

EUROGRAPHICS 2016
THE 37th ANNUAL CONFERENCE OF THE
EUROPEAN ASSOCIATION FOR COMPUTER GRAPHICS

Character contact re-positioning under large environment deformation

Steve Tonneau[1], Rami Ali Al-Ashqar[2], Julien Pettré[3],
Taku Komura[2], Nicolas Mansard[1]

[1]CNRS-LAAS [2]University of Edinburgh [3]Inria Rennes

Motion capture does not consider contact interactions



Motion capture does not consider contact interactions

Virtual characters are very good mimes

Geometry does not matter...
… except for the user !



Motion capture does not consider contact interactions

Virtual characters are very good mimes

Geometry does not matter...
… except for the user !

How to adapt the motion to
position errors / deformed geometries?



⇒ A very rich list of contributions on motion editing
[Watkin et al. 95,Kovar et al. 02, Pettré et al. 03, Choi et al. 03,Kim et al. '09, Levine '12...]

Relationship descriptors [Al Ashqar et al. 2013]

Joint positions described by:

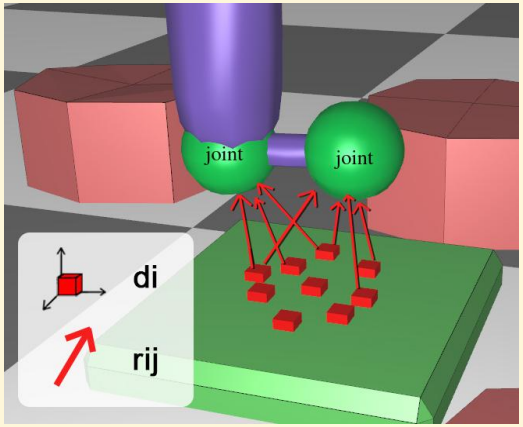
a configuration **q**

Relationship descriptors [Al Ashqar et al. 2013]

Joint positions described by:

a configuration **q**

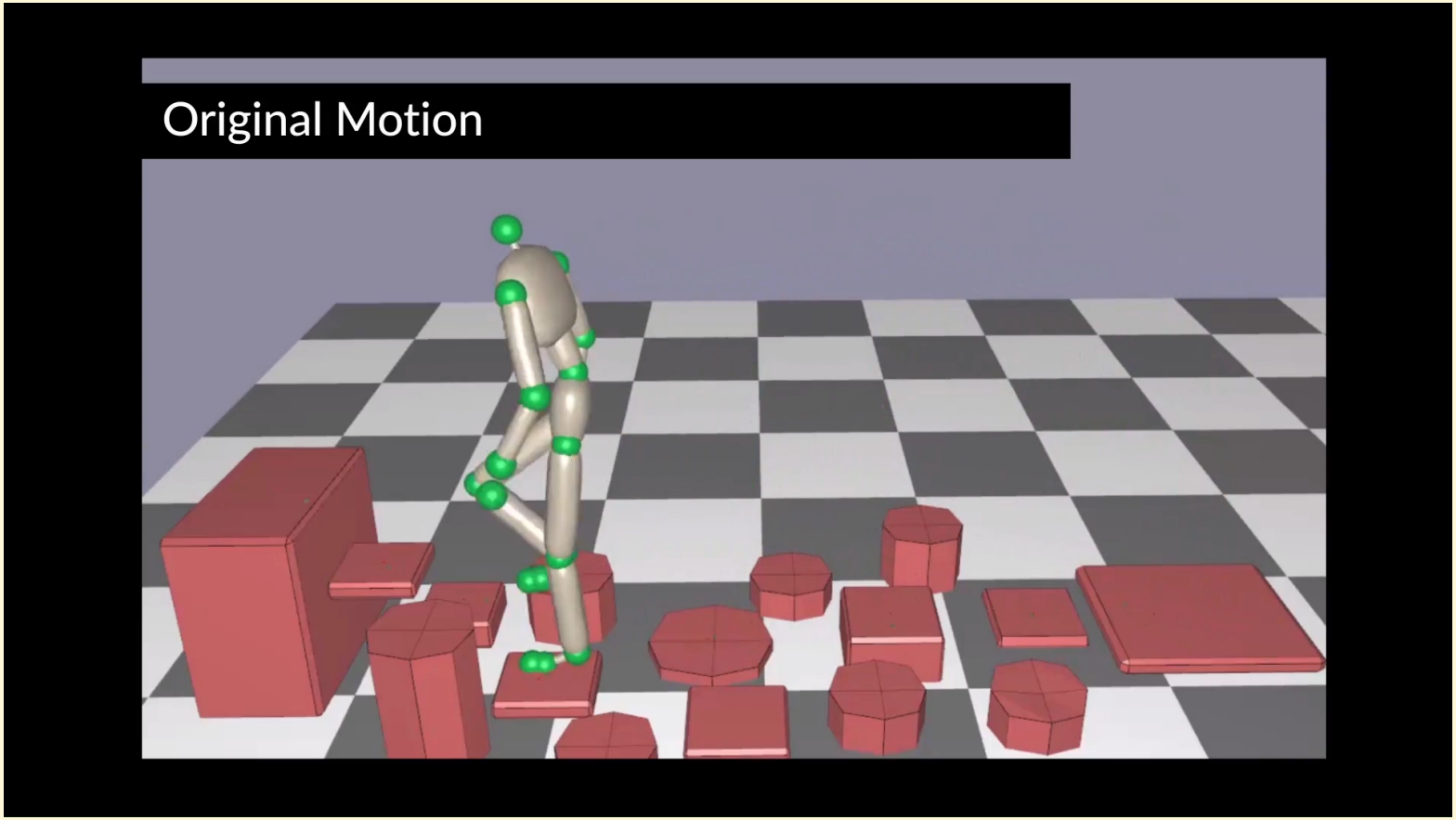
relative positions to elements of geometry



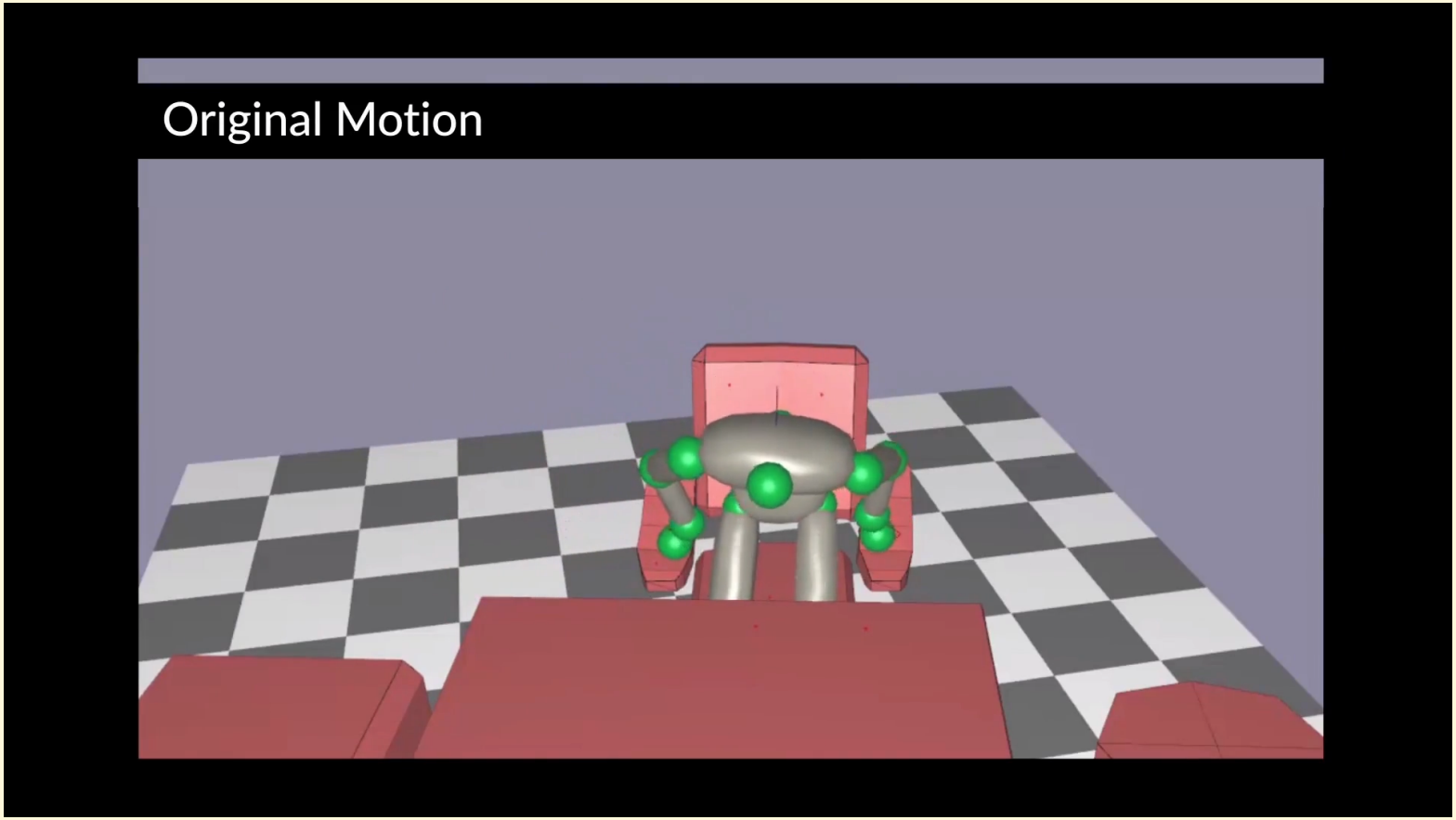
Geometry deformed =>

update joint position to preserve relationship

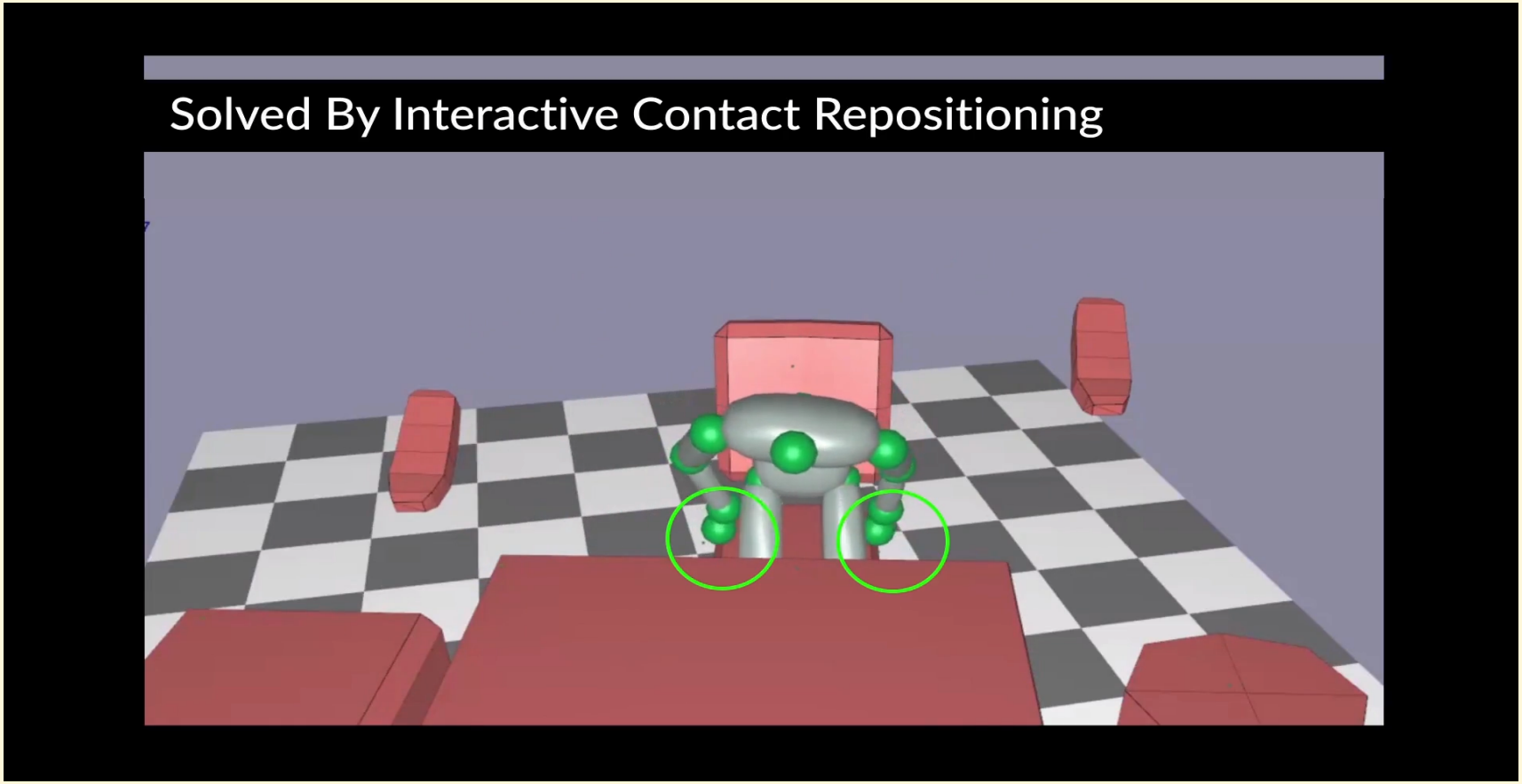
Relationship descriptors [Al Ashqar et al. 2013]



Motion editing fails when deformation is too important



Solution: use contact repositioning



Paper contribution

We propose a framework to automatically:

Detect contact failure upon geometry deformation

Automatically compute replacement contact

Adapt reference motion to new contact:

- With fidelity w.r.t. original motion
- Preserving motion continuity

Contact generation through procedural techniques

- Contact Invariant Optimization
  [Mordatch et al. '12]

- Monte Carlo approaches
  [Hämäläinen et al. '15]

- Sampling based, discrete contact planning
  [Bretl et al. 05, Escande et al. '08]
  [Tonneau et al. '15-16, Carpentier et al. '15]

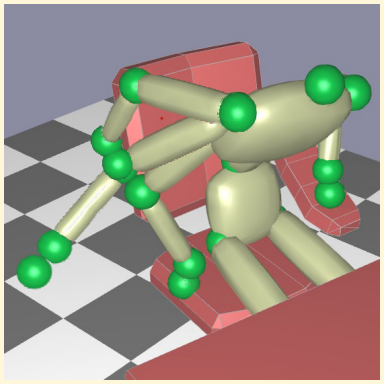
Contact generation through procedural techniques

- Sampling based, discrete contact planning
  [Tonneau et al. '15-16, Carpentier et al. '15]



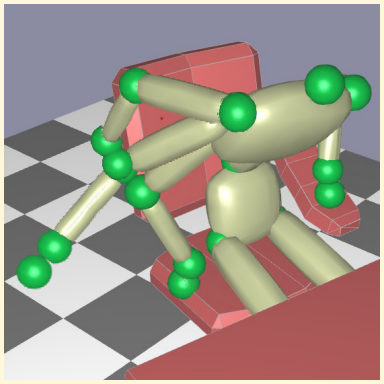
Generating a contact for one limb

Generating a contact for one limb

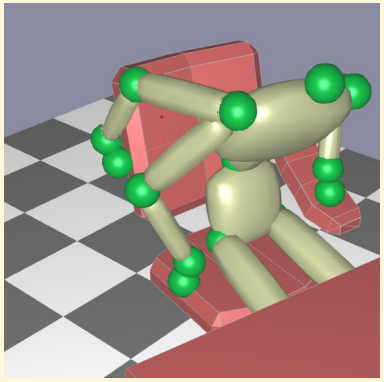


Sampling limb candidates

Generating a contact for one limb

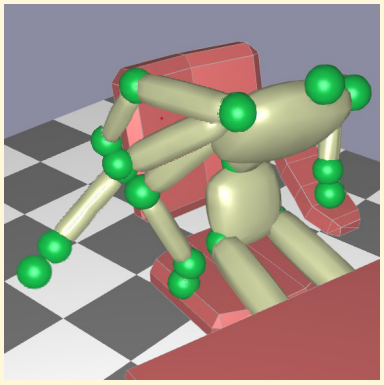


Sampling limb candidates

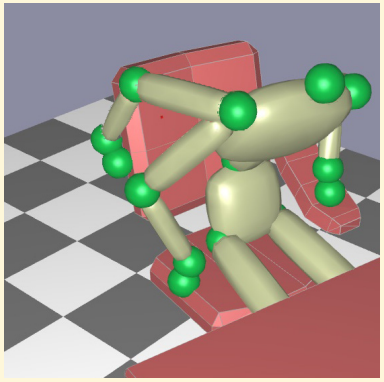


Proximity query

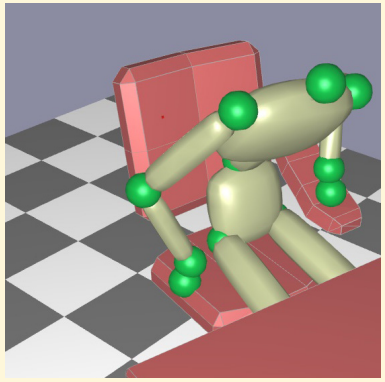
Generating a contact for one limb



Sampling limb candidates

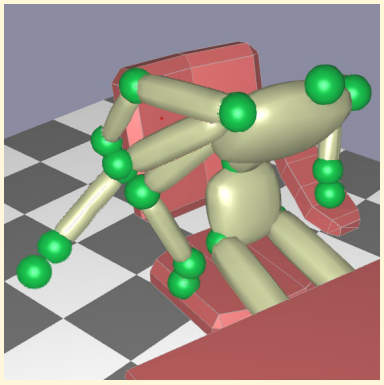


Proximity query

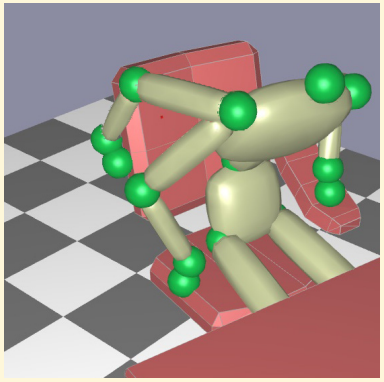


Keep best candidate

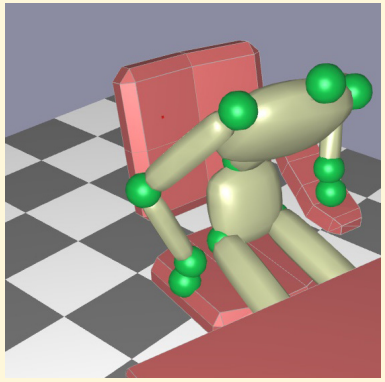
Generating a contact for one limb



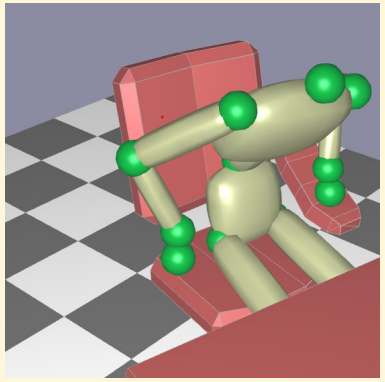
Sampling limb candidates



Proximity query

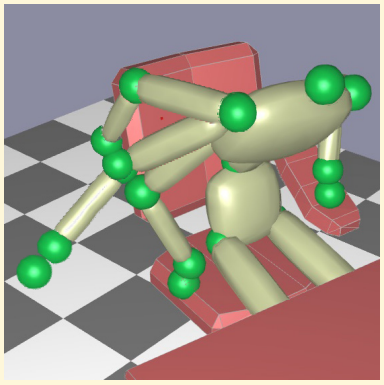


Keep best candidate

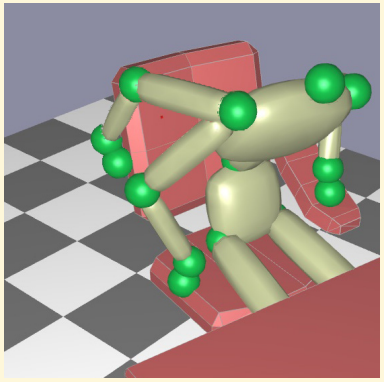


Projection

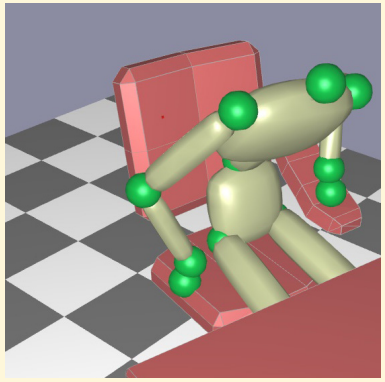
Generating a contact for one limb



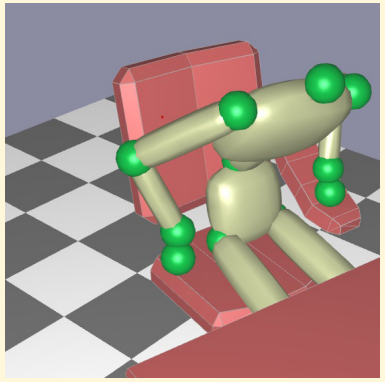
Sampling limb candidates



Proximity query



Keep best candidate



Projection

What is the “best” candidate ?

Characterize a contact with the force actuation profile

Purpose of contact: force exertion

Characterize a contact with the force actuation profile

Purpose of contact: force exertion

⇒ Identify kinematically actuation capabilities?

✔ The limb jacobian gives us the force ellipsoid

Characterize a contact with the force actuation profile

Purpose of contact: force exertion

⇒ Identify kinematically actuation capabilities?

✔ The limb jacobian gives us the force ellipsoid

Characterize a contact with the force actuation profile

Purpose of contact: force exertion

⇒ Identify kinematically actuation capabilities?

✔ The limb jacobian gives us the force ellipsoid

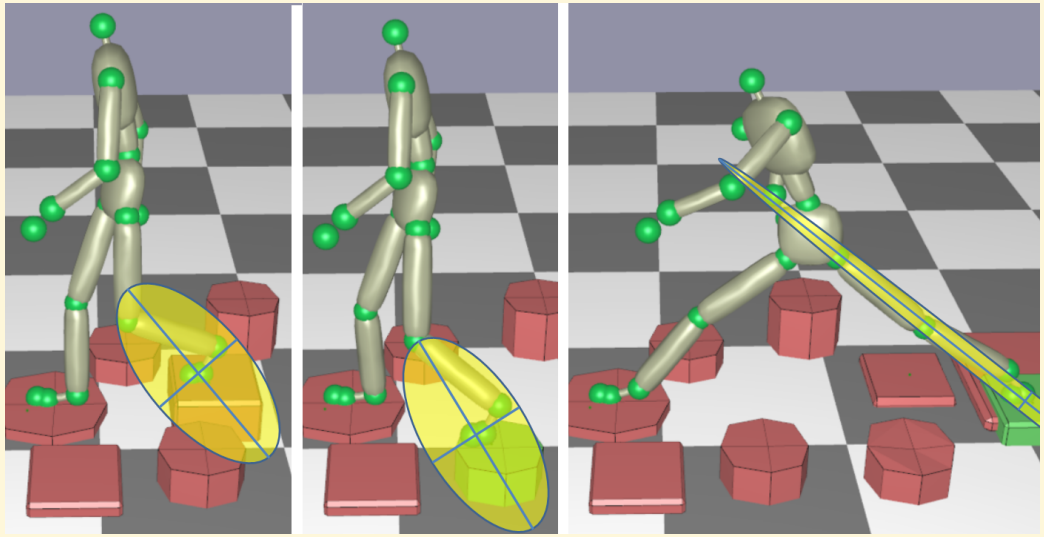


Characterize a contact with the force actuation profile

Purpose of contact: force exertion

⇒ Identify kinematically actuation capabilities?

✔ The limb jacobian gives us the force ellipsoid



Characterize a contact with the force actuation profile

Purpose of contact: force exertion

⇒ Identify kinematically actuation capabilities?

✔ The limb jacobian gives us the force ellipsoid

We also consider ''dynamic'' equilibrium

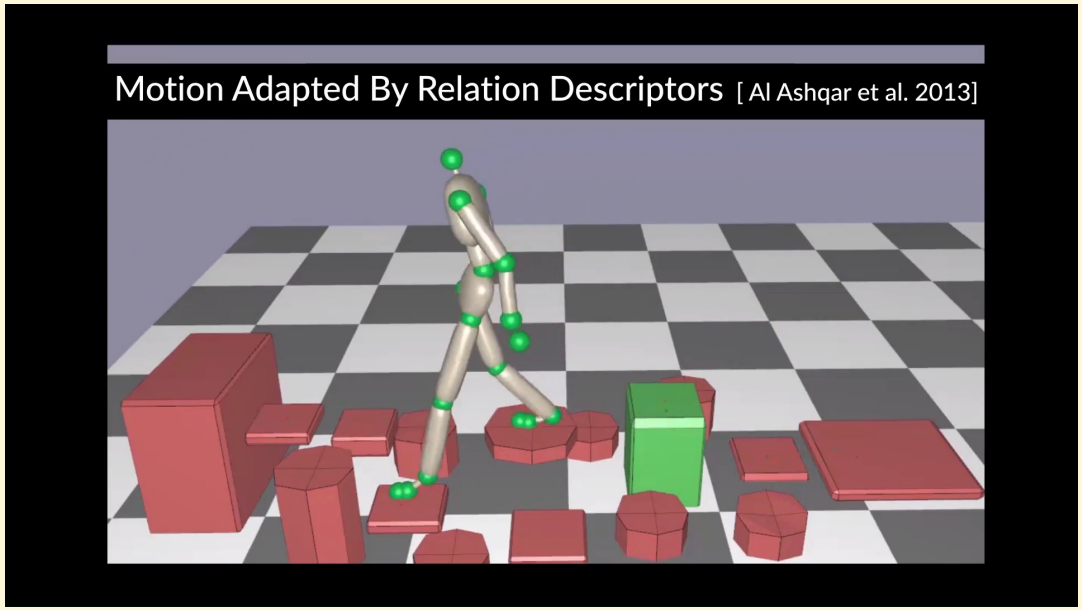
✔ Can be verified with an efficient LP

Motion adaptation scheme

- Upon deformation: relationship descriptors

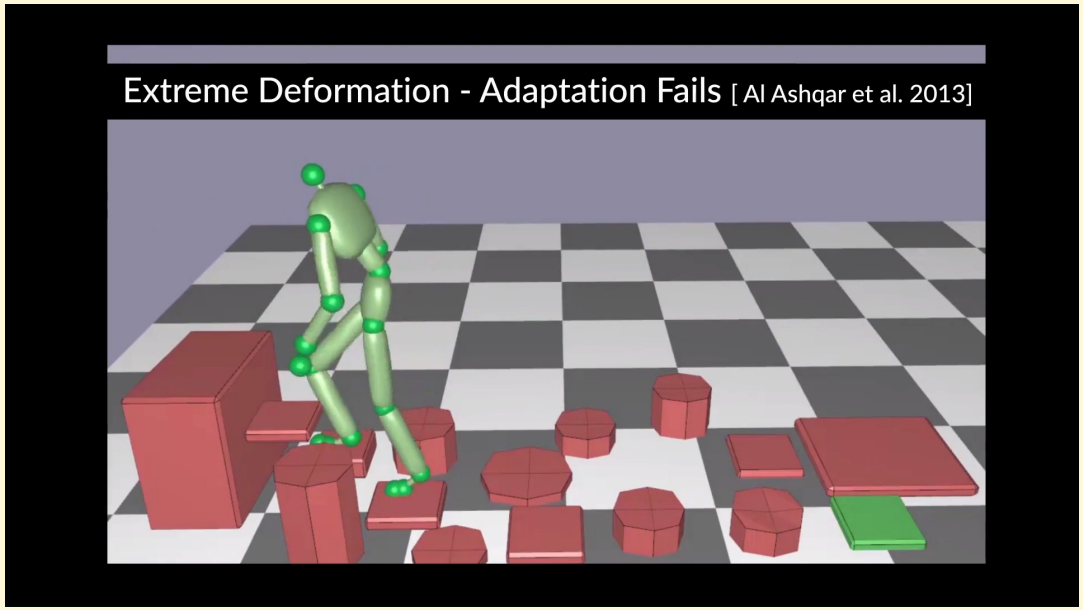
Motion adaptation scheme

- Upon deformation: relationship descriptors



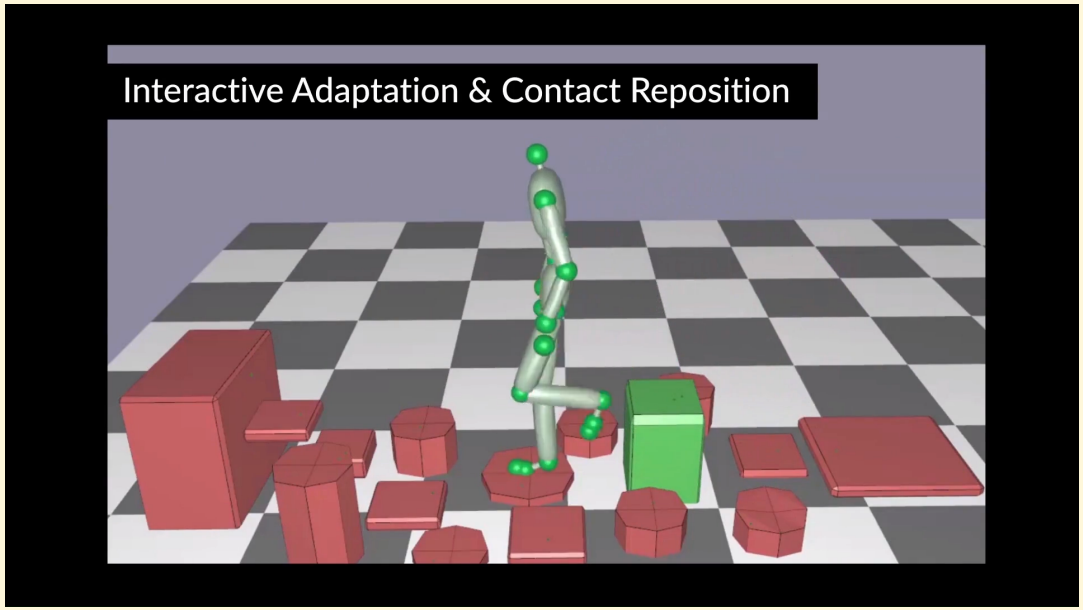
Motion adaptation scheme

- Upon deformation: relationship descriptors
- Detect contact failure (joint limits / collision)



Motion adaptation scheme

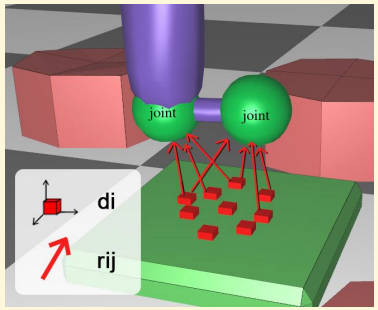
- Upon deformation: relationship descriptors
- Detect contact failure (joint limits / collision)
- Re-position contact ($\Leftrightarrow$ Force Actuation Profile)



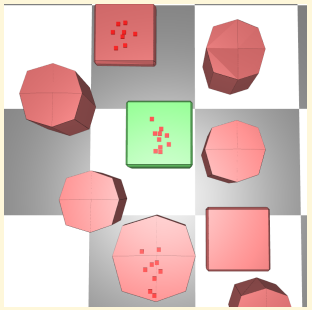
Preserving motion continuity by updating descriptors

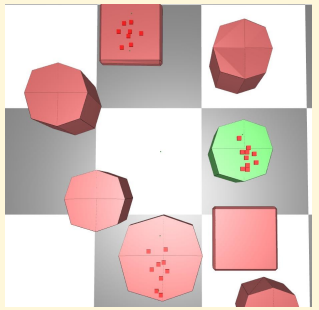
Solve a non linear optimization problem:

- adjust **di** such that weighted sum of vectors point to new joint position
- **rij** invariant

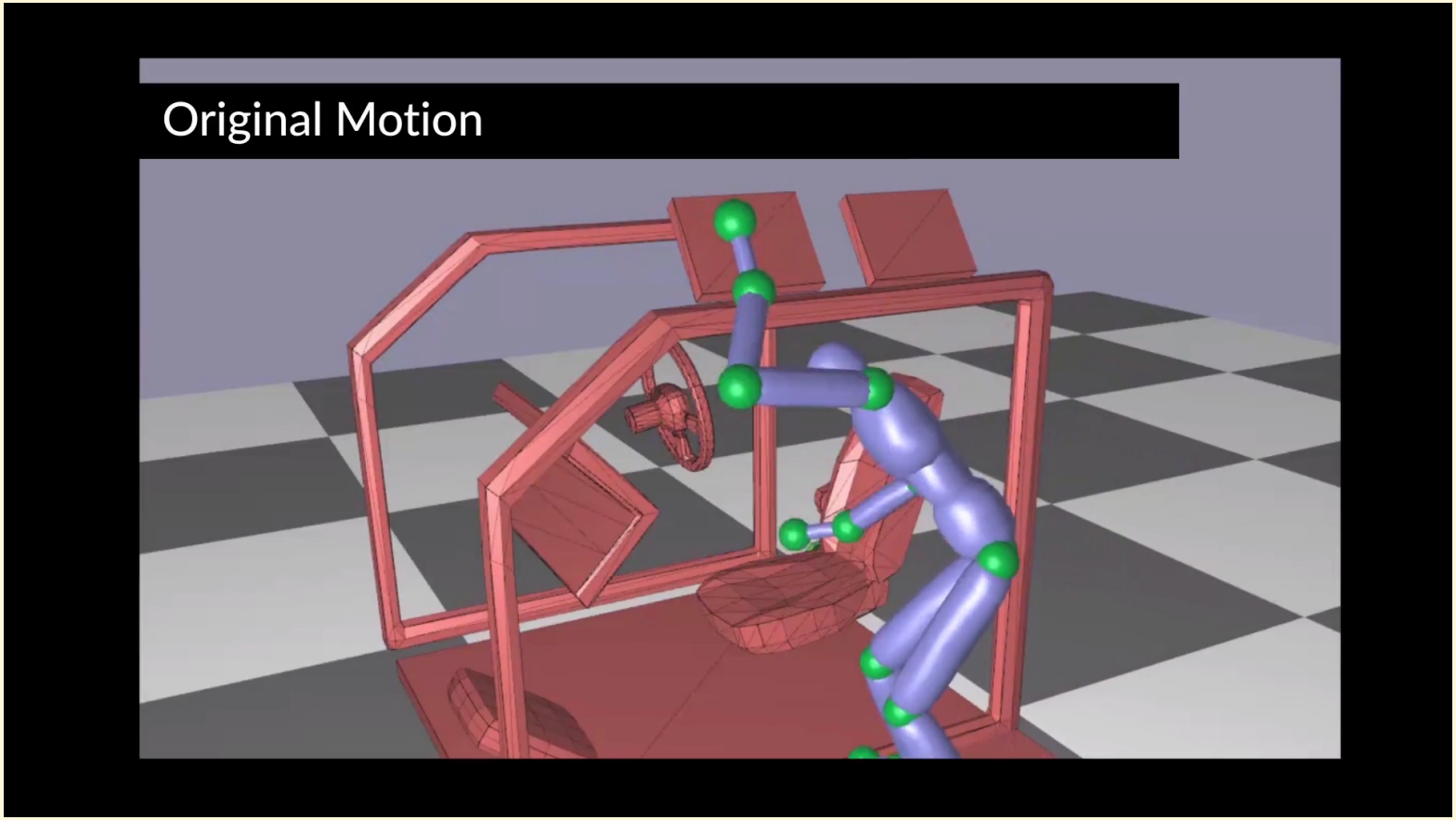


Edited motion can in turn be edited





Additional results: car ingress



Car ingress: removing the car top, continuity preserved

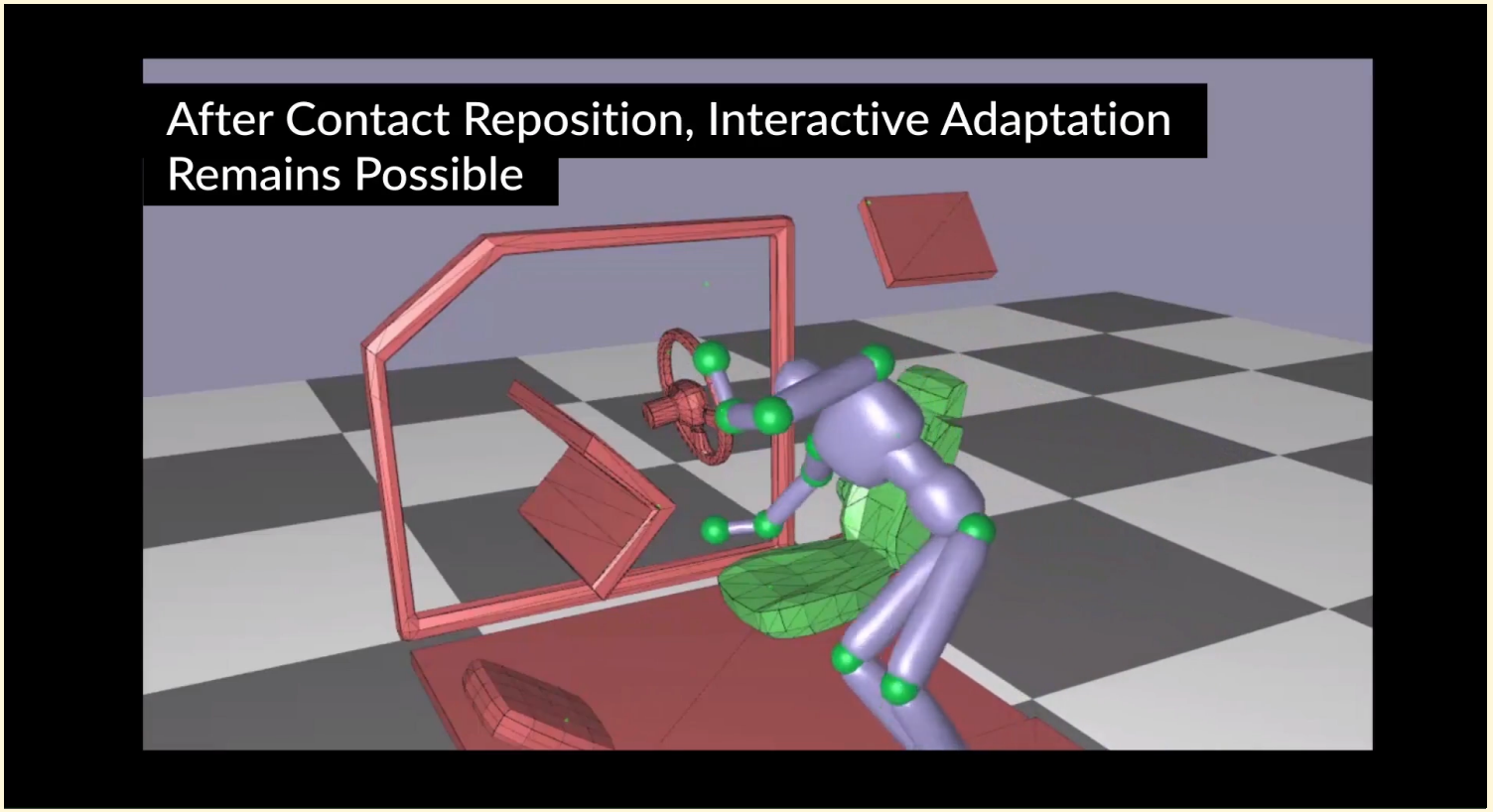
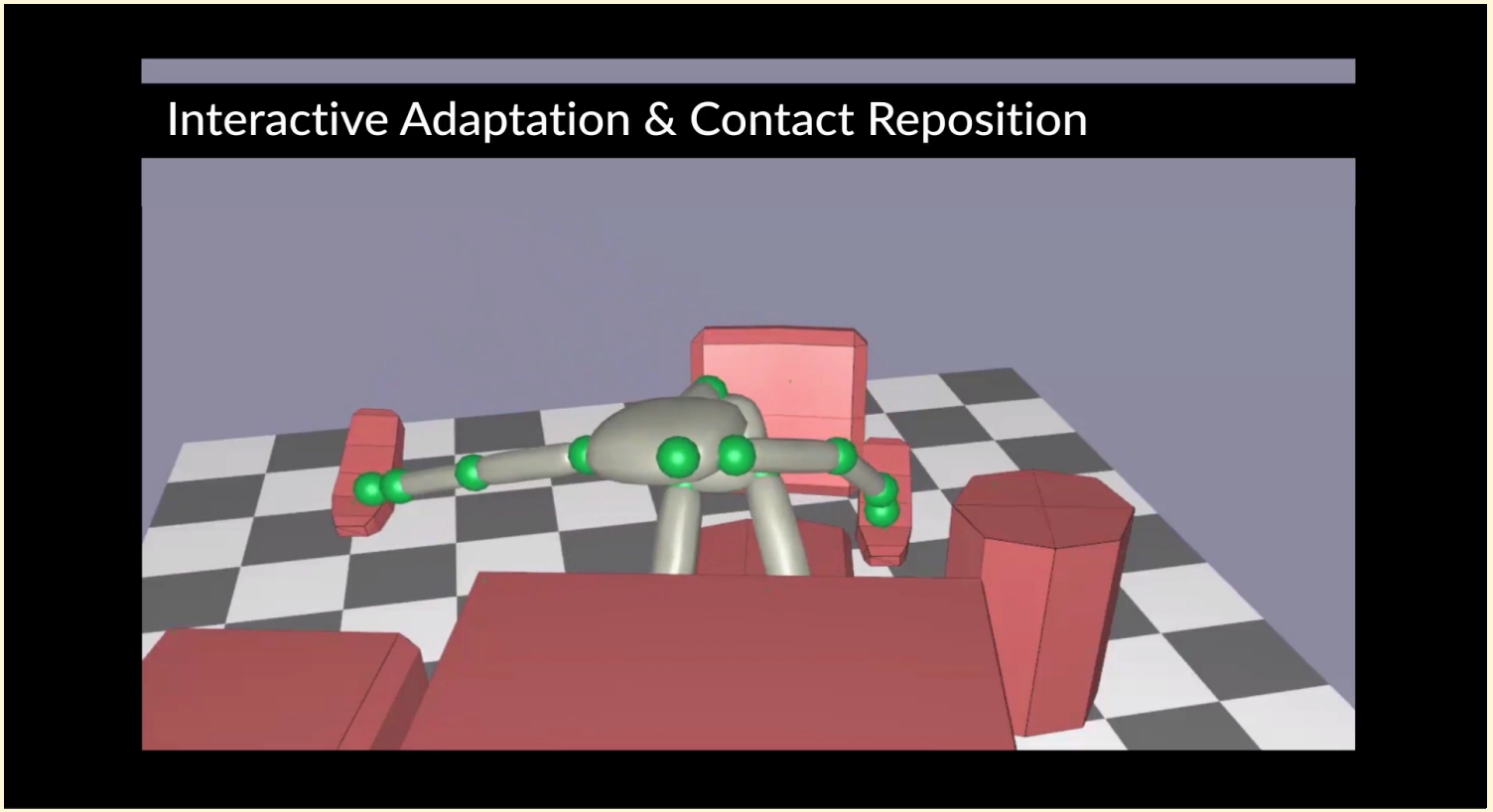


Table example



Still, failures can occur

Illustration Case

Desk Interaction Example

Special Case : Extreme Deformation, Violations + Contact Reposition Fail

Solution : Replan Root Trajectory + Reposition Contacts

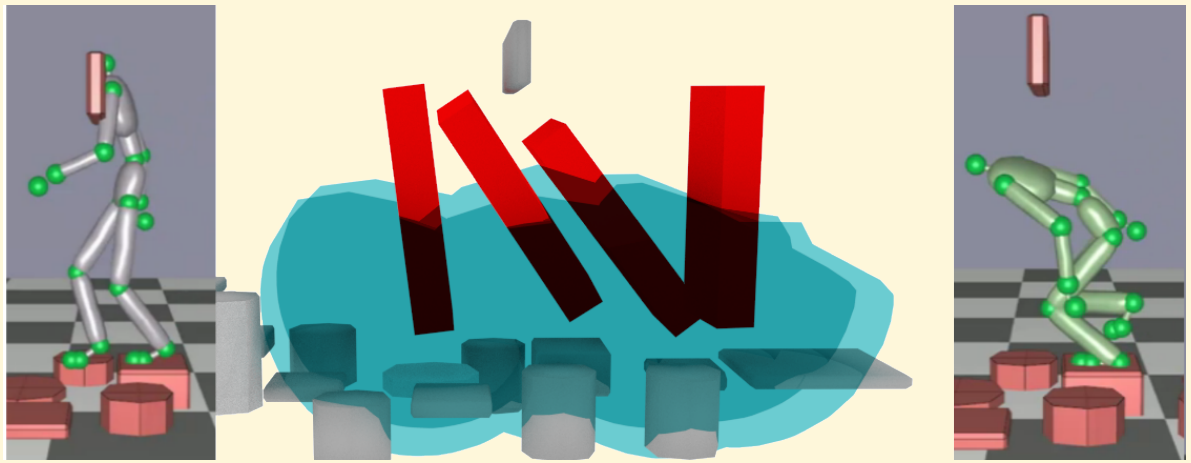
Still, failures can occur

Local replanning of root trajectory:

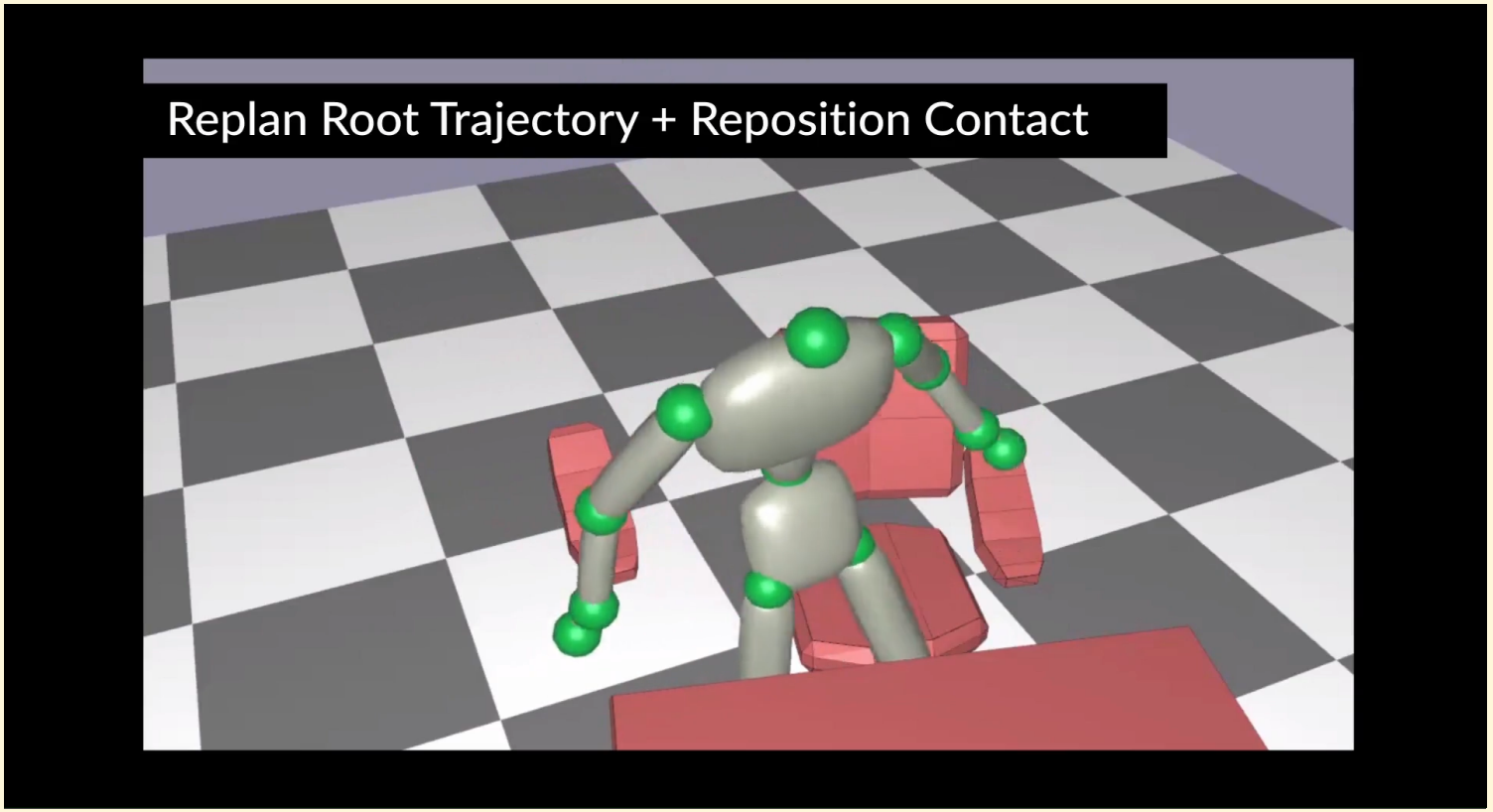
Sampling based RRT biased to preserve valid contacts

Planning in low dimensional space considering:

- Reachable workspace (blue)
- Trunk bounding box (red)



Root trajectory adaptaion



Conclusions on a contact re-positioning framework

Interactive motion editing handling large geometry deformation

Sequential adaptation for minimum deformation

Immediate application for 3D authoring tools

Video games: how to adapt to sudden changes?

Our code is partially open source

Library for:

- Motion planning
- Contact generation
- Equilibrium

part of the humanoid path planner project:

https://github.com/humanoid-path-planner/hpp-rbprm

Acknowledgements

Thanks to the anonymous reviewers

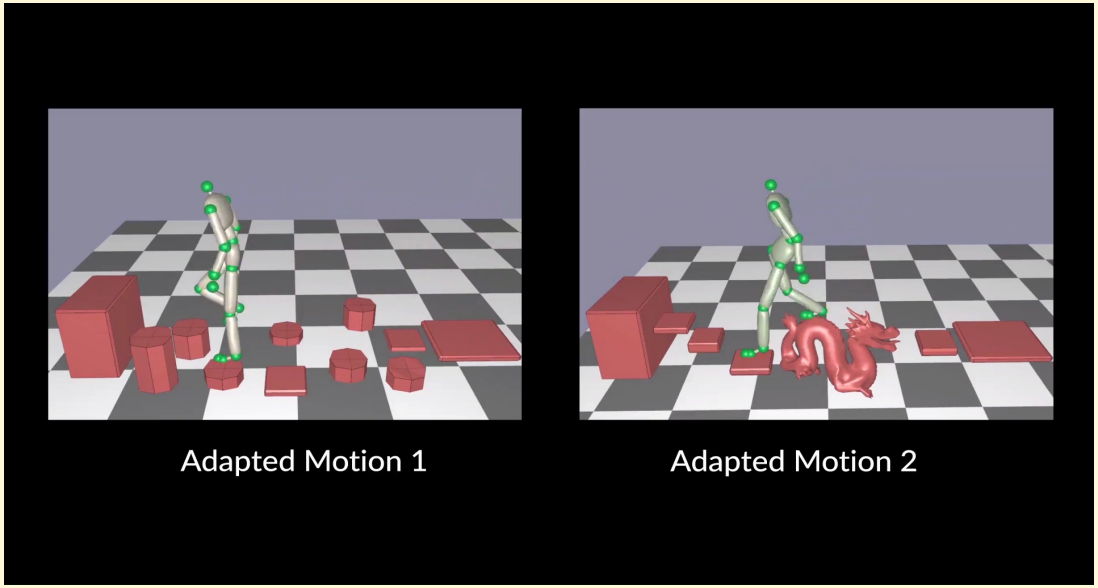
who greatly helped improving the paper

FP7 projects:

- Euroc
- TOMSY

ANR project Entracte

Thanks!



Open source code partially available on github:

Motion planning, contact generation, equilibrium library:

github.com/humanoid-path-planner/ (http://cpc.cx/fxS)

References on stevetonneau.fr