

by Sean Lynn, and converted to c by Pete MacMillin, with improved numerics

TAKEOFF2.c is a c implementation of a modified approach proposed by Krenkel and Salzman to solve for aircraft takeoff. The original methodology solved the aircraft equations of motion parametrically, whereas this program solves through a time-step integration technique. The methodology was also modified to calculate Balanced Field Length (BFL) for preliminary design purposes. Balanced field length is often, as here, defined as the distance required to make either a One Engine Inoperative (OEI) takeoff (including obstacle passage) or a braked stop when an engine fails at critical engine failure speed.

Krenkel and Salzman assumed thrust vectoring capability and thrust variation with velocity to create the balance of forces equations on the aircraft during its takeoff run. All necessary input parameters can be obtained from the performance parameters necessary for preliminary aircraft design. Some sources for estimating design parameters are located below.

The program is broken into two parts, each describing different aspects in a takeoff analysis:

- 1) Normal Takeoff - From stop to liftoff to passage over a 35ft (11m) obstacle.
- 2) Balance Field Length calculation - Iterative solution to find where the engine can fail so that the distance to perform a OEI takeoff is equal to the distance to brake to a stop.

Each section prints out important times, distances and velocities for the takeoff run. For normal takeoff, incremental time distance and velocity data is output as well.

## **References**

Krenkel, A.R., Salzman, A., "Takeoff Performance of Jet-Propelled Conventional and Vectored Thrust STOL Aircraft", *Journal of Aircraft*, Vol. 5, No. 5, 1968, pp. 429.

Roskam, Jan, *Airplane Design Part 1: Preliminary Sizing of Airplanes*, Roskam Aviation and Engineering Corp., Ottawa, KS, 1989.

Torenbeek, Egbert, *Synthesis of Subsonic Airplane Design*, Kluwer Academic Publishers, Norwell, MA, 1982.

## Input Format

The input is placed in a file called takeoff.in, designed to be self-explanatory and to provide a ready reference to some of the input parameters. As will be seen, each input line contains a description of the input necessary for that line. Note that all input is in English engineering units, as is the output. A sample input file is shown next.

### Sample Input:

```
dc9 Mar 97 run < case title, can be up to 80 characters long
0.002376900 < Atmospheric density (slug/ft^3)
95000.0 < Weight of aircraft (lbs)
1000.0 < Wing area (ft^2)
2.000 < CLmax - maximum lift coefficient of the aircraft
0.30 < CLgrd - lift coefficient for ground run
1.6500 < CLair - lift coefficient for climb segment
.080 < CDgrd - drag coefficient for ground run
0.121 < CDair - drag coefficient for climb segment
.025 < MUgrd - rolling friction coefficient *Note 1*
.3 < MUBrk - braking friction coefficient *Note 2*
0.0 < lambda - thrust deflection angle
1.1 < k - stall margin *Note 3*
3.0 < Time between engine failure and braking (sec) *Note 4*
35. < Obstacle height (ft)
2 < Number of engines
31450.0 29835.0 28475.0 < 3 thrusts (lbs) *Note 5*
0. 111.6 334.0 < 3 velocities (for thrusts) (ft/s)
3.0 < Time required for rotation
2 < Print flag
```

Note 1: Rolling friction coefficient is typically:

Dry concrete/Asphalt	- 0.02
Hard turf/Gravel	- 0.04
Short, dry grass	- 0.05
Long grass	- 0.10
Soft Ground	- 0.10 to 0.30

Note 2: Braking friction coefficient is typically:

0.20 to 0.40 with good assumptions being, 0.30 or 0.35

Note 3: Takeoff speed is usually defined as  $V_{to} = k * V_{stall}$ , where  $k$  is the stall margin and  $V_{stall}$  is the aircraft stall speed.  $k$  is usually defined as 1.1, although 1.2 is also used, (ie. the takeoff speed is 10% to 20% higher than stall speed).

Note 4: This is the time lag between engine failure and the decision to begin braking.  
(by MIL-M-007700B this is 3 sec after failure)

Note 5: These three thrusts are used to calculate a quadratic thrust curve for the aircraft engine. Each thrust should correspond to the velocity below it. For cases with unknown thrust curves a constant thrust can be entered for three different velocities.



