A Short Paper On Theoretical Turbulence

1. The Task to Be Accomplished

Turbulence — the last unsolved problem of classical physics. This Short Paper follows a narrow path of:

1. Specify the Problem – Turbulent Shear Flows
2. Select a Form of Solution – $\mathbf{u^i(x,y,z,t;\alpha)}$
3. Explore the Representation of $\mathbf{u^i(x,y,z,t;\alpha)}$ – Wiener Functionals and Quantile Functions
4. Discuss a Galerkin-Optimized numerical approximation to $\mathbf{u^i(x,y,z,t;\alpha)}$
5. Display some Tentative Results – Suggestive, but not Conclusive
6. Brief Description of a Quantile Approach

There are no proofs here, only references to the literature.

Why all the bother? We want to show that the equations of motion for ***random*** $\mathbf{u^i(x,y,z,t;\alpha)}$ can be converted into equivalent equations in several ***non-random*** functions of the form $\boldsymbol{\phi(x,y,z,t)}$. In other words, transform the ***randomness*** out of the **PDE**'s! Such equations – when solved – are solved once and for all.

2. Definition: Problem and Form of Solution

2.1. The Problem – Turbulent Shear Flows

Consider a steady turbulent flow in a wide rectangular channel – ***Plane Poiseuille Flow*** – with random velocities $\mathbf{u^i(x,y,z,t;\alpha)}$. The velocities are ***stationary*** and ***ergodic*** on **{x,z,t}**, they have bounded local energy, they are $\mathbf{C^k}$ and have $\mathbf{C^k}$ distribution functions[1]. Many experimental results have shown that the random part of these velocities is not entirely Gaussian (exemplified by "Skewness" and "Flat-ness"), but very close to it.

We adopt the usual assumptions:

1. The Fluid is simple Newtonian
2. The Navier-Stokes and Mass-Conservation **PDE**'s apply
3. The Fluid is Incompressible
4. The Viscosity is constant, independent of temperature
5. We accept "***Reynolds Averaging***" and the **RANS** – aka ***Ergodic***

These assumptions are universally adopted by experimental, empirical, theoretical and numerical re-searchers for incompressible steady flows.

[1] $\mathbf{C^k}$ means that the function and its first k derivatives are continuous. Some writers say "**k**-continuous" in place $\mathbf{C^k}$.

2.2. Form of Solution – $u^i(x,y,z,t,\alpha)$

For us, ***fluid turbulence*** is the presence of ***random*** velocities, pressures, etc. At the most fundamental level (per Wiener, Lumley, et al.):

> "A Random Physical Quantity (say a turbulent velocity) is a ***real*** function of ***real*** parameters space, time, and a ***real*** (sampling) parameter $\alpha\in[0,1]$."

Therefore, a velocity component is just $\mathbf{u(x,y,z,t;\alpha)}$, whose average is:

$$\mathcal{E}\{u(x,y,z,t;\alpha)\} = \bar{u}(x,y,z,t) = \int_0^1 u(x,y,z,t;\alpha)\cdot d\alpha \qquad (2.2.1)$$

So, the parameter α selects a particular path from the sample space, and averages are simple integrals on α over the interval $[0,1]$[2].

That's about it!

It is not trivial to prove that this representation is valid, but in fact, this can be done for smooth random quantities – including all physical quantities in continuum physics. This is the formalism initiated by Steinhaus, developed by Wiener[3] and later used by Yaglom[4] and Lumley[5]. It is a simple representation of random quantities with all the helpful machinery of real analysis, especially Hilbert Spaces. We do not have to deal with measure theory, sigma algebras, etc.

Our entire research program is based on the proposition that the velocities in steady shear flows (and in fact in many other flows, e.g. **HIT**) are properly represented by:

$$\mathbf{u^i(x,y,z,t;\alpha)} \qquad (2.2.2)$$

Now we proceed to represent these Random Quantities by various expansion techniques including ***Wiener Functionals*** and (later) ***Quantile Functions***.

3. First Steps to a Proper Representation

3.1. Brownian Motion – The Wiener Process

Our Tools – the Wiener Machinery – are based on the Wiener Process (aka Brownian Motion):

$$r(t;\alpha) \qquad (3.1.1)$$

This is a ***Gaussian*** process with these properties:

1. $\mathbf{r(0;\alpha)=0}$ for all α
2. $\mathbf{r(t;\alpha)}$ is Continuous for all **t** and all α
3. $\mathbf{r(t;\alpha)}$ is ***not*** locally Bounded Variation
4. $\mathbf{r(t;\alpha)}$ has Independent Identically Distributed Gaussian increments

---

[2] The parlance "u(..) is indexed over [0,1]" is sometimes used.

[3] See Wiener "Nonlinear Problems in Random Theory".

[4] See Yaglom "Stationary Random Functions", especially section 2.8.

[5] See Lumley "Stochastic Tools in Turbulence" especially section 1.6.

Many more interesting properties are developed in the literature[6], but the most important properties for us are: continuous, ***not*** bounded variation, and **iid** Gaussian increments.

3.2. The Wiener Stochastic Convolution Integral

Consider the ***Wiener Stochastic Convolution Integral***[7]:

$$f(t;\alpha) = \int_a^b F(t+s)\, dr_s(s;\alpha) \qquad (3.2.1)$$

This is a standard Riemann-Stieltjes integral whose existence is assured[8] because **F(..)** has bounded variation and **r(..)** is continuous. The integral exists with infinite limits provided that **F(..)** decreases sufficiently rapidly.

Then **f(t;α)** is ***Gaussian*** and ***stationary*** on **t**. Moreover, the ***Cross-Correlation*** of two such functions is:

$$\mathcal{E}\{f(t;\alpha)\cdot g(t+\tau;\alpha)\} = \int_{-\infty}^{+\infty} F(t)\cdot G(t+\tau)\cdot dt \qquad (3.2.2)$$

Similarly, the ***Auto-Correlation*** and ***Mean-Square*** are:

$$\begin{aligned} &\mathcal{E}\{f(t;\alpha)\cdot f(t+\tau;\alpha)\} = \int_{-\infty}^{+\infty} F(t)\cdot F(t+\tau)\cdot dt && \text{Auto-Correlation} \\ &\mathcal{E}\{f(t;\alpha)\cdot f(t;\alpha)\} = \int_{-\infty}^{+\infty} F(t)\cdot F(t)\cdot dt: && \text{Mean-Square} \end{aligned} \qquad (3.2.3)$$

Proof of these equations is tedious but straightforward[9].

Some observations:

1. **f(t;α)** is Gaussian and stationary on **t**.
2. There is a useful ***isometry*** between **f(..)** and **F(..)**
3. **f(t;α)** is square-integrable and stationary over infinite limits, hence **f(t;α)** is ***Ergodic*** on t and **α**, i.e. infinite averages over **t** are equal to ensemble averages defined as integrals on **α** over **[0,1]**.
4. **t**-derivatives exist for smooth enough **F(..)**, i.e.:

$$\frac{d}{dt} f(t;\alpha) = \int_{-\infty}^{+\infty} \left(\frac{d}{dt} F(t+s) \right) \cdot d_s r(s;\alpha) \qquad (3.2.4)$$

Finally, every stationary, Gaussian random function with finite mean and variance is ***completely*** specified by its mean and auto-correlation function[10], and any autocorrelation function can be generated with the appropriate kernel to the Wiener Integral. This is very important.

---

[6] The Wikipedia Article "Wiener Process" is quite good.
[7] Yaglom observed that this is a classic case of "separation of variables".
[8] This is a standard result of sophomore level real analysis, but sometimes misstated in the literature.
[9] See Wiener, ibid
[10] See Yaglom "Stationary Random Functions", especially section 1.3.

We now have a tool for completely specifying a stationary Gaussian Random Function along with its derivative suitable for plugging into a differential equation. The tasks remaining are to extend this description to:

1. Non-Gaussian Random Functions
2. Multiple parameters, i.e. **(x,y,z,t)** in place of just **(t)**
3. Vector Values – $\mathbf{u^i(..)}$ – and multiple degrees of freedom

3.3. Wiener Polynomial Functionals

Recall the ***Wiener Stochastic Convolution Integral***:

$$u(t;\alpha) = \int U(t+s)\, dr_s(s;\alpha) \tag{3.3.1}$$

This is a complete representation for Gaussian random functions. We can add a second, third and more non-Gaussian terms as follows:

$$u(t;\alpha) = \begin{cases} u_0 = U_0 = \mathcal{E}\{u\} & \text{Mean Value} \\ \int U_1(t+s)\, dr_s(s;\alpha) & 1^{st}\ \text{Term, Gaussian} \\ \iint U_2(t+s_1, t+s_2)\, dr_s(s_1;\alpha)\, dr_s(s_2;\alpha) & 2^{nd}\ \text{Term non-Gaussian} \\ \iiint U_3(\cdots,\cdots,\cdots)\, dr(..)\, dr(..)\, dr(..) & 3^{rd}\ \text{Term non-Gaussian} \\ +\cdots+ & \text{etc.} \end{cases} \tag{3.3.2}$$

These are Wiener's ***Homogeneous Polynomial Functionals*** (aka **HPF**'s). His (functional) notation is:

$$\begin{aligned} u(t;\alpha) &= \sum_{k=0}^{\infty} u_k(t;\alpha) \\ u_k(t;\alpha) &= \int \cdots \int U_k(t+s_1, t+s_2, \cdots, t+s_k) \cdot \prod_{n=1}^{k} dr(s_n;\alpha) \\ &= \mathcal{H}_k\left[U_k(s_1, s_2, \cdots, s_k);\alpha\right] \end{aligned} \tag{3.3.3}$$

Here $\mathcal{H}_k[U_k(..);\alpha]$ is the $\mathbf{k^{th}}$-order **HPF**[11]. Note that the kernel $\mathbf{U_k(..)}$ may be taken as symmetrical in its **k** arguments. Also note that the $\mathcal{H}_k[..;\alpha]$ are not mutually orthogonal. Wiener used a Gram-Schmidt process to generate an orthogonal set of ***Orthogonal Polynomial Functionals*** (**OPF**'s) – the $\mathcal{G}_k[U_k(..);\alpha]$. These will be quite useful, especially for steady shear flows.

We emphasize here that:

1. $\mathbf{U_k(..)}$ is symmetrical in its **k**-arguments.
2. $\mathbf{U_k(..)}$ is bounded-variation on all **k**-arguments.
3. $\mathbf{U_k(..)}$ will have well defined derivatives.
4. $\mathbf{U_k(..)}$ is square integrable over $[-\infty,+\infty]$ on all **k**-arguments.
5. $\mathbf{U_k(..)}$ can be represented by a sum of basis functions in its Hilbert Space.

[11] Wiener used a semi-colon rather than a comma to separate the α parameter. This is stylistic only – there is no mathematical significance.

These $\mathcal{H}_k[U_k(..);\alpha]$ and $\mathcal{G}_k[U_k(..);\alpha]$ are the main tools used in our study of theoretical turbulence. There remain the tasks of extending these tools to:

1. Multiple parameters – i.e. **(x,y,z,t)** in place of just **(t)**
2. Vector Values and Multiple Degrees of freedom

In addition, we will explore the use of ***Quantile*** functions to reduce simplify the representation of velocities, $\mathbf{u^i(x,y,z,t;\alpha)}$ and other random quantities.

4. Extension of the Wiener Functionals

4.1. Multiple Parameters

Recall the ***Wiener Stochastic Convolution Integral***:

$$u(t;\alpha) = \int U(t+s)\, dr_s(s;\alpha) \qquad (4.1.1)$$

It is complete for single-parameter Gaussian random functions. The **HPF**'s and **OPF**'s extend this representation to any square-integrable, stationary random function of a single parameter.

We now extend this integral to ***two*** parameters formally by the integral:

$$u(x,y;\alpha) = \int U(x+x_1, y+y_1)\, dr(x_1,y_1;\alpha) \qquad (4.1.2)$$

This integral is taken to be a ***surface*** integral, and $\mathbf{dr(x,y;\alpha)}$ is a surface element whose variance is $\mathbf{d\sigma(x,y)}$ – the ***area*** of the surface element. This heuristic description can be made quite rigorous.[12] Moreover, three parameter integrals are defined as:

$$\begin{aligned} u(x,y,z,t;\alpha) &= \int U(x+x_1, y+y_1, z+z_1, t)\, dr(x_1,y_1,z_1;\alpha) \quad \text{For HIT, t is free} \\ u(x,t,z,y;\alpha) &= \int U(x+x_1, t+t_1, z+z_1, y)\, dr(x_1,t_1,z_1;\alpha) \quad \text{For PPF, y is free} \end{aligned} \qquad (4.1.3)$$

This integral is taken to be a ***volume*** integral, and $\mathbf{dr(x,y,z;\alpha)}$ is a ***volume*** element whose variance is $\mathbf{d\tau(x,y,z)}$ – the volume of the volume element. The first example is stationary and ergodic on **{x,y,z}** and is good for ***Homogeneous Isotropic Turbulence*** (**HIT**) with "**t**" as a free parameter. The second example is stationary and ergodic on **{x,t,z}** and is good for ***Plane Poiseuille Flow*** (**PPF**) with "**y**" as a free parameter.

These 2-parameter and 3-parameter integrals can be used as the basis ***Polynomial Functionals*** (both **HPF** and **OPF**) with no difficulty. The $\mathcal{H}_k[U_k(..);\alpha]$ and $\mathcal{G}_k[U_k(..);\alpha]$ so formed are the main tools used in our study of theoretical turbulence. There remain the tasks of extending these tools to:

1. Vector Values and Multiple Degrees of freedom

4.2. Vector Values and Multiple Degrees of Freedom

Naively, we might construct a ***Vector*** Wiener integral as:

[12] See Wiener and Poduska ScD Thesis '62. Also see RandomTheoryOfTurbulence

$$u^i(t;\alpha) = \int U^i(t+s)\, dr_s(s;\alpha) \qquad (4.2.1)$$

But then, given full knowledge of all $\mathbf{U^i(..)}$, all $\mathbf{u^i(..;\alpha)}$ can be determined uniquely from knowledge of only one – i.e. the $\mathbf{u^i(..;\alpha)}$ are completely ***mutually*** dependent. This leads to the (somewhat murky) notion of "***degrees-of-freedom***" (**DOF**) – i.e. how many ***entirely independent*** quantities are required to represent a given vector random function[13]. This is difficult to answer.

However, for a given number of **DOF**, the appropriate form of the Wiener Integral is:

$$u^i(t;\alpha) = \int U^{i\beta}(t+s)\, dr_s^{\beta}(s;\alpha) \qquad (4.2.2)$$

The sum on β is over the number of **DOF**, and the $\mathbf{dr^{\beta}(s;\alpha)}$ are mutually independent.

Some examples may help:

1. For **PPF**, Mass-Conservation implies that given two velocities, the third is uniquely determined. Thus, the maximum number of **DOF** required is 2. It is possible (but unknown) that the Navier-Stokes equations further reduce the required number of **DOF** to 1.
2. A larger number of **DOF** may be used for convenience with no harm.
3. A similar analysis says that **HIT** requires only 2 **DOF**. In this case, we know that reduction to 1 **DOF** does not yield all solutions.
4. For the **MFD** case of **PPF**, we may require 4 **DOF** to accommodate both the random velocity and **B**-field vectors.

The "degrees-of-freedom" issue adds considerable complexity to the Wiener Polynomial Functional representation of random velocities – $\mathbf{u^i(x,y,z,t;\alpha)}$ and other random quantities. This is somewhat mitigated by the use of Symbolic Algebra software – we use Maple.

4.3. Summary: Wiener Polynomial Functionals

We have presented above the Wiener Machinery appropriate for the study of certain turbulent flows, including **HIT** and **PPF**. The tools are quite complex in detail, but there are computer tools to help the development. Specifically:

1. Polynomial Functionals, $\mathcal{H}_k[U_k(..);\alpha]$ and $\mathcal{G}_k[U_k(..);\alpha]$ which completely represent random turbulent quantities, e.g. $\mathbf{u^i(x,y,z,t;\alpha)}$
2. These Functionals are based on the ***Wiener Stochastic Convolution Integral***, which has been extended to multiple parameters and vector-valued quantities with multiple degrees of freedom.

These Polynomial Functionals have many advantages, chief of which (for us) are:

1. Space and Time Derivatives – essential for solving **PDE**'s – are simply expressed.
2. The Kernels are smooth non-random functions. Once determined, they are known forever.
3. Specific flows can be generated (for a given α) and visualized. This corresponds exactly to a **DNS** study of a flow.

The task now is to apply these tools.

[13] There is a bit more to the story.

5. Galerkin-Optimized Representation of $u^i(x,y,z,t;\alpha)$

5.1. The Basic Idea

The solution to ***turbulent* PPF** velocities (i.e. the ***random* $\mathbf{u^i(x,y,z,t;\alpha)}$**) lies in determining the kernels of the $\mathcal{G}_k[U_k(..);\alpha]$ (i.e. the ***non-random*** deterministic $\mathbf{U_k(..)}$) via the equations of motion.

For our initial implementation, we choose to represent these kernels with truncated expansions and "optimize" the approximation with a ***Ritz-Galerkin*** technique[14]. The process is complex, but can be broken down into simpler steps as follows:

1. Set the ***velocities*** to the **curl** of a ***stream function***, i.e. $\mathbf{u^i(x,y,z,t;\alpha)}$=curl($\mathbf{s^i(x,y,z,t;\alpha)}$), Thus, the Mass-Conservation equation is satisfied identically.
2. Set the stream function to a ***finite truncated*** sum of **OPF** functionals with **1-DOF** thus:

$$u^i(x,t,z,y;\alpha) = \nabla^{i\times j} s^j(x,t,z,y;\alpha) \qquad \text{curl of stream function}$$

$$s^i(x,t,z,y;\alpha) = \sum_{k=0}^{2} s_k^i(x,t,z,y;\alpha) \qquad \text{truncate to three terms, 1-DOF}$$

$$s_k^i(x,t,z,y;\alpha) = \mathcal{G}_k\left[S_k^i(s_1,s_2,\cdots,s_k);\alpha\right] \qquad \text{each term is an OPF}$$

3. Expand the $\mathbf{S^i_k(..)}$ with Hermite Functions[15] on **{x,t,z}** with coefficients as functions of **y**.
4. Expand these **y**-coefficients in selected functions on $\mathbf{y\in[-1,+1]}$ with constant parameters.
5. Insert into the Navier-Stokes to determine an ***Error Residual***.
6. Truncate this ***Error Residual*** to terms in the original stream function representation.
7. Minimize some scalar norm of the ***Error Residual*** with regard to the free parameters.

We call this the **GERM** – ***Galerkin Error Residual Method***. This is a VERY complex process in detail, but in concept it is straightforward.

5.2. Developing the Model

There is much more to be said about selecting the form of these Kernel expansions:

1. Symmetry conditions will greatly reduce the number of expansion parameters. In particular, there are many {**y,z**} symmetries that are applied directly to the kernel.
2. Wall conditions will shape the **y**-function formulation. In particular, mid-channel behavior should be slowly varying while wall behavior should be sharply exponential.
3. We can and do require the mean $\mathcal{G}_0[U_0(..);\alpha]$ equations to be satisfied identically.

[14] There are of course many other techniques to try, including a Quantile function approach described elsewhere.

[15] Hermite functions are Hermite polynomials times the sqrt of the exponential weighting function. There are several other good prospects.

All of this would be impossibly complex for mere mortals to deal with were it not for ***Symbolic Algebra Manipulation*** (**SAM**) systems – we use Maple. Often the Maple expressions are over 10,000 symbols and operators.(!)

The Maple programs generate many sub-models with varying structure of the Stream Function, varying number of Hermite Functions, varying complexity of the **y**-coefficients, etc. Often there are some 200-300 individual numerical parameters to optimize. The Maple program then emits **CPP** statements to be incorporated into a minimization program which can run for days and weeks.

The **CPP** program computes a norm for the ***Error Residual***. Currently this is the square of the error residual integrated over all space and time, and normalized by a measure of the turbulent velocity intensity. This gives a scalar norm to be minimized against all free parameters.

The minimization algorithms are simple Conjugate-Gradient and Powell algorithms, repeated ***ad nausea*** till a minimum is reached. This is done for a range of 161 Reynolds Numbers[16] from 1000 to 10,000,000. Some models give excellent results compared to physical experiment and **DNS**.

5.3. Some Computation Results

I have explored many configurations of my **GERM** formulation. Many have resulted in poor and unphysical results. Some have yielded very satisfactory results over a wide range of **Re**. Here I discuss one of the best results.

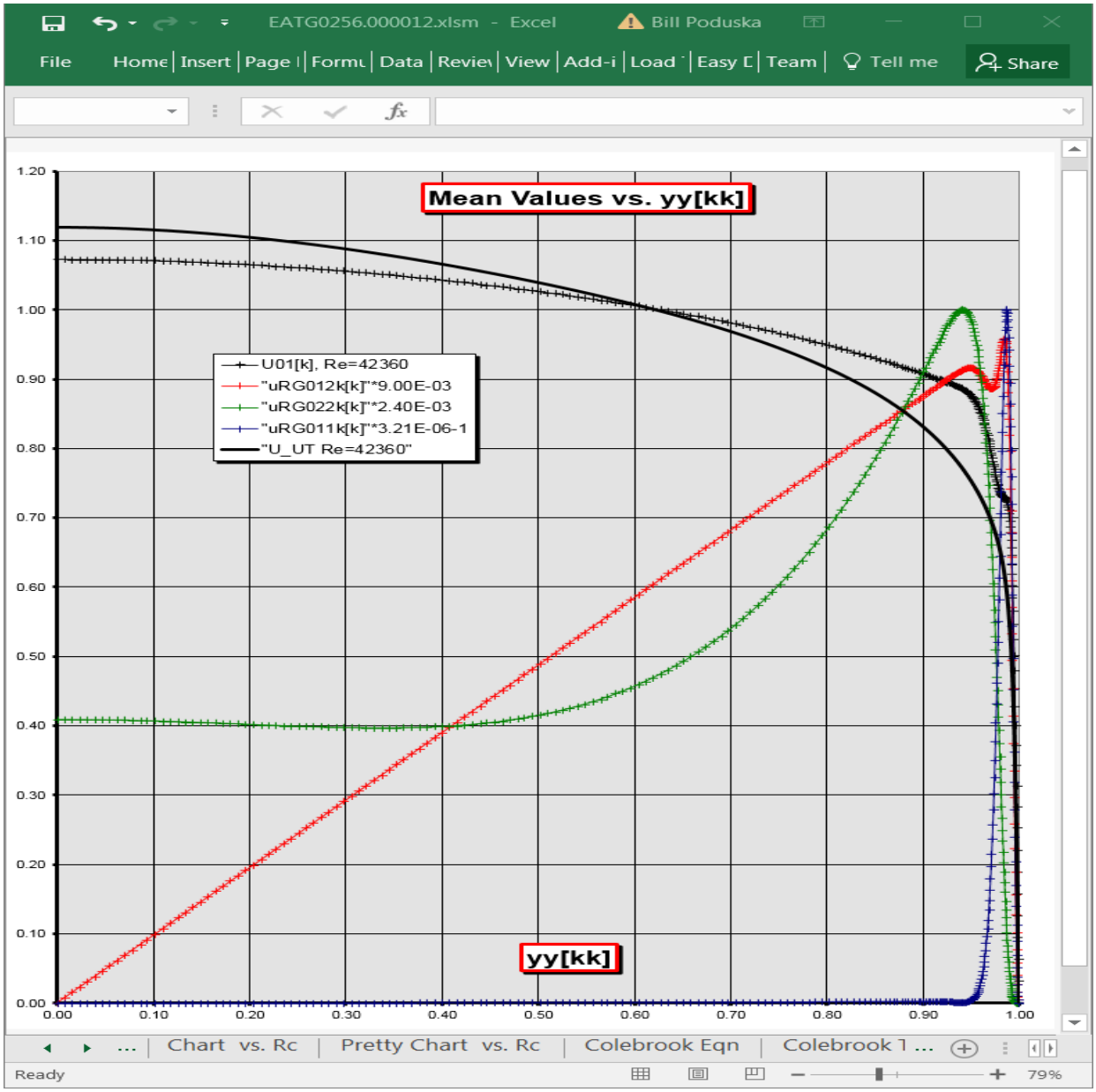
Of course, I might be accused of "cherry-picking" with one eye cocked toward the answer! To this I quickly confess. So, I do not claim this is a ***solution*** but only that the results are very ***suggestive***.

Some details:

1. The $\mathcal{G}_0$**[..]** equations are satisfied by direct integration for $\mathbf{U_1(y)}$.
2. The $\mathcal{G}_1$**[..]** stream function is $\mathbf{[S_1,0,S_3]}$ which we dub the **MS0S** case.
   a. Hermite function expansion on **{x,t,z}** is **{4,4,2}** terms respectively.
   b. **y**-expansion is $\mathbf{\{1,y^2\}}$ in center channel and $\sim \exp\left(c \cdot \sqrt{Re} \cdot y\right)$ at the walls.
3. The $\mathcal{G}_2$**[..]** stream function is also $\mathbf{[S_1,0,S_3]}$.
   a. Hermite function expansion on **{x,t,z}** is **{2,2,2}** terms respectively.
   b. **y**-expansion is same as $\mathcal{G}_1$**[..]** term.
4. The Galerkin style error residual is truncated to the $\mathcal{G}_1$**[..]** and $\mathcal{G}_2$**[..]** terms.
   a. Further truncation is to the Hermite functions in the beginning expansion.
   b. No truncation of the **y**-terms
5. The scalar norm is the Navier-Stokes residual squared and integrated over **{x,t,z,y}**, then normalized by the magnitude of the Reynolds Stresses, i.e. $\mathcal{E}\mathbf{(u_x u_y)}$.
6. This scalar norm is then minimized by variation of all 168 free parameters.

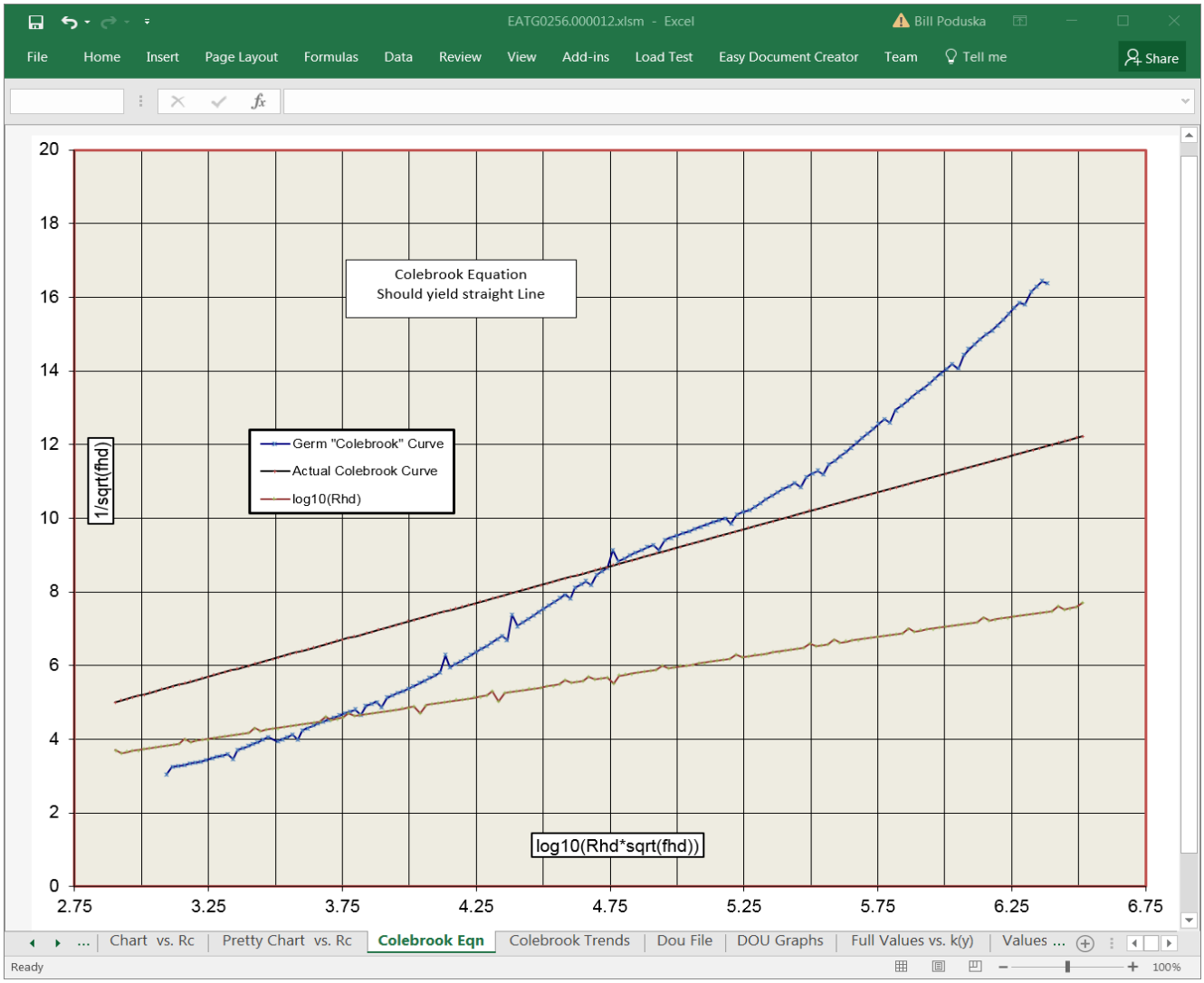
[16] We use Bulk Velocity and Half-Channel width in **Re=LV/ν**.

The minimization procedure is carried out for 161 Re numbers from 1000 to 10,000,000 with very interesting results. Two screenshots illustrate the results: The first screenshot below shows the mean values for our Re=42360 (R_{hd}=4*42360=169440) along with the Mean velocity of the **UT DNS**[17] for their Re=43478 (Re_tau=1995). Also shown are the Reynolds stresses and the mean-square of the streamwise random velocity. The form of the curves seems quite good, but there is the expected "ring-ing" at the walls.



[17] Lee and Moser, Direct numerical simulation of turbulent channel flow up to Re_tau = 5200, 2015, Journal of Fluid Mechanics, vol. 774, pp. 395-415

Another interesting screenshot shows the result of plotting the Colebrook formula, i.e **1/sqrt(f_{hd})** against **log10(R_{hd}*sqrt(f_{hd}))** which should empirically be a straight line.



The pips on our "GERM" curve are just markers: the "down" pips are at 1,2,4,8,10 leading digits of Re. The "up" pips are Re={20000, 42360, 126000} of the UT study.

The results are far from perfect, but – to me at least – they are certainly suggestive.

6. The Quantile Process and GaussXF Transform

6.1. Quantile Functions

Consider a velocity component **u(t;α)** in a **PPF** channel at a certain point (say $\mathbf{x}^k$) and in a certain direction (say $\boldsymbol{\kappa}^i$) so that:

$$u(t;\alpha) = \kappa^i\, u^i(x^k, t;\alpha) \qquad (6.1.1)$$

This random function has a **cdf_u(..)** (***cumulative distribution function for u***) which is:

$$\mathcal{Pr}\{u(t;\alpha) \le \lambda\} = cdf_u(\lambda) \quad \text{stationary, i.e. not-dependent on t} \tag{6.1.2}$$

The range of **$cdf_u(..)$** is **[0,1]** and the domain is **[-inf,+inf]**. Moreover, for physical situations, **$cdf_u(..)$** is smooth and strictly increasing and has as many derivatives as we need.

Now construct a new ***random function*** as:

$$q(t;\alpha) = cdf_u(u(t;\alpha)) \tag{6.1.3}$$

What is the **cdf** of **q**?

$$\begin{aligned} &\mathcal{Pr}\{u(t;\alpha) \le \lambda\} = cdf_u(\lambda) && \text{Standard Definition} \\ &\mathcal{Pr}\{\underbrace{cdf_u(u(t;\alpha))}_{=q(t;\alpha)} \le \underbrace{cdf_u(\lambda)}_{\triangleq \mu}\} = \underbrace{cdf_u(\lambda)}_{\triangleq \mu} && \text{Because } cdf_u(\lambda) \text{ is monotone} \\ &\mathcal{Pr}\{q(t;\alpha) \le \mu\} = \mu && \text{!!! The Uniform Distribution} \end{aligned} \tag{6.1.4}$$

Now the Standard Normal **$cdf_N(..)$** and its inverse (quantile function) are[18]:

$$\begin{aligned} &\Psi(\lambda) = \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{\lambda}{\sqrt{2}}\right)\right] && \text{The "Normal" Distribution} \\ &\Psi^{-1}(\mu) = \sqrt{2}\operatorname{erf}^{-1}(2\mu - 1) && \text{The "Normal" Quantile Function} \end{aligned} \tag{6.1.5}$$

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

$$\phi(t;\alpha) = \Psi^{-1}\left(q(t;\alpha)\right) = \Psi^{-1}\left(cdf_u(u(t;\alpha))\right)$$

The resulting **$\phi(t;\alpha)$** is a ***Normal Gaussian Random Variable*** with **Mean=0** and **Variance=1**! It is also ***stationary*** and ***ergodic*** on "**t**" because **$u(t;\alpha)$** and **$q(t;\alpha)$** are so. Moreover, the Wiener Integral:

$$\phi(t;\alpha) = \int \Phi(t+s)\, dr_s(s;\alpha) \tag{6.1.6}$$

over infinite limits is also Gaussian and can represent ***any*** stationary Gaussian Random Function with the appropriate kernel **$\Phi(..)$**. Finally, every stationary, Gaussian random function with finite mean and variance is ***completely*** specified by its mean and auto-correlation function[19], and any autocorrelation function can be generated with the appropriate kernel to the Wiener Integral.

Summary: We have demonstrated that there exist ***monotone functions*** such that:

$$\begin{aligned} &\phi(t;\alpha) = M_{u2\phi}(u(t;\alpha)) \quad M_{u2\phi}(\lambda) \triangleq \Psi^{-1}(cdf_u(\lambda)) \\ &u(t;\alpha) = M_{\phi 2u}(\phi(t;\alpha)) \quad M_{\phi 2u}(\lambda) \triangleq cdf^{-1}(\Psi(\lambda)) \end{aligned} \tag{6.1.7}$$

So, we can represent **$u(t;\alpha)$** precisely by one Monotone Function **$M_{f2u}(..)$** and one Wiener Kernel **F(..)**. This is a substantial savings over the infinite number of kernels for the **OPF** Polynomial Functionals.

Authors Note: I call this the "**Gaussian Transform**" or **GaussXF** or **GXF**. I have not developed the Maple and CPP code to use with a Galerkin-Optimization algorithm, but it seems to be very promising.

[18] The Wikipedia article "Normal Distribution is quite good.
[19] See Yaglom "Stationary Random Functions", especially section 1.3.

6.2. Application of Quantile Functions to Turbulent Flows

For turbulent **PPF**, we require a velocity $\mathbf{u^i(x,y,z,t;\alpha)}$ representation which:

1. Extends beyond ***Gaussian*** Distributed Random Functions
2. Extends to all four space-time parameters
3. Extends to Multiple Degrees of Freedom

The **GaussXF** process inherently extends beyond simple Gaussian Random Functions. Furthermore, the extension to multiple parameters described in section 4.2 above translates over directly. Extension to Multiple Degrees of Freedom is more complicated, but can be readily done.

For simplicity, consider two components[20] of $\mathbf{u^i(x,y,z,t;\alpha)}$ at a fixed point in the channel $\mathbf{x^k}$ say $\mathbf{u^x(t;\alpha)}$ and $\mathbf{u^y(t;\alpha)}$. As above, we may write:

$$u^x(t;\alpha) = M_{\phi 2 u^x}(\phi^x(t;\alpha))$$
$$u^y(t;\alpha) = M_{\phi 2 u^y}(\phi^y(t;\alpha) \qquad (6.2.1)$$

Here, the two $\mathbf{M_{xxx}(..)}$'s are different functions, and the two $\boldsymbol{\phi}\mathbf{(..)}$'s are Gaussian but **NOT** necessarily independent of each other. We may write the two $\boldsymbol{\phi}\mathbf{(..)}$'s as:

$$\phi^i(t,\alpha) = \int \Psi^{i\beta}(t+s)\, dr^\beta(s,\alpha) \quad \beta \in [1,2] \qquad (6.2.2)$$

So, much like above, we can represent $\mathbf{u(t;\alpha)}$ precisely by two Monotone Functions $\mathbf{M_{f2ui}(..)}$ and one four-component Wiener Kernel Array $\boldsymbol{\Psi}^{\mathbf{i\beta}}\mathbf{(..)}$. This is a substantial savings over the infinite number of kernels for the **OPF** Polynomial Functionals

This is a very promising avenue of approach.

[20] The third velocity $\mathbf{u^z(t;\alpha)}$ is specified by the mass-conservation equation.

LOCAL TABLE OF CONTENTS:

(3052 words)