Surface wrinkling of a thin liquid-infused membrane

Jiayu Wang[1], Arnaud Antkowiak[1], Sébastien Neukirch[1]

Institut Jean le Rond ∂'Alembert, Sorbonne Université, 75005, Paris, France
jiayu.wang@sorbonne-universite.fr

Elasto-capillary effects are due to the interactions between surface tension and elastic deformation of slender structures. Recent studies have highlighted diverse deformations induced by capillary forces, such as softening the sharp geometry of a soft substrate, bending and buckling of flexible fibers, and folding, wrapping, and wrinkling of thin sheets[1]. Our study builds upon experimental observations of a nano-fibrous liquid-infused tissue[2]. Under slight compression, it spontaneously develops wrinkles through elasto-capillarity effects. Upon further contraction, specific regions undergo substantial collapse, forming surface reservoirs that enhance the membrane's deformability. The remarkable deformability of this synthetic system closely resembles that of cell membranes, making it intriguing for applications in stretchable electronics, smart textiles, and soft biomedical devices.

To elucidate the underlying mechanism, we employ theoretical and numerical modeling. The system is simplified to a thickness-neglected, inextensible membrane confined within a liquid layer of constant volume. The configuration of the membrane-liquid system under certain compression is framed as an optimization problem and the nonlinear problem is solved with the help of the open-source tool CasADi[3].

The model reveals homogeneous wrinkles with slight compression. As compression increases beyond a certain threshold, wrinkles localize at one specific spot on the membrane. This transition provides insights into experimental observations, prompting further investigations into the behaviors of liquid-infused membranes.

Références

1. J. Bico *et al.*, *Annual Review of Fluid Mechanics*, **50.1**, p. 296-299 (2018).
2. P. Grandgeorge, *Science*, **360.6386**, p. 629-659 (2018).
3. J. Andersson *et al.*, *Mathematical Programming Computation*, **11.1**, p. 1-36 (2019).