Algorithm 1: Generate random 3D trajectory (Initial basic version)

Initialize: $a = 0.9, b = 1, \boldsymbol{x}_0 = (0,0,0), \boldsymbol{q}_0, \boldsymbol{v}_0 = (0,0,0)$

/* $\boldsymbol{x}_0$ is the initial position vector, $\boldsymbol{q}_0$ is the quaternion describing initial orientation, and $\boldsymbol{v}_0$ is the initial velocity vector */

for $t \leftarrow 0$ **to** T **do**

  if *check for obstacles on line:* $\boldsymbol{x}_t + \alpha \boldsymbol{v}_t$ *is true* **then**

    /* $\alpha > 0$ tells the look ahead distance */

    /* change heading or velocity slightly towards better direction */

    $\boldsymbol{v} \leftarrow \boldsymbol{v}_t$

    while *check for obstacles on line:* $\boldsymbol{x}_t + \alpha \boldsymbol{v}$ *is true* **do**

      $\zeta \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

      $\boldsymbol{v} \leftarrow a\boldsymbol{v}_{t-1} + b\zeta$

      /* note: $\boldsymbol{v}$ is just a temporary velocity variable until we find a better direction */

    end

    $\boldsymbol{v}_t \leftarrow \boldsymbol{v}$

  end

  $\boldsymbol{x}_{t+1} \leftarrow \boldsymbol{x}_t + \boldsymbol{v}_t$

  /* sample η from multivariate (3 dimensional) standard normal distribution */

  $\eta \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

  $\boldsymbol{v}_{t+1} \leftarrow a\boldsymbol{v}_t + b\eta$

  /* Quaternion orientation is not required for initial version */

  /* can skip this update */

  $\boldsymbol{q}_{t+1} \leftarrow \mathsf{FindQuaternion}(\boldsymbol{v}, \mathbf{y}_{\mathsf{world}})$

end

Function FindQuaternion ($\boldsymbol{v}$, $\mathbf{y}_{world}$)

  /* assumes our heading in always along y axis of 'body' frame */

  Normalize $\boldsymbol{v}$ to unit vector $\hat{\boldsymbol{v}}$

  $\theta \leftarrow \arccos(\hat{\boldsymbol{v}} \cdot \boldsymbol{y}_{world})$

  $\hat{\boldsymbol{u}} \leftarrow (\hat{\boldsymbol{v}} \times \boldsymbol{y}_{world})$

  $\boldsymbol{q} = (\ \cos(\frac{\theta}{2}),\ \sin(\frac{\theta}{2})\ \hat{\boldsymbol{u}}\ )$

  return $\boldsymbol{q}$

Sample trajectory (below)

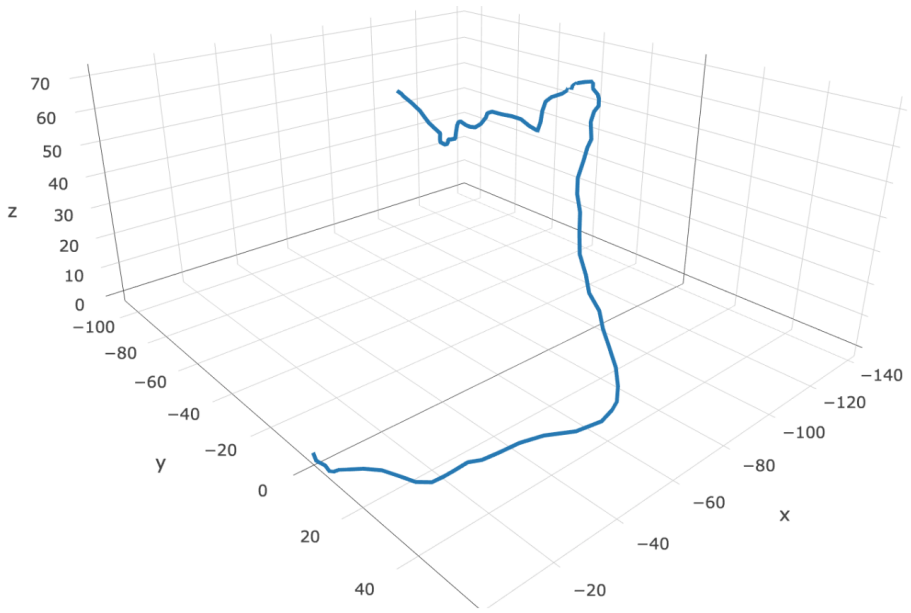

Figure 1: Sample 3D trajectory generated by above algorithm (without quaternions)