## Assignment for Week 5 readings: (due Tues Nov. 16)

Federico Bernabei, Muhammad Sunny Nazeer, Michele Francesco Penna, Ebrahim Shahabishalghouni and Mark Cutkosky. For any questions or help: [Contattaci: #week5 sul Slack](https://topicsinmulti-b5g1638.slack.com/archives/C02KQ2DGGCW)

Goal: How to describe and solve rolling contact problems for manipulation, following the formalisms from the classic Montana (1988) paper.

* [Week 5 Googleslides presentation](https://docs.google.com/presentation/d/1Qu07KbBA4mMohoL-noqDr-M44IgdbfJxyLUmsM5WoYA/edit?usp=sharing) All files are in [Week5 Folder on Google drive](https://drive.google.com/drive/folders/1I1pYoH2tEftNzy6uw15T6PJBrbSrV_9J?usp=sharing)

### Q1

The differential geometry formalisms in the Montana IJRR paper describe arbitrary smooth, curved surfaces (manifolds) on which rolling could take place. For some of this material you may also find it helpful to look at the [appendix of A. Okamura’s thesis](https://drive.google.com/file/d/12vazR-X4PstOcoznA5nAUbxCyZ0NifIm/view?usp=sharing), which covers some of the same material. An important idea is that any smooth surface can locally be described in terms of three matrices: [K], [T], [M] for *curvature*, *torsion* and *metric*, respectively.

For the case of a sphere rolling on a flat or rounded object, the general equations lead to simple results. The symbolic algebra script **MontanaKmat.py** uses Sympy to derive the [K] matrix following Montana’s method, starting with the definition of a spherical coordinate system (eq 11) and applying equations (6) to obtain the result in (14) and then applying (7-9) to obtain (15). Your job is to finish the derivation to compute also the [M] and [T] matrices. If it all works, you should get the results of equation (15) for the case of a sphere.

### Q2

Consider a sphere rolling on a flat surface -- as could be the case for rounded fingertips manipulating a rectangular block or [CMU’s BallBot](https://www.ri.cmu.edu/robot/ballbot/) or [DIsney’s BB-8](https://disney.fandom.com/wiki/BB-8) travelling on a flat surface. The [K], [T] and [M] matrices are even simpler for a flat surface than for a sphere. Montana provides them for his Example 2, (eq 40) although we should be careful in using those results because he assumes 𝝍 = 0.

We have set up a Sympy script, **Sphere-on-Flat-roll-new.py** to derive the rolling equations for a sphere ‘obj1’ on a plane ‘obj2’. The script assumes no slippage (vx = vy = 0) and that the frames on obj1 and obj2 are initially aligned (𝝍 = 0) but they can deviate over time.

**Q2.1** The values in the script are currently set so that only ωx is nonzero, i.e., there is only angular velocity about the local x axis. *Without running the script*, what do you expect the trajectory of the common contact point to be in the (x,y) coordinate system on the flat surface?   
Now run the script and see if it matches your expectation.

**Q2.2** Can you find a set of velocities for which the motion of the contact traces a circle in the (x,y,) plane? Assuming you find something that works, does the circle diameter match your expectation?

**Q3.3** A more subtle question: Why is the shape of the trajectory for having ωy = 0 (and the other omega terms nonzero) different from having only ωx = 0 ?