such as Pearson distributions and Metalog equations – but I think the simple sort is good enough for now.

With this simple sort technique, we have the **$cdf^s(\lambda,y)$** for **Ny** points. It will be interesting to see just how the **$cdf^s(\lambda,y)$** varies with **y**.

Using this approximate **$cdf^s(\lambda,y)$**, we can compute the related quantile function **$q^s(x,y,t)$** by table lookup in the **$cdf^s(\lambda,y)$**. From this we can compute the related Normal function **$\phi^s(x,y,t)$**.

5.3. Estimating B(..), the Autocorrelation Function of $\phi^s(x,y,t)$

We can now compute **B(x,y,t)** from the computed values of **$\phi^s(x,y,t)$** using standard software packages – especially convenient in SciPy.

Some caution is well advised in computing the autocorrelation function on the **x**-axis. The **x**-discretization is based on a repeating box. So, autocorrelations beyond **Nx/2** are very suspect. Perhaps wise to limit bounds of the autocorrelation function to **NX/4** with no wrapping. There is much computer-experimentation to be done here.

The ***Wiener Stochastic Convolution Integral*** for ***two*** stationary parameters is:

$$\phi(x,y,t,\alpha) = \int \Phi(x-x_1,y,t-t_1)dr(x_1,t_1,\alpha) \qquad (5.3.1)$$

The corresponding autocorrelation function is:

$$B(\xi,y,\tau) = \iint \Phi(x+\xi,y,t+\tau)\cdot\Phi(x,y,t)\cdot dxdt \qquad (5.3.2)$$

And the corresponding Fourier transform of **B(..)** on **x** and **t** is:

$$B(\kappa,y,\sigma) = \Phi(\kappa,y,\sigma)\cdot\Phi(-\kappa,y,-\sigma) \qquad (5.3.3)$$

Thus the discretized **$\Phi(\kappa,y,\sigma)$** can be computed and from this **$\Phi(x,y,t)$** comes by a **FFT**.

5.4. Summary of the GXF Measurement Process

The measurement and synthesizing process described requires these steps:

1. <u>**Design and run a DNS package especially for 2D PPF**</u>
   The design is more oriented to flexibility and modification than to ultimate accuracy. So, Python is the preferred language.

---

[9] This algorithm can be somewhat improved by using quantum cell midpoints. There are several more refined algorithms for estimating the **cdf**.

2. <u>**Compute Approximate cdf by simple sort**</u>
   This will give reasonable quantile and related Normal **RV**'s. But this may be lacking in determining the tails of the **cdf**.
3. <u>**Compute the Numerical Autocorrelation Function**</u>
   This will likely require smoothing of the experimental $\mathbf{B(\kappa,y,\sigma)}$ before deriving $\mathbf{\Phi(\kappa,y,\sigma)}$. Possibly by frequency truncation, or perhaps a Gaussian Filter.
4. <u>**Approximate the Kernel of the Wiener Integral.**</u>
   After smoothing $\mathbf{B(\kappa,y,\sigma)}$, deriving the $\mathbf{\Phi(\kappa,y,\sigma)}$ is in essence a "**sqrt**" operation, reminiscent of the ***Wiener-Hopf*** equation/process.
5. <u>**Use these results to synthesize 2D PPF Turbulence**</u>
   The synthesis process – i.e. generate a sample flow – can be done numerically by ***discretizing*** the ***Wiener Stochastic Convolution Integral***. We use a random number generator for $\mathbf{\Delta r(x,t)}$ and a suitable $\mathbf{\Phi(x,y,t)}$ to generate a discrete sample $\mathbf{\phi(x,y,t,\alpha)}$ by the summation:
   $$\phi(x,y,t) = \sum_{p,q} \Phi(x-\xi_p, y, t-\sigma_q)\cdot \Delta r(\xi_p,\sigma_q)$$
   Then apply the $\mathbf{M_{\phi 2s}(..)}$ to generate a discrete sample stream function $\mathbf{s(x,y,t)}$. **QEF**

The end result is a good and refinable mechanism to measure the functions needed by the GXF.

6. Outline: A Ritz-Galerkin Approximate Solution to NSE

6.1. Approximate Computed Solutions

There are several ways to compute an approximate solution to the **NSE** in **2D PPF** configuration. One traditional way – **DNS** – is to directly simulate the **NSE** over a space lattice and time-step the equations. The result is a large array of data which is very useful, but quite specific to a given flow situation.

In contrast, we seek to approximate the smooth functions of the **GXF** process with a few parameters to be optimized as a "***Best Fit***" in the **NSE**. We explore here the ***Galerkin Error Residual Method*** sometimes known as the **GERM** method.

6.2. The Basic Best-Fit Process

We describe a ***Ritz-Galerkin*** method for obtaining an approximate solution to the **NSE** in **2D PPF** configuration using the ***Gaussian Transform*** method described above.

The process is as follows:

1. Define a parameterized class of convergent approximate solutions which match the boundary conditions and other constraints, i.e., Conservation of Mass. This is fundamentally the ***Galerkin*** methodology.
2. Insert this approximate solution into the **NSE** resulting in an error ***residual***.
3. Define a scalar norm to be the Mean Square Residual integrated across the channel.
4. Minimize the scalar norm with regard to the free parameters in the approximation.

Specifically:

1. Approximate the **CDF** using an **n**-term ***Metalog Distribution***
2. Approximate the Wiener integral kernel $\mathbf{\Phi(x,y,t)}$ by an **m**-term truncated ***Hermite-Function*** Series

This process automatically satisfies all boundary conditions as well as the Conservation of Mass equation. It can be refined by increasing the number of **n**-terms and **m**-terms, ultimately converging to the exact answer.

6.3. Some Observations and Considerations

There are many possible refinements and improvements to this process. But at a minimum, we have demonstrated a convergent path to a full solution. And from this – or other – approximation to the two basic functions, we can synthesize a realization of the turbulent flow.

We expect the **DNS** results to provide guidance to the structure of the **CDF** and **$\Phi(x,y,t)$** especially to gauge how these functions vary with Reynolds Number.

7. A Quick Look at 3D PPF Flows

7.1. Basic 3D Random Quantities

Incompressible **3D PPF** are completely characterized by two velocities (say) **$u^x(x,y,z,t,\alpha)$** and **$u^z(x,y,z,t,\alpha)$**. From these, the third velocity **$u^y(x,y,z,t,\alpha)$** is obtained from the Conservation of Mass equation. We say that **3D PPF** has two degrees of freedom or **2DOF**. A simple and satisfactory way to account for this is to use two independent Wiener Integrals shown schematically as follows[10]:

$$\begin{aligned}\phi^x(-,\alpha) &= \int \Phi^{x1}(-)\,dr^1(-,\alpha) + \int \Phi^{x2}(-)\,dr^2(-,\alpha)\\ \phi^z(-,\alpha) &= \int \Phi^{z1}(-)\,dr^1(-,\alpha) + \int \Phi^{z2}(-)\,dr^2(-,\alpha)\end{aligned} \qquad (7.1.1)$$

Where **$r^1(-,\alpha)$** and **$r^2(-,\alpha)$** are two independent Wiener processes.

If the flow were compressible, there would be **3DOF**. If the flow were incompressible **MHD** assuming Bullard's equation, there would be **4DOF**.

7.2. The Associated 3D quantile and Normal Functions

The **Associated Quantile Functions** are:

$$\begin{aligned}q^x(x,y,z,t,\alpha) &= cdf^x\left(u^x(x,y,z,t,\alpha),\ y\right)\\ q^z(x,y,z,t,\alpha) &= cdf^z\left(u^z(x,y,z,t,\alpha),\ y\right)\end{aligned} \qquad (7.2.1)$$

The **Associated Normal Functions** are:

$$\begin{aligned}\phi^x(x,y,z,t,\alpha) &= M^x_{u2\phi}\left(u^x(x,y,z,t,\alpha),\ y\right) \qquad M^x_{u2\phi}(\lambda,y) = icdf^N\left(cdf^x(\lambda,y)\right)\\ \phi^z(x,y,z,t,\alpha) &= M^z_{u2\phi}\left(u^z(x,y,z,t,\alpha),\ y\right) \qquad M^z_{u2\phi}(\lambda,y) = icdf^N\left(cdf^z(\lambda,y)\right)\end{aligned} \qquad (7.2.2)$$

These functions – **$\phi^x(x,y,z,t,\alpha)$** and **$\phi^z(x,y,z,t,\alpha)$** – are ***Standard Normal*** random functions, stationary and ergodic on **(x,z,t)**, and parametrically dependent on "**y**". The appropriate Wiener Integral is:

$$\phi^i(x,t,z,y,\alpha) = \int \Phi^{i\beta}(x-x_1,t-t_1,z-z_1,y)\,dr^\beta(x_1,t_1,z_1,\alpha) \quad \text{For } i=\{x,z\},\ \beta=\{x,z\} \qquad (7.2.3)$$

This integral is a ***volume*** integral, and **$dr(x,t,\alpha)$** is a volume element whose variance is **$d\tau(x,t)$** – the ***volume*** of the element.

Also, **$\Phi^{i\beta}(..)$** can be determined from the vector autocorrelation function **$B^{ij}(..)$**. So, the **3D PPF** velocities are determined by 4 smooth **L_2** correlation functions having 3 parameters, together with 2 smooth strictly increasing functions $M^i_{\phi 2u}(\lambda,y)$ having 2 parameters. Schematically, the construction is:

---

[10] This heuristic argument can be made fully rigorous.

$$\Phi^{i\beta}(x,y,z,t) \xrightarrow[\int \Phi^{i\beta} dr^{\beta}(..,\alpha)]{\text{Wiener Integral}} \phi^{i}(x,y,z,t,\alpha)$$
$$\xrightarrow[\text{Normal cdf}(\lambda)]{\Psi(\lambda)} q^{i}(x,y,z,t,\alpha) \qquad (7.2.4)$$
$$\xrightarrow[\text{inverse cdf}^{i}(\lambda,y)]{} u^{i}(x,y,z,t,\alpha)$$

These 6 functions are deterministic – ***Not*** Random. So, they can be determined from the equations of motion analytically if possible; approximately (say) by a Galerkin process; or numerically if necessary. Moreover, they can be measured from experiment or **DNS** runs.

7.3. Comments on 3D PPF Solution

Comparing 3D PPF to 2D PPF: The two sections above show that **3D PPF** is much more complex ***in detail*** – 4 correlation function vs. 1, and 2 **cdf** functions vs. 1. But **3D PPF** involves no greater ***conceptual*** complexity. This supports – if not justifies – the initial study of the **2D** cases for guidance in the study of the full **3D** cases. M. Lesieur and U. Frisch seem to agree.

Other Flows: Many other flows can be analyzed by similar techniques including: **HIT** Homogeneous Isotropic turbulence, turbulent wakes behind spheres and cylinders, and the **MFD** cases of all these flows.

8. Summary and Conclusions

8.1. Basic Results

We have demonstrated that there is a closed-form solution to many stationary and ergodic turbulent flows which consists of:

1. A few very regular real functions
2. Wiener's Stochastic Convolution Integral

This solution allows us to calculate and visualize any specific instance of a flow – as does a **DNS** run.

For example, for **2D PPF**, the method used is this schematically:

$$s = M(\phi, y) \quad \text{Physical Stream Function from Normal Process}$$
$$\phi = \int \Phi \, dr \quad \text{Normal Gaussian Process from Wiener Integral} \qquad (8.1.1)$$

Or more formally and precisely:

$$s(x,y,t,\alpha) = M_{\phi 2s}\left(\phi^{s}(x,y,t,\alpha),\ y\right) \qquad s \text{ from } \phi^{s}$$
$$\phi^{s}(x,y,t,\alpha) = \int \Phi^{s}(x-x_1, y, t-t_1)\, dr(x_1,t_1,\alpha) \qquad \phi^{s} \text{ from Wiener Integral} \qquad (8.1.2)$$

There are two very ordinary functions involved:

1. $M^{s}_{\phi 2u}(\lambda, y)$ very smooth on both parameters and strictly increasing on λ
2. $\Phi^{s}(x,y,t)$ very smooth and square integrable on all parameters

For **3D PPF**, a similar analysis leads to two **M**'s and four **Φ**'s with similar smooth properties.

8.2. Next Steps

This analysis leads to (at least) three fruitful avenues to explore:

1. Use **DNS** results to numerically determine the **M**'s and **Φ**'s for many **Reynolds** Numbers
2. Approximate **M**'s and **Φ**'s with parameterized functions and Optimize by a **Galerkin** Process
3. Attempt to solve the equations of motion directly for the **M**'s and **Φ**'s

For 1: We intend to develop **2D PPF** runs with modest accuracy over a range of Reynolds Numbers. This will provide a basis for seeing the dependence of the **M**'s and **Φ**'s on **Re** and **y**.

For 2: Guided by the results of **"1"** above, we will use standard models for both **M**'s and **Φ**'s – e.g. parameterized ***Metalog Distributions*** for **M**'s and various rapidly decreasing formulations of the **Φ**'s.

For 3: This will be an extremely difficult task. We do have guidance from **"1"** and **"2"** above, as well as a host of symmetry and boundary conditions, but the task is daunting even with Symbolic Math packages, e.g. Maple. The prospects for Isotropic turbulence are somewhat brighter because there is only one **M** function and the **Φ**'s must have an Isotropic structure.

9. References

9.1. Books

[1] **Norbert Wiener**, *Nonlinear Problems in Random Theory*, MIT Press, Cambridge, MA, 1958
[2] **A.M. Yaglom**, *An Introduction to the Theory of Stationary Functions*,
Prentice Hall, Englewood Cliffs, NJ, 1962. ISBN 0-486-60579-5
[3] **John L. Lumley**, *Stochastic Tools in Turbulence*, Dover, New York 1970) ISBN-13 978-0-486-46270-7

9.2. Papers

[4] **M. Lee and-R. Moser**, *Direct numerical simulation of turbulent channel flow up to Re_τ≈5200*,
(http://journals.cambridge.org/article_S0022112015002682)
[5] **M. Mortensen, H. P. Langtangen**, *High performance Python for direct numerical simulations of turbulent flows*, Computer Physics Communications 203 (2016) 53–65. doi:10.1016/j.cpc.2016.02.005.
[6] **J. W. Poduska**, *Random Theory of Turbulence*, ScD. Thesis, MIT 1962