MovementI

O move(lv: real, av: real)

ObstacleI

⚡ obstacle

🔷 marXbotSoftware

🖥 marXbotServices

P MovementI
ⓘ ObstacleI

obstacle

ref Movement

obstacle

async

⚗ Movement

R MovementI
ⓘ ObstacleI

ref stm_ref0 = SMMovement

obstacle

obstacle

⚗ SMMovement

R MovementI
π lvel: real, avel: real
π PI: real
🕓 turnTimer
ⓘ ObstacleI

obstacle

ⓘ

Moving

entry move(lvel, 0); wait (1)

obstacle #turnTimer

Turning

entry move(0, avel)

[since(turnTimer)>=PI/avel]