Why Math Matters:
Rethinking Drone Flight Stability

last revised 02/19 – LEJ

This whitepaper discusses the importance of mathematics in the context of our proprietary Folded Geometry Code (FGC)™. Digital Aerolus UAVs integrate the same advanced FGC™ mathematics used in space navigation as key components of our aircraft flight control and operating system.

OVERVIEW:

Aertos series AUVs achieve stable flight without GPS through proprietary Folded Geometry flight Code (“FGC”). FGC polls the vehicle's sensors and systems to construct a geometric algebraic space with multidimensional operators. These multidimensional vision and system inputs enable FGC to continuously reconstruct its mathematical model and project what is likely to happen next. This unified flight space model is called the drone's Theory of World.

All local viewpoints in the Theory of World unified space are interconnected by transforms, so information from one viewpoint transforms cleanly to any other viewpoint. This system-wide mathematical underpinning resolves problems in flight stability, vision field management, and AI behaviors that today's UAS market increasingly demands.

DISCUSSION:

All Digital Aerolus UAVs in the Aertos series use our proprietary Folded Geometry Flight Code (“FGC”).

FGC-enabled UAVs poll all sensors and systems and use the collected data to calculate a unified flight space. Constructing a geometric algebraic space with multidimensional operators enables the drone to understand its world by applying a consistent, local predictive model derived from various state estimations. Digital Aerolus calls this the Theory of World.

DA's unified approach must consider the following dimensions:

- six dimensions (x, y, z; roll, pitch, yaw) of the drone view (the view from the drone's perspective)
- six dimensions of the world view (the view from the operator's perspective)
- additional similar dimensional sets for each vision system perspective

In contrast, nearly all commercial drones — with the possible exception of classified military hardware — have a non-unified approach. We believe that the non-unified approach, even if can be technically achieved, is prone to inaccuracies and bugs, and therefore less than optimal.

In essence, the non-unified approach is reactive, while the unified approach is predictive. DA's approach constructs a unified multi-dimensional model that maintains all perspectives across all times. Rather than simply reacting to a stream of information and using it to calculate what to do next, FGC manages and unifies numerous multidimensional vision system inputs and continually adapts and reconstructs its mathematical model to project what is likely to happen next.

Digital Aerolus FGC technology performs the optimization for every sensor or actuator as late as possible. Rather than trying to respond to an avalanche of information from individual and often overly optimized small spaces, FGC proactively constructs and continually maintains a unified space model. This requires a unique approach to our mathematics and software. Because we use the same computational space everywhere, all local viewpoints are interconnected by transforms. The information from one viewpoint always transforms to any other viewpoint. Errors and uncertainties also transform cleanly, including the uncertainty inherent in the transform itself.

This approach also allows us to integrate vision and flight-integrated AI behaviors. For example, we can map information from gyroscopes into the same geometric algebraic space used by an optical flow system - and vice versa. More precisely, we can deploy an information filter such as a Kalman filter to fuse the information provided by the gyroscopes and the imaging system. A drone with only these two sensors would construct an orientation model — along with the full state vector, covariance matrix, and its history — to build its Theory of World.

We are particularly interested in further developing these areas:

- determining and correcting sensor drifts;
- mapping or projecting internal images;
- vision-based AI behaviors; and
- movement planning.

Exploring each of these areas of interest requires a different approach. For example, correcting drift behavior requires processing and correcting the true down vector. But, a flying vehicle cannot acquire a true down vector from a purely inertial system.

Consider the problems associated with coordinated turns in airplanes: some external reference is always required. A pressure sensor can provide a reasonably good measurement of height over medium timescales (~10 sec), but such timeframes are too slow for stabilization loops running at >100Hz. The drone’s Theory of World determines a delta between the measurement and a high-quality prediction. As long as the delta is due to an error in orientation, FGC computes a low-frequency correction to the down vector. Doing this keeps the drone stable indefinitely.

Unlike cars or airplanes, drones can rapidly move in multiple different directions at the same time. In gaming terms, drones can strafe right or left, while pitching up and down, while circling — all simultaneously. Rapidly changing multiple behaviors can severely impair a drone's imaging system: it is likely to suffer motion blur resulting in orientation loss unless it is perfectly gimbaled.

And even with perfect gimbaling, the vehicle might not be able to keep up with the flood of rapidly changing information. A unified space, however, internally projects images in advance using information from the avionics. This provides the image processing system with a head start even in situations where motion blur would otherwise be overwhelming. The avionics can also collect information from the image processing system to creating an optical flow sensing pathway. FGC integrates all this information into the drone's Theory of World. This also allows us to easily control the camera gimbal.

This unified approach becomes especially important when multiple drones fly simultaneously. Each vehicle occupies and manages its own space, and those spaces must be connected. Again, by using the proper mathematics and geometry space models, connecting them is a simple (if uncertain) transform.

The control space also remains the same whether the drone flies independently or in concert. FGC maps individual paths, orbits, and waypoints into the overall geometric model. For example, a GPS waypoint containing latitude, longitude, and elevation data becomes a sphere in the model space and is mapped as a geometric object. FGC also models orbits or arcs as simple geometric representations similar to “tubes” that connect the waypoint spheres.

Intersections between the models represent likely collision hazards. Because the models are constructed in the same space as the photogrammetry the imaging system builds, looking for intersections is straightforward. Errors and uncertainties also have geometric representations, and FCG handles these dynamically. Locations in space are represented as spheres with a radius of either the measurement uncertainty or the required accuracy. The transformation includes the uncertainty, resulting in a radius is either larger (for a more uncertain measurement) or smaller (when more accuracy required to offset the transform uncertainty). Thus, the drone’s Theory of World continually includes uncertainty.

This approach becomes particularly relevant in, for example, a highway bridge inspection application, where it is critical to closely approach the target without hitting it. Wind, updrafts, turbulence, and passing high-speed vehicles create a dynamic environment of uncertainty. Imagine a sphere around the drone that represents its knowledge of where it is, and a tube of increasing diameter that represents the probability path the drone should navigate into.

The drone models its understanding of the world as a mathematical expression. Using a single space simplifies reasoning about the whole system, reduces latency, and tracks uncertainties. Should a simple intersection test fail, the drone can immediately execute a safety behavior.

Consider as a thought experiment a hypothetical "blue sky" cluster application where dozens of mapping drones swarm over an urban area:

Individual drones must constantly move in order to maintain even coverage over the target area while avoiding each other and managing microcurrents of wind as well as the myriad of jet wash columns produced by such a swarm. Each vehicle provides ephemeris information about its position and velocity to the swarm, as well video via wireless links. The unified swarm uses this ephemeric information to maintain and manage the swarm and to produce inter-drone photogrammetry. This allows us to construct a shared real-time 3D model landscape of the surface while maintaining the relative positioning of the drones with high accuracy. On the ground, more powerful computers can, for example, texture this model with projected video to provide full "3D video" for remote-reality applications, or to track objects for augmented reality.

Such a swarm would act like the archetypical Beowulf cluster: a network of identical consumer-grade computers PCs networked by software to share processing in order to achieve high-performance parallel computing behavior from inexpensive computer hardware. In our example, each drone behaves like a node in the cluster that contributes its information into the overall computational space.

Data always flows from the bottom up - from each drone up towards the global model constructed by the software, which then projects instructions back to the drones. Using such a single space metaphor simplifies reasoning about the whole system, reduces latency, and tracks uncertainties.

SUMMARY:

In summary, Digital Aerolus has invented and constructed a set of complementary technologies based on a consistent system-wide mathematical underpinning that resolve the problems in flight stability, vision field management, and AI behaviors that today's growing UAS market increasingly demands.

* * * * * * *

Understanding how counting dimensions works requires distinguishing between dimensions and degrees of freedom.

For example, FGC manages:

- *the 6 dimensions of the drone position and orientation*
- *their 6 1st derivatives in their respective tangent spaces as they constantly change for velocity and angular velocity.*
- *their 12 2nd derivatives for acceleration (thrust, angular acceleration, torque)*
- *and that's before they get moved around.*

In essence, our discussion centers on how to understand information. Every component of the system must remain on the same communication plane with respect to how things are calculated, and must retain all support contexts like units, scales and timing . If the system accomplishes this high level communication, then moving from one view to another remains a simple process regardless of how much information that shift requires.

This moment-by-moment information migration process must be based on deterministic, well understood, and low-risk mathematical objects. It must also be elegant and seamless, and must never depend upon custom pieces of code authored independently by different designers with different objectives for their sensors or readouts.

As a simple example, think of the set of books and accounting for a project, sitting inside the set of books and accounting for a division, sitting inside the set of books and accounting for a company. If every project does things differently, reconciling everything at higher levels becomes increasingly difficult -- if not impossible. Then, imagine two companies merging and trying to reconcile their independent systems.

* * * * * * *