slipslide Group

John D. Baker

https://github.com/bakerjd99/jacks/blob/master/slipslide/slipslide.ijs

SHA-256: 82f8d1c7ee31754a4169c4a92127dac5dc6c2e01e0d58d9d7cb29bf5e69609c0

December 28, 2023

Contents

slipslide Overview

slipslide is a J script that estimates how far slowly moving (< 20 m/sec) objects slide on a a perfectly flat frictionless plane when only acted upon by stationary sea-level air resistance.

slipslide was written to explore side topics that came up during the composition of a story.

slipslide Interface

```
lyinghuman    [6] slide parameters for a human lying down facing wind
shootermarble [7] slide parameters for 19mm glass shooter marble
slipslide0    [8] estimate slide of object on frictionless plane
```

Getting Ziggy with slipslide

The J slipslide0 verb does not play to J's strengths. Loopy J code does not exhibit stellar performance. I write loopy code when I'm thinking things through. Later, I will recode, sometimes in J and other languages. In slipslide0's case, I used the new programming language zig.

Zig code for slipslide0 is stored in the play dictionary.

```
NB. display zig slipside0
require 'general/jod'
od ;:'play utils' [ 3 od ''
4 disp 'slipslide0_zig'
```

There is a simple build test that compiles this code.

```
NB. display build test
1 disp 'build_slipslide_zig'
```

Build executable.

```
rtt 'build_slipslide_zig'
```

Even the zig debug version of `slipslide0` way outperforms J.

```
NB. load slipslide script - make with mls 'slipslide'
load 'slipslide'

9!:11 [ 17  NB. high print precision

NB. times (seconds) and space (bytes) to execute sentence
ts=:6!:2 , 7!:2@]

NB. marble moving two hours
marb=: 7200000 0.001,shootermarble 1

NB. time space J
ts '(|. 2{.marb) slipslide0 2 }. marb'

NB. J temp path - shell verb
tmp=: linpathsep`winpathsep@.(IFWIN) jpath '~temp/'
sh=: (2!:0)`shell@.(IFWIN)
```

```
NB. time space zig
ts 'sh tmp,''slipslide0 '','':marb'
```

slipslide Source Code

```
NB.*slipslide  s--  estimate  slide  distance  of  objects  on
NB. frictionless plane.
NB.
NB. Estimate how  far  objects  will slide  on  a  perfectly flat
NB. frictionless plane  when acted  upon  only  by stationary air
NB. resistance.
NB.
NB. verbatim: interface word(s):
NB. ------------------------------------------------------------------------
NB.  lyinghuman    - slide parameters for a human lying down facing wind
NB.  shootermarble - slide parameters for 19mm glass shooter marble
NB.  slipslide0    - estimate slide of object on frictionless plane
NB.
NB. created: 2023Dec21
NB. ------------------------------------------------------------------------
NB. 23dec26 slight refactor - correct final count
NB. 23dec27 compare with zig version

coclass 'slipslide'


NB.*end-header

NB. interface words (IFACEWORDSslipslide) group
IFACEWORDSslipslide=: <;._1 ' lyinghuman shootermarble slipslide0'
```

```
NB. root words (ROOTWORDSslipslide) group
ROOTWORDSslipslide=: <;._1 ' IFACEWORDSslipslide ROOTWORDSslipslide VMDslipslide linpathsep lyinghuman shoo
>..>termarble slipslide0 winpathsep'


NB. version, make count and date
VMDslipslide=: '0.5.1';6;'27 Dec 2023 16:12:51'

NB. standardizes path delimiter to linux forward / slash
linpathsep=: '/'&(('\' I.@:= ])} )

lyinghuman=: 3 : 0

NB.*lyinghuman v-- slide parameters for a human lying down facing wind.
NB.
NB. monad:  fl =. lyinghuman faV
NB.
NB.   lyinghuman 8.8  NB. roll down frictionless 4m

NB.   air density (kg/m^3)
NB. https://www.wolframalpha.com/input?i=air+density+at+sea+level+in+kilograms+per+cubic+meter
rho=. 1.226

NB. human mass (kg)
hm=. 75

NB. drag coefficient around same as car
```

```
NB. https://physics.info/drag/
c=. 0.35

NB. head forward cross section area (m^2)
ha=. 0.2

NB. air, drag, area, mass, velocity
rho,c,ha,hm,y
)

shootermarble=: 3 : 0

NB.*shootermarble v-- slide parameters for 19mm glass shooter marble.
NB.
NB. monad:  fl =. shootermarble faV
NB.
NB.   shootermarble 1     NB. 1 m/sec
NB.   shootermarble 8.8   NB. roll down frictionless 4m

NB.   air density (kg/m^3)
NB. https://www.wolframalpha.com/input?i=air+density+at+sea+level+in+kilograms+per+cubic+meter
rho=. 1.226

NB. glass density (kg/m^3)
NB. https://www.wolframalpha.com/input?i=2520+kilograms+per+cubic+meter&assumption=%22ClashPrefs%22+-%3E+%
>..>22%22
gd=. 2520
```

```
NB. radius shooter marble (m)
NB. https://www.moonmarble.com/t2-marbleinfo.aspx
rm=. 0.0095

NB. mass of shooter marble (kg)
mm=. gd * (4%3) * 1p1 * rm^3

NB. drag coefficient ideal sphere
NB. https://physics.info/drag/
c=. 0.5

NB. area shooter marble (m^2)
ma=. 1p1 * rm^2

NB. air, sphere drag, area marble, mass marble, velocity
rho,c,ma,mm,y
)


slipslide0=: 3 : 0

NB.*slipslide0  v-- estimate  slide  of  object  on  frictionless
NB. plane.
NB.
NB. This verb estimates how far a slowly  moving <20 m/sec object
NB. will slide  on a perfectly flat frictionless  plane when only
NB. acted upon by air resistance.
NB.
```

```
NB. verbatim:
NB.
NB. The basic formula is: R = ½ CAv^2  https://physics.info/drag/
NB.
NB. R   drag force (Newtons) (kg*m/sec^2)
NB.     air density (kg/m^3)
NB. C   coefficient of drag
NB.     constant determined by experiment
NB. A   projected area (m^2)
NB. v   velocity (m/sec)
NB.
NB. monad:  flSva =. slipslide fl
NB.
NB.   NB. air, sphere drag, area marble, mass marble, velocity
NB.   slip=. shootermarble 1
NB.   slipslide0 shootermarble 1
NB.
NB. dyad:  flSva =. fldTCnt slipslide fl
NB.
NB.   NB. zig test case - show many digits
NB.   9!:11 [ 17
NB.   0.001 25 slipslide0 shootermarble 1
NB.
NB.   NB. a 1 m/sec marble is still slowly moving
NB.   NB. after 2 hours and has rolled around 1/2 km
NB.   (0.001,1000 * 3600 * 2) slipslide0 slip
NB.
```

```
NB.   NB. spreadsheet cross check
NB.   0.001 19970 slipslide0 slip
NB.
NB.   NB. a human is still sliding after two hours
NB.   (0.001,1000 * 3600 * 2) slipslide0 lyinghuman 8.8

0.001 1000 slipslide0 y
:

'rho C A M vn'=. y [ 'dT cnt'=. x

NB. drag constant
drgc=. 0.5 * rho * C * A

NB. initial acceleration and drag
an=. rn % M [ rn=. drgc * vn^2

S=.  0  NB. total distance

for_step. i. cnt do.
  dS=. dT * vn       NB. step distance
  vn=. vn - an * dT  NB. new velocity (decreasing)

  NB. new acceleration and drag
  an=. rn % M [ rn=. drgc * vn^2

  S=. S + dS
```

```
  NB. smoutput step, dS, vn, S
end.

NB. distance, end velocity, acceleration, step count
S,vn,an,cnt
)

NB. standardizes path delimiter to windows back \ slash
winpathsep=: '\'&(('/' I.@:= ])} )

NB.POST_slipslide post processor.

smoutput IFACE=:  (0 : 0)
NB. (slipslide) interface word(s): 20231227j161251
NB. --------------------------
NB. lyinghuman     NB. slide parameters for a human lying down facing wind
NB. shootermarble  NB. slide parameters for 19mm glass shooter marble
NB. slipslide0     NB. estimate slide of object on frictionless plane
)

cocurrent 'base'
coinsert  'slipslide'
```

Index