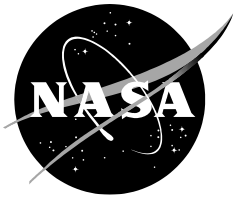
NASA/SP- 20205006016



Summer 2020 Intern Presentation

Blake Hartwell
Universities Space Research Association
Ames, Moffett Field

Jose Sebastian Monzon
Universities Space Research Association
Ames, Moffett Field

Kevin Curt Oran
Universities Space Research Association
Ames, Moffett Field

Katerina Lee Quinn
Universities Space Research Association
Ames, Moffett Field

John Daniel Schroebel
Volunteer Internship Program
Ames, Moffett Field

Jocelyn Sun
Universities Space Research Association
Ames, Moffett Field

Simon A Vecchioni
Universities Space Research Association
Ames, Moffett Field

Jason Marshall Chapman
Universities Space Research Association
Ames, Moffett Field

August 2020

NASA STI Program Report Series

The NASA STI Program collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NTRS Registered and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA Programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include organizing and publishing research results, distributing specialized research announcements and feeds, providing information desk and personal search support, and enabling data exchange services.

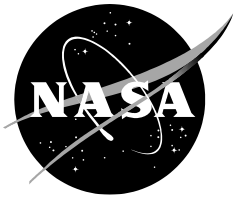
For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at http://www.sti.nasa.gov

- E-mail your question to help@sti.nasa.gov

- Phone the NASA STI Information Desk at 757-864-9658

NASA/SP- 20205006016



Summer 2020 Intern Presentation

Blake Hartwell
Universities Space Research Association
Ames, Moffett Field

Jose Sebastian Monzon
Universities Space Research Association
Ames, Moffett Field

Kevin Curt Oran
Universities Space Research Association
Ames, Moffett Field

Katerina Lee Quinn
Universities Space Research Association
Ames, Moffett Field

John Daniel Schroebel
Volunteer Internship Program
Ames, Moffett Field

Jocelyn Sun
Universities Space Research Association
Ames, Moffett Field

Simon A Vecchioni
Universities Space Research Association
Ames, Moffett Field

Jason Marshall Chapman
Universities Space Research Association
Ames, Moffett Field

Prepared for
2020 Summer Intern Virtual Presentations
Moffett Field, CA August 2020

National Aeronautics and
Space Administration

*NASA Ames Research
Center*

August, 2020

Multi-Objective Flight Control of High Aspect Ratio Flexible Wing Aircraft

Spring – Summer 2020

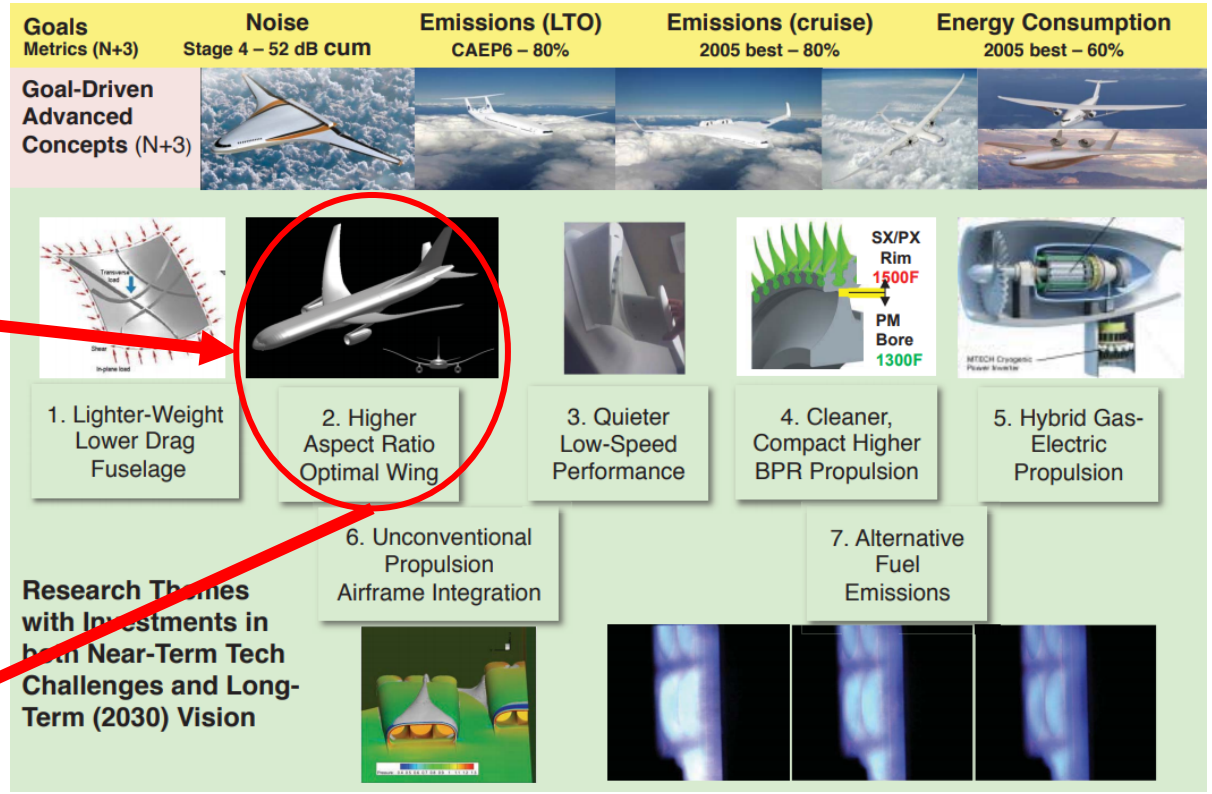
Blake Hartwell NASA ARC Intern (USRA)
Dr. Nhan Nguyen Senior Research Scientist and Technical Group Lead – TI Division, ACES Group

Introduction Project Scope

Aeronautics Research Mission Directorate [1]

STRATEGIC THRUSTS
ARMD Research is Organized into Six Strategic Thrusts
Strategic Thrust 1: Safe, Efficient Growth in Global Operations
Strategic Thrust 2: Innovation in Commercial Supersonic Aircraft
Strategic Thrust 3: Ultra-Efficient Subsonic Transports
Strategic Thrust 4: Safe, Quiet, and Affordable Vertical Lift Air Vehicles
Strategic Thrust 5: In-Time System-Wide Safety Assurance
Strategic Thrust 6: Assured Autonomy for Aviation Transformation

Advanced Air Transport Technology (AATT) [2]

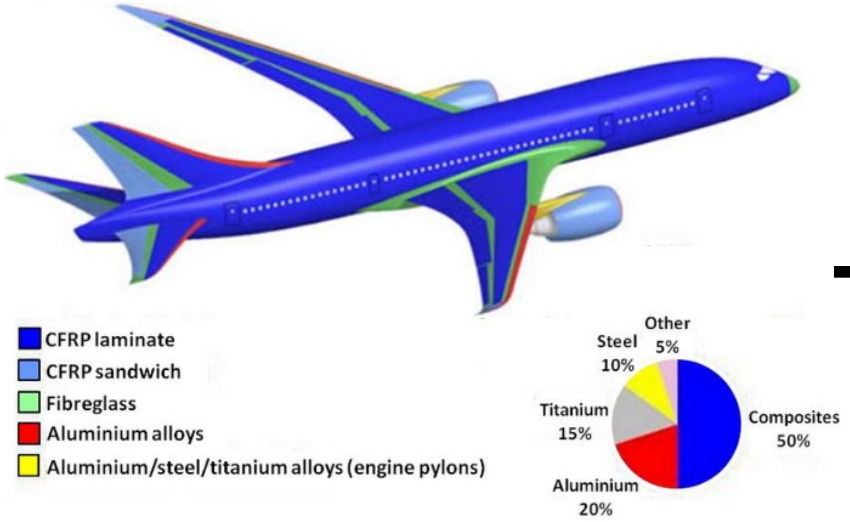


- Next-generation aircraft designed with higher aspect ratio wings to improve aerodynamic efficiency

Boeing SUGAR Volt concept aircraft

Background Flexible Wing Aircraft

- Significant weight savings using advanced composite technology enables high aspect ratio wing designs.
- Higher aspect ratio and reduced material stiffness increases wing flexibility.



Composite structures [3]

+

High aspect ratio wings

=

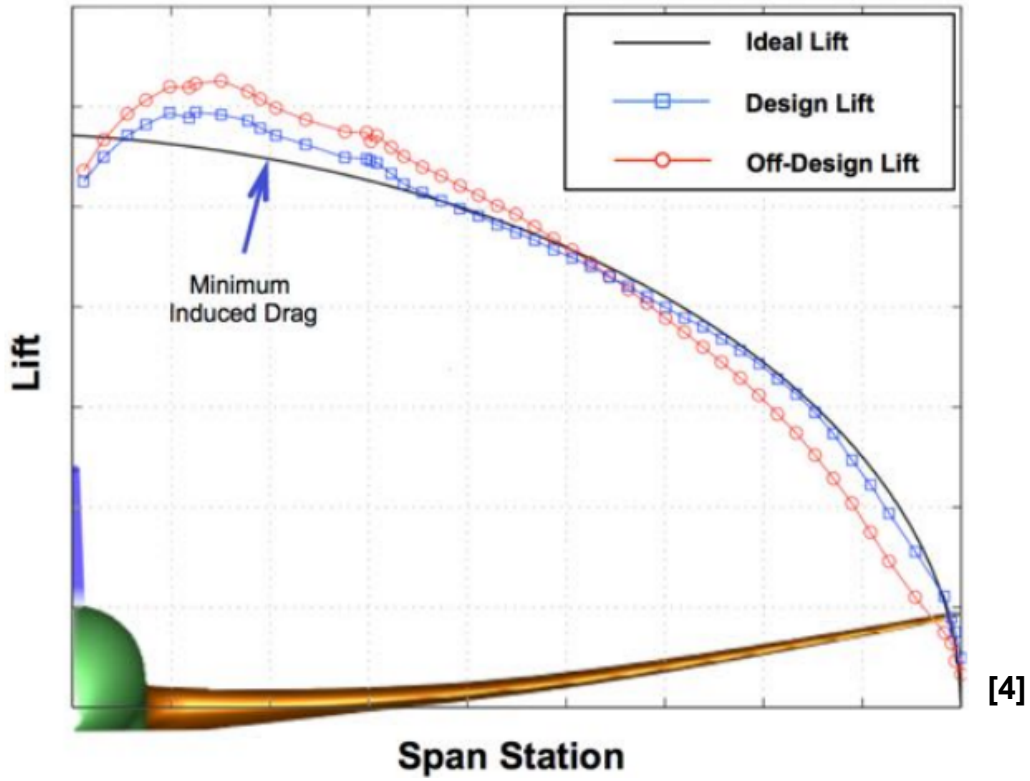
- **Increased fuel efficiency**
- **Weight reductions**
- **Higher wing flexibility**

Background Flexible Wing Aircraft

Higher wing flexibility impacts…

Aerodynamics

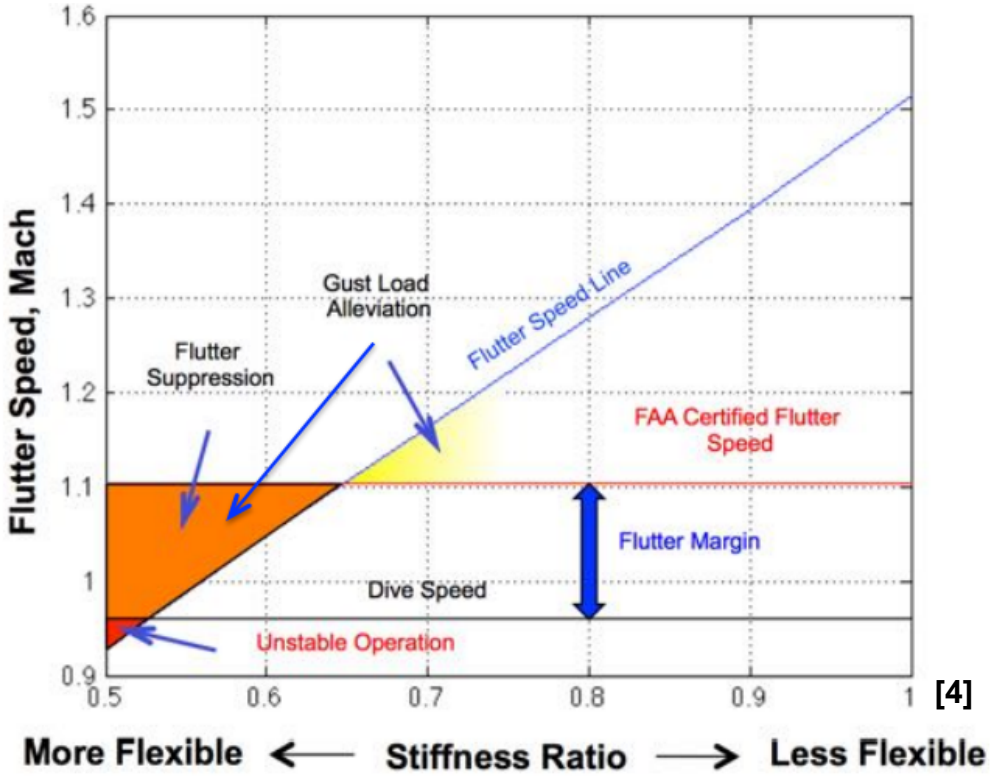
- Increased wing deflection at off-design flight conditions causes suboptimal lift distribution – i.e., increased drag, higher fuel consumption



[4]

Flight Loads, Ride Quality, and Stability

- Increased wing flexibility exacerbates gust/turbulence response and decreases stability margins

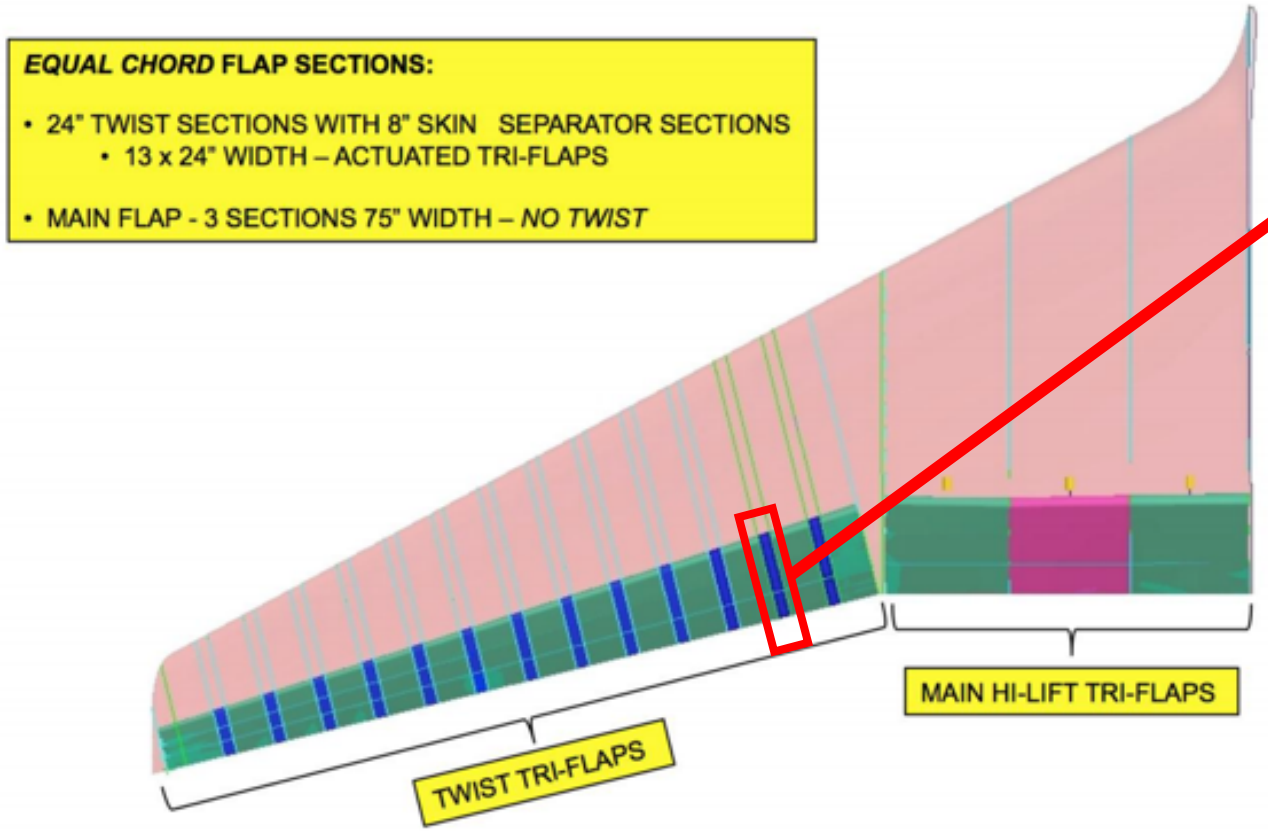


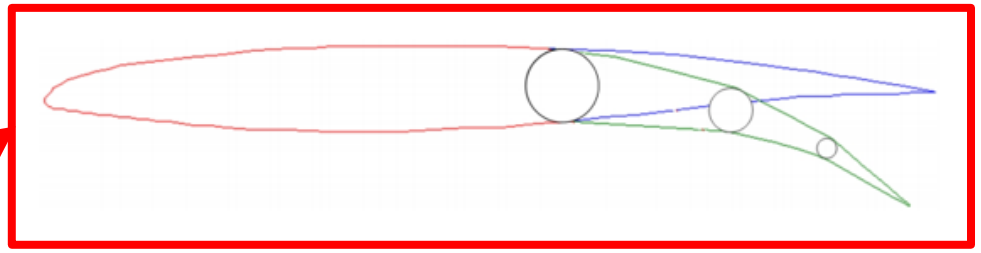
[4]

Performance Adaptive Aeroelastic Wing (PAAW) Technology

Variable Camber Continuous Trailing Edge Flaps (VCCTEF) [5]

- Developed by NASA and Boeing Research & Technology to provide flexible high aspect ratio wings with aeroelastic compensation



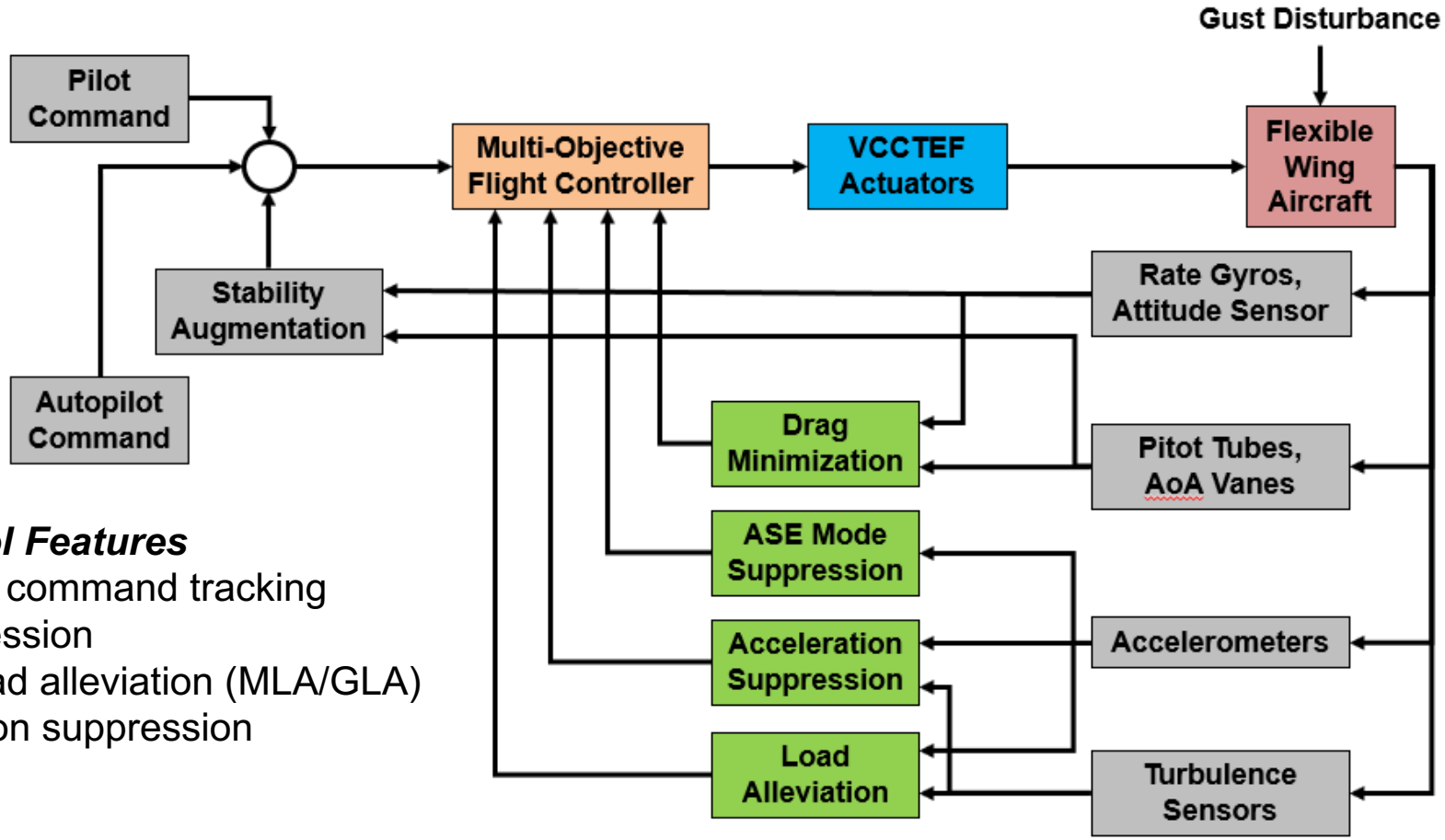


Flap segment cross-section

- Continuous trailing edge enables spanwise tailoring of lift distribution
- Multi-segment variable camber enables chordwise tailoring of pressure distribution on upper and lower wing surfaces

Multi-Objective Flight Control

- Systems with multi-functional flight control surfaces use a multi-objective flight control (MOC) design.
- MOC is an optimal control design that accounts for multiple competing flight objectives when formulating optimal control solutions, all while providing ideal compromises between such requirements.



Control Features

- Conventional pilot command tracking
- ASE mode suppression
- Maneuver/gust load alleviation (MLA/GLA)
- Aircraft acceleration suppression
- Drag minimization

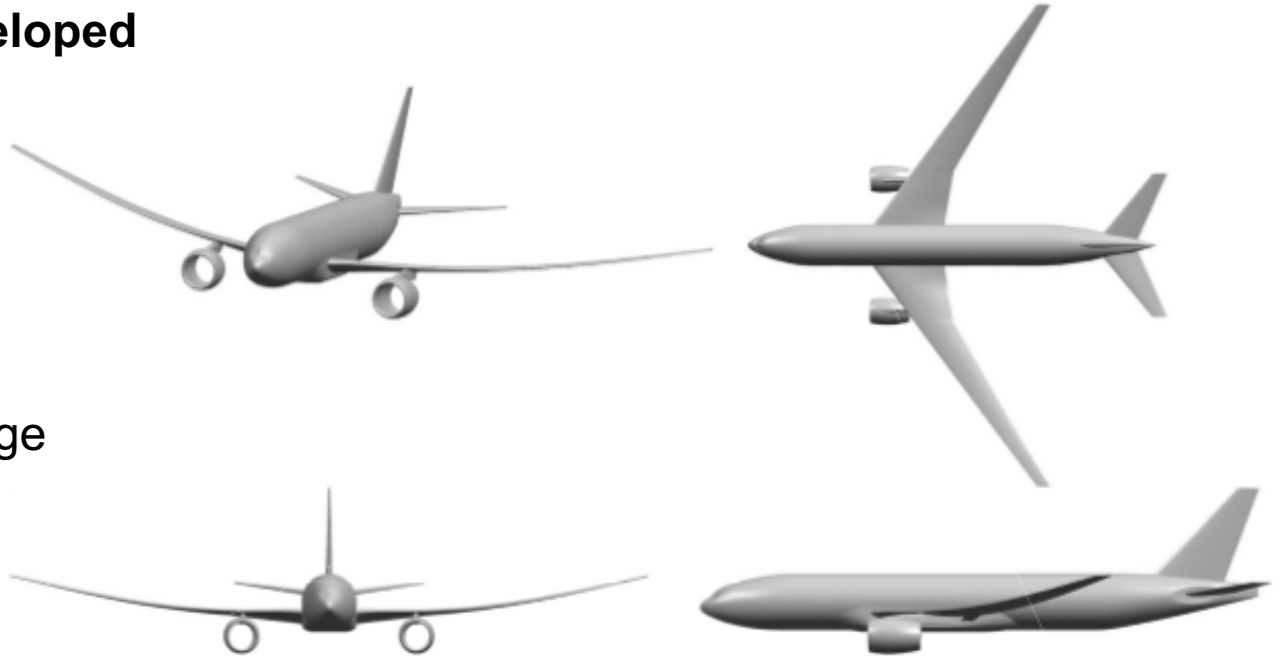
Objectives Spring-Summer 2020

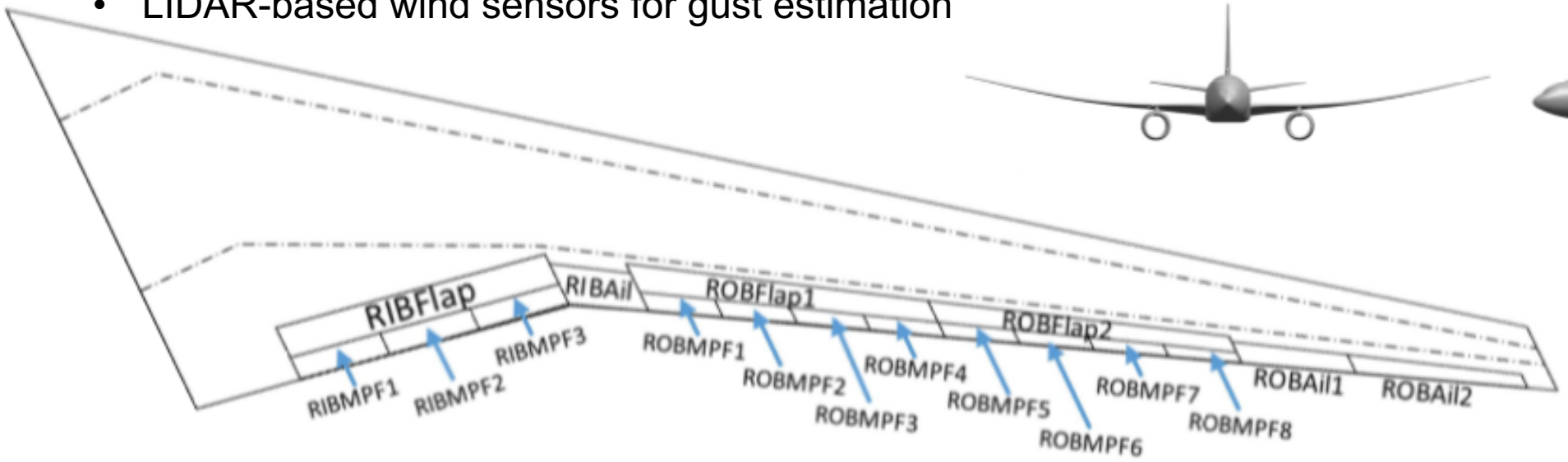
- Design a LQG multi-objective flight controller for the Common Research Model (CRM 13) provided by Boeing Research & Technology.
- Implement the flight controller in a nonlinear, 6DoF Simulink model provided by Boeing.
- Run simulations to assess MOC performance metrics, and validate merits associated with flight objectives.

Common Research Model (CRM13)

777-like configuration with a 13.5 aspect ratio developed by Boeing Research & Technology [6]

- **Distributed control surfaces**
  - 17 control surfaces/wing, 1 elevator, 1 rudder
- **Advanced sensing approaches**
  - Distributed accelerometers: 38/wing + 5 fuselage
  - LIDAR-based wind sensors for gust estimation





Plant Description of Linear CRM13 Model

Gust-Disturbed, ASE State Space Representation

$$\begin{bmatrix}\dot{x_r}\\ \dot{x_e}\end{bmatrix} = Ax + Bu + w = \begin{bmatrix}A_{rr} & A_{re}\\ A_{er} & A_{ee}\end{bmatrix}\begin{bmatrix}x_r\\ x_e\end{bmatrix} + \begin{bmatrix}B_{ru_r} & B_{ru_e} & B_{rd}\\ B_{eu_r} & B_{eu_e} & B_{ed}\end{bmatrix}\begin{bmatrix}u_r\\ u_e\\ d = \begin{bmatrix}g\\ \dot{g}\end{bmatrix}\end{bmatrix}$$

Plant Output from Full Rigid Body Access & Accelerometers

$$y = \begin{bmatrix}x_r\\ a\\ n\\ P\end{bmatrix} = \begin{bmatrix}I & 0\\ a_{x_r} & a_{x_e}\\ n_{x_r} & n_{x_e}\\ P_{x_r} & P_{x_e}\end{bmatrix}\begin{bmatrix}x_r\\ x_e\end{bmatrix} + \begin{bmatrix}0 & 0 & 0\\ a_{u_r} & a_{u_e} & a_d\\ n_{u_r} & n_{u_e} & n_d\\ P_{u_r} & P_{u_e} & P_d\end{bmatrix}\begin{bmatrix}u_r\\ u_e\\ d = \begin{bmatrix}g\\ \dot{g}\end{bmatrix}\end{bmatrix}$$

Load Responses

$$P = P_x x + P_u u + P_d d$$

Fuselage Acceleration Measurements

$$a = a_x x + a_u u + a_d d$$

Wing Acceleration Measurements

$$n = n_x x + n_u u + n_d d$$

Drag Model

$$\Delta C_D = C_{D_x} x + C_{D_u} u + x^T C_{D_{x^2}} x + x^T C_{D_{xu}} u + u^T C_{D_{u^2}} u$$

Observer Designs for Full-State Estimation

Simple Kalman Filter

$$\begin{bmatrix}\dot{\widehat{x_r}}\\ \dot{\widehat{x_e}}\end{bmatrix} = \left(\begin{bmatrix}A_{rr} & A_{re}\\ A_{er} & A_{ee}\end{bmatrix} - L\begin{bmatrix}I & 0\\ a_{x_r} & a_{x_e}\\ n_{x_r} & n_{x_e}\end{bmatrix}\right)\begin{bmatrix}\widehat{x_r}\\ \widehat{x_e}\end{bmatrix} + \left[\left(\begin{bmatrix}B_{ru_r} & B_{ru_e} & B_{rd}\\ B_{eu_r} & B_{eu_e} & B_{ed}\end{bmatrix} - L\begin{bmatrix}0 & 0 & 0\\ a_{u_r} & a_{u_e} & a_d\\ n_{u_r} & n_{u_e} & n_d\end{bmatrix}\right) \quad L\right]\begin{bmatrix}u_r\\ u_e\\ \hat{d}\\ y^*\end{bmatrix}$$

- The gust estimate $\hat{d}$ is obtained from LIDAR measurements available in the Boeing simulation model

Extended State Kalman Filter with Gust Estimation

$$\frac{d}{dt}\begin{bmatrix}\hat{x}\\ \hat{d} = \begin{bmatrix}\hat{g}\\ \dot{\hat{g}}\end{bmatrix}\end{bmatrix} = \begin{bmatrix}A & B_d\\ \begin{matrix}0 & 0\\ 0 & 0\end{matrix} & \begin{matrix}1\\ 0\end{matrix}\end{bmatrix}\begin{bmatrix}\hat{x}\\ \hat{d}\end{bmatrix} + \begin{bmatrix}B_{u_r} & B_{u_e}\\ \begin{matrix}0 & 0\\ 0 & 0\end{matrix}\end{bmatrix}\begin{bmatrix}u_r\\ u_e\end{bmatrix} + L(y - \hat{y})$$

$$\hat{y} = \begin{bmatrix}C & D_d\end{bmatrix}\begin{bmatrix}\hat{x}\\ \hat{d}\end{bmatrix} + \begin{bmatrix}D_{u_r} & D_{u_e}\end{bmatrix}\begin{bmatrix}u_r\\ u_e\end{bmatrix}$$

Recursive Least Squares Method for Gust Estimation

$$\dot{\widehat{w_1}} = I_r\dot{\widehat{w_r}} = -\Gamma_r\left({G_r}^{-1}\varepsilon_r + \sigma_r\widehat{w_1}\right)$$

$$\dot{\widehat{w_2}} = I_e\dot{\widehat{w_e}} = -\Gamma_e\left({a_{w_e}}^T\left(a_{w_e}{a_{w_e}}^T\right)^{-1}\varepsilon_e + \sigma_e\widehat{w_2}\right)$$

Multi-Objective Optimal Control Formulation

Cost Function

Command tracking and ASE mode suppression

Acceleration suppression

Drag minimization

$$J = \lim_{t_f \to \infty} \frac{1}{2} \int_0^{t_f} [\overbrace{(G_x\hat{x} - G_r r)^T Q (G_x\hat{x} - G_r r)}^{\text{Command tracking and ASE mode suppression}} + \underbrace{u^T R u}_{\text{Actuator expenditure}} + \overbrace{n^T Q_n n}^{\text{Acceleration suppression}} + \underbrace{P^T Q_P P}_{\text{Loading response (GLA/MLA)}} + \overbrace{Q_D \Delta C_D}^{\text{Drag minimization}}] \, dt$$

Actuator expenditure

Loading response (GLA/MLA)

LQG Optimal Control Law

$$u = K_x\hat{x} + K_r r + K_d\hat{d} + \Lambda_0$$

$$K_x = -\bar{R}^{-1}\left(B_u{}^T W + n_u{}^T Q_n n_x + P_u{}^T Q_P P_x + \frac{1}{2} Q_D C_{D_{xu}}{}^T\right)$$

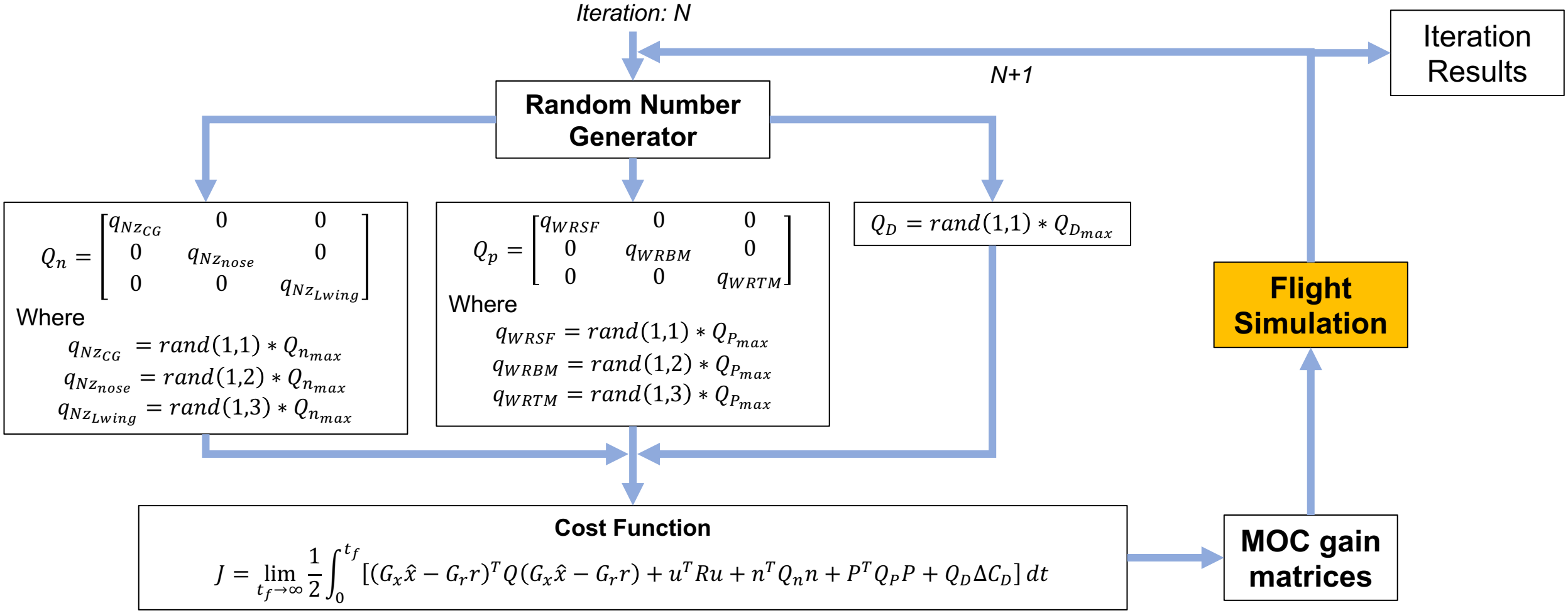
$$K_r = -\bar{R}^{-1} B_u{}^T V_r$$

$$K_d = -\bar{R}^{-1}\left(B_u{}^T V_d + n_u{}^T Q_n n_d + P_u{}^T Q_P P_d\right)$$

$$\Lambda_0 = -\bar{R}^{-1}\left(B_u{}^T V_0 + \frac{1}{2} Q_D C_{D_u}{}^T\right)$$

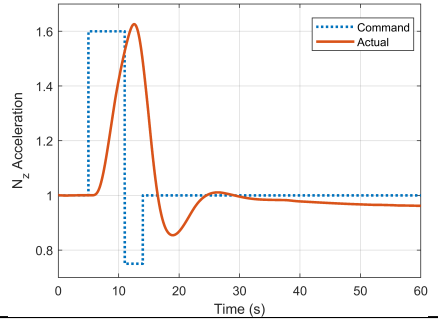
Iterative Weights Selection

- Iterative MOC tuning performed by programming gain calculations inside a FOR loop and using a random number generator to assign cost function weights for each iteration.

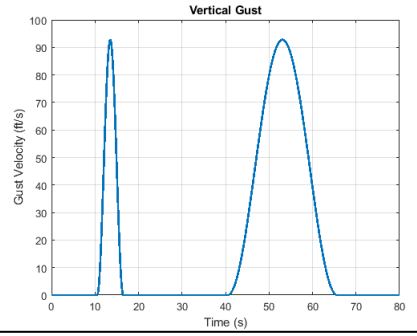


Flight Simulations

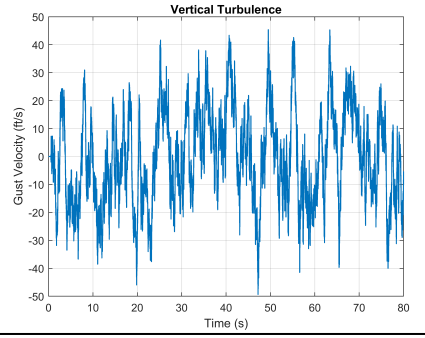
- **Unstable Flight Conditions (1.6g Pitch-Up Maneuver)**
  - Alt. = 21,256 ft
  - Weight = 383,546 lb (Fuel Empty)
  - M = 0.88



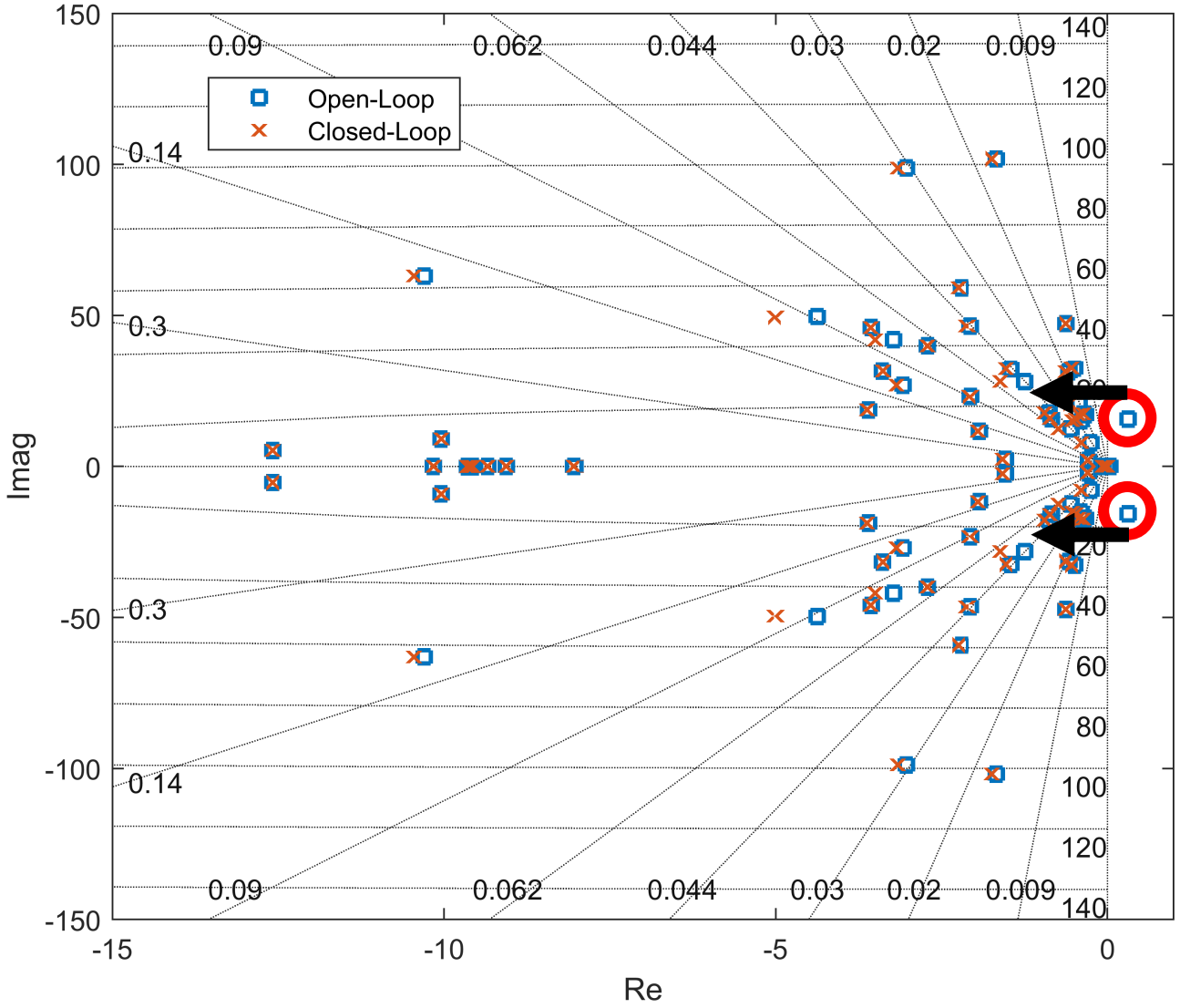
- **Discrete Gust Encounters**
  - Alt. = 36,983 ft
  - Weight = 581,497 lb
  - Mach = 0.85



- **Turbulence Encounter**
  - Alt. = 36,983 ft
  - Weight = 581,497 lb
  - Mach = 0.85



Flight Simulations Unstable Flight Conditions



Flight Conditions

- Alt. = 21,256 ft
- Weight = 383,546 lb (Fuel Empty)
- M = 0.88

MOC Design V_0.1 – ASE Mode Suppression ON

Modal Rate: $\dot{q}_e = \frac{1}{\zeta}$

Modal Deflection: $q_e = \frac{\omega^2}{\zeta}$

$Q_P = [0; \quad 0; \quad 0]$
$Q_n = [0; \quad 0; \quad 0]$
$Q_D = 0$

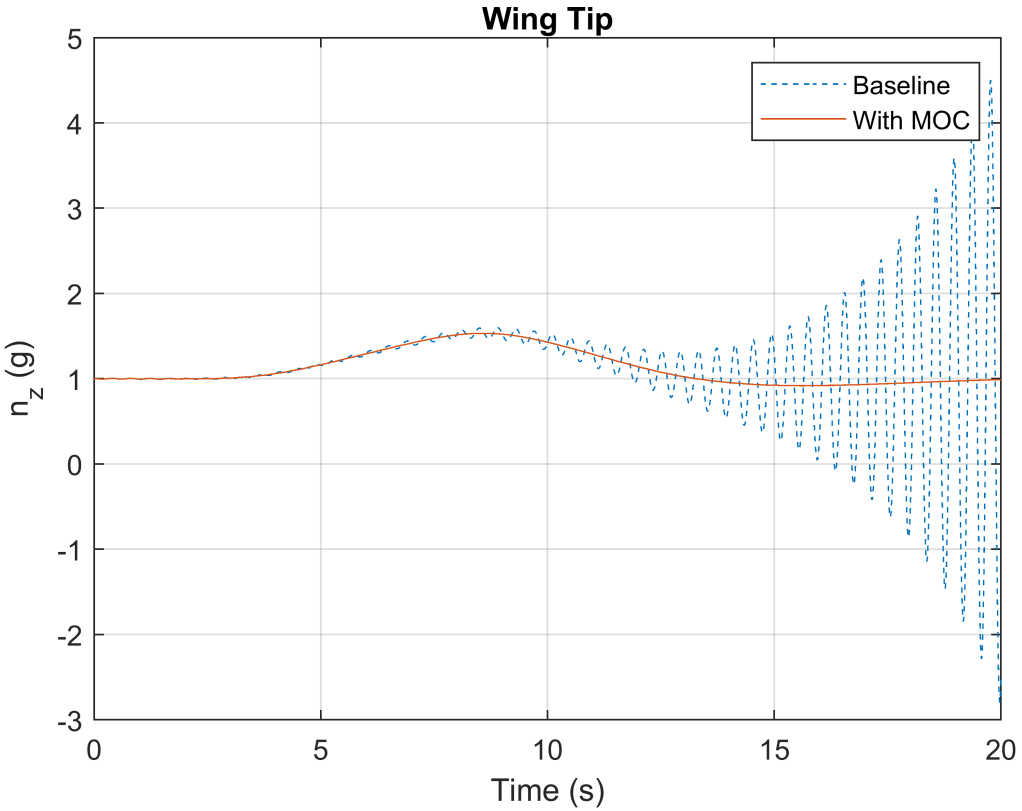
- Unstable poles shifted to left side of s-plane when applying ASE mode suppression.

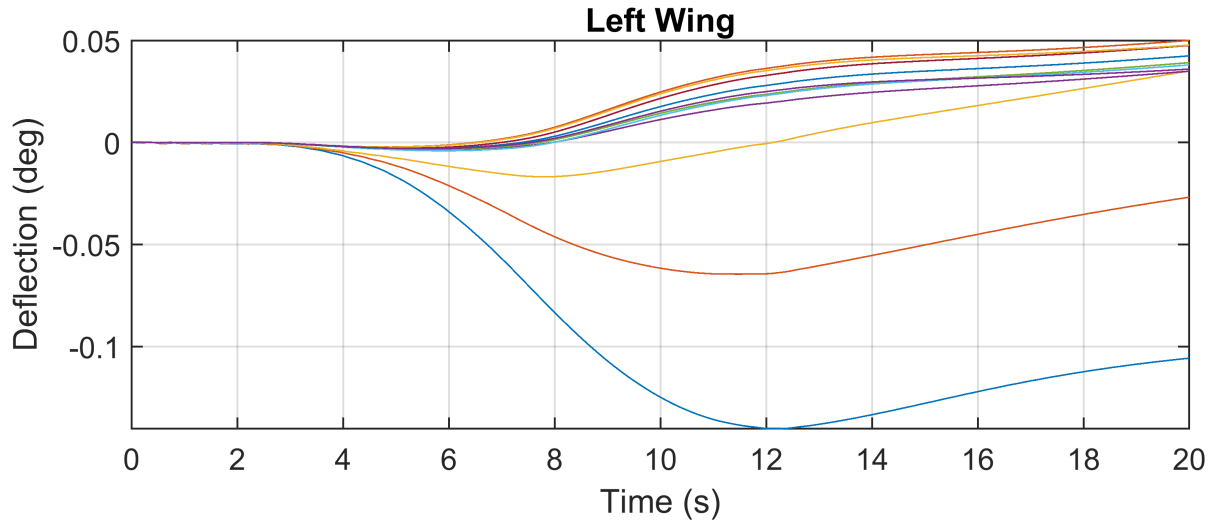
Flight Simulations Unstable Flight Conditions

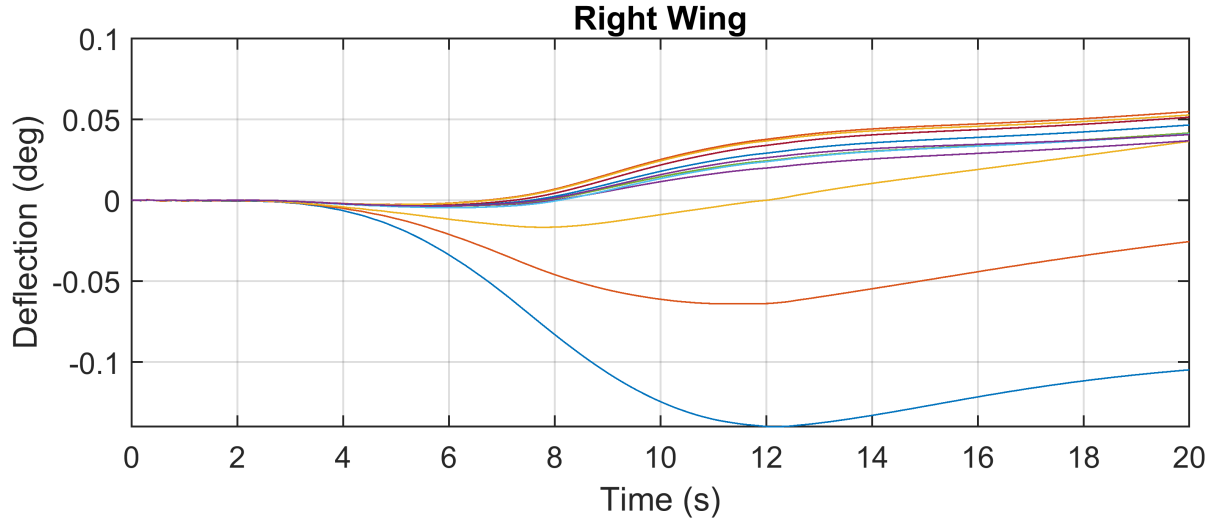
MOC Design V_0.1 – ASE Mode Suppression ON

Modal Rate: $\dot{q}_e = \frac{1}{\zeta}$

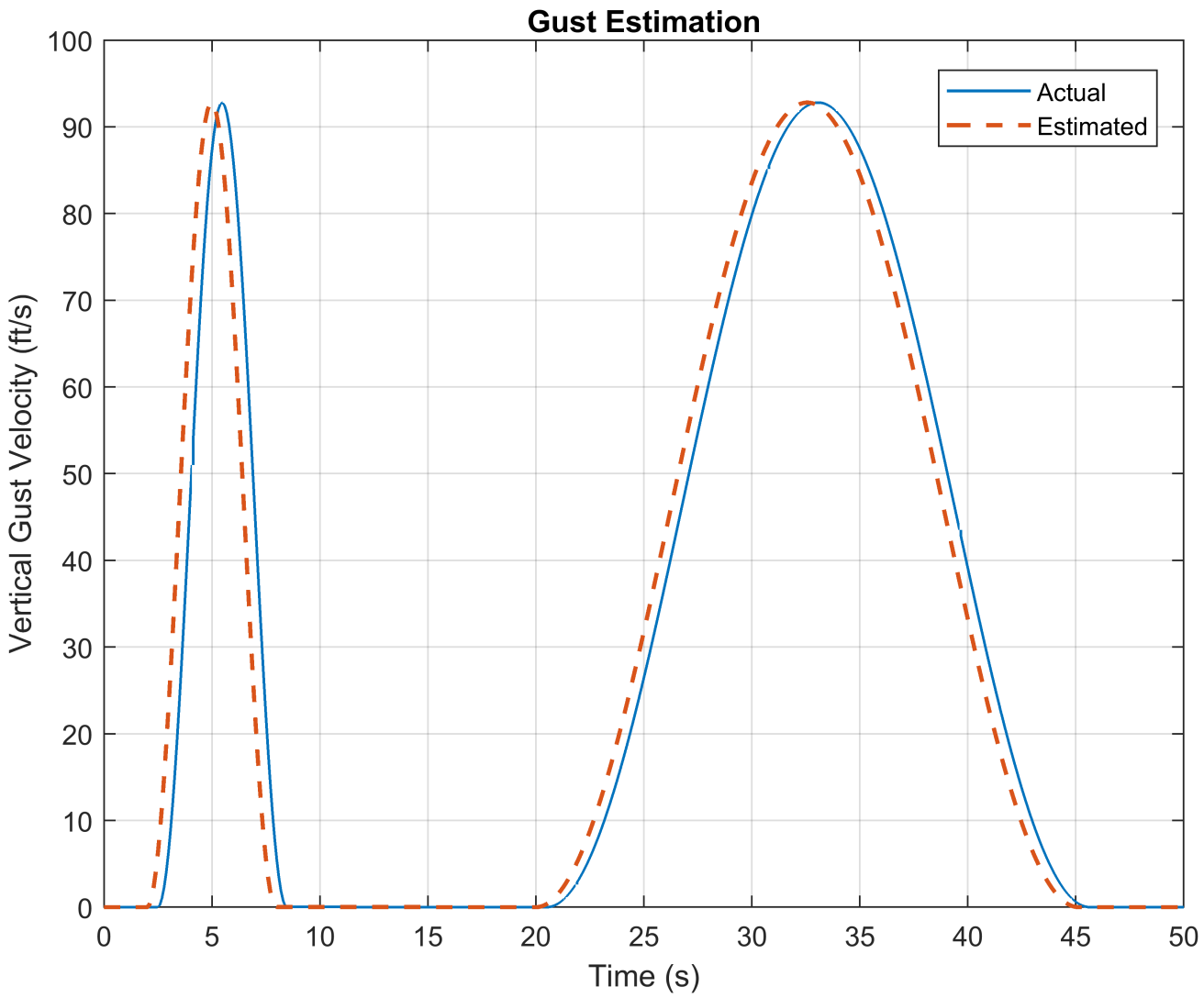
Modal Deflection: $q_e = \frac{\omega^2}{\zeta}$







Flight Simulations Discrete Gust Encounters



Flight Conditions

- Alt. = 36,983 ft
- Weight = 581,497 lb
- Mach = 0.85

Design V_4.2 – Gust Load Alleviation ON

$Q_P = [0; \quad 60; \quad 10]$

$Q_n = [0; \quad 0; \quad 0]$

$Q_D = 0$

- Gusts estimated using onboard LIDAR wind sensors

Flight Simulations Discrete Gust Encounters

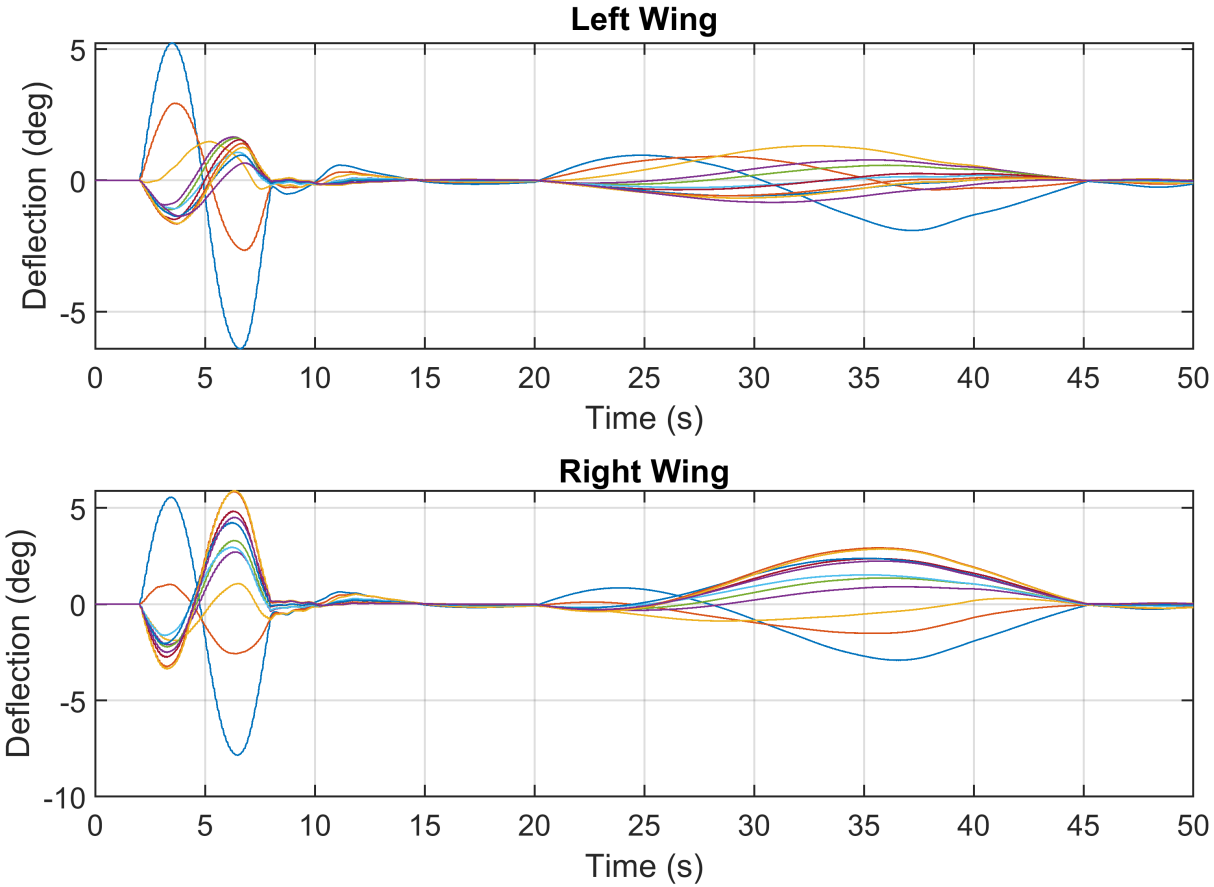
Design V_4.2 – Gust Load Alleviation ON

$Q_P = [0; \quad 60; \quad 10]$

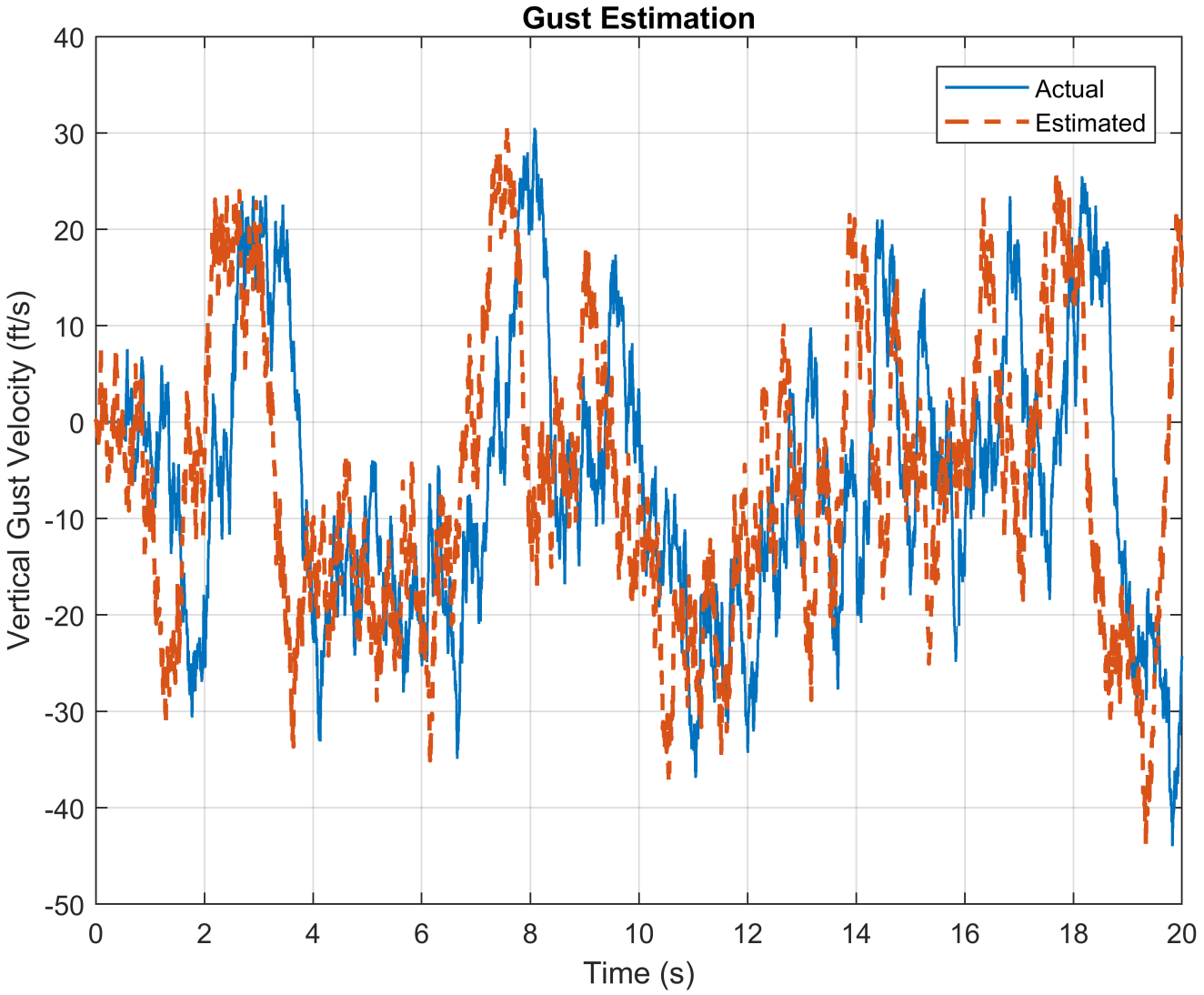
$Q_n = [0; \quad 0; \quad 0]$

$Q_D = 0$

	Baseline	With MOC	Impact (%)
Max WRBM (ft-lb)	5.896E+06	4.674E+06	-20.72
Max WRTM (ft-lb)	4.854E+05	3.807E+05	-21.57
Max WRSF (lb)	5.754E+03	4.403E+03	-23.48
Max n_z at CG (g's)	1.573	1.549	-1.50
Max n_z Left Wing (g)	1.658	1.633	-1.50
Max n_z Pilot Seat (g)	2.015	1.899	-5.76
Avg. L/D Ratio	21.493	21.834	1.59



Flight Simulations Turbulence Encounter



Flight Conditions

- Alt. = 36,983 ft
- Weight = 581,497 lb
- Mach = 0.85

Design V_4.2 – Acceleration Suppression ON

$Q_P = [0; \quad 0; \quad 0]$

$Q_n = [5; \quad 5; \quad 5]$

$Q_D = 0$

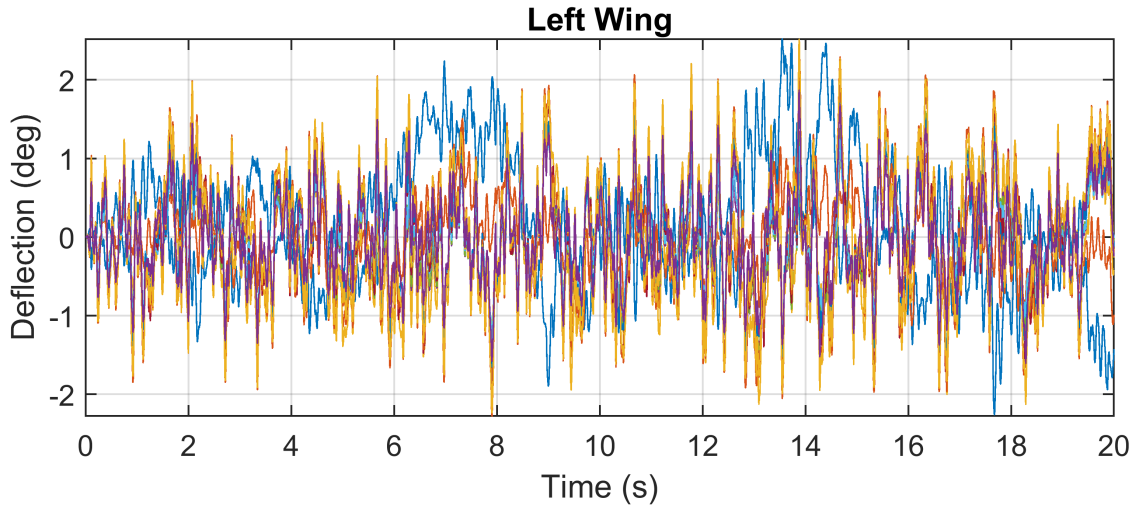
- Turbulence estimated using onboard LIDAR wind sensors

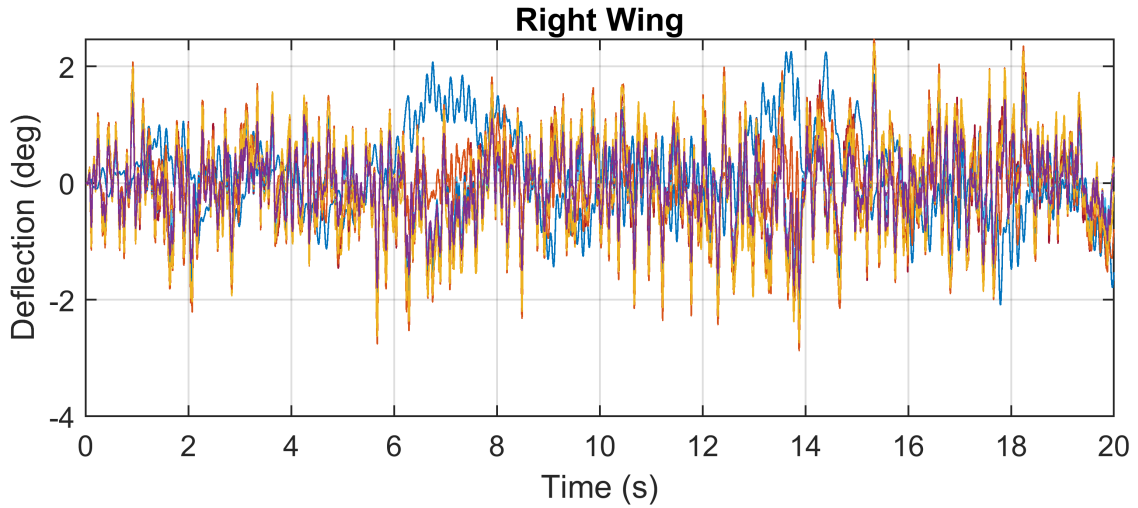
Flight Simulations Turbulence Encounter

Design V_4.2 – Acceleration Suppression ON

$Q_P = [0; \quad 0; \quad 0]$
$Q_n = [5; \quad 5; \quad 5]$
$Q_D = 0$

	Baseline	With MOC	Impact (%)
Max WRBM (ft-lb)	3.296E+07	3.126E+07	-5.16
Max WRTM (ft-lb)	7.630E+06	7.165E+06	-6.09
Max WRSF (lb)	4.024E+04	3.784E+04	-5.96
Max n_z at CG (g's)	2.154	2.159E+00	0.24
Max n_z Left Wing (g)	2.100	2.061E+00	-1.83
Max n_z Pilot Seat (g)	2.062	2.050	-0.54
Avg. L/D Ratio	17.012	17.258	1.45





Conclusions

- Multi-objective optimal control provides high aspect ratio flexible wing aircraft with aeroelastic compensation.
- Aeroservoelastic mode suppression stabilizes wing flutter modes during unstable aircraft flight conditions.
- Gust load alleviation provided >20% reductions in maximum wing loading during discrete gust encounters
- Gust load alleviation is more effective for discrete gust encounters, acceleration suppression is more effective for turbulence encounters.

Acknowledgments

- **Dr. Nhan Nguyen –** Senior Research Scientist and Technical Group Lead of Advanced Control and Evolvable Systems Group, Intelligent Systems Division, NASA Ames

- **Dr. Kelley Hashemi –** Aerospace Engineer, Intelligent Systems Division, NASA Ames

- **Dr. Jack Quindlen -** GNC&A Engineer, Boeing Research & Technology

- **Universities Space Research Association**

References

[1] National Aeronautics and Space Administration, “NASA Aeronautics Strategic Implementation Plan”, www.nasa.gov

[2] Viken, S., “Advanced Air Vehicles Program (AAVP): Advanced Air Transport Technology Project (AATT)”, www.nasa.gov, March 25, 2015.

[3] Wanhill, R., “Boeing 787: The Boeing Company”, 11/01/2017.

[4] Nguyen, N., “Flight Control of Flexible Aircraft”, NESC GNC Meeting at NASA ARC, January 25, 2017.

[5] Nguyen, N., Ting, E., and Chaparro, D., “Development of an Integrated Nonlinear Aeroservoelastic Flight Dynamic Model of the NASA Generic Transport Model,” AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA-2018-2210, January 2018.

[6] Boeing Research & Technology, “Full-Span CRM-13 Aircraft Design”.

Questions?

Observations of Methane towards Orion IRc2 with SOFIA/EXES

Jose S. Monzon[1], Naseem Rangwala[2,3], Sarah Nickerson[2,3]

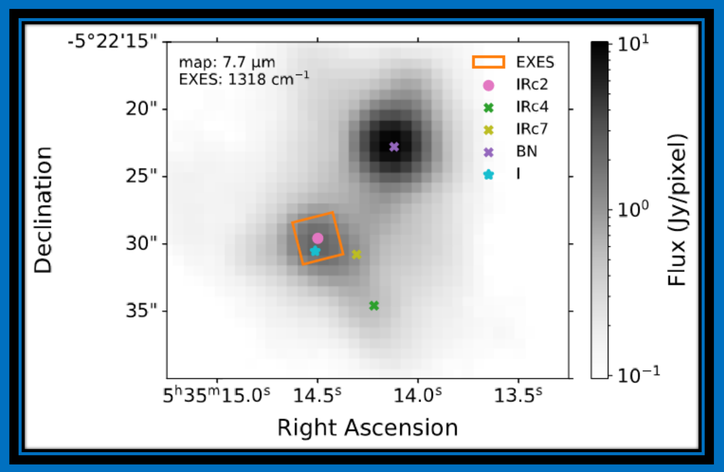
{1} University of California, Santa Cruz; 1156 High St., Santa Cruz, CA 95064, USA
{2} Space Science and Astrobiology Division, NASA Ames Research Center, Moffet Field, CA, 94035 USA
{3} Bay Area Environmental Research Institute, Moffet Field, CA, 94035, USA

National Aeronautics and Space Administration

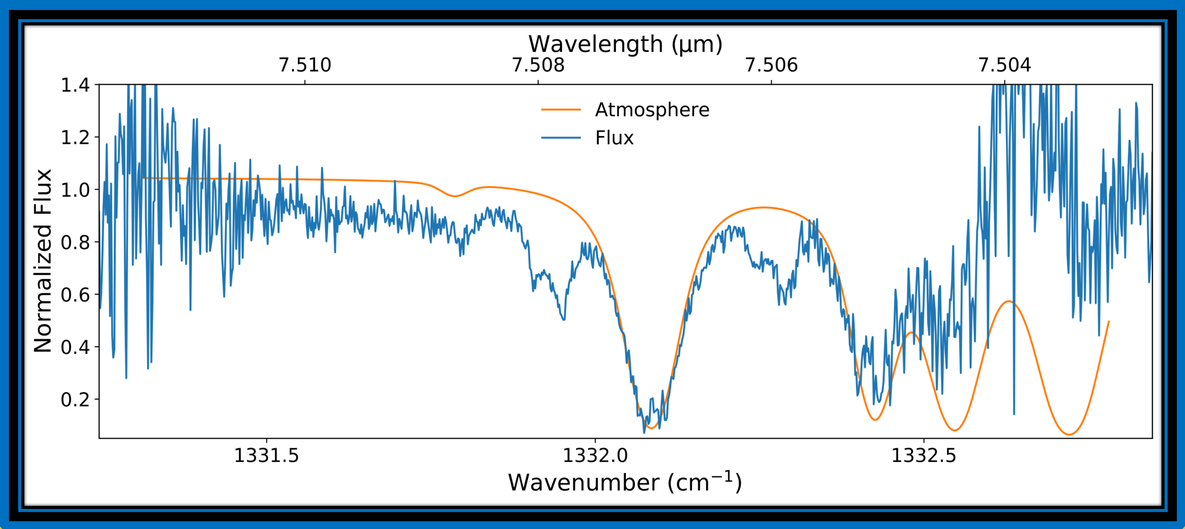
ABSTRACT

We present high-resolution (R~60,000) MIR detections of methane in the interstellar medium. Our observations were obtained with the Echelon-Cross-Echelle Spectrograph aboard the Stratospheric Observatory for Infrared Astronomy and are the first of CH4 towards the Orion IRc2 hot core. Exceptional resolution distinguishes 6 individual rovibrational transitions spanning 7.5 to 7.66μm We resolve the primary velocity component for CH4 as -7.7± 0.5 km/s with excitation temperature as 210 ± 89 K and column density as 2.07 ± 0.51×10^{17} cm^{-2}We also detect a secondary velocity component but our analysis is incomplete. These results give insight into the basic principles of astrochemical networks in the ISM and paired with observations of other chemical species, could provide exciting proto-planetary tracers.

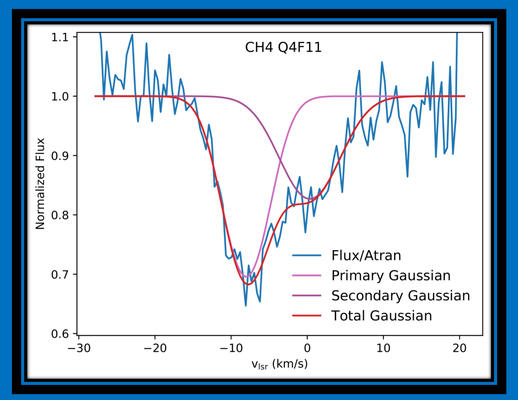
DATA SAMPLE

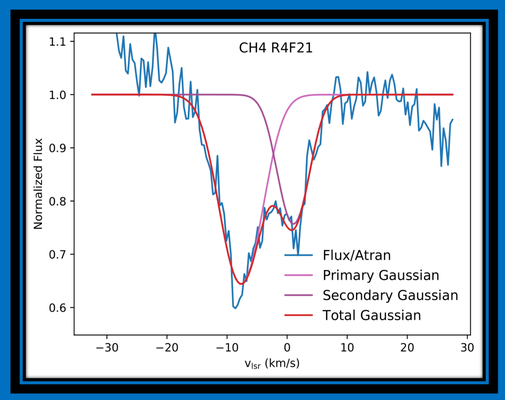


- **Observations of Orion IRc2 with the SOFIA observatory were made at altitudes from about 42000 to 44000ft. The EXES slit is shown with respect to photometric observations from FORCAST[1] .**
- **Spectra were acquired in the cross-dispersed high resolution mode with a slit width of 3.2" giving a resolving power of about 5 kms^{-1} .**
- **Several different wavelengths settings were observed for the larger scale inventory project, but we identified CH_4 in the setting which is centered at 7.58μm or 1318 cm^{-1} [2] .**



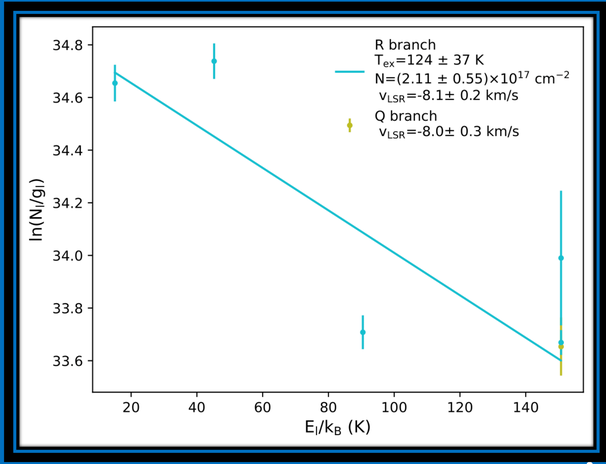
GAUSSIAN FITTING





- **As some atmosphere lines are mixed in with our observations, we divided out a smooth atmospheric model to EXES's resolution and normalized our flux[3] .**
- **We then ran a peak finding algorithm to select strong absorption lines and fit gaussians to the transitions. All orders required a double gaussian to account for the two velocity components.**
- **Using the HITRAN database and its Python interface, HAPI, we were able to determine the column density of each transition[3] .**

RESULTS



- **We assumed all molecules are in local thermal equilibrium (LTE) such that they follow the Boltzmann distribution [4] . Our linear fit details the excitation temperatures and column densities for CH_4.**

As we are the first to measure CH_4 towards IRc2, our results are not directly comparable to other studies. However, Boogert et al. 2004[5] studied CH_4 towards massive protostar NGC 7538 and measured similar temperatures (T_{ex} ~ 90K) but smaller column densities (N ~ 1.0 x 10^{16} cm^{-2}).

CH_4 is proposed to be the starting point of a rich chemistry leading to the complex organic molecules[5] and could even help us understand how biology develops on other worlds.

- **Our next steps include reassessing the atmospheric correction and finishing analysis on the secondary component.**

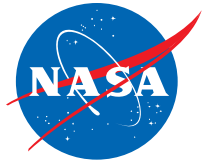
www.nasa.gov

Ames Research Center

REFERENCES

[1] - De Buizer, J. M., Morris, M. R., Becklin, E. E., et al. 2012, ApJ: Letters, 749, 35, doi: 10.1088/2041-8205/749/2/L23
[2] - Rangwala, N., Colgan, S. W. J., Le Gal, R., et al. 2018, ApJ, 856, 9, doi: 10.3847/1538-4357/aaab66
[3] - Nickerson et al, 2020, in prep
[4] - Goldsmith, P. F., & Langer, W. D. 1999, ApJ, 517, 209, doi: 10.1086/307195
[5] - Boogert, A.C.A., Blake, G.A., and Oberg, K. (2004) Astrophys. J. 615 344-353.

jsmonzon@ucsc.edu



Static Analysis Integration for NASA's core Flight System (cFS)

Kevin Oran

Pat Castle, Code TI

Summer 2020

Introduction – Kevin Oran

I am the oldest of three siblings. My family lives in Elkhorn, Nebraska. I studied mechanical and computer engineering at Iowa State. I am beginning a Masters Program in Electrical Engineering at Stanford this fall.

I love camping and hiking. I can't wait to spend some time outdoors and in the mountains when I get to California in August.

Background of Project

- The Starling mission will launch a cluster of cube satellites to demonstrate swarm technologies to enable future autonomous control of satellite networks.
- The Starling flight software is built on the core Flight System (cFS) framework which provides a publish and subscribe architecture for applications.
- Currently, the Starling team is only required to use the compiler flags –Wall and –pedantic for static analysis (in addition to their unit tests and system tests).

Research Questions

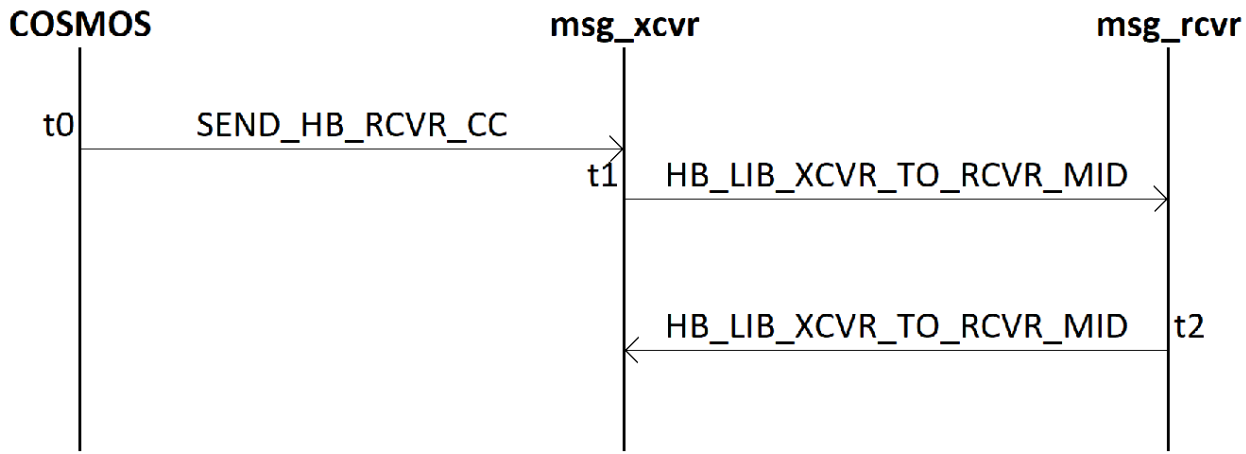
- How can the Starling team integrate a dedicated static analyzer into the build system?
- What are the feature and performance differences between the given static analyzers: gcc-10 static analyzer, clang static analyzer, and Inference Kernel for Open Static Analyzers (IKOS)?
- How can the Starling team use the static analyzer in their development process?
- Can a static analyzer detect errors that can arise from applications interacting with each other over the cFS software bus?

Methods

- Created two apps and a library for cFS in order to introduce bugs for the analyzers to find.

  The two apps were required to respond to commands and provide telemetry to the ground system, COSMOS.

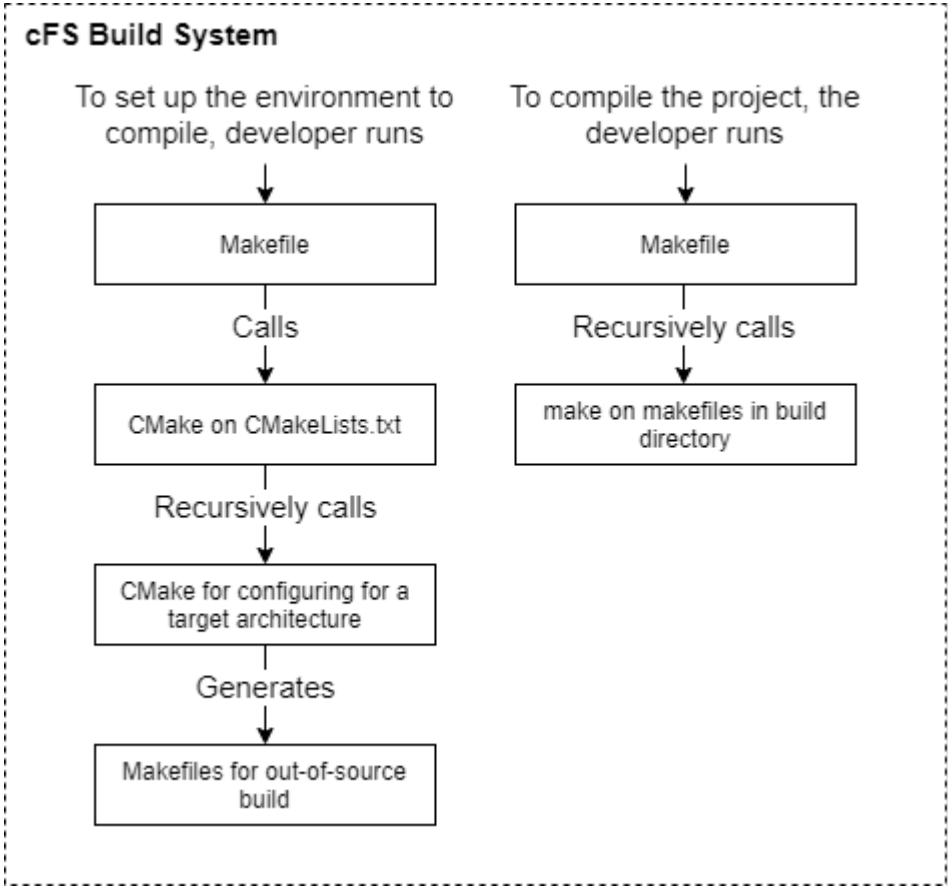
- Integrated the three analyzers into the build system to see their features and explore how they work



Sequence diagram of the message sequence carried out by my example code.

Challenges – Build System

- The build system for cFS is layered and complicated.
- Both IKOS and Clang provide tools to analyze a project by using its build system. They were difficult to incorporate correctly into this layered build system.
- cFS has outstanding issues with using the SIMULATION variable for target builds.



My diagram of how the cFS build system is structured.

Challenges – IKOS Support

- IKOS documentation fails to build in certain environments.
- IKOS does not provide tools to trace how the analyzer progresses through program control flow.
- Linking a model to abstract the core Flight Executive prevents IKOS from finding trivial bugs in my example applications. These bugs are found when analyzing the apps without linking any cFE implementation.

```
Attempting to analyze project
[*] Running ikos preprocessor
[*] Running ikos analyzer
[*] Translating LLVM bitcode to AR
[*] Running liveness analysis
[*] Running widening hint analysis
[*] Running interprocedural value analysis
[*] Analyzing entry point 'MXCVR_AppMain'
[*] Checking properties for entry point 'MXCVR_AppMain'

# Time stats:
ikos-analyzer: 0.107 sec
ikos-pp      : 0.017 sec

# Summary:
Total number of checks                : 117
Total number of unreachable checks    : 0
Total number of safe checks           : 108
Total number of definite unsafe checks: 1
Total number of warnings              : 8

The program is definitely UNSAFE
```

Screenshot of ikos results identifying valid warnings and errors

```
Attempting to analyze project
[*] Running ikos preprocessor
[*] Running ikos analyzer
[*] Translating LLVM bitcode to AR
[*] Running liveness analysis
[*] Running widening hint analysis
[*] Running interprocedural value analysis
[*] Analyzing entry point 'MXCVR_AppMain'
[*] Checking properties for entry point 'MXCVR_AppMain'

# Time stats:
ikos-analyzer: 0.196 sec
ikos-pp      : 0.054 sec

# Summary:
Total number of checks                : 113
Total number of unreachable checks    : 0
Total number of safe checks           : 113
Total number of definite unsafe checks: 0
Total number of warnings              : 0

The program is SAFE
```

Screenshot of ikos results evaluating the same code after including the cFE model

Findings/Further discussions

- Integration success

  IKOS can be used to analyze cFS apps or the cFE and can be configured within the cmake build system and one script.

  Clang Analyzer can be used to analyze the project and cFS applications.

  GCC 10 analyzer can be used on the project and applications

- Some errors are not found by all analyzers.

  My confluence documentation will include a discussion of differences in types of checkers implemented by each analyzer.

- The analyzers don't do a great job at finding bugs where they are introduced into the code.

  It's easy to spot where there could be potential for a null pointer dereference within a function, but because clang and gcc-10 work while compiling, they don't detect errors within the context of the complete program.

- I am going to do a brief writeup to provide feedback to the IKOS team with my observations from using IKOS this summer.

Discussion

- Analyzer behavior

  Clang and gcc-10 static analyzers run at build time so they only catch bugs in code that needs to be recompiled.

  IKOS will collect built files and link them before analyzing. This maintains the benefits of make because only code that is modified needs to be rebuilt before running a complete analysis on the project.

- Output differences

  IKOS and Clang provide browser-based graphical environments to help navigate the reports they generate. gcc-10 does not.

  Clang and gcc10 do a better job of showing the control flow when calling out a warning or error.

IKOS and Clang Report Examples

```
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
/*                                                                       */
/* HB_WriteMessage                                                       */
/*                                                                       */
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
void HB_WriteMessage(char *msg_buf, HB_Pkt_t *packet)
{

    packet->MsgLen = strlen(msg_buf);
    // Function ought to check the HB_MAX_MSG_LENGTH against this buffer, but doesn't
    //if(packet->MsgLen > HB_MAX_MSG_LENGTH){
    //    packet->MsgLen = HB_MAX_MSG_LENGTH;
    //}
    int i;
    for(i=0; i < packet->MsgLen; i++)
```

1 Assuming 'i' is >= field 'MsgLen' →

2 ← Loop condition is false. Execution continues on line 82 →

```
    {
        packet->Msg[i] = msg_buf[i];
    }
    int x = 7;
    int y = 0;
```

3 ← 'y' initialized to 0 →

```
    int z = x / y;
```

4 ← Division by zero

```
} /* End HB_WriteMessage */
```

Screenshot of the Clang Analyzer's scan-view tool correctly catching the division by zero error, but missing the buffer overflow (clang wouldn't show them at the same time if it caught both)

```
64 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
65 /*                                                                       */
66 /* HB_WriteMessage                                                       */
67 /*                                                                       */
68 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
69 void HB_WriteMessage(char *msg_buf, HB_Pkt_t *packet)
70 {
71
72     packet->MsgLen = strlen(msg_buf);
73     // Function ought to check the HB_MAX_MSG_LENGTH against this buffer, but doesn't
74     //if(packet->MsgLen > HB_MAX_MSG_LENGTH){
75     //    packet->MsgLen = HB_MAX_MSG_LENGTH;
76     //}
77     int i;
78     for(i=0; i < packet->MsgLen; i++)
79     {
80         packet->Msg[i] = msg_buf[i];
```

⚠ **80:26:** possible buffer overflow, accessing index between 0 and 2147483647 of constant "This message is too long for the buffer." of 41 elements
⚠ **80:24:** possible buffer overflow, pointer '&packet->Msg[(int64_t)i]' accesses 1 bytes at offset between 13 and 2147483660 bytes of global variable 'MXCVR_HeartbeatPkt' of size 37 bytes

```
81     }
82     int x = 7;
83     int y = 0;
84     int z = x / y;
```

⛔ **84:15:** division by zero

```
85 } /* End HB_WriteMessage */
```

Screenshot of IKOS's ikos-view tool correctly catching the buffer overflow error and the division by zero.

Future work

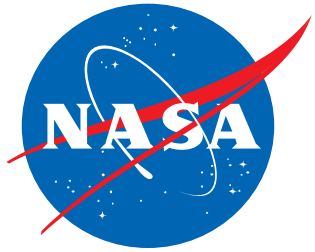
- The Starling team should take a look at the three analyzer options, identify checks that will catch more-critical bugs without introducing too many false positives, and incorporate regular static analysis as part of their development process.
- Some time effort should be spent updating the cFE model for ikos analysis and debugging why IKOS doesn’t catch errors when the model is used.
- I was not able to demonstrate or rule out discovering bugs that result from applications interacting on the software bus. With a working cFE model, adding functionality to support messaging in the model may enable the analyzers to look at app interactions.

Acknowledgments

I would like to extend a huge thank you to USRA, NASA, and ARC for making remote internships happen this summer. I'd also like to thank Pat for having me as an intern and helping me through the project. I have gained a lot of practical experience tinkering with make and cmake as well as using some publicly available static analysis tools.

References

- Starling Tech Overview https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20180007374.pdf
- IKOS public repository https://github.com/NASA-SW-VnV/ikos
- GCC 10 Documentation https://gcc.gnu.org/onlinedocs/gcc-10.2.0/gcc/Static-Analyzer-Options.html#Static-Analyzer-Options
- Clang Analyzer https://clang-analyzer.llvm.org/scan-build.html#scanbuild


NASA

National Aeronautics and Space Administration

Metal-Organic Frameworks (MOFs) for CO2 Capture in Cabin Environments

Katerina Quinn

Mentor: Darrell Jan, SCB

Summer 2020

Credit: NASA Image and Video Library

www.nasa.gov

About Me

- Undergrad: Chemical Engineering, University of South Florida
- USF Undergraduate Research: Algae-based biofuels, high-temperature catalysts for fuel production
- Johnson & Johnson: Light-management contact lenses
- NASA Kennedy Space Center: Plasma arc-gasification of waste & OSCAR Project
- Hobbies: Scuba diving, Disney, traveling

Introduction

- Carbon dioxide removal is crucial in air cabin environments
  - Crew members onboard a space craft exhale CO_2
  - The enclosed cabin becomes toxic if CO_2 is not removed
  - Air purification is required for oxygen regeneration [1]
  - Technology that selectively removes CO_2 is needed

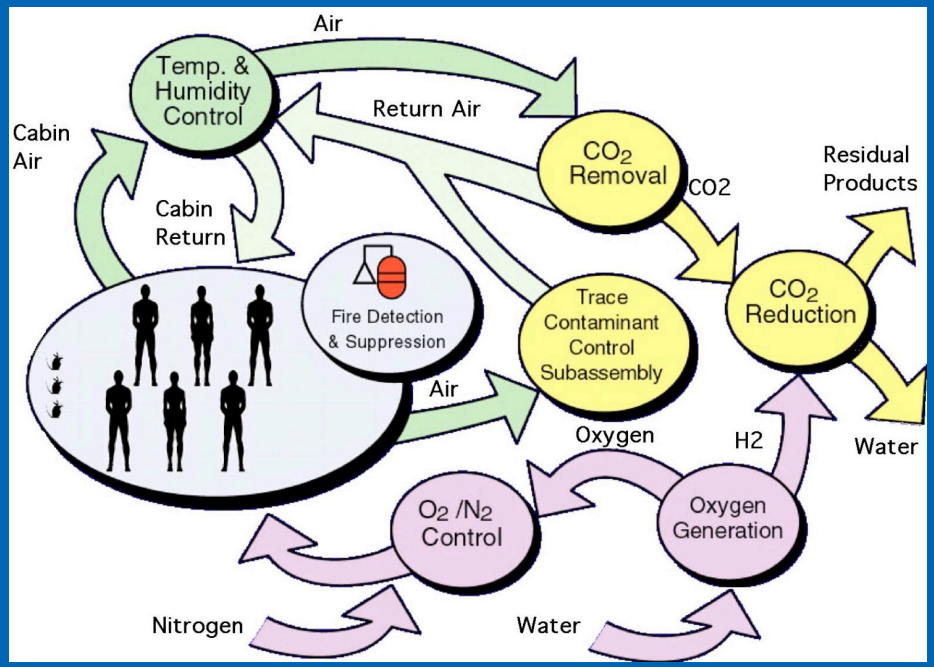
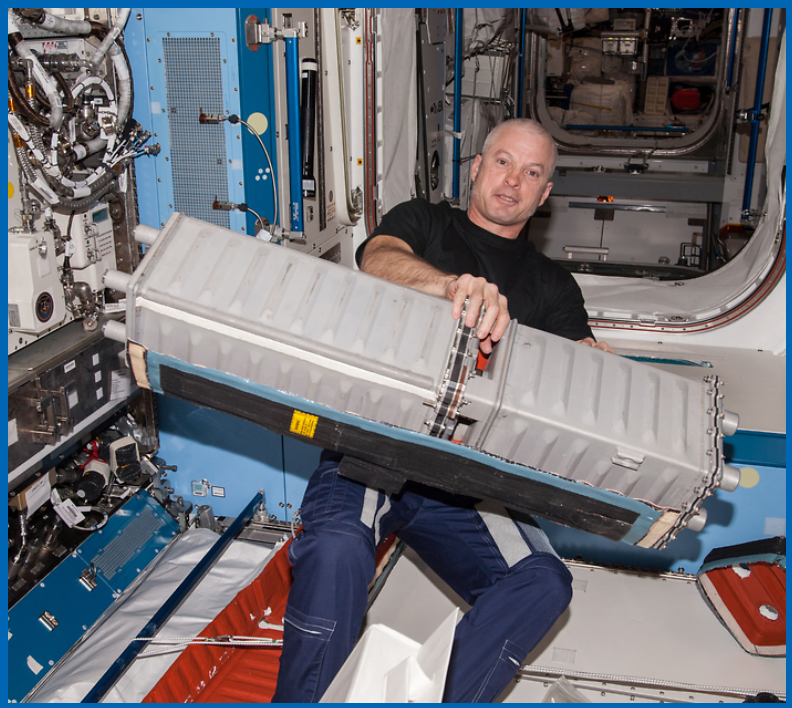


Figure 1. Closed Loop Air Revitalization System [1]

Background of Project

- Current system on the International Space Station (ISS) is the Carbon Dioxide Removal Assembly (CDRA)
- Under development is Carbon Dioxide Compression and Storage (CRCS) system
  - Utilizes Air-Cooled Temperature Swing Adsorption Compressor (AC-TSAC) [2].
  - Sorbent used for carbon dioxide removal requires significant energy
- MOFs are currently being investigated as a promising candidate sorbent for CO_2 removal



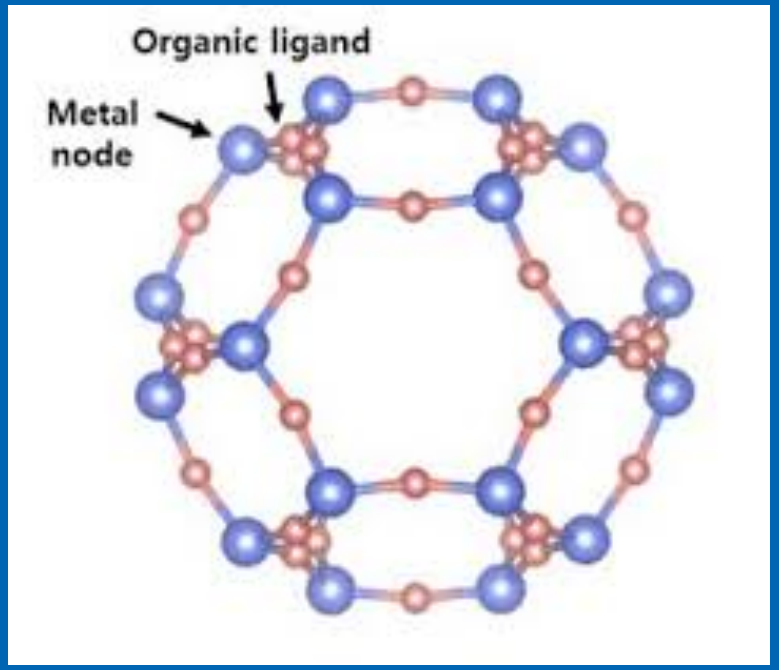
Credit: NASA Image and Video Library

Objectives

- Conduct a literature review on carbon dioxide removal with MOFs in order to identify a candidate for potential use
- Develop synthesis and testing procedures for various MOF candidates
- Find commercial MOFs for possible use
- Identify instrumentation needed for testing

What are MOFs?

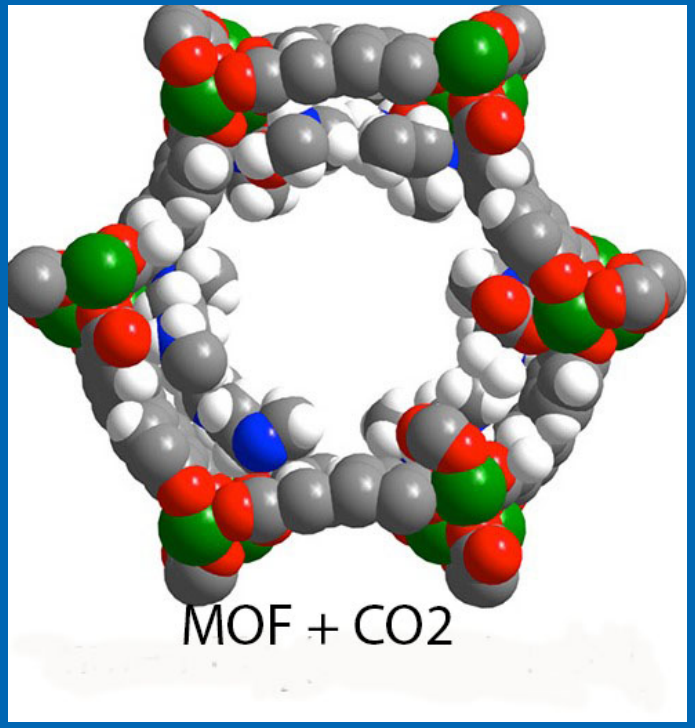
- A crystalline cage-like structure formation
  - Consisting of metal ions and organic linker molecules
- Large internal surface area
  - Resulting from the large internal cavity
- Highly porous and multi-chemical functionality
  - Allowing researchers to manipulate its structure
- Application for carbon dioxide (CO_2)
  - Capture and release



Credit: nanowerk.com

Benefits of MOFs

- CO2 capture and release requires minimal temperature change
- High selectivity for CO_2 over other compounds
  - Such as nitrogen, oxygen and water
- Large storage capacity for CO_2
  - Absorbed and desorbed
- Reduction in overall energy usage



Credit: newscenter.lbl.gov

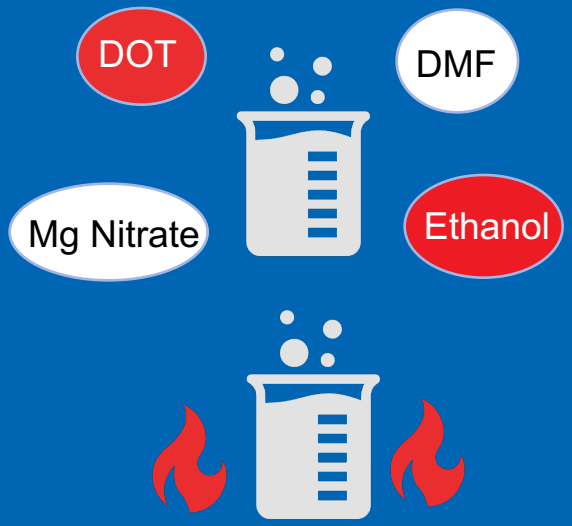
Applications of MOFs to Space Environments

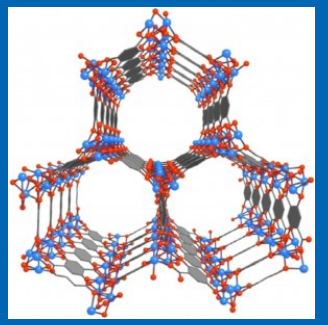
- Improve existing carbon dioxide removal systems on ISS and future long-duration spacecraft
  - More robust sorbent
  - Lower energy needs
  - Increased efficiency
- MOF could be integrated into CRCS system
  - Must remove CO_2 at required rate for four-person crew of 0.17 kg/hr [2]

Credit: NASA

Typical MOF Synthesis Method (Mg-MOF-74 [3])

1. A mixture consisting of 2,5-dihydroxyterephthalic acid (DOT), magnesium nitrate, N,N-dimethylformamide (DMF), ethanol and deionized water is sonicated
2. The solution is then placed in an autoclave and raised to a high temperature in order to allow for a complete reaction
3. The DMF is then decanted off and the MOF is washed with methanol
4. The precipitate is filtered and washed a second time with methanol before vacuum drying



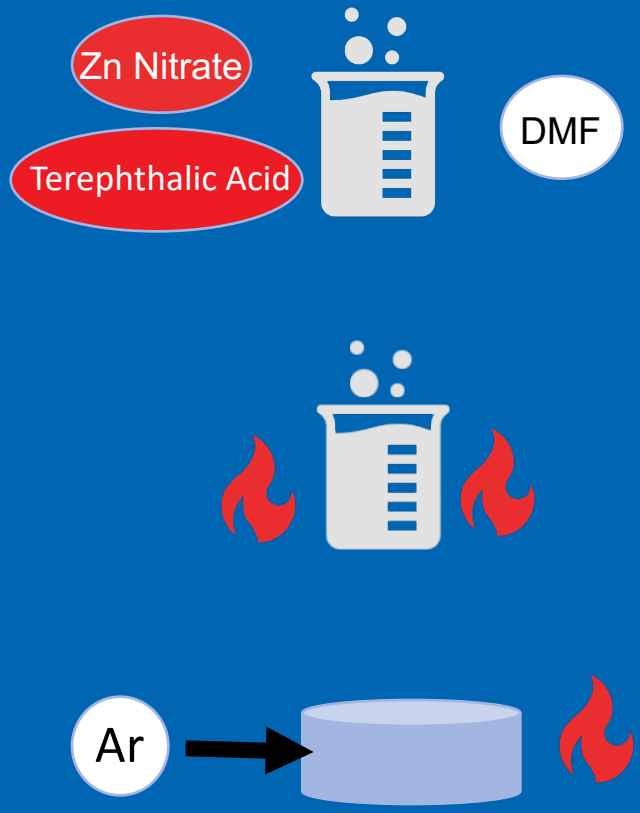


Credit: Berkeley Lab

Mg-MOF-74

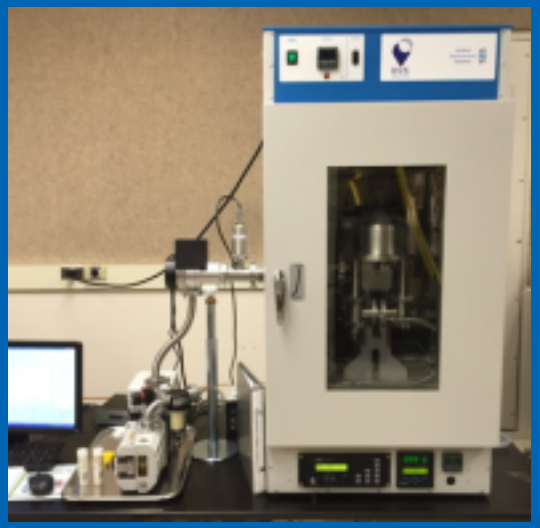
Alternative Synthesis Method (Carbonization of MOF-5) [4]

1. A mixture of zinc nitrate hexahydrate, terephthalic acid and N,N-dimethylformamide (DMF) is sonicated
2. The solution is then placed in an autoclave and raised to a high temperature
3. White powder is produced in the vacuum oven when solvent solution is heated
4. Sample is placed into a furnace and heated under argon gas for **carbonization**
5. The MOF is placed in hydrochloric acid and filtered
6. The filtrate is washed with distilled water and ethanol, then dried using dry air



Instruments to Measure Adsorption

ASAP2020



DVS

Testing also to be completed:

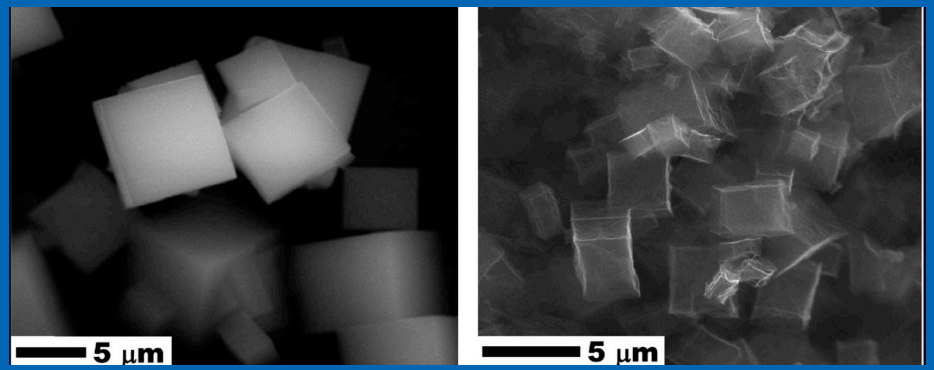
- Surface Area
- Pore Diameter
- Pore Volume
- Regeneration Temperature

MOFs Absorption Compared to Other Materials

Material	Absorption Capacity (mmol/g)	CO2 Uptake Temperature (K)	BET Surface Area (cm^2/g)
Zeolite 13X [2]	3.35	298	330
Mg-MOF-74 [2]	8.61	298	1174
Bio-MOF-1[3] [5]	4.62	273	800
Carbonized MOF-5 [3]	2.43	273	1884

Carbonized MOFs Data

Material	CO2 Absorption Capacity (mmol/g)	Specific Surface Area (cm^2/g)	Total Pore Volume (cm^3/g)
Pristine MOF-5 [3]	1.30	477	0.33
Carbonized MOF-5 [3]	2.43	1884	1.84



Before carbonization

After carbonization

MOF-5 [4]

Potential MOF Suppliers

NASA currently contracted Mosaic Materials, Inc. to develop adsorbents for life-support on Mars [5]

Sigma-Aldrich carries MOFs made by BASF that have gas adsorption abilities [6]

STREM Chemicals, Inc. carries MOFs with adsorption capabilities [7]

MOF Technologies specializes in tailorable MOFs [8]

Conclusion

- Metal-Organic Frameworks have promising potential as a sorbent candidate
  - High surface area
  - High porosity
  - Low Energy
- Synthesis and testing procedures researched and developed for testing
- Potential partners for commercial production of MOFs

Credit: NASA Image and Video Library

Future Research & Implications

- Testing of MOFs in simulated cabin environment conditions
- Side by side testing of pristine-vs-carbonized MOF
- Potential microgravity test on ISS or commercial crew environment
- Publication of MOFs research

Acknowledgments

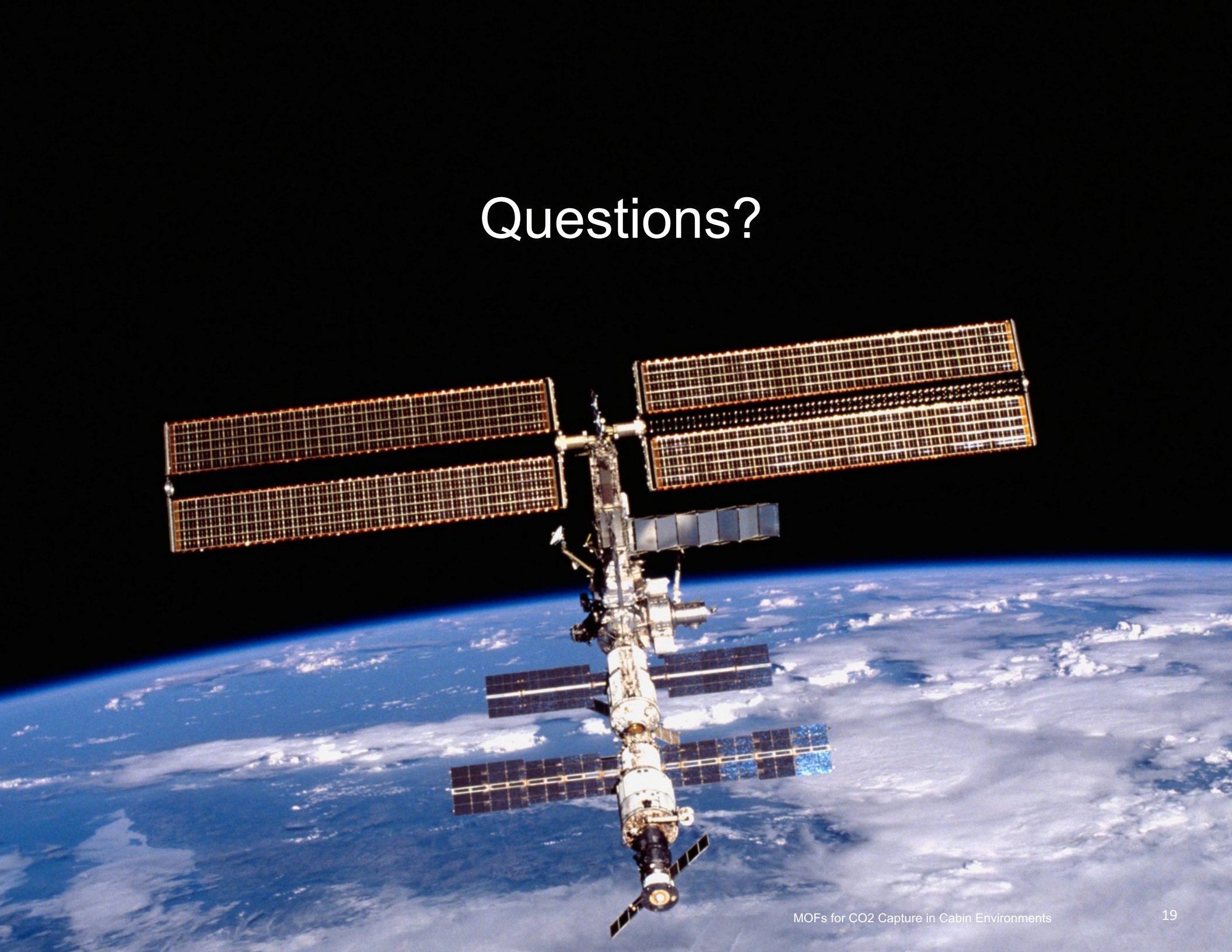
I would like to say thank you to my wonderful mentor Darrell Jan and the rest of the Air Revitalization team at Ames Research center!

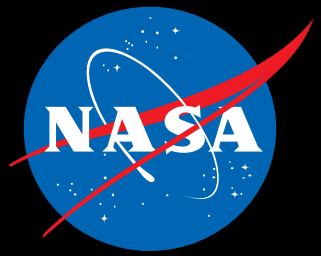
Such an amazing (and unique) summer and experience at NASA!

References

- [1] James C. Knox, Melissa Campbell, Lee A. Miller, Lila Mulloth, Mini Varghese, Bernadette Luna. “Integrated Test and Evaluation of a 4-Bed Molecular Sieve, Temperature Swing Adsorption Compressor, and Sabatier Engineering Development Unit,” 2006.
- [2] James C. Knox. “Development of Carbon Dioxide Removal Systems For NASA’s Deep Space Exploration Missions 2017-2018,” 2018.
- [3] Zongbi Bao, Liang Yu, Qilong Ren, Xiuyang Lu, Shuguang Deng. “Adsorption of CO2 and CH4 on a Magnesium-based Metal Organic Framework,” Journal of Colloid and Interface Science. Pgs.549-555. 2011.
- [4] Wojciech Kukulka, Krzysztof Cendrowski, Beata Michalkiewicz and Ewa Mijowska. “MOF-5 derived carbon as material for CO2 Absorption,” RSC Advances. 2019.
- [5]] Efficient CO2 and H20 Removal with Novel Adsorbents for Life Support Applications on Mars. SBIR. 2020.
- [6] Sigma-Aldrich. Metal Organic Frameworks (MOFs). 2020.
- [7] STREM Chemicals. MOFs. 2020.
- [8] MOF Technologies. Adsorbent Materials. 2020.

Questions?




NASA

Back up Slides

Prices of MOFs Commercial

Company	Material Name	Empirical Formula	Available Amounts	Pricing
Sigma-Aldrich [6]	Basolite® C 300	$C_{18}H_6Cu_3O_{12}$	10 g, 100 g	$388, $2310
Sigma-Aldrich [6]	Basolite® A100	$C_8H_5AlO_5$	10 g, 100 g, 500 g	$390, $1430, $5830
Sigma-Aldrich [6]	2,2′-Dinitro-4,4′-stilbenedicarboxylic acid	$C_{16}H_{10}N_2O_8$	500 mg	$240
STREM Chemicals [7]	aluminum MOF, MIL-101(Al)-NH2	$C_{24}H_{19}Al_3ClN_3O_{15}$	500 mg, 2 g	$69, $207
STREM Chemicals [7]	Copper benzene-1,3,5-tricarboxylate MOF	$C_{18}H_6Cu_3O_{12}$	500 mg, 2 g	$34, $102
STREM Chemicals [7]	Zinc 2-methylimidazole MOF	$C_8H_{10}N_4Zn$	1 g, 5 g	$40, $92

National Aeronautics and Space Administration

Wind Blade Design and Analysis for Mars TENG

John Schroebel | California Polytechnic State University | California Space Grant Consortium
Mentor: Myeonglok Seol | Universities Space Research Association | NASA Ames Research Center

BACKGROUND

Triboelectric nanogenerators (TENGs) utilize the repeated contact and separation of two materials to produce an electric current. TENGs offer greater power output for a given generator mass than traditional electromagnetic generators; they can also be 3D printed, allowing them to be produced and deployed as needed by the In-Space Manufacturing (ISM) protocol. This makes TENGs viable candidates for power generation on interplanetary missions.

Mars is particularly well-suited for TENGs. Although its atmospheric density is only ~2% of Earth's, its wind speeds average approximately 10 m/s and can exceed 60 m/s during storms, providing plenty of harvestable mechanical energy. A wind-powered TENG could supplement solar panels in producing electricity, and would be able to provide critical power in the event of a dust storm.

OBJECTIVES

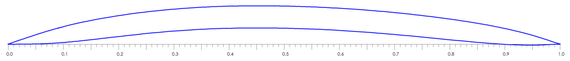
- Design several wind blades for the drum-type TENG, optimized for a range of wind speeds and RPM/torque combinations
- Characterize the blades' performance
- Produce CAD model of the blades so that they can be 3D printed and tested in Mars Surface Wind Tunnel

METHODS

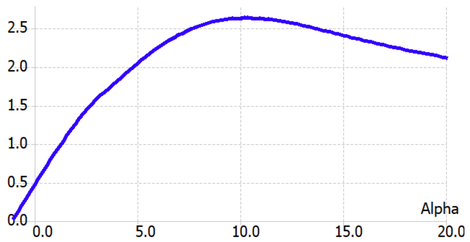
- XFOIL (open source) was used for airfoil analysis
- QBlade (open source) was used for blade design and analysis
- Solidworks (Dassault Systèmes) was used to create the CAD models

AIRFOIL DESIGN

- The Martian atmosphere's low density and the blades' relatively low speeds mean that the blades will be operating in the ultra-low Reynolds number regime (<1000), resulting in unusual aerodynamic properties
- Thin, cambered plates with sharp leading edges have been shown to provide best lift/drag ratio (C_L/C_D) in this regime
- 5% camber at 45% chord provided best performance
- 4% thickness was chosen to minimize thickness, while ensuring that structural integrity is maintained



Profile of airfoil used for the wind blades



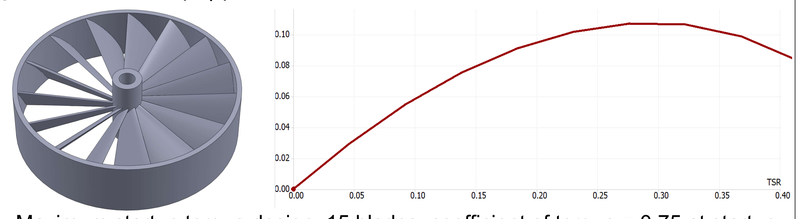
C_L/C_D vs Angle of Attack of airfoil

BLADE DESIGN

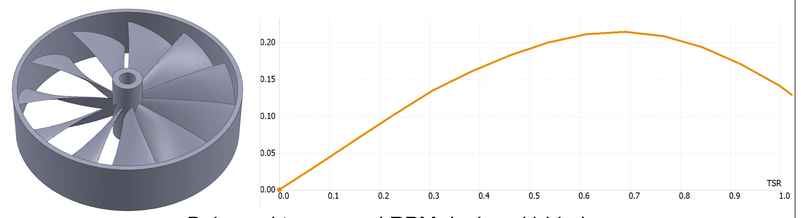
- Tip speed ratios (TSRs) from 0.2 to 1.5 were considered; higher TSRs generally produce more power, but TSRs above 1.5 would result in unreasonably high RPM
- Number of blades was chosen to allow turbulent air from one blade to be blown past the rotor before next blade takes its place
- Blade chord was optimized using QBlade algorithm, which seeks to slow the airflow to 1/3 of its original velocity for maximum efficiency (per Betz's Law)
- Blade twist was optimized using QBlade algorithm, which seeks maximum lift/drag ratio at the design TSR at each point along the blade

RESULTS

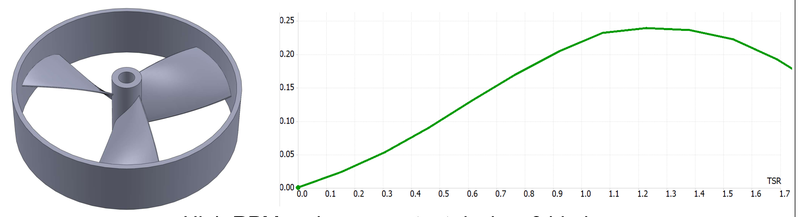
The preliminary CAD models and graphs of coefficients of performance (C_P) vs TSR are shown below:



Maximum startup torque design, 15 blades, coefficient of torque = 0.75 at startup



Balanced torque and RPM design, 11 blades



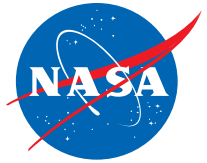
High RPM and power output design, 3 blades

CONCLUSION

- These blade designs exhibit moderate performance – modern wind turbines have C_P values approaching 0.5
- To increase performance, a larger wind blade would be needed to achieve higher TSRs at reasonable RPM
- A possible measure to increase performance is to add a duct to force more wind through the blades

ACKNOWLEDGEMENTS

I would like to thank Dr. Seol for his help and guidance throughout my internship, and Dr. Meyya Meyyappan for helping to make this opportunity possible.



A Python Tool for Urban Air Mobility Noise Analysis

Jocelyn Sun, Cornell University

Hok K. Ng, ARC-AFT

Summer 2020

Background

- The noise disruption caused by Urban Air Mobility (UAM) vehicles will be a major factor in public acceptance of UAM
- Urban Air Transport Disruption Management (UATDM) platform aims to determine the optimal flight trajectory for UAM aircraft, take into consideration weather and noise disruptions to surrounding urban communities

Project Summary

- Developed visualization and analytics for UAM noise disruption management software AIRNOISE-UAM
- Processed over 130 MB X2's study data and open source data in Dallas-Fort Worth area
- Created interactive visualization using Python's NumPy, GeoPandas, and Bokeh libraries
- Add features to analyze potential disruptions to local communities of X2 routes

Data Source

- X2 UAM simulation study data including 16 flight routes, 15 vertiports, and 112 waypoints
- Open source raw data (100 MB) including US Census statistics, noise sensitive facility, city borders, airspace classifications, highway, and water data in Dallas Fort-Worth area

Data cleaning

- Filter data within study area boundaries
- Extracted coordinates, project to Mercator
- Changed cleaning process depending on dataset (different geometry objects)

	name	route	geometry
0	U-UAM100	KDAQ,UUU00,AUX06,BUX00,UUU01,UUU02,KRAN	LINESTRING (-97.09259 33.16886, -97.09784 33.1...
1	U-UAM150	KRAN,VVV00,VVV01,BVX00,AVX05,VVV02,KDAQ	LINESTRING (-97.08727 32.75381, -97.08822 32.7...
2	U-UAM200	KADD,BUSHH,BUSHH,BSH24,BSH25,JAMS10,JAMX22,FFD...	LINESTRING (-96.83951 32.97064, -96.87030 32.9...
3	U-UAM250	KDT3,DAFYY,JAMII,RTE19,JAMS3,BTDFW,MDDWY,BIGGI...	LINESTRING (-96.80421 32.77254, -96.82875 32.7...
4	U-UAM300	KFR1,JAMS25,SHTHD,JAMS30,JAMX02,JAMX01,JAMS11,...	LINESTRING (-96.84217 33.15093, -96.84931 33.1...
5	N-UAM100	KDOA,AAA00,ZAX01,JAMX02,AVX05,AUX06,AAA06,AAA0...	LINESTRING (-96.66204 33.12863, -96.68188 33.1...
6	N-UAM150	KKEG,BBB00,BBB01,BUX00,BVX00,JAMX01,ZBX01,BBB0...	LINESTRING (-97.19431 32.90195, -97.19218 32.9...
7	N-UAM200	KCAT,CASBOY,XXX01,ZAX01,ZBX01,MYLTL,YYLIE,CASU...	LINESTRING (-96.82689 33.14486, -96.82428 33.1...
8	N-UAM217	KCAT,CASBOY,XXX01,ZAX01,ZBX01,MYLTL,YYLIE,FGX0...	LINESTRING (-96.82689 33.14486, -96.82428 33.1...
9	N-UAM251	KDF4,JAMS2,NEW21,JAMSE,DUCCC,MESTR,JAMS50,DDF2...	LINESTRING (-97.03858 32.90326, -97.03914 32.9...
10	N-UAM268	KDF4,JAMS2,NEW21,JAMSE,DUCCC,SPDDD,JAMS5E,JAMS...	LINESTRING (-97.03858 32.90326, -97.03914 32.9...
11	N-UAM285	KDF4,JAMS2,NEW21,JAMSE,DUCCC,SPDDD,JAMS5E,JAMS...	LINESTRING (-97.03858 32.90326, -97.03914 32.9...
12	N-UAM302	KDL6,JAMS55,JAMS54,DDF27,JAMII,NURDY,NEW60,ILI...	LINESTRING (-96.85398 32.84132, -96.84181 32.8...
13	N-UAM352	KDT4,RUBLL,CNT31,CNT30,CNT26,CNT21,CNT20,CASUI...	LINESTRING (-96.80792 32.77969, -96.79315 32.7...
14	N-UAM402	KMQT,EEE00,DWLB,LBDW,FHX04,FGX03,YYLIE,ZBX01,Z...	LINESTRING (-96.59123 32.78088, -96.59677 32.7...
15	N-UAM452	KNAS,JAMS53,JAMS52,INJOE,RTE12,RTE13,RTE19,JAM...	LINESTRING (-97.07816 32.75405, -97.07493 32.7...

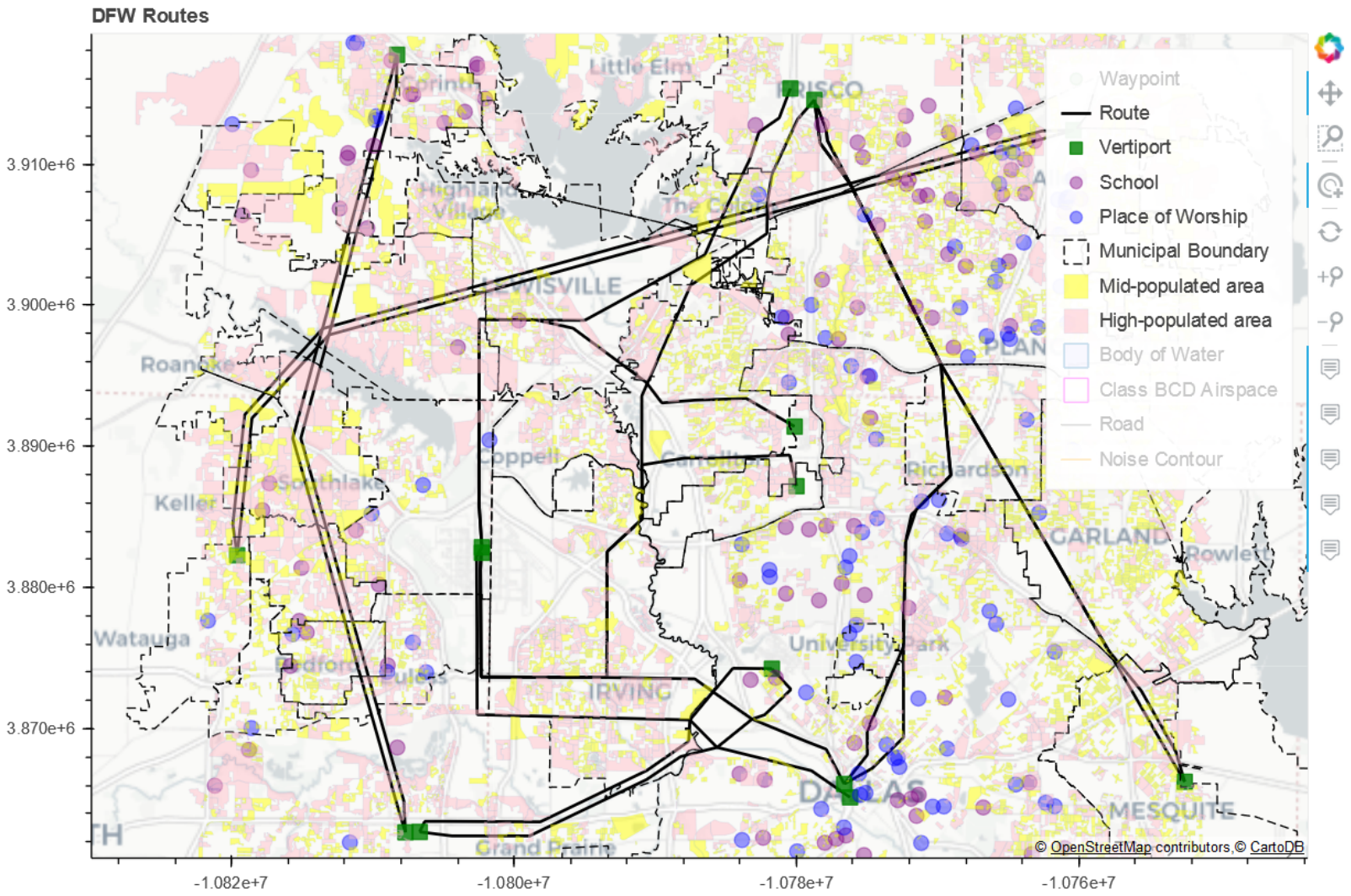
Sample X2 route data in dataframe format

Visualization

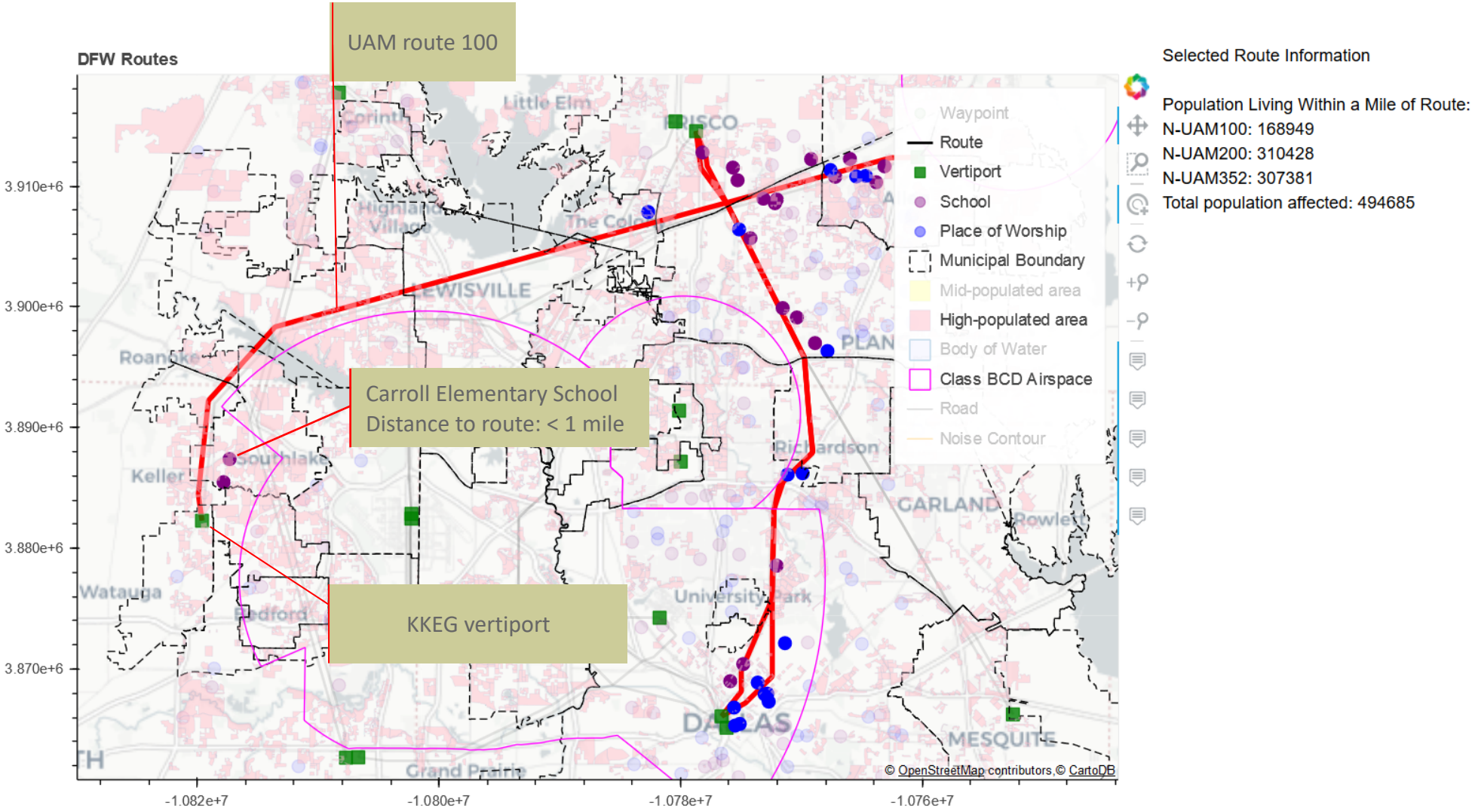
- Use Bokeh library to plot over 10,000 geometry shapes simultaneously for web visualization
- Interactive display
- Saved as HTML file to easily incorporate into other applications

Analytics

- Developed **real-time analytics** using Javascript
- Functions include
    - Highlighted facilities close to selected route
    - Count household and population number within a distance



X2 UAM study data with 16 routes, 15 vertiports, 112 waypoints, and 189 community features in DFW area



Use case: highlight near-by noise sensitive facilities (school and place of worships) when a route is selected

Code Optimization

Problem	Solution
Slow loading and data processing	Pre-loading data and saving in files, DataFrame vectorization
Large datasets lag visualization	Dynamic display, only necessary glyphs on map

Project Contributions

- Develop large data visualization and analytics using X2 UAM simulation data and open source geodata
- Data processing and analysis scripts directly used with noise analysis software
  - Output files generated by code used as inputs for AIRNOISE-UAM
  - Can also use script that reports noise impacts
- Analysis used to help determine optimal flight routes

Route Name	Households Near Route	Population Near Route	Schools Near Route	Places of Worship Near Route	Population near arrival vertiport	Population near departure vertiport	Households near arrival vertiport	Households near departure vertiport
N-UAM150	62590	166868	12	5	4962	16099	2025	5631
N-UAM251	43626	106278	0	3	27322	77	13780	23

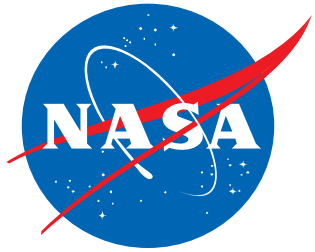
Future Use and Recommendations

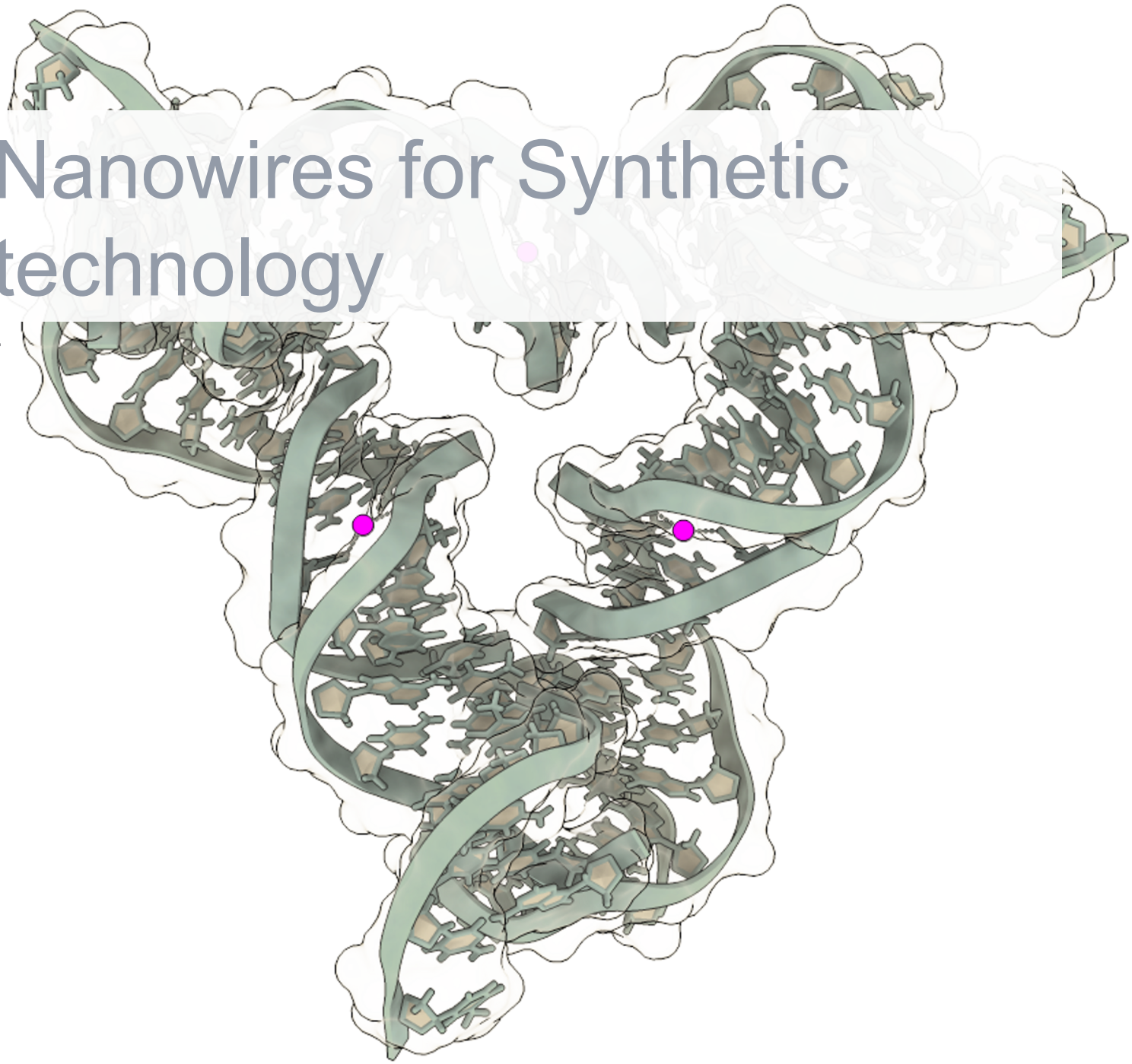
- Additional features to be implemented
- Code optimization
- Translation of tool for other uses

Acknowledgments and References

- Hok K. Ng, Jinhua Li, Yun Zheng, and Haley Feck

- Li, J., Chen, N., Ng, H. K., and Sridhar, B., “Simple Tool for Aircraft Noise-Reduction Route Design,” *15th AIAA Aviation Technology, Integration, and Operations Conference*, Dallas, TX, 2015.
- Li, J., Sridhar, B., Xue, M., and Ng, H., “AIRNOISE: a Tool for Preliminary Noise-Abatement Terminal Approach Route Design,” *16th AIAA Aviation Technology, Integration, and Operations Conference*, Washington, D.C., 2016.


NASA



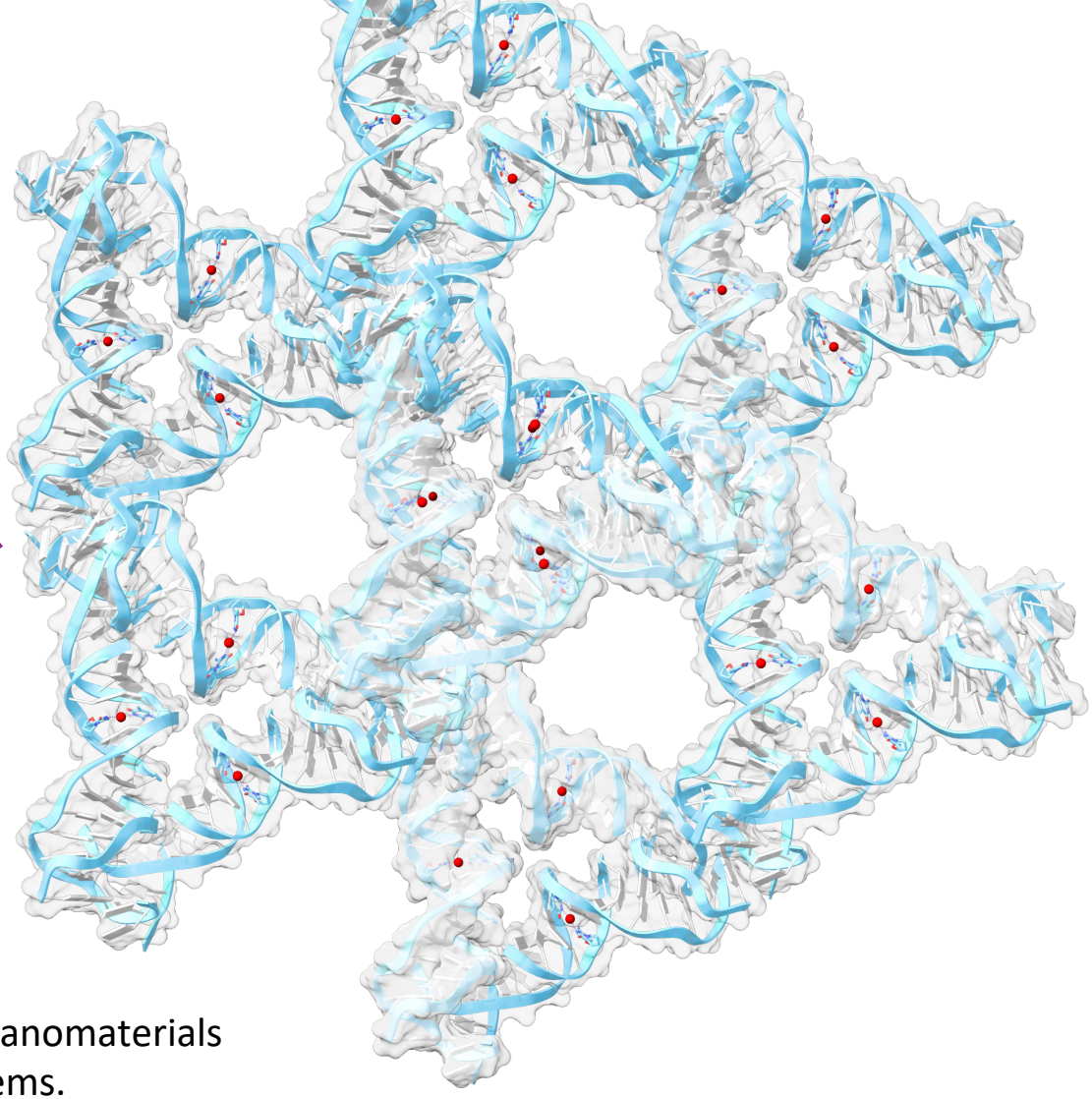
Biological Nanowires for Synthetic Space Biotechnology

Dr. Lynn Rothschild, SST

Summer 2020

Prior work funded through NSTRF ’14 Grant

Introduction



DNA crystals to develop physiochemical library of mmDNA base pairs

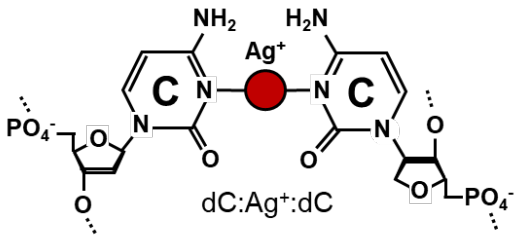
Chemical library for *electronic* DNA crystals for space technology

Research Goal: Assembly of **active** nanomaterials using biological systems.

Past and Present Work: Integrating metal-mediated DNA (**mmDNA**) base pairs into structural DNA nanotechnology.

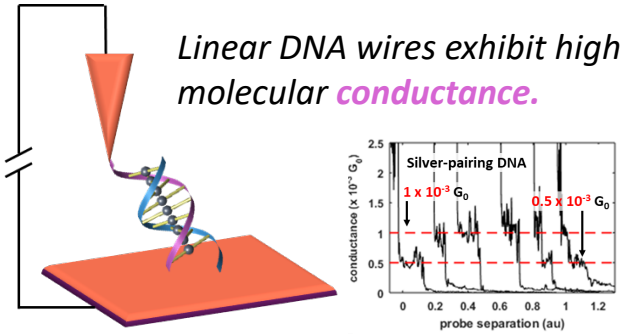
Background of Project / Literature Review

*Silver(I) ions can be reliably assembled into a **single-atom chain** at the core of the double helix.*



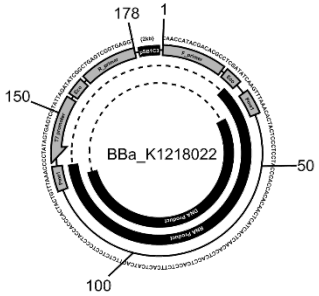
Vecchioni, Rothschild, Wind, et al, J Self-Assembly Mol Electronics 2019, 6.1, 61-90.

*Linear DNA wires exhibit high molecular **conductance.***



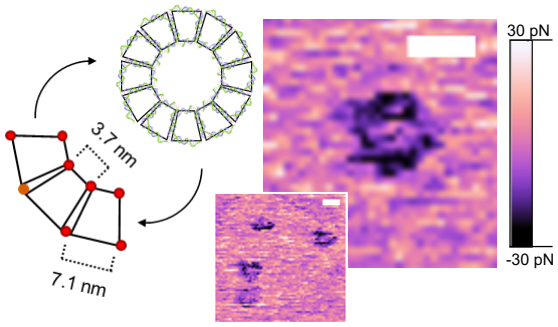
Toomey, **Vecchioni**, Rothschild, Wind, et al, J Phys Chem C 2016, 120, 7804.

*DNA nanowires can be encoded into bacterial plasmids for ISRU-based **biosynthesis**.*



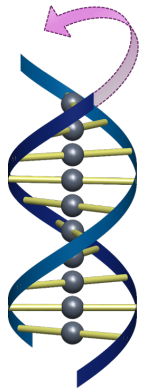
Vecchioni, Wind, Rothschild, et al, Sci Rep 2019, 9:6942.

*Nanowires can be self-assembled into a variety of **2D arrays**.*



Vecchioni, Rothschild, Wind, *Manuscript in preparation*

*Higher-order arrays infeasible: strain observed on canonical B-form nanostructures, but precise rotational dynamics **unknown**.*



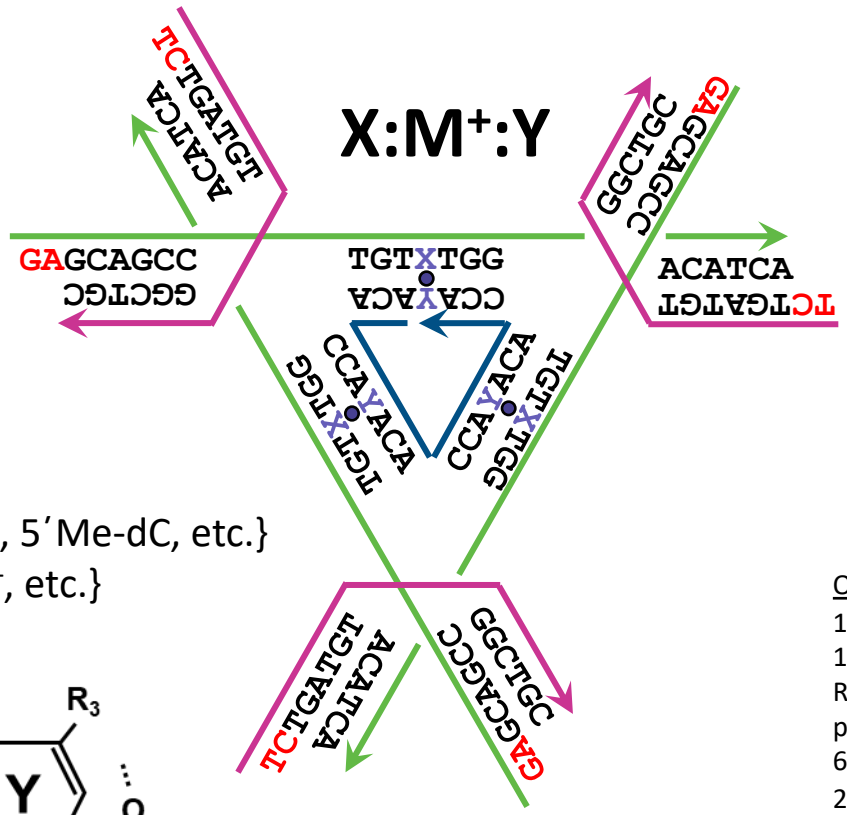
2019-2020: Assemble 3D biowire nanostructures and solve crystal structure. Analyze, optimize, add more ions, repeat.

Research Questions

- Can structural DNA nanotechnology act as a template for metal-mediated DNA base pairs?
- Will metal-mediated base pairing allow DNA to act as an electrically-active space nanotechnology?
- What are the exact structural implications of metal coordination inside the double helix, and how can this knowledge be used to build more integrated nanodevices?

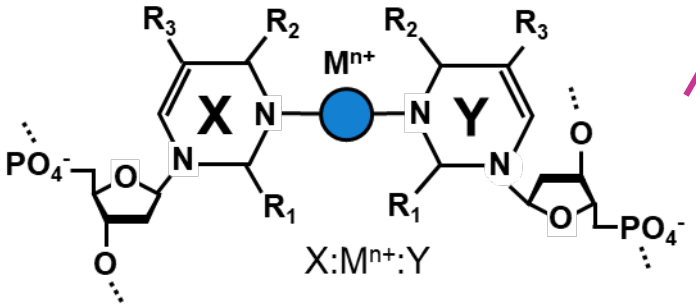
Methods of Crystallization

2-Turn Triangle: 7 bp (**7XY1**)



X,Y ⊆ {dC, dT, dU, iso-dC, 5′Me-dC, etc.}

$M^+ \subseteq \{Ag^+, Hg^{2+}, Pd^{2+}, Li^+, etc.\}$

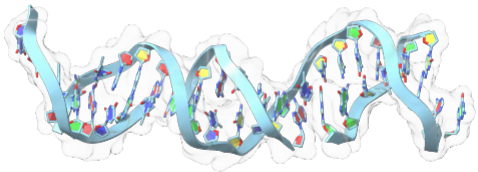


The DNA tensegrity triangle self-assembles into a macromolecular crystal using DNA hybridization. It is a **3D DNA nanostructure** that is amenable to XRD analysis.

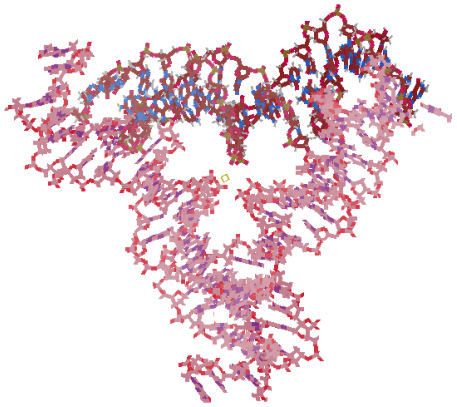
Optimized Buffer Conditions *(occasional deviation)*
10 mM MOPS
125 mM magnesium sulfate
Reservoir 10x salts
pH 7.7 (adjusted with NaOH)
60 pmol motif in 10 µL drop
2:1 M^+:XY stoichiometry (360 pmol ion)

Hanging Drop Annealing
275 hours
65 °C → 20 °C at 0.4 °C/hr
20 °C → 4 °C at 0.1 °C/hr

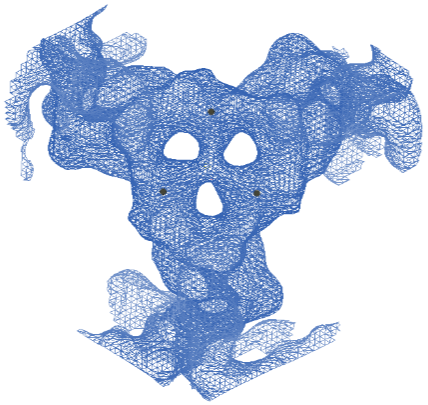
Methods of Data Analysis



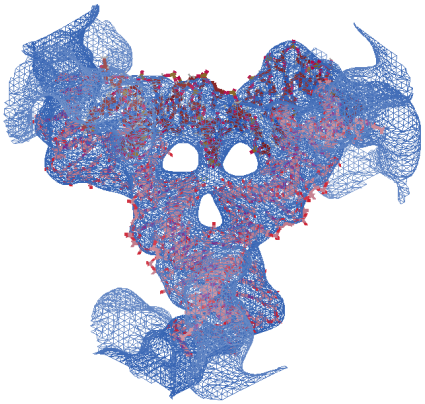
1) 3GBI edited in Coot and ChimeraX to build correct sequence



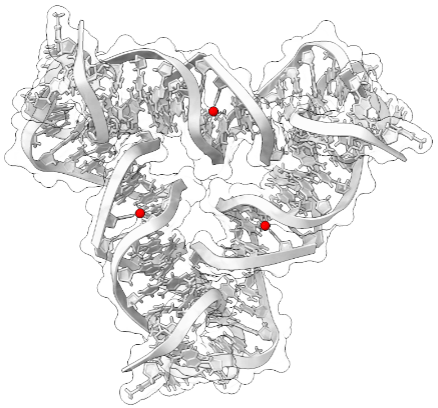
2) MR performed on anomalous diffraction data



3) SAD analysis performed on MR result to identify ion sites (too few HA sites to solve full structure with SAD)

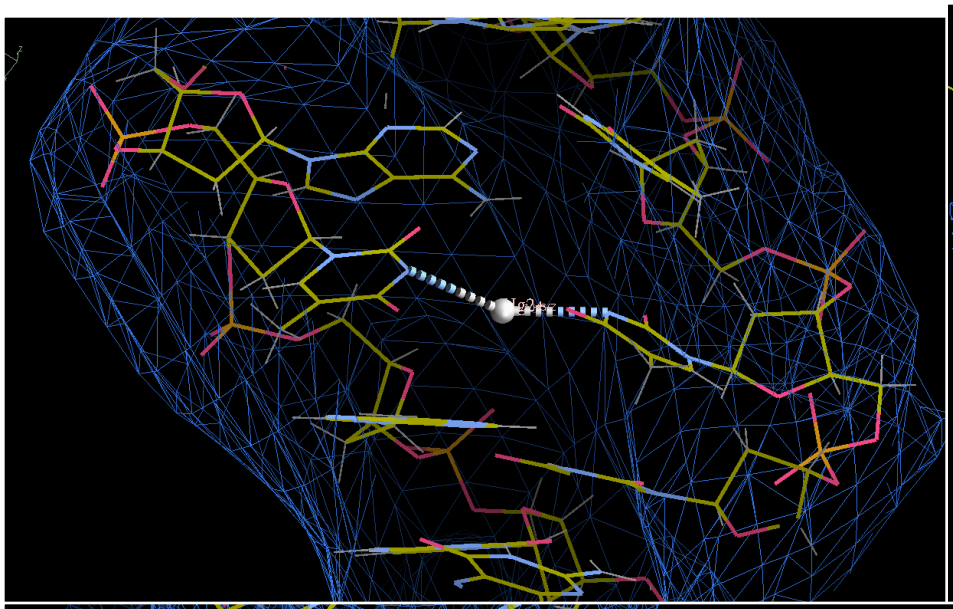


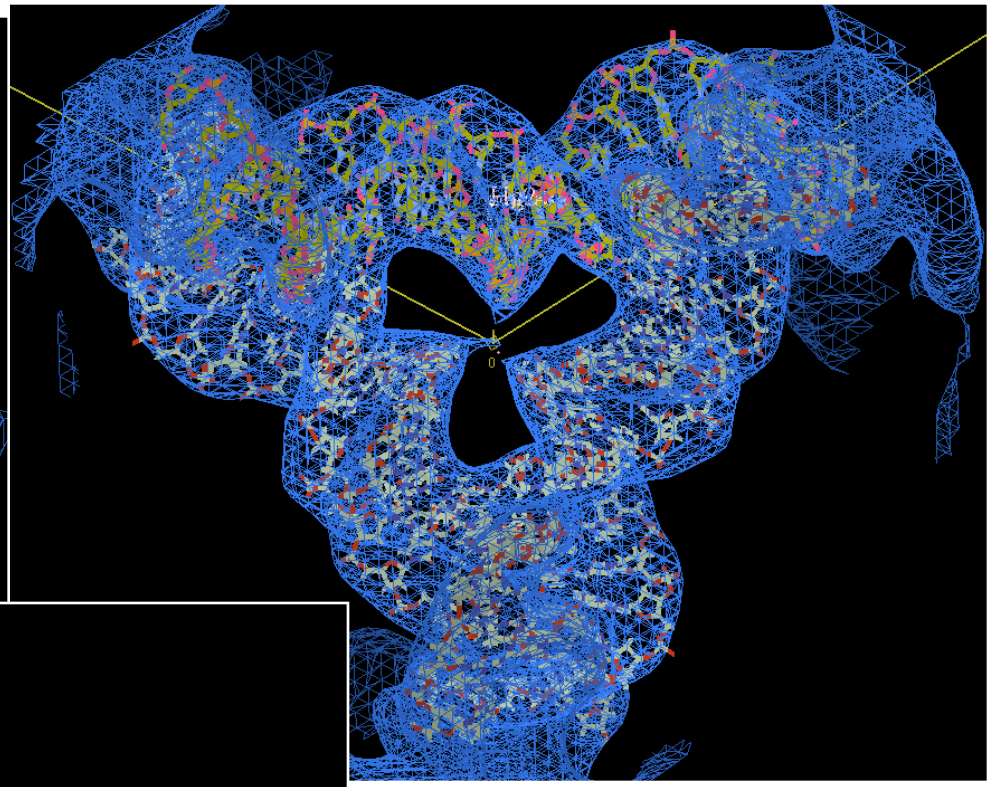
4) MR result merged with heavy atom sites; restraints generated for possible metal linkage, restraints for non-standard residues built in Phenix

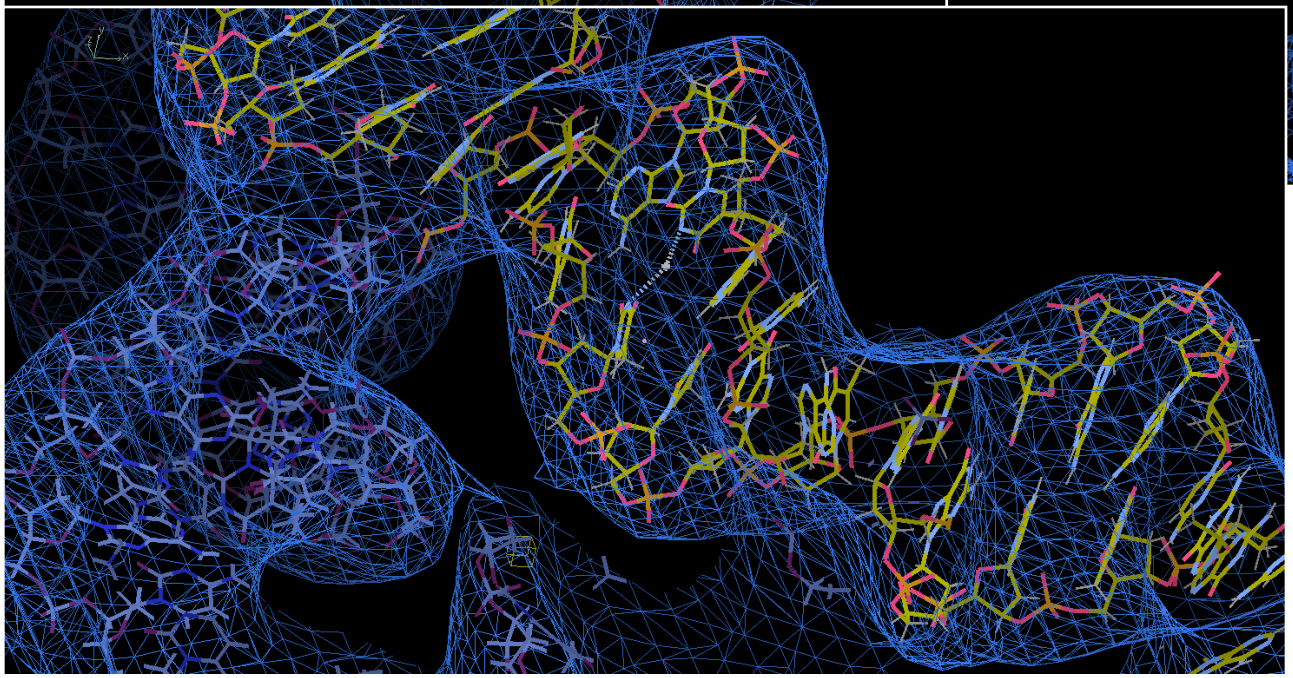


5) Iterative refinement and adjustment in Phenix and Coot

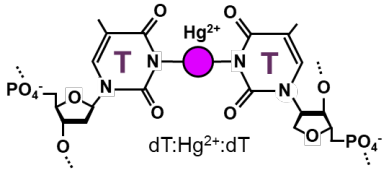
Solution of T:Hg^{2+}:T base pair structure





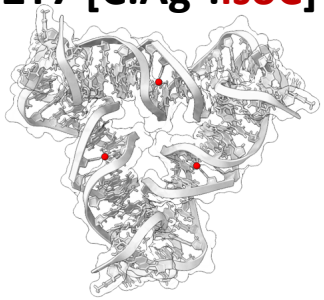


[7TT1_Hg^{2+}]
2Fo-Fc at 1.0 σ



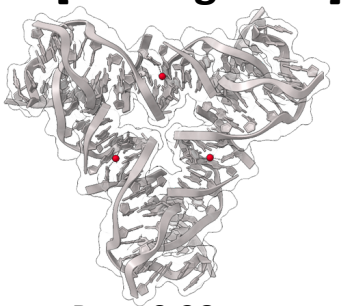
Crystal Structures of mmDNA Pairs

2T7 [C:Ag^+:isoC]



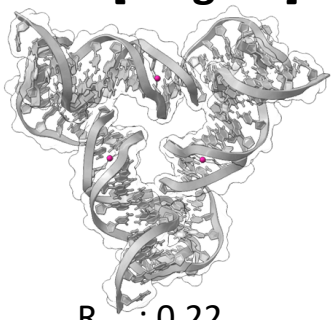
R_{free}: 0.25
Resolution: 5.50 Å
R3 Cell: 68.8 Å
**needs finalizing*

2T7 [isoC:Ag^+:isoC]



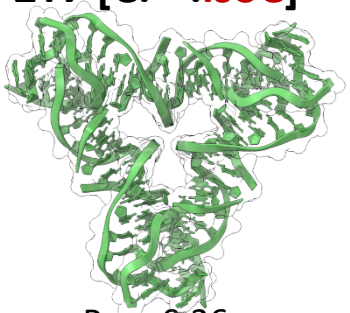
R_{free}: 0.28
Resolution: 6.10 Å
R3 Cell: 67.9 Å

2T7 [T:Hg^{2+}:T]



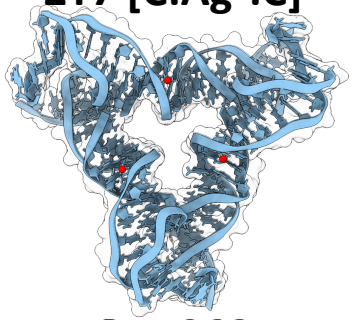
R_{free}: 0.22
Resolution: 5.50 Å
R3 Cell: 69.0 Å

2T7 [C:◌:isoC]



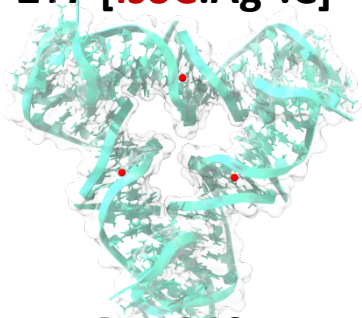
R_{free}: 0.26
Resolution: 4.30 Å
R3 Cell: 69.1 Å
**needs finalizing*

2T7 [C:Ag^+:C]



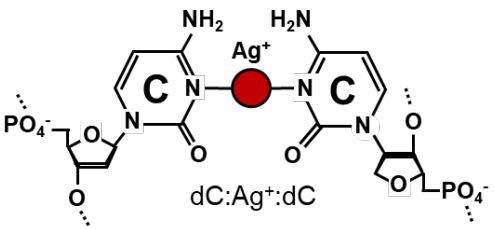
R_{free}: 0.26
Resolution: 5.00 Å
P1 $\overline{\text{Cell}}$: 68.0 Å
**non-anomalous; needs another collection*

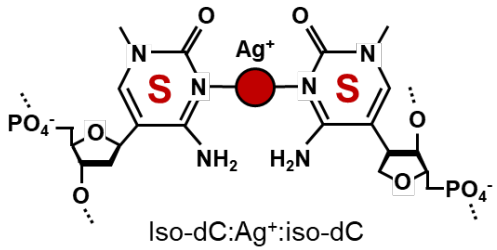
2T7 [isoC:Ag^+:C]

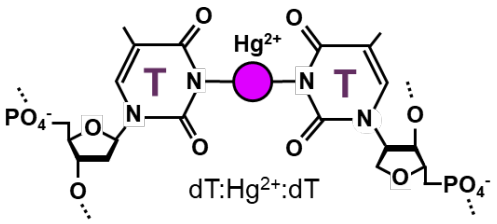


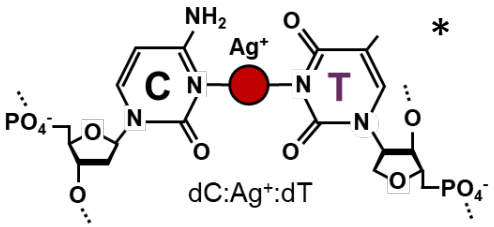
R_{free}: 0.26
Resolution: 8.00 Å
R3 Cell: 68.5 Å
**low quality data; needs another collection*

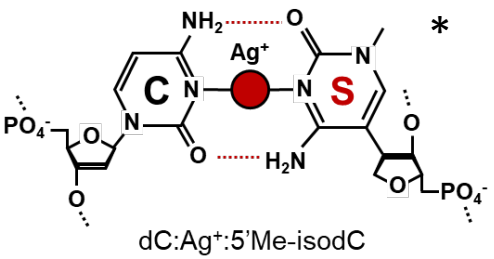
Current State of Research

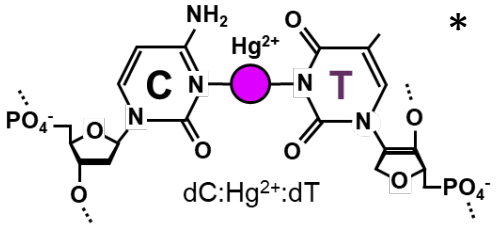






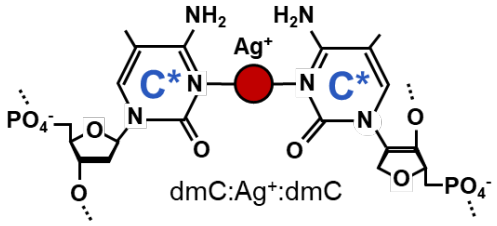


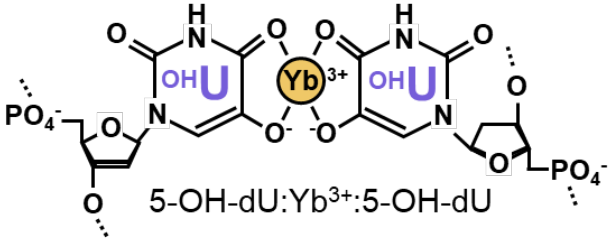


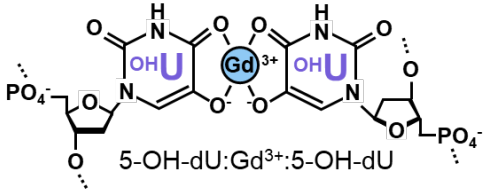


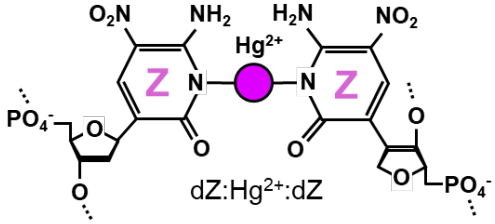
*isomers also crystallized

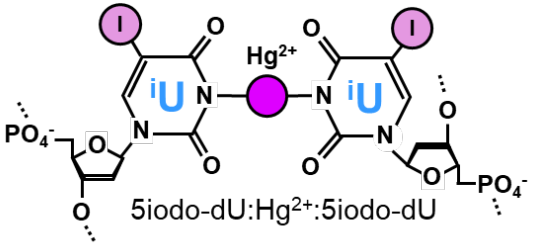
Next steps for metal pair development:

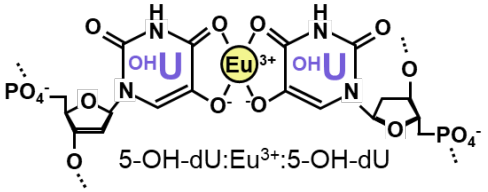






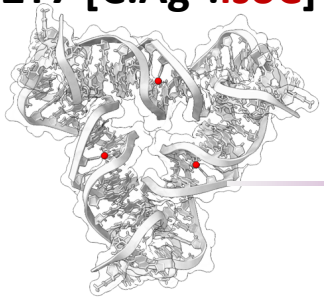






Future Work

2T7 [C:Ag⁺:isoC]

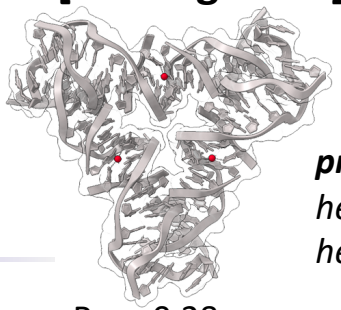


R_{free}: 0.25
Resolution: 5.50 Å
R3 Cell: 68.8 Å

Finalize current solutions, especially isoC geometry

Phenix doesn't support IMC base pairs: generate manual restraints

2T7 [isoC:Ag⁺:isoC]

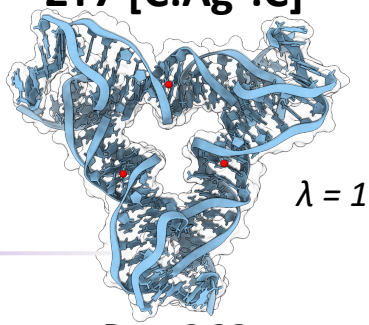


propeller twist
helical twist
helical diameter

R_{free}: 0.28
Resolution: 6.10 Å
R3 Cell: 67.9 Å

Carry out **geometry analysis** through WebX3D

2T7 [C:Ag⁺:C]



λ = 1.67 Å at APS

R_{free}: 0.28
Resolution: 5.00 Å
P1 $\overline{\text{Cell}}$: 68.0 Å

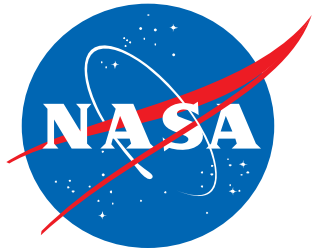
Reshoot low quality crystals at anomalous wavelengths

Lessons Learned

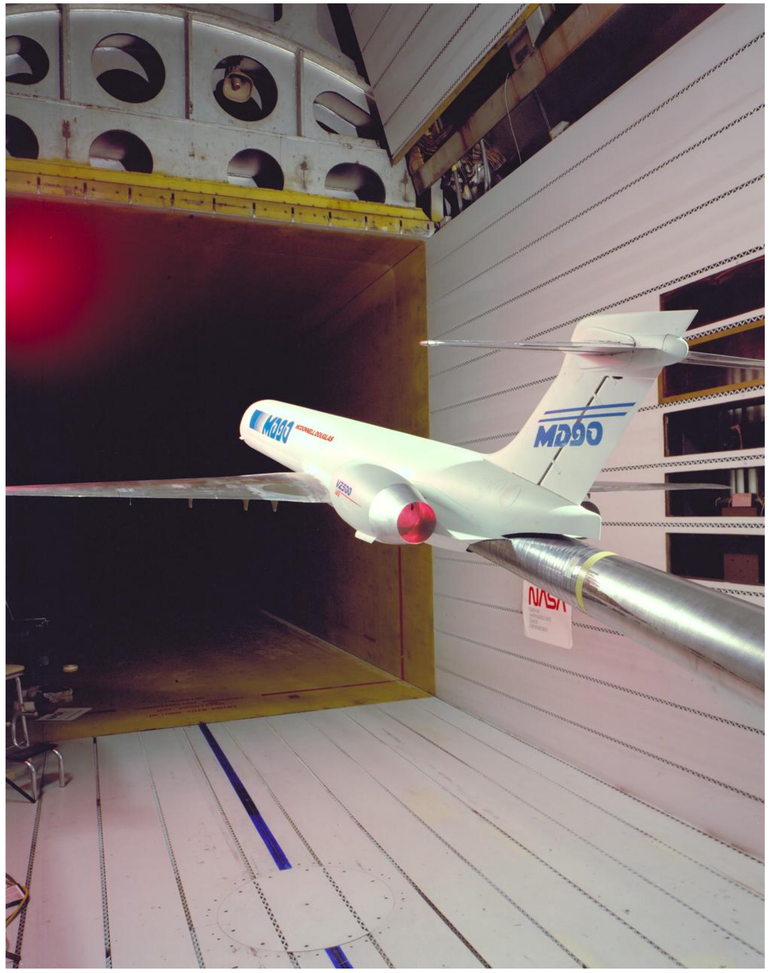
- DNA can self-assemble into macroscale objects while incorporating metal-mediated base pairs
- DNA crystals allow for the solution of structures in orthogonal base chemistry
- There is much more work to be done in this space to build a full library of mmDNA

Acknowledgments

- Many thanks to Dr. Lynn Rothschild for believing in the BioWires project from the start!
- Thanks to the members of the 2013 Stanford-Brown iGEM team for getting this started.
- Thanks to Professor Nadrian Seeman and Ruojie Sha at NYU for hosting experiments
- And many thanks to the Ames CIF 2020 grant, which partially funded this work


NASA

Statistical Process Control in the 11-by 11-Foot Transonic Wind Tunnel



Jason Chapman
California Polytechnic State University – San Luis Obispo

Unitary Plan Wind Tunnel

- The 11-by 11-foot Transonic Wind Tunnel (11-foot TWT) is part of the Unitary Plan Wind Tunnel (UPWT) complex at NASA's Ames Research Center at Moffett Field, California.
- The 11-foot TWT is a closed-return, variable-density tunnel with a fixed-geometry, ventilated test section.

Aerial view of Ames UPWT complex

UHB Semi-Span Model in 11-ft TWT

Introduction

- Responsible for two MATLAB scripts concerned with statistical process control in the 11-foot.

- Uncertainty in the computed tunnel conditions (i.e. Mach Number, Reynolds Number, etc.) required error propagation using Monte Carlo Analysis.
  - Monte Carlo analysis used to analyze random uncertainty correlated with repeatability.
  - Analysis intended to show overall uncertainty in tunnel conditions with the current tunnel instrumentation used to define the facility tunnel conditions.
- Standard Process Control for Short Static Pipe calibration.
  - Control Charts program established for Short Static Pipe testing.
  - Analysis intended to track stability of the centerline static pressure calibration test.

Methods

- Monte Carlo analysis method used to track uncertainty associated with repeatability (precision error) for Long Static Pipe.
  - Data is measured from sensors.
  - Random uncertainty associated with repeat runs is multiplied into an array of Gaussian numbers.
  - Uncertainty values are added to measured data points.
  - Tunnel condition equations are used to calculate model reference values.
  - Standard deviation of result is measurement uncertainty due to repeatability.
- Generated a single MATLAB script to track bias and random uncertainty.
  - Same approach, with bias uncertainty added to measured data points as well.

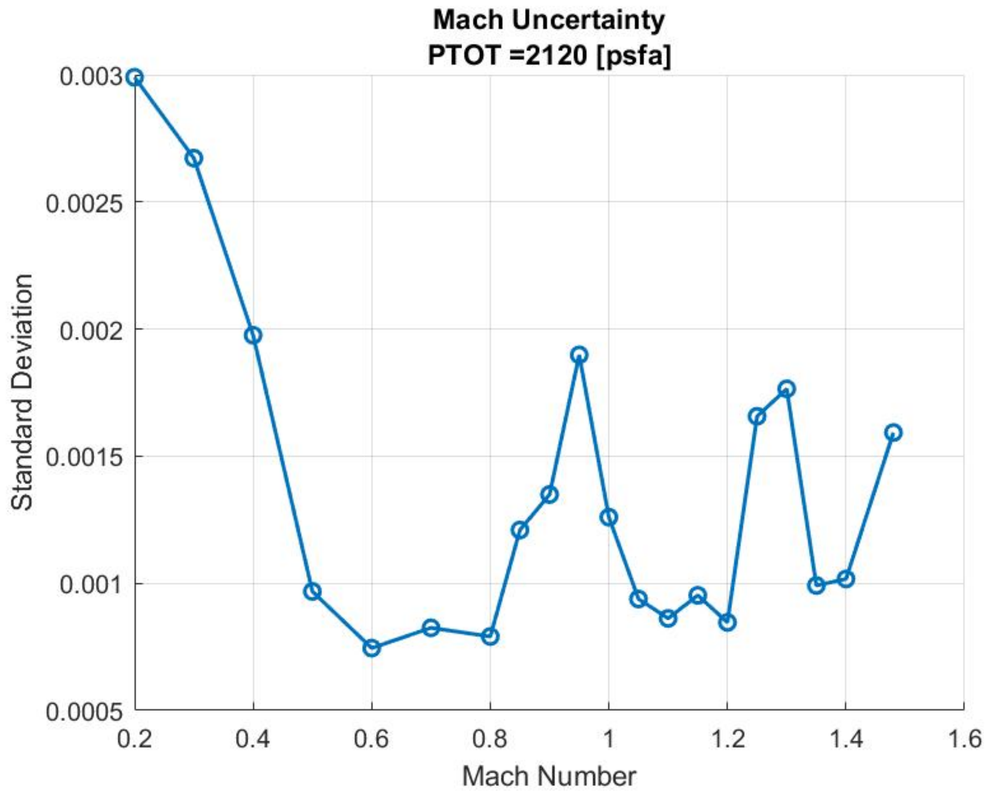
Methods Continued

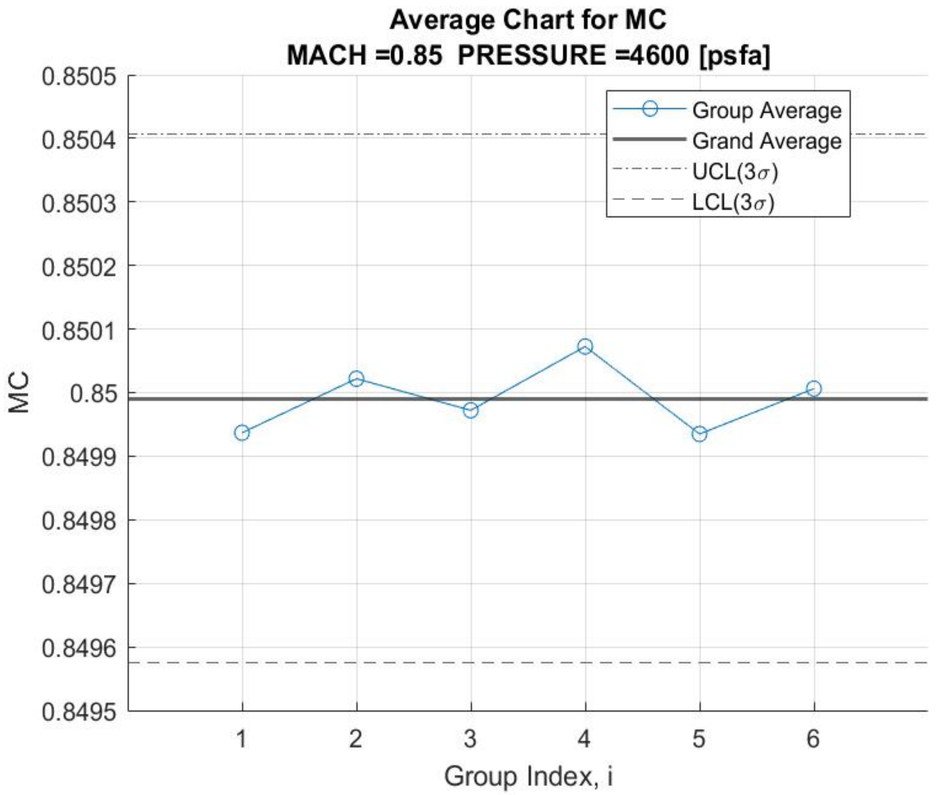
- Control Charts created for Short Static Pipe (SSP) that is used for check standards and tunnel calibration.
  - Control chart for variable averages, ranges, and moving ranges.
  - Track quality of tunnel conditions, individual taps, and average of 4 taps.

- MATLAB function can be used to create control charts for any time the SSP is ran.
  - Find all groups of runs with same test condition (5 conditions).
  - Average runs in group and compute statistical quality characteristic between groups.
  - Chart values with control limits held to 3 sigma.

SSP Installed in the 11ft. TWT

Results


Mach Uncertainty
PTOT =2120 [psfa]
Standard Deviation
Mach Number
0.003
0.0025
0.002
0.0015
0.001
0.0005
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6


Average Chart for MC
MACH =0.85 PRESSURE =4600 [psfa]
MC
Group Index, i
Group Average
Grand Average
UCL(3σ)
LCL(3σ)
0.8505
0.8504
0.8503
0.8502
0.8501
0.85
0.8499
0.8498
0.8497
0.8496
0.8495
1
2
3
4
5
6

Conclusions

- One program to calculate fossilized uncertainty, bias uncertainty, and random uncertainty associated with repeatability.
  - Can be expanded to include other types of random uncertainty (noise).
  - Eventual implementation of near-time uncertainty analysis for each run in tunnel.

- One program to create average, range, and moving range control charts for a number of variables including MC, PT, TTF, CPS for individual taps, and CPSBAR.
  - Will be used to track stability of centerline static pressure for each calibration test.
  - Eventual implementation of near-time control chart creation for calibration tests.

Lessons Learned

- Technical Skills:
  - Data analysis and statistical process control
  - Scientific research
  - MATLAB

- Soft Skills:
  - Virtual learning
  - Remote working
  - Individual accountability

Acknowledgements

- I would like to take the time to thank:
  - My mentor, Lindsey Drone
  - The members of the characterization team, Bethany White, Lindsey Drone, Max Amaya, and Ross Flach
  - Intern coordinates Abel Morelos, Carlos Aguirre, and Haley Fleck

References:

[1] - *NASA Digital Archive: ARC-1990-AC90-0481-8*
[2] - *NASA Digital Archive: ARC-1967-A38286-2*
[3] - *NASA Digital Archive: ARC-2017*
[4] - *AIAA ARC SSP 170123: Initial Testing of the Ames Unitary Short Static Pipe*
[5] - *All plots created from MATLAB functions*

Thank You.

Questions?