Helicopter Performance Modeling

Improving helicopter modeling in environmental models

David Senzig, November 5, 2014

Volpe The National Transportation Systems Center

Advancing transportation innovation for the public good

U.S. Department of Transportation
Office of the Secretary of Transportation
John A. Volpe National Transportation Systems Center

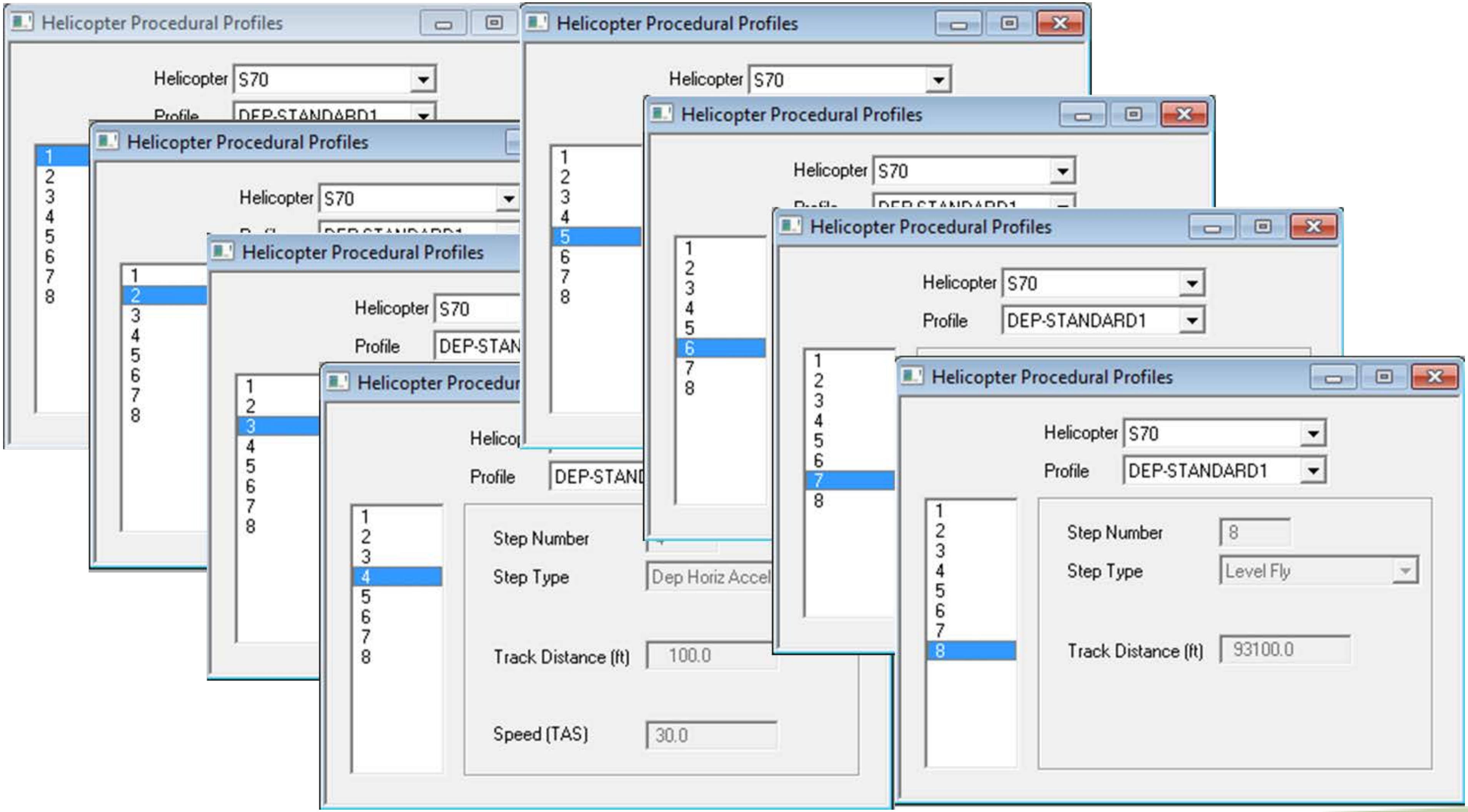
Helicopter Performance Modeling

- Current methods
- Goals
- Possible methods for improvements
- Next steps

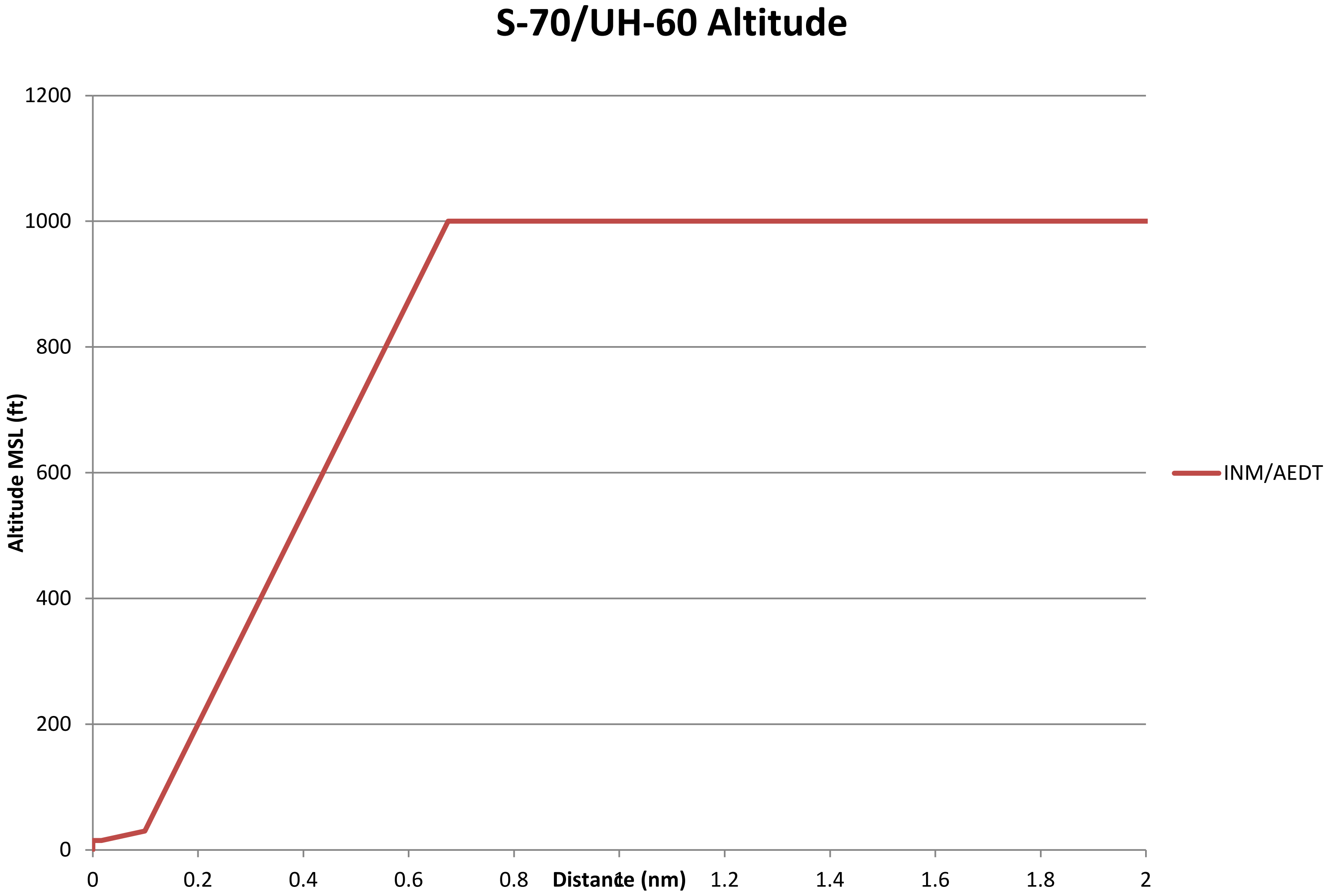
Current Methods

- No standards available for modeling performance
- FAA's Integrated Noise Model (INM) & Aviation Environmental Design Tool (AEDT) use modal methods

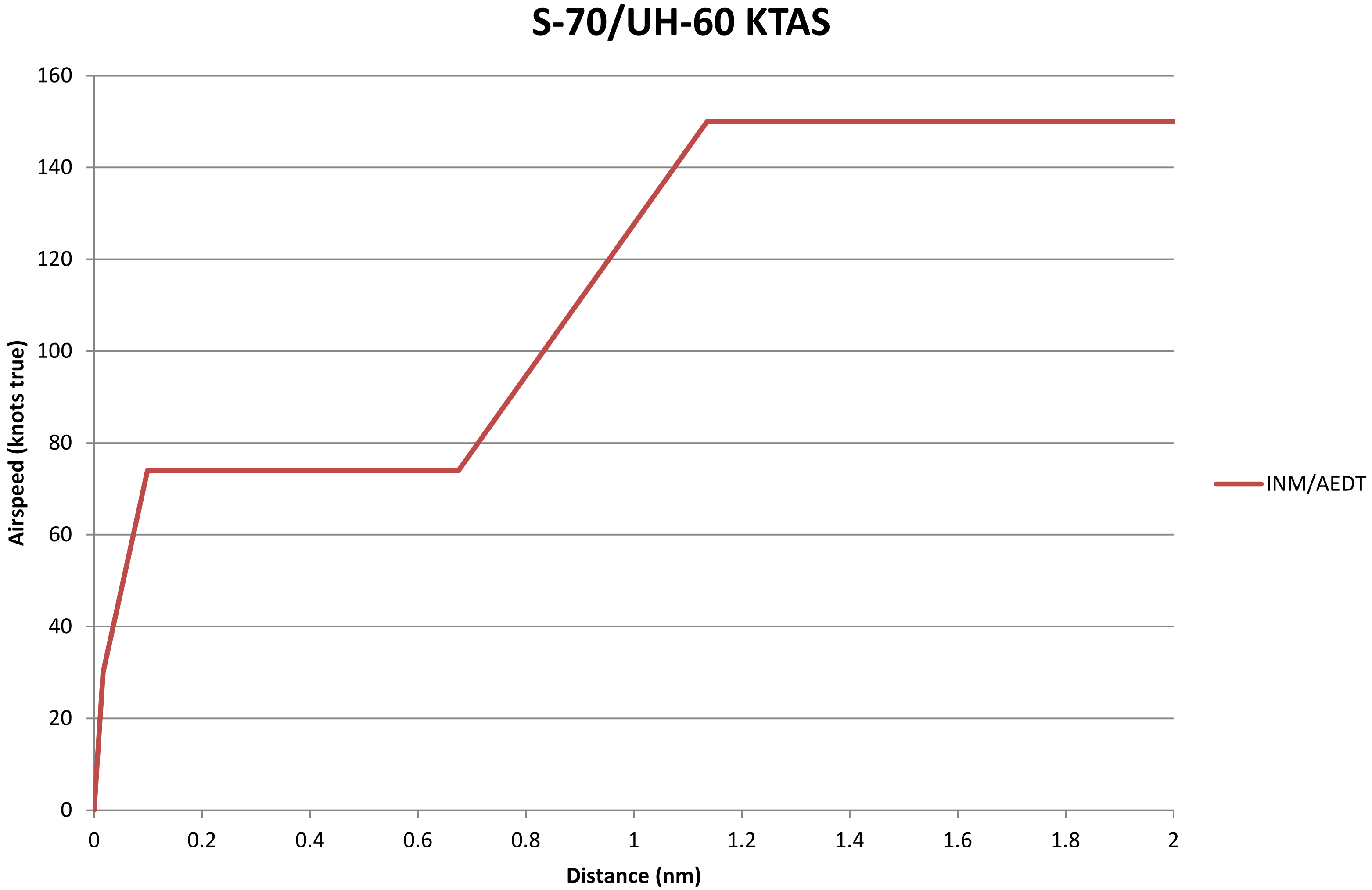
Current Methods – modal methods



Current Methods



Current Methods



Current Methods

- Mode-based performance, not physical
  - 14 modes available in AEDT
  - Not influenced by weight or atmospheric effects
- No power data, so no fuel consumption data directly available
- Fuel consumption set to ICAO climb-out (85% power) in AEDT
- Noise is mode-based
  - Modified by blade tip Mach number

Goals - technical

- Physics-based performance
- Use required and available power data
- Fuel consumption based on power required
- Improve noise modeling where possible

Goals - strategic

- Work with OEMs to improve methods
- Standard methods of performance (AIR?)
- Work with ACRP projects and other research venues to improve helicopter environmental modeling

Possible improvement methods

- Force-balance methods similar to SAE-AIR-1845
  - Or BADA Total Energy Methods
- Table look-up of torque (power)/thrust (weight) / speed (advance ratio)
  - Non-dimensional CT, CP (CQ), mu
- Fuel consumption from ICAO, U.S. Army methods, or table look-ups
- Emissions from ICAO database using BFF2, AIR 5715 methods

Possible improvement methods

- Helicopter thrust coefficient:
  - $C_T = W/\pi R^2 \rho V_t^2$
- Helicopter power and torque coefficient:
  - $C_P = C_Q = P/\pi R^2 \rho {V_t}^3$
- Helicopter advance ratio:
  - $\mu = V_0/V_t$

Possible improvement methods

UH60 Data
OUT OF GROUND EFFECT TORQUE COEFFICIENTS,
$CQ \cdot 10^5$ vs. $CT \cdot 10^4$ and μ

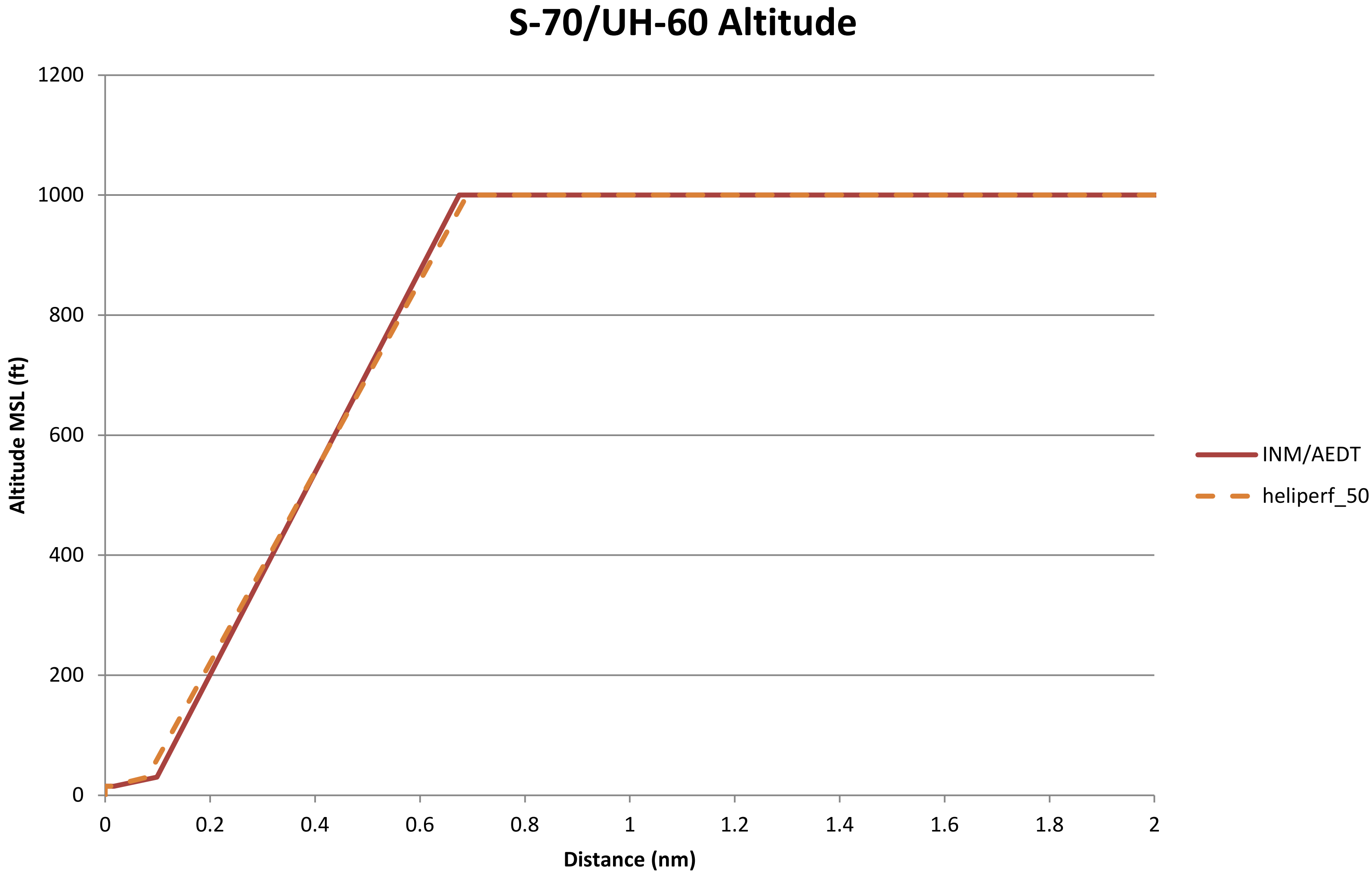
Weight →

NUMUD = 14				NCTD = 7			
μ/CT	66.0	72.0	78.0	84.0	90.0	96.0	102.0
0.00	63.2	70.9	79.0	87.3	95.9	104.9	114.1
0.12	34.5	38.5	43.5	48.4	54.0	60.2	68.2
0.14	34.0	37.2	41.5	46.0	50.7	56.5	63.5
0.16	33.0	36.2	39.7	43.8	48.2	53.5	60.0
0.18	32.6	35.5	38.5	42.5	46.7	51.5	57.8
0.20	32.7	35.2	38.4	42.4	46.5	51.4	57.1
0.22	33.5	36.0	39.2	43.0	47.2	52.0	58.0
0.24	35.2	37.8	41.0	45.0	49.2	54.2	60.5
0.26	38.3	40.7	44.0	48.0	52.6	58.0	65.5
0.28	41.7	44.5	48.5	52.5	57.7	64.2	73.0
0.30	47.0	49.6	54.0	59.0	65.0	73.0	86.0
0.32	54.0	56.1	61.0	66.3	74.2	85.0	102.5
0.34	62.0	64.2	69.5	76.1	85.3	99.0	130.0
0.36	71.5	74.0	80.0	87.5	100.1	118.5	155.0

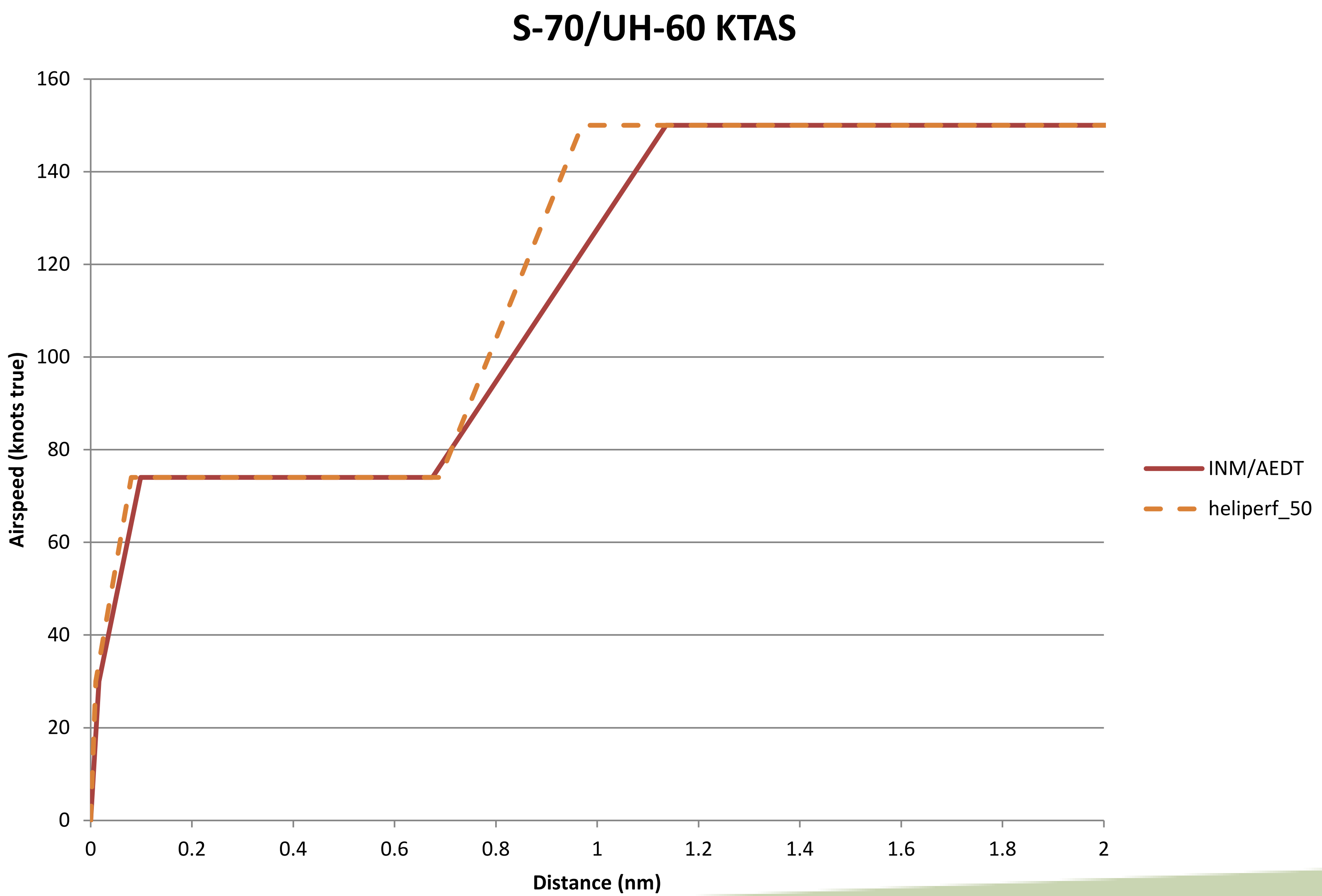
Speed ↓

Data in the middle of the table are power required

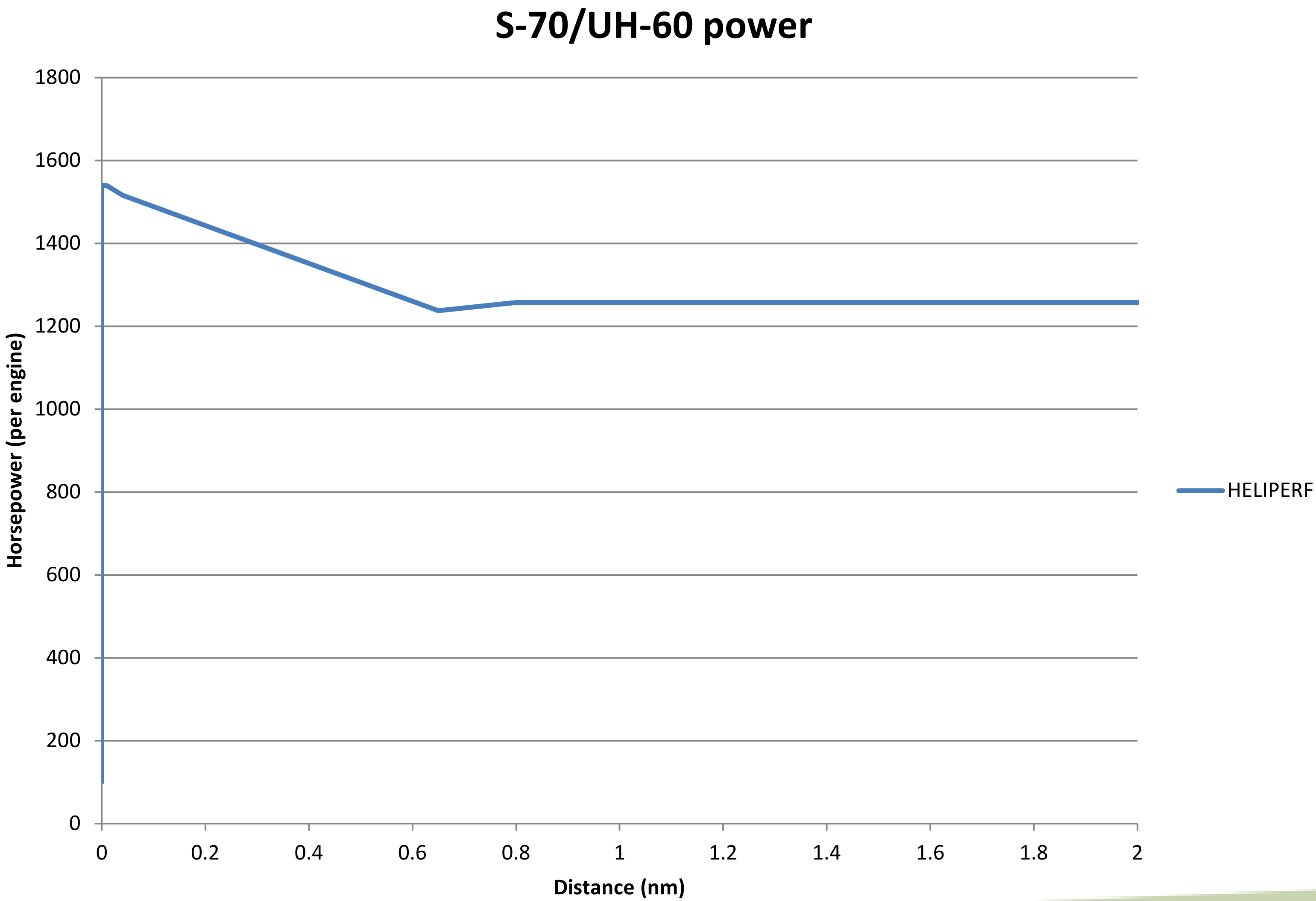
Possible improvement methods



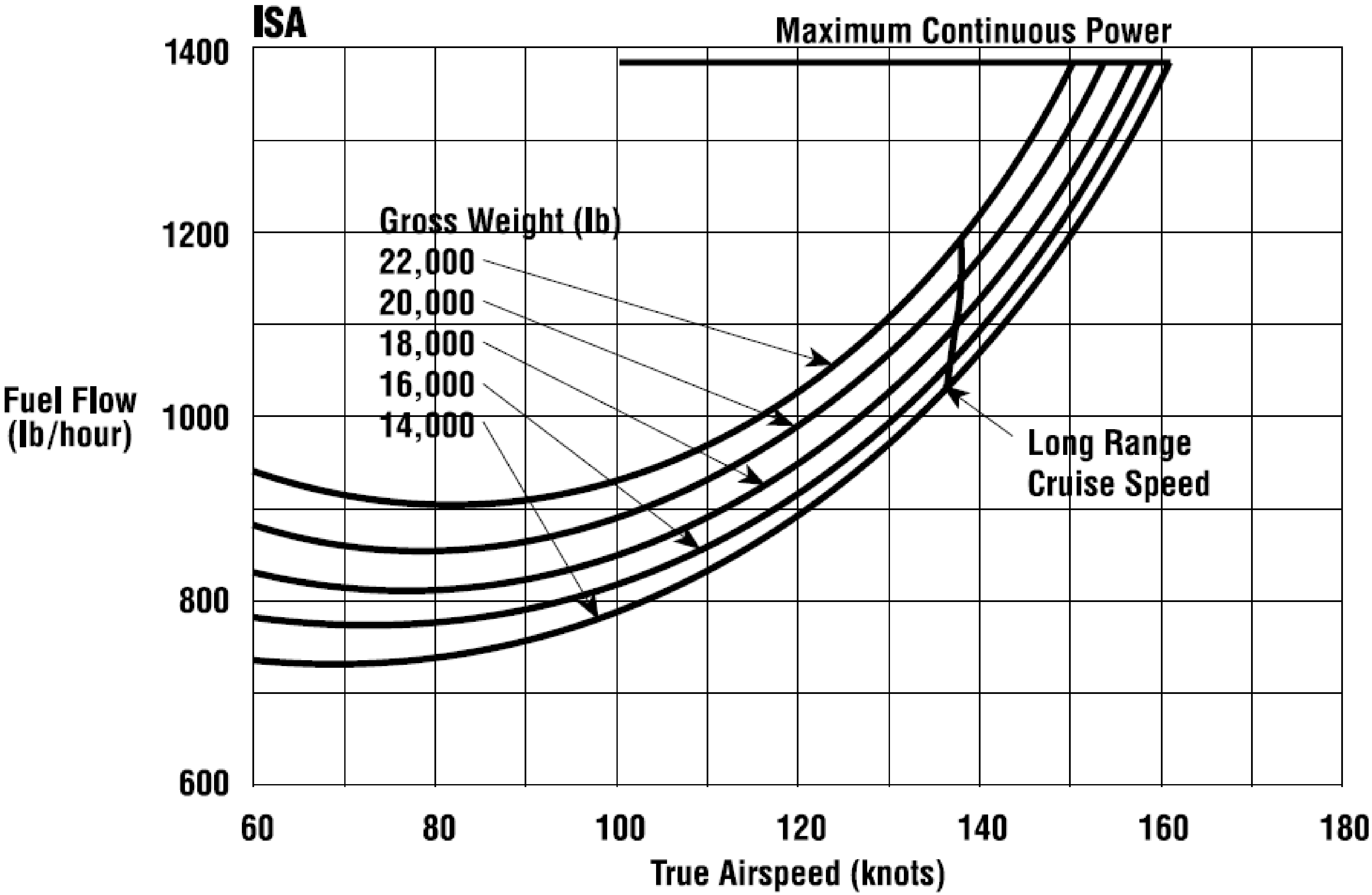
Possible improvement methods



Possible improvement methods



Possible improvement methods



Next Steps

- Concurrence of SMEs on best methods
- Determine what (if any) performance metrics can be used to predict noise
- Expand method to include fuel consumption
- Compare to FDR fuel data
- Do we want to decide now if an AIR is needed?

Backup Slides

- Pseudo-code for accelerating climb
- F90 code for accelerating climb
- Data required in current code

Pseudo code for INM/AEDT Climbing Segments

```
Do for each 1 KTAS increase

    Update KTAS

    Update ROC (assumes flight angle is constant)

    Update Power Available based on Translational Lift Speed

    Calculate μ (ratio of forward speed to main rotor tip speed)

    Lookup CQ from steady, horizontal data table (function of μ and CT)

    Calculate power required for horizontal flight

    Calculate power required for climb at ROC

    Calculate total Power Required (horizontal + climb)

    Calculate thrust angle from Power Required and Power Available

    Calculate Excess Thrust

    Calculate the acceleration from the excess thrust and the current weight

    Calculate the equations of motion to determine time and distance traveled

    Update speed, altitude, atmospheric data

End Do
```

Fortran code for Accelerated Climbing Segments

```
DO I = 1, N_KTAS ! Note that we need to update CT when we add the fuel consumption component

    NEW_KTAS = OLD_KTAS + 1.

    TEMP_NODE%POS%KTAS = NEW_KTAS
    RCS = NEW_KTAS * KTS2FPS * SIN(FLIGHT_ANGLE)
    TEMP_NODE%POS%ROCD = RCS * 60.

    CALL POWER_AVAILABLE(HELI,TEMP_NODE,TEMP_NODE%POS%ALTITUDE,LOCAL_WX%TEMP) ! Update Power available; in HELI_FUNC_f90
    CALL PAVAIL_F90(HELI,TEMP_NODE) ! Power Available, loads all the speeds at this call for the particular weight

    IF(CURRENT_NODE%STATE%IGE.AND.NEW_KTAS.LT.TRANSLATIONAL_LIFT_KCAS) THEN ! At the translational lift speed, we start using to MCP
        POWER_AVAILABLE_HP = TEMP_NODE%STATE%IRP%POWER_AVAILABLE_2ENG ! Use IRP if in gound effect
    ELSE
        POWER_AVAILABLE_HP = TEMP_NODE%STATE%MCP%POWER_AVAILABLE_2ENG ! Use MCP if not (assumes we are away from the dead man's curve)
    END IF

    MU = (NEW_KTAS * KTS2FTS) / HELI%MAIN_ROTOR%TIP_SPEED ! Check if this is true - isn't VTIP a function of VFTS also? Equation 4 in Kiwan

    DCQ = CDX * DSCFA * (MU * MU * MU) ! From CALL_DRAG in HELI_FUNC_F90, DSCFA is the change in the scale factor
    CQ0 = CQINT_F90(MU,TEMP_NODE%STATE%CT,HELI%MAIN_ROTOR%TIP_SPEED,CQD) !CQ required for horizontal flight; page 9 in Kiwan
    CQ = (CQ0 + DCQ) / HELI%MAIN_ROTOR%EFF ! correct the torque coefficient for drag and efficiency differences, Equation 15 in Kiwan

    POWER_REQUIRED_HORIZONTAL = CQ * TEMP_NODE%STATE%POWER_SCALE_FACTOR ! power required, equation 16 in Kiwan
    POWER_REQUIRED_CLIMB = RCS * (WEIGHT + 0.5 * CD_TOP * TOP_AREA * RHO * RCS*RCS)/HP ! Power is the speed *(weight + fuselage drag
    POWER_REQUIRED_HP = POWER_REQUIRED_HORIZONTAL + POWER_REQUIRED_CLIMB

    IF(POWER_REQUIRED_HP.GT.POWER_AVAILABLE_HP) THEN ! out of power
        PRINT *, " Out of power at ",NEW_KTAS, " knots and ",CURRENT_NODE%POS%ALTITUDE
        EXIT
    END IF

    THRUST_ANGLE = ACOS(POWER_REQUIRED_HP/POWER_AVAILABLE_HP)
    EXCESS_THRUST = FIND_EXCESS_THRUST(TEMP_NODE%STATE%WEIGHT,THRUST_ANGLE) ! thrust is pounds, Not fed back to reported power , HELI_FUNC_F90
    ACCEL = HELI_ACCEL_FROM_TEM(EXCESS_THRUST,NEW_KTAS,TEMP_NODE%STATE%WEIGHT,RCS) ! heli_func_f90, BADA total energy model

    DELTA_TIME = KTS2FPS / ACCEL
    DELTA_S = ((NEW_KTAS * KTS2FPS)**2 - (OLD_KTAS * KTS2FPS)**2) /(2. * ACCEL)
    TIME = TIME + DELTA_TIME
    S = S + DELTA_S

    OLD_KTAS = NEW_KTAS

    TEMP_NODE%POS%ALTITUDE = TEMP_NODE%POS%ALTITUDE  + RCS * DELTA_TIME
    CALL UPDATE_WX(LOCAL_WX,TEMP_NODE%POS%ALTITUDE)

END DO
```

Data required for current code

- Basics: TOW, number of engines, size of rotors, RPM, distance from main rotor to tail rotor, etc – HNM/INM/AEDT or public data
- OGE CQ as a function of CT and μ (tabular, 2D)
- IGE CQ as a function of CT (tabular, 1D)
- Drag data: area and CD for front and top
- Engine power (IRP and MCP) as a function of altitude and speed
- Need main rotor profile drag data if we want to break out tail rotor power (not currently required)